STARS

2013

Modeling Learner Mood In Realtime Through Biosensors For Intelligent Tutoring Improvements

Keith Brawner
University of Central Florida

MODELING LEARNER MOOD IN REALTIME THROUGH BIOSENSORS FOR
INTELLIGENT TUTORING IMPROVEMENTS

by

KEITH W. BRAWNER
B. S. University of Central Florida, 2008
M. S. University of Central Florida, 2010

A dissertation offered in partial fulfillment of the requirements
for a degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2013

Major Professor: Avelino Gonzalez

ABSTRACT

Computer-based instructors, just like their human counterparts, should monitor the emotional and cognitive states of their students in order to adapt instructional technique. Doing so requires a model of student state to be available at run time, but this has historically been difficult. Because people are different, generalized models have not been able to be validated. As a person's cognitive and affective state vary over time of day and seasonally, individualized models have had differing difficulties. The simultaneous creation and execution of an individualized model, in real time, represents the last option for modeling such cognitive and affective states. This dissertation presents and evaluates four differing techniques for the creation of cognitive and affective models that are created on-line and in real time for each individual user as alternatives to generalized models. Each of these techniques involves making predictions and modifications to the model in real time, addressing the real time datastream problems of infinite length, detection of new concepts, and responding to how concepts change over time. Additionally, with the knowledge that a user is physically present, this work investigates the contribution that the occasional direct user query can add to the overall quality of such models. The research described in this dissertation finds that the creation of a reasonable quality affective model is possible with an infinitesimal amount of time and without "ground truth" knowledge of the user, which is shown across three different emotional states. Creation of a cognitive model in the same fashion, however, was not possible via direct AI modeling, even with all of the "ground truth" information available, which is shown across four different cognitive states.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION AND BACKGROUND

Human-to-human tutoring on a one-to-one basis by an expert instructor is the most effective form of instruction found to date. In the most famous study of human tutoring (Bloom 1984), an improvement of approximately two letter grades resulted from such one-on-one human tutoring. Tutored learners outperformed 98% of classroom learners in extensive experiments, showing a clear difference between those with and those without tutoring.

Developments in Artificial Intelligence (AI) over the past few decades suggest that computers could provide the equivalent of one-to-one, human-to-human instruction, with the associated educational advantages that it brings. Such a field of study is known as *Computer Based Training* (CBT). In the early days of CBT research, however, computers provided little more than the content provided in the early types of e-books. As the field advanced in lockstep with advances in AI, CBT morphed into an immensely more useful tool. This was enabled by the new ability of computers to provide feedback to learners, judge their understanding, accurately model their learning, and measure their performance in addition to providing underlying knowledge and educational links between content. This functionality has begun to closely approximate human tutoring. CBT has evolved with it into what is now called *Intelligent Tutoring Systems* (ITSs). This line of research has led to the speculation that intelligent tutoring by computers holds the promise of eventually becoming superior to human tutoring, and the preferred

method of instruction for many training needs (Scandura 2011). This forms the basis of the research described in this dissertation.

For intelligent tutoring to perform as successfully as expert human tutors, the actions of the human tutor should be closely studied and emulated. Human tutoring in general, and instructional practices specifically, are dedicated to the skill-based, cognitive and affective outcomes of the learner (Kraiger et al. 1993). While humans are natively able to sense affect and cognition through experience with a lifetime of social interactions, it has been technical challenge for computer systems to detect and classify these states (Woolf 2009b).

Some examples of cognitive states include attention, engagement, confusion drowsiness, and workload, while examples of affective states include anxiety, arousal, boredom, frustration, and stress. It is reasonable to believe that a computer system that is sensitive to these changes in learner states can positively impact learning goals (D'Mello et al. 2007; Graesser et al. 2007; Lepper and Woolverton 2002). It is also reasonable to believe that the instructional approach for a learner who is confused/aroused is different than the approach a learner who is inattentive/frustrated (Lester 2011). While these are reasonable assumptions, the underlying detection and classification of these states is a prerequisite to autonomously supporting differing instructional approaches, and advances in this field have been slow.

The reasons for these difficulties in affective and cognitive classifications are many and varied, as is presented in the second chapter of this dissertation. Briefly, they

stem from the singular cause that learners are different from each other. Generalized and individualized models of human affect have not seen successful transfer into educational practice. Furthermore, the models that *have* been constructed and evaluated take longer to construct than the duration of a typical training session, meaning that the learner has physically left the room prior to the prediction of his/her state becoming available. This renders these models impractical for use in applications where user state assessments are required in real time for instructional strategy selection.

The research described in this dissertation extends the state of the art by creating an emotional model for a learner in real time. It does this through an analysis of the state of the art of ITS research and affective modeling, before looking to artificial intelligence tools and methods that can mitigate these problems. This dissertation is tested on a carefully collected dataset of cognitive and affective sensors that are appropriate for classroom settings. Before continuing with this dissertation, it is appropriate to set the background for this research through a broadly-reaching look at human tutors, CBT, adaptive training, and ITSs. The discussion then moves to focus on a comprehensive review of the cognitive and affective models of learners implemented to date. We begin with a discussion of the background to the works reported here.

1.1.   Background

Although one-to-one human-to-human tutoring from expert tutoring has been shown to be the most *effective* manner of instruction (Bloom 1984), it is not practical for each learner to be singularly instructed by an educational professional. This renders individual instruction unavailable for the vast majority of training needs. The traditional classroom model of one-to-many human-to-human instruction is more *efficient* than one-to-one human-to-human tutoring, as one teacher is able to be shared by several learners. As a hypothetical example, if the state of Florida implemented a one-to-one tutoring mandate for current class sizes, teacher costs would rise significantly (see Table 1 for current class size mandates). Although education could be optimized through one-to-one, human-to-human instruction, the efficiency gains of one-to-many instruction and shared resources would be lost.

Table 1 - Florida class size limits imposed by Florida's Article IX

Grade Group	Maximum Number of Students Allowed in a Core Class by Fall 2010
K-3	18
4-8	22
9-12	25

Computer software, unlike teacher-based instruction, has a "write once, use anywhere" nature (Curtin 1998). While there are associated maintenance and hardware costs with computer instruction, the largest portion of monetary investment in an educational computer system is represented by the initial system and contained instruction; the largest portion of monetary investment in classroom instruction is teacher salary and training.

The incremental cost to provide this computer system-based instruction to additional learners is very low, especially when compared with the costs of providing additional human teachers. This type of computer-based instruction is already dramatically more *efficient* than face-to-face instruction, by between 70% and 90%, depending on the metric used (Woolf 2010). It logically follows that the creators of computer instruction should strive to emulate the *effectiveness* of one-to-one expert human instruction.

A hypothesis on ITSs holds that individualized instruction, as effective as one-on-one human instruction, can be given via computer. This has the potential to be as *effective* as human instruction, and as *efficient* as CBT. However, this has yet to be unequivocally shown via the literature. While ITSs have been shown to be more effective than classroom-based alternatives (Verdú et al. 2008), they have yet to be as *effective* as one-to-one human instruction (Koedinger et al. 1997; Woolf 2009b). These studies are evaluated through the analysis of 'learning gain effect size', so it is useful to include a discussion of how this is calculated, and the historically observed effects.

1.2.   Measuring Learning Gains

The goal of instruction is to increase the amount of knowledge that a learner retains, or the amount of practice the learner is able to perform unassisted. The most common way to measure this type of effectiveness is to use the 'weigh the brain' method of pre- and post-testing. This method consists first of a pre-test, administered to the control and experimental groups. The control group is then exposed to the instruction in the way that is typical for the content, representing the "business as usual" case. The experimental

group, on the other hand, is given an instructional intervention. Typical interventions may include items such as computer training vs. live training, differing time constraints, differing content, differing feedback, or differing training systems. Afterwards, a post-test is given to both groups to determine the relative levels of increase in their mastery of content knowledge. Four important measures are developed from these data: the experimental/control means, and the experimental/control standard deviations. The difference between the mean of the experimental and control groups is the *learning gain effect size* ($(\mu_e - \mu_c)/\sigma_c$). A learning gain of '0' represents that the two methods of instruction were statistically equivalent. Typically, an intervention with a learning gain effect size of 0.25 is considered significant for the Department of Education (Clearinghouse 2008). The study of effect sizes allows the experimenter to remove sensitivity effects of populations (Schulze 2004), and is the most common way to study the differences which are inherently present in training.

One of the long-term conclusions of the study of learning effect size is that deep levels of content comprehension do not typically occur via classroom instruction (Bransford et al. 2000). Different studies have identified different 'worst ways to learn' such as very large class sizes (Cuseo 2007), textbook reading (Zwaan and Singer 2003), and unguided experience (Kirschner et al. 2006). However, a reader must be very careful in the conclusions drawn from educational research. For example, despite smaller class sizes being known as better suited to learning, it is a common misconception that they *guarantee* additional gains in learning; smaller class sizes only *allow* for the possibility of

teachers taking advantage of additional opportunities for instruction and tailoring to the learners' needs (Haddad 1978).

Haddad found that smaller class sizes do allow for content to be tailored to individual learner needs, allowing the instructor to provide more elaborative examples, adapt content difficulty, transition to other content sooner, and additional tailoring of content, all of which are directly correlated with gains in learning. It logically follows that this rule holds true to the smallest possible number: size one. In fact, this has been observed across several studies. Cohen's meta-analysis of novice tutoring has been shown to have an effect size of 0.4, or one half of a letter grade (Cohen 1992), indicating that untrained but knowledgeable instructors providing one-on-one attention are able to produce significant gains in learning. As discussed earlier, one-on-one human-to-human tutoring, from an expert tutor, holds the promise of two effect sizes (Bloom 1984), as shown in Figure 1.
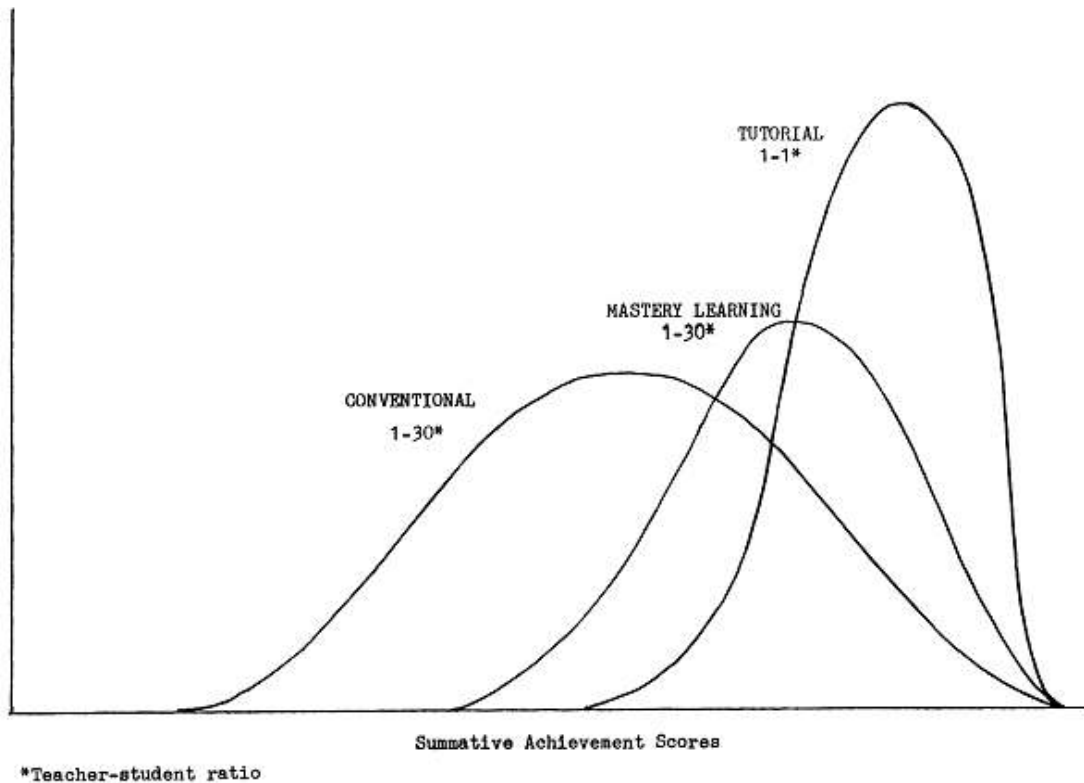
Figure 1 – Achievement distribution for learners under conventional, mastery learning, and tutorial instruction. Original figure (Bloom 1984).

Table 2 – Types of human-to-human learning gains

Type	Learning Gain	Citation
Conventional	0 (Baseline)	N/A
Novice Human Tutor	.4	(Cohen 1992)
Mastery-Based Instruction	1.0	(Bloom 1984; Verdú et al. 2008)
Expert Human Tutor	2.0	(Bloom 1984; Fletcher 2011)

1.3.   Early Computer-Based & Adaptive Training

The terms *computer-based training*, *computer adaptive training*, and *intelligent tutoring* represent the evolution of the practice of using computers for training purposes. Traditional *Computer-Based Training* provides no feedback or interactive elements, and is the modern equivalent of reading an e-book. *Computer adaptive training* consists of

the methods for scaling content difficulty to the user, usually based upon the previously observed performance data. Computer adaptive training leaves the problems of motivation, attention, engagement, and such others up to the user, rather than managing them through the training system. *Intelligent tutoring* currently encompasses all the above terms plus a wide variety of additional actions to be discussed in the human tutoring section 1.4.1.

The origins of the idea using of computers for instruction are nearly as old as the concept of a computer itself. Work in this area of Computer Assisted Instruction (CAI) begins with psychologist B. F. Skinner and his 'linear programs' (Skinner 1954). A 'linear program' would present content to the learner in a prescribed, static, order. After a certain amount of content presentation time, which varied from system to system, the instructional program would come to an impasse that required learner action, with the intent of forcing the learner to think deeply about the problem. After the learner action was complete (correctly or incorrectly), the program would present the correct answer to the learner and move on to the next series of content objects (Skinner 1954; Skinner 1958).

Skinner argued for the idea that the actual response of the learner, if correctly instructed, would always be correct (Skinner 1954; Skinner 1958). Given that learners' answers were always correct, the program could proceed to the instruction of the next content. It was Skinner's belief that negative, or corrective, feedback was detrimental to the learning process. In Skinner's systems, all learners were presented the same content

regardless of background, views, motivation, emotional impact, skills, ability, etc., and the actual learner responses were ignored. An experienced educational professional will note that this is not generally aligned with modern best practices, as is shown in later work (Heift 2004; Lyster and Ranta 1997; Schachter 1991).

Research involving adapting the training to learner responses followed a short time afterwards (Crowder 1959). In this approach, a different frame of instruction would be selected based upon the answer given in the previous frame. This allows for the material to be customized to the learner's needs, and came to be known as a *branching program*. It represented the first instances of computer-based individual tailoring of instruction. At the time, this type of research was centered on the teaching of well-defined concepts and domains. A natural extension of this research was the *generation of content*, rather than loading content from memory, for learner practice. This was only possible with well-constrained problems and assessments. These types of educational content creating systems became known as *generative systems*.

In the 1960s, the generative system technique of content creation provided drastically reduced memory usage, allowing for more content to be presented to the learner. Each time that a content element would be selected and loaded from a previous iteration of the system, it could be generated dynamically. This method experienced reasonable success through the late 1960s and early 1970s. (Suppes 1966; Uhr 1969; Woods and Hartley 1971). This, in turn, gave way to the early versions of Intelligent Tutoring Systems of the 1980s, as is shown in rough historical context in Figure 2.

Linear Programs 1950s	→	Branching Programs 1960s	→	Generative CAI Late 1960s	→	ITSs 1980s
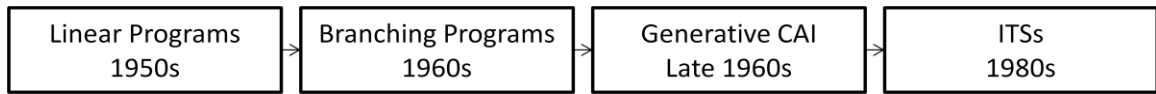
Figure 2 –Computer Assisted Instruction To Intelligent Tutoring System Timeline (Nwana 1990)

The idea of intelligent tutoring is not new. The concept of replacing the function of teachers as content presenters with computer services is presented fairly early in the literature, dating back to 1973 (Hartley and Sleeman 1973). An early idea about these types of systems was that they would be used in a different manner than face-to-face instruction, such as a study aid or supplemental homework assignment. While computer teachers have many advantages (eg. can teach many different subjects, during all hours of the day, with little downtime or preparation, and in geographical areas where a teacher cannot serve), ITSs have traditionally been poorer in function than their human counterparts. This is an instructional tradeoff between the cost and availability advantages of computer instruction and the higher effectiveness of human instruction (VanLehn et al. 2005). Just as a tradeoff is made in order to teach many learners in a classroom, rather than one-on-one, a tradeoff can be made to teach via computer, rather than via human.

The latest advances in ITS deal with systems that are sensitive to the emotional and cognitive needs of the learner in order to implement instructional strategies accordingly (Banda and Robinson 2011; Blanchard et al. 2009; Dragon et al. 2008; Lester 2011; Picard 2006; Robison et al. 2009; Woolf 2009a; Woolf et al. 2009; Woolf 2009b). The intelligent tutoring term represents computer instruction in the way that tutors instruct. To develop an effective intelligent tutoring system, one must first look for

inspiration from the effective human tutors, then analyze the types of systems that support these needs, and finally work to fill the significant research gaps that exist in modeling learners in order to provide this level of feedback. This dissertation seeks to exactly address the problems in modeling learners.

1.4. Tutoring

1.4.1. Human Tutoring

Several strategies exist for providing education, including:

- Experiential learning, example: fixing a flat (without prior experience)

- Activity-based learning, example: reading a book on mathematics

- Classroom-based learning, example: biology lecture

- Tutored learning, example: one-on-one physics problem solving

Education attempts to follow a logical cost-benefit curve, but while the absolute effectiveness of the above strategies are unknown, but the relative effectiveness of each of these strategies *is* known. These strategies are listed in increasing order of effectiveness. The primary reason for these educational decisions is cost, which also increases down the list. More effective forms of instruction cost more.

1.4.2. Different Types of Instructional Intervention

Tutored learning occurs though a series of instructional decisions, making it helpful to discuss a few of the items of human-to-human instruction that have direct computer-based instructional implementations. The most common manner of teaching revolves

around the *sounding*, or presentation of information, and *listening* for echos, or the assessment of knowledge based on previously presented information. Tutoring acts may expand this model in one or more ways, such as:

- Short feedback: on-the-spot elucidation or reiteration of a particular aspect of previous instruction

- Pump: An attempt to elicit information, such as, for example: "Why do you think apples fall?"

- Prompt: A direct request for specific concept, such as, for example: "In what state is Random Access Memory (RAM) in when a computer is off?"

- Elaboration: the expansion of a previous answer, such as, for example: "Yes, the obfuscation of underlying mortgage assets was part of the subprime mortgage crisis, but the influence of a boom/bust cycle, homeowner speculation, high-risk banking practices, mortgage fraud, and Governmental policy cannot be ignored."

- Correction: informing the learner of a better answer, such as, for example: "Not quite right. Ted's gift to his supervisor constitutes an ethical breach because it exceeds $10".

- Hint: an indication of the correct answer, such as, for example: "this activity occurs underwater"

- Curriculum Script: The ordered segment of instruction, such as, for example: the Earth, then Sun, then Solar System, then other planets method of teaching astronomy

Each of these variations on the traditional teaching model represents a way in which the instructor may interact with the learner. These tactics have rough equivalents in a computer system. Each of these tactics is one or more ways to contribute to an overall strategy of instruction.

1.4.3. *Tutoring Strategies For Humans And Computers*

The most commonly held belief is that expert human tutors adopt several categories of strategic instruction in order to effectively teach (Holland and Gallagher 2006). This classification of learning categories has guided ITS research into systems that operate primarily in one of these areas. These categories of strategic instruction include:

- Tutor-centric instruction

- Learner-centric instruction

- Interaction-centric instruction

1.4.3.1. TUTOR-CENTRIC INSTRUCTION

The tutor-centric category of instruction can be broken down into a number of strategies and tactics. This can be performed through watching learner actions, monitoring, and modeling the knowledge of the learner as he/she interacts with the system. Knowledge monitoring, in either human or computer tutoring, occurs through knowledge demonstration activities of the learner. This monitoring of knowledge can result in the accurate assessment of the learner knowledge and lead to the accurate tailoring of difficulty level to the individual (Ingleton 2000). The second phase of tutor-centric research is focused on the idea that expert human tutor strategies can be emulated in

computer systems (Wikipedia 2012; Woolf 2009b). The third phase of this strategy is to monitor and manipulate the learner's affect and motivation to learn (James 1884; Lepper et al. 1993). Furthermore, these can be combined in order to identify various types of tutor activities that result in learning gain (Hu et al. 2009), examples of which are detailed later in this section.

1.4.3.2. LEARNER-CENTRIC INSTRUCTION

D'Mello et. al (2010) contend that the learner-centric hypothesis "contains the idea that learners are active participants in the construction of their own knowledge, rather than being mere information receptacles". One of the components in the learner-centric research thrust is that the individual self-regulates his/her own learning. Another component is that the learner's self-efficacy and motivation are high, which gives the tutor full responsibility for the facilitation of knowledge transition from content repository to stored knowledge. This hypothesis is measured through the traditional effect size measurement detailed earlier.

To further explain learner-centric learning, a case study of physics instruction was conducted by Chi (1996). In this study, the 'model' of instruction consisted of:

1. The tutor asking an initiating question

2. Learner providing a preliminary answer

3. Tutor feedback on the answer (corrective feedback, didactic explanations, and suggestive feedback)

4. Tutor scaffolding, taking multiple turns (Graesser et al. 1995) (providing outlines, recommended documents, storyboards, task modeling, giving advice)

5. Tutor assessment of understanding

Although the tutor typically pursued a specific plan of action (ie. that the learner will eventually be able to diagram forces), the opportunities for the learning occur through the interactions. As an example, Chi states: concepts numbered 1, 2, and 9 were learned through hinting, question exchange, and explicit instruction, respectively. "Thus, the tutee learned not from the tutor's instructional skills such as diagnosing misconceived knowledge or giving didactic explanations, but rather, from interactions with the tutor…" (Chi 1996). This is a prime example of learner-centric instruction.

1.4.3.3.   INTERACTION-CENTRIC INSTRUCTION

The interaction-centric hypothesis draws from the idea that interactions between the learner and the instructor, or between the learner and other learners, are the important component of learning. Research in this area additionally focuses on the social learning concept that states that learners frequently learn more from each other than from the instructor (Kapoor and Picard 2005a). Social learning and collaborative learning are closely related, and have analogous comparisons to traditional classroom learning, such as the activities of asking a question in class and forming a study group (Soller 2001). Wiley and Bailey show that collaborative learning in the internet reading domain is more effective than the absence of it, providing evidence for collaborative interaction as a learning method (Wiley and Bailey 2006). Other forms of interaction-centric learning date back to the earliest forms of learning, including the Socratic Method (asking

questions in order to stimulate critical thinking) and reciprocal teaching (student-to-student dialogue-based instruction) (Palinscar and Brown 1984).

Another example of researchers pioneering interaction-centric intelligent tutoring can be found in the ASSESSment system (Feng et al. 2010). The ASSESSment system presents a large-scale problem to the learner that must be decomposed into its parts. Each of these problems has a series of well-defined steps that the learner must complete. If the learner fails on any given part, then they may ask for a hint, with varying levels of hinting. As the learner interacts and asks for hints, the system develops a repository of learner knowledge through the correct/incorrect answers, and information that required hints. This process simultaneously allows the learner to practice the skills being developed and the system to accurately measure his/her knowledge. The learner is able to advance learning on poorly mastered concepts, while still progressing through the problem-sequenced steps.

The ASSESSment system models the learner through the series of exchanges between the student and system (Feng et al. 2010). This is similar to adaptive, or intelligent, testing (Conejo et al. 2004). Each of the interactions between the student and system is taken as evidence of current learner understanding. This drives the selection of the next segment of information presented to the learner. Therefore, as the learner interacts with the system, the system is continuously testing learner ability.

Another example of interaction-centric learning can be found in a similar system, although developed without hints. Computer Adaptive Tests (CATs) are administered by

a computer and ask questions of progressively increasing difficulty. If the learner struggles or succeeds with the current test question, the questions become easier or more difficult, respectively. In this way, the learner's ability can be precisely calculated. The interaction-centric ASSESSment system is an outgrowth of the field in this direction. CATs are currently used in the modern Graduate Records Exam (Van Der Linden and Glas 2000), but only after having been widely reported in the literature (Weiss and Kingsbury 1984).

1.4.3.4. STRATEGIC NOTE

The three types of systemic instructional strategies presented are all related to the selection of appropriate actions to take. Should the tutor model the necessary knowledge to select the next items to teach, respond directly to the learner, or allow the learner to acquire knowledge through interactions with the system or others? Regardless of the choice of instructional application, each tutor-selected action is taken with respect to the learner, his/her learning goals and observable state. In order to provide feedback, giving a pump/prompt/elaboration/correction/hint, or adjust the script of the curriculum, there must be an underlying learner assessment that consists of more than simple competency. While humans are able to create complex, multi-variable, models of a learner state without particular effort, it is a technically challenging task for computer-based system. We discuss this in the following chapters and sections.

1.5.   Intelligent Tutoring

The topic of this dissertation relates to the use of learner modeling within an intelligent tutoring system. Both the use and the novelty of the work contained in this dissertation rely heavily on the advances in ITS research, as an affective learner model is not useful without the underpinning tutoring capability. As such, a brief review of the concepts and functions of intelligent tutoring provides background to the direction of the current work.

An intelligent tutor was described early as a "computer program that [is] designed to incorporate techniques from the AI community in order to provide tutors which know *what* they teach, *who* they teach, and *how* to teach it" (Pajares and Miller 1994). Naturally, the earliest ITSs, just like the earliest forms of AI, addressed well-defined problems with crisp, clear, rules that govern their behavior. The below list of ITS systems and domains serve as an example of the systems which practice this behavior.

Table 3 - A comprehensive list of ITSs cerca 1990, (Pajares and Miller 1994). References available in original work.

ITS	Domain	Reference (date)
ACE/PSM	NMR Spectra Interpretation	Sleeman (1975)
ATDSE	Basic Subtraction	Attisha & Yazdani (1983)
ARITHMEKIT	Basic Subtraction	Brown (1983)
ALGEBRALAND	Algebraic Proofs	Brown (1983)
BIP-I/BIP-II	Basic Programming	Barr et al. (1976)
BLOCKS Tutor	Troubleshooting in a BLOCKS World	Brown & Burton (1978b)
BRIDGE	Programming	Burton (1982)
BUGGY	Basic Subtraction	Brown & Burton (1978a)
DEBUGGY	Basic Subtraction	Burton (1982)
EDSMB	Basic Multiplication	Attisha & Yazdani (1984)
EUROHELP	UNIX Mail	Breuker (1987)
EXCHECK	Basic Logic	Blaine (1982)
FGA	Basic French Grammar	Barchan et al. (1986)

ITS	Domain	Reference (date)
FITS	Basic Fractions Addition	Nwana (1990)
FLOW Tutor	FLOW Computer Language	Genter (1977)
GEOMETRY Tutor	Geometry Proofs	Anderson et al. (1985a)
GERMAN Tutor	Basic German	Weischedel et al. (1978)
GUIDON I/II	Basic Medical Diagnosis	Clancey (1987)
INTEGRATION Tutor	Basic Integral Calculus	Kimball (1982)
LISP Tutor	Lisp Programming	Anderson and Reiser (1985)
LMS	Basic Algebra	Sleeman and Smith (1981)
MACSSYMA Advisor	Use of MACSYMA	Genesereth (1982)
MALT	Basic Machine Language Programming	Koffman & Blount (1975)
MEO-Tutor	Basic Pascal Programming	Woolf and McDonald (1984)
METEOROLOGY ITS	Basic Meteorology	Brown et al. (1973)
NEOMYCIN	Medical Diagnosis	Clancey & Letsinger (1981)
PIXIE	Basic Algebra	Sleeman (1987)
PROUST	Pascal Programming	Soloway & Johnson (1984)
QUADRATIC Tutor	Quadratic Equations	O'Shea (1982)
QUEST	Basic Electrics	White & Frederiksen (1985)
SCENT-3 Advisor	List Programming	McCalla et al. (1988)
SCHOLAR	South American Geographical Facts	Carbonell (1970)
SIERRA	Learning Basic Arithmetic Procedures	VanLehn (1987)
SOPHIE I/II/III	Basic Electronic Troubleshooting	Brown et al. (1982)
SPADE	Basic LOGO Programming	Goldstein & Miller (1976)
SPIRIT	Probability Theory	Barzilay (1985)
STEAMER	Marine Steam Propulsion	Hollan et al. (1984)
TALUS	Basic Lisp Programming	Murray (1987)
THEVENIN	Basic Electrical Circuits	Joobbani & Talukdar (1985)
TUTOR	British Highway Code	Davies et al. (1985)
WEST	Basic Arithmetic Skills	Brown & Burton (1978b)
WHY	Basic Meteorology	Collins & Stevens (1982)
WUSOR	Wiley and Bailey	Goldstein (1982)

Table 3 shows the common use of intelligent tutoring technology in the late 1990s from its emergence in the early 1980s from the Computer Aided Instruction (CAI) systems, branching instructional systems, and generative systems, all discussed in earlier sections (section 1.3) of this dissertation. Since 1990, it has become intractable to meaningfully survey every system containing ITS technology. The predominant questions of the 1990s ITS research community were:

1. "Is intelligent tutoring just old wine in a new bottle, or is it a new vintage?" (Ok-choon et al. 1987). This question asks whether the ITS field is simply an outgrowth of educational research into the digital domain, or whether new types of research/instruction are possible.

2. "Is intelligent tutoring really possible?" (Ridgway 1988) This question asks whether a ITS system can ever fully implement the instructional capability of its human counterpart.

As a field that now combines artificial intelligence with cognitive psychology, educational research, psychophysiology, instructional design, knowledge ontology, and other aspects of instruction, it is safe to say that the field has changed significantly since this question was originally posed. This argues for rendering ITS as new vintage (Vandewaetere et al. 2011). Additionally, not only is intelligent tutoring possible, but practical, as multiple systems have been used in numerous studies with beneficial findings. One such example is a 15% increase in learning gains, which meets Department of Education standards (Clearinghouse 2008), in learning from the Pittsburgh Urban Mathematics Project Algebra Tutor, by 470 learners, in a relatively unforgiving

environment (Koedinger et al. 1997). A 100% increase was observed in the same study on tasks which were directly targeted by the tutor. While earlier research strove for learning gains of 0, or "as good as the classroom", modern ITS systems produce an average of one effect size of learning gain, or about one letter grade (Verdú et al. 2008), and currently strive for more. This is an important point to mention: where it is available, *intelligent tutoring outperforms classroom-based learning, at significantly less operational cost*.

1.6.   Reasons for an ITS

The reasons for the creation of ITSs have not changed meaningfully since their inceptions. The primary reasons are for:

- The research of learning theories, processes, and interactions (Anderson 1987; Sottilare et al. 2011a)

- The practical use of an efficient, possibly very effective, teaching system (Mitrovic et al. 2007; Sottilare et al. 2011b)

1.6.1. Research-Purposed Intelligent Tutoring Systems

Initially, the construction of a research testbed system was to provide experimental evidence to researchers on effective methods of instruction, in order to better inform classroom teachers. An ITS is effective in this, as it allows the experimenter to explicitly control the actions of the teaching system. This is different from other educational research, where deviations from the prescribed independent variable occur frequently (Slavin 2002). These deviations can be simple, such as selection of learners for tutoring,

or they can be deliberate, such as tutoring only the learners who were willing to stay for extra time with the system (Mitrovic and Ohlsson 1999). They can also be more complex such as individual tutor or system biases unknowingly correcting for different types of behavior. Simply put, a human instructor cannot reliably follow one path of instructional strategy execution when he/she believes that it will negatively impact a learner. A research-focused computer system can remove the implicit biases present in human instruction, making it useful for educational research.

While the modern research-focused ITSs now concentrate on ITS educational research, these systems are historically successful. Examples of these successes include classical systems such as Anderson's system to study learning theory (Anderson 1987). Other successes include the development of more accurate theories of cognition (Burns and Capps 1988). Research-focused systems are not designed for the purpose of achieving learning gains, and are usually designed by psychologists or educational researchers. However, there is potential for use-focused systems, designed by engineers, for real-world use, to produce measured learning gains.

1.6.2. Use-Focused Systems

The other reason for creation of an ITS is their practical use. ITSs have been shown to be successful in teaching through several metrics (Ridgway 1988). Ridgway (1988) reported a four-to-one time advantage shown over human tutoring. Additional metrics include instructor cost, resource allotment, classroom cost, time on subject, knowledge on subject, challenge presented to the learner, and others (Woolf 2009b). However, the true

metric of success for an ITS is no different than that of any other software system: its use. The use of an ITS indicates that the final user perceives the system to have more value than the alternatives.

1.6.3. *Functions of an ITS*

From the earliest intelligent tutoring system to the latest, all have had to address the fundamental functions of teaching (Beck et al. 1996). These component modules have mostly been agreed upon by the ITS research community (Barr and Feigenbaum 1982; Bonnet 1985; Wenger 1987). Each of these components is discussed in brief detail in order to present where the research presented within this dissertation will fit within a broader research context. They are, in brief:

1. A training system for user interaction (simulation, sequence of video presentations, webpage, etc.), which can present content to them

2. Learner performance assessment

3. Learner trait and performance monitoring

4. Determination/Application of appropriate instructional pedagogical strategies

5. A communication component to share interactions and data with other systems

1.6.3.1. COMMUNICATION

The least scientifically interesting component of an ITS is the module that functions as communication medium to other systems (Nkambou 2010). This is a required function, of course, but it is typically done in a simplistic manner. The most common system to which an ITS communicates to is the Learning Management System (LMS). The LMS

keeps a record of high-level learner performance across learning content in order to recommend additional content, or as a gateway to additional content (Bohl et al. 2002). One example of an LMS is a university's undergraduate prerequisite matrix, which when coupled with the grades of an individual learner, serves this function. Another example of a method of external ITS communication is through the internet to a generative system (Capuano et al. 2000).

1.6.3.2.   DOMAIN CONTENT

Most obviously, any automated teaching system must contain the content that it is to teach. Just as there are several approaches to learning, there are several types of content-based instruction. Many of these have their analogy to classical methods of instruction, but are instead performed within a computer system. The typical forms of instruction are:

- Book / Webpage (Brusilovsky et al. 1998)

- Presentation / Powerpoint (Hu et al. 2009)

- Real World Experience / Virtual World Experience (Shute and Glaser 1991)

  - Note that this is among the worst ways to learn, research in this vein shows very little payoff in learning gain, and the dated citation is reflective of the trend away from this type of instruction, see (Kirschner et al. 2006) for more information

- Demonstration / Guided Exploratory World (Lane et al. 2011)

- Story / Scenario Examples (Rowe et al. 2010b)

The above listed types of systems are designed to tailor the content to the learner. The architectures that support these activities are well designed. They include an engine for the change of pedagogy, and possibly a method for the generation of new content (Patil and Abraham 2010). They may adapt content from assessment of the learning style of the individual (Klašnja-Milićevića et al. 2011), or adapt feedback through asking metacognitive questions (Roll et al. 2011). However, they have not historically performed the same functions of a human tutor sensing and responding to affect.

The other critical component of the domain information is the learner assessment model (Sottilare 2010) that measures learner performance in various tasks. The traditional way to perform these measurements is with a system of rules that identify correct or incorrect interactions with the system, desirable and undesirable behavior, actions, or answers. One type of method for performing this action is through expert modeling (Nwana 1990). Although some systems have used a more complex method of assessment, the use of alternative methods is limited through the time and difficulty of construction coupled with an unknown gain in learning (Conati 2010). Other methods, such as Latent Semantic Analysis (LSA) on free-response-typed answer systems have additionally met with limited success (He et al. 2009). As a practical matter, a system of rules authored by experts, or by an expert and programmer together, in their domain of expertise is still the standard practice.

1.6.3.3.   INSTRUCTIONAL STRATEGY SELECTION

A model of pedagogy is similar to a college major in Education. While this component does not have knowledge of *what* to teach, to *whom* it is being taught, or *which* mistakes are being made, it does have knowledge and processes about *how* to teach. This process is can be directly coupled to the content, as in the case of constraint-based tutoring (Mitrovic et al. 2007). However, modern ITS research is coming to the conclusion that it is better to have a separate model of instructional strategy, as in the case of AutoTutor (Olney et al. 2010), Logic ITA (Lesta and Yacef 2002), and the Generalized Intelligent Framework for Tutoring (Sottilare et al. 2012a). The processes involved here can be as simple as a classification into auditory or visual learners getting visual or auditory content, or other more complex classifications such as information process, perception, reception, or understanding learners who learn best through reflection, demonstration, presentation, and sequencing, respectively (Klašnja-Milićevića et al. 2011).

Commonly applied pedagogical strategies are derived from research on techniques and tactics employed by expert human tutors in a one-on-one learning environment, which were found to improve performance outcomes by roughly 1.0 effect size (Boulay and Luckin 2001; Person and Graesser 2003; VanLehn 2011). To this effect, instructional components are tailored prior to interaction to better suit a user's ability within a given domain, and guidance and adaptation are facilitated in real-time based on monitored system interactions. These functions expand beyond pedagogical approaches implemented in previously developed ITSs that solely use feedback in response to error (Anderson et al. 1987; Mason and Bruning 2001). With this information, an ITS can

focus on the knowledge components associated with a diagnosed deficiency. However, a model of pedagogy is tied to the inputs it receives from the model of the learner; the output recommendations are only as good as the models that are informing the pedagogy.

1.6.3.4. LEARNER MODEL

The learner model, which is the area of ITSs of most interest to this dissertation, has the purpose of tracking variables that can assist in teaching the learner. The most simplistic learner models track only his/her performance. However, in an ongoing push towards highly adaptable and individualized training (Army 2011; Woolf 2010), there is a demonstrated desire to assess the cognitive and affective states of the individuals in order to tailor training. The purpose of this model is to inform an instructional strategy engine about the learner, for the purpose of making an instructional decision. (Beck et al. 1996) said it best with the following statement: "Since the purpose of the learner model is to provide data for the pedagogical module of the system, all of the information gathered should be able to be used by the tutor."

The core aspect of student modeling is to provide the student "with the right content at the right time in the right way" (Fischer 2001). These models can be constructed from the learners themselves (Hothi and Hall 1998), or via a computer system (Shute and Psotka 1994). Rather than allow the learners to construct their own model, it is more common to use a computer-constructed learner model from observable data.

There are several traditional user items of interest to modeling. The below list provides a sample of the types of user models that have been applied, with various levels of success. This list indicates that learner modeling research in ITSs is currently active, and provides the groundwork for the affectively- and cognitively-based work to be presented in Chapter 2 - Affective Learner Models:

- Learner models based on performance data:
  - "Buggy", or "Perturbation" models (Brown and VanLehn 1980; Holt et al. 1994)
  - Model-tracing (Neches et al. 1987)
  - Overlay model of understanding (Rickel 1989)
  - Classification-based systems (Charniak 1991)
  - Fuzzy set mistake modeling (Katz et al. 1992)
  - Constraint-based modeling (Ohlsson 1994)
  - Example Tracing, or psuedo-tutors (Hockenberry 2005)
- Learner models based on other data:
  - Affect (D'Mello et al. 2007)
  - Cognition (Corbett 2001; Jaques et al. 2011)
  - Demographic information (Arroyo et al. 2006)
  - Motivation (Tvarožek and BIeliková 2009)
  - Cognitive preferences (Navarro et al. 2006)
  - Learning Style (Cha et al. 2006)
  - Gaming behavior (Cocea et al. 2009)

- o Trust (Hassell 2005)

- o Mood (Carole and Hyokyeong)

- o Experienced emotions (Sidney et al. 2005)

The most common learner models are those based on performance information. This is for the simple reason that it is the element of the learner model that the ITS seeks to optimize. The standard of one effect size of ITS improvement in learning has been achieved through the modeling of performance, but further gain has been infrequently seen (VanLehn 2011). It is now becoming clear that new forms of modeling are required in order to achieve the second standard deviation of improvement currently observed in human tutoring, and has been highlighted as a challenge in intelligent tutoring (Brawner et al. 2011; Woolf 2009b).

1.6.4. Current Challenges in Intelligent Tutoring

The ITS research field is multi-facetted and multi-disciplinary field. It ranges from computer science/engineering to cognitive psychology, to learning science, to educational practice. Each of these consists of multiple subfields, such as the computer science areas of ontological management, affective computing, artificial intelligence, and computer networks. It can be difficult to fully grasp the complexities of the interactions. As such, in 2009, a federally-funded report was commissioned by the leaders of the various related fields to provide a full picture and direct the future research in this area (Woolf 2010).

This report was published as a short, 80-page book that considers the needs of educational advances for the next 20 years. It was published with an emphasis towards

global educational development as part of discussions with the Global Resources for Online Education (GROE) project. To date, this represents the most comprehensive, forward-looking, long-term, collaborative plan of study that has been published.

In the initial report, the educational challenges are decomposed into several key areas of interest to learning: personalizing education, assessing learning, supporting social learning, diminishing boundaries, developing alternative teaching strategies, enhancing the role of stakeholders, and addressing policy changes. These areas of interest to learning are then distilled to a number of educational technology challenges. The technical, rather than political, challenges in this area said to be user modeling, mobile tools, networking tools, serious games, intelligent environments, educational data mining, and rich interfaces. This research looks at the educational grand challenge of education personalization through the research perspectives of educational data mining for affective user models.

2. AFFECTIVE LEARNER MODELING

2.1. Introduction

The purpose of a learner model is to inform instructional strategies. Learner models may be based on a variety of data sources, such as performance, personality, or trait data. The current learner modeling techniques focus on performance and ignore the emotional and cognitive state of the learner, while human tutors dedicate significant attention to these items (Kim and Baylor 2006). It is logical to believe that a computer tutor should also pay attention to affective state, and the research discussed in this chapter presents various techniques to do so.

This chapter shows the current state of the art of affective learner modeling, with a focus on the current knowledge base. Within the last three years, the research community has discovered that generalized affective models have limited accuracy (Robison et al. 2010), and transfer poorly (Sabourin et al. 2011). Individualized models of affect, while more accurate than their generalized counterparts, are also difficult to transfer to instructional settings (Cooper et al. 2010). Although dramatic increases in accuracy may not necessarily aid in instruction, dynamic modeling methods can increase in model accuracy (AlZoubi et al. 2009). The analysis of the results of this research presents a research gap which is addressed in Chapter 3.

2.2.   Affect and Learning

Human tutors perform complicated tasks well beyond the scope of content-addition, to areas such as guiding questions, examples, and splices (Person and Graesser 2003). Expert human tutors perform several types of actions, but primarily focus on assessing the emotional and cognitive states of the student in order to improve learning (Kim and Baylor 2006). Studies have shown that human tutors are devoted to the motivation of learners as much to as their cognitive and informational goals (Lepper and Hodell 1989; Woolf 2009b).

Because of the role of affect in the learning process, extensive work has been done to measure the cognitive and emotional states of the students. This has been done by incorporating biological sensors to monitor both behavioral and physiological markers for the purpose of automating learning systems (Ahlstrom and Friedman-Bern 2006; Berka et al. 2007; D'Mello et al. 2007 ; McQuiggan et al. 2007). Because of the link between physiology and psychology (Coles 1989), affective and cognitive states leave traces of their existence within physiological measurements.   These physiological artifacts of affective responses, as a component of emotional and cognitive states during learning, are addressed in depth in Section 4.2.

2.3.   Learner Models

Woolf describes *user modeling*, in the previously mentioned roadmap, as a process that identifies and represents learner competencies and learning achievements, including content skills, knowledge about learning, metacognitive awareness, and affective

characteristics (Woolf 2010). The basic notion that drives the creation of learner models is that additional learner-specific information can be leveraged for clues or recommendations for appropriate actions to take. However, there is no clear research on the best type of model to construct, or the desired level of detail contained within it. Examples of learner model creation methods include production rules, buggy models, example tracing, Bayesian networks, expert overlays of learner performance, and other AI methods to be discussed in this chapter. The research interest in learner models has been primarily performance-based, and includes models of tasks, subtasks, behaviors, skills, or interactions with the tutoring system. While the impact of a specific method of model construction is still under investigation, it is agreed that the creation of these models can be a time-intensive process, as shown later in this section.

One of the earliest systems to model the performance of a learner is a rule-based system (Anderson 1987). Production rules, one of the early forms of AI decision making, composing such systems, match an input to an output. This output of a rule may perform as an input to another rule. In rule-based systems, the rules can grow in complexity and number as more rules are created. This allows for the creation of highly specified detail within a model, but rule-based systems are traditionally labor-intensive to construct. Small to medium rule-based systems are heavily used, but larger ones tend to be ineffective and time consuming to construct, as shown in the Table 4 summary after discussion of other types of systems.

A "buggy" or "perturbation" model is able to assess performance based upon a group of student actions, which represent an underlying cause. An underlying type of model for this field assumes that there is a "royal road" or one path for the learner to take in order to obtain the desired result (correct answer, completed course, etc.). The actions that learner takes may differ from this road, because of a misconception, lack of underlying knowledge, or accident. The mission of a buggy model is to assess this deviation to determine the underlying cause. The creation of buggy models is also time consuming, as it requires a model for all possible mistakes that a learner can make.

Constraint-based models are a combination of the buggy idea of modeling all possible causes of error and the production rule idea of creating general rules to violate. These models have not historically required less time to create, as shown in Table 4, but allow for varying levels of detail. This method of knowledge monitoring has seen widespread use (Mitrovic and Ohlsson 1999; Mitrovic et al. 2006; Ohlsson 1994).

Another form of modeling human performance transfers the knowledge encoding activity from being expert-based to engineer-based. An engineer is able to create an AI-enabled solution, such as a Bayesian network, which can examine data from performance to automatically create a model. While this form of authoring requires relatively little time, it is only able to function at a high level, or with vaguely defined concepts (Arroyo et al. 2006).

Overlay models are a different form of knowledge monitoring. This form of knowledge modeling intends to have an expert overlay, which the learner has

demonstrated a subset. As the student interacts with the system, levels of this expert

model are checked off until a reasonable number of them have been observed and the

student is considered an expert. The PLATO West (Burton and Brown 1976) and

SHERLOCK (Katz et al. 1992) systems are examples of ITSs which have opted for this

technique.

Table 4 - Types of learner models of performance, levels of detail, development time, and learning
effects (Folsom-Kovarik 2012)

Learner Model	Model Detail	Lowest Reported Development Time to Learning Time Ratio	Highest Reported Effect on Learning
Production Rules and model tracing	High: all subtasks	200:1	1.2 (compared to classroom)
Perturbation and buggy models	High: some or all subtasks	No reports	Not significant
Example Tracing	Moderate: some subtasks, not all; sometimes tasks	18:1	0.75, compared to paper homework
Constraint-Based models	Moderate: some or all subtasks or tasks, or a mix	220:1	1.3, compared to briefing and handout
Bayesian networks and other classifiers	Low: tasks or skills	No reports	0.7 compared to learning the tasks with no hints
Overlay models	Low: tasks or skills; or some subtasks	No reports	1.02 compared to on-the-job training

2.4.   Data Mining

It is always highly desirable to automate time-consuming solutions. The use of AI,

machine learning, statistics, and a large volume of transactional data stored across

databases is one such way to attempt automation. The above methods are able to create

links and establish relationships between events in a process called *discovery* (Fayyad et

al. 1996), and is commonly used among internet applications (Madria et al. 1999; Srivastava et al. 2000; Zaïane et al. 1998). Given that the process of model creation can be time consuming, data mining presents an attractive solution. In this section, we describe how data mining has been used to build learner models.

2.5. Mining Data for Learner Models

Automatic creation of learner models through data mining has been applied to performance-based models with reasonable success (Conati 2010). This has been done in areas where there is relatively little transactional data, rather than in affective domains where there is large volumes of data, because of millisecond resolution data collection. Unfortunately, although there is *more* data, this does not necessarily indicate more *meaning*, as it does not come with a label, such as 'happy' or 'bored'. Analyzing large bodies of data to establish patterns was only performed in domains where there was relatively high payoff. The research has primarily focused on performance models, as correct/incorrect actions are easily identifiable. Extensively looking at both transaction and physiological data had been cost-prohibitive until the advent of modern processors, and research in this area was sparse. Although some work in this area was perform in the late 1990s, the field of educational data mining began to take root in the mid 2000s (Romero and Ventura 2007).

Concepts in educational data mining revolve around educators, learners, or administrators. In learner educational data mining, the relevant topics are the prediction, clustering, relationship mining, data distillation, and model discovery (Baker 2010).

Among the topics of prediction, there is learner knowledge, learner actions, and affect. Learner knowledge can be explicitly tested through content presentation or exercises, and it is significantly easier to assess as it relates directly to the learning process. Learner actions are also frequently directly related to the learning process, and can be predicted via traditional methods as an individual learner is likely to do what other, similar, learners have done.

The prediction and classification of *affect* has given researchers difficulty, as the data behind affective models has been very specific to the learner being assessed, and it is difficult to obtain a 'ground truth' of emotional state compared to content comprehension, and it is difficult to establish the meaning of a given set of measurements. Progress in the field of educational data mining for student learners has been slow, with regards to affect, and the required algorithms have been cost-prohibitive to implement. As such, while the field has been historically overlooked, it is fertile ground for this advance. This is the specific subject of the research presented in this dissertation.

2.6. Affective Tutoring

Human tutors respond to the needs of the learner by sensing his/her affective state. A ITS system that performs the same function can be known as an affective ITS. The notion of a computer system that performs similarly in this respect is relatively recent. This idea dates back to 2002, beginning with probabilistic models of emotion through the interaction with learning systems (Conati 2002). In the initial works on the subject, the

use of Bayesian networks was introduced, along with the ideas of extensive post-processing, and limited transfer. Conati sought to model the emotions of the learners who were playing the game "Prime Climb", a game for teaching various aspects of mathematics, such as factoring.

The concept behind this kind of modeling was that emotional representation was a measurement of hidden variables of the learner's cognitive state. This cognitive state caused observable actions, which were detectable via bodily sensors. In theory, a hidden model of emotions can be derived from these data measurements. This early study (Conati 2002), although the first in educational affective computing, encountered implementation and validation problems that still confound the field of affective tutoring. Although the models created in this study were reasonably successful, the validation of emotional modeling work was not performed.

Conati's educational affective computing work was expanded into the creation of more accurate predictive models. One example of gains in the area is the multi-modal detection algorithms of Kapoor and Picard (2005). The hope is that the classification of emotion can lead the system to make instructional decisions that benefit the learner. These decisions may be in the form of hinting, prompting, pumping, providing remedial content (see section 1.4 for more information), or even the manipulation of a virtual character within a teaching environment to provide additional guidance or conversation (Nkambou 2006). For the system to intervene in real time, the implementable models of

emotion must be constructed well enough to make decisions about their recommendations in real time.

The developments and tribulations encountered in the creation of a system that can predict affect are discussed throughout this section. They are logically divided by the authors that conducted the research. The largest and most specifically relevant efforts are discussed. Each of these studies points towards the failure of either generalized models, later used individualized models, static models, or offline-created models. This section ends with the most highly individualized and adaptive models that have been constructed in order to more fully prepare the reader for the technical challenges of this dissertation discussed in Chapter 3.

2.7. AutoTutor

It is impossible to perform a comprehensive survey of affect-sensitive tutoring systems without first considering the foundations upon which they have been constructed. The previous sections of this dissertation assert that Intelligent Tutoring Systems are a relatively well-established domain of computer science and psychology research. However, although the idea of an intelligent tutor dates back to the 1970s with Hartley and Sleeman's work (1973), the truly relevant work began nearly three decades later, with AutoTutor (Wiemer-Hastings et al. 1998).

AutoTutor, in its initial version, was a system intended to teach a wide variety of subjects. This concept is reflected through some of the earlier research improvements, and primarily through the extensive evaluation of human tutors plus the separation of

content from teaching strategies. AutoTutor was initially able to execute dialog moves similar to those that were observed in humans: short feedback, pumps, prompts, elaborations, corrections, and hints (see section 1.4 for more information). The separable components of the underlying system consisted of the curriculum script, language extraction, speech act classification, latent semantic analysis, topic selection, dialog move generation, and a talking head (Graesser et al. 1999).

Since that time, many studies have been performed within the framework that AutoTutor provides, including the variation of teaching tactics (Graesser et al. 2001), the modeling of learner performance (Jackson et al. 2003), the development of lesson authoring tools (Jackson et al. 2003), as well as similar tasks that represent the maturation of a software product from a research prototype. The most relevant things about AutoTutor to this dissertation are the lessons that AutoTutor research has taught about the creation of a learner model and selection of dialog moves (e.g., instructional strategies) from data regarding the learners' affective and cognitive states.

Graesser first began to examine affect shortly after AutoTutor was created, with a workshop geared towards affective responses (Person et al. 1999). However, early questions in dialogue-centric research were: "how emotionally loaded should responses be?", and "when should the system provide purely motivational cues?". The sensing of learner affect, rather than affective agent responses, would not become a research topic for six additional years (Craig et al. 2004). The AutoTutor project, during this period of

research, observed a effect size of 0.8 (Graesser et al. 2003). This observation is comparable with other tutoring systems in the same time period (Koedinger et al. 1997).

At the time of this study, it was believed that there were four emotional quadrants (Kort et al. 2001) across two axes: affect and knowledge, as shown in Figure 3. The theory is that learners take a learning path from quadrant IV to II to III to I, representing the learning path from first exposure to the material to its eventual understanding by the learner. At the time of Kort et al.'s study, the automatic coding of emotional states by computers or artificial intelligence algorithms was not feasible, due to the computational complexity involve. As a consequence, the 34 subjects who used dialogue interactions with AutoTutor were manually labeled for emotional state by expert coders. This was done in order to attempt to construct a model of the emotional states that were productive for learning. The results of this effort are shown in Table **5**, which draws the conclusion that 'Confused' and 'Flow' states lead to learning, but 'Boredom' does not.

Constructive Learning

Disappointment
Puzzlement
Confusion

Awe
Satisfaction
Curiosity

II   I

Negative
Affect

Positive
Affect

III   IV

Frustration
Discard
Misconceptions

Hopefulness
Fresh research
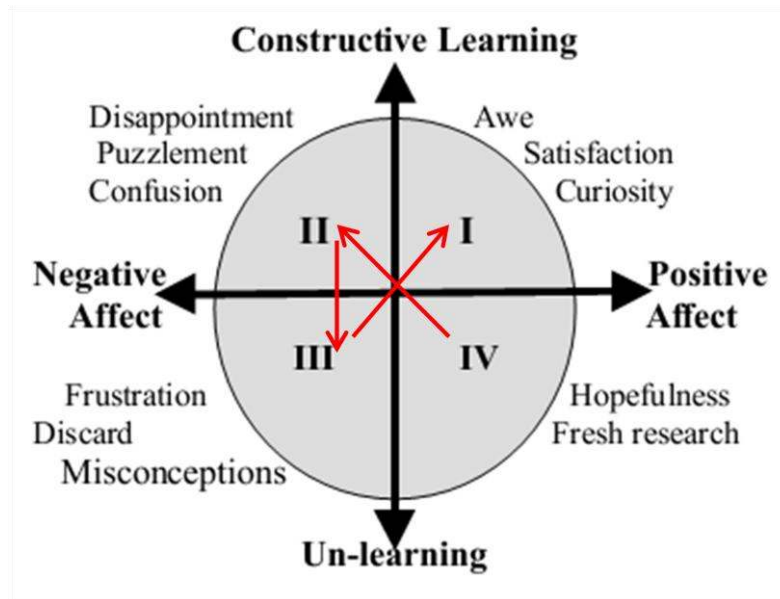
Un-learning

Figure 3 – Affective Knowledge Zones For Affective ITS Development (Kort et al. 2001)

Table 5 - learning gain correlation with manually-tagged emotional states, from (Craig et al. 2004), asterisks denote significance

Measure	Mean	Standard Deviation	Learning gains correlation
Boredom	0.18	0.2	-0.39*
Confusion	0.07	0.11	0.33*
Eureka	0.0003	0.02	0.03
Flow	0.45	0.28	0.29*
Frustration	0.03	0.09	-0.06

Several years later, the authors of AutoTutor constructed a system to automatically classify the affective states of the persons using it (Graesser et al. 2005). Several improvements to AutoTutor were made at this time, resulting in a combination of architectural components, such as the natural language functions. Additionally, a *bystander Turing Test* was conducted, and it was determined that a human could not tell the difference between a human-made or computer-made dialogue move. However, the

most notable improvement was the addition of four different categories of sensors to automatically detect emotion: facial expression, body posture, keyboard pressure, and mouse pressure. These sensors were previously part of other experiments in the burgeoning field of affective computing (Sidney et al. 2005).

Ultimately, this improved AutoTutor system was used in a manner that will be seen several times in this dissertation chapter. The system was used to teach, with recordings of the sensors taken to build predictive models of emotion. Presumably, these models would be used as part of a future system for the purpose of driving instructional strategies. However, the results in this regard were disappointing, as the authors were not able to produce an accurate model of emotion (Graesser et al. 2007). The authors state that the "next step is to build an emotion-sensitive AutoTutor that will promote both learning gains and more engagement in the learner." To the best of our knowledge this has never been performed, indicating the failure to transfer the offline-created population models of affect.

Work with AutoTutor did not stop, and is still an active area of research, with more than twenty involved researchers. AutoTutor has become a well-published project, with subjects in various domains, types of instructional strategies, knowledge construction, authoring tools, and human-to-human tutoring work leverage. The issue of affect, however, has never been addressed satisfactorily because of a complex series of problems mentioned throughout this chapter. It includes the policy that learner sensor hookups were to be generally discouraged, as well as the poor transfer of affective

models to a new population. More specifically, although a fundamentally simple set of sensors for affect detection was discovered, the AutoTutor affect classification system was never able to predict emotions in real time and this remains a barrier to the continued work in the area.

2.8. Crystal Island Experiments

The AutoTutor group has attempted to address the problem of intelligent tutoring systems through the study of human tutors and dialogue interactions. Blanchard's work (Blanchard et al. 2007), has attempted to model affect through the use of expensive, sensitive, highly-tuned sensors, and artificial intelligence. Crystal Island attempts to model affect from a very different angle, through the use of digital characters in grade school classrooms. Middle-school students interact with the "Crystal Island" experimental testbed, a virtual environment with instructional elements and pedagogical characters for the purpose of teaching microbiological concepts. The Crystal Island work begins with the study of motivational statements and full-body affective responses of an avatar, initially named COSMO (Lester et al. 1999).

By 2007, Lester et al. had collected enough learner response data on domain-specific interactions to start examining the prediction of frustration. This is a logical extension of affective models; if the user keeps telling the system that he is frustrated, then this should be a predicable occasion and can be mitigated. Measures of temporal interactions, location features, intentional features, and physiological response from blood volume pulse and galvanic skin response were collected and classified using

machine learning algorithms (McQuiggan et al. 2007). The result was an offline-created model shown in Table 6 that appears valid but was not validated. Overall, Table 6 shows the predictive accuracy of a handful of AI methods. These findings, rather than being validated, were used as inputs to other studies.

Table 6 - Results for UNC study of frustration prediction (McQuiggan et al. 2007)

	Unigram with Flattening Constant	Unigram with Good Turing	Bigram with Flattening Constant	Bigram with Good Turing	Naïve Bayes	SVM	Decision Tree
Accuracy	68.5%	73.4%	73.6%	73.5%	75.7%	82.2%	88.8%
Precision	60.1%	60.3%	61.6%	60.8%	76.3%	82.2%	88.7%
Recall	52.6%	59.6%	60.3%	59.9%	75.7%	81.9%	88.9%

The testbed for these experiments was the study of Crystal Island. As the user plays the game, various researchers on the island become sick, exhibit symptoms, and provide advice for the completion of scenarios. The user is free to interact with items in the environment, including chemistry lab sets, viewing posters, collecting samples of material, and other biological investigative behaviors. The users are asked about their emotional state in seven minute intervals, and can provide text response supplementing the state (Robison et al. 2010).

A study of 115 college learners (three classes) who used this system was conducted (Robison et al. 2010). The learner-reported measures of emotion were taken into account in an effort to predict emotional state transitions. These state transitions represent user transitions in the emotional state space, ie. from 'bored' to 'frustrated' or from 'confusion' to 'delight'. A 10-fold cross-validation Weka analysis using Bayesian

networks, linear regression, decision trees, and support vector machines revealed a predictive accuracy of 72% against the baseline of 68%. A report of only 5% improvement above baseline after leveraging the most complex artificial intelligence methods available shows effectiveness of generalized affective state transition models. There are few trends which are applicable across *all* individuals, and they are not reliable. This is another example of a model which has unknown implementation value, as it was not validated in an operational environment.

Sabourin et al. continued this line of research through the investigation of generalized affective models (Sabourin et al. 2011). This study contained data from 260 learners from two schools, and included an additional machine learning feature not previously seen. This method is the injection of experimenter domain knowledge in an attempt to eliminate statistical options and aid in algorithm performance, called a Dynamic Bayesian Network. The use of experimenter knowledge during model creation is extremely rare, as it assures that the model is not able to transfer to another domain, and is the only time such a method is discussed in this dissertation chapter. This study is one of only two validation studies, and necessitates a discussion of the results.

In short, as shown in Table 7, the models created by Sabourin et al. dramatically underperformed baseline measurements. The authors conclude with the statement that although "models were evaluated in a subject-independent manner, they were not successfully able to extend to a future population. This finding is particularly interesting given the strong similarities between the two populations." (Sabourin et al. 2011). The

addition of participants, use of advanced AI methods, and even *a priori* experimenter

knowledge about the domain were not enough to create a generalized model of affect

(Sabourin et al. 2011). It is possible that this is a case of model 'over fitting'. However,

models that have been overly fit typically have artificially large predictive accuracy

compared to baseline, which has not been observed.

Table 7 - The failure of AI methods to perform better than baseline upon unseen data (Sabourin et al. 2011)

	Emotion Accuracy	Valence Accuracy
Baseline	24.6%	56.7%
Bayes Net	17.9%	45.6%
Dynamic Bayes Net	25.9%	52.9%

There is evidence to suggest that Sabourin and Lester are moving away from

work in the area of affective modeling (Rowe et al. 2010a; Rowe et al. 2010b; Sabourin

et al. 2012a; Sabourin et al. 2012b). This is one of the two studies that cast the most light

on the problem of affective modeling. This study performs an attempt at validation, the

study of an attempted generalized model, the study of state prediction (rather than

classification), and the actionable data available for system use.

2.9. Educational Psychology

The above studies with AutoTutor and Crystal Island should not be interpreted to

conclude that *all* post-hoc analysis's of data are a poor idea. Many useful pieces of

information can be extracted during post processing. For example, group reaction to

marketing data or clinical research for stress management can be captured and analyzed

for the impact of various marketing messages or stressors, respectively (Hernandez et al.

2011; Picard 2011). In the educational domain, this task is akin to a cross-cutting cultural study of the impact of educational games (Conati 2002). An example of a useful generalized finding from the post-hoc analysis of physiological data is that a well designed intelligent tutoring system can be as engaging as a well designed game (Rodrigo et al. 2007).

In another cross-cutting study of learner frustration detection in an online computer science course, Rodrigo and Baker (2009) generated linear regression models from Weka cross-validation. As would be expected from the previously mentioned studies, the model shows weak correlation and prediction accuracy, which are marginal improvements over baseline. However, the authors found that it was possible to predict learner frustration from the observation of the interactions, but that the created generalized models do not accurately predict future interactions. They can show what *has* happened via interpretation of labels, but are unable to predict what *will* happen in the future. Interestingly, they find that individualized models perform robustly when they are taken as part of long-term interactions within the same system, but do not include any measure of physiological data. Once an individualized long-term model is constructed from interaction data, it remains valid, within that system, for an extended period of time. The authors suggest that in future work they will use more frequent detection reports of keystroke and mouse movement data in order to construct models with more predictive accuracy (Rodrigo and Baker 2009).

Not to be discouraged, the authors' later folded their study into a follow-on study to see how affective state *transitions* have an effect on the overall learning in the system. The finding was that the affective states of boredom and confusion were the most commonly observed states. Baker et al. matched these findings with the findings of the AutoTutor studies to conclude that the educational "downward spiral" consists of a boredom state followed by an inescapable frustration state (Baker et al. 2010). However, in their conclusion section they reflect that the group models of emotion are dependent on the system used, and the population which uses it. The authors suspect that there are scenarios for which this type of modeling is possible, but have since changed research interests, and not followed this line of research (Baker et al. 2012a; Gowda et al. 2012; Muldner et al. 2011; Soriano et al. 2012; Wixon et al. 2012). These findings indicate that an individualized model may be applicable, and transferable to a new system, but this remains a research gap that is addressed in this dissertation.

2.10. Affective Sensor Development

Investigation on affective sensors started from the grounded basis of educational psychology (Vygotsky 1978). A prevalent idea in the ITS literature is that there is a Zone of Proximal Development where the user is challenged enough to learn, but not so challenged as to become frustrated or stressed, as shown in Figure 4. Murray and Arroyo began their research by asserting that this zone can be detected through system-specific interactions (Murray and Arroyo 2002). The authors use these interactions to gauge the overall skill level of the learner. Given that this is a performance model of a learner,

rather than an affective one, it generalizes well across various domains (Cooper et al. 2011; Murray and Arroyo 2002; Murray and Arroyo 2003).
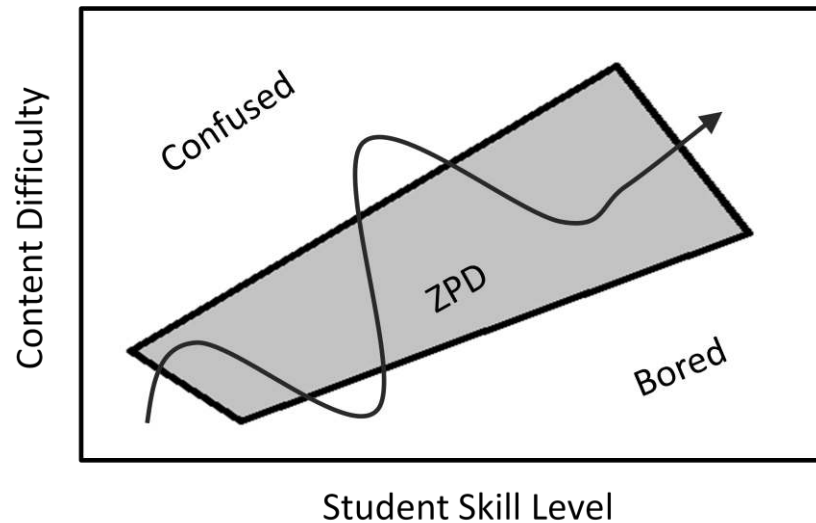
Figure 4 - Zone of Proximal Development (Murray and Arroyo 2002)

Murray and Arroyo's work dovetails nicely with the simultaneous research efforts within other groups. If the cognitive state can be accurately assessed, then an intervention can be generated to cope with the problem of learning, as shown in Figure 5. Indeed, the authors were reporting 80-90% accurate classification of state via Bayesian networks (Arroyo and Woolf 2005). This was combined with a suite of sensors including webcam-provided Facial Action Coding System data (a method for interpretation of affective facial data), posture sensing devices, skin conductance, and a pressure sensitive mouse. This was performed in the hope that the generation of a pedagogical intervention engine would be able to use these created learning models to drive decision making.

**Probability of engagement/motivation states
(estimated from 115 problem solving sequences of 12 problems)**
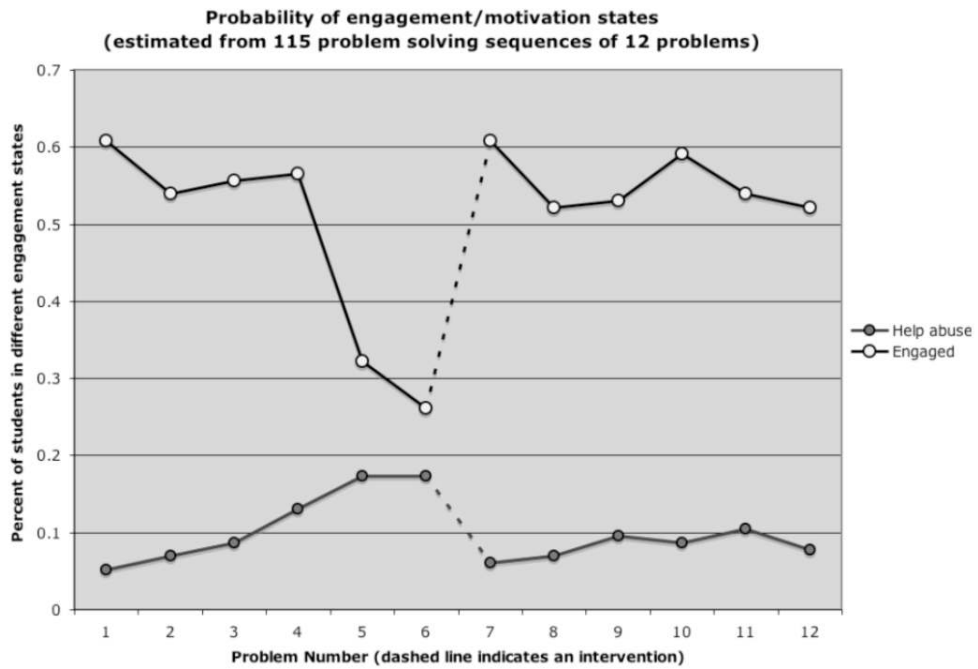
Figure 5 - The theorized effects of pedagogical interaction within an affect-sensitive ITS (Woolf et al. 2007)

The above reviews have conveyed that the problem of affect detection within intelligent tutoring remains a difficult problem. Dragon et al.'s study shows evidence that the physiological detection of affect was troublesome (Dragon et al. 2008). This study was conducted study with 34 learners using the Wyang Outpost intelligent tutoring system for mathematics with the sensor suite described below and an emphasis towards head, hand, and chair position. The findings of this study were that the measurement of affective state is possible, and that the suite of sensors can be used to measure it.

Figure 6 - Sensors used across several studies - (Arroyo et al. 2009)

This sensor suite consists of a webcam that is able to recognize emotive facial expressions such as *concentrating* or *interested* with software called MindReader. A GSR wristband is used to capture variance in arousal levels. Pressure-sensitive seat cushions were used in combination with an accelerometer to measure learner posture and activity. Finally, a pressure sensitive mouse was also used to infer the general frustration level of the user. The data from all of these sensors are combined differently in offline analysis to determine the best methods of multi-modal support. This sensor suite is used across a variety of studies, either in whole or in part (Arroyo et al. 2009; D Mello and Graesser 2007; Dennerlein et al. 2003; El Kaliouby and Robinson 2004)

This research motivated Arroyo et al.'s oft-cited study and paper utilizing emotional sensors in a school setting (Arroyo et al. 2009). In Arroyo et al.'s study, the authors were given permission to use hardware-based sensors inside of a classroom environment for experimentation. Rather than using Weka, a popular AI toolkit, they used only linear regression models, varying the availability of the sensors in order to

determine the sensors that were most able to predict affect. Unsurprisingly, they found that all of the sensors contribute towards the total picture of the learner, and that 60% of the variance can be explained via the models that they have produced. This is another study that was able to reasonably detect affective state in offline processing across a population.

It has been nearly three years since this study, and it begs the question of "what has happened since?". The closest clue that can be found is in 2011 by the same authors (Cooper et al. 2011). In this paper, they once again claim it is possible to create affective models from these data, and show cross-validated 90% accuracy compared against 60% baseline accuracy. With these results, the authors carried forward to a validation study in the same classroom, with the same subject, one semester later. However, the results of Table **8** indicate that *none* of the classifiers are able to outperform baseline measurements of emotion in the second semester (Cooper et al. 2010).

Table 8 - Evaluation of sensor framework from Fall to Spring semesters, with no validated accuracy above baseline (Cooper et al. 2010)

Model	Accuracy (%)		Sensitivity (%)		Specificity (%)	
	Fall	Spring	Fall	Spring	Fall	Spring
confBaseline	65.06	62.58	72.22	76.13	55.56	44.14
confTutorA	70.49	65.49	47.07	46.04	**90.43**	**84.88**
confTutorM	68.64	67.53	52.31	52.26	**82.41**	**80.68**
confSeat	65.70	67.13	54.63	60.17	**79.26**	**70.32**
intBaseline	42.42	78.30	0	0	81.82	100.00
intMouse	**83.56**	63.34	**29.73**	5.09	90.54	81.60
intCamera	69.44	57.65	**52.08**	**12.11**	64.58	68.53
excBaseline	46.31	74.31	0	0	96.15	100.00
excTutor	**73.62**	62.99	**36.54**	**12.45**	87.88	77.28
excCamera	66.33	51.53	**38.67**	**28.39**	72.00	52.24
excCameraSeat	70.67	43.34	**32.00**	**15.97**	83.00	54.07

The linear regression classification shown in Table **8** shows the creation of eight different models and three baseline metrics for the detection of the cognitive states of confidence (conf), interested (int) and excited (exc). Given that these models were being tested on a population different from the one in that they were collected and trained on, it is expected that performance will degrade somewhat. While performance is expected to degrade, it is still expected that the results will be superior to a baseline classifier, and the authors estimated this drop to be "between 2% and 15%" (Cooper et al. 2010). Values marked in bold highlight the results that are significantly better than baseline, and the reader can see that *none* of the eight models used in the Spring perform on the metric of accuracy. Given that the model is not accurate, it is not meaningful that it is more specific, or sensitive, although *half* of the models fail on this metric as well. The Fall dataset used

"just under 100 students", while the Spring dataset used "over 500 students", indicating the widest availability of data presented in this dissertation.

The finding that *none* of the linear regression models constructed across this time horizon are able to classify better than baseline is surprising unless one looks at the underlying psychological situation. Individuals are very different from each other (Miller et al. 1987). The Fall data used leave-one-out cross-validation, which uses all learners except one to build a model. The Spring dataset was used for validation, and simply used the best models produced from the Fall dataset. The individual differences present in the Fall data allow one person to be unique enough from the other 99 to throw off the classification accuracy. The differences present in the Spring dataset indicate that the 500 following people are significantly different from the previous 100. While this study is able to determine that meaningful generalized models can be *constructed*, it is not able to conclude that individual models can be *transferred* to another training session.

In our opinion, developed through numerous conversations with field researchers, research paper readings, and E-mail exchanges, the problem of affective modeling reached a dead end for this research team. There is simply not enough data to create individualized models. Furthermore, these individualized models are as unlikely to transfer as the generalized models from AutoTutor or Crystal Island. The generalized model has been shown to be invalid, and the models created in real time are too difficult to construct. As such, the problem has turned into one that was hard, was unlikely to work initially, and was not funded. Nevertheless, this second major validation study

reports findings similar to the Crystal Island experiments (Sabourin et al. 2011), that is that generalized emotional models do not transfer well to field use.

2.11. Realtime Mental State Classification

Research in the area of computer adaption to real time physiological signals has additionally been performed in the area of game adaption to learning. Citing some of the earlier work with educational games seeking affect sensitivity (Conati 2002), Blanchard et al. argue for the inappropriateness of the traditional approach of learner query (Blanchard et al. 2007). The simplest and most effective way to garner affect classifications is simply to ask the user. However, Blanchard is correct in his analysis that asking the user provides sparse data, cannot react to fast-paced training (such as educational games), and suffers from user bias, which has been historically positively-oriented and culturally-biased (Healey 2011).

Blanchard et al. (2007) believed that the use of a combination of sensors would obtain the user's emotional state without bias, and successfully account for individual differences within the data. In much the same ways as the dataset used in this dissertation work, a combination of everything that the authors could beg, borrow, or steal was used for the measurement of physiological state, including skin temperature, respiration, heart rate, blood volume pressure, galvanic skin response, surface electromyography (EMG), and electroencephalography (EEG). They criticize other researchers for the use of post-hoc analysis, and highlighted the need for a real time or

"predictive model" approach that is able to quickly classify a given set of inputs for use in real time pedagogical adaption.

With all these data channels across multiple users and multiple time periods, one would think that the construction of a usable individualized model would have been possible. Blanchard et al. underestimated the large individual differences present in physiological data, and include several graphs in their paper to highlight the difficulty (see Figure 7 for an example of one such graph). In concluding, they argue for multimodal detection while casting doubt on the availability of a classification model of emotion. In the authors' words:

> "[individual physiological differences] raise doubts about the relevance of using a predictive model approach for adaptation. Indeed, with such a level of inter and intra individual variability, what could be the significance of deductions obtained from data collected at different times, on different learners, in different conditions when the physiological reference frame is different?"
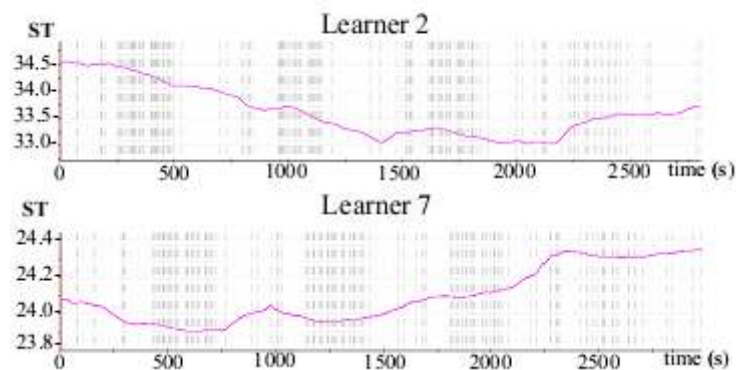
Figure 7 - Large variations in individuals shown in (Blanchard et al. 2007)

58

This did not, however, stop the authors from tackling the problem in slightly differing ways, as they published three papers on this topic in 2010 (Chaouachi et al. 2010; Chaouachi and Frasson 2010; Frasson and Chalfoun 2010). The first of these papers shows that the cognitive engagement index is positively correlated with the states of interest to learning. It suffers, however, from the same problem as many of these works; the post-hoc analysis of data with the presumption that the model will transfer to unseen subjects within differing timeframes. This presumption is carried forward in the second of these papers, into the domain of performance assessment. Again post-analysis discovers that the constructed EEG metrics correlate positively with emotional state, as measured via engagement and arousal. These emotional states are positively correlated with task performance, and the construction of individualized models is "not only possible but highly recommended" (Chaouachi and Frasson 2010). The third of these papers indicates that the determination of affect is difficult, moves for the inclusion of *additional sensors*, suggests firmer techniques for individualized model baselining and induction, and suggests the idea of subliminal learning. Subliminal learning includes the use of unseen cues on the content being taught so the learner is able to more easily learn content.

Once again, a research team who was intent on the construction of affective learner models for the purpose of developing affect-specific tutoring strategies is presented above. It is especially odd to note that skin temperature, respiration, heart rate, blood volume pressure, galvanic skin response, surface electromyography (EMG), and electroencephalography (EEG) could not provide a consistent assessment of emotional

state. Greeted with moderate initial success at the development of group models, they moved to individualized models. When the individualized models could not stand up to validation tests, they wrote papers suggesting more individualized approaches and more thorough baseline evaluations. Finally, as evidenced by work at a recent conference (Chalfoun and Frasson 2012), the problem is abandoned in favor of the use of EEG systems for cognitive priming and subliminal learning. This leaves the problem of usable real time affective models to other researchers, and is the specific subject of the research presented in this dissertation.

2.12. Individualized Mental Models

Certain types of signals naturally lend themselves toward individualized approaches. The best example is the EEG signal. The brain of each human is highly individualized (Medina 2008), and consequently, the EEG brain models must also be highly individualized. Traditional studies in the realm of EEG have hinged upon the development of highly individualistic models. The most obvious example of this is application of intensive periods of brain scans prior to brain surgery (Medina 2008). A standard approach to the problem of individualism can be seen in the affective EEG models described below.

In AlZoubi et al.'s research (AlZoubi et al. 2008) into EEG models, participants were taught to play Pong, an early computer game. The participants were told to think of moving their left and right arms, while connected to an EEG measurement system. After this, a model of left and right arm movement was constructed for each participant. The

participant then had to think of left and right arm movement in order to control a virtual cursor. The interesting findings were that models were highly individualized, that the best offline classification system was never the best online classification system. Furthermore, they found that offline classification models experienced sharp decrease in reliability when transitioned to practical use (AlZoubi et al. 2008). These findings are consistent with the findings presented by other researchers earlier in this dissertation.

Other work has shown that a small amount of caffeine can be enough to differentiate a previously created model from the current observation (Su et al. 2010). Thus, even if a transferable, person-specific, intraday, affective model could be created, it could still be rendered invalid for a training session through a caffeinated beverage such as a cup of coffee. As little caffeine as contained in a glass of tea is enough to perturb models of performance (Durlach 1998). This effect is also observed across other types of physiological data such as GSR (Hollenstein et al. 2012), EEG (Pollock et al. 1981), heart rate variability (Rauh et al. 2006), blood pressure (Nurminen et al. 1999), and others (Clarke and Macrae 1988).

Among the concepts presented at the Intelligent Tutoring Systems 2012 conference, was "if a cup of coffee breaks your model, it is not a very good model" during a talk on real time classification (Brawner et al. 2012). On a practical level, the amount of caffeine, sleep, or other physiological trend cannot be explicitly controlled prior to interaction with an ITS. Unfortunately, because of this problem, it is not likely

that an individualized model of affect is more usable in real world situations than the generalized ones presented earlier.

Further work in this area by AlZoubi et. al (2009) indicates that affective signal classification is possible from the EEG sensor array (AlZoubi et al. 2009). This approach has shown modest success, however, as they cite significant difficulties arising from user fatigue, electrode drift, changes in electrode impedance, and user cognitive state modulation (ie. attention, motivation, vigilance, or others). AlZoubi et al. argues that the problem inherent in these physiological signals is their non-linear nature, and that the failure of other models is because of the underlying linear assumptions. They indicate that the models are erroneously learned when it is assumed that the underlying concept is stationary, when in fact it is drifting across the sampling space (Hulten et al. 2001). As such, they hypothesize that nonlinear algorithms could be implemented to work satisfactorily. AlZoubi et al. empirically show this success through an injection of adaptive algorithmic techniques into the standard Weka techniques shown above, with *greatly* increased performance, as shown in Table 9 (AlZoubi et al. 2009).

Table 9 - Performance of adaptive algorithms against their static counterparts (AlZoubi et al. 2009)

Method	Static		Adaptive	
Classifier/windowSize	AvgErrorRate	STD	AvgErrorRate	STD
Knn/250	0.710	0,140	0.207	0.134
Knn/450	0.714	0.143	0.247	0.145
Knn/900	0.622	0.158	0.288	0.155
NaiveBayes/250	0.694	0.132	0.464	0.153
NaiveBayes/450	0.660	0.124	0.492	0.141
NaiveBayes/900	0.616	0.131	0.507	0.142
SVM/250	0.716	0.129	0.437	0.147
SVM/450	0.704	0.138	0.493	0.159
SVM/900	0.707	0.144	0.542	0.156

While this type of approach can be seen to boost the performance of the offline models, it is not appropriate for online use, because the algorithmic approach used here loops over all previous data windows for each injection of a new data window. In terms of computational complexity, this is $O(N^n)$, taking an exponentially longer time to develop a prediction with each additional data point. Any approach that can be implemented in real time must be of $O(k)$ magnitude, using a time-resolvable finite number of operations per each new data segment, as discussed later in Chapters 4 and 5. An observed unique feature of this type of approach, however, is that the general error *decreases* over time with adaption, while it *increases* over time with the traditional static affective models (AlZoubi et al. 2009). This is a highly desirable type of trait, indicating that the adaptive model improves with additional data, while the static model erodes.

With such an adaptive approach, AlZoubi et al. turned to the problem of day-to-day differences in multichannel physiology (Alzoubi et al. 2011). They conclude with a laboratory study with induced emotions that it would be possible for such an approach to

63

be implemented in the field. However, they paint the picture of the problems that still remain:

- how to use these algorithms on sparsely labeled data (real world)

- validating the algorithms in a person-independent manner

- alternative methods for classifier development and change detection

The problems present a solid research roadmap of unsolved problems in the field. This dissertation proposes methods of modeling these data that mitigate the difficulties currently faced.

2.13. Conclusion

We respect the research and tenacity of each of the aforementioned researchers. Each of them, directly or through association, has looked for individual or generalized models of learner affect that could be transferable and implementable within an intelligent tutoring system. Through the concerted effort, there have been two notable studies where researchers were able to put systems that appeared to function into practice (Cooper et al. 2010; Sabourin et al. 2011).

Unfortunately, each of these systems was shown not to perform well under the pressures of the real world. There are not enough individual data available to create individualized models (Cooper et al. 2010). Even if there were enough data available, complications related to individualized monitoring and daily differences would invalidate them (Alzoubi et al. 2011). Generalized emotional models barely perform better than

baseline, even when all of the offline AI methods in Weka are used in their construction (Robison et al. 2010). Even worse, they have been shown not to transfer well to the real world (Sabourin et al. 2011). This evidence points to a significant gap within the field.

Just as individual differences in height, intelligence, values, and personality are observed, the impact of emotional stimulus manifests itself differently among participants. Particularly in the realm of physiological sensors, there are differences wide enough to invalidate generalized predictive models. However, there are many difficulties even among predictive models that are individually tailored.

There are multiple conferences in the field dedicated to the use of physiological data correlated to various experiences among individuals or groups. However, problems related to individual differences drive the solution of individual analysis. This typically involves an approach where a researcher post-analyzes the data to look for correlations with subject-experienced events. While the post-facto treatment of the data has been of great aid to psychology researchers, an engineered system needs to use the data stream to respond to the needs of its users in real time (Dolan and Behrens 2012). To perform this task, these data streams would have to be parsed, interpreted, and classified into a state in real time.

Given that there is not likely to be a valid, generalized, model for predicting emotion across a population, adapting models for specific individuals would appear to be an alternative solution. However, people are fundamentally different, even with respect to the simplest readings. For instance, the highly individual nature of Galvanic Skin

Response (GSR) makes it virtually impossible to compare baselines across different people (Bersak et al. 2001). This makes the adoption of a baseline difficult. Additionally, even if an individual model were to exist, it would likely be invalid during the next training session. The reasons for this are legion, and include mood change across days, electrode drift, changes in default impedance of varying sensors, modulation across mental states such as boredom and attention (Alzoubi et al. 2011). Fundamentally, even if a model were adapted to a specific individual, that individual would appear very different to the modeled system upon the start of the next training session.

Note that there are large problems with judging a system based upon its accuracy. The least of these is that the accuracy of model prediction has no clear effect on learning effect size. Both large and small effect sizes may be observed from an increase in accuracy (Koren 2008). This disconnect further stresses that models should be built for their *use* rather than their predictive accuracy, as the end goal of an ITS is based around *instructional use*, rather than *user assessment use*, although accurate user assessment does aid in instruction. This highlights the need for real time adaptive approaches that can sacrifice accuracy in exchange for ubiquitous availability during learning sessions.

The emergence of adaptive affect classification, which has only recently begun, is a valid starting point for this dissertation. The authors of this dissertation have shown that adaptive algorithms (AlZoubi et al. 2009) dramatically outperform their static counterparts (Cooper et al. 2010; Sabourin et al. 2011). Additionally, the dynamic algorithms decrease in error over time, which is a highly desirable trait of any machine

learning method. While they have shown that these individualized models are possible (Alzoubi et al. 2011), they have not attempted implementation with real time constraints, which is what this dissertation addresses. Real time constraints call for different types of physiological data filtering, sparse labeling, and real time constrained methods.

The world is not an ideal place where the perfect solution to a problem always works perfectly. Engineers are trained in the concept of *trade space* in order to optimize towards multiple simultaneous goals. Engineers make compromises on solutions in order for the entire system to benefit. In the realm of affective models, there are several variables to trade from:

- Availability/Time – when the model is created

- Robustness – how well the model transfers to an unknown population

- Accuracy – how well the model classifies on a current population

- Sensitivity/Specificity – reaction to false positives/negatives

The sensitivity and specificity of potential solutions have been the engineering tradeoffs in all of the solutions shown in this dissertation. The other research discussed in this chapter has exclusively favored accuracy, in the hope that highly accurate models using offline data can transfer to the classroom. However, these robust models have been elusive, and we are not aware of a robust affective classification model at the time of this writing. Furthermore, the time to create a model has been largely ignored by affective models created offline. The other researchers who have created these models have not indicated the CPU time taken to create them, considering it to be irrelevant to the

majority of the work. The AI approaches used have primarily been in the form of Bayesian approaches which are time-variant, taking progressively longer to classify with each additional data point, making them impossible to run in real time.

We assert in this dissertation that the research community is making the wrong tradeoff. The key attribute of an affective learner model should be *availability*, or when the model is able to classify. Specifically, the model should be able to recommend instructional interventions at any time they can be gainfully used. Given that these instructional interventions are available in real time, the model needs to also be available in real time. While it would be ideal for an offline-created model to be transferred to an online mode, this simply has not happened. The chosen approach, *by necessity*, needs to be an online created model constructed for the individual after they have first started using the system in a learning session. This approach is a tradeoff, and given this tradeoff, the sensitivity/specificity of the model is likely to be low, with little or no robust transfer to other learners, and lower overall accuracy. These tradeoffs are made with the hope that the model will be *useful*, which is where all other methods to date have failed.

To summarize this chapter:

- Generalized models of affect have limited accuracy (Robison et al. 2010)

- Generalized models do not transfer well (Sabourin et al. 2011)

- Individualized models, while more accurate, also fail to transfer (Cooper et al. 2010)

68

- Adaptive algorithms for affective classification dramatically outperform static alternatives (AlZoubi et al. 2009)

- Increases in overall accuracy may not aid instruction (Koren 2008)

- Classification availability is more important than accuracy

  - A classification *now* is better than a better classification *later*, as *later* is too late to implement pedagogy

- An approach using adaptive algorithms to individualized models in real time provides classification availability, and address problems faced in affective model construction

3. PROBLEM DEFINITION

The previous chapters have shown a clear need for good affective models in the use of an ITS. Many other researchers have attempted to study this problem from various aspects and they have built the theoretical underpinnings of the current work. The use of affective learner models is still among the most promising technologies for the tailoring of individual training. In the first chapter of this dissertation, it was shown that one-to-one human-to-human tutoring has historically been the most effective way of instruction, and that human tutors manage learner emotional and cognitive state through affective interactions. Intelligent computer tutoring should emulate a strategy that has proven to be effective, and must develop effective real time emotional classification in order to do so.

Specifically, we propose to create a system to solve this problem in real time through the combination of the works of several others. The first part of the solution is to show that online methods of model creation are comparable to their offline counterparts. The second part of this solution is to make sense of the data through unsupervised, adaptive, machine learning algorithms such as Growing Neural Gas (Holmstrom 2002) and Adaptive Resonance Theory (Carpenter and Grossberg 1995), showing that these methods will transfer when supervised information is not available. The third part is to determine the impact of semi-supervised ground-truth labels, and how frequently they should be obtained to construct real time models with comparable accuracy to the offline models.

3.1. Hypothesis

The hypothesis of this research is that useful cognitive and affective learner models can be constructed in real time. These models are learner-specific, as each learner is an individual. Furthermore, we hypothesize that these highly individualistic models of cognition and affect, created in real time, can achieve accuracy on par, although possibly slightly diminished, with the offline models created for the same learner. This contributes significantly to the fields of affective computing and intelligent tutoring systems in the following ways:

- Diminishes the significant problem of individual differences

- Provides an affective model that is independent of cultural bias

- Increases the availability of cognitive/affective models of the learner

- Merges together the works performed in the various, somewhat disparate, fields of affective computing, simulation, training, intelligent tutoring, educational data mining, data stream/digital signal processing, and artificial intelligence.

The previous chapters frame our effort of the author to solve part of an important problem in a novel manner. Highly individualized models of cognition/affect have never before been constructed in real time. Intelligent tutoring systems are desired to be adaptive to the need of their learners through assessment of their mood, from sensor data, from the same learner in real time, with classification aided through self-assessments. This dissertation addresses this problem in a manner that no other research has, through making data availability the primary engineering tradeoff. It is expected that this research

will involve the selection of various types of artificial intelligence classification, the initial evaluation of these algorithms for online, real time, semi-supervised learning, and the validation of this approach on another physiological datastream of differing population, and the adaption of these algorithms to the problem at hand. Publication in this field has already been frequent. This speaks to the novelty and interest of the work and the educational merit of this type of practical engineering solution.

4. DATA OF INTEREST FOR AFFECTIVE AND COGNITIVE MODELING

The previous chapters have discussed intelligent tutoring systems, the important role of affect and cognition in the tutoring process, and the challenges faced in the creation of useful models of these processes. Chapter 5 will the discuss machine learning methods used for the processing of realtime data and Chapter 6 will discuss the results of this processing. However, it is important to discuss the data used to build these models, the types of sensors used to collect them, the experiments that produced them, and the initial baselines for fair comparison of machine learning algorithms. Chapter 4 has been set aside for this purpose.

4.1. Introduction

The above sections have described open research gaps that exist for models derived from sensor data, with a particular emphasis on the gap of real time creation and simultaneous evaluation. However, in order to create an affective or cognitive model, one must first have data available to analyze. This issue can be deceptively difficult, as the availability of a context-appropriate dataset is limited. An ideal data set includes several features, such as previous analysis, domain-independent collection on states of interest, on a population of interest, with relevant sensors for inclusion. These features are identified in the below list, and discussed next.

- Relevant states to learning

- Ability to be transferred beyond the system of creation

- Created on a relevant population

- Created using cost-appropriate sensors

- Contained labeled data

- Have previously established models

The first feature of an ideal dataset is that the collected state information should *hold research grounding* in the field of education. At a minimum, the collected state information should have learning relevance. An example of a dataset that should not be included is the Pose, Illumination, and Expression database (Gross et al. 2010). This database shows actors with various expressions under various lighting conditions. While the expressions of actors could potentially represent underlying cognitive or emotional states, these are not explicitly labeled in the database. Datasets where it is not possible to deduce emotional or cognitive states should be discarded.

The second feature of an ideal dataset is that the data be collected in a context where it *can be transferred* to another population. There have been several studies with emotional collection which are only transferable to a similar system. One example includes Baker's dataset, which draws emotional inference based on the actions that the student takes within a learning environment (Baker et al. 2012b). Another example of data which are not appropriate for inclusion is 'gaming the system' predictive models, which predicts whether the student is meticulously studying based on their interaction

74

with system-dependent screen elements (Baker et al. 2004). Even assuming that a researcher could achieve 100% accuracy, this model would only be relevant to the ITS which records these system-dependent actions, as other ITS systems will have differing interaction events as a natural part of teaching different subjects. This type of model is referred to as an interaction-based model, which may be contrasted with a models based upon collection of sensor data. Sensor-based models have transferability, as a sensor can supplement a system, while interaction-based ones are dependent on the system of interaction. Sensor-based models are of interest to the research described in this dissertation, as it hopes to address the needs of many ITSs.

The third feature of an ideal dataset is that it should be collected on a *population of interest*. Populations of interest explicitly include people who are learners, ideally while they are learning, at the various levels of potential ITS application (K-12, college, or adult). It should not include, for instance, data collected during gaming activities (Sykes and Brown 2003), or from a marketing research study (Laparra-Hernández et al. 2009).

The fourth feature of an ideal dataset is that it *uses sensors* that are appropriate for classroom use. While the algorithmic results of this dissertation are available for any domain that would benefit from rapidly constructed models, the purpose is to improve intelligent tutoring. As such, it is desirable to select the sensors that are feasible to use in the classroom. An example of a dataset that is *not* appropriate for inclusion is one that

uses, exclusively, a $50,000 EEG headset requiring 30 minutes of setup (Stevens et al. 2008).

The fifth feature of an ideal dataset is that it has *labeled states of interest*. It is not possible to evaluate the effectiveness of model creation without a metric for success. Labels are used in this research to evaluate unsupervised, semi-supervised, and supervised model creation alike. While it is possible to create models from unlabelled data, it is not possible to judge their value. Additionally, without labels, the next discussed feature is rendered impossible.

Finally, it is preferable for a researcher to compare against benchmarks which have been set by others. This allows the other researchers to optimize their methods, eliminates any potentially induced biases, and strengthens the conclusions. As such, the sixth and final feature of an ideal dataset is that it has *already been analyzed* by another reseacher or research team. This gives the work described in this dissertation a comparison benchmark.

Two datasets are used in this research. To the best of the author's knowledge, there is only one dataset in existence that meets all of the above qualifications, and was collected partially for this purpose. However, the first three chapters of this dissertation contend that online model creation can generalize to different populations, individuals, times, and areas of research. This claim calls for the inclusion of a minimum of two datasets that includes these items. A second dataset is included as part of this work to show transfer. The upcoming portions of this chapter will describe the reasons for various items of

inclusion, and a side-by-side description of the features of each dataset. It is useful to include a preview description of each study here.

The first dataset was collected as part of an experiment to evaluate low-cost sensors. College-aged military learners experienced a breadth of learning-relevant emotions while watching videos or playing video games. They were measured by a suite of sensors. Cognitive states, such as distraction, are labeled with a high-cost sensor. Affective states, such as frustration, are labeled with a self-reporting tool. Models developed under this effort are designed to replace the high-cost sensors measures. This dataset, and the experiment from which it was produced, is referred to as Dataset #1, or as the Low-Cost Sensors Dataset. The experiment which created it is described in greater detail in section 4.4. The features of this dataset are described in summary in Table 10.

The second dataset was collected as part of an experiment to evaluate physiological response to situations of changing workload, a cognitively relevant learning state. College students experienced simultaneous tasking on detecting changes and indentifying threats on a displayed monitor. Their cognitive state was monitored by a suite of sensors, with the data cognitively labeled with a high cost sensor. Models developed under this effort are intended to aid in classification of workload, with the intent of having a system compensate during times of high/low operator workload. This dataset, and the experiment that produced it, is referred to as Dataset #2, or as the Human-Computer Interaction Dataset. The experiment which created it is described in greater detail in section 4.5.

The reason that Dataset #2 is included in this research is to prove that realtime, individual-specific, modeling techniques from sensor measurement are transferrable to a new population and purpose. As the methods for creating the individualized models do not inherently contain information about the population, there is no reason to think that they would not satisfy the general transfer criterion; however, it still requires proof. The inclusion of this second dataset is intended to show the transferability of the realtime modeling approach. Only cognitive models will be created from Dataset #2 because it does not include any affective measures. The features of this dataset are described in summary in Table 10.

Dataset #1 is ideal for the creation of real time methods of model generation for the purpose of intelligent tutoring. To the best knowledge of the author, no other "perfect" dataset exists besides this one. However, the desire to create realtime models from physiological signals is not limited to the field of intelligent tutoring. Dataset #2 shows the application of physiological sensors in the area of Human Computer Interaction (HCI) as part of the University of Central Florida's (UCF's) Institute for Simulation and Training (IST) Human Agents for Training and Simulation (HATS) project.

Table 10 – Actual dataset features

Dataset	Relevant States	Transferability beyond system of collection	Relevant Population	Relevant Sensors (cost)	Labeled Data	Evaluated Models
#1 Low-Cost Sensors	Cognitive *and* Affective	Yes	Yes	Yes	Yes	Yes
#2 Human-Computer Interaction	Cognitive but *not* Affective	Yes	Yes	No	Yes	No

The experiments that led to these datasets were conducted by other researchers, with little/no input from the author. A new analysis, using different (and arguably more appropriate) methods of model construction is appropriate, given the historical issues presented in the first three chapters. Because of the intertwined nature of data collection, analysis, and the analytical expansion through a realtime modeling approach presented in this dissertation, it is useful to discuss why these experiments were conducted, their relevance to ITS research, and their initial conclusions. New methods of model creation are discussed in the following chapters.

This chapter briefly describes the affective and cognitive states of broad interest for data capture. It is followed by the discussion of the sensors used in the two datasets to capture these states, a brief description of the experiment that produced each dataset, the initial project, purpose, and the models created through data analysis. For Dataset #1, the initial offline models created for the analysis of this dataset are considered to be the initial benchmarks. For Dataset #2, the online models created as part of this research are

evaluated for overall model quality. Each of these tasks is discussed in this chapter, the real time methods described in Chapter 5, evaluated in Chapter 6, and summarized in Chapter 7.

4.2. Affective and Cognitive States

Cognitive phenomenon consist of mental state activities such as working memory load, executive function, attention, and sensory information processing (Derakhshan and Eysenck 2010). In short, this is a state of mind consisting of various types of awareness of the environment. *Affective* phenomena, on the other hand, consist of emotions attitudes, moods, and traits (Davidson et al. 2003). Rather than the total mental state, these affective states consist of the reactionary biases to stimuli within the environment. Both of these models are of interest to human learning, and to machines that teach, as is explained in the following section.

4.2.1. Cognitive States Of Interest To Learning

Research on physiologically adaptive systems has traditionally focused on operational environments. Examples of this are the systems within the Defense Advanced Research Projects Agency (DARPA) project for "Improving Warfighter Information Intake Under Stress through Augmented Cognition" (Raley et al. 2004). This includes the cognitive state bottlenecks that can result from fast-paced decision-making under stress, and can include such items as working memory and attention. Table 11 shows the identified cognitive states of meaning to specific operational environments (Morrison et al. 2006).

Table 11 - Cognitive Information bottlenecks identified for system action by DARPA projects

Industry Team	Military Application	Transition Sponsor	Primary Bottleneck
Honeywell	Dismounted Soldier	US Army	Attention
Daimier Chrysler	Armored Vehicle Driver	USMC	Sensory Input
Lockheed Martin	Tactical Strike Coord	ONR	Working Memory
Boeing	UCAV operator	USAF	Executive Function

ITSs are not for optimizing the processing of information, but instead are tailored to influence the learning process. Examples of where the learning process should be manipulated include, for example, an instance where mental workload causes delays in information processing, causing the user to incorrectly interpret information (Ryu and Myung 2005), or when large reductions in memory performance result from divided attention (Craik et al. 1996). It is desirable to avoid these types of situations within an ITS through some type of intervention, provided that it is possible to identify these states in realtime. The cognitive states of primary interest to learning are 1) workload, 2) attention, and 3) engagement. These states have been the most positively associated with learning gains in a significant portion of the literature, and are addressed further below.

The first cognitive state that shows significant relevance to learning is attention. The impact of attention on learning is clear: increased attention produces increased retention and increased performance. It should come as no surprise that increased attention is positively associated with quicker reaction time (Craik et al. 1996). While few improvements in memory recall result from increased attention, divided attention is correlated with lower results in retention (Small 1996). In task-specific learning,

increased attention focuses on the items of interest to the task, and increases overall performance (Ahissar and Hochstein 2002).

The second cognitive state that has been extensively studied is engagement. Similar to attention, lack of engagement is empirically correlated with a decrease in learning (Woolf et al. 2007). Among military tasks, increasing levels of engagement rise linearly with increasing levels of task difficulty (Berka et al. 2004). Low levels of engagement can be assumed to be indicative of non-participation in the learning environment, and related back to attention (Dorneich et al. 2007).

The third cognitive state that has empirically proven its relevance to learning is workload. Again the result is clear: users who have high workloads have corresponding decreases in performance and retention (Gonzalez 2005). Mental workloads are mediators to various aspects of perception, cognition (including learning), and even motor tasks (Parasuraman and Caggiano 2002). Measurement of workload can assist in the ability of the system not to overtask the user.

Although this is not an exhaustive list of cognitive states that have influence over learning, it provides a good baseline of items of interest. As shown later in this chapter, the cognitive states of attention, engagement, and workload are readily detectable using relatively inexpensive commercial sensors, or, alternatively using a single high-cost sensor. Additionally, each of the two experiments discussed in this chapter have identified the relevance of these cognitive states. It is important to monitor these various states of cognition in real time to provide remediation during the learning period.

4.2.2. Affective States Of Interest To Learning

There are many affective states that are linked to learning effectiveness. A short list includes:

- Anxiety (Pintrich and De Groot 1990)

- Arousal (Bradley et al. 1992; McQuiggan et al. 2007)

- Boredom (Craig et al. 2004)

- Confidence (Pajares and Miller 1994)

- Confusion (D'Mello et al. 2007)

- Frustration (McQuiggan et al. 2007)

- Joy (Fredrickson 1998)

- Motivation (Craig et al. 2004)

- Sadness (Bower 1992)

- Shame (Ingleton 2000)

- Surprise (Holland and Gallagher 2006)

- Wonderment (Campbell 2006)

The above list is not complete, as there are additional affective states that can be psychologically linked to learning, such as anger and disappointment. Potentially, many of these affective states could be measured as part of an experiment. Dataset #1 measures three of these affective states of interest: 1) arousal, 2) boredom, and 3) frustration. The last two of these states are negatively associated with increases in learning.

Arousal is a psychological and physiological state produced by the autonomic nervous system. Increased arousal naturally leads to increased heart rate, blood pressure, and sensory alertness. High arousal has been positively correlated with high retention (McQuiggan et al. 2007). Additionally, low arousal has been positively correlated with rapid forgetting (Kleinsmith and Kaplan 1963). In brief, something that invokes a measurement of high arousal can safely be assumed to be an item good for learning, as people learn about what excites them. Specifically, arousal indicates memory retention relating to the arousing event (Bradley et al. 1992). The reader should note that the experimenters of Dataset #1 have called this state 'Fear', but anxiety, fear, and arousal are all measured through the selected sensors and labeling techniques.

Boredom is an emotional state of being generally disinterested in the surroundings, and has been described as "an unpleasant, transient affective state in which the individual feels a pervasive lack of interest in and difficulty concentrating on the current activity." (Fisher 1993). Rather unsurprising are the psychological research findings showing boredom as leading to lower retention and decreased ability to apply information (Small 1996). Increased levels of boredom are negatively correlated with learning gain (Craig et al. 2004).

Frustration is an affective state associated with failure to meet set goals. The greater the failure, and the greater the amount of failed effort, the more frustrated a learner can become. Frustration causes the user to focus on the frustrating item, eventually diverting the learner away from learning goals and ultimately impeding

learning (McQuiggan et al. 2007). Frustration is not inherently negative to learning, if it is to cause arousal, or increase attention, but is generally associated with non-learning activity.

From the above discussion, we can conclude that arousal, frustration, and boredom have significant impact on learning. Although these conclusions are not shocking, they can present a representation of the learner. A classroom teacher or one-on-one tutor who is able to successfully classify these affective states among their learners can work to steer the learners' emotions away from states that have poor learning implications. In the same way, an ITS that is able to classify these emotional states has the potential to respond to them. How to respond to these states (e.g. what to do about a bored student) is beyond the scope of this dissertation, but the detection of these three measurements can provide an affective picture of the learner.

The datasets, and experiments that produced them, identify all (Dataset #1), or some (Dataset #2) of these measures, in addition to providing a significant amount of other data. Next we discuss the sensors used to capture these states before discussing the purpose, participants, experiment, analysis, and results of each experiment. This dissertation then expands on the models which were created as part of these efforts through the machine learning techniques described in Chapter 5.

4.3.   Application-Appropriate Sensors and Sensors Suites

Affective and cognitive states are closely related, but can change independently, and be modeled independently.   The affective and cognitive states of arousal, frustration, boredom, attention, engagement, and workload provide a sufficient affective and cognitive sample, when measured by a sensor suite, to justify the adaption of various types of desired instructional protocol.   A smaller subset of this type of sensor suite is used in similar research (Calvo and D'Mello 2012; Graesser and D'Mello 2012).   While it is desirable to measure additional states for instructional purposes, a transferable, six-dimensional, real time, learner model is expected to be of high value to affect-sensitive tutoring systems (Alexander et al. 2012; Graesser et al. 2012; Sottilare 2009).   If these states are to provide the 'minimal' set of states that are relevant to detect, then the sensors to detect them would be an example of the minimum amount of hardware required to detect them.   These states are able to be reasonably measured with a small sensor suite of five sensors, as shown in analysis section 4.4.3.   How to detect these states is well researched: arousal can be reliably detected via GSR sensor (Bradley et al. 1992); boredom and frustration can both be detected via behavioral motion sensing (D'Mello and Graesser 2007; Woolf 2009b); attention engagement and workload may all be sensed via an EEG head cap and ECG sensor (Ahlstrom and Friedman-Bern 2006; Berka et al. 2007). Each of these sensors was selected for low cost, and is discussed in sections 4.4 and 4.5.

There are many advanced ways of sensing the emotion of the person at the other end of the keyboard.   In an effort to make affective-sensitive training ubiquitous

throughout the area of training, the cost must be on comparable to the computer system used to train (Carroll et al. 2011). Additionally, the sensors used should not be readily apparent, or uncomfortable, to the learner who is being sensed. While a 'wearable' sensor is not new, basic modern sensors can cost upwards of $1,000 (Picard 2011). This is compared to the basic desktop computer purchase of approximately $400. A summary table of the sensors to be discussed in this section and used in this dissertation as part of the Dataset #1 is presented in Table 12, while the higher-cost cognitive sensors of the Dataset #2 are detailed in Table 13.

Table 12 - Summary of Sensors used, Affective States, and Cognitive States (Experiment #1 – Low Cost Sensors)

Sensor	Affective State	Cognitive State
ABM EEG (Ground Truth measure) Neurosky EEG		Attention, Engagement, Distraction, Drowsiness, Workload
Eye-tracker		Attention, Drowsiness, Workload
EmoPro (Ground Truth measure)	Anger, Anxiety, Arousal, Boredom, Fear, Stress	
Zephyr Heart Rate Monitor	Anger, Anxiety, Arousal, Boredom, Fear, Stress	Attention
Phidget Chair Pressure Sensor (posture)	Arousal, Boredom, Frustration	Engagement, Flow
Vernier Motion Detector (posture)	Arousal, Boredom, Frustration	Engagement, Flow

Table 13 - Summary of Sensors used and Cognitive States (Experiment #2 – Human Computer Interaction)

Sensor	Cognitive State
Eyetracker (Ground Truth measure)	Attention, Engagement, Workload

4.3.1. Sensor Hardware (Dataset #1 – Low Cost Sensors)

The two baseline measures used in the collection of Dataset #1 were EmoPro™ and an Advanced Brain Monitoring (ABM) EEG B-Alert™ X-10 Headset. EmoPro™ is a validated electronic emotional profiling tool (Champney and Stanney 2007). The ABM headset includes validated classification measures of workload, engagement, and distraction (Johnson et al. 2011). The ABM measures gives 10-channel, millisecond-by-milisecond resolution of cognitive state to the data collected, while the EmoPro™ metrics must, by necessity, be questioned after an emotional episode. A sample of these labeled data, as well as further discussion of sensors measurements is shown in APPENDIX A. Each of the validation measures are 'high cost' sensors.

Briefly, the research question that the study that produced Dataset #1 addresses is "Can you replicate the measures of validated, high-cost, obtrusive sensors with yet-to-be-validated, low-cost, unobtrusive ones?". In this regard, the initial conclusions drawn from Dataset #1 was that low-cost sensors *were,* to a reasonable degree, able to measure which are able to mirror the functionality of the validated, high cost, intrusive sensors. The Dataset #1 models show that the transition of a ITS system into a classroom setting could be accomplished with a suite of low-cost sensors and tuned computer models with minimal loss of functionality. A fully instrumented participant is shown in Figure 8. The

purpose to the remainder of this section is to describe, in detail, the exact sensors used as part of the study.

Figure 8 – Fully Instrumented Participant

4.3.1.1. LOW-COST EEG

The Neurosky Mindset EEG system based around a single-point, dry-contact forehead sensor. This sensor provides data on the Delta, Theta, Alpha, Beta, and Gamma brainwave blend, and produces measures of Attention and Meditation (NeuroSky 2007) .

The band power levels are output in the Delta, Theta, Alpha, Beta, and Gamma ranges. These measures of Attention and Meditation have not been validated in experimental research, and were used as part of one of the models in this study. Data measures on this sensor are provided in realtime via Bluetooth connection. This sensor produced measures of Alpha1, Alpha2, Gamma1, Gamma2, Delta, Beta1, Beta2, Theta, Attention, and Meditation, as discussed and shown graphically in Appendix A-1.

4.3.1.2.   EYE TRACKING

The hardware for the low-cost eye tracking solution was composed of a Thorlab DCC1545M monochrome camera, mount, a Opteka HD2 37mmR72 720 nm infrared X-Ray IR filter, and two IR010 Night Vision IR lights. This was then linked to a ITU Gaze Tracker Open Source software solution to determine eye position. A USB connection was used to collect and store the realtime data. This sensor produced the measure of Left Eye Pupil Diameter, as discussed and depicted graphically in Appendix A-5.

4.3.1.3.   HEART RATE SENSOR

The Zephyr HxM^{TM} BT heart rate sensor is a strap-based heart rate sensor that is affixed to the target's chest or midsection. Software internal to the sensor reports out measures of average heart rate over a Bluetooth connection. A future study should take note of sensor-free heart rate detection present within CardioCam, or similar technology (Mone 2011; Picard 2011). This sensor produced the measure of Heart, as discussed and depicted graphically in Appendix A-2.

4.3.1.4.   CHAIR SENSOR

The chair sensor used for this effort is custom-designed, but used a suite of sensors available commercially. Specifically, eight Phidget pressure sensors were used, with four on the bottom of the chair and four on the back of the chair. A USB connection was used to collect and store data in realtime. This sensor produced measures of Chair1-8, with the first four measures corresponding to the back of the chair and the last 4 measures corresponding to the seat, as discussed and depicted in Appendix A-4. A future study should take note of the Microsoft Kinect research team (Zhang 2012b).

4.3.1.5.   MOTION DETECTOR

A motion detection sensor was used to determine the position, velocity, and acceleration data of objects moving in front of it. When placed between the computer and the participant, it can determine changes in posture, as the participants lean forward/backward in the chair. These data somewhat overlaps with chair posture data, and was collected in realtime via USB interface. This sensor produced the measure of Motion, as discussed and depicted visually in Appendix A-3. A future study should likewise take note of Microsoft Kinect Technology (Zhang 2012a).

4.3.1.6.   DIFFERENCE-BASED FEATURES

Each of the sensors has produced several measures. A reasonable attempt to perform feature extraction on this dataset was not attempted by the original experimenters. However, the original experimenters constructed several types of derived features for data interpretation. Each of these features is calculated from the difference between the currently observed datapoint and the immediately previous one. This was done in order

91

to add classification accuracy to the models. The derived measures are: Alpha1Diff, Alpha2Diff, Gamma1Diff, Gamma2Diff, DeltaDiff, Beta1Diff, Beta2Diff, ThetaDiff, AttentionDiff, MeditationDiff, HeartRateDiff, and MotionDiff, and they are discussed and shown graphically in Appendix A-6.

4.3.2. Sensor Hardware Suite For Dataset #2 (Human Computer Interaction Experiment)

During the time that participants took part in the experiment to collect Dataset #2, they were simultaneously physiologically monitored via a different suite of sensors, including an EEG, a Transcranial Doppler system, a functional Near Infrared Imaging strip, an ECG system, and an eye tracking system. This variety of sensors is what initially made this dataset attractive. The study which produced Dataset #2 has not yet been able to construct models of workload from the integration of these sensors, and only of the sensors used in the study outputs validated metrics. True class labels are produced from this sensor, in the form of the Index of Cognitive Activity (ICA). These class labels are required for the production and evaluation of the online models. This is the only sensor used in this study, as it is the only one which is able to assure the experimenter that it has meaning.

4.3.2.1. EYE TRACKING

Seeing Machines faceLAB 5 desk-mounted eye tracking system was used with two cameras (one per eye) and a central IR source. This system measures movements of the eye, called saccades, how long the eye stays fixated on a point, called fixation duration,

and changes in pupil diameter.   These measurements can be combined, with an amount of filtering and feature extraction, to produce the labeled measure of Index of Cognitive Activity. This sensor produces two measures: FixationDuration, PupilDiameter; and the labeled measure IndexofCognitveActivity, as discussed and shown in APPENDIX B and Appendix B-2, respectively.

Figure 9 – FaceLab 5 System (SeeingMachines 2012)

4.3.3. *Sensor Hardware Suite Summary*

Two experiments have produced two datasets, which include multiple measures of states. The sensors and states of measurement are described in Table 12 and Table 13.   The exact models which were created from each of these studies are described later in this chapter. A summary of the sensors, measures, and Datasets, and the location of example graphs is included in Table 14.

Table 14 - Summary of sensors measurements

Dataset	Sensor	Measures	Appendix
#1	ABM EEG (ground truth)	HighEngagement Distraction Workload	A-7
#1	EmoPro (ground truth)	Anger Boredom Fear	A-8
#1	Neurosky EEG	Alpha1, Alpha2, Gamma1, Gamma2, Delta, Beta1, Beta2, Theta, Attention, Meditation	A-1
#1	Zephyr HxM	Heart	A-2
#1	Motion (custom)	Motion	A-3
#1	Chair (custom)	Chair1-8	A-4
#1	Eye Tracking (custom)	LeftEyePupilDiameter	A-5
#1	Difference-based features (software creation)	Alpha1Diff, Alpha2Diff, Gamma1Diff, Gamma2Diff, DeltaDiff, Beta1Diff, Beta2Diff, ThetaDiff, AttentionDiff, MeditationDiff, HeartRateDiff, and MotionDiff	A-6
#2	FaceLab 5 ICA (ground truth)	IndexofCognitveActivity	B-2
#2	FaceLab 5 ICA	FixationDuration, PupilDiameter	B-1

4.4.   Dataset One: Low Cost Sensor Experiment

This section will describe the relevant features of the experiment which led to the collection of Dataset #1.   This includes the purpose for the original collection, the experiment which collected it, the initial analysis and results, and how this dissertation work will expand it.

4.4.1. Purpose (Dataset #1)

The first three chapters of this dissertation served to show that one problem of intelligent tutoring relates to sensing the affective and cognitive states of the learner. The ITS pipeline relies upon the sensing of the learner, the correct classification of the learner state, and the informed selection of instructional strategies to mitigate or improve this state while training. Each of these presents a significant problem to the field, and is part of the reason why mastery-based ITSs have been prevalent: they can ignore state-based instruction and focus on content.

The selection of sensors that are possible to use in a classroom setting is non-trivial. It is not envisioned that each learner will sit all day at a computer ITS with a tube of contact gel and issued a 10-channel ABM EEG system, at a cost of $50,000 per seat. However, it is rare to find a sensor that: 1) costs less than the computer system ($400), 2) is not intrusive to the user, and 3) can accurately measure affective and cognitive state. Rather than create a single sensor capable of serving these functions, the Army Research Laboratory designed a suite of sensors (section 4.3.1) that together can provide part of the functionality of the high-cost intrusive sensor suite. Exactly how much functionality this suite of low cost sensors can provide is concluded as part of the original study and detailed in 4.4.3 and 0.

The selected sensors were part of an initial pilot study, published earlier in 2011, by Carroll (et al. 2011) about the appropriate selection of sensors. The initial study found meaningful effect sizes, and determined the cause of several kinds of errors, but

contained very few participants or clean collection. There were many small details of software and hardware which were resolved by the original experimenters for the conduct of a full experiment and meaningful numbers of participants. This encouraged a further, full-scale, study with more participants in order to fully evaluate a system of sensors. This study was conducted with the permission of Institutional Review Boards from Keller Medical Center, Design Interactive, and US Army Research Laboratory, the United States Military Academy (USMA) at West Point.

4.4.2. Participants and Experiment (Dataset #1)

A power analysis was conducted for this study and determined that 18 participants were necessary to determine which of the sensors could reliably determine affective and cognitive state information from the participants. Although 27 data sets were collected, only 14 of them provided usable cognitive labels, and 19 provided usable emotional labels because of unreliable sensor information. Each of the sensors used for this study was selected because of its low cost, which is typically correlated with low reliability. The 13 discarded sets of data are primarily due to one or more of the sensor datastreams being unavailable, which renders it impossible to evaluate which of the sensors contribute to a group model of affect or cognition. The population of interest is United States Military Academy (USMA) cadets, with 9 to 44 months of experience at West Point. This is roughly equivalent to a population of modern college students. The majority of the members of the population were Plebes (first year learners) enrolled in the Behavioral Sciences and Leadership (BS&L) Department's General Psychology (PL100) course.

Participants were asked to undertake a visual vigilance task, watch video clips from the movie Halloween, and My Bodyguard, and play several scenarios within the Army's Virtual Battlespace 2 (VBS2) video game. The video segment from Halloween has been previously validated to induce Fear/Anxiety, while the video segment from My Bodyguard has previously been validated to induce Anger/Frustration (Hewig et al. 2005). The VBS2 scenarios 1, 3, 4, and 6 contained limited visual perception (validated to produce fear/anger/workload), large numbers of enemies (validated to produce fear/anger/workload/engagement), annoying sounds (validated to produce anger/workload/distraction), or equipment malfunction (validated to produce anger/fear/workload/distraction) (Jones et al. 2012). The cognitive and affective states, and the tasks which induced them are presented in Table 15.

During each of these tasks, data were collected via the low cost sensors, and cognitively compared against the ABM EEG headset baseline with millisecond-by-millisecond resolution. After each of these events, the participant was affectively measured with the use of the EmoPro tool, and all data from the experience were labeled to be of that class (eg, anger/boredom/frustration). The EmoPro labels represent over five minutes of real time prior to a single label and correspond to a large number of data points. Events were kept short to increase the resolution of the EmoPro data.

Table 15 – Summary of tasks and states during Dataset #1 experiment

	Affective State			Cognitive State		
	Boredom	Anxiety / Fear	Anger / Frustration	Workload	Engagement	Distraction
Task	Visual vigilance					
Movie Clip		Halloween	My Bodyguard			
VBS2 Scenario		46	1346	1346	1346	1346

4.4.3. Analysis (Dataset #1)

The initial analysis of dataset provides a baseline to the classification efforts presented later in this research. The last item of interest on the checklist of features which described an ideal dataset was that it had already been analyzed using a type of offline method. It is not useful for this dissertation work to construct online models with nothing against which to compare. This analysis process has already been undertaken as part of the conduct of the first experimenters. The online and active methods discussed in Chapter 5 expand this analysis work through the rapid construction and the intermittent use of labels.

The initial classification algorithms considered for this dataset by the original analyzer, Ruben Padron, represent a broad spectrum of AI approaches: Logistic Regression Classification, k-Nearest Neighbor, Decision Tree Learning, Logistic Model Trees, Artificial Neural Networks (ANNs), Bayesian Networks, and Support Vector Classification. The reasons they have given for inclusion/non-inclusion for each of these methods are discussed briefly in Table 16. For the purposes of this dissertation, the realtime suitability is mentioned alongside the table, and is discussed deeper in Chapter 5.

Table 16 – Artificial Intelligence Methods Initially Considered for Offline Data Processing

Method	Inclusion	Reason	Real Time Application?
Logistic Regression Classification	Yes	Logistic Regression can easily have a 'goodness of fit' metric through R^2 statistical metric, and classify linear relationships between variables.	No
Decision Tree Learning	No	Although decision trees are capable of representing a wide swath of the classification space, they suffer from the 'curse of dimensionality', and cannot represent a non linearly-separable function	No
k-Nearest Neighbor	No	The k-NN approach does not allow the data set to be analyzed objectively for goodness of fit. As such, it was not included in the initial study. Given that it is real time capable, it will be included in the final study	Yes
Logistic Model Trees	Yes	The LMT approach allows for the gross separation of the data, followed by the linear regression on the reduced dataset, neatly solving the problems which are faced separately.	No
Artificial Neural Networks (ANNs)	No	The combined concerns of uninterpretable models, local minimum, and overfitting inclined the original experimenter away from this approach.	No
Bayesian Networks	No	This was ruled out in favor a method which is able to estimate correlations among variables (to determine which sensors are relevant)	No
Support Vector Machine Classification	No	SVMs have been ruled out for the same reasons as BN and NN approaches.	Somewhat

Additionally, it is worth mentioning that a binary classification of all states may not necessarily be the most appropriate method for intelligent tutoring systems. As an example Processing Efficiency Theory (Eysenck and Calvo 1992) and Direction of Attention Theory (Wine 1971) both indicate that multiple levels of classification, such as high/medium/low, are more appropriate to the task. In order to present a fair comparison between online and offline modeling techniques, the author cannot modify the dataset or labels. However, as these tasks are intended for inclusion and use, the recommendation

for a 3-step or 5-step classification model should be noted, and is discussed further in the concluding notes.

4.4.4. Results (Dataset #1)

The results were analyzed (Carroll et al. 2011) for how well the combined sensor set is able to detect the labeled state of the learner. The Logistic Model Regression method was encompassed in the technique of Logistic Model Trees that was selected as the method to use with 10-fold cross-validation. The sample was analyzed with the Receiver Operator Characteristic (ROC) benchmark (Hanley 1989), which plots the proportion of correctly-classified observations from the positive class (true positive rate) against the incorrectly-classified observations (false positive rate). The Area Under the Curve (AUC) of this function was calculated. The AUC ROC is designed to compensate for the misleading figures of "percentage accuracy" for unbalanced data. The AUC ROC measurement allows an algorithm with lower overall error rates, either true positive or false negative, to score well (Hanley and McNeil 1983), as the all of the categories of possible classification are weighted equally. In general, AUC metrics of greater than 0.8 are considered good, while classifiers lower than 0.6 are considered poor; those scoring in the 0.2 range in between those values are considered to be fair.

Table 17 – Results of the initial models on Dataset #1 – Which sensors can detect which states?

Sensor	EmoPro Measures			ABM Measures		
	Anger	Anxiety/Fear	Boredom	Engagement	Distraction	Workload
HR			X	X	X	
Eye Track						
EEG		X	X			
Chair		X		X	X	X
Distance		X	X	X		X
Classification (AUC)	NA	.83	.79	.80	.81	.82

The reader should note that there are a number of created models shown in Table 17. Each of these models was created independent of the others, resulting in three models of emotion and three models of cognition. These regression models may be linearly and independently combined for multiple attribute assessment. In total, this combined model presents a picture of which sensors (e.g. chair) are able to discover each 'ground truth' measure (e.g. anxiety). For the purposes of this dissertation, each of the evaluated machine learning methods will be compared against each of these data sources.

4.4.4.1. CREATED MODELS (DATASET #1)

The initial experiment by Carroll aimed to create six models in total (Carroll et al. 2011). Three of these were to be on affective features, with the remaining three to be on cognitive features. The cognitive labels were engagement, distraction, and workload, while the affective labels were anger, anxiety, and boredom. Through analysis, five out of six of these models were created successfully, with a model for anger being the exception. Carroll hypothesized that there were not enough instances of anger present in the dataset to create an effective model of any of the subjects. This dissertation work,

however, does not see a need to *exclude* the attempt to create a model of anger from this data. While offline, population-based, methods could not establish predictive meaning, online, individualistic models may be able to do so.

4.4.4.2.   SUMMARY OF THE LOW COST SENSOR DATASET FEATURES FOR CREATED MODELS (DATASET #1)

Effectively, this dataset has 32 dimensions across all timescales. There is a 33rd feature, time, which was explicitly not used in the construction of models from Dataset #1. While is not *explicitly* used for offline-created linear regression trees of the initially created models, it is *implicitly* used during real-time processing, as realtime-capable algorithms are sensitive to the order of presentation of data. This sensitivity to the order of data presentation may or may not convey an advantage, depending on the algorithm, but is hypothesized to aid based on previous research findings (Brawner and Gonzalez 2011). A summary of the data used to create each model in the initial study is shown in Table 18, while Appendix A-9 shows an example of a single data point, and the APPENDIX A to this dissertation shows examples of each feature of data over time.

Table 18 – Summary and example of features used in each created model

	Appendix	Boredom	Distraction	Engagement	Fear	Workload
Alpha1	A-1				X	
Alpha2	A-1	X			X	
Gamma1	A-1	X			X	
Gamma2	A-1				X	
Delta	A-1				X	
Beta1	A-1				X	
Beta2	A-1				X	
Theta	A-1				X	
Attention	A-1				X	
Meditation	A-1				X	
Left Eye Pupil Diameter	A-5				X	
Heart	A-2		X	X	X	
Chair 1-4	A-4					
Chair 5-8	A-4		X	X	X	X
Motion	A-3			X	X	X
Alpha1Diff	A-6				X	
Alpha2Diff	A-6				X	
Gamma1Diff	A-6	X			X	
Gamma2Diff	A-6				X	
DeltaDiff	A-6				X	
Beta1Diff	A-6	X			X	
Beta2Diff	A-6	X			X	
ThetaDiff	A-6				X	
AttentionDiff	A-6				X	
MeditationDiff	A-6				X	
HeartDiff	A-6	X			X	
MotionDiff	A-6				X	

4.4.5. Expansion (Dataset #1)

The reader should consider the initial goal of the experiment which produced Dataset #1 when viewing the results (Carroll et al. 2012). The goal of the experiment was to use classification techniques in order to evaluate how well a set of low cost sensors is able to mimic the performance of the higher-cost counterparts. The goal of *this dissertation* is similar, but different: to create and evaluate online algorithms comparable to their offline counterparts, expanding the state of the art through making emotional/cognitive models *available* rather than *accurate*. The initial analysis of Dataset #1 was performed in an offline manner, using the same type of classifiers that were used in previous studies mentioned in Chapter 2. These methods are not used in this dissertation because of their offline nature and group-based modeling approach, which are discussed further in Section 5.3.

Given the conclusions about the study of which low-cost sensors are able to successfully mimic their high-cost counterparts (shown in Table 17), it is known to be *possible* to create predictive classifiers on this sort of data, and that the sensors available are able to detect the results of the six types of cognitive and affective models. The initial benchmarks in the construction of this dataset provide a good starting point for the work described in this dissertation in the evaluation of real time classification metrics, and provide a dataset that is likely to be applicable to future studies in ITS research.

4.5. Dataset Two: Human-Computer Interaction

4.5.1. Purpose (Dataset #2)

The experiment that produced the Dataset #2 was part of a larger suite of experiments, each of which was targeted towards different objectives. The first of these was the objective to examine the relationship of workload and multi-tasking performance as part of a Mixed Initiative Experimental (MIX) testbed, which incorporates theory-driven tasks into a moderately high-fidelity military simulation designed for multi-tasking and physiological data capture (Reinerman-Jones et al. 2010). Another objective was to validate previously-created created models of human performance. The most relevant experimental purpose is to *create generalized models of physiological response* to situations of changing workload in order to preemptively reduce workload in the future (Barber and Hudson 2011). The dataset which is of interest to this dissertation is the one which has collected physiological measures from various sensors for workload classification. The results of the experiment which produced Dataset #2 are currently unpublished, but performed at the University of Central Florida Institute for Simulation and Training by Lauren Reinerman-Jones and Julian Abich.

4.5.2. Participants and Experiment (Dataset #2)

The experiment consisted of two simultaneous tasks shown in Figure 10: change detection and threat detection. During a change detection task, the participant must note when an item on the lower half of the screen changes, which can be either of icon, color, or location. During a threat detection task, the image of a hostile militant is presented

somewhere in the upper half of the environment. There are five levels of threat/change stimulus frequency across four scenarios. The first scenario presents only a change detection task, while the second presents only a threat detection task, while the remaining two scenarios present varying levels of stimulus frequency among the tasks. These tasks variations are intended to cause variations among cognitive variables such as engagement, distraction, and workload. More information on the experiment and experimental setup is available in recent publication (Vogel-Walcutt and Abich 2011).

Figure 10 – MIX Testbed showing Threat Detection (Top) and Change Detection (Bottom) (IST 2012)

The participants were recruited from a population of undergraduate college students from several universities. They were required not to have ingested alcohol 24 hours prior to

the study, and ingested neither caffeine nor nicotine two hours prior. The total experiment length was three hours.

4.5.3. Analysis (Dataset #2)

The initial dataset, unfortunately, does not yet have created models built upon it. The experiment collected measures of EEG activity, functional near-infrared imaging, and other physiological measures, but has not yet created labels models to test against. As such, these other physiological sensors are not used in this dissertation work. However, the FaceLab 5 sensor produce measures which have been validated (Bartels and Marshall 2012; Palinko et al. 2010), and used in complex tasks (Halverson et al. 2012). This assures the experimenter that reliable models can be created from the data. A sample of the available data is shown in Appendix B-3, as it was earlier shown for the many-dimensional data of Dataset #1.

4.5.4. Expansion (Dataset #2)

There were two objectives to the physiologically measured subset of the experiment that produced Dataset #2, as conducted by Dr. Reinerman-Jones. The first of these was to determine more cost-effective measures of workload as garnered from a suite of sensors. The second objective was to build models/classifiers of an individuals' workload. It is expected that the cognitive models of workload created with offline methods for human-computer interaction purposes will degrade over time for the same reasons as the ones created for ITS purposes (population differences, individual differences, and intraday differences). The research to collect Dataset #2 can logically be expanded via the

methods proposed in the first three chapters, using online and active learning methods to rapidly construct and use individualized models. If an individual model can be created in real time, it would represent a more robust approach to model creation, and a new method for workload measurement. This has application in HCI (Zander et al. 2010), robotics (Harriott et al. 2012), and other domains (Majumdar and Ochieng 2002; Parasuraman et al. 2009).

4.6.   Summary

Many types of models were created and are discussed over the course of this dissertation, so it is useful to include a summary of the models created and their comparisons. Several models of varying type were created from the analysis of Dataset #1 and #2. The Dataset #1 analysis created six models from two ground truth labeling systems on the same data. The ABM EEG was used for the three types of cognitive labels, while the EmoPro tool was used for the remaining three types of affective labels. Dataset #2 used the ABM EEG system for labeling differing cognitive states under varying levels of workload. The Low-Cost Sensor study used a generalized regression model, while the Threat and Change Detection study used a generalized eyetracking approach. Each study had a different population.

Table 19 – Types of models and their comparisons

Comparison Study	Population	Type of feature	Name of Feature
Low-Cost Sensor	Westpoint	Affective	Anger
Low-Cost Sensor	Westpoint	Affective	Anxiety/Fear
Low-Cost Sensor	Westpoint	Affective	Boredom
Low-Cost Sensor	Westpoint	Cognitive	Engagement
Low-Cost Sensor	Westpoint	Cognitive	Distraction
Low-Cost Sensor	Westpoint	Cognitive	Workload
Human-Computer Interaction	College Students	Cognitive	Workload

There is not a conclusive way to test whether an AI approach will generalize to *all* datasets of a problem domain. Table 19 shows that the methods evaluated in subsequent chapters are tested against two populations, with two different types of features, across seven of different model outputs. This large number of created, individualized, models is each tested across an amount of supervision, with fractional data. It is reasonable to think that an approach that can address this wide variety of situations will, at minimum, provide a starting solution to the problem of rapid individual model creation.

5. ALGORITHMS FOR REALTIME PROCESSING

The prior sections have made it clear that affective and cognitive models are needed in order to appropriately adjust instructional strategy. They have also shown that the current methods of offline analysis are not generalizable to populations, and are not usable after a matter of hours of learner unavailability. This creates a research gap in the area of model construction and realtime utilization. Logically, only algorithms that can cope with the challenges of realtime computing are able to address this research need.

There are four main problems with realtime data, each of which is discussed in this chapter. In brief, they are 1) the data can be of potentially infinite length, 2) concept detection, 3) concept drift, and 4) concept evolution (Beringer and Hüllermeier 2006). The combination of these issues present a problem for whichever type of algorithm is used to solve it. The realtime construction and use approach necessitates a stream model of the data, with the following assumptions, and corresponding design limitations, as Beringer outlines:

- The data cannot be requested, and may be available only for a short time

    o Operations must be done on the data as they become available

- The order of the data points is outside of the control of the program

    o Knowledge about prior points must be encoded, if they are to be related to each other

- The dataset is of infinite length

- It is not possible to store or analyze all of the data

- Data elements are not available for repeated request (data volatility)

  - Data must either be saved or discarded

  - Practical memory limits necessitate the discard of most data

  - Practical processing limits necessitate the discard of most data

- There are strict time constraints

  - Data must be processed in real time

  - Data can change quickly

  - An approximate solution is an acceptable substitute for an ideal one (Considine et al. 2004)

After a discussion of the problems with processing real time data, and a further discussion of the issues presented in affective modeling, each of the algorithms tested on the data is discussed. These include a type of clustering, an adaptive linear approach, Adaptive Resonance Theory (ART), and a technique for Growing Neural Gas (GNG). Each of these algorithms required several non-trivial modifications to become appropriate for the task, and these modifications are discussed. After a discussion of these different approaches is presented, the performance of each method on the datasets of Chapters 4 is shown, and conclusions are drawn.

5.1.   The Problems with Real Time Data

In this section we will explain the fundamental problems of realtime data.   These fundamental problems are infinite length, concept detection, concept drift, and concept evolution. Each of these items is explained in depth in order to frame the discussion of algorithms later in this chapter, as each algorithm addresses these problems in a fundamentally different fashion.

5.1.1. Infinite Length

The first and most obvious problem with real time datastreams is that the stream is of unknown length (duration). The software developer is not able to determine *a priori* how long the session with the learner will last. New data points come in continuously, but typically at a constant rate. The most significant effect this has on algorithm selection and development is the unavailability of historical data. While an algorithm may be able to utilize a number of clusters, weighted vectors, or other *encoded* historical data, it is not able to *directly* analyze historical data for this encoding. Encodings impose reduce memory limitations, but the growth of encoded representations typically increases computational cost in the comparisons between encodings.

The problem of infinite length may be somewhat mitigated through the use of windowing techniques. This involves looking at a small segment of the data at one time, training on it, and creating a new segment of training data. This method has shown success in developing quicker training times with normal AI methods (LeCun et al. 1998) described in section 5.3, but has been shown to lack in performance when compared to a

112

well-constructed online version (Shalev-Shwartz et al. 2004). Furthermore, the addition of windowing adds another variable of experimentation to the methods already being analyzed. Experimentation with windowing is likely to have less overall effect on the problem than experimentation with differing forms of stream processing. This dissertation focuses on stream processing, while acknowledging the advantages that certain windowing techniques may bring.

This limitation rules out many AI techniques that analyze historical data as part of model construction. For instance, probabilistic approaches such as Bayesian Networks require an update that considers all observed data in order to construct a new model, and performing this step for each additional data point is not feasible. Other approaches, such as reinforcement learning and genetic approaches are also inappropriate, as they require the testing of the algorithm on the historical labeled data in order to improve. The discarded classes of AI solutions are discussed further in Subsection 5.3.

5.1.2. Concept Detection

Given that an algorithm could be made to deal successfully with infinite data length, the next problem that it would face is the detection of a new concept. When the learner starts a session, the algorithm begins with no historical knowledge and no encoded knowledge. It will then be presented with data that it must sort into a group, cluster, structure, encoded via weight vector, or other otherwise. These encoded knowledge groups will eventually have meaning added to them (student performance data, self-report data, etc.), through the course of a training session.

As such, it is likely that the first presented point will represent the first class/cluster/grouping of information. Figure 11a shows a blank algorithmic slate that has had a single point added to it. Figure 11b shows the algorithmic response to the addition of this first point. This response can be made solely based on the determination that the datapoint is different from the previously established encodings where none exist. As it is not likely that all of the data presented is of a singular class, a future data point will need to be classified differently. The algorithm must determine a way to separate this datapoint from other datapoints with which it will be presented at a later time, including the detection of additional concepts. This problem is related to the realtime outlier detection problem (Subramaniam et al. 2006). Figure 12a shows the later addition of a differing class of data, along with Figure 12b, which shows the ideal algorithmic response to a differing class of data.
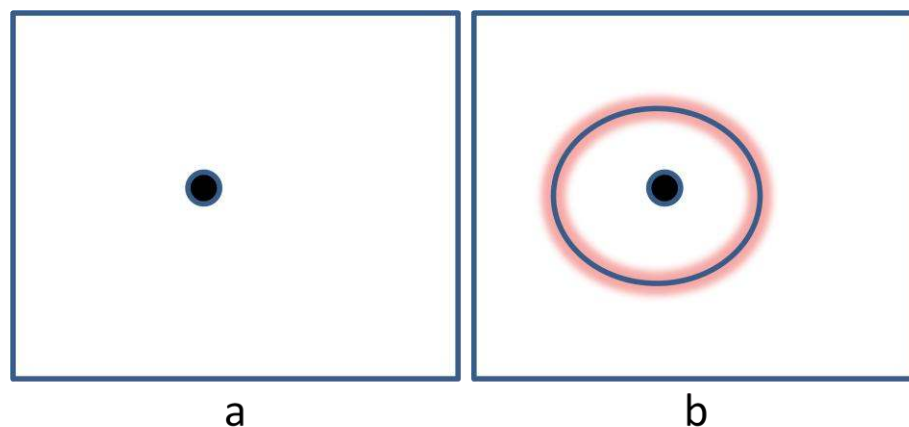
Figure 11 - Initial Concept Detection

114
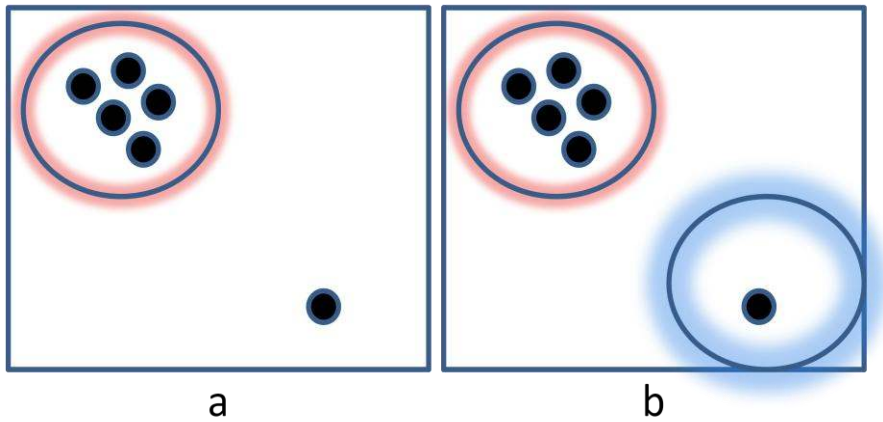
Figure 12 - Secondary (Novel) Concept Detection

5.1.3. Concept Drift

Concept drift refers to the changing nature of a concept over time. After a concept is detected, patterns associated with it may be subsequently present. It is the challenge of the selected algorithm to establish the similarity of the new data points to a previously established class without labels. If these new data points are related to the previous ones, they should be encoded similarly. Each concept will represent itself uniquely over time, and each algorithm must be able to cope with these observed changes.

Figure 13 shows how an algorithm may deal with the problem of an emerging class through expanding a classification boundary. A previously established classification boundary is expanded to deal with the neighboring objects. The first classification boundary is shown on the left, and the newly established boundary that relates the newer points to the older ones is shown on the right.
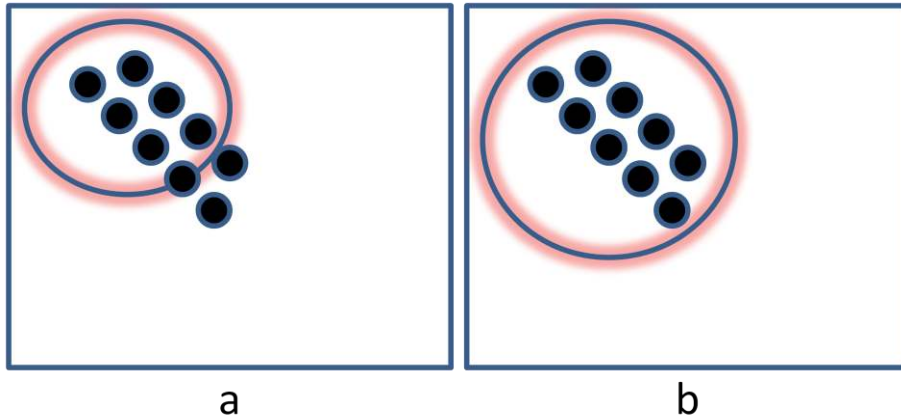
Figure 13 - Concept Drift

5.1.4. *Concept Evolution*

The detection of a concept, such as trainee state, may not present itself in a single, unified, manner. In the domain of affective computing, a learner state such as 'confusion' may present itself as a wide variety of sensor and behavioral measures. As an example, a learner may put his head on the desk *or* slouch in a chair while he/she thinks about a particularly hard problem. Both of these actions are representative of the underlying state, but are significantly different actions. If the algorithm is expressly informed that two groups of data are similar, it should be capable of associating them to be related. Figure 14 shows two groups of data which are labeled as similar by an outside entity, and shows how the classification (left) changes (right) after the presentation of labels.
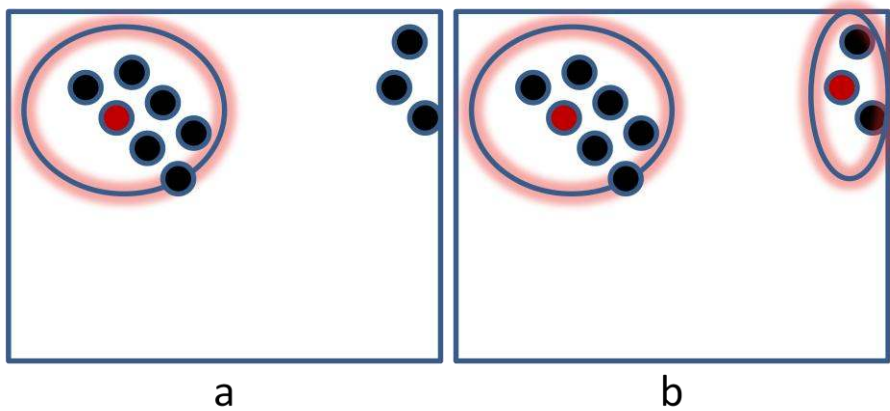
Figure 14 - Evolution of a single concept, determined to be the same state through outside labeling information as shown in red

5.1.5. *Discussion*

Just as it would be fortunate if there was a general-purpose group model of emotion, it would be fortunate if there was one algorithm that met the needs of this specific problem. Instead, there is a list of features that any algorithm must have in order to deal with the fundamental datastream problem. This checklist of mandatory algorithmic features is shown in Table 20.

Table 20 – A checklist of features for realtime AI algorithms

Infinite Length	Concept Detection	Concept Drift	Concept Evolution

5.2.   Real Data

As a practical consideration, there *is* the availability of the *occasional* labeled data point. This section phrases the problem of making use of this occasional information and presents a two-part solution to the problem of algorithmically modeling this useful

information. The guiding recommendations of this section are implemented in each of the algorithms tested within this dissertation.

5.2.1. Problem

In addition to each of the problem with realtime classification, there is a problem with how each algorithm adjusts to the nature of the *underlying* datastream. The data which has been gathered as part of Dataset #1 and Dataset #2 are unique with respect to real world data in that it has labels. *Each* datapoint has an associated label. The cognitive labels of Dataset #1 and #2 have been provided via expensive data collection hardware, in the form of an EEG headset. The affective labels of Dataset #1 have been infrequently collected after an emotional event, rather than immediately via headset.

The labels which came provided with Dataset #1 and Dataset #2 were costly to obtain. The first of these costs was the direct expense. A validated affective labeling system is expensive (time, money, personnel resources) to design and validate. A validated cognitive sensing system is expensive to purchase, as shown by the $50,000 pricetag of the ABM EEG. The second of these costs was time. For the affective labels of Dataset #1, the participant must stop the event, think about how they are feeling, and label this state. This process takes approximately 5 minutes of the total 60 minutes allocated. For the cognitive labels of Dataset #1 and Dataset #2, the participant must be fitted with the EEG system, and have their baseline EEG state recorded and saved for future use. This process takes 60 minutes, representing significant preparation time for a

40 minute collection period. The time spent in either of these two events is time that would be better spent learning from an ITS.

It is reasonable to assume that this type of "ground truth" information will be not reliably available in the future (Conati 2011; Kokini et al. 2012). The learner cannot be asked how they are feeling during each second of a learning session. The learner cannot spend the first part of every training session being fitted with EEG systems and contact gel. It is foreseen that unobtrusive sensors that require a minimum of calibration will be used as part of the learning classroom of the future (Carroll et al. 2011). These systems provide a minimal amount of "ground truth" information about the state of the learner. This purpose of the research described in this dissertation is not to construct models for their own sake, or for their comparison and evaluation, but for their use. The use of these models necessitates an approach where labels are neither inherently available nor entirely absent.

5.2.2. Solution Part One: Semi-Supervised Adaption

The problem of inherent label unreliability is solvable. The machine learning community has traditionally segmented on the ideas of "supervised" (with labels) or "unsupervised" learning (without labels). However, a new field is beginning to emerge to address this problem, known as *semi-supervised*, or *transductive* learning (Zhu 2005). Semi-supervised methods use information contained in the unlabeled data to 1) make inferences on the structure of the labeled data, 2) reprioritize the classification of prior data points.

Each of the methods used in this dissertation is screened for the ability to deal with all of the problems of realtime data classification, and the ability to handle the real world issue of limited label availability. If a method does not have an implementation for semi-supervision, one was created for it, and is detailed in the appropriate section. The most important feature of each algorithm is its ability to deal with the realtime data problems. It is expected that some information about the user may be available during runtime, regardless of the level of supervision being used in model creation. The user can be asked directly about their state, if it is done occasionally, and this information can be used to help build a model.

A semi-supervised capacity has been added to the clustering, ART, and linear regression approaches discussed in this chapter. The exact implementation follows an active learning implementation as discussed next in section 5.2.3. The exact implementation that has been added to the algorithm is dependent on the algorithm itself.

5.2.3. Solution Part Two: Active Learning

There is a special category of semi-supervised learning which is applicable to the issue of user modeling called *active learning*. Active learning involves exploiting the data structure of the semi-supervised version of an algorithm in order to request labels, provided that there is an 'oracle' which is capable of granting these label requests. When an algorithm is able to assess which locations of datapoints will have significant impact on the overall classification performance, it is useful to be able to request them. Dagupta and Langford present a review of active learning methods, when to request labels, and

why to do so. In short, there are two reasons to make use of active learning to request the labels of data points: 1) exploit cluster structure, and 2) efficiently search through hypotheses (Dasgupta and Langford 2009).

In the case of the work of this dissertation, the active learning modifications to algorithms are not explicitly appropriate for realtime implementation, as they make use of historical data. On a practical level, however, it is possible to generate label requests in realtime, if it is done occasionally, as the total runtime data presented in Table 23 shows. However, this implementation is intended to investigate the promise that the occasional labeled data point can have. The guidance of Dasgupa et al. has been followed for the selection of active learning data point selections (Dasgupta et al. 2007), described in each algorithms section. In this dissertation, this is represented through the label request of the largest unknown classification category. This is done a total of five times, which represents a user query roughly every six minutes. This frequency of query is consistent with research on how often a user can be reasonably asked to provide this information (Hernandez et al. 2011). The generation of this occasional label request, although not explicitly realtime appropriate, was not found to increase overall running time beyond realtime.

5.3.   Non-Selected Classes of Artificial Intelligence Application

Many artificial intelligence methods are *not* appropriate for realtime selection. Each of these methods may make use of historical data, may not adjust existing models of data dynamically, may not automatically respond to new types of data, or respond well to the

changing nature of data over time. It is useful to include, as a brief list, some of the forms of AI that are applicable to well-known problems, but which are *not* applicable to the problems addressed by this dissertation. A literature review of commonly available approaches (Koranne 2011) provides a roadmap to this section.

5.3.1. *Bayesian Approaches*

This section encompasses Bayesian Networks, Causal Networks, Probabilistic Networks, and other statistical approaches. Bayesian approaches to model construction rely on the construction of a probability map in order to create an optimal model. The creation of this model must take all historical data into account for model construction, rendering typical approaches unacceptable. As one author looking for realtime Bayesian solutions put it: "in general, both the *exact belief update* and *belief revision* are NP-hard" (Guo and Hsu 2002). One solution to this is the approximation of solutions, but the approximations are also mathematically proven to be NP-hard (Abdelbar and Hedetniemi 1998; Dagum and Luby 1993). It is possible, via problem transformation, to solve NP-hard problems in polynomial time, but they cannot be solved in the linear time required for realtime approaches (Woeginger 2003).

5.3.2. *Evolutionary or Genetic Approaches*

Evolutionary approaches have seen recently popularity in the AI community (Davis 1991; Haupt and Haupt 2004; Teoh et al. 2012). This class of solutions encompasses Genetic Algorithms (GAs), evolutionary programming, evolution strategies, genetic programming, particle swarm optimization, and other complex adaptive systems.

Evolutionary approaches, in their most generalized form, utilize an encoded model of a solution combined with a combination method, a selection method, and an evaluation function (Eberhart and Shi 1998). The evaluation function determines the 'fitness' of instances in the population of possible solutions, such that 'fit' instances may be selected and combined with other fit instances to create a new solution. This algorithm is applied iteratively. The determination of fitness (iterating through historical data points) in combination with the iterative nature (iterating through hundreds of possible solutions) of these approaches renders it impractical for real time constraints.

5.3.3. Expert Systems

There has been significant work in the creation of "expert systems", which use rule-based, case-based, context-based, cognitively modeled, or knowledge-engineered methods to emulate the decision-making ability of a human (Jackson 1990). In the realm of physiological sensor measurements, there are very few experts from which to construct a model, and the author is aware of none. Even if there were such experts present, it would be unlikely for their knowledge to transfer well between individuals or groups, for the reasons seen in Chapter 2. While an individualized expert system can be constructed solely from the datastream with automated analysis techniques (Trinh 2009), these methods still require the use of historical data, rendering them inappropriate for linear time application because of the problems presented with infinite data length.

5.3.4. Agent-Based Systems Approaches

Agent-based systems approaches fall into two categories. The first category is that of an expert agent, which interacts with other agents as part of its operation. This is a method by which to bring together the various sub-disciplines of the AI community (Jennings 2000). In an affective ITS, the reader may imagine a software agent that continuously informs an outside agent, such as a teacher, of the emotional state of the learner. While this approach is relevant, the construction of such an agent must be undertaken with another AI method. This type of category of approach is skirting the solution, rather than solving it.

The second kind of agent-based approach is that of a complex adaptive system (Holland 1992). In this type of system, the solution is modeled as the behavior of each of a number of software agents acting within an environment. The approach encompasses some of the genetic methods described early. Other examples are Ant Colony Optimiztion (Dorigo and Di Caro 1999), swarm intelligence methods (Beni and Wang 1993), and stochastic diffusion search (Beni and Wang 1993). This type of method is rendered inappropriate because of the computation time which it takes to arrive at a good solution. There are not proofs for the discussion of these computational times, as the algorithms are stochastic in nature, but experimental testing by the author has shown that convergence on a solution takes longer than the incoming frequency of data. This testing is confirmed by Martens et al. (2011), which identifies the need for real-time appropriate swarm intelligence models for data mining applications (Martens et al. 2011). The

creation of this type of solution, and its adjustment to semi-supervised knowledge, is outside the scope of this dissertation and left to future research.

5.3.5. *Reinforcement Approaches*

Reinforcement learning, like the other types of machine learning presented earlier in this chapter, covers a wide swath of AI methods. Artificial Neural Networks, Support Vector Machines, Monte Carlo methods for policy iteration, Q-Learning, and many others make use of this type of learning method (Sutton and Barto 1998). When an experimenter is able to define a solution, they can make good use of a knowledge-based approach. When an experimenter is able to describe fractions of a good solution, but not the entire solution, they can use agent-based and evolutionary approaches. When the optimal set of input/output mappings is unclear but outputs have a known desired value, a policy of "reinforcing" good solutions becomes attractive. At its simplest, reinforcement approaches rely upon a simulation of an environment, where an agent acts, and is given a reward. Gradient descent backpropagation with neural networks typifies this type of solution (Widrow and Lehr 1990). These solutions require both a model of the environment, a model of reward, and a method of iterating a solution over an amount of inputs. The process of iteration is inappropriate for a datastream of potentially infinite length, which renders it inappropriate for a solution to the real time datastream problem, even when modified for incremental changes.

5.3.6. Hybrid Methods

The types of hybrid methods are too numerous to mention here. An example of a hybrid method is the NeuroEvolution of Augmenting Topologies, which combines reinforcement-based Artificial Neural Networks with Genetic Algorithm approaches (Stanley and Miikkulainen 2002). This example becomes an impractical solution because of the nature of genetic (5.3.2) and reinforcement (5.3.5) approaches alike. Other hybrid learning methods include neural methods for establishing case-based reasoning, genetic clustering, agent-based clustering, regressive linear programming, and simulated annealing (Abraham et al. 2009). Each of these methods is not appropriate because one, or the other, form of its hybrid approach makes use of historical data, does not establish new categories, does not adjust categories to new solutions, or does not respond to underlying changes of a category.

5.3.7. Discussion

When searching for machine learning methods that can deal with infinite data length, concept detection, concept drift, concept evolution, and lack of label availability, there are remarkably few items from which to select. In some cases, most of the features of an algorithm are available without significant modification. In this instance, the work done as part of this dissertation has made modifications to the underlying algorithm in order to render it appropriate to the problem. In other cases, such as is the case with Support Vector Machines, there has been misaligned field growth. Transductive SVMs make use of unlabeled data for future prediction (Zhang and Oles 2000), but the approach is too dissimilar from the 'online' or 'active learning' SVM approach which is capable of

realtime processing (Schohn and Cohn 2000). The research gap between online and transductive Support Vector Machines is an interesting problem, discussed in section 7.3: Future Work.

5.4. Selected Artificial Intelligence Classification Methods

5.4.1. Introduction

The first four items on Table 20 (infinite length, concept detection, concept drift, and concept evolution) are mandatory items for any selected algorithm. Failure to deal with these fundamental datastream problems renders the algorithm infeasible for processing of the realtime physiological data of Dataset #1 and Dataset #2. It is desirable, but not necessary, for the selected algorithm to naturally respond to the occasional presence of labels. The selected clustering method and the selected ART method do not do this (but have been modified to), while the selected methods of growing neural gasses and linear regression have this functionality encoded as part of their operation. As such, it was expected that the performance of the latter methods will be superior to that of the former. The remainder of this section discusses each selected method, and the modifications which occurred to address the problem. A checklist of features for an ideal AI algorithm is below, with semi-supervision being optional.

Table 21 – A checklist of features for realtime AI algorithms (semi-supervision is optional)

Infinite Length	Concept Detection	Concept Drift	Concept Evolution	Semi-Supervision

127

5.4.2. Clustering

5.4.2.1. DESCRIPTION

Clustering is the first method which is appropriate for real time analysis. As Jain (2008) says: "Organizing data into sensible groups is one of the most fundamental modes of understanding and learning" (Jain 2008). Clustering is a method of grouping data into a category, before establishing the other characteristics of interest to classification. Clusters are traditionally evaluated for fitness based on a distance metric. Clustering represents a standard approach for dealing with data of an unlabelled class, and is the baseline method attempted as part of this dissertation.

One of the most popular methods and simple methods of clustering is k-Means (Jain 2008; Steinhaus 1957). However, the k-means algorithm which attempts to simultaneously classify and separate clusters is considered NP-hard (Jain 2008). Expectation-Maximization (EM) has been a favored method for determination of the number of clusters in the Expectation step, and the classification of these clusters in the Maximization step (Fayyad et al. 1998). This EM process of guessing is computationally difficult portion of the EM process, rendering it inappropriate for real time, or processor-limited, applications. Modifications must be made by the experimenter to the initial algorithm in order to render it real time feasible. Examples of different approaches include online agglomerative clustering (Guedalia et al. 1998), or incremental updates to a previously established clustering base (Brawner and Gonzalez 2011).

5.4.2.2. REAL TIME APPROACH AND SELECTION

The clustering method examined in this dissertation was chosen for several reasons. Firstly, like all other methods throughout this chapter, this method was determined to meet the algorithmic specifications for real time signal processing. Secondly, clustering has been shown to be a data processing technique of wide applicability, and has been applied as a solution to a broad number of problems as a "first pass" examination (Jain 2008). Thirdly, this clustering approach has been proven relevant in the category of real time classification of physiological signals. Engler and Schnel attempted to validate this approach through the input of individualized, sequential, multi-day, workload measurements (Engler and Schnel 2012). Engler and Schnel found that the created model degraded over time due to individual day-to-day differences, but was highly (99%) accurate initially (Engler and Schnel 2012). This lends credence to the idea that this type of approach is valid for initial analysis, and could have positive results.

5.4.2.3. ADDRESSING THE PROBLEMS OF REALTIME DATA

The clustering approach taken in this dissertation responds to all four problems of real time datastream classification. The problem of infinite length is addressed through not saving historical data. As a new data point is presented to the algorithm, it is either assigned to an existing cluster or a new cluster must be formed. These clusters encode data. Although the list of clusters must be searched with each new point, this is kept to a minimum acceptable number for rapid performance. Initial experiments show that with unlimited cluster growth allowed, the number of clusters never exceeded more than 1% of total data.

The problem of novel concept detection is addressed through the creation of a new cluster for data which falls outside of known boundaries. Concept drift is addressed through the slight movement of the cluster centroid in the direction of the newly presented data. Concept evolution is addressed through the application of labels to a cluster as it is established, allowing the cluster to grow and move about the sampling space while still being identified as the same class. These solutions can be seen below in the descriptions of the algorithm.

5.4.2.4.   MODIFICATIONS MADE

The realtime algorithm was modified to deal with clustering labeled data. Mixed-classification clusters *are* allowed to be created. The clustering is built on the underlying data, with each cluster maintaining a list of the labels which have been associated with it. The classified label of the cluster is maintained as the majority class label of the points which helped to establish it.

This algorithm was modified for active learning through the creation of a label response policy. When the implementation is asked for a label, it responds with a known point belonging to the current largest unlabelled cluster, as detailed in the below.

5.4.2.5. INITIAL CLUSTERING ALGORITHM (NOT REALTIME APPROPRIATE)

For 'K' in a range determined by the experimenter
  Given a number of clusters 'K', select 'K' points randomly as the centroids for
  clusters
  Assign all objects in the dataset to the nearest centroid 'C'
  Compute the centroid of the objects now in 'C', move centroid to this point
  Repeat these steps until the centroids do not move (convergence)
Evaluate the goodness of the fit (typically via distance metric)
Continue to select a higher 'K' value until the fit is maximized

5.4.2.6. CLUSTERING ALGORITHM USED (INCLUDES REALTIME MODIFICATIONS)

For each new point, incrementally
  Compare each point to all known centroids
  If no cluster is within range of <vigilance parameter> this point is a new centroid
  Otherwise, move the matched cluster <delta parameter> in new point direction
  Check to see whether it is appropriate to merge this centroid with another
  Keep track of the number of points in these centroids, label if possible
  Keep track of the last point which modified this centroid

5.4.2.7. ADDITIONAL MODIFICATIONS MADE FOR SEMI-SUPERVISED ACTIVE LEARNING

When a label is requested
  Find the largest size centroid which does not currently have a label
  Return the last seen datapoint which modified this centroid

5.4.3. Adaptive Resonance Theory (ART)

5.4.3.1. DESCRIPTION

ART is a type of neural network architecture which classifies objects based on the activation of nodes in a structure. It was developed to classify data in a one-pass learning environment (Carpenter and Grossberg 1995), and has historic performance roughly equivalent to neural networks, but with significantly reduced training time. In its most

131

basic form, ART draws *n*-dimensional hypercubes around similar input patterns, where *n* is the dimension of the input data. Matched data are those that fall within the smallest hypercube or of the class of the closest available hypercube. Hypercubes are expanded to compensate for new data in accordance with parameter settings. The locations of the hypercubes are stored as weight vectors. Although sometimes viewed as a disadvantage, ART systems are capable of one-pass learning, which makes them appropriate for realtime classification problems. This feature of ART adds sensitivity to the input order of data. This is anticipated to assist in the classification of affective computing signals, where the order of the input data is relevant to the underlying affective signal, as shown in experiments with subliminal sensitivity (Carpenter and Grossberg 1987).

Initial ART implementations (Carpenter et al. 1991a) show that important events can be captured quickly, novelty classes can be detected and classified, and that dataset learning could be accomplished with half of the available data. This lends credibility to the hypothesis that semi-supervised learning will aid in the overall model quality by using a sampling of labels. Because of the self-stabilizing nature of the system, it is able to continue learning until all encoding memory is used, which is not likely to occur during a standard training session because of the heavily encoded nature of the weight vectors the established hypercubes. Furthermore, initial ART systems have been shown to respond well to 22-dimensional space (Carpenter et al. 1991a), which is comparable to the dimensionality of the dissertation dataset space, as discussed in Chapter 4. Recent experiments show this to be a reasonably valid technique for the classification of emotions from physiological signals such as GSR, heart rate, and respiration rate

(Monajati et al. 2012). Recent efforts have been applied to improve the overall speed of performance, which is relevant to the real time data problem (Castro et al. 2004).

5.4.3.2. REAL TIME APPROACH AND SELECTION

ART addresses the continuous nature of the real time data stream problem through knowledge encoding, which obviates the need for tracking prior datapoints. Similar to above clustering approach, there is still a need to iterate across all of the currently classified classes, but this small fraction of the overall data can be quickly processed, and does not expand significantly during runtime. ART addresses the problem of novel class detection through the creation of a new class if it falls outside a predefined threshold, and tracks developing classes through the expansion of the encoded hypercubes. Concept drift is addressed through the classification boundary modification in the presence of new data, which adjusts for concept evolution both with and without the presence of labels.

In short, ART presents an approach that is capable of rapid, on-line learning, with novelty detection, across high-dimensional data. Recently, they have been applied to a fragment of the underlying real time model construction problem (Cannady and Garcia 2001). There is significant evidence to believe that their performance will be more than adequate (Hoens et al. 2012).

5.4.3.3. MODIFICATIONS MADE

Modifications were made to the original algorithm for allowing it to deal with labeled data. The labels can be thought of as an overlay to the data. This is represented as a

property of the class, a 'map', which maps the index value of each hypercube to a known class. When asked for the classification of a cluster, or a point which belongs to a cluster, a map of (clusters->labels) is consulted, and the class label is returned to the algorithm. This does not change the performance of the unsupervised method, as the clusters are not used for construction of the ART structure.

There are several times when the known class label *does* comes into play, 1) at time of hypercube creation, 2) when an existing hypercube is matched within the vigilance threshold, and 3) when semi-supervised methods backlabel an existing hypercube. For 1), at time of creation, the label is mapped in the map. For 2), hypercubes of conflicting classes are disallowed existence, instead defaulting to creating a newer and smaller class of hypercube within the existing one. For 3), backlabelling serves to label each of the points within an existing class index to the label provided. Each of these modifications is detailed below.

5.4.3.4. ALGORITHM USED (REALTIME CAPABLE WITHOUT MODIFICATION)

For each new datapoint
Compute each neurons' weighted activation to it ($yi = \Sigma w_{ij}*x_i$)
Select the neuron with the highest activation
Test if this neuron is within vigilance (x_i fuzzyAnd w_x < vigilence)
If it is, Update the weights ($w_i = learningRate*x_i + (1-learningRate)*w_i$)
Otherwise, create a new category with x_i weights

5.4.3.5. MODIFICATIONS MADE FOR SUPERVISED LEARNING

Mixed-class clusters are disallowed existence
an overlay mapping of labels to clusters is maintained

134

5.4.3.6. ADDITIONAL PSUEDO-CODE MODIFICATIONS MADE FOR SEMI-SUPERVISED ACTIVE LEARNING

When adding any new datapoint, keep a map of the amount of data associated with w_i

When adding a new labeled datapoint, keep a map of the w_i's which have labels

When a label is requested, For all of the w_i in the map, look for the ones without label
The largest is unlabeled w_i is the winner
return the points associated with this largest, unlabeled classification category

5.4.4. Online Semi-Supervised Growing Neural Gas (OSSGNG)

5.4.4.1. DESCRIPTION

Neural Gas is a robustly converging alternative to the k-means approach of clustering that finds optimal representations based on feature vectors. These feature vectors construct a topographical map overlaying the data. An example of such an overlay map is included in Figure 15. This approach has its roots in Self Organizing Maps (SOMs) (Kohonen 1982) and Neural Gas topologies (Martinetz and Schulten 1991). Growing Neural Gas (GNG) is an incremental version of Neural Gas which is appropriate for datastream analysis (Holmstrom 2002), and was initially proposed by Fritzke (Fritzke 1995). Semi-Supervised GNGs are a further outgrowth of these methods to make use of unlabelled datapoints for classification (Zaki and Yin 2008).
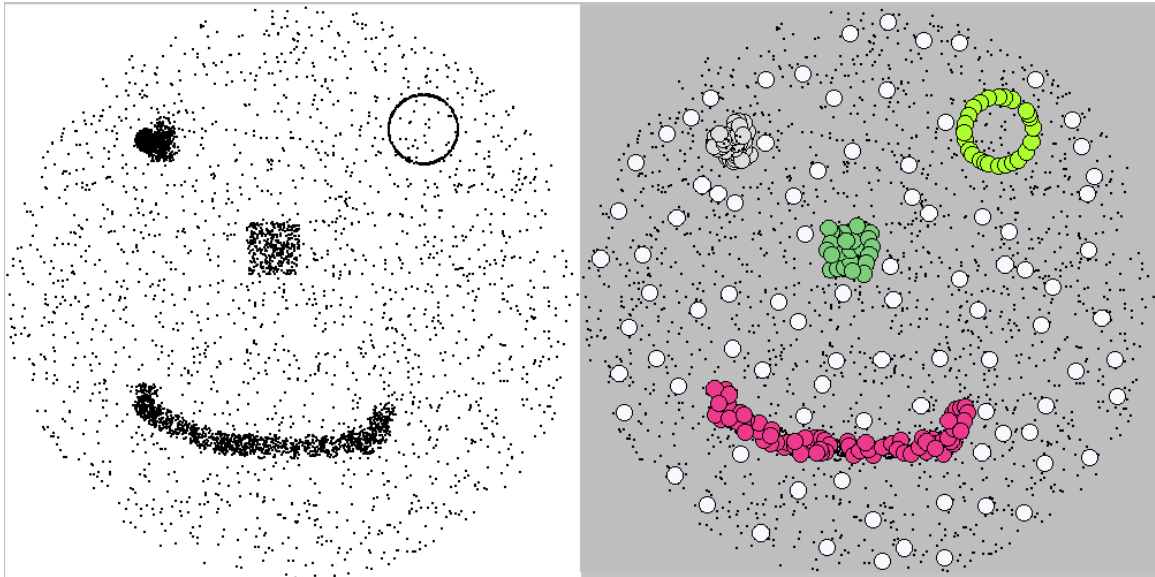
Figure 15 - GNG developed structure in presence of noised data. All data is unlabeled. Image displays raw data feed (left), and classification categories (right). Colors are representative of different classes. All data is unlabeled.

The GNG algorithm has additionally grown from research in competitive Hebbian Learning (Martinetz 1993), which learns from the collective excitation of neighboring regions. The primary portion of this algorithm is the connection of 'close' centers via 'edge' connections in response to a presented input pattern. These edge-connected items respond together to new input patterns. This idea is extended into GNG through the addition of finite nodes to represent the space, their subsequent edge connections, and their movement in the classification space. Updates to the network, although statistical, are performed with only local information. This sole use of local information is what makes it appropriate to the realtime, data-constrained, problem presented in this dissertation.

The initial semi-supervised algorithm for GNG (Zaki and Yin 2008) is an incremental improvement from the SOM, Neural Gas, Hebbian, and Expectation

Maximization (Moon 1996). The EM algorithm was used in this implementation to assign class labels to existing classes of unknown data (E-step), maximize the marginal likelihood of the parameter selection (M-step), and retraining the classifier for a new result. This approach is obviously not appropriate for realtime implementation, because of step-based solution iteration and non-linear time complexity of EM (Hofmann 2001).

Beyer and Cimiano have modified the initial algorithm to remove the dependence on the EM nature (Beyer and Cimiano 2011), making it appropriate to realtime problems. They present Online, Semi-Supervied, Growing Neural Gasses (OSSGNG) as a topographical mapping algorithm synthesized from the various contributing fields. They examine several metrics for determination of the establishment of clusters, and find that the minimum distance metric has the best performance on problems of interest. This dissertation uses the metric recommended.

There are several reasons why this implementation of neural gasses was chosen. The first is that it is representative of the field of Self Organizing Maps, which are sufficiently different from the clustering methods neural methods already discussed. Another reason is that the GNG method has had some research into semi-supervision (Zaki and Yin 2008), with favorable results. Finally, and most specifically, is the specific method of online, semi-supervised, GNG has been shown to outperform the semi-supervised method on a number of problems (Beyer and Cimiano 2011).

The OSSGNG approach addresses each of the fundamental problems with realtime data. Infinite length is addressed through the encoding of knowledge into a

connected series of nodes. Concept detection, drift, and evolution are handled through the occasional injection of new nodes and the aging of existing nodes. New node injection allows GNG methods to recognize new classes and associate with known structure, while aging nodes allows for the continuous evolution of concepts.

5.4.4.2. MODIFICATIONS MADE

The algorithm that was used as part of the experiments in this dissertation was obtained from contacting the researchers of the "Online Semi-Supervised Growing Neural Gas" paper (Beyer and Cimiano). This method is adapted to handling semi-supervised information in an online fashion, so no modification to the core routine was made. However, there has been substantial technical work behind the scenes to adapt it to the problem at hand. A brief and incomplete list includes making it a software library, generating a Python interface to the library, reformatting the structure of data that the algorithm expects, repairing major memory allocation errors, and making a thread-compatible library for performance-based data runs.

The most scientifically significant modification made was to add an active learning component. To the knowledge of the author, this is the first time active learning has been used within the GNG family of AI algorithms. The active learning component keeps track of the encoded knowledge that has been mapped to a label. The largest structure with an unknown label is considered to be the most interesting class and responds to the request for a label. The general assessment algorithm described in section 6.2.1 then assigns a majority-class label to these points.

This method was not found to have acceptable performance in requesting labels. In order to make this method computationally tractable, it was speeded up through the computation of a representational centroid, reducing the distance-based computations by over 100-fold. As a performance note, the list of points was also modified to be held in a list sorted with mergesort, which can be searched via binary search. This keeps the computational complexity during sort to O(n*log(n)) and the search complexity to O(log(n)). This is *not* appropriate for true realtime processing, but *is* appropriate for practicality, as reported later in Section 6.3.1:Timing. Briefly, an algorithm can make use of label requests if it does so infrequently. Note that this modification allows the structure of the data, rather than the labeled points, of the problem to dictate the classification boundaries. The algorithms and modifications are detailed in the below sections.

5.4.4.3. INITIAL PSEUDO-CODE GNG ALGORITHM (NOT REALTIME APPROPRIATE)

Present a new point and find the two closest items (s_1 and s_2)
Increment the age of all edges coming from s_1
Compute the local error of s_1 (error = squared distance from weight to input)
Move s_1 and its edge-connected nodes towards x_i in two fashions:
  Directly connected nodes: $\triangle w = e_b(x_i - w_{s1})$
  Indirectly connected nodes: $\triangle w = e_n(x_i - w_{s1})$
If s_1 and s_2 are edge-connected, set the age of the edge to 0
Remove all edges older than the maximum age, if a node has no edge now, remove it
If it is time to present a new node:
  Determine largest error node network from earlier calculated local errors
  Determine the largest error point node in this network
  Insert a node halfway between these two items, create edges, remove previous
Decrease all error by a factor, Alpha
Check for convergence (maximum network size, small adjustments, etc.)

5.4.4.4.   INITIAL PSUEDO-CODE SSGNG ALGORITHM (NOT REALTIME

APPROPRIATE)

Present the set of labeled data (LD) to the network, train only on it, label accordingly
Present an input from unlabeled data set (UD), x_j, with the previous distance metric
Label x_j according to the winning node, remove it from UD, enter it into the LD' set
Loop until UD set is empty
Present LD and LD' to evaluate performance

5.4.4.5.   OSSGNG ALGORITHM (USED)

Present a datapoint, finding the two closest items s_1 and s_2
If there is a missing label, assign a label based on the nearest item (unlabeled is possible)
Increment ages (detailed originally)
Proceed with GNG steps, do not loop to reevaluate

5.4.4.6.   ADDITIONAL PSUEDO-CODE MODIFICATIONS MADE FOR ACTIVE

LEARNING (FIRST REVISION)

When a label is requested, find the network of the largest unknown class
Look through the data to find points which align to the map
Request the labels of this list of unknown-class-mapped points

5.4.4.7.   ADDITIONAL PSUEDO-CODE MODIFICATIONS MADE FOR ACTIVE

LEARNING (SECOND/USED REVISION)

When a label is requested, find the network of the largest unknown class
Compute the centroid of this node-created network
Find and request the label of the point closest to the centroid

5.4.5. *Vowpal Wabbit (VW)*

The previous methods discussed typically favor accuracy from among the various engineering tradeoffs. Vowpal Wabbit is a software package implementation developed by John Langford at Yahoo! Research. The goal of this implementation and algorithm is to be *fast* and use *as little data as possible*, with the assumption that labels are available (Langford et al. 2007). It makes extensive use of gradient descent and multiple passes over the data to train a variety of encoded weight vectors. The background assumption to the initial problem of interest is that the data of interest is too large to process efficiently, and that rapid training is critical. This approach was developed specifically for large-scale search operations. The initial algorithm is described below.

5.4.5.1. ORIGINAL ALGORITHM

Start with $\forall i$: $w_i = 0$ Within the loop:
> *Get an example:* $x \in (\infty, \infty)$
> *Make a prediction:* $y = \Sigma_i w_i x_i$
> *Learn the Truth:* $y \in [0,1]$ *with importance I*
> *Update the weight:* $w_i = w_i + 2\eta(y\text{-}y_i)I$
Repeat for specified number of passes or other convergence criteria

It is useful to note that the Vowpal Wabbit code has been optimized to be simple, fast, and flexible. The core idea behind this implementation is that data would be optimized for very rapid iteration and convergence. Each line of the above pseudo code does not depend significantly on the previous line, or on any previous data, and relies only on weight encodings. This makes the core algorithm capable of extensive caching, hashing, and scaling to multiple processors, computers, and servers. This is designed to function on datasets with large numbers of features and examples. For example,

Langford tested against a dataset with 10^9 features across 10^7 examples (Langford et al. 2009).

The implementation of the VW set is able to use a variety of loss functions to calculate the rolling error represented in the weight update, including squared, hinge, logistic, and quantile. This dissertation makes use of the Support Vector Machine Hinge Loss, as each of the models used in Dataset #1 and Dataset #2 is a subdivided binary classification problem (e.g. Bored or Not Bored). Hinge Loss has been shown to be preferred for the reasons that, for binary classification, it converges more quickly, results in less approximation error, and has better generalization performance in theory, when compared to logistic and squared methods (Rosasco et al. 2004). The hinge loss function can be represented as a function of the predicted class and weight, *V(w,y) = max(1-wy,0)*.

Much work has been done in the area of active and semi-supervised learning with linear regression models for the purposes of search optimization (Beygelzimer et al. 2010a; Beygelzimer et al. 2010b; Duchi et al. 2010; Hoffman et al. 2010; Langford et al. 2009; McMahan and Streeter 2010). The crux of this research has relied upon the ability to establish importance weights of various data, minimization of data passes, or time optimization. Operations of O(n*log(n)) have been obtained to establish the most significant categories of data.

Much of this research is not relevant to the topic of this dissertation, as importance weighting and regression-based approaches are not realtime-appropriate solutions for the reason that they make use of historical data. The availability of the

realtime constructed model is more significant than the time for labeling. However, a

realtime active learning approach has been implemented as part of this work, based on

the approach taken from Beygelzimer (Beygelzimer et al. 2010a).

5.4.5.2. SEMI-SUPERVISED, ACTIVE LEARNING ALGORITHM

Obtain an unlabeled data example
Calculate the resultant error
      $h_k = argmin(err(h, S_{k-1})$, *h belongs to currentHypothesis)*
      $h_k' = argmin(err(h, S_{k-1})$, *h belongs to currentHypothesis OR is miscorrect)*
Caluclate the probability of labeling by finding s in the below equation
      $G_k = error(h_k') - error(h_k)$
      $G_k = (c_1/\sqrt{s} - c_1 + 1)*\sqrt{C_0*log(k)/(k-1)})+(c_2/s - c_2 + 1)*C_0*log(k)/(k-1)$
Randomly determine if a label is needed with probability $P_L = s$
      C_0 *is a experimenter parameter, c_1 is $5+2\sqrt{2}$, $c_2 = 5$,*
      k is the data point number, s is $\varepsilon(0,1)$ which solves G_k

Note that the semi-supervised algorithm does not cope well with completely unlabeled

data. No adjustments to the encoded weight vectors will occur if the probability of

labeling a point is not able to find a point to label. As such, the performance of this

version of this implementation was not expected to perform well on the data of interest,

while the supervised and unsupervised approaches were expected to have good

performance. Initially, the C_0 parameter was set so as to use significantly *more* labeled

data than the other algorithms, but to assure partial convergence. The ideal number of the

C_0 parameter is *two*, as has been theoretically proven (Beygelzimer et al. 2010a), as was

set during testing.

5.4.5.3.   ADDITIONAL MODIFICATIONS MADE

Few modifications were made to the basic implementation aside from significant software development technical challenges such as running Unix-oriented, C++-coded, programs in Windows-based, python-scripted environment. The current implementation of VW supports 78 command line parameters to modify, tweak, and report performance. A yearly tutorial is given in order for new users to understand the wide variety of settings that this implementation uses (Langford et al. 2010; Langford et al. 2007). It was found unnecessary to invent further complications to configuration.

There were two classes modifications made to adapt the above method to the problem of this dissertation. The first and largest modification was made to support a very incremental version of online learning. There were two forms of this adjustment. The first was to alter the loss function to one which did not require gradient descent and convergence. Coupled with this modification, the learning rate was modified to be adaptive in order to respond dynamically to the incoming data. The class of modification was added to support the occasional labeled data point in accordance with the active learning research (Beygelzimer et al. 2011).

5.5.   Conclusion

There are many different methods, and an entire field, dedicated to the most fundamental problem in machine learning: creating meaning from data. At this time we have discussed a clustering paradigm, a neural network approach, a graphical model, and a linear regression technique. Each of these four algorithms is selected as part of the state

of the art in their respectively fields, and each approach is sufficiently different from the others so as to warrant pursuit. All fundamental approaches covered in modern literature reviews (Jain 2008; Jain et al. 1999; Meireles et al. 2003; Quah and Sriganesh 2008; Tsai et al. 2009) are covered as part of this dissertation, which significantly limits the search space for an alternative approach. Modifications were invented for algorithmic adaptation as well as semi-supervised and active learning. At the initial time of writing, it is fundamentally unknown which, if any, of these approaches to model construction would be the most successful. A discussion will follow the results and comparisons based on the successes or failures of these algorithms.

6. RESULTS AND COMPARISON

The results of the experiments conducted are reported and discussed in this chapter. Prior to the presentation of numerous graphs of results, the initial benchmark comparison is presented in section 6.1. Following this, we discuss the general evaluation algorithm, how it averts the problem of contamination of data and labels over the course of a data run, and how the results are generated. Experimental adjustments, preliminary testing, and the running parameters are briefly discussed in Section 6.3. Finally, the discussion in Section 6.4 presents the questions, answers, and reasoning to the experimental questions addressed by this dissertation. These are summarized in Section 6.5, with conclusions and future work discussed in Chapter 7.

6.1. Initial Benchmarking

Before a discussion of the results of the testing of the various the algorithms, it is useful to discuss the initial models of comparison. These models represent the best effort of other researchers with the "infinite" time available in offline approaches. Each of these models is additionally constructed with all of the data, and with all of the true class labels. With all data, all labels, infinite time, and well-reasoned research approaches, these models represent the gold standard against which to compare our online, realtime models developed as part of this dissertation. It is not expected that an online model with significantly constrained time, limited data, and limited label availability will be superior in performance to these benchmark models. This represents the trade-off of accuracy for availability previously discussed in Chapter 3 of this dissertation.

Table 22 presents these initial benchmarks for experimentation, as created by the offline experimenters. Dataset #2 does not yet have benchmark models, so a quality-based comparison is impossible at this time. In the absence of a metric provided by the original experimenter, models created for Dataset #2 will be evaluated using the same AUC ROC metric as for Dataset #1. The AUC ROC metric used as part of the model evaluation in Table 22 is explained in further detail next in Section 6.1.1. The reader should note that no model for the Anger state was successfully created by the Dataset #1 offline experimenters.

Table 22 – Finalized Results Dataset #1 (Low-Cost Sensors)

	EmoPro Measures			ABM Measures		
	Anger	Anxiety/Fear	Boredom	Engagement	Distraction	Workload
Classification (AUC)	NA (<0.6)	.83	.79	.80	.81	.82

6.1.1. Area Under the Receiver Operating Characteristic Curve

The Area Under the Curve (AUC) of the Received Operating Characteristic (ROC) is a standard measure of the success of a modeling approach (Hanley 1989; Hanley and McNeil 1983). This metric is computed in the manner described in section 0. Generally, the AUC ROC measurement in binary classification problems places equal importance on each classification. It is designed to penalize simple majority-class classification boundaries (e.g. 90% of the data is from one class). In general, AUC metrics of greater than 0.8 are considered excellent, while classifiers lower than 0.6 are considered poor;

those scoring in the 0.2 range in between those values are considered to be acceptable but not optimal.

During the evaluation, described next in Section 6.2, each of the algorithms is iteratively queried for its computed label of each datapoint, and this is compared against the true label of the point, from the "ground truth" measure described in section 4.3 and Table 12. This is performed with a fractional amount of the data, on a per user basis, in order to generate the graphs seen later in this chapter.

6.1.2. *Full Results Located in the Appendices*

As part of this dissertation, several types of model creation algorithms are evaluated. Each of these algorithms is capable of realtime processing of the data. Each algorithm uses supervised, semi-supervised, and unsupervised labeling schemes for data analysis. As discussed in Chapter 5, four algorithms are compared (clustering, ART, GNG, VW). As such, for this dataset, there are 72 models which are created and discussed – the combination of six models (i.e. Anger, Fear, Boredom, Engagement, Distraction, and Workload), three types of labeling (i.e. supervised, unsupervised, and semi-supervised), and four algorithms (i.e. ART, clustering, VW, and GNG). In the same way, one model, with three labelings, and four algorithms is created for Dataset #2. Rather than discuss these 84 (74+12) models separately, they are discussed in summary within Section 6.4. The full models are presented within APPENDIX C, and organized by set of results, rather than by research question.

6.2.   General Evaluation Notes

Each algorithm is compared fairly against each of the other algorithms through the use of library functionality. Each implemented algorithm described in Section 5.4 adheres to a programmatic standard for evaluation. This standardization is done for several reasons. The first is to make sure that the true class labels are always handled separately from the data, assuring that each algorithm is completely unable to garner extra information from the previous run, or from the labels. The second is to assure that each pair of algorithm and labeling scheme is given, explicitly, exactly the same information as to make decisions as each other pairing. The third is to provide an environment for testing future, or additional, algorithms on different datastreams. Before discussing the results of the experiment the reader should be assured of the fairness of evaluation. Sections 6.2.1, 6.2.2, and 6.1.1 describe the general algorithm used to evaluate all algorithms, how the impact of labels is evaluated and the evaluation metrics used.

6.2.1. General Evaluation Algorithm

The general evaluation algorithm that controls how evaluations are performed is:

METHOD_LIST = [ART, VW, GNG, clustering]

Initial setup, loading of data into a structure, loading of labels
Initialize all clustering algorithms
  For method in METHOD_LIST: METHODIMPLLIST.append(MethodInitialize())
  For each 10% of the data, labels:
    For each of [unsupervised, semisupervised, supervised]:
      For method in METHODIMPLLIST:
        evaluateMethod(method, data, labels, supervision)
        deleteMethodAndContainedData()

evaluateMethod(method, data, labels, supervision):
  switch(supervision)
  unsupervised: method.addUnlabelleddata(data), evaluate
  supervised: method.addLabelledData(data, labels), evaluate
  semi-: method.addUnlabelleddata(data), label 5 requests, evaluate

evaluate
  while method.labelRequest() returns points
    calculateMajorityClassOfPoints
    method.label(calculatedClassMajority)
  evaluateAgainstBenchmark   (AUC ROC)

Given that this is a general evaluation algorithm, it requires each of the realtime AI algorithms to provide a uniform amount of functionality. In some cases, this is standard functionality provided by the designer of the algorithm, as is the case with clustering. However, in some cases, as mentioned above, is it non-trivial to engineer a solution, as is the case with GNG. From the above general algorithm, each individual AI method must be able to accommodate the below functionality:

Init(params) – initializes the algorithm and does all required setup work
AddLabeledData – Takes datapoints/labels and inputs them one-by-one to the method
AddUnlabeledData – Takes datapoints and inputs them one-by-one to the method
Classification(point) – Returns the suspected class of the point
LabelPoints(points) – Labels all points to the algorithm (does not adjust classifications)
LabelRequest() – Returns points, suspected to be of the same, most interesting class
Evaluate (data, labels) – Returns the list of predictions and true classifications
Clear() – Deletes all data contained within the algorithm

Source code to each of the methods, the testing environment, and a template for future testing with Python 2.7.3 functionality can be provided upon request directed to the author. Each of the methods implemented was tested with a unit test, calling each of these functions on a dataset of over 200 points to determine overall classification ability, initial time-sensitive performance, and general assurance of the implementation.

6.2.2. *Assessing the Impact of Labels*

In the general algorithm for assessment, after each of the algorithms have classified all of the points in the dataset, each algorithm is queried for unlabelled classification preference (e.g. "what categories have unknown labels?"). It responds with a list of points which belong to a class or cluster of unknown label (e.g. "the category that has these points"). In the evaluation algorithm, each of these points is examined for its true class label. The majority label of this group of points is returned to the algorithm for classification (e.g. "the majority of those points have label '0', label them as such").

This cycle is repeated until the algorithm is able to compute a label, whether correct or incorrect, for all points. After all points are known to fall into a category, the algorithm is ready for evaluation. The newly labeled cluster can be evaluated for how

151

well it performs at representing the labels, which is impossible without obtaining a predicted value for all points, as is the case with unsupervised learning. An example of this is shown in Figure 16, and discussed next.
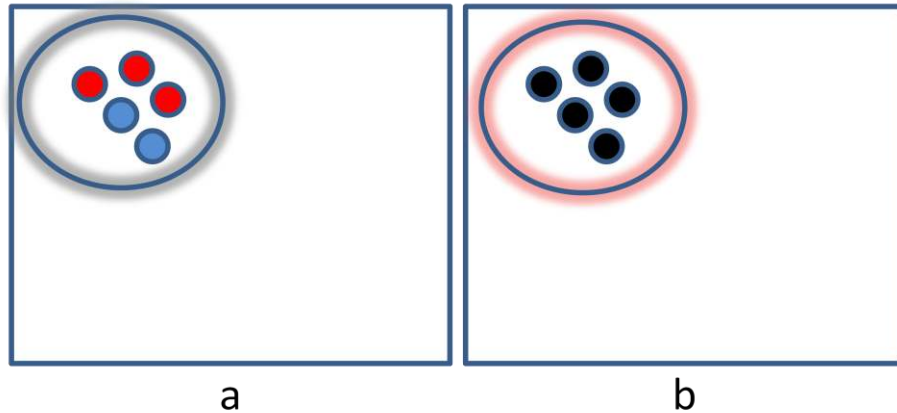
Figure 16 – Example of evaluation algorithm labeling an unlabeled cluster

To give a specific example, each of the algorithms frequently has a classification cluster which incorrectly maps points into a class of mixed labels. An example of this is a single cluster which contains 3 points of class '0' and 2 points of class '1', which are represented as red and blue in Figure 16a. Initially, an algorithm has classified five points as belonging to a cluster of unknown label (Figure 16a), as shown by grey cluster outline. The evaluation algorithm is aware of the label of each specific point (red or blue in Figure 16a). When the realtime AI algorithm is asked by the general evaluation algorithm for the label of this cluster, it is unknown; the AI algorithm responds with belonging points. The general evaluation algorithm assigns the majority-class classification to these points and gives them to the AI algorithm. The AI algorithm now

classifies this cluster as the majority-class (Figure 16b), and can now be evaluated according to modeling ability.

Note that a classifier that creates a *single unsupervised cluster* under this approach, will always, at minimum, classify 50% of the available data correctly through general evaluation labeling, representing a majority-class classifier. This corresponds to a ROC measurement of 0.5, which is the worst possible classification performance. Based on the finished algorithm, a correct/incorrect mapping of labels is created in order to evaluate the effectiveness of the method for model creation as described in Section 6.1.1.

6.3. Experimental Adjustments, Timing, Preliminary Testing, and Results

As with many AI projects, some amount of experimental adjustment is required for proper operation. Data might need to be reformatted, parameters may need to be set, labeling may change, and algorithms may need to be modified slightly. This section describes the initial testing and changes on both the datasets.

6.3.1. Timing

Firstly, this dissertation contends that it is possible to create useful realtime models. While it took many days to create all of the models used in the results section of this dissertation, this approach used 10 incremental models (one for each additional 10% of data, in order to graph performance over time) for each of 18+ participants and 84 models, resulting in ~17,000 models in total. Including the timing data for the construction of a single model allows the reader to easily verify that it is possible to

153

create a model in realtime. The creation of all three sets (supervised, unsupervised, semi-supervised) of boredom models is used as an example of the time taken to create an individualized model. Approximately 45 minutes of data were processed to produce the timing data summarized in Table 23. The represents the frequency of data response from each algorithm, summarized in Table 25. These timing data were generated using a single core of a 2.66 GhZ laptop computer.

Table 23 – Time, in seconds, required to create a single model of boredom. 2500 seconds of data were used.

Algorithm	Unsupervised	Supervised	Semi-Supervised
Clustering	0.062	0.058	0.312
ART	0.106	0.112	0.401
VW	0.045	0.046	0.056
GNG	99.816	73.787	120.634

Table 24 – Time, in seconds, required to respond to a single point. Anything over 0.3 is unacceptable.

Algorithm	Unsupervised	Supervised	Semi-Supervised
Clustering	6.4e-05	6.0e-05	3.2e-04
ART	1.1e-04	1.2e-04	4.2e-04
VW	4.7e-05	4.8e-05	5.8e-05
GNG	0.104	0.077	0.125

The creation of the Boredom model in Table 23 takes between 0.045 and 120 seconds, depending on the algorithm and labeling scheme. The fastest performance is consistently reported from VW, where each additional point presents only three multiplications, one addition, and one subtraction operation. GNG may have over 100 operations per datapoint, but the number of computations is finite and computationally linear. In the worst case, on modest hardware and a single CPU, a model for 45 minutes of data is

created in two minutes. All of the models except GNG were created in less than a second. This test experimentally proves what was theoretically proven in Section 5.4; that realtime algorithms are able to create models in real time.

6.3.2. Data Normalization (Dataset #1)

Preliminary analysis using Dataset #1 showed exceptionally poor results with both the ART classifier and the incremental k-means classifier on the first two users. Each of these scored a 0.5 AUC ROC value, regardless of the user and type of model. This was suspected to be because of parameter settings issues, as the recommended parameters were for normalized data (Brawner and Gonzalez 2011). Differences among individuals make the selection of a uniform parameter set difficult, if not impossible. Normalization on a per-user basis makes it possible to select a set of algorithmic parameters which are universally appropriate. The data for each user were normalized with respect to the user in order to allow each algorithm to operate within the same geometric space. In the real world, the maximum and minimum values for a user will not be known *a priori*. In such cases, the maximum and minimum values reported by the sensor can be used for normalization. The algorithm used to normalize the data is shown below, and was implemented prior to any results presented in this section.

For each user in listOfUsers
  Find the maximum and minimum value for the user: max and min
  For each oldDataPoint for the user:
    newDataPoint = (oldDataPoint-min)/(max-min)

6.3.3. Resolution Collapse (Dataset #2)

Initial runs using Dataset #2 data resulted in a number of problems relating to the size of the dataset. This dataset was initially collected with approximately 14,000 Hz resolution, which has grossly oversampled outputs. Changes in eye fixation and pupil diameter were not observed to change with this frequency. Thus, the dataset was downsampled 25% (only every 4th point) to simplify time and memory requirements, with a resulting resolution of 3,500 Hz. 3,500 Hz likely represents oversampling as well, but brings the total amount of data to manageable size. This brought the total data across 20 participants from approximately 300MB to 75MB. An example of a downsampled datapoint showing little variability is shown in Appendix B-3.

6.3.4. Running Parameters

Each of the four algorithms contains certain parameters which need to be set. In the evaluation of the approach of creating realtime models, these parameters were set to the recommended values of the respective papers. The parameters for the first batch of results are shown in Table 29. The parameter settings used in this research are derived from author contact or literature review. ART parameters are derived from initial literature (Carpenter and Grossberg 1995). Clustering parameters are drawn from author contact (Brawner and Gonzalez 2011) and standard library functionality (Jones et al.

156

2001). GNG parameters are drawn from author contact (Beyer and Cimiano 2011). VW

parameters were set as recommended by the various literature discussed in section 5.4.5,

and from online tutorial information (Langford et al. 2010).

Table 25 – Summary of initial parameter settings for tested algorithms

Algorithm	Parameter	Brief Description	Initial Value
k-means Clustering	Delta	Maximum amount of cluster movement allowed	0.1
	Vigilance	Maximum distance to be considered into a matching cluster	0.2
ART	Max Number Categories	Maximum number of categories which are allowed to be established	Unlimited
	Vigilance	Affects the possible classification distance for new points	0.75
	Bias	Small number for cluster activation to be above 0	0.00001
	Learning Rate	Amount of adjustment during each pass through the data (should always be 1 for one pass learning)	1.0
	Complement Code	Includes the inverse of a feature as an additional dimension.	False
VW	Loss function	The model of error introduced from a point. Square loss is used by default, but research indicates that hinge loss is better for a small number of passes.	Hinge
	Adaptive Learning Rates	Adjusts the learning rate downward (decreasing the importance) for points which have been previously observed	False

Algorithm	Parameter	Brief Description	Initial Value
OSSGNG	Epsilon Beta	See equations in 5.4.4. Amount of weight adjustment for connected node activation.	0.1
	Epsilon Nu	See equations in 5.4.4. Amount of weight adjustment for indirectly connected node activation.	0.0006
	Alpha	Error adjustment for a network	0.5
	Delta	Error adjustment for a neuron	0.0005
	Lamda	See equations in 5.4.4. Controls neuron addition rate.	300
	Maximum Node Age	How long neurons may exist	100
	Maximum Nodes	Maximum number of neurons	200

6.3.5. *Reduced Feature Set*

Only some of the features of the total datastream were used in the offline-created models of the original researchers, as originally shown in Table 18 and reprinted below as Table 27. In some of the experiments, as discussed in future sections, the reduced feature set was used as a comparison. Given that the offline modeling efforts made use of the same data, these comparisons may still be viewed as fair.

Table 26 – Summary and example of features used in each created model. Reprint of Table 18. No model of Anger above 0.6 ROC value was created with offline approaches.

	Appendix	Boredom	Distraction	Engagement	Fear	Workload
Alpha1	A-1				X	
Alpha2	A-1	X			X	
Gamma1	A-1	X			X	
Gamma2	A-1				X	
Delta	A-1				X	
Beta1	A-1				X	
Beta2	A-1				X	
Theta	A-1				X	
Attention	A-1				X	
Meditation	A-1				X	
Left Eye Pupil Diameter	A-5				X	
Heart	A-2		X	X	X	
Chair 1-4	A-4					
Chair 5-8	A-4		X	X	X	X
Motion	A-3			X	X	X
Alpha1Diff	A-6				X	
Alpha2Diff	A-6				X	
Gamma1Diff	A-6	X			X	
Gamma2Diff	A-6				X	
DeltaDiff	A-6				X	
Beta1Diff	A-6	X			X	
Beta2Diff	A-6	X			X	
ThetaDiff	A-6				X	
AttentionDiff	A-6				X	
MeditationDiff	A-6				X	
HeartDiff	A-6	X			X	
MotionDiff	A-6				X	

6.3.6. *Summary of Direct Data Analysis and Controls*

Before discussing results, the reader should be assured that the algorithms are presented

as they are discussed in the preceding chapters, and that a fair comparison is made. We

seek to compare two sets of models. The first set of models was created by other

researchers using offline AI algorithms in a generalized fashion. This is theorized to

show poor transfer to a population for the reasons discussed in Chapter 2. We created a second set of models that use online AI algorithms in an individualized fashion. In order to conduct a fair comparison of these approaches, all other variables which do not relate to individualization or online approach should be held constant. Additionally, the reader should be assured that the algorithms perform as theorized.

Windowing approaches, filtering, feature extraction, combinations of features, and creation of a new datastream from a kernel are some techniques that are commonly used for boosting algorithmic classification quality (Guyon et al. 2006). None of these approaches is taken in this dissertation in order to isolate independent variables from controls. All models created as part of this dissertation have the same inputs as the offline models created by other researchers, which renders a fair comparison.

In order to conduct this comparison fairly, this dissertation uses the same metric of quality as the original researchers, as discussed within Section 6.1.1. A single evaluation algorithm was created to give each algorithm exactly the same data, using the same function calls for each algorithm, as discussed within Section 6.2. Each algorithm is shown to perform in realtime, as theorized in Chapter 5 and as directly measured and confirmed in section 6.3.1. Individual normalization, as an experimental variable, was changed slightly, as discussed in 6.3.2. These actions have created a framework for the unbiased discussion of performance and these are presented next, within Section 6.4.

6.4. Experimental Results

In this section, the research questions and results are presented and discussed. Each research question is discussed in this section, and the key findings are summarized in the summary sections 6.4.8, 6.4.15, and 6.5. These research questions are discussed in the list below, before moving to a discussion of the experiments:

1a.   Can a quality *cognitive* model be constructed with *fully supervised* realtime algorithms?

1b.   Can a quality *affective* model be constructed with *fully supervised* realtime algorithms?

2a. Can a quality *cognitive* model be constructed with *unsupervised* realtime algorithms?

2b. Can a quality *affective* model be constructed with *unsupervised* realtime algorithms?

3a. Do *semi-supervised* and active learning approaches improve *cognitive* model quality?

3b. Do *semi-supervised* and active learning approaches improve *affective* model quality?

6.4.1. Analysis of Quality of Model Outputs

The primary item of interest to realtime model creation is the goodness of fit of the model, over time, based on the AUC ROC metric and the previously established benchmarks discussed in Section 6.1. The x-axis of each graph presented in the results section is time, with each line corresponding to a measured evaluation. All evaluations are measured with the AUC ROC metric.

Three types of AUC ROC measures are taken: "all", "next", and "prev". The "all" ROC measure represents the ability of the model to correctly predict all of the data that has so far been presented. The "prev" measure represents the ability of the current model to accurately classify the most recently observed data. "Recently observed", in this instance, refers to the previous 10% of data. The "next" measure represents the ability of the current model to accurately predict the upcoming data. "Upcoming data", in this instance, refers to the next 10%. The measurements of these three items indicate whether a method is able to correctly model the data presented recently, in total, and in the future. The graphs presented in this section use these metrics, graphed or averaged over time, to determine the adequacy of each model. An example of which data are used to generate a measure of each of these qualities is shown, in Table 27.

Table 27 – Example of the meaning of the "all", "next", and "prev" measures of AUC ROC evaluative point when evaluated at 50% and 100%.

	Data presented for evaluation
Previous	10% of total data. Most recent data. Example for 50%: Data from 40-50%. Example for 100%: Data from 90-100%.
All	50% of total data. All data so far Example for 50%: Data from 0-50% Example for 100%: Data from 0-100%
Next	10% of total data. Next data, predictive. Example for 50%: Data from 50-60% Example for 100%: N/A

The graphs in the below sections represent the averages of qualities of each model over time for all test subjects. There are ten points where each algorithm is evaluated for goodness of fit, at each 10% of the data, with the final point being at 100%. As an example of what each evaluative point represents, the evaluative point at 20% for a

Boredom model produced via ART method will represent a ROC value, when given 20% of the data, on the ability to model that 20%, averaged across all users.

Multiple evaluation criteria (e.g. previous, next, and total quality), algorithmic methods (e g. clustering, ART, GNG, VW), and models (e g. Distraction, Engagement, Workload, Anger, Fear, Boredom) must be presented as concisely as possible to draw conclusions. For the sake of simplicity, these have been combined into a few two-by-two grids of methods which each contain three dimensions of trend lines for three models, when a clear trend is present among all data. This results in a low quality image which has easily observable trend. Graphs shown in this section are presented in higher quality, divided by result set, in APPENDIX C, but are shown in a compressed form for overall trend analysis and discussion within the below sections. Each of these graphs, when presenting all measures, uses one of two legends, depending on whether cognitive or affective models are created. The legends are shown below in Figure 17 and Figure 18, respectively.

——Distraction all	——Distraction next	——Distraction prev
——Engagement all	——Engagement next	——Engagement prev
——Workload all	——Workload next	——Workload prev
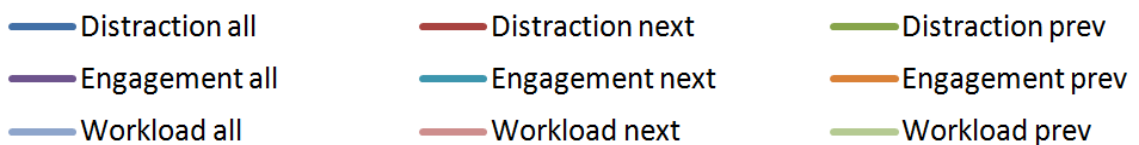
Figure 17 – Legend for Cognitive Models
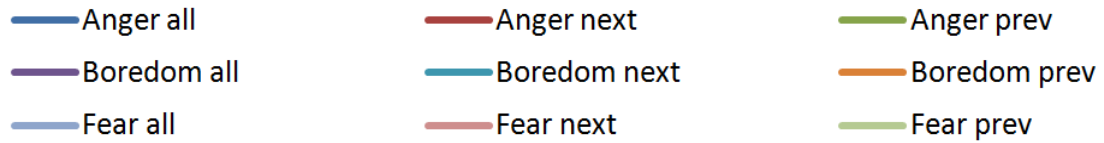
Figure 18 – Legend of Affective Models

6.4.1.1.   BENCHMARKS OF "ALL", "NEXT", AND "PREVIOUS" ADJUST IN

CONCERT

We theorized that a model may be useful for more than total model quality.   An

algorithm may be useful if it is able to model how states are anticipated to change or the

changes that have been recently experienced.   The "next" and "previous" measures of

ROC were created to observe whether this modeling behavior occurs.   In general, it was

found that these measures tend to reflect on another, and to adjust together. This is shown

clearly in Figure 19 and Figure 20, the graphs of supervised Anger models for

participants 4137 and 4111.   Once the three models are aligned at a single datapoint, they

adjust together, which is an indication that they are measuring a similar item.   These

participants were chosen for general model variability and typical example purposes, but

the trend is present for all participants.
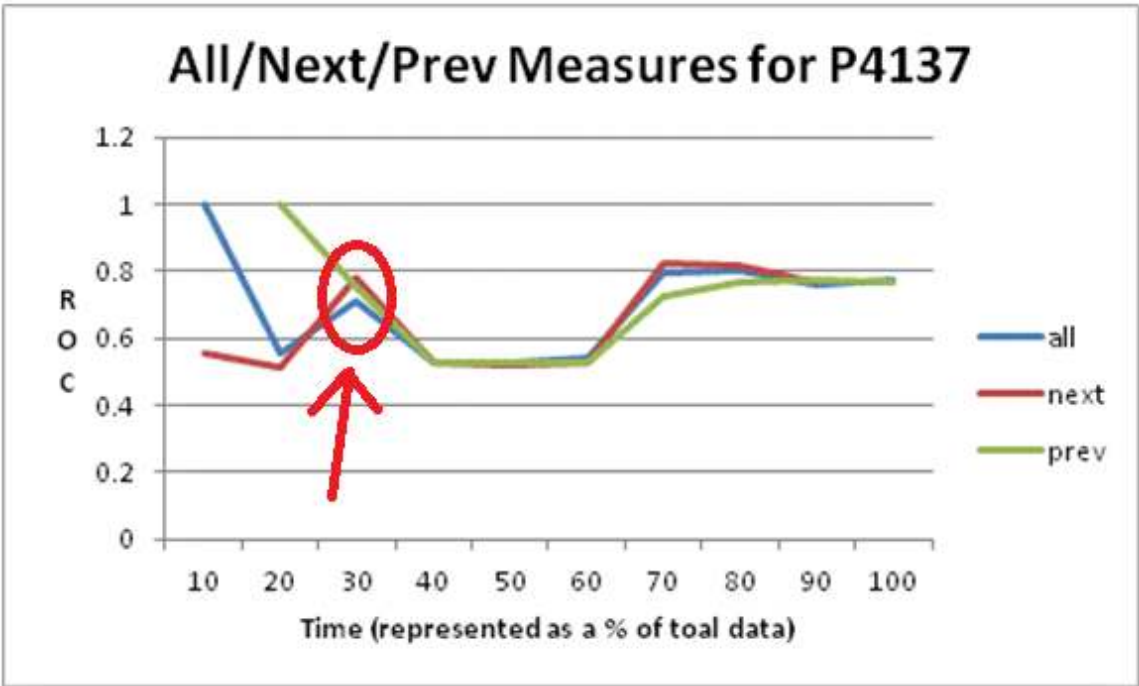
Figure 19 – All, Next, and Previous measures of model quality for Participant 4137. The three measures move in concert with each other after 30% of the data is presented.
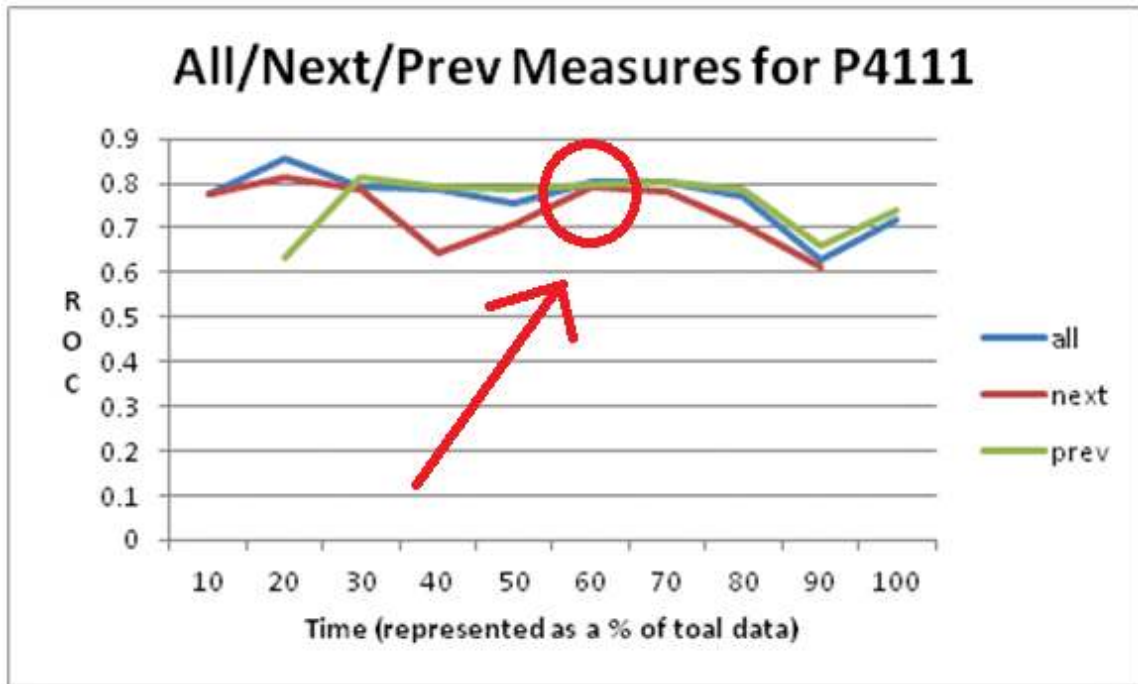
Figure 20 – All, Next, and Previous measures of model quality for Participant 4111. The three measures move in concert with each other after 60% of the data is presented.

In situations where is it appropriate to showing and discussing only one metric, the metric which has the greatest informative value should be selected. The "previous" metric is selected for this functionality for several reasons. Firstly, this is the metric of the most recent state of the participant, which has the most value to an instructional system. Secondly, this metric has the tendency to be accurate longer than the others, to degrade slowest, and to improve the quickest. Finally, the measure of the ability to model the most recent student state is more instructionally interesting than the measure of ability to model all student states presented so far (all), or of the ability to predict the next student state (next). In these cases, the below Figure 21 shows the abbreviated legend for affective models.

167

Figure 21 – Abbreviated Legend of Affective Models

Some graphs will be presented and discussed in gridded format, while others will be presented and discussed singularly. Some figures will present all algorithms, while other figures will present only one. Some figures will present multiple variations of labeling scheme, while others will only present a single instance. In each case, the author has attempted to select the few, among multiple, variables which provide clearest distinction to the reader. In any of the cases, APPENDIX C shows graphically intensive measures of all models, algorithms, labeling schemes, and measures of quality. All of the figures presented in this chapter can be constructed directly from images in APPENDIX C, without direct access to the data.

6.4.2. Research Question 1a - Supervised Realtime Creation of Cognitive Models

The question that the discussion within this subsection, and the first question asked as part of this research, is "Can a quality *cognitive* model be constructed with *fully supervised* realtime algorithms?". In order to answer this question, models of Distraction, Engagement, and Workload were created using Dataset #1 data and labels discussed in section 4.4 using only the supervised portions of the methods discussed in section 5.4. Only supervised methods were used in order to construct an apples-to-apples comparison of realtime methods using labeled data to offline methods using labeled data.

Four methods, three evaluation criteria, and three models results in thirty-six dimensions to show. For the sake of simplicity, these are combined. Each graph shows the performance of three models and three evaluation criteria over time. Four such graphs are combined into one image of performance, shown in Figure 22. Higher quality images of these same data are presented in APPENDIX C.
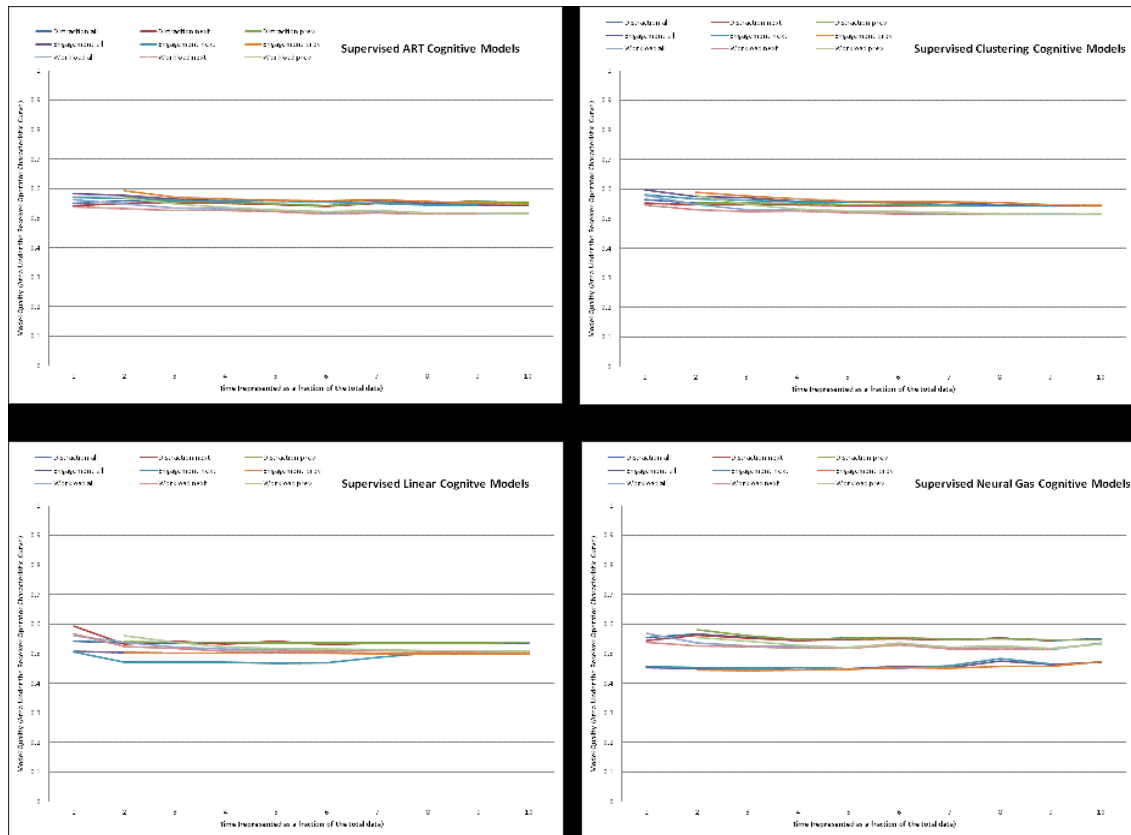
Figure 22 – Summary of realtime cognitive modeling ability with across all algorithms using the initial parameter settings

The Figure 22 graph for Distraction, Workload, and Engagement does not show model quality above 0.6, and are considered poor quality by AUC ROC measures. Trend data for all collected measure of ROC shows the same results. It is clear from visual inspection of Figure 22 that the models are universally poor for all labels and all methods. This leads the conclusion that it is *not* possible, via direct realtime AI method, to produce a model of cognitive state of acceptable quality with the algorithms selected. However, further testing has been performed as a part of this dissertation work to conclude this with certainty. This is described in additional testing of Sections 6.4.3-6.4.7

6.4.3. Research Question 2a – Unsupervised Cognitive Model Creation

The question that the discussion in this subsection seeks to answer is "Can a quality *cognitive* model be constructed with *unsupervised* realtime algorithms?". This would be the case if the addition of labeled information to the realtime algorithms was in conflict with the data being used to build the models, as discussed below. Figure 24 is used to draw conclusions for this experiment.

One must ask why we bother testing unsupervised algorithms when those supervised failed to produce acceptable models of cognitive states, as shown in the previous section. The answer is that there would be improvement in the cognitive models produced via unsupervised algorithms if the labeling information was in conflict with the underlying stream. This would occur if supervised algorithms were forcing the groupings of inappropriate clusters, where unsupervised algorithms were not. An example of this is shown pictorially in Figure 23. It is more likely that this occurs in the opposite manner, where labeling information prevents the formation of inappropriate clusters, but only occurs when labels match the underlying information.

a (supervised)
0.5 ROC value

b (unsupervised)
1.0 ROC value
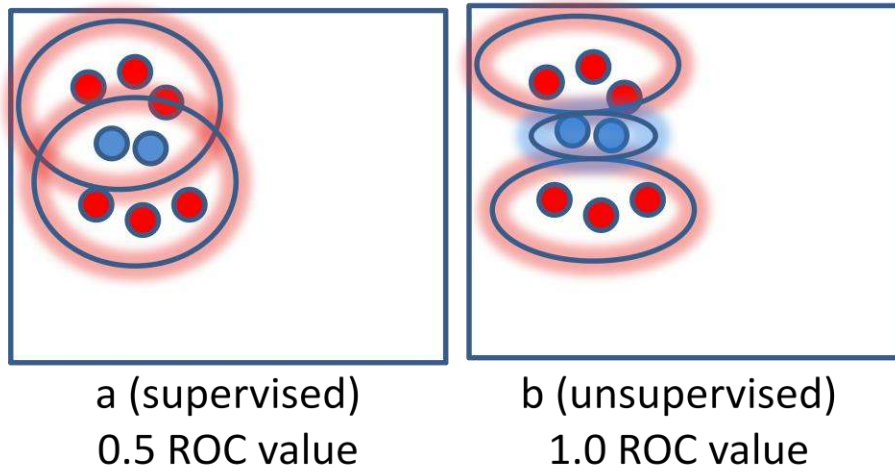
Figure 23 – Possible explanation for why an unsupervised algorithm (b) would outperform a supervised one (a). Phenomenon not observed for unsupervised cognitive models shown in Figure 24.

Figure 24 – Summary of realtime unsupervised cognitive modeling ability across all algorithms using initial parameter settings

Figure 24 shows the AUC ROC metrics used, and that none of them exceed a 0.6 threshold level. A visual inspection of Figure 24 indicates no improvement in model quality resulting from the lack of labeling information. It can be seen that the cognitive models created in realtime through direct AI approach are low in quality. It can be safely said that there is not conflict between the labels and the datastream that they represent based on two observed features: 1) the offline approaches were able to successfully model the problem, and 2) the removal of labeling information does not produce a higher quality model. Reasons for this and ways to mitigate it are discussed in Section 6.4.8.

The way to mitigate this problem is likely to be through customized feature extraction techniques. The used of these techniques is beyond the scope of this dissertation because it is not what was done for the offline models which are our comparison benchmark.

6.4.4. Research Question 3a – Semi-Supervised Cognitive Model Creation

In further attempt to isolate that labeling information is not the issue in the failure to create cognitive models, the semi-supervised versions of the algorithms were tested on cognitive model creation. Figure 25 shows the effect that semi-supervised algorithms have on cognitive models. The curves of Figure 25 are all consistent and stable – and all below the 0.6 AUC minimum for acceptability. The cognitive models show poor performance with both supervised and unsupervised methods, as seen in the previous two sections. Because of this, there is no reason to believe that they will benefit from semi-supervised modeling techniques, which label only occasionally. This is tested for the sake of completeness. It is confirmed that the semi-supervised algorithms indeed failed to create acceptable models of cognitive states, as observed in Figure 25.
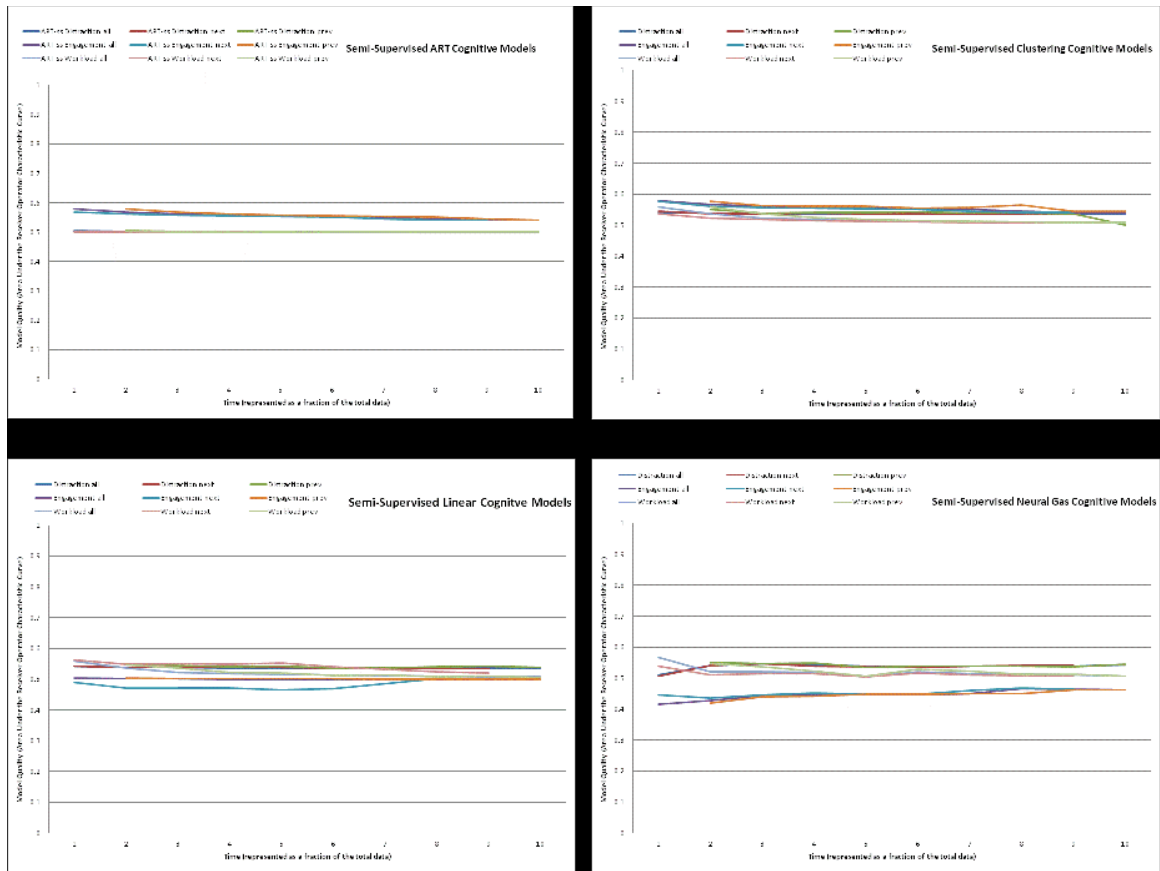
Figure 25 – Summary of realtime semi-supervised cognitive modeling ability across all algorithms using initial parameter settings

6.4.5. Revised Parameter Settings for Cognitive Models

It is possible reason for the failures above could be that the initially recommended and tested parameter settings were inappropriate for the problem of cognitive modeling. Fundamentally, the clustering and classification algorithms used in this dissertation match input data with output data. The most general solution to this problem is an input-output matching machine, where a given input results in nearest neighbor output. We considered that possibly, that the initial parameter setting represented too large of a

175

solution generalization from input-output matching, represented by creating too large of a class or cluster of data to be of use when creating cognitive models.

In an attempt to remedy this, the parameter settings were changed in order to establish a more fine-grained model of the labels, in the hope of creating a higher quality model. Generally, parameters were modified to create smaller groupings. These changes are presented in Table 28, and the reasoning for each change is discussed below.

Table 28 – Summary of parameter settings for tested algorithms for Results Set #1 (Dataset #1 cognitive and affective models)

Algorithm	Parameter	Brief Description	Initial Value	Revised Value
k-means Clustering	Delta	Maximum amount of cluster movement allowed	0.1	0.05
	Vigilance	Maximum distance to be considered into a matching cluster	0.2	0.05
ART	Max Number Categories	Maximum number of categories which are allowed to be established	Unlimited	Unlimited
	Vigilance	Affects the possible classification distance for new points	0.75	0.25
	Bias	Small number for cluster activation to be above 0	0.00001	0.00001
	Learning Rate	Amount of adjustment during each pass through the data (should always be 1 for one pass learning)	1.0	1.0
	Complement Code	Includes the inverse of a feature as an additional dimension.	False	True

Algorithm	Parameter	Brief Description	Initial Value	Revised Value
VW	Loss function	The model of error introduced from a point. Square loss is used by default, but research indicates that hinge loss is better for a small number of passes.	Hinge	Hinge
	Adaptive Learning Rates	Adjusts the learning rate downward (decreasing the importance) for points which have been previously observed	False	True
OSSGNG	Epsilon Beta	See equations in 5.4.4. Amount of weight adjustment for connected node activation.	0.1	0.1
	Epsilon Nu	See equations in 5.4.4. Amount of weight adjustment for indirectly connected node activation.	0.0006	0.0006
	Alpha	Error adjustment for a network	0.5	0.8
	Delta	Error adjustment for a neuron	0.0005	0.0005
	Lamda	See equations in 5.4.4. Controls neuron addition rate.	300	300
	Maximum Node Age	How long neurons may exist	100	50
	Maximum Nodes	Maximum number of neurons	200	300

The delta and vigilance parameters of clustering determine how much distance an established cluster can move in response to a new point and how "close" a new point must be to an existing cluster, respectively. Making these parameters smaller is an effort to make fewer adjustments to established clusters, and to classify fewer total points as

belonging to a single class. The specific parameter changes are discussed below in additional detail.

The vigilance parameter of ART is similar to that of clustering and was adjusted from a value of 0.75 to 0.25 in an effort to establish smaller overall hypercubes. Complement coding has been shown to aid in binary classification (Carpenter et al. 1991b), and was added to the problem in an attempt to boost classification accuracy.

Vowpal Wabbit has many tunable parameters, but only a few which are relevant to the purposes of realtime classification. The learning rate was adjusted to be adaptive in order to compensate for the lack of multi-pass learning. It was found to have no effect on the modeling ability, as shown in the later sections.

The OSSGNG algorithm has more parameters than the other algorithms because of the interconnections between the nodes which overlay the sampling space. Several adjustments were made in order to attempt to boost created model quality. OSSSGNG is the only algorithm which contains a model of 'forgetting' through the Maximum Node Age parameter. The age of nodes was shortened to adjust the algorithm to respond more rapidly to trends. Similarly, the total number of nodes was increased to model the space in a more finite fashion, with a modification to the Alpha parameter to punish more harshly for error. More nodes, with less memory, that are more error-sensitive were thought to increase model quality. This theory turned out to be accurate, as shown in the later in Section 6.4.13 and 6.4.14.

There is some trepidation by the researcher in creating smaller cluster sizes, as the unsupervised and semi-supervised models would be less transferable to the field. In ITS research, it is desirable to have known user states via labels. The reduction of cluster size in order to create finer models of performance results in a similar reduction in communication of state information for ITS use, which is worrisome. These adjusted parameter settings were used to recreate the supervised, unsupervised, and semi-supervised tests performed in 6.4.2, 6.4.3, and 6.4.4, respectively. The results of these tests are shown in Figure 26, Figure 27, and Figure 28, respectively.
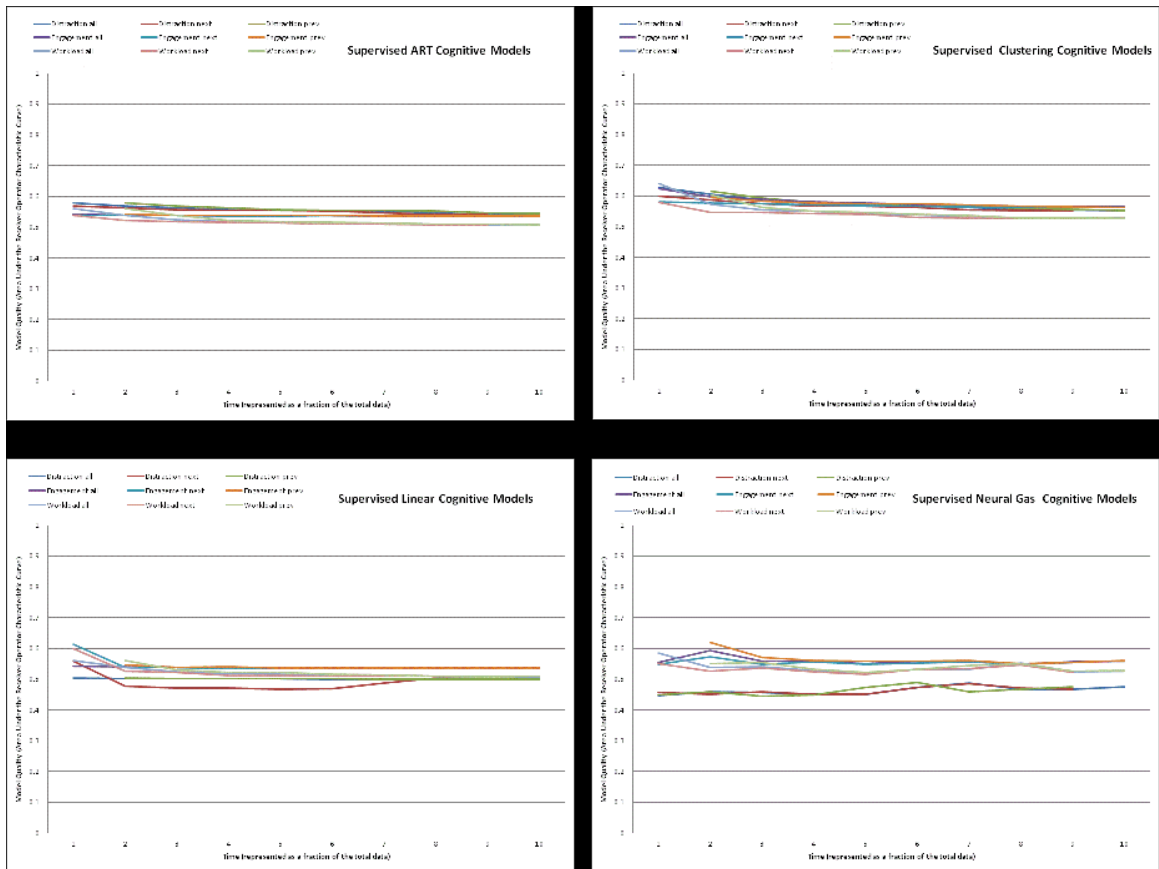
Figure 26 – Summary of realtime supervised cognitive modeling ability with across all algorithms using the revised parameter settings
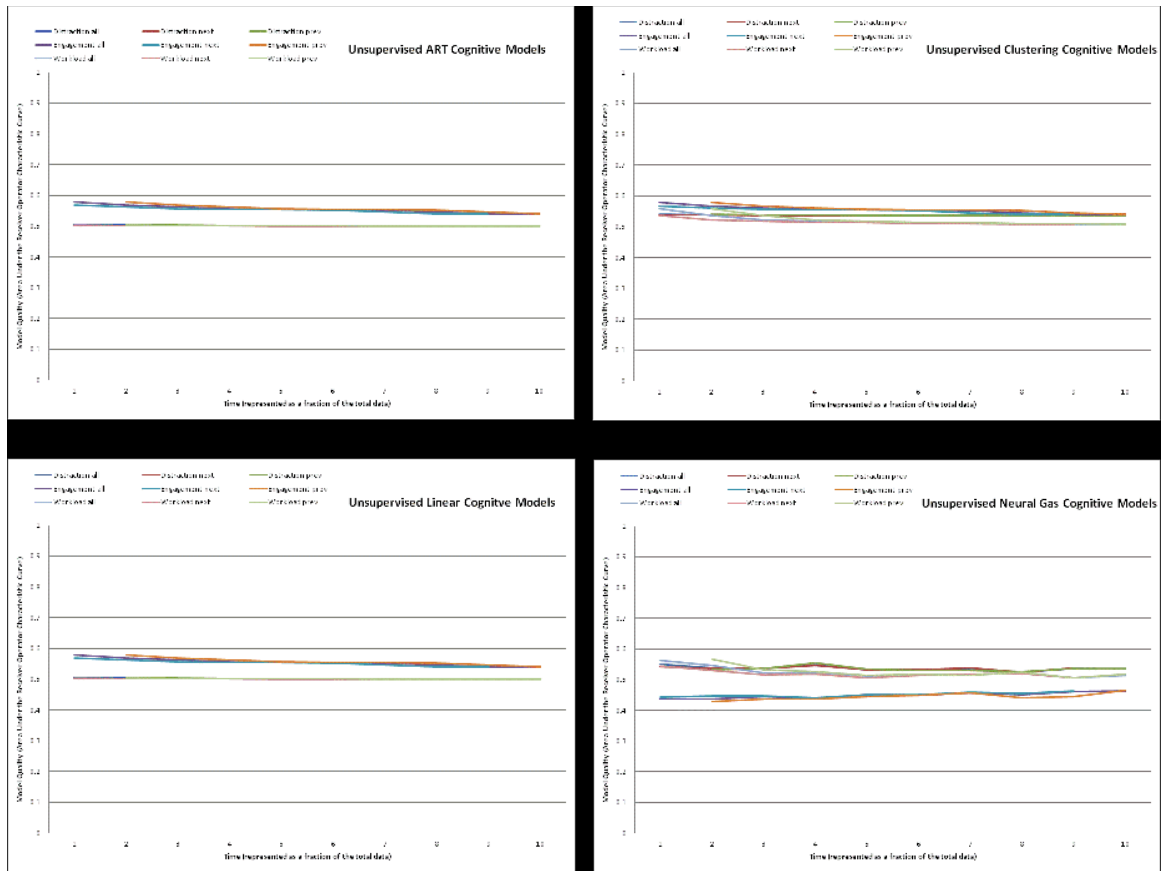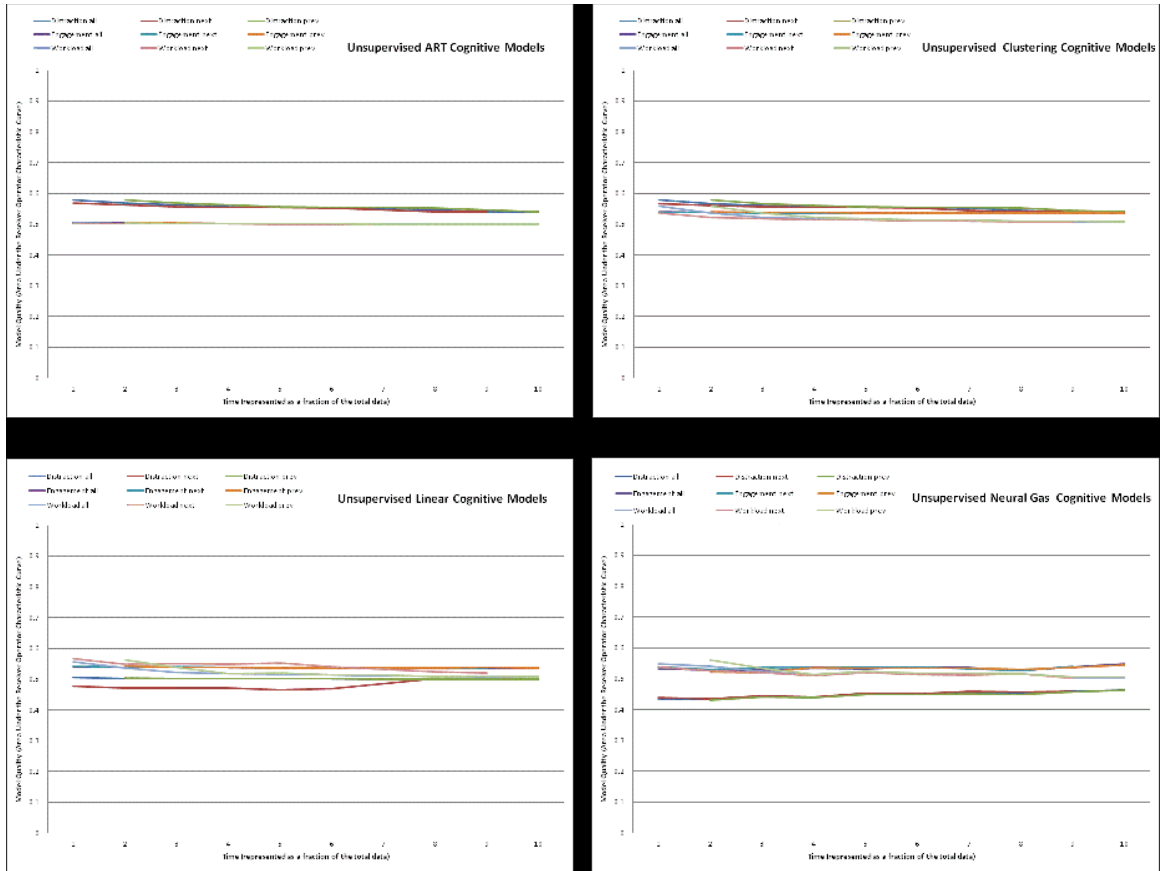
Figure 27 – Summary of realtime unsupervised cognitive modeling ability across all algorithms using revised parameter settings

180
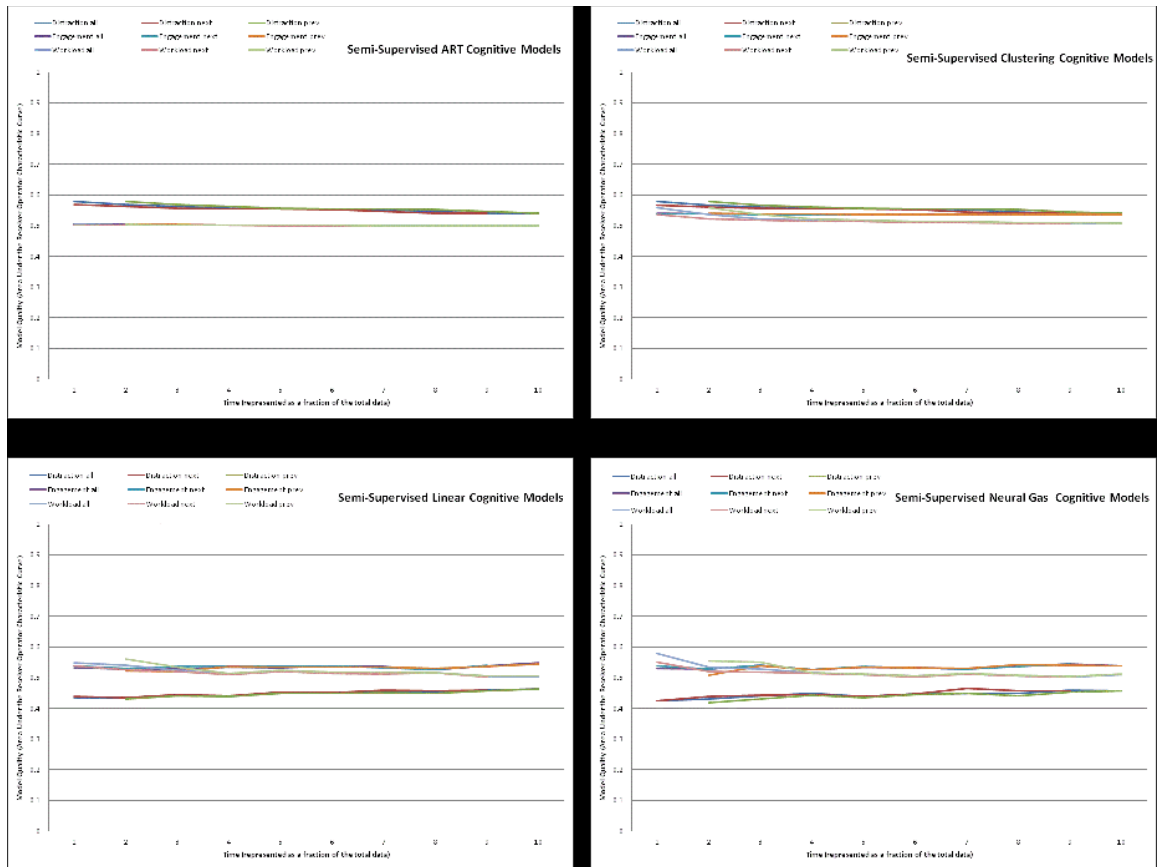
Figure 28 – Summary of realtime semi-supervised cognitive modeling ability across all algorithms using revised parameter settings

Unfortunately Figure 26, Figure 27, and Figure 28 show no improvement in the models across all algorithms and labeling schemes. All curves are consistent with each other, equally stable, and below 0.6 AUC, thereby indicating failure. The failure to fit these finer-grained models results in the theory that the data is *noisy* and that this noise was reduced by offline experiments, as discussed next.

6.4.6. *Reduced Feature Set Cognitive Models*

Linear regression modeling approaches were used in the original offline input data with the intent of developing equations which classify the inputs. The output of such an approach is a set of equations, using *some* of the input variables, which classify the input patterns into output classifications. These equations frequently do not use all of the input variables. The original regression models created for the benchmark offline models determined that several of the features of the datastream were unnecessary. Note that Frustration (an affective state) was the exception to this rule, as it used all of the factors reflected in the data. Given that these features were considered noise to the offline models, it was proposed that their removal might aid in overall classification quality for the online models.

The question that the discussion in this subsection seeks to answer is "When eliminating features determined to be of little use during offline analysis, is overall model quality improved for cognitive models?". Only some of the features of the total datastream were used in the offline-created models of the original researchers, as originally shown in Table 18 and reprinted below as Table 29. Figure 29 shows the effect that the removal of these features had on overall cognitive model quality.

Table 29 – Summary and example of features used in each created model. Partial reprint of Table 18. No model of Anger above 0.6 ROC value was created with offline approaches.

	Appendix	Distraction	Engagement	Workload
Alpha1	A-1			
Alpha2	A-1			
Gamma1	A-1			
Gamma2	A-1			
Delta	A-1			
Beta1	A-1			
Beta2	A-1			
Theta	A-1			
Attention	A-1			
Meditation	A-1			
Left Eye Pupil Diameter	A-5			
Heart	A-2	X	X	
Chair 1-4	A-4			
Chair 5-8	A-4	X	X	X
Motion	A-3		X	X
Alpha1Diff	A-6			
Alpha2Diff	A-6			
Gamma1Diff	A-6			
Gamma2Diff	A-6			
DeltaDiff	A-6			
Beta1Diff	A-6			
Beta2Diff	A-6			
ThetaDiff	A-6			
AttentionDiff	A-6			
MeditationDiff	A-6			
HeartDiff	A-6			
MotionDiff	A-6			
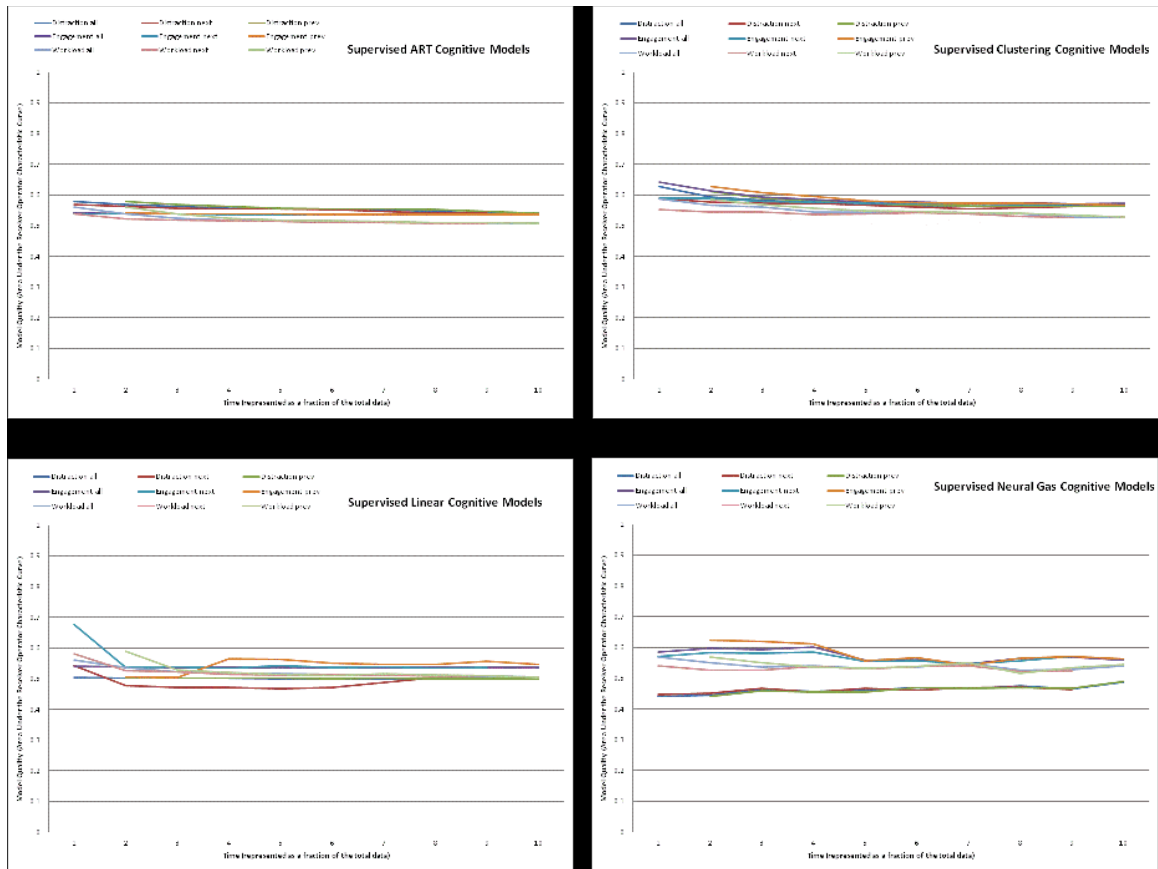
Figure 29 – Summary of realtime cognitive modeling ability across all algorithms using the revised parameter settings and reduced feature set for Dataset #1

When visually comparing Figure 26 to Figure 29 to gauge the effect that feature removal might have had on the produced cognitive models, it can be seen that there was no noticeable improvement gained from the elimination of "noise" data. It is clear that cognitive models created using the reduced-feature dataset do not achieve the minimum quality benchmarks overall. In the cognitive case, realtime model quality is too low to draw a conclusion on the effect of "noise" reduction. We suspect that the removal of features for the cognitive models had a negative effect, but there is not enough data to

back this assertion. It is certain that the feature removal did not *aid* overall model quality, but it is undetermined whether it *hurt*.

6.4.7. *Cognitive Model Generalization*

The question that the discussion in this subsection seeks to answer is "Does the method of creation for realtime *cognitive* models generalize to a *second dataset*?". We anticipate that it will not, given the poor experiences on the cognitive models of Dataset #1. Figure 30 shows the results of the experiment to test this hypothesis.
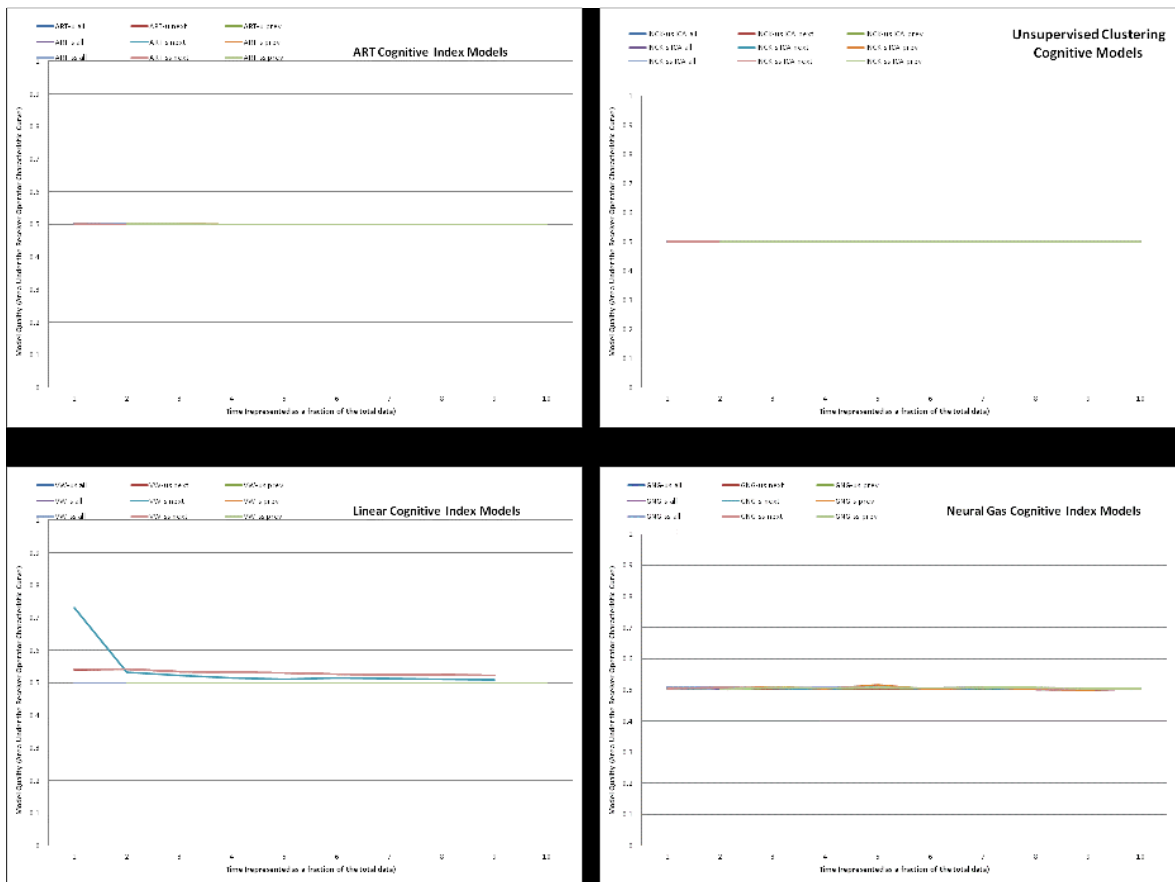
Figure 30 – Summary of realtime supervised, unsupervised, and semi-supervised cognitive modeling ability across all algorithms using revised parameter settings on Dataset #2.

185

Except for an early spike in the supervised VW-created models that quickly subsides, it is evident from a visual inspection of the curves in Figure 30 that the same algorithms used in Dataset #1 were equally unable to create acceptable models of cognitive states with Dataset #2. Given the poor results on the first dataset, this is unsurprising. Two sets of parameters (initial and revised), three labeling approaches (unsupervised, supervised, and semi-supervised), a revised input set, and four cognitive models (Distraction, Engagement, and Workload, ICA) have failed to produce reliable models. No further attempts to improve this situation were made. The summary of these experiments is included next.

6.4.8. Cognitive Modeling Summary

The initial three research questions, and subsequent three new questions, which were asked as part of this work are below:

1a. Can a quality *cognitive* model be constructed with *fully supervised* realtime algorithms?

2a. Can a quality *cognitive* model be constructed with *unsupervised* realtime algorithms?

3a. Do *semi-supervised* and active learning approaches improve *cognitive* model quality?

4. Does a change of parameter settings to reflect finer-grained clusters create higher quality models?

186

5. Does reducing the set of features to only the features used on cognitive model outputs create higher quality models?

6. Do the models approaches generalize to another dataset (Dataset #2)?

Quality realtime models of cognition were not able to be created as part of the work in the section which answers each of these research questions regardless of labeling scheme, parameter setting, feature set, or Dataset. In a fair comparison, where the same input data is presented to both the offline models and the online models, the offline approaches were able to create quality models where the online approaches were not.

The results of the cognitive modeling experiments on Dataset #1 and Dataset #2 are disappointing, as no viable cognitive model was able to be created during the course of this research. This is especially discouraging when one examines the contributing factors towards cognitive modeling in the previously created models, by others, using offline techniques. There are several hypotheses for the failure of the cognitive modeling algorithms. The first hypothesis was that the model quality was degrading over time because of bad parameter settings and was addressed through a modification of parameters to support smaller overall cluster sizes. The second hypothesis was that the algorithms were ineffectively classifying data that were noisy and was addressed through the creation of a set of limited-data results. The third hypothesis was that the approach was viable on another dataset and was addressed through testing on this dataset. None of these approaches were able to produce usable models of cognition.

In response to this lack of usable models of cognition, a series of additional parameter settings were attempted for ART. ART is the best-performing algorithm across both affect and cognition, and various values of the vigilance parameter were attempted. These were not shown to aid in cognitive model creation, but are included for completeness in APPENDIX D.

6.4.9. Research Question 1b - Supervised Realtime Creation of Affective Models

The question that the discussion within this subsection seeks to answer is "Can a quality *affective* model be constructed with *fully supervised* realtime algorithms?". In order to answer this question, models of Anger, Fear, and Boredom were created from Dataset #1 labels, discussed in section 4.4, using only the supervised methods discussed in Section 5.4. Only supervised methods were used in order to construct an apples-to-apples comparison of realtime methods using labeled data to offline methods using labeled data. The results required to draw this conclusions to this question are presented in Figure 31 and in the same manner as the previous section, and in Figure 32 using a arrangement figure. These figures are presented with only the "previous" measure taken, as the "all" and "next" measures confused the figure for discussion, as previously mentioned in Section 6.4.1. Full results are available within Appendix C-1.
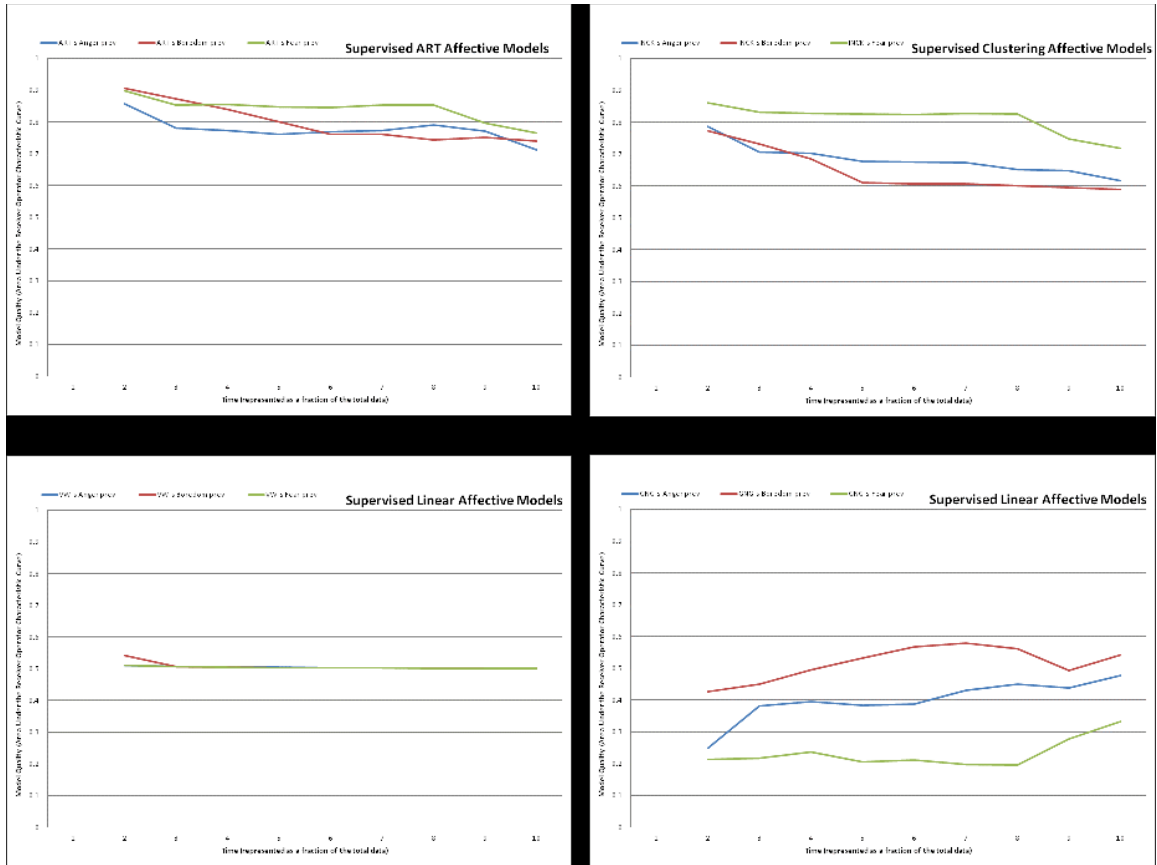
Figure 31 – Summary of supervised realtime affective modeling ability across all algorithms using the initial parameter settings
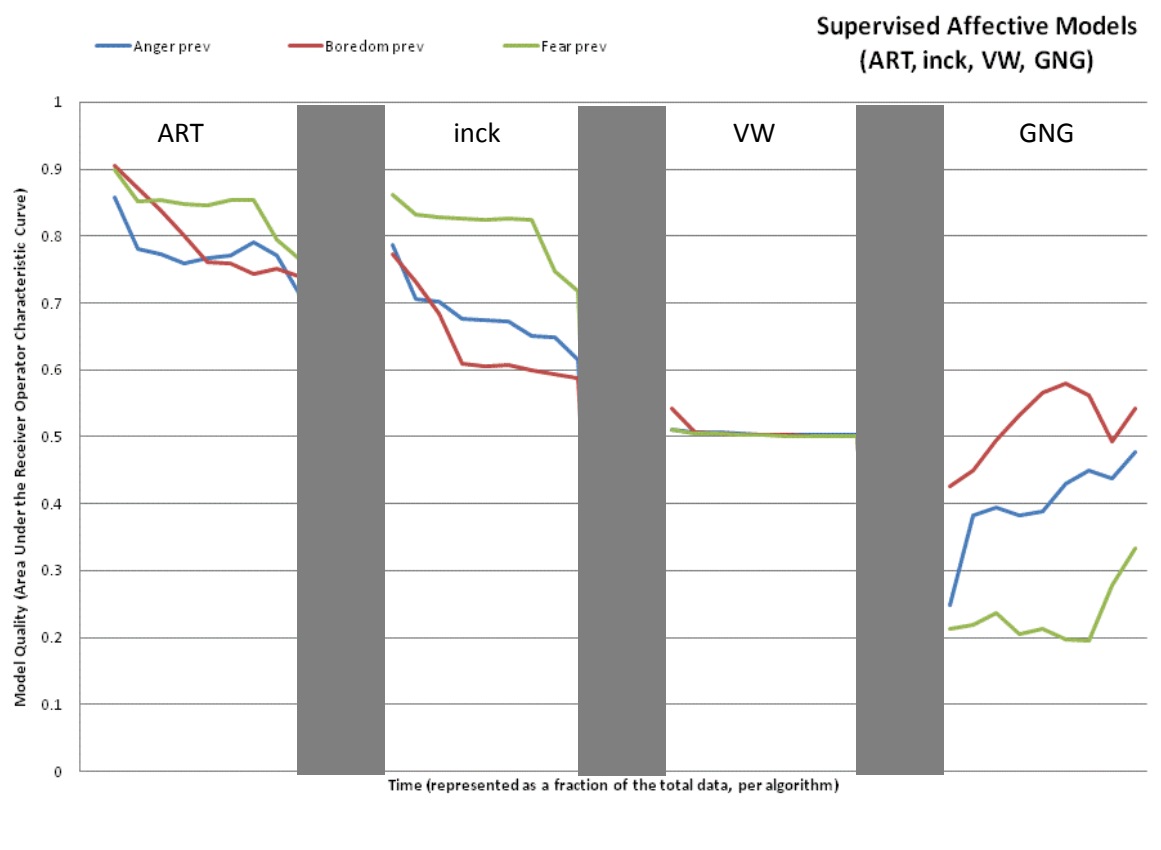
**Figure 32 – Affective modeling quality, as measured over time by AUC ROC on the most recent 10%
of data, with all algorithms in supervised fashion.**

Figure 31 and Figure 32 generally show that acceptable *affective* models *are* able to be

created in realtime. All of the methods for ART result in final model quality higher than

0.7. The majority of the clustering models also result in comparable quality. However,

from visual inspection of these figures, it is clearly evident that VW and GNG were at no

point in time able to exceed the 0.6 AUC threshold of acceptability. The complicated and

dynamic nature of the provided graphs call for a more in-depth discussion of the two

best-performing methods (ART and clustering). Table 30, Table 31, and Table 32 focus

this discussion on the ART models, while Table 33, Table 34, and Table 35 focus on the clustering models.

These six tables show the model performance for each user (vertically) across time (horizontally). The average model quality for each user is shown, bolded, on the right, as an indication of how well the user was modeled across the training session. Average model quality at a given percentage of the data is shown at the bottom. The average average model quality is mathematically equivalent whether it is taken from the user average or time average, and is used as an overall indication of quality for numeric discussion. As an example, the number 0.776 will be used as an indication of the quality of the supervised ART models of Anger using the initial parameter set, as presented in Table 30.

Table 30 – Anger model qualities with supervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.54	**0.947**
4133	0.58	0.58	0.58	0.58	0.54	0.51	0.68	0.69	0.50	**0.584**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	1.00	0.67	0.69	0.71	0.70	0.71	0.77	0.82	0.70	**0.753**
4111	0.63	0.81	0.79	0.78	0.79	0.80	0.79	0.66	0.74	**0.756**
4115	0.99	0.87	0.95	0.97	0.97	0.90	0.75	0.74	0.75	**0.878**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	0.78	0.64	0.65	0.62	0.70	0.78	0.74	0.77	0.79	**0.719**
4137	1.00	0.76	0.53	0.53	0.53	0.73	0.77	0.77	0.76	**0.709**
4101	1.00	1.00	1.00	1.00	1.00	1.00	0.81	0.50	0.50	**0.868**
4117	0.56	0.52	0.50	0.50	0.57	0.53	0.58	0.56	0.56	**0.545**
4102	0.56	0.56	0.56	0.56	0.66	0.50	0.73	0.78	0.50	**0.602**
4105	0.76	0.70	0.76	0.66	0.65	0.64	0.70	0.70	0.58	**0.682**
4104	1.00	0.68	0.85	0.86	0.86	0.87	0.87	0.87	0.87	**0.859**
4107	1.00	1.00	0.99	0.63	0.63	0.63	0.63	0.63	0.63	**0.749**
4106	0.63	0.63	0.50	0.66	0.67	0.64	0.68	0.69	0.70	**0.645**
4112	0.91	0.64	0.64	0.67	0.58	0.69	0.76	0.77	0.84	**0.723**
4132	0.87	0.75	0.67	0.70	0.75	0.74	0.74	0.72	0.56	**0.724**
Average	**0.857**	**0.780**	**0.772**	**0.760**	**0.768**	**0.772**	**0.790**	**0.771**	**0.712**	**0.776**
Total Usable (avg ROC >0.6):			17			Percent Usable:			89%	

Except for user #4133 and #4117, the average AUC for the entire time for 19 users are above the 0.6 acceptable threshold, and many are well in excess of 0.7. By any definition, these results indicate success in building a realtime model of the Anger state. This is especially relevant for the Anger state, as it was not possible to model this state with offline methods.

Table 31 - Boredom model qualities with supervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.96	0.92	0.92	0.77	0.70	0.71	0.68	0.72	0.58	**0.773**
4131	0.97	0.66	0.97	0.97	0.97	0.97	0.97	0.97	0.97	**0.939**
4127	0.63	0.67	0.60	0.60	0.53	0.51	0.62	0.51	0.65	**0.591**
4121	0.80	0.95	0.82	0.81	0.83	0.83	0.81	0.84	0.77	**0.829**
4111	1.00	1.00	1.00	0.75	0.73	0.79	0.83	0.74	0.79	**0.846**
4115	1.00	1.00	1.00	0.95	0.63	0.91	0.52	0.79	0.58	**0.821**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.91	0.78	0.74	0.73	0.76	0.78	0.80	0.81	0.82	**0.792**
4101	0.66	0.66	0.66	0.64	0.65	0.74	0.72	0.64	0.63	**0.665**
4117	1.00	0.67	0.52	0.67	0.72	0.51	0.56	0.58	0.59	**0.648**
4102	0.85	0.79	0.78	0.85	0.80	0.67	0.71	0.76	0.81	**0.780**
4105	0.80	0.84	0.84	0.69	0.62	0.66	0.63	0.62	0.66	**0.708**
4104	1.00	1.00	0.75	0.87	0.80	0.74	0.66	0.73	0.75	**0.810**
4107	1.00	1.00	1.00	0.75	0.79	0.74	0.79	0.79	0.79	**0.848**
4106	0.65	0.75	0.67	0.75	0.70	0.64	0.70	0.70	0.72	**0.699**
4112	0.98	0.88	0.88	0.88	0.78	0.78	0.65	0.60	0.58	**0.779**
4132	1.00	1.00	0.79	0.89	0.84	0.85	0.87	0.86	0.75	**0.873**
Average	**0.906**	**0.872**	**0.839**	**0.800**	**0.760**	**0.760**	**0.743**	**0.750**	**0.739**	**0.796**
Total Usable (avg ROC >0.6):		18			Percent Usable:		95%			

Table 31 shows the results for Boredom using the ART algorithm. The results for Boredom exceed the already excellent results seen for Anger. 95% of the subjects (only one exception) were able to be modeled at a AUC of >0.6, with most of them significantly higher.

Table 32 - Fear model qualities with supervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.55	0.55	0.55	0.55	0.53	0.51	0.68	0.70	0.59	**0.578**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	0.83	0.64	0.79	0.62	0.75	0.75	0.74	0.72	0.66	**0.722**
4111	0.61	0.53	0.52	0.52	0.52	0.52	0.52	0.54	0.52	**0.535**
4115	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4137	1.00	0.58	0.54	0.58	0.58	0.58	0.58	0.58	0.58	**0.625**
4101	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4117	0.53	0.51	0.53	0.51	0.51	0.53	0.54	0.54	0.63	**0.534**
4102	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.52	0.51	**0.892**
4105	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.82	0.82	**0.960**
4104	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.64	**0.960**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4106	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.54	0.51	**0.895**
4112	0.98	0.72	0.71	0.71	0.65	0.73	0.63	0.59	0.57	**0.698**
4132	0.58	0.67	0.59	0.60	0.52	0.59	0.52	0.56	0.52	**0.572**
Average	**0.898**	**0.853**	**0.854**	**0.847**	**0.846**	**0.853**	**0.853**	**0.795**	**0.765**	**0.841**
Total Usable (avg ROC >0.6):			15			Percent Usable:			79%	

Table 32 shows the Fear models created by the ART method. For Fear, although the variability was greater (only 15 out of 19 subjects were >0.6), the results for 15 were clearly excellent, with a final average of 0.841 AUC. This result, combined with the other results, indicate that ART was able to model the affect states very effectively.

Table 33 –Anger model qualities with supervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.58	**0.953**
4133	0.58	0.58	0.58	0.58	0.53	0.52	0.51	0.50	0.50	**0.545**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	0.96	0.61	0.54	0.53	0.57	0.58	0.63	0.63	0.64	**0.632**
4111	0.55	0.60	0.54	0.58	0.60	0.61	0.62	0.61	0.58	**0.589**
4115	0.92	0.69	0.69	0.66	0.66	0.61	0.50	0.50	0.50	**0.639**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	0.60	0.51	0.56	0.55	0.55	0.50	0.50	0.50	0.50	**0.531**
4137	1.00	0.71	0.69	0.58	0.53	0.56	0.61	0.58	0.61	**0.652**
4101	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	0.50	**0.835**
4117	0.56	0.53	0.56	0.50	0.50	0.49	0.54	0.54	0.51	**0.526**
4102	0.56	0.56	0.56	0.56	0.51	0.50	0.50	0.50	0.50	**0.531**
4105	0.49	0.51	0.57	0.62	0.64	0.64	0.63	0.63	0.62	**0.596**
4104	1.00	0.54	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.570**
4107	1.00	1.00	1.00	0.63	0.63	0.63	0.63	0.63	0.63	**0.750**
4106	0.59	0.52	0.50	0.50	0.53	0.52	0.50	0.50	0.50	**0.519**
4112	0.62	0.54	0.54	0.53	0.53	0.58	0.62	0.65	0.50	**0.570**
4132	0.51	0.53	0.50	0.50	0.50	0.52	0.53	0.50	0.50	**0.511**
Average	**0.787**	**0.706**	**0.702**	**0.676**	**0.674**	**0.673**	**0.651**	**0.648**	**0.616**	**0.681**
Total Usable (avg ROC >0.6):		9			Percent Usable:			47%		

The results for the clustering algorithm for the Anger model indicate a successful modeling process, but not nearly as effective as what was seen with ART in Table 30. Nevertheless, the total average AUC of 0.681 is in the acceptable level. The other mildly disappointing results is that only 9 of the 19 subjects (47%) scored an average AUC of >0.6.

Table 34 - Boredom model qualities with supervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.51	0.51	0.51	0.50	0.54	0.53	0.50	0.50	**0.513**
4131	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4127	0.58	0.58	0.57	0.57	0.54	0.54	0.51	0.51	0.51	**0.546**
4121	0.66	0.56	0.56	0.54	0.54	0.63	0.71	0.68	0.69	**0.620**
4111	1.00	1.00	1.00	0.74	0.58	0.60	0.62	0.61	0.53	**0.741**
4115	1.00	1.00	0.99	0.55	0.53	0.53	0.53	0.53	0.53	**0.686**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.97	0.66	0.51	0.53	0.51	0.51	0.51	0.51	0.51	**0.579**
4101	0.52	0.52	0.52	0.52	0.64	0.57	0.57	0.57	0.56	**0.553**
4117	1.00	0.67	0.52	0.55	0.58	0.55	0.51	0.50	0.50	**0.597**
4102	0.76	0.67	0.60	0.60	0.63	0.60	0.57	0.57	0.57	**0.619**
4105	0.53	0.51	0.51	0.54	0.62	0.62	0.56	0.56	0.52	**0.552**
4104	1.00	1.00	0.56	0.55	0.52	0.50	0.50	0.50	0.50	**0.627**
4107	1.00	1.00	1.00	0.61	0.58	0.59	0.59	0.59	0.59	**0.728**
4106	0.55	0.55	0.52	0.52	0.53	0.51	0.51	0.50	0.50	**0.521**
4112	0.61	0.63	0.60	0.59	0.58	0.61	0.58	0.55	0.55	**0.589**
4132	0.51	0.54	0.51	0.54	0.51	0.51	0.50	0.50	0.50	**0.514**
Average	**0.773**	**0.732**	**0.685**	**0.610**	**0.605**	**0.607**	**0.600**	**0.594**	**0.588**	**0.644**
Total Usable (avg ROC >0.6):			9			Percent Usable:			47%	

The results for the Boredom state with the clustering algorithm are roughly similar to those found for the Anger state of Table 33; they are good, but not as good as the results for the ART algorithm. Despite the fact that only 47% of models were worthwhile for participants, the total value of 0.644 indicates that they are usable, on average.

Table 35 - Fear model qualities with supervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.55	0.55	0.55	0.55	0.52	0.52	0.51	0.50	0.52	**0.530**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	0.60	0.58	0.52	0.51	0.51	0.51	0.51	0.51	0.51	**0.527**
4111	0.56	0.53	0.52	0.52	0.52	0.52	0.52	0.51	0.51	**0.525**
4115	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4137	1.00	0.54	0.54	0.54	0.54	0.54	0.54	0.54	0.54	**0.593**
4101	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4117	0.53	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.509**
4102	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.52	0.51	**0.892**
4105	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.56	0.50	**0.896**
4104	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	**0.946**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4106	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	**0.890**
4112	0.61	0.57	0.58	0.55	0.55	0.59	0.57	0.55	0.54	**0.568**
4132	0.52	0.52	0.50	0.50	0.50	0.50	0.50	0.51	0.50	**0.507**
Average	**0.861**	**0.831**	**0.828**	**0.826**	**0.824**	**0.826**	**0.825**	**0.748**	**0.718**	**0.810**
Total Usable (avg ROC >0.6):		12			Percent Usable:			63%		

The Fear model created with the clustering algorithm fared much better than the prior two models. The 0.81 average AUC value is excellent. However, the 63% usability number, while better than obtained for Anger or Boredom, is shy of what was obtained through the ART approach. The offline methods, using all available labeled data, created models of Anger, Fear, and Boredom of <0.6, 0.83, and 0.79 in quality, respectively (see Table 22), resulting in an ability to create two of the three models. A model of Anger was not successfully created through offline experimentation.

This can be compared with the ART ability to produce models of 0.776, 0.796, and 0.841 (for Anger, Fear, and Boredom) in overall model quality when using the recommended parameter settings. Supervised ART is able to successfully model, *using an infinitesimal fraction of the total data* at a time, with little overall degradation in quality. This fraction represents one datapoint, rather than the use of all datapoints, and corresponds to approximately 0.1% of the total for a participant, and a much smaller fraction when thinking about a model built from multiple participants. Clustering methods are additionally able to create models with overall quality greater than 0.6, with values of 0.681, 0.644, and 0.810. It is clear that fully supervised realtime methods can perform comparably to the fully supervised offline methods. The individualized ART models generally *outperform* their generalized offline equivalents in all cases, as shown clearly in Table 36, which compares the supervised results.

Table 36 – Summary of supervised ART (Table 30, Table 31, Table 32) and clustering (Table 33, Table 34, Table 35) when compared against the offline equivalents.

Model	Anger	Fear	Boredom
Offline	NA (<0.6)	0.83	0.79
ART	0.776	0.841	0.796
Clustering	0.681	0.810	0.644

The models produced using the online regression in VW and SOMs in GNG are not discussed in the above table as they did not reach sufficient levels of quality. The ART and clustering approaches taken in this dissertation clearly outperformed the VW and GNG approaches. Reasons for this trend are discussed next, before resuming the discussion of the various research questions.

6.4.10. Discussions of Specific Algorithms

Now that a few results graphs and tables have been presented, it is appropriate to discuss general trends among algorithms reflected in the remaining results graphs throughout this chapter, using the initial figures as the example. The first of these algorithmic trends is that the GNG methods do not obey the trends seen in this other algorithms of the data. The second is the performance of VW. These trends are discussed below, before returning to the discussion of research questions.

6.4.10.1. GROWING NEURAL GASSES BEHAVES DIFFERENTLY

Growing Neural Gas is a relatively new technique for pattern recognition. It has seen increasing use in the research areas of image recognition (García-Rodríguez et al. 2007) and topology learning (Prudent and Ennaji 2005). Our previous research has revealed that it responds well to the injection of uniform noise information (Brawner and Gonzalez 2011). Fundamentally, the GNG approach creates an overlay to the data which detects edges in patterns and forms the areas interior to the edges into clusters. The boundary edges clusters serve to identify unique groups of data among the dimensions of the input space.

When data are closely aligned in the sampling space, segmentation of the data becomes difficult. Figure 33 shows the classification of normalized raw EEG information via the GNG approach, where only five classes of data are established during one hour of raw data, with one class covering the vast majority of the sampling space. As a reference, a clustering approach similar to the one taken in this dissertation established

199

thirty classes of data on the same dataset, with approximately even division among them (Brawner and Gonzalez 2011).
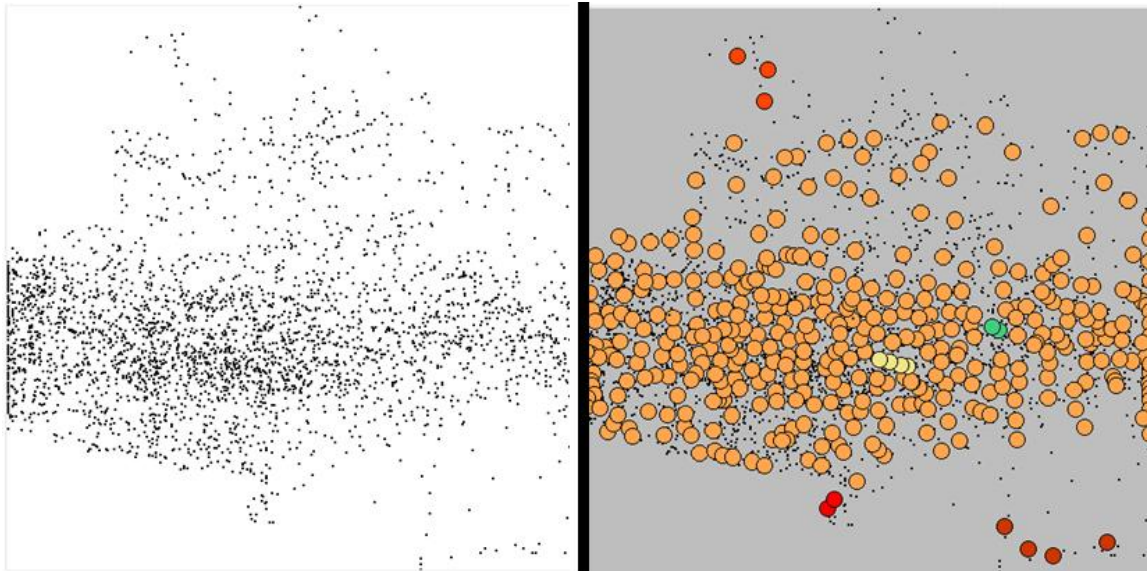
Figure 33 – Plot of normalized "Engagement" metric (x-axis) against "Short Term Excitement" (y-axis). Data is measured from the eMotive EEG Sensor using a slightly different GNG approach. For more information, see (Brawner and Gonzalez 2011). The left side of the image shows raw data while the right side shows classification categories. GNG is implemented in an unsupervised fashion, and creates one large cluster.

The graphs plotted in APPENDIX A show that the dataset has raw features that are not clearly segmentable over time. Additionally, the features have a tendency to move through the sampling space fluidly, leading to difficulty in establishment of classification boundaries. These two features of the data determine the approach of the GNG algorithm on the problem, leading to a general trend that the GNG approach establishes *one large classification cluster* of the entire sampling space. This large cluster grows until it has encompassed all of the data available, with few exceptions. The ROC measure for such a cluster is 0.5. While the GNG algorithm appears to "improve" in quality over time and

eventually reaches 0.5 AUC, it is a model of the baseline majority-class classifier, and does not produce usable models for any of the research questions of interest.

The observation that GNG does not produce usable models in any condition renders the safe removal of the approach from the discussion throughout this chapter. This phenomenon is surprising. The Online Semi-Supervised Growing Neural Gas (OSSGNG) models implemented by Beyer and Cimiano is the only approach in this dissertation which met all of the realtime AI algorithm checklist features shown in Table 21 (Beyer and Cimiano 2011). The consistently best performing algorithms were the ones we invented or most heavily modified for adaption to this work, rather than the approaches which were invented for the solution of this problem.

6.4.10.2. Vowpal Wabbit Underperforms

Each algorithm models a different AI approach. While GNG represents a topographical overlay of the data, VW represents an approach to linear regression modeling. VW adjusts weight vectors towards classes of labeled data, which increases a reliance on labeling information. When there are few states and feature sets in which to model, VW performs much better than the other algorithms.

New concept detection, however, has disastrous results in its overall performance. VW degrades to minimum performance quickly, and does not display any aptitude towards individual model recovery (as seen in the clustering and ART tables). The brittleness of the VW models is displayed through the remaining chapter. Although VW will have an initially higher performance standard, when compared to the rest of the AI

implemented in this dissertation, it will also have baseline performance for the longest period of time. The discussion of VW has been ignored in favor of discussion with ART and clustering, as the overall performance is lower, the models behave in more brittle fashion, the least amount of performance boost from labeling information is observed, and it was generally implemented for comparison against offline linear regression models.

6.4.11. *Research Question 2b - Unsupervised Affective Model Creation*

The question that the discussion within this subsection seeks to answer is "Can a quality *affective* model be constructed with *unsupervised* realtime algorithms?". Only unsupervised versions of the methods in Section 5.4 were used in this section, as they are the only version able to be modeled without the benefit of labels. If models of reasonable quality are able to be created without the use of labeled information, this would mark a significant extension to the original work, as models of users could be created without their direct knowledge or interaction, aside from sensor measurement. A realtime model created without labeling information is able to forego the stage of asking the user about their affective state, and instead use this time for training within the ITS. Figure 34 and Figure 35 show the initial results of this experiment in the same fashion as the previous section.
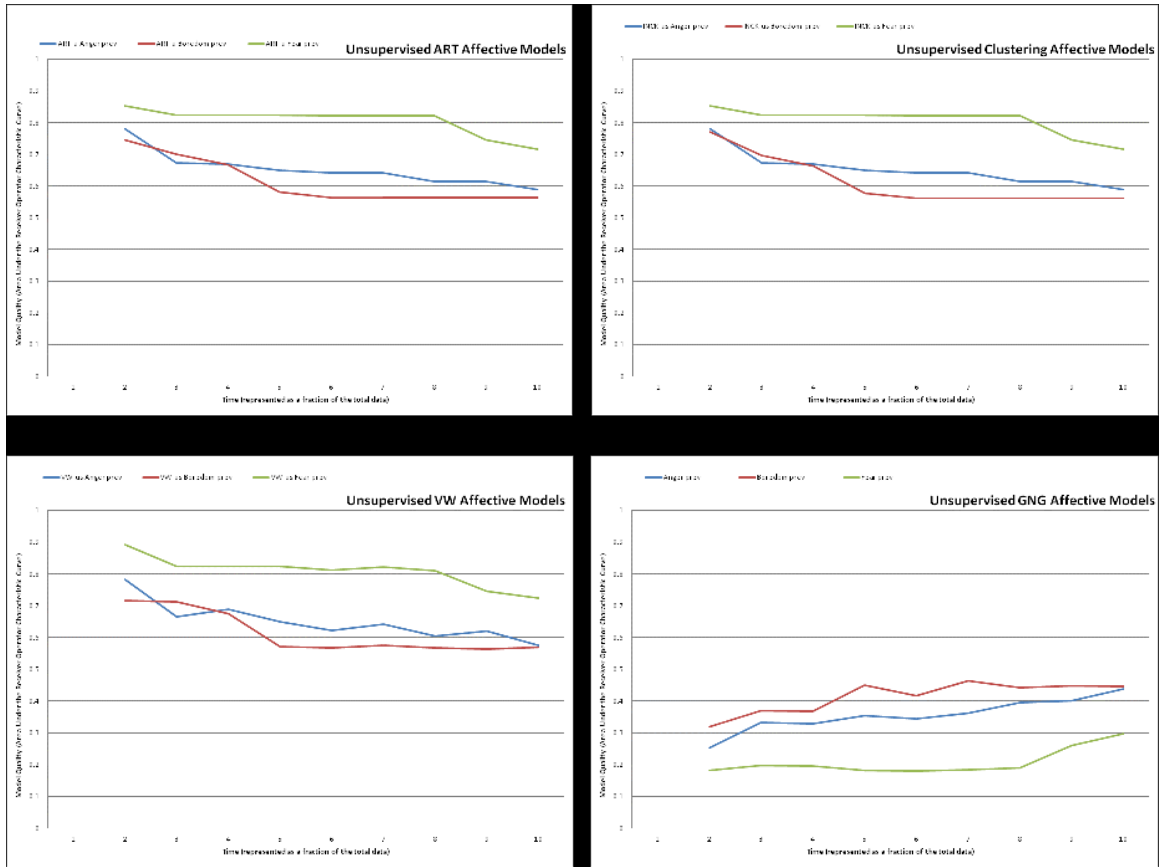
Figure 34 – Summary of realtime unsupervised affective modeling ability across all algorithms using initial parameter settings
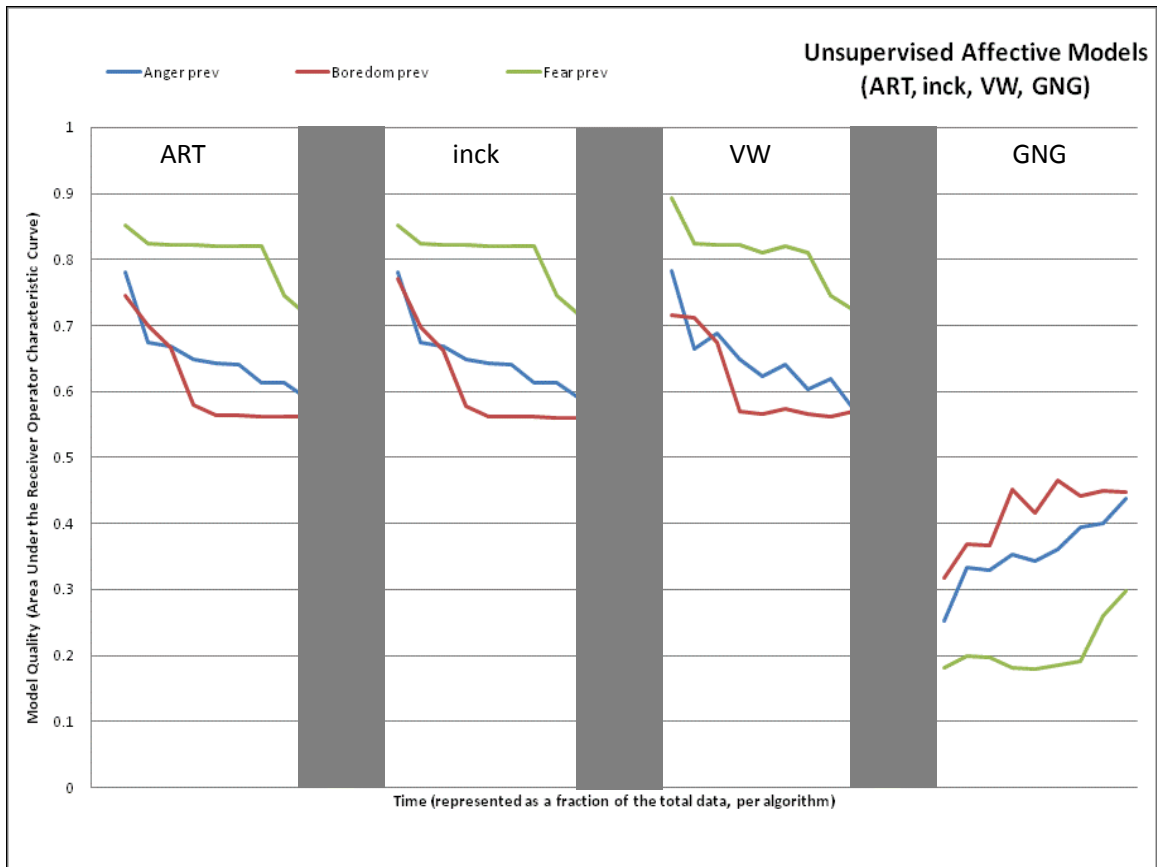
Figure 35 – Affective modeling quality, as measured over time by AUC ROC on the most recent 10% of data, with all algorithms in unsupervised fashion.

Once again, the performance of the models is difficult to grasp directly from visual inspection, and warrants a closer look into the results obtained. Table 37, Table 38, and Table 39 show the quality of ART created models without labeling information over time, while Table 40, Table 41, Table 42 show similar information for clustering. Table 43 summarizes the results of these tables. These tables show the numeric information for all models and all participants in order to conduct logical comparison of the resultant degradation in model quality.

Table 37 – Anger model qualities with unsupervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.54	**0.949**
4133	0.58	0.58	0.58	0.58	0.52	0.50	0.50	0.50	0.50	**0.540**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	1.00	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.560**
4111	0.56	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.509**
4115	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.559**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.506**
4137	1.00	0.56	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.570**
4101	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	0.50	**0.835**
4117	0.56	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.510**
4102	0.56	0.56	0.56	0.56	0.51	0.50	0.50	0.50	0.50	**0.531**
4105	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4104	1.00	0.54	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.565**
4107	1.00	1.00	1.00	0.63	0.63	0.63	0.63	0.63	0.63	**0.750**
4106	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
Average	**0.781**	**0.674**	**0.668**	**0.648**	**0.642**	**0.641**	**0.614**	**0.614**	**0.589**	**0.652**
Total Usable (avg ROC >0.6):		6			Percent Usable:			32%		

Overall, a model of Anger is able to be created from the unsupervised version of the ART algorithm which is usable, on average. This averagely usable model is only usable for a total of 32% of the participants, due to the nature of the modeling approach. The offline approaches to a model of Anger were *not* able to produce a model in quality greater than 0.6 with supervised labeling approaches, while the online models without labels *are* able to create a model with 0.65 in quality, effectively outperforming the offline models.

Table 38 – Boredom model qualities with unsupervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
4131	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4127	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4121	0.52	0.52	0.52	0.51	0.51	0.51	0.51	0.50	0.50	**0.510**
4111	1.00	1.00	1.00	0.75	0.51	0.50	0.50	0.50	0.50	**0.696**
4115	1.00	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	**0.673**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.511**
4101	0.52	0.52	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.508**
4117	1.00	0.67	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.579**
4102	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.558**
4105	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.506**
4104	1.00	1.00	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.615**
4107	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4106	0.53	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.508**
4112	0.51	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	**0.507**
4132	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.505**
Average	**0.745**	**0.701**	**0.666**	**0.580**	**0.564**	**0.563**	**0.563**	**0.563**	**0.563**	**0.612**
Total Usable (avg ROC >0.6):		7			Percent Usable:			37%		

Boredom model qualities using the ART algorithm in unsupervised fashion are roughly equivalent to the qualities produced for models of Anger. This results in an overall value of 0.612, which is usable for 37% of the subject population. These are encouraging results, considering no information on the actual state of the participant was given in this approach.

Table 39 – Fear model qualities with unsupervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.55	0.55	0.55	0.55	0.52	0.50	0.50	0.50	0.50	**0.525**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	0.54	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4111	0.56	0.53	0.52	0.52	0.52	0.52	0.52	0.51	0.50	**0.524**
4115	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4137	1.00	0.54	0.54	0.54	0.54	0.54	0.54	0.54	0.54	**0.593**
4101	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4117	0.53	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.510**
4102	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.52	0.51	**0.892**
4105	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.56	0.50	**0.896**
4104	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	**0.946**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4106	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	**0.891**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.507**
Average	**0.853**	**0.824**	**0.823**	**0.823**	**0.821**	**0.821**	**0.820**	**0.745**	**0.715**	**0.805**
Total Usable (avg ROC >0.6):		12			Percent Usable:			63%		

Unsupervised models of Fear created by the ART algorithm are comparable in quality to their supervised versions. When comparing the unsupervised models (0.805) with their supervised equivalents (0.841), one can draw the conclusion that the introduction of labeling information does not aid significantly. Labeling information boosted overall quality, and created 3 additional usable models for individual participants, but involved an unrealistic amount of information. It is hoped that semi-supervised information can bridge the gap between these created models.

Table 40 – Anger model qualities with unsupervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.54	**0.949**
4133	0.58	0.58	0.58	0.58	0.52	0.50	0.50	0.50	0.50	**0.540**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	1.00	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.560**
4111	0.56	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.509**
4115	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.559**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.506**
4137	1.00	0.56	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.570**
4101	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	0.50	**0.835**
4117	0.56	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.510**
4102	0.56	0.56	0.56	0.56	0.51	0.50	0.50	0.50	0.50	**0.531**
4105	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4104	1.00	0.54	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.565**
4107	1.00	1.00	1.00	0.63	0.63	0.63	0.63	0.63	0.63	**0.750**
4106	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
Average	0.781	0.674	0.668	0.648	0.642	0.641	0.614	0.614	0.589	**0.652**
Total Usable (avg ROC >0.6):			6			Percent Usable:		32%		

The performance of unsupervised Anger models created using clustering is barely acceptable, with total quality levels of 0.652. Clustering and ART modeled these states nearly identically, and outperform their offline equivalents with labeled data. While barely acceptable, it is worthwhile to note that this closely marks the real world performance, when labeling information is not present.

Table 41 – Boredom model qualities with unsupervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.506**
4131	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4127	0.54	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.507**
4121	0.54	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.508**
4111	1.00	1.00	1.00	0.75	0.51	0.50	0.50	0.50	0.50	**0.696**
4115	1.00	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	**0.673**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	1.00	0.53	0.51	0.51	0.51	0.50	0.50	0.50	0.50	**0.562**
4101	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4117	1.00	0.67	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.579**
4102	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.558**
4105	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4104	1.00	1.00	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.615**
4107	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4106	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
Average	**0.770**	**0.697**	**0.662**	**0.577**	**0.562**	**0.561**	**0.561**	**0.561**	**0.561**	**0.612**
Total Usable (avg ROC >0.6):			7			Percent Usable:			37%	

The overall unsupervised Boredom model qualities produced by clustering are comparable to the similar ones produced by ART, as they both reflect 0.612 in aggregate. Each of these produced 7 individually usable participant models without any labeling information.

Table 42 – Fear model qualities with unsupervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.55	0.55	0.55	0.55	0.52	0.50	0.50	0.50	0.50	**0.525**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	0.54	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4111	0.56	0.53	0.52	0.52	0.52	0.52	0.52	0.51	0.50	**0.524**
4115	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4137	1.00	0.54	0.54	0.54	0.54	0.54	0.54	0.54	0.54	**0.593**
4101	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4117	0.53	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.510**
4102	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.52	0.51	**0.892**
4105	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.56	0.50	**0.896**
4104	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	**0.946**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4106	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	**0.891**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.507**
Average	**0.853**	**0.824**	**0.823**	**0.823**	**0.821**	**0.821**	**0.820**	**0.745**	**0.715**	**0.805**
Total Usable (avg ROC >0.6):		12			Percent Usable:			63%		

Unsupervised models of Fear created using clustering are comparable to their supervised versions as they produced aggregate values of 0.805 and 0.810, respectively. This leads to the conclusion that labeling information was not particularly helpful in the establishment of categories of data for this affective state. Now that each of the three models of Anger, Boredom, and Fear have been briefly discussed with clustering and ART, they can be summarily discussed with the aid of the below table.

Table 43 – Summary of supervised ART (Table 30, Table 31, Table 32) and clustering (Table 33, Table 34, Table 35) when compared against unsupervised version of ART (Table 37, Table 38, and Table 39) and clustering (Table 40, Table 41, Table 42)

Model	Anger	Boredom	Fear
Supervised ART	0.776	0.796	0.841
Unsupervised ART	0.652	0.612	0.805
Change	-0.124	-0.184	-0.036
Supervised Clustering	0.681	0.644	0.810
Unsupervised Clustering	0.652	0.612	0.805
Change	-0.029	-0.032	-0.005

Firstly, the reader will note that there is not any improvement of an individual model over time within the unsupervised versions of these models. As an example, the supervised ART Fear model User 4121 (Table 32) improves over time, from a low of 0.62 to a high of 0.75. The same model for this user, when constructed without supervision (Table 39), starts at 0.54 and never recovers, ending with a 0.51 value. Labeling information allows for higher quality model construction when state changes are not obvious to the algorithm. The idea that being algorithmically informed of labels allows a model to better predict labeling information is intuitive, and is expressly confirmed in the resulting data.

Secondly, it is obvious from Table 36 that the unsupervised models are poorer in overall quality. The use of labeling information allows models to be of higher quality overall. These algorithms, however, are created for their use in real world settings, where labeling information is not available with fine resolution. There is no comparison against offline models for unsupervised models, as the offline models are not predicted to be useful, for the reasons of transferability discussed in Chapter 2. The testing of

unsupervised parameters allows the researcher to estimate how well constructed model quality will be within the field of use.

In general, with model qualities of 0.652, 0.612, 0.805 for the Anger, Boredom, and Fear models, respectively, making them barely usable. Only a model of Fear is able to be both good in quality and created in realtime for users, on average. Even the model of Fear is only reliable for two thirds of the population, while the other models are usable for approximately one third of the population. It is worth noting that the models of Anger and Boredom *approach* meaningful levels of classification using VW, clustering, and ART methods of creation. The tuning of parameters in a similar fashion to the cognitive models of Section 6.4.5 is performed in order to attempt to gain quality improvements through finer-grained cluster sizes.

The question that this subsection attempts to answer is "Can a quality *affective* model be constructed with *unsupervised* realtime algorithms?". The answer to the question is that a quality affective model *can* be made in realtime, but may not be valid for a significant portion of the population. There are several implications of this finding, which depend on the perspective field. The fields considered in this section are the field of Intelligent Tutoring Systems and the field of psychology, and are discussed next.

From a psychological perspective, the reason for this bifurcated behavior is simple: some users are more expressive than others. Unsupervised models were created without in-depth labeling information about user state. If a user is expressive about their state (e.g. physically recoiling from the computer, clear change in heart rate, etc.), then

in-depth labels are not required. The algorithm will model this state transition and does not require information about the new state for quality model construction. Users which present distinct states need little labeling information, leading to quality models despite lack of labeling information.

From an ITS perspective, information about participant state is not required at millisecond-by-millisecond resolution, as instructional interventions operate on a longer timescale. The ITS is interested in states when they have known labels, which is not possible under a completely unsupervised approach. Affective models only need to occasionally communicate information about student state to an instructional engine, as changes to instruction within an ITS occur infrequently. A model should communicate information only when the state is known, which makes use of semi-supervised approaches.

Having a model which is only occasionally reliable *is acceptable* to ITS systems in two occasions. The first occasion is that it does not communicate unreliable information, or only communicates state information when the state is known. The second occasion is if it informs the ITS of its reliability. An example of this is an affective model which communicates a message such as "This module has only 5% confidence that this user is Bored".

The reader should observe that it is not possible with realtime individualized completely unsupervised approaches to communicate information such as "Bored". Instead, the algorithm communicates "Cluster 5". While "Cluster 5" may be a quality

model of state, as shown in Table 36, it has little instructional meaning. It would be more desirable to communicate this state as "Bored".

Unsupervised models were created in order to represent the worst possible performance of labeled information. This sets the lower bound for comparison of the semi-supervised methods which closer approximate the real world problem, as discussed in Section 5.2. This lower bound can be compared against the two established fully supervised bounds presented by offline and online approaches.

Garnering information about the learner to give mostly-unsupervised algorithms information about the true state and an estimate of reliability is undertaken in section 6.4.12, with a discussion of semi-supervised learning methods. Additionally, it is possible that this information can be used to build higher quality models, as part of well-reasoned active learning selections of labeled datapoints.

6.4.12. Research Question 3b - Semi-Supervised and Active Learning for Affective Models

The question that the discussion within this subsection seeks to answer is "Do *semi-supervised* and active learning approaches improve *affective* model quality?". As discussed in Section 5.2, it is possible to ask the user directly, on occasion, for a point of labeled data. For the models with barely acceptable average quality, does the injection of the occasional label help? Figure 36 and Figure 37 graphically show the effect that this has on overall model quality.
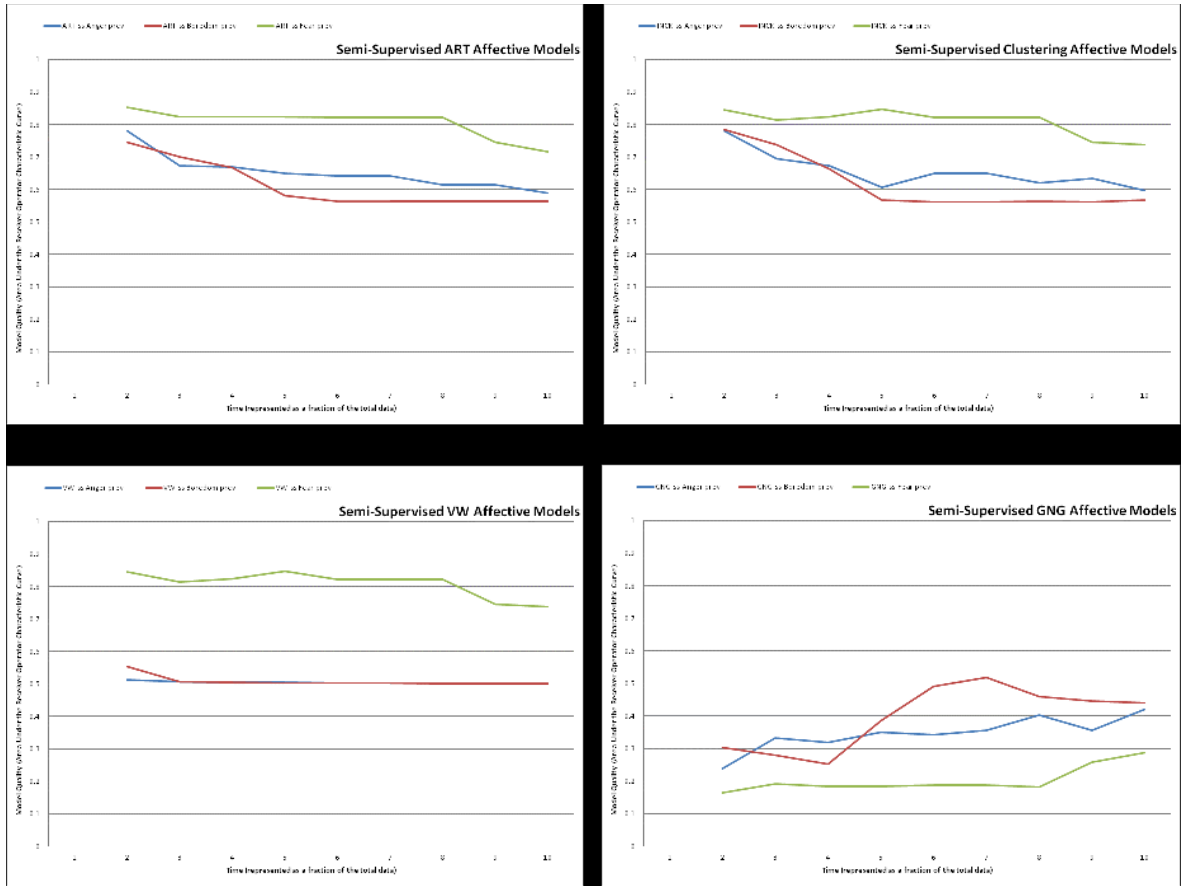
Figure 36 – Summary of realtime semi-supervised affective modeling ability across all algorithms using initial parameter settings
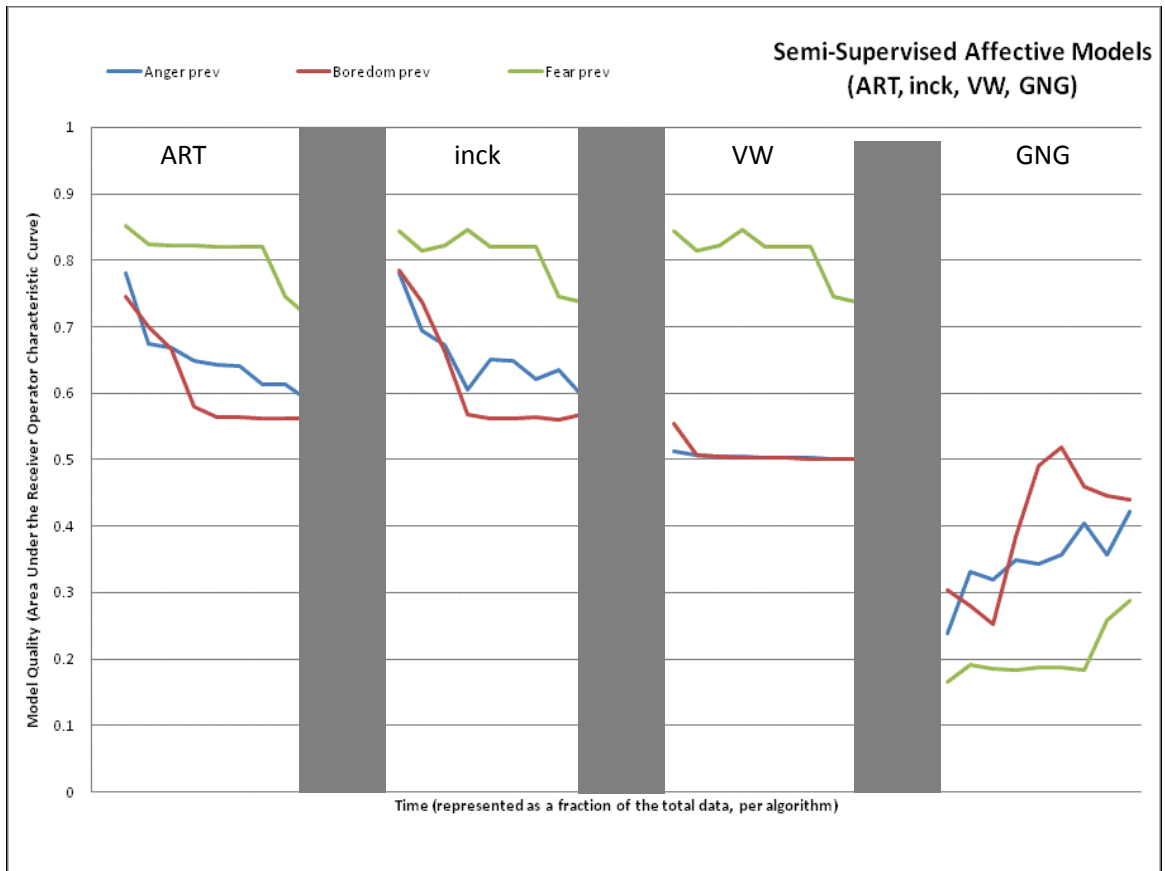
Figure 37 – Affective modeling quality, as measured over time by AUC ROC on the most recent 10% of data, with all algorithms in semi-supervised fashion.

The answer, when comparing Figure 36 and Figure 37 to Figure 34 and Figure 35, is unclear. Once again, the graphs of Figure 36 and Figure 37 should be examined in further depth to determine exactly the effect that semi-supervision had on overall quality. This is performed within the semi-supervised ART tables (Table 44, Table 45, Table 46) and semi-supervised clustering table (Table 47, Table 48, Table 49), for the two best performing methods. These results are summarized across all tables in Table 50, before discussion.

216

VW performed poorly on two of the three models. Additionally, VW still experiences the brittleness discussed earlier. Given these two items, the discussion of the following tables will focus on the two best performing algorithms (ART and clustering), as these are the most likely to be useful in the field.

Table 44 – Anger model qualities with semi-supervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.54	**0.949**
4133	0.58	0.58	0.58	0.58	0.52	0.50	0.50	0.50	0.50	**0.540**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	1.00	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.560**
4111	0.56	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.509**
4115	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.559**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.506**
4137	1.00	0.56	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.570**
4101	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	0.50	**0.835**
4117	0.56	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.510**
4102	0.56	0.56	0.56	0.56	0.51	0.50	0.50	0.50	0.50	**0.531**
4105	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4104	1.00	0.54	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.565**
4107	1.00	1.00	1.00	0.63	0.63	0.63	0.63	0.63	0.63	**0.750**
4106	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
Average	**0.781**	**0.674**	**0.668**	**0.648**	**0.642**	**0.641**	**0.614**	**0.614**	**0.589**	**0.652**
Total Usable (avg ROC >0.6):		6			Percent Usable:			32%		

Semi-supervised methods of creating models of Anger have no effect on the overall quality of models created, when compared to the unsupervised models. While they give context, as discussed above, they do not outperform the offline approaches.

Table 45 – Boredom model qualities with semi-supervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
4131	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4127	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4121	0.52	0.52	0.52	0.51	0.51	0.51	0.51	0.50	0.50	**0.510**
4111	1.00	1.00	1.00	0.75	0.51	0.50	0.50	0.50	0.50	**0.696**
4115	1.00	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	**0.673**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.511**
4101	0.52	0.52	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.508**
4117	1.00	0.67	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.579**
4102	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.558**
4105	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.506**
4104	1.00	1.00	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.615**
4107	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4106	0.53	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.508**
4112	0.51	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	**0.507**
4132	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.505**
Average	**0.745**	**0.701**	**0.666**	**0.580**	**0.564**	**0.563**	**0.563**	**0.563**	**0.563**	**0.612**
Total Usable (avg ROC >0.6):			7			Percent Usable:		37%		

Similar to semi-supervised ART models of Anger, the semi-supervised ART models of Boredom experienced no improvement in quality due to the injection of labeling information. The quality produced in this fashion is identical to the quality produced via unsupervised clustering models, and is barely acceptable overall.

Table 46 – Fear model qualities with semi-supervised ART algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.55	0.55	0.55	0.55	0.52	0.50	0.50	0.50	0.50	**0.525**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4121	0.54	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4111	0.56	0.53	0.52	0.52	0.52	0.52	0.52	0.51	0.50	**0.524**
4115	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4137	1.00	0.54	0.54	0.54	0.54	0.54	0.54	0.54	0.54	**0.593**
4101	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4117	0.53	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.50	**0.510**
4102	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.52	0.51	**0.892**
4105	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.56	0.50	**0.896**
4104	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	**0.946**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4106	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	0.50	**0.891**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.507**
Average	**0.853**	**0.824**	**0.823**	**0.823**	**0.821**	**0.821**	**0.820**	**0.745**	**0.715**	**0.805**
Total Usable (avg ROC >0.6):		12			Percent Usable:			63%		

As noted for the semi-supervised ART models of Boredom, the semi-supervised ART models of Fear obtained quality which matches the unsupervised clustering and ART models. While it adds context, the semi-supervision added to ART has not, in any case, produced more usable models or higher overall quality. This finding is discussed in greater depth after an examination of the clustering performance.

Table 47 – Anger model qualities with semi-supervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4131	0.51	1.00	0.50	0.50	0.50	0.50	0.50	0.63	0.50	**0.572**
4127	1.00	0.54	1.00	0.51	0.63	0.63	0.63	0.51	0.63	**0.672**
4121	1.00	0.50	0.51	0.50	0.51	0.51	0.51	0.50	0.51	**0.560**
4111	0.51	0.56	0.50	0.56	0.50	0.50	0.50	0.50	0.50	**0.516**
4115	0.56	0.51	0.56	0.50	0.51	0.50	0.50	0.50	0.50	**0.518**
4135	0.56	1.00	0.50	0.51	0.50	0.50	0.50	0.51	0.50	**0.566**
4136	1.00	0.51	1.00	0.50	1.00	1.00	0.51	0.50	0.50	**0.724**
4137	1.00	0.51	0.51	1.00	0.51	0.51	0.51	1.00	0.50	**0.674**
4101	0.53	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.561**
4117	1.00	1.00	1.00	0.50	1.00	1.00	1.00	0.50	1.00	**0.890**
4102	1.00	0.58	0.50	0.51	0.50	0.50	0.50	1.00	0.50	**0.623**
4105	1.00	1.00	0.50	1.00	0.50	0.50	0.50	1.00	0.50	**0.724**
4104	1.00	0.75	0.51	0.58	1.00	1.00	0.50	0.50	1.00	**0.761**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4106	0.58	0.79	1.00	0.76	0.52	0.50	1.00	0.75	0.50	**0.712**
4112	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	0.50	**0.889**
4132	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.54	0.52	**0.895**
Average	**0.804**	**0.751**	**0.690**	**0.655**	**0.668**	**0.667**	**0.640**	**0.629**	**0.588**	**0.677**
Total Usable (avg ROC >0.6):			11			Percent Usable:		58%		

Semi-supervised methods of clustering have increased overall model quality significantly, when compared to the unsupervised approaches. In order of discussion, from supervised, to unsupervised, to semi-supervised, overall model quality for Anger is 0.681, 0.652, and 0.677, which indicates that semi-supervision has increased overall quality.

The more interesting finding is that semi-supervision has increased the number of individually usable models. Unsupervised methods produce 6 usable models, while supervised methods result in 9 usable models. Semi-supervised methods have targeted

220

the most relevant data points, resulting in 11 individually usable models, which is greater than in either other case. The invention of the clustering method of semi-supervision is a significant contribution, as it boosts overall model quality while significantly increasing the number of usable models. This finding is discussed in greater depth in the summary section.

Table 48 – Boredom model qualities with semi-supervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4131	1.00	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.612**
4127	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4121	0.51	0.50	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.505**
4111	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.557**
4115	1.00	0.53	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.560**
4135	0.51	1.00	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.561**
4136	1.00	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.613**
4137	1.00	1.00	0.51	0.51	0.51	0.50	0.60	0.50	0.60	**0.637**
4101	1.00	1.00	1.00	0.60	0.60	0.60	1.00	0.60	1.00	**0.822**
4117	1.00	0.51	1.00	1.00	1.00	1.00	0.51	1.00	0.51	**0.836**
4102	1.00	0.50	1.00	0.51	0.51	0.51	0.50	0.51	0.50	**0.617**
4105	0.54	0.51	1.00	0.50	0.51	0.50	0.50	0.50	0.50	**0.563**
4104	0.54	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.562**
4107	0.51	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4106	0.51	0.51	0.50	0.51	0.50	0.50	0.50	0.50	1.00	**0.559**
4112	1.00	0.76	0.51	1.00	0.50	0.50	1.00	0.50	0.50	**0.697**
4132	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
Average	**0.796**	**0.728**	**0.662**	**0.590**	**0.562**	**0.561**	**0.587**	**0.561**	**0.587**	**0.626**
Total Usable (avg ROC >0.6):		9			Percent Usable:		47%			

The semi-supervised models of Boredom created by the online clustering algorithm have similar findings to those discussed in the Anger section. The added semi-supervision produced model quality less than full supervision, but greater than no supervision. The

more interesting finding is that semi-supervised methods have produced as many usable individual models as fully supervised methods. This finding is discussed in greater depth in the summary section.

Table 49 – Fear model qualities with semi-supervised clustering algorithm using initial parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.506**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4131	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	1.00	**0.946**
4127	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.51	**0.946**
4121	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	**0.945**
4111	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.56	0.51	**0.896**
4115	0.53	0.51	1.00	1.00	1.00	1.00	1.00	0.52	0.50	**0.784**
4135	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	1.00	**0.672**
4136	1.00	0.54	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**0.949**
4137	1.00	1.00	0.54	1.00	0.54	0.54	0.54	0.54	1.00	**0.745**
4101	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4117	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	**0.945**
4102	0.56	0.53	1.00	0.52	1.00	1.00	1.00	1.00	0.51	**0.791**
4105	0.54	0.52	0.52	0.51	0.52	0.52	0.52	0.51	1.00	**0.575**
4104	1.00	1.00	0.51	1.00	0.51	0.51	0.51	0.51	1.00	**0.727**
4107	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	**0.945**
4106	0.55	0.55	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**0.900**
4112	1.00	1.00	0.55	0.53	0.52	0.50	0.50	0.50	0.50	**0.623**
4132	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
Average	**0.853**	**0.824**	**0.823**	**0.846**	**0.821**	**0.821**	**0.820**	**0.745**	**0.739**	**0.810**
Total Usable (avg ROC >0.6):		16			Percent Usable:			84%		

As observed with the semi-supervised clustering models of Anger and Boredom, the occasional labeled data point has significantly increased the number of usable models. Both full supervision and no supervision resulted in 12 individually usable models of Fear, while semi-supervision resulted in 16. Average model quality in the semi-

supervised case is identical to the fully supervised case, despite the significant withholding of labeled information.

Table 50 – Summary of all ART and clustering tables thus far

Model	Anger	Boredom	Fear
Supervised ART	0.776	0.796	0.841
Unsupervised ART	0.652	0.612	0.805
Semi-Supervised ART	0.652	0.612	0.805
Supervised Clustering	0.681	0.644	0.810
Unsupervised Clustering	0.652	0.612	0.805
Semi-Supervised Clustering	0.677	0.626	0.810

Firstly, the reader should note the effect that semi-supervised methods have had on the ART and clustering algorithms. They have had no effect on ART performance, while having significant effect on clustering quality. The reasons for this are discussed next.

The occasional labeled point did not help ART performance. The reason for this how labeling information is used in the establishment of clusters in Section 5.4.3. Labeling information is used to separate one cluster from another. When only five labels are given to the data, and these are only given to the largest class of data, there is not enough differentiation to have an effect on the model. The labeling information given to ART is merely associating a label with an existing cluster, rather than aiding in the establishment of a new cluster.

Active learning is performed differently in each algorithmic case. In brief, ART requests the label of the largest cluster, VW selects a point which minimizes the hypothesis error, GNG selects the centroid of an established network, and clustering

requests the label of the last datapoint seen on the largest cluster. The approach taken with clustering selects a point which, according to Table 47, Table 48, Table 49, was misclassified. This selection results in the improvement of the model.

Using Anger as an example, supervised clustering produces nine usable models while unsupervised clustering produces only six. Semi-supervised approaches lead to the production of *eleven* usable models. This gain in performance furthers a deeper look into how many models were usable across each method and labeling scheme, and is shown in Table 51.

Table 51 – Summary of all ART and clustering usable models thus far. Each number represents how many usable affective models were created, of 19 total.

Model	Anger	Boredom	Fear
Supervised ART	17	18	15
Unsupervised ART	6	7	12
Semi-Supervised ART	6	7	12
Supervised Clustering	9	9	12
Unsupervised Clustering	6	7	12
Semi-Supervised Clustering	11	9	16

Semi-supervised clustering redeems a number of the models of affect. It outperforms unsupervised and semi-supervised ART, as well as all of the other methods of clustering. This performance is done with only *five* labeled datapoints per user, and their intelligent selection, while remaining realtime appropriate. The selection, in the instance of clustering, is determined by the last point which was categorized to be belonging to the largest class of unlabeled data. The selection of an appropriate datapoint to label can remove the confusion caused by numerous inconsistent labels, which is why it

outperforms supervised clustering. This selection is also used to give meaning to unsupervised clusters, which boosts overall model performance. The story of the success of semi-supervised clustering is best told in the story of User 4117, shown below in Table 52.

Table 52 – Differing supervision of clustering for User 4117 Anger models

clustering labeling	User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
supervised	4117	0.563	0.527	0.555	0.503	0.503	0.490	0.545	0.541	0.510	**0.526**
unsupervised	4117	0.563	0.510	0.504	0.503	0.503	0.503	0.502	0.501	0.501	**0.510**
semi-supervised	4117	1.000	1.000	1.000	0.503	1.000	1.000	1.000	0.503	1.000	**0.890**

The algorithmic selection of five labeling points belonging to the largest class of data for User 4117 boosts performance from unacceptable levels to near-perfect levels. This occurs though labeling conflict, where a cluster has multiple conflicting labels. The approach of using a point which is representative of the cluster to determine the total cluster label redeems data which may have previously been misclassified.

However, as mentioned in the preceding section, each algorithm is not able to draw conclusions from the data classifications without the injection of the occasional point. Each algorithm must identify a group of datapoints as "Cluster #1" or "Category 4". These unsupervised classification mechanisms are not useful to an ITS, despite that they may be accurately modeling the individual. Giving context, via a labeled datapoint request, to a previously established cluster is an important part of ITS research. This allows the algorithm to associate "Cluster 1" with "Boredom", which has instructional implications. The finding from this section is that infrequently requesting labeled

datapoints both aids in overall model quality *and* allows for the establishment of instructional meaning. The answer to the question of "Do *semi-supervised* and active learning approaches improve *affective* model quality?" is "Yes, it helps to both establish cluster meaning and to improve overall model quality."

6.4.13. *Revised Parameter Settings for Affective Models*

While the cognitive models presented in Section 6.4.5 did not benefit from the creation of smaller cluster sizes, it is possible that the affective models could benefit from the same type of change. The parameters in this section were modified in the same fashion, with the same reasoning, as discussed in Section 6.4.5 and Table 28. The research question addressed by Figure 38, Figure 39, and Figure 40 is "Does a change of parameter settings to reflect finer-grained clusters create higher quality models?"
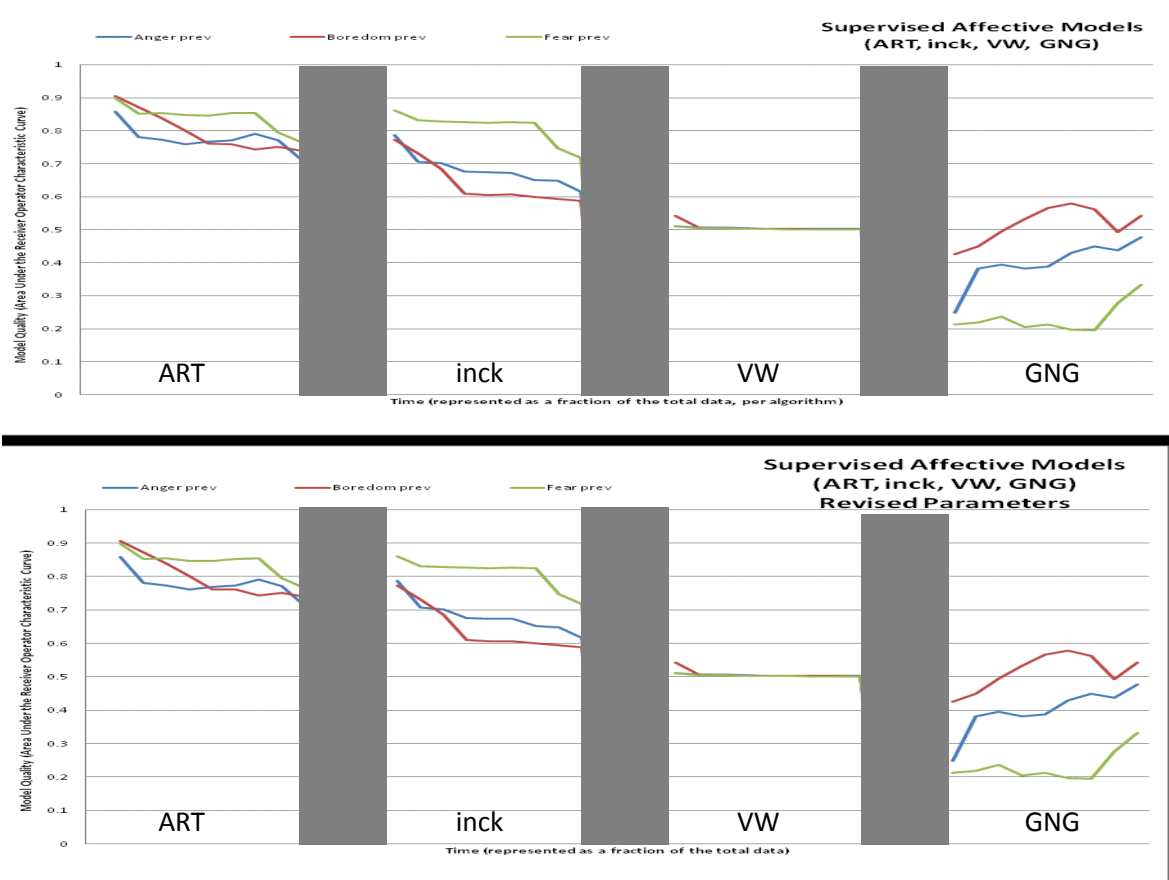
Figure 38 – Performance of all supervised algorithms and both parameter sets for all affective models using the previous measure. From left to right, the algorithms shown are ART, clustering, VW, and GNG.

As can be seen via visual inspection of trends, there was no significant change observed from a change of parameter settings in the quality of constructed models at any time. Being given roughly twice the number of categories of classification does not significantly aid overall in the modeling of this specific affective dataset. This finding is a repeat of the finding observed from the same change in cognitive models.
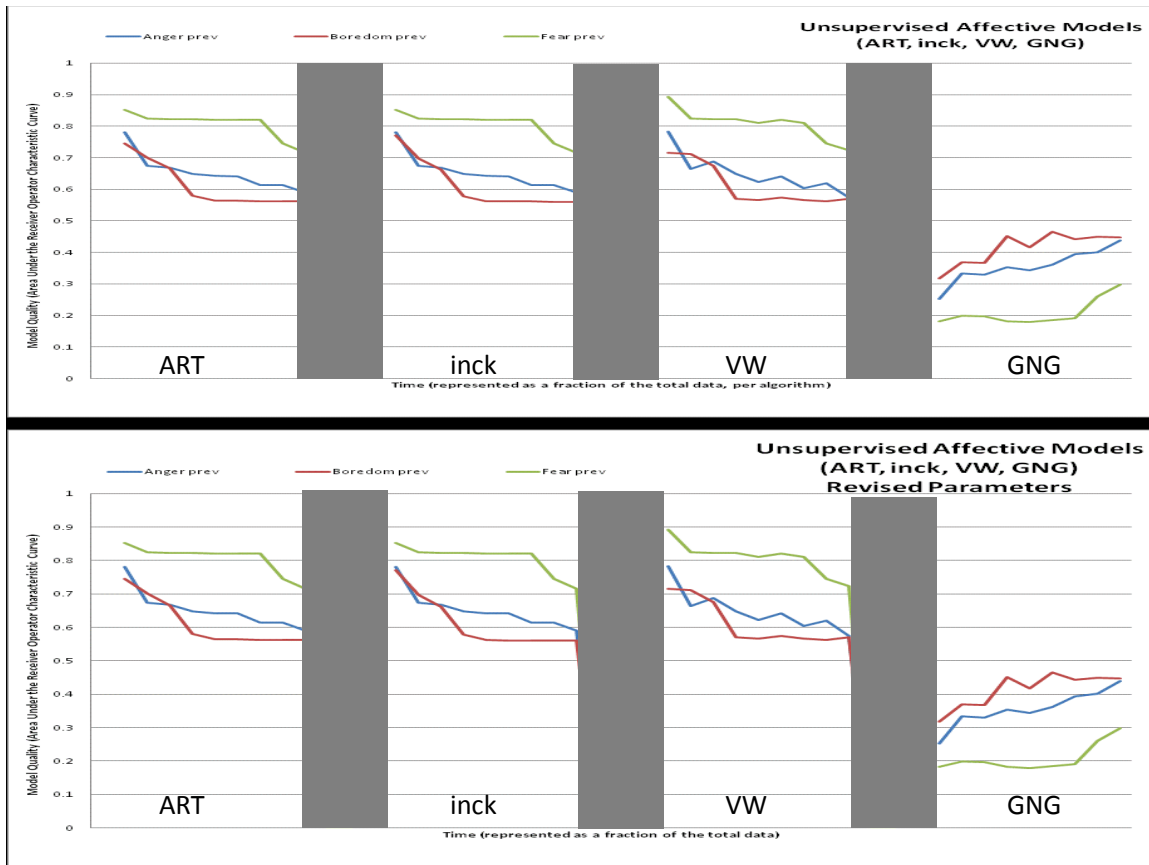
Figure 39 – Performance of all unsupervised algorithms and both parameter sets for all affective models using the previous measure. From left to right, the algorithms shown are ART, clustering, VW, and GNG.

The change in parameter set for unsupervised models has the same overall effect as the one for supervised models. A brief visual inspection of Figure 39 reveals no discernible difference between the parameters. This is validated in the experimental tables, which are not shown, as no conclusion can be drawn from them.
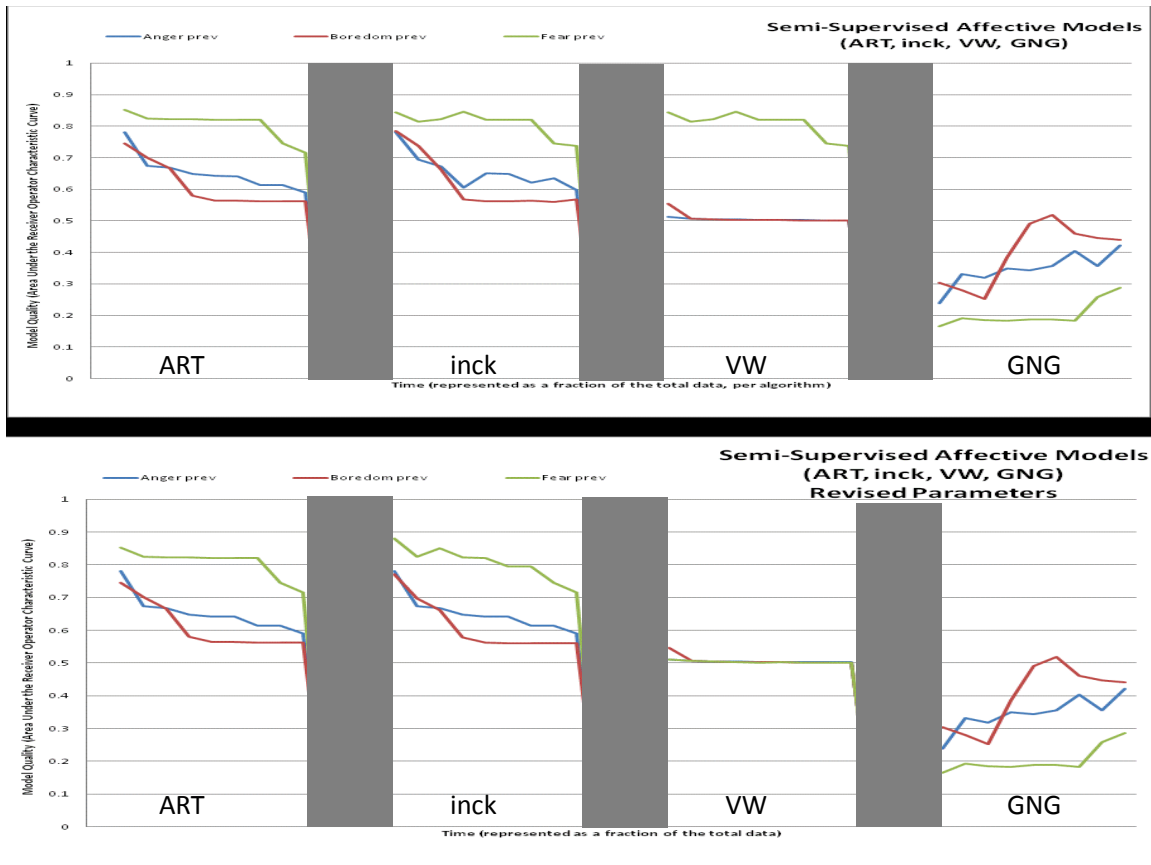
Figure 40 – Performance of all semi-supervised algorithms and both parameter sets for all affective models using the previous measure. From left to right, the algorithms shown are ART, clustering, VW, and GNG.

There are three items worth mentioning about the above differences in Figure 38, Figure 39, and Figure 40, which are little overall improvements, significantly reduced performance for VW, and overall clustering improvements. These items for discussion are shown most clearly in Figure 40, which shows semi-supervised performance.

Figure 40 shows VW experiencing significantly reduced performance in the case of the model of Fear through the use of adaptive learning rates. The use of adaptive

learning rates is designed to allow mostly-unsupervised models to more closely approximate the supervised equivalents (Agarwal et al. 2011). In this rare instance, the unsupervised models *outperform* the supervised models, leaving the supervised model approximation to have overall *net negative* effect. This finding is consistent with the observations of Agarwal et al., where adaptive one-pass learning more closely approximated supervised learning, but has resulted in a performance decrease in this instance (Agarwal et al. 2011).

As part of further testing, a series of additional parameter settings were attempted for ART. ART is the best-performing algorithm across both affect and cognition, and various values of the vigilance parameter were attempted. These were not shown to aid in significantly from initially chose parameter settings, but are included for completeness in APPENDIX D.

Small improvements were observed in the semi-supervised clustering methods, which take longer to decay through the use of smaller cluster sizes. Because of this observation, the tests conducted in Section 6.4.14, discussed next, use revised parameter settings. The other cases show no improvement in overall model quality, which is consistent with the results from the cognitive models. In answer to the research question, the change of parameter settings has a small positive overall effect when labeling information is limited with no harmful effect in other cases.

6.4.14. Reduced Feature Set Affective Models

As discussed in Section 6.4.6, the offline linear regression models created by other researchers did not make use of all features of the data. For completeness, the use of the reduced feature set is tested on the affective models, in order to answer the developed research question. This question is "When eliminating features determined to be of little use during offline analysis, is overall model quality improved for either cognitive or affective models?"

The reader should note that, of the three affective labels (Boredom, Anger, Fear), only Boredom is used in this experiment. An initial model of Anger was not able to be created using offline algorithms of the other researchers, and therefore does not have a reduced input feature set. The model of Fear created by the offline researchers used all of the available features, so is identical to the earlier created models. The exact features used are shown in Table 53, but are briefly the Alpha, Gamma, and Heart features. The below figures show the trend of the reduced feature set models when compared to the initial models.

Table 53 – Summary and example of features used in each created model. Partial reprint of Table 18. No model of Anger above 0.6 ROC value was created with offline approaches.

	Appendix	Boredom	Fear
Alpha1	A-1		X
Alpha2	A-1	X	X
Gamma1	A-1	X	X
Gamma2	A-1		X
Delta	A-1		X
Beta1	A-1		X
Beta2	A-1		X
Theta	A-1		X
Attention	A-1		X
Meditation	A-1		X
Left Eye Pupil Diameter	A-5		X
Heart	A-2		X
Chair 1-4	A-4		
Chair 5-8	A-4		X
Motion	A-3		X
Alpha1Diff	A-6		X
Alpha2Diff	A-6		X
Gamma1Diff	A-6	X	X
Gamma2Diff	A-6		X
DeltaDiff	A-6		X
Beta1Diff	A-6	X	X
Beta2Diff	A-6	X	X
ThetaDiff	A-6		X
AttentionDiff	A-6		X
MeditationDiff	A-6		X
HeartDiff	A-6	X	X
MotionDiff	A-6		X

Figure 41 – Performance of all supervised algorithms for Boredom models using the previous measure. From left to right, the algorithms shown are ART, clustering, VW, and GNG.

Reducing the number of features available for the supervised ART and clustering is worthy of discussion. A significant decrease in overall quality is observed for ART, which had an initial plateau above 0.7, and was reduced to a plateau value of less than 0.6. Clustering, contrarily, experienced no overall degradation due to the lack of features. The implications to experimenters are less clear in the supervised case, and the results from un- and semi-supervised methods are presented next.
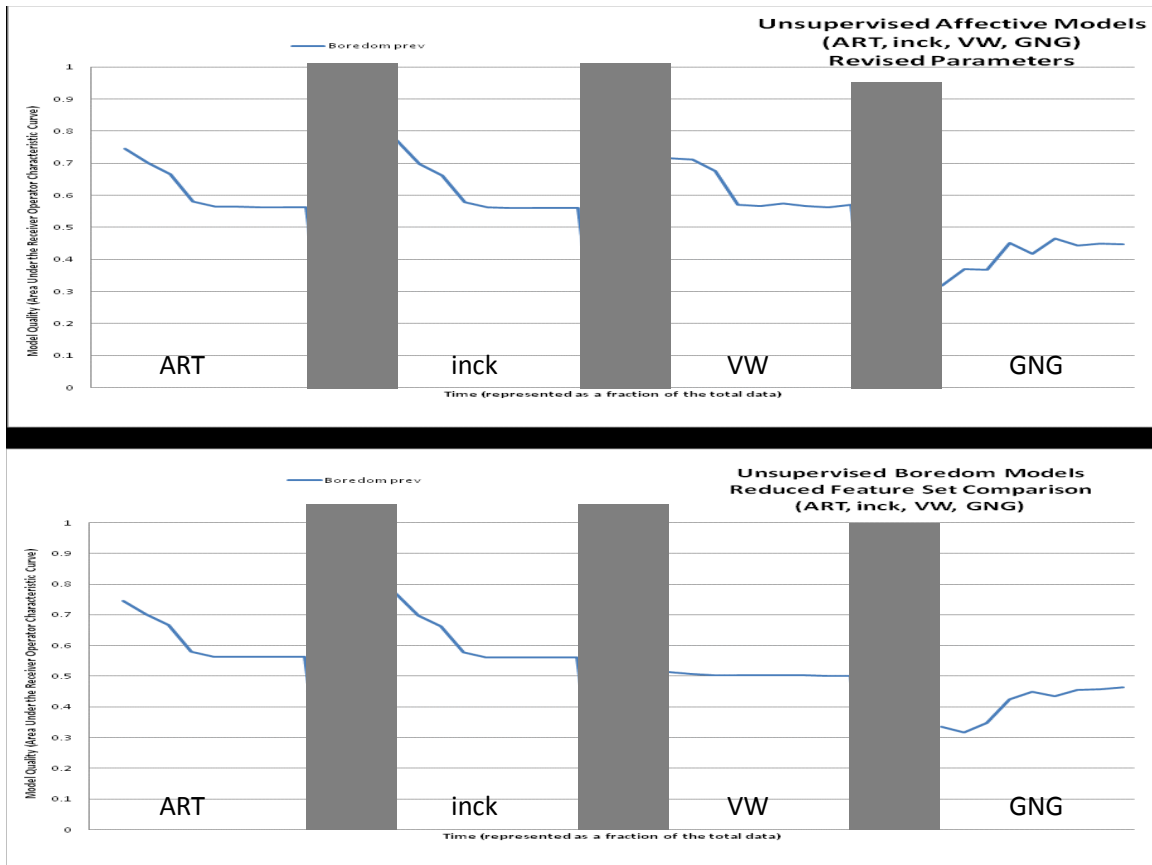
Figure 42 – Performance of all unsupervised algorithms for Boredom models using the previous measure. From left to right, the algorithms shown are ART, clustering, VW, and GNG.

Unlike in the supervised case, the unsupervised reduced feature set has no immediately observable change in algorithmic performance. If this visual inspection observation were true, it would imply that an experimenter interested in the Boredom state would not have needed to collect extra sensor information from sensor chair, motion sensor, or heart rate monitor. These figures indicate a further discussion of the differences between the reduced features set and full feature set is required.
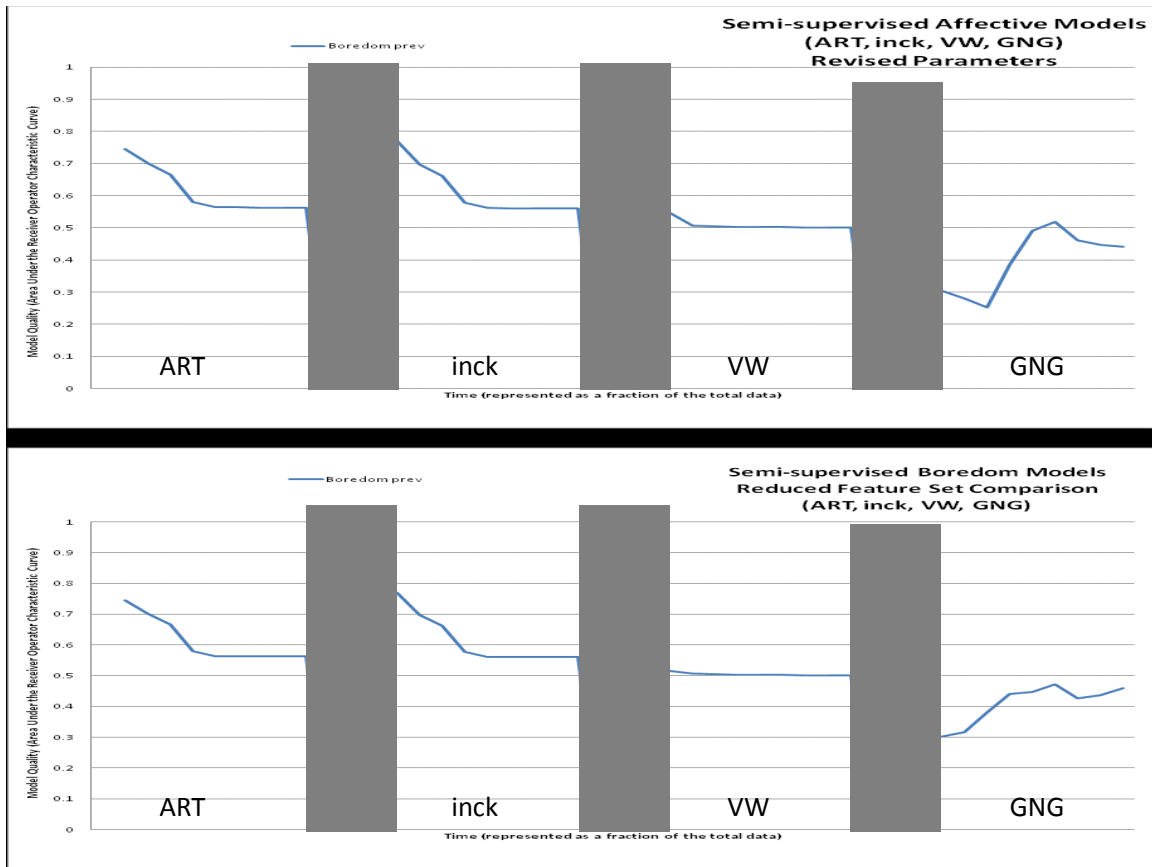
Figure 43 – Performance of all semi-supervised algorithms for Boredom models using the previous measure. From left to right, the algorithms shown are ART, clustering, VW, and GNG.

The similarities among Figure 41, Figure 42, and Figure 43 provides a justification for further study of how many of these models are usable when using a much smaller fraction of the overall data and sensor set. This is performed with the top two performing algorithms (ART and clustering) in the manner of the previous section, and presented in Table 54, Table 55, and Table 56 for ART and Table 57, Table 58, and Table 59 for clustering. These results are summarized across all tables in Table 60 prior to further discussion.

Table 54 – Boredom model qualities with supervised ART algorithm using reduced feature set and revised parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.96	0.92	0.92	0.77	0.70	0.71	0.68	0.72	0.58	**0.773**
4131	0.97	0.66	0.97	0.97	0.97	0.97	0.97	0.97	0.97	**0.939**
4127	0.63	0.67	0.60	0.60	0.53	0.51	0.62	0.51	0.65	**0.591**
4121	0.80	0.95	0.82	0.81	0.83	0.83	0.81	0.84	0.77	**0.829**
4111	1.00	1.00	1.00	0.75	0.73	0.79	0.83	0.74	0.79	**0.846**
4115	1.00	1.00	1.00	0.95	0.63	0.91	0.52	0.79	0.58	**0.821**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.91	0.78	0.74	0.73	0.76	0.78	0.80	0.81	0.82	**0.792**
4101	0.66	0.66	0.66	0.64	0.65	0.74	0.72	0.64	0.63	**0.665**
4117	1.00	0.67	0.52	0.67	0.72	0.51	0.56	0.58	0.59	**0.648**
4102	0.85	0.79	0.78	0.85	0.80	0.67	0.71	0.76	0.81	**0.780**
4105	0.80	0.84	0.84	0.69	0.62	0.66	0.63	0.62	0.66	**0.708**
4104	1.00	1.00	0.75	0.87	0.80	0.74	0.66	0.73	0.75	**0.810**
4107	1.00	1.00	1.00	0.75	0.79	0.74	0.79	0.79	0.79	**0.848**
4106	0.65	0.75	0.67	0.75	0.70	0.64	0.70	0.70	0.72	**0.699**
4112	0.98	0.88	0.88	0.88	0.78	0.78	0.65	0.60	0.58	**0.779**
4132	1.00	1.00	0.79	0.89	0.84	0.85	0.87	0.86	0.75	**0.873**
Average	**0.906**	**0.872**	**0.839**	**0.800**	**0.760**	**0.760**	**0.743**	**0.750**	**0.739**	**0.796**
Total Usable (avg ROC >0.6):				18		Percent Usable:		95%		

These Boredom model qualities can be compared with the initial reporting. The initial model models created 18 individually usable models and an average model quality of 0.796. Overall, there is no change resultant from the removal of three of the sensors.

Table 55 – Boredom model qualities with unsupervised ART algorithm using reduced feature set and revised parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
4131	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4127	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4121	0.52	0.52	0.52	0.51	0.51	0.51	0.51	0.50	0.50	**0.510**
4111	1.00	1.00	1.00	0.75	0.51	0.50	0.50	0.50	0.50	**0.696**
4115	1.00	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	**0.673**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.511**
4101	0.52	0.52	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.508**
4117	1.00	0.67	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.579**
4102	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.558**
4105	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.506**
4104	1.00	1.00	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.615**
4107	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4106	0.53	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.508**
4112	0.51	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	**0.507**
4132	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.505**
Average	**0.745**	**0.701**	**0.666**	**0.580**	**0.564**	**0.563**	**0.563**	**0.563**	**0.563**	**0.612**
Total Usable (avg ROC >0.6):		7			Percent Usable:			37%		

As was observed in the supervised ART case, the removal of features from the datastream had little effect on the number of acceptable models or overall model quality. The full feature set also produced 7 individually usable models, with a final average AUC value of 0.612. Given that the ART semi-supervised implementation has followed the unsupervised implementation in all cases presented so far, it is expected that these results will be similar in the semi-supervised case.

Table 56 – Boredom model qualities with semi-supervised ART algorithm using reduced feature set and revised parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.504**
4131	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4127	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4121	0.52	0.52	0.52	0.51	0.51	0.51	0.51	0.50	0.50	**0.510**
4111	1.00	1.00	1.00	0.75	0.51	0.50	0.50	0.50	0.50	**0.696**
4115	1.00	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	**0.673**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.52	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.511**
4101	0.52	0.52	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.508**
4117	1.00	0.67	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.579**
4102	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.558**
4105	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.506**
4104	1.00	1.00	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.615**
4107	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4106	0.53	0.53	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.508**
4112	0.51	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	**0.507**
4132	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	0.50	**0.505**
Average	**0.745**	**0.701**	**0.666**	**0.580**	**0.564**	**0.563**	**0.563**	**0.563**	**0.563**	**0.612**
Total Usable (avg ROC >0.6):		7			Percent Usable:			37%		

The prediction made after the previous table holds true; semi-supervised ART methods on a reduced feature set have produced the same number of usable models and the same value of overall model quality which was observed with the earlier full feature set. This implication is discussed further in the summary of this section.

Table 57 – Boredom model qualities with supervised clustering algorithm using reduced feature set and revised parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.51	0.51	0.51	0.50	0.54	0.53	0.50	0.50	**0.513**
4131	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	**0.513**
4127	0.58	0.58	0.57	0.57	0.54	0.54	0.51	0.51	0.51	**0.546**
4121	0.66	0.56	0.56	0.54	0.54	0.63	0.71	0.68	0.69	**0.620**
4111	1.00	1.00	1.00	0.74	0.58	0.60	0.62	0.61	0.53	**0.741**
4115	1.00	1.00	0.99	0.55	0.53	0.53	0.53	0.53	0.53	**0.686**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	0.97	0.66	0.51	0.53	0.51	0.51	0.51	0.51	0.51	**0.579**
4101	0.52	0.52	0.52	0.52	0.64	0.57	0.57	0.57	0.56	**0.553**
4117	1.00	0.67	0.52	0.55	0.58	0.55	0.51	0.50	0.50	**0.597**
4102	0.76	0.67	0.60	0.60	0.63	0.60	0.57	0.57	0.57	**0.619**
4105	0.53	0.51	0.51	0.54	0.62	0.62	0.56	0.56	0.52	**0.552**
4104	1.00	1.00	0.56	0.55	0.52	0.50	0.50	0.50	0.50	**0.627**
4107	1.00	1.00	1.00	0.61	0.58	0.59	0.59	0.59	0.59	**0.728**
4106	0.55	0.55	0.52	0.52	0.53	0.51	0.51	0.50	0.50	**0.521**
4112	0.61	0.63	0.60	0.59	0.58	0.61	0.58	0.55	0.55	**0.589**
4132	0.51	0.54	0.51	0.54	0.51	0.51	0.50	0.50	0.50	**0.514**
Average	**0.773**	**0.732**	**0.685**	**0.610**	**0.605**	**0.607**	**0.600**	**0.594**	**0.588**	**0.644**
Total Usable (avg ROC >0.6):		9			Percent Usable:			47%		

The supervised Boredom models created via clustering with the reduced feature set do not differ in overall quality or number of acceptable models. They produce and overall AUC measure of 0.644, and 9 usable models. This finding is similar to the on observed previously from ART and via visual inspection of the figures earlier in this section.

Table 58 – Boredom model qualities with unsupervised clustering algorithm using reduced feature set and revised parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4133	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.50	0.50	**0.506**
4131	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4127	0.54	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.507**
4121	0.54	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.508**
4111	1.00	1.00	1.00	0.75	0.51	0.50	0.50	0.50	0.50	**0.696**
4115	1.00	1.00	1.00	0.51	0.51	0.51	0.51	0.51	0.51	**0.673**
4135	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
4136	1.00	1.00	1.00	0.60	0.60	0.60	0.60	0.60	0.60	**0.733**
4137	1.00	0.53	0.51	0.51	0.51	0.50	0.50	0.50	0.50	**0.562**
4101	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4117	1.00	0.67	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.579**
4102	1.00	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.558**
4105	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4104	1.00	1.00	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.615**
4107	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4106	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4112	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4132	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
Average	**0.770**	**0.697**	**0.662**	**0.577**	**0.562**	**0.561**	**0.561**	**0.561**	**0.561**	**0.612**
Total Usable (avg ROC >0.6):		7		Percent Usable:			37%			

The above table further indicates that the removal of features identified by the offline experimenters to contain little value had no overall effect on model quality. The unsupervised Boredom models produced via clustering resulted in a 0.612 overall quality with 7 usable models in both cases.

Table 59 – Boredom model qualities with semi-supervised clustering algorithm using reduced feature set and revised parameters

User	20%	30%	40%	50%	60%	70%	80%	90%	100%	Avg
4134	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4133	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.502**
4131	1.00	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.612**
4127	1.00	1.00	1.00	0.56	0.52	0.52	0.52	0.52	0.52	**0.687**
4121	0.51	0.50	0.52	0.50	0.50	0.50	0.50	0.50	0.50	**0.505**
4111	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.557**
4115	1.00	0.53	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.560**
4135	0.51	1.00	0.52	0.51	0.50	0.50	0.50	0.50	0.50	**0.561**
4136	1.00	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.613**
4137	1.00	1.00	0.51	0.51	0.51	0.50	0.60	0.50	0.60	**0.637**
4101	1.00	1.00	1.00	0.60	0.60	0.60	1.00	0.60	1.00	**0.822**
4117	1.00	0.51	1.00	1.00	1.00	1.00	0.51	1.00	0.51	**0.836**
4102	1.00	0.50	1.00	0.51	0.51	0.51	0.50	0.51	0.50	**0.617**
4105	0.54	0.51	1.00	0.50	0.51	0.50	0.50	0.50	0.50	**0.563**
4104	0.54	1.00	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.562**
4107	0.51	0.51	0.50	0.50	0.50	0.50	0.50	0.50	0.50	**0.503**
4106	0.51	0.51	0.50	0.51	0.50	0.50	0.50	0.50	1.00	**0.559**
4112	1.00	0.76	0.51	1.00	0.50	0.50	1.00	0.50	0.72	**0.722**
4132	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**1.000**
Average	**0.796**	**0.728**	**0.662**	**0.590**	**0.562**	**0.561**	**0.587**	**0.561**	**0.598**	**0.627**
Total Usable (avg ROC >0.6):				9		Percent Usable:			47%	

The above table mirrors the findings of the previous five; removal of features extraneous to offline analysis has no effect on online model quality. The Boredom models produced via semi-supervised clustering on the reduced feature set result in 9 usable models and overall quality of 0.627. This is slightly better than the 9 usable models and 0.626 quality observed in the full feature set.

**Table 60 – Summary of quality metrics and usable models for ART and clustering Boredom models
with reduced feature set**

Model	AUC		Individually Usable Models	
	Boredom Original	Boredom Reduced	Boredom Original	Boredom Reduced
Supervised ART	0.796	0.796	18	18
Unsupervised ART	0.612	0.612	7	7
Semi-Supervised ART	0.612	0.612	7	7
Supervised Clustering	0.644	0.644	9	9
Unsupervised Clustering	0.612	0.612	7	7
Semi-Supervised Clustering	0.626	0.627	9	9

These results are encouraging, as they indicate that not all sensors were required to construct realtime models of Boredom. The use of the reduced feature set found in the original offline models did not hurt overall model quality, as shown in Table 60. This has the implication that only two sensors (EEG and Heart) were required in order to create a model of Boredom. An ITS looking for this state could obtain this type information with lower cost when compared with information about Anger or Fear. Additionally, this finding supports the recommendation that offline models can be created in order to *inform* the decisions of online model data collection. This finding indicates that future experiments should attempt offline modeling *for feature reduction* prior to online modeling *for use*, and that offline modeling approaches taken by other researchers in the fashion of Chapter 2 are not wasted effort.

6.4.15. *Affective Modeling Summary*

The initial three research questions, and the two subsequently developed questions, asked as part of this work are below:

1b.   Can a quality *affective* model be constructed with *fully supervised* realtime algorithms?

2b. Can a quality *affective* model be constructed with *unsupervised* realtime algorithms?

3b. Do *semi-supervised* and active learning approaches improve *affective* model quality, when compared to the unsupervised approaches?

4.   Does a change of parameter settings to reflect finer-grained clusters create higher quality models?

5. Does reducing the set of features to only the features used on affective model outputs create higher quality models?

In brief, the answers are that quality affective models can be constructed using supervised, unsupervised, and semi-supervised approaches, where very infrequent semi-supervision information can increase the number of usable models beyond the other approaches, while fine-grained clusters using fewer overall features produce results of similar quality.   Each of these answers warrants further discussion, through use of Summary Table 61.

Table 61 – Summary of all ART and clustering tables

Model	Anger	Boredom	Boredom (Reduced)	Fear	Usable?
Offline Linear Regression	NA (<0.6)	0.79	NA	0.83	Some
Supervised ART	0.776	0.796	0.796	0.841	Yes
Unsupervised ART	0.652	0.612	0.612	0.805	Yes
Semi-Supervised ART	0.652	0.612	0.612	0.805	Yes
Supervised Clustering	0.681	0.644	0.644	0.810	Yes
Unsupervised Clustering	0.652	0.612	0.612	0.805	Yes
Semi-Supervised Clustering	0.677	0.626	0.627	0.810	Yes

6.4.15.1. SUPERVISED AND UNSUPERVISED MODELS

The results from the creation of the affective models are encouraging. The previously

created affective models achieved quality of <0.6, 0.83, and 0.79, while supervised ART

is able to *outperform*, on *all benchmarks*, the offline approach *using a infinitesimal*

fraction of the total data. This succinctly answers the question of whether online models

can be created and indicates that the future research of others should be conducted in this

fashion.

The research conducted as part of this dissertation has not lost track of the goal:

the creation of student models for use in an ITS setting. With this goal in mind, a more

valuable metric of success is how well the algorithms for creating models perform when

given little labeling information, as is the case in an ITS. When looking at the research

by this metric, the ART and clustering models are equivalent, while the offline models

are expected to have poor quality for the reasons discussed within Chapter 2. The

research conducted in this dissertation indicates that the algorithmic creation of such

models will *be able to transfer* to use. This represents a significant contribution to the field, as no other model has been found in the literature that can make this claim.

6.4.15.2. SEMI-SUPERVISED MODELS

Three experimental results are considered as part of this dissertation. The first is the impractical example of an "all knowing" system that reports fully supervised true user state, which is intended to represent the best possible classification performance for any algorithm. The second example is complete lack of labeling information about user state to the algorithm of classification, which results in algorithmically encoded knowledge of classification (e.g. "Cluster #17") but not of state (e.g. "Bored"). The third example represents direct user query every few minutes, resulting in *some* algorithmically encoded knowledge of state. The difference between the first representation and the third is on the order of thousands of datapoints, but realistically represents the level of user annoyance. The difference between the second example and third is only five datapoints, but represents the difference between a program which requires user interaction and a background process.

The selection of appropriate classes for user query is an active learning problem in AI. This is complicated by the idea that the active learning conducted should also be realtime appropriate. The implementation of realtime algorithms with realtime active learning is a significant contribution to the field of AI for the reasons described in Section 5.2. The *invention* of realtime active learning components for online clustering (Section 5.4) is shown to *significantly increase* the number of usable models of affect (Section

6.4.12). This increases the number of usable models beyond supervised methods, as shown in the reprinted Table 62.

Table 62 – Summary of all ART and clustering usable models. Each number represents, out of 19, how many usable affective models were created. Reprint of Table 51.

Model	Anger	Boredom	Fear
Supervised ART	17	18	15
Unsupervised ART	6	7	12
Semi-Supervised ART	6	7	12
Supervised Clustering	9	9	12
Unsupervised Clustering	6	7	12
Semi-Supervised Clustering	11	9	16

6.4.15.3. REDUCED FEATURE SETS

There are two relevant findings resulting from the use of the reduced feature set. The first of these findings is that offline analysis can contribute to online analysis. This has ITS consequences in the limitation of physically applied sensors through the findings of linear regression models.

The second of these findings is that the algorithms presented here are fairly robust to noise. The use of features that did not contribute classification value, without reduced model performance, is an indication that the approaches taken in this dissertation are robust to noise. This finding can be exploited through the artificial creation of dataset features, and may result in higher overall model quality. While this was not done, for reasons of fair comparison to offline models discussed in Section 6.3.5, further work to exploit and examine this phenomenon is suggested in Section 7.3.

6.5.   Summary

While each of the research questions from section 6.4 has been answered in the preceding subsections, it is useful to include a summary of their answers. This summary is below:

1a, 2a, 3a: Can a quality *cognitive* model be constructed with *fully supervised, unsupervised, or semi-supervised* realtime algorithms?

> No. No usable cognitive model was created as part of this work.

1b.   Can a quality *affective* model be constructed with *fully supervised* realtime algorithms?

> Yes. Additionally, realtime affective models are of similar quality to their offline equivalents.

2b. Can a quality *affective* model be constructed with *unsupervised* realtime algorithms?

> Yes. Additionally, these are transferable to a field of use.

3b. Do *semi-supervised* and active learning approaches improve *affective* model quality?

> Yes. Invented methods are additionally shown to improve the number of usable models.

4. Does a change of parameter settings to reflect finer-grained clusters create higher quality models?

> Cognitive model quality was unaltered as a result of changes in parameter setting.

Affective model quality produced through clustering was slightly improved because of parameter setting changes, while other algorithmic performance was unaltered.

5. Does reducing the set of features to only the features used on cognitive model outputs create higher quality models?

Cognitive model quality was unaltered because of reduced feature set. This finding is indicative of the trend of not producing usable cognitive models.

Affective model quality was unaltered because of reduced feature set. This finding is indicative that a reduced set of sensors may be used, if suggested through offline analysis.

6. Do the *cognitive* models approaches generalize to another dataset?

No. No usable cognitive model was created on Dataset #2 as part of this work.

6.5.1. Summary Discussion Notes

The affective and cognitive models were built from the same input data. This presents the question: "Why are the affective models stronger in performance than the cognitive ones?". We present the idea that affective states are less transient over time. For instance, as shown in Appendix A-7, the HighEngagement metric reported from the ABM headset changes multiple times per second, ranging between high and low. In contrast, the Anger metric reported from the EmoPro measurement tool Appendix A-8,

changed only twice over the course of the training session for user 4102. This subject was affectively modeled nearly perfectly via a variety of algorithmic approaches.

Slower changes among the observed states are much easier to algorithmically observe among physiological and behavioral data, resulting in higher overall model quality. The EmoPro measure of affective state is a self-report metric, however, with the implication that a state cannot be labeled second-by-second. In order to label affective states in a more fine-grained fashion, personnel could be used to label states as they were observed. The collection of such a dataset to perform thusly is recommended in section 7.3.

Overall, this dissertation makes the contribution of a proof of concept that reasonable quality affective models can be created in realtime, presents several methods to use, determines which of these is most appropriate for the task, validates that these methods would transfer to the field, and invents an approach for boosting overall model quality. The implications of these findings, the discussion of areas of future work uncovered during this work, recommendations for other researchers, and a summary of this dissertation are included next.

7. SUMMARY, CONCLUSIONS, AND FUTURE WORK

Chapters 1, 2, and 3 of this work contend that Intelligent Tutoring Systems are useful; that they could be more useful with the creation of better models of student state; that the creation of improved student models has been met with limited success; and that this is primarily the result of poor engineering tradeoff decisions. Optimizing the accuracy of a model is not meaningful if it is not able to be used for the student. The algorithms presented in this dissertation have made a different trade-off decision; models should be useful first and accurate second.

Chapter 5 presents a framework for determining which algorithms are to be considered appropriate for this problem, selects a representative sampling of algorithms from the field, and improves upon their implementation through semi-supervision active learning. Chapter 6 shows and discusses the failure in creating cognitive models in this fashion. However, it also shows that the affective models created using these availability-driven approaches are comparable in quality to those ones that are accuracy-driven. Chapter 6 also shows that the adaptations for active learning, invented here, help to improve overall model quality. The implication of this work is clear: these algorithms create models that can are useful in application.

7.1.   Conclusions

There are many variations on the goal of the field of artificial intelligence, such as defining it as "The study of how to make computers do things at which, at the moment, people are better" (Rich and Knight 1991), "The study of the computations that make it

possible to perceive, reason, and act" (Winston 1992), or "The branch of computer science that is concerned with the automation of intelligent behavior" (Luger 2005). We choose to define the fundamental goal of the field of Artificial Intelligence as "emulating or surpassing human performance through the recognition of patterns and the establishment of pattern meaning for the purpose of producing action". Under this definition, it is useful to do so instantaneously, and while asking as few questions about the world as possible. Many AI approaches have been created for pattern recognition while looking at all possible data (ANNs, GAs, etc.), while fewer have been developed while looking at a single data point. Many AI approaches have been created to make use of a large amount of pre-classified data, while fewer have been developed to ask questions about observed trends. *All* of the approaches pursued in this dissertation attempt to solve what we consider the most fundamental problem in AI: instantaneous classification of patterns while simultaneously questioning their meaning.

Just as it is desirable to have a general purpose model of cognitive and emotional state for all individuals, it is desired for *one* algorithm to have near-instantaneous, near-perfect performance on *all* problems. The "No Free Lunch" theorem indicates that there is no *one* approach which will outperform *all* others on *all* problems (Wolpert and Macready 1997). These leaves the selection of appropriate algorithms to the AI expert (Rice 1975), at least until someone constructs an AI system which is able to *select* an optimal algorithm, rather than *implement* it (Gagliolo and Schmidhuber 2006; Kotthoff et al. 2011). Until such a time as this is complete, an AI researcher must hypothesize about the class of problem that he/she is given, and the types of approach which will be useful

for it. Given that this dissertation presents an approach that has never been attempted, the author has surveyed the field for applicable approaches.

Each chosen method represents a different approach to establishing models from data in realtime. Online clustering represents the method of dealing with online data of unknown classification through establishing and adjusting areas of the sampling space. Vowpal Wabbit represents the online approach to linear regression modeling, corresponding to the initial offline modeling approach chosen by the Dataset #1 experimenters. Adaptive Resonance Theory represents a neural network approach to online modeling, previously shown to have good one-pass learning results. Growing Neural Gasses represents the Self Organizing Map approach to establishing structure among data. Before testing, it was not known which of these classes of solution, if any, would be appropriate for the fundamental problem of rapidly establishing models from physiological signals.

The performance of supervised, unsupervised, and semi-supervised modeling algorithms on cognitive and affective models is summarized individually in Section 6.4.8, and 6.4.15, and in summary in Section 6.5. A brief review of this summary is that realtime cognitive models (Distraction, Engagement, Workload) were not able to be constructed with any algorithm (ART, clustering, VW, GNG), labeling approach (supervised, unsupervised, semi-supervised), parameter settings, feature set, or Dataset, while affective models were able to perform acceptably with ART and clustering in all circumstances. Additionally, realtime semi-supervised active learning, as implemented in

the clustering approach, was shown to have significant impact for affective model creation, and the two most successful algorithms are shown to be robust to noise. However, this work was not performed without issues or surprises.

7.2. Issues and Surprises

In general, there were fewer issues than surprises encountered during this dissertation. The primary issue faced during this dissertation was the implementation of each algorithm. Vowpal Wabbit is written in C++ and incorporated through the use of precompiled binary with executable wrapper code (written in Python) and library functionality code (written in Python). Online Semi-Supervised Growing Neural Gasses is written in C++ and incorporated into Python through use of a program to automatically generate software interface libraries, after learning the software interface library configuration process. Adaptive Resonance Theory was implemented in C, and then re-implemented in Python. It was simpler to just re-implement the tested and invented clustering algorithm in Python, given the simplicity. All of these were encoded into library, threaded, and tested using the same controlling program in order to assure fair evaluation. Cross-language, library-driven, thread-safe support for programming has certainly come a long way in the last decade, but is still a non-trivial issue, and was the largest issue overcome during this dissertation process.

There were a few surprises encountered during this research. The first of these is that majority of researchers in Intelligent Tutoring Systems appear to be generating *recommendations* for software, rather than the software itself. This is in stark contrast to

the research performed in Artificial Intelligence, where a new algorithm is developed for a research paper, proven successful, and posted on the internet for wide distribution. A byproduct of this trend is that no form of student modeling or dataset from other ITS researchers could be used as part of this work. Research undertaken as part of this work is anticipated to transfer to the field through implementation as open source software and made publicly available, in alignment with the AI field.

The next surprise was that there has been a dearth of research in the field of realtime datastreams. AI research has focused on classification accuracy, function approximation, statistical modeling, and optimal choice within finite state machine simulations. The algorithms implemented in this dissertation are research byproducts from the problems of credit card fraud detection, identifying pirate traffic in network analysis, and classification of webpages to optimize search results. These are relatively unlikely places to find AI for student modeling. It appears that the field abandoned the idea of rapid problem solving in the mid-1990s, along with the rise in processing power. Research addressing realtime semi-supervised and active learning is similarly sparse.

The OSSGNG and VW algorithms were predicted to perform better than the ART and clustering algorithms. OSSGNG and VW had implemented semi-supervised (OSSGNG) and active (VW) learning methods already, and had shown good performance in publication. It was surprising to see that the research in this dissertation outperformed these two approaches to a level where their performance was not worth in-depth discussion. This surprise further indicates that algorithms for realtime semi-supervised

active learning on datastreams have significant room for improvement, as the implemented improvements are relatively intuitive in nature.

The online models produced during this research are individualized, rather than generalized, which makes comparison to the offline models somewhat different. In this fashion, the offline models are able to drastically outperform their online counterparts. The finding that the online affective models can match the performance of the offline affective models was unexpected. It was expected that the online models, given a fraction of the data and time, would perform somewhat worse. It was surprising that they were able to compare favorably, despite significant limitations.

Lastly, it was surprising that the online cognitive models were of low quality, when contrasted with the offline models. This is discussed in significantly deeper depth next, in Future Work.

7.3. Future Work

Part of the goal of the publication of any research project is to put the work in a larger context. This work directly interfaces with many fields, including machine learning, computer programming, architectural development, instructional strategy selection, human computer interaction, modeling and simulation, classroom instruction, and others. The work in these areas is not yet finished, and here we will present some of the problems uncovered during the course of the research. These future research efforts are structured from the "ground up", first dealing with AI and datastream problems and lastly discussing instructional implications.

Several approaches may assist in the creation of realtime models of cognition and affect. In short, they are windowing techniques, feature extraction techniques, feature expansion, improvements in realtime active learning, collection of a new affective dataset for validation and comparison, merging this work into an ITS framework to provide back to the field, and initial adjustment of instructional strategy based on state. These approaches are discussed next, after a focused discussion on the hypothesis most likely to produce usable models of cognition.

7.3.1. Feature Extraction

Realtime preprocessing of a datastream for feature extraction purposes is a related research vein. This can include statistical metrics, such as the mean/median/mode/standard deviation inside of a window, extrapolation of trend, traditional electrical engineering approaches such as a high pass filter, derivatives, or other approaches. A given problem may have more than one type of filtering approach taken in realtime, such as the band-pass filtering, derivative, squaring, integration, and thresholding of the QRS signal present in heartbeats (Brawner and Goldberg 2012; Pan and Tompkins 1985). It is likely that a developed approach will be specific to the physiological signal that it models, while all of the methods presented in this dissertation could adjust to an additional dimension of data without underlying algorithmic modifications. Preprocessing development is signal-specific, while realtime processing is signal-agnostic. The types and variations of realtime physiological signal filtering are interesting areas of research.

7.3.1.1. STATISTICAL FEATURES

Parameter adjustment for cognitive models from the initial parameter set to the revised parameter set had no effect on the quality of the models. Overall, the number of classifications or clustering categories was doubled as a result of these adjustments. It is surprising that such an increase in the granularity of sampling had no overall effect on model quality. This observation leads us to believe that offline, historical, and trend data are important to the overall construction of the cognitive models, as is the case with the ICA metric.

Given that the quality of affective models did not diminish significantly through the addition of features determined by offline modelers to be 'noise', the injection of a single statistical feature was attempted as part of this dissertation work. A five second moving window average was added to each of the 21 features of Dataset #1, resulting in 42 total features. Each of the methods of supervision and algorithms was tested against this new dataset. This single feature was not observed to increase total quality of either affective or cognitive models, and is shown in Appendix C-4.

The injection of this single feature is only an exploratory analysis for how much additional data should be considered in a statistical feature. Varying the length of statistical feature extraction should be considered, as well as other methods for feature extraction. A summary of statistical modifications which may be attempted is shown in Table 63.

Table 63 – Summary of signal agnostic statistical feature extraction techniques

Approach	Example
Rolling Average	Average of the last 5 seconds (Appendix C-4)
Variance	Variance of the last 5 seconds
Standard Deviation	Standard Deviation of the last 5 seconds
Root Mean Square	RMS of the last 5 seconds
Derivative	Average derivative of a smoothed 5-second signal
Integral	Average integral of a smoothed 5-second signal
Signal Power	Square Root of the integral of the second derivative of the signal (Brawner and Goldberg 2012) over the last 5 seconds
Variations in time	All above approaches, 10 seconds rather than 5

7.3.1.2. SIGNAL SPECIFIC APPROACHES

It is possible for each of the sensor signals to have customized feature extraction methods, which is likely to boost overall performance of the cognitive modeling techniques for the rapidly changing signal. This is opposed to the direct signal values used by the offline modelers and the comparison work in this dissertation. The methods taken in this dissertation have relied upon *direct* AI methods of modeling so as to generalize to differing sets of sensors. Future attempts at cognitive models should attempt signal-specific feature extraction techniques.

It is likely that feature extraction will play a key role in the future development of cognitive models. As an example, consider the P300 Event Related Potential, which is embedded within EEG signals (Donchin et al. 2000). The P300 event related potential has been linked to a number of neurological phenomena, and is an aggregate measure from multiple simultaneous EEG channels of data. Efforts to detect this signal in realtime have been met with mixed success (Donchin et al. 2000). This feature detection

is performed *prior to* being used in AI methods (Bostanov 2004). The methods presented in this dissertation rely upon direct processing of raw EEG data, and may not successfully group the P300 event related potential appropriately.

This feature extraction is very specific to the signal in question and does not generalize to unknown signals, unlike all of the methods presented in this dissertation. Any feature extraction undertaken during this dissertation would not result in a fair comparison to offline methods, as discussed in Section 6.3. Furthermore, all of the methods taken here are appropriate to *all* sensor datastreams, while the creation of customized feature extraction for one of the twenty-two dimensions of the input set will *not* be appropriate for general inclusion. A summary of specific feature extraction methods which may be appropriate for generating higher quality realtime models is included in Table 64.

Table 64 – Summary of signal specific feature extraction techniques

Sensor	Feature Extraction	Citation
EEG	Shannon Entropy	(Stevens and Galloway 2013)
EEG	P300 Region Activity	(Dal Seno et al. 2010)
Heart	Time since between last beat (heart rate)	(Pan and Tompkins 1985)
Heart	Heart Rate Variability	(Malik et al. 1996)
Sonar (Distance)	Kalman Filter (for tracking)	(Welch and Bishop 1995)
Sonar (Distance)	Leaning information (forward/backward binary feature extraction)	
Chair Sensors	Posture by Mixture of Gaussians	(Mota and Picard 2003)
Chair Sensors	Activity Level (low/med/high)	(Kapoor and Picard 2005b)
Eye Tracking	Discrete Wavelet Transforms	(Candes et al. 2006)
Eye Tracking	Scale Invariant Feature Transform	(Lalonde et al. 2007)

7.3.2. Intelligent Tutoring Systems

The first three chapters of this dissertation contend that learner models of affect and cognition can aid in the selection of a learning strategy, and that a learner model should be created using an individualized and realtime approach. The next three chapters show that it is possible for this to be performed for classification of affect. The clearest avenue for future work is the integration of this work into an intelligent tutoring system.

The methods presented here for realtime modeling were not created for the purpose of creation. The use of these methods has been a driving force behind their development. The logical next step is to merge the work presented here into an intelligent tutoring system, whether for testing, validation, or use. At the time of this writing, the Generalized Intelligent Framework for Tutoring (GIFT) project by Army

Research Laboratory has over 200 users, two running experiments, four planned experiments, and an upcoming workshop at the Artificial Intelligence in Education conference. It is anticipated that the next release of the GIFT framework will incorporate the researched improvements in individualized student modeling, as the author is very familiar with the project, developers, controlling organization, and timeline of the project. The outputs of this dissertation are intended to be presented back to the field through integration into this community-driven research platform, with the recommendations for parameter settings chosen in APPENDIX D.

GIFT has been designed based on the idea of a learning effect chain, as shown in Figure 44. This has the derived requirement for separable software modules, which have defined inputs and outputs, as shown in Figure 45. The defined process of the learner module is to take sensor and performance data and form it into a "picture of the learner" from which to make pedagogical decisions. The work in this dissertation has been specifically targeted to make this type of decision.
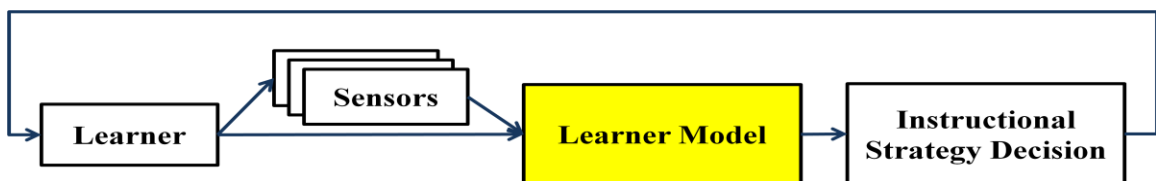
Figure 44 – Learning effect chain diagram which drives GIFT development (Sottilare et al. 2012b). Learner model is highlighted for effect of indicating where this research is intended to transition.
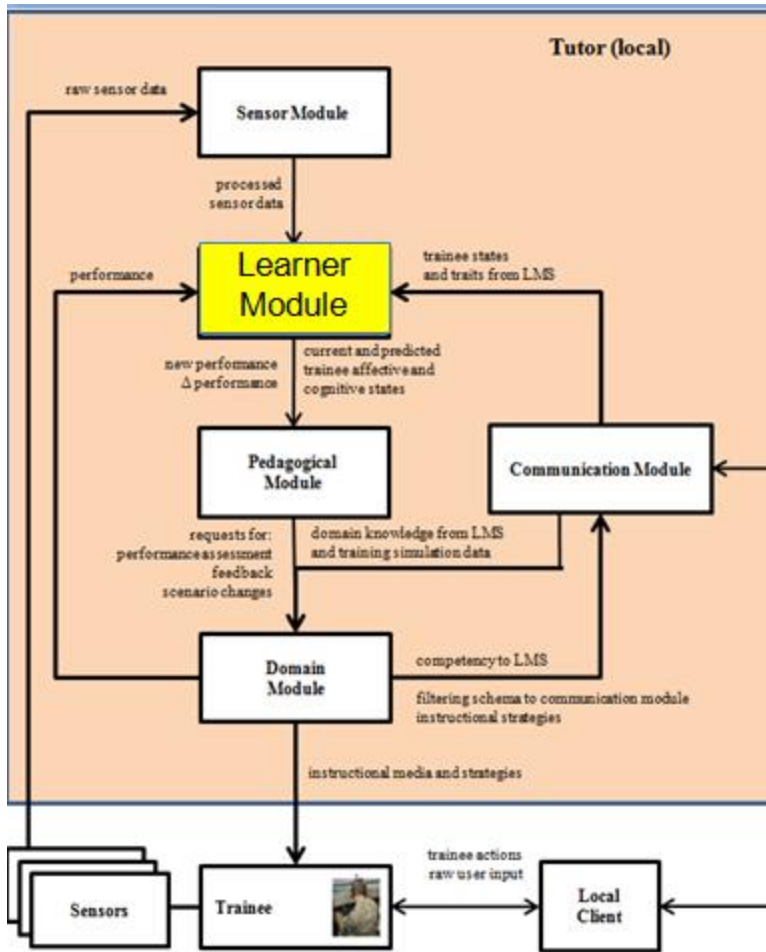
Figure 45 – Derived GIFT diagram of functional modules (Sottilare et al. 2012b).

Of course, knowledge of student state is not enough information, by itself, to inform how instruction should be adapted. For example, a learner which is anxious during test-taking may require no instructional intervention, while a leaner anxious during initial training exposure may need the pace of material presentation slowed. GIFT 3.0 presents a framework for pedagogy, as informed by state classification machines that adjusts content. Figure 46 shows an example of a prototype authoring interface, developed by Dignitas Technologies, with the purpose of creating such a relationship.

Other work is done by the University of Central Florida's Institute for Simulation and Training to create domain-independent pedagogy (Goldberg et al. 2012). The functional architectural component of GIFT which uses this technology is called the Engine for Macro-Adaptive Pedagogy, or EMAP. Further developments are currently in process for a strategy recommendation engine for micro-adaption, which will likely be more state-dependent than its macro-adaptive counterpart.
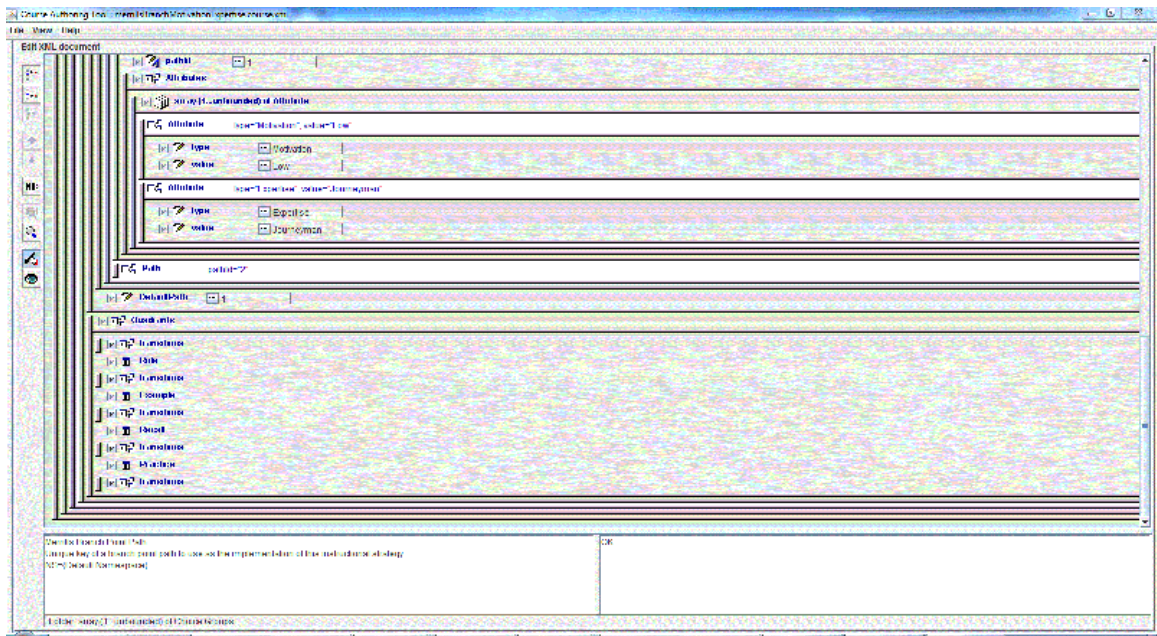
Figure 46 – Possible adaption of instructional pedagogy based on Merrill's Branching Theory and learner variables. Learner variables may be either sensor/state-driven or survey-driven.

Work in this dissertation to classify affective and cognitive states is intended to function as a part of architecture to support intelligent tutoring. The GIFT architecture is the intended architecture for the transition of this technology. It already collects various sensor characteristics such as electro-dermal response, and posture data from the

Microsoft Kinect. It makes instructional strategy recommendations based on a decision tree of traits, states, and performance. It does not, however, contain a module for merging performance and sensor data into states for decisions. The work presented in this dissertation is the first of its kind to do so in a manner which can withstand validation; this presents a clear path for use.

7.3.3. Other Avenues for Future Work

The first of these other approaches is that a windowing technique may be more appropriate than initially supposed. It is difficult for an algorithm to build a model of the entire datastream while only being able to adjust to the smallest mathematically possible slice of it at any time. Windowing techniques and additional derived measures may assist in the creation of a model by giving trend data, reducing noise, or eliminating true outliers. The examination of how to create the correct window size which balances the explicit delay in real time performance against the benefits of multiple data point analysis is an interesting problem.

One of the findings in this work is that all realtime model construction approaches are relatively insensitive to the injection of extraneous data. This is an interesting finding which is worth investigating further, as it has consequences for research in datastream filtering. If it is known *a priori* that the creation of additional features will not meaningfully impact the construction of a model, then it is advantageous to create many features. For instance, a 1-dimensional feature of GSR may be expanded into many features such as: mean over the last 3 seconds, mean over the last 5 seconds, standard

deviation over the last 3 seconds, signal power (Brawner and Goldberg 2012) over the last 300 miliseconds, or others. The expansion of features may present a simpler problem to algorithmic processing, as only a few signal values that are correlated with the true label are needed over the entire featureset. It is unknown if multiple-filtering for dataset expansion is harmless to overall accuracy, as this would have resulted in unfair comparison to offline models, but leaves room for future research.

The current methods for realtime active learning leave something to be desired. The determination of the confusion that an individual point contributes to the whole of the model, without examination of the model, is a difficult problem. Realtime methods of active learning are not readily available, and had to be invented as a part of this dissertation. A few ideas to improve realtime active learning techniques include attempting to get a label when the most recently presented datapoint is determined not to belong to any of the previously observed clusters, requesting the label of a point which is near to the current fringe of a cluster, and propagating the label of a point across clusters and points for a short period of time. The effect of any of these decisions is currently unknown, and presents an interesting vein of research.

An interesting question has been asked of the author many times during the writing of this dissertation: "After this model has been built, for an individual, in realtime, what do you do with it?". The answer, currently, is to discard it. The research indicates that static individualized models degrade in quality over time, as the individual changes. The research presented in this dissertation presents methods for dynamic and

individualized approaches which are able to adapt to individual trends over time. Do they degrade? Is there a benefit to keeping a model created in a previous training session? To which sensors do such a benefit, if any, extend? The evaluation of transferability of an individualized model requires an experiment where individuals are brought back into an experimental or laboratory setting after a period of absence. The author is not aware of a dataset which has measured this type of learning interaction.

Another interesting area of future research is the validation of the techniques of realtime monitoring of the student. The affective technique is somewhat validated with the creation three sets of models, but further validation should be performed. Unfortunately, there is not a data set on which to validate these measures, as discussed in Chapter 4. As part of this research, it has come to the attention of the author that such a dataset would have meaningful contribution to the body of research. A project of this nature, informed by the research done in this dissertation, may involve an unobtrusive and wearable sensor or Kinect sensor (to replace a motion sensor and the chair sensors), and fine-grained affective coding. A project of this nature could validate their approach on the dataset used as part of this dissertation, and should meet the requirements of Table 10, the checklist of features dataset inclusion.

It is possible that interactive user query will result in overall better quality models, as the algorithms are fed misinformation in the time between initial outlier classification and true class label. It is intended to test this hypothesis with affective data that has finer resolution, such as described above. The problem of how/when to query the user to add

information about a state or cluster is still an open problem of research. Although the work performed in this dissertation shows it is not often required, this has yet to be validated experimentally.

The models created by other researchers have classified learner state into one of two categories, forming a binary classification problem. For example, a learner is classified as 'anxious', or 'not anxious'. There is research which indicates that binary classification may not be most appropriate to the task (Eysenck and Calvo 1992; Wine 1971). This research indicates that a moderate level of anxiety results in the ideal state. Further work should be undertaken to classify the various values of varying state on a 3-point, 5-point, or 7-point Likert scale (Likert 1932).

7.4.   Dissertation Summary

Intelligent tutoring systems should mimic human tutors in order to achieve greater gains in learning. Doing so involves monitoring affective and cognitive states of users as they interact with the tutor. "One size fits all" generalized models have been shown not to transfer to practical application because individuals are different from each other. Individualized models, however more accurate, are also unusable, primarily because of normal variations in behavior and physiology. Only individualized models with very rapid creation times are hypothesized to create instructional value, but they have never before been created.

This dissertation presents four methods for the creation of four types of cognitive and three types of affective models, and experiments with how often the "true" label

information, provided by the student, is needed. It concludes by determining that more research is needed for the rapidly-changing cognitive states, but that individualized affective models can be rapidly created with minimum degradation in quality. Furthermore, it was found that these models can be created with minimal information about the true affective state of the user.

The ability to affectively model the student presents a possible solution to informing pedagogical instruction, such as instructing 'bored' students differently. By modeling individual learners, instruction can be more effectively individualized and overall learning can increase. The methods presented here detail how to do so for affective states, and show promise towards doing the same with cognitive states. This research is *significant*, as it addresses what other researchers have considered a significant problem, *novel*, in that new algorithms were created for the purpose of solving this problem, and *useful*, in that it is proven to be applicable to the field.

APPENDIX A   GRAPHS OF SENSOR MEASUREMENTS FOR PARTICIPANT 4104 FROM DATASET #1

The below graphs from Dataset #1 are shown in the fashion that they are given to the machine learning algorithms described throughout this dissertation. Each feature of each sensor is shown one dimensionally for clarity, but is input as a batch. The x dimension of each graphs is "number of datapoints", which corresponds to time. The number of datapoints corresponds to approximately 40 minutes of data, but varies for each participant. The y axis of the below figures is a normalized measure of the sensor output. This normalization requires that the y axis has no units. A brief description of the measurements of each sensor is included for completeness.

Appendix A-1   Neurosky Measurements for Participant 4104

Alpha is a measurement of neural oscillation in the frequency range of 8-12 Hz. In general, increased activity in the alpha band has been correlated with drowsiness and sleep. They have been detected at higher levels during meditation and relaxation. The Alpha and Alpha2 represent the readings on the left and right side of the forehead from the Neurosky sensor.

Alpha2
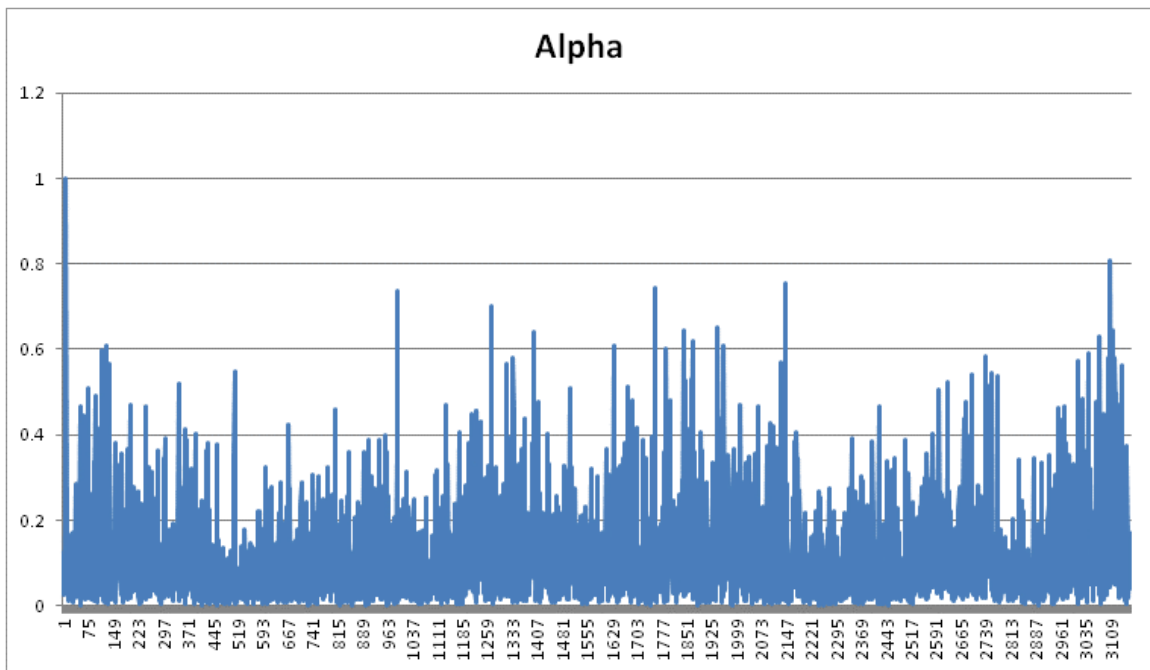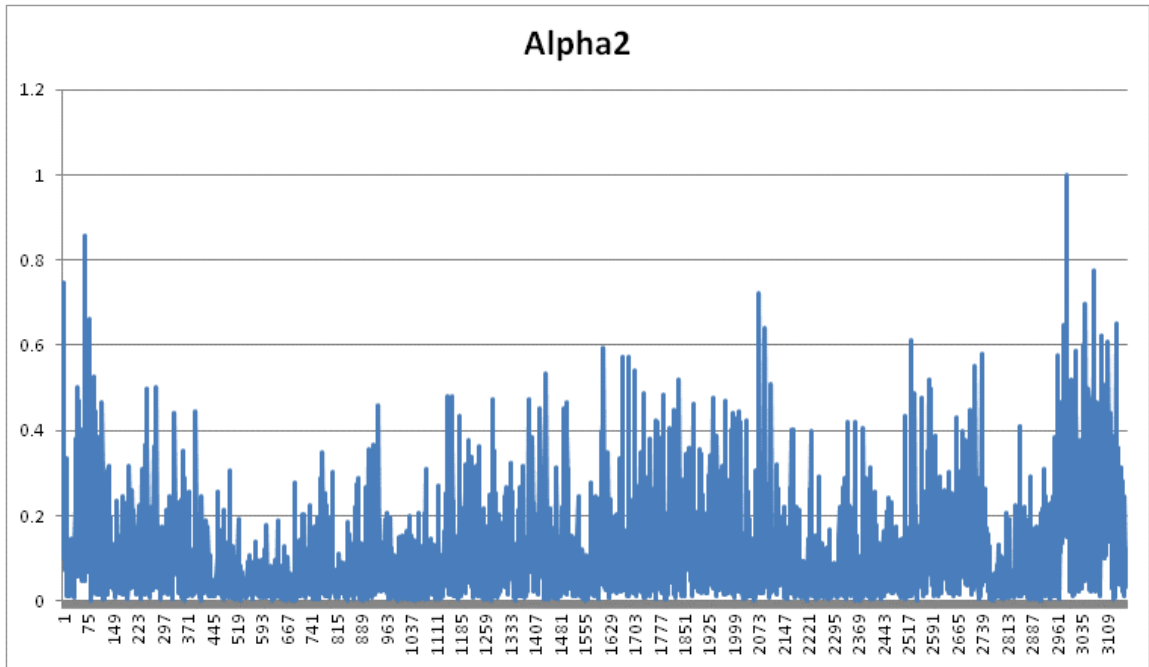
The gamma brainwave is measured between the 25-100 Hz frequency. Higher frequencies have been linked to language and cognition (Benasich et al. 2008). It is possible that gamma waves represent a mis-measurement of EEG signals, and instead correspond to small eye movements (Yuval-Greenberg et al. 2008). Either of these features may be of interest to cognitive and affective models. The Gamma and Gamma2 represent the readings on the left and right side of the forehead from the Neurosky sensor.

Gamma2

274
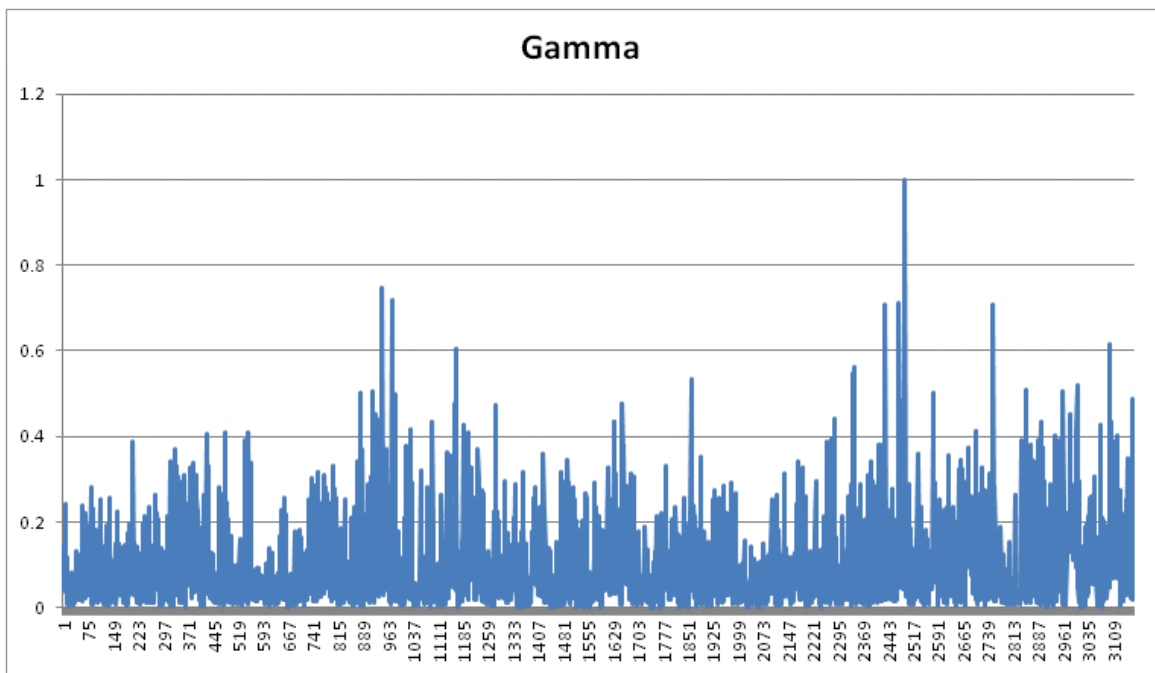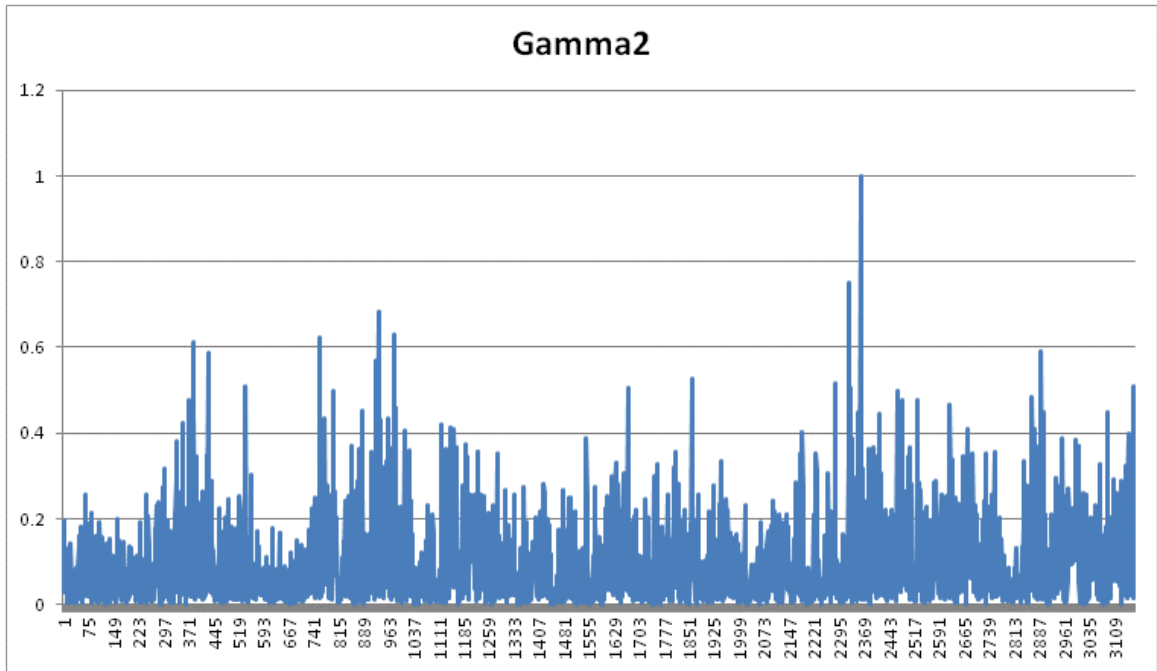
The beta brainwave is measured between the 12-30 Hz frequency. The beta wave is associated with normal waking consciousness and interacts with the alpha wave during cognition (Pfurtscheller and Klimesch 1992). Responses in the motor cortex are also known to increase the prevalence of beta waves.   The Beta and Beta2 represent the readings on the left and right side of the forehead from the Neurosky sensor.

Beta
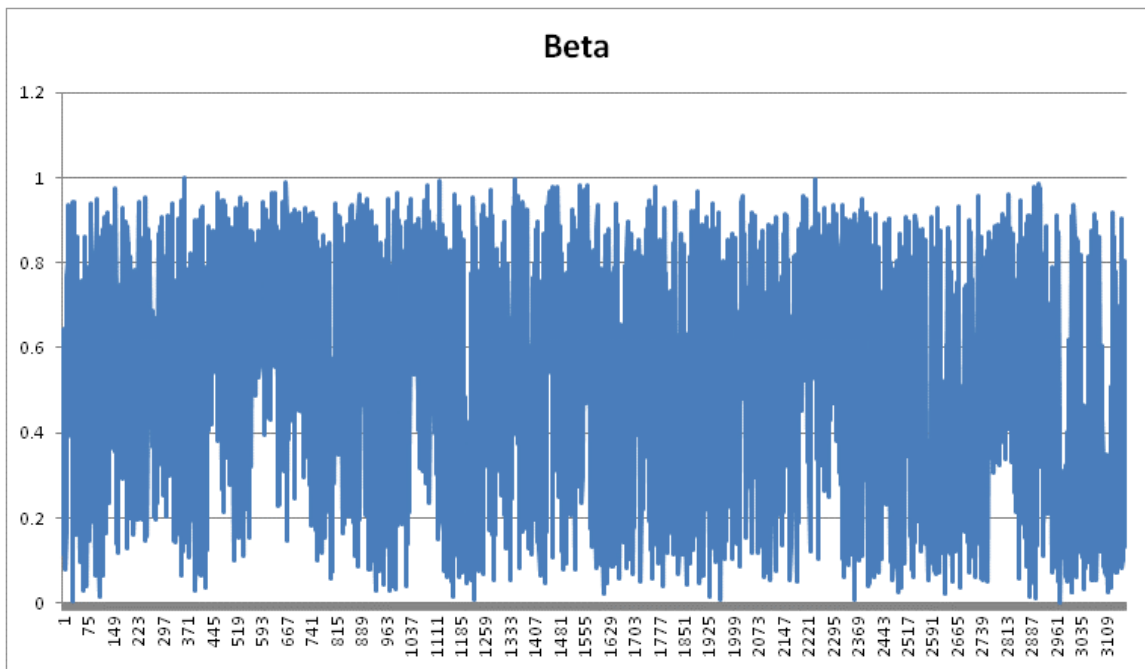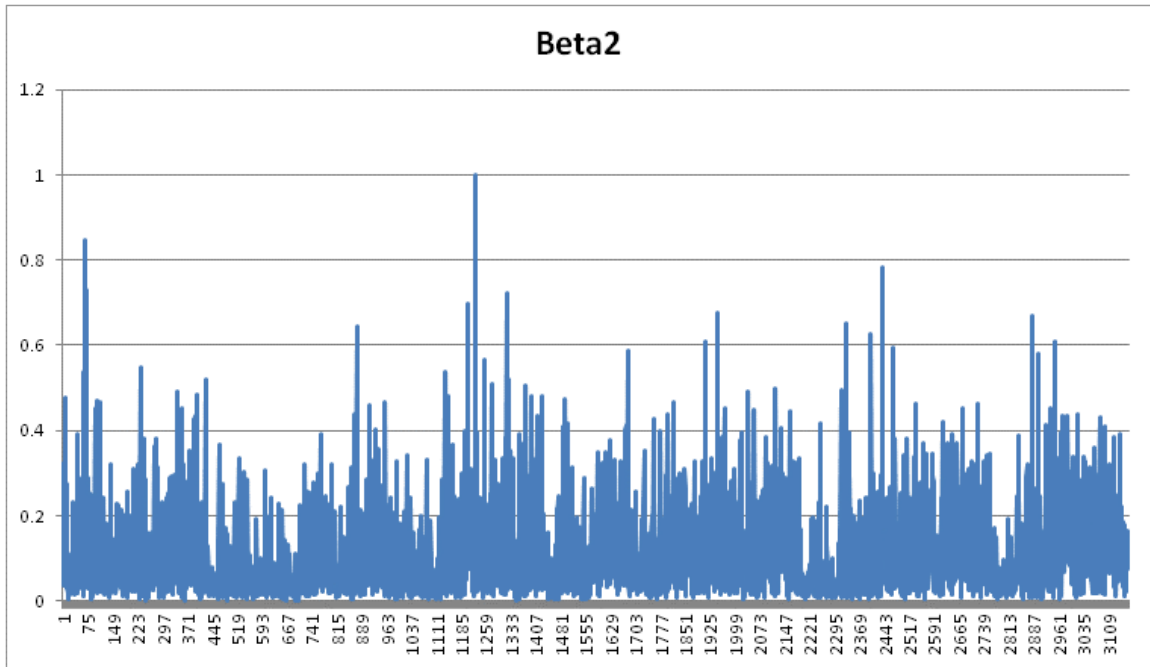
The Delta wave is measured between 0 and 4 Hz. It is associated with the deepest stages

of sleep, and is used to characterize the depth of sleep (Tononi and Cirelli 2006).

Delta
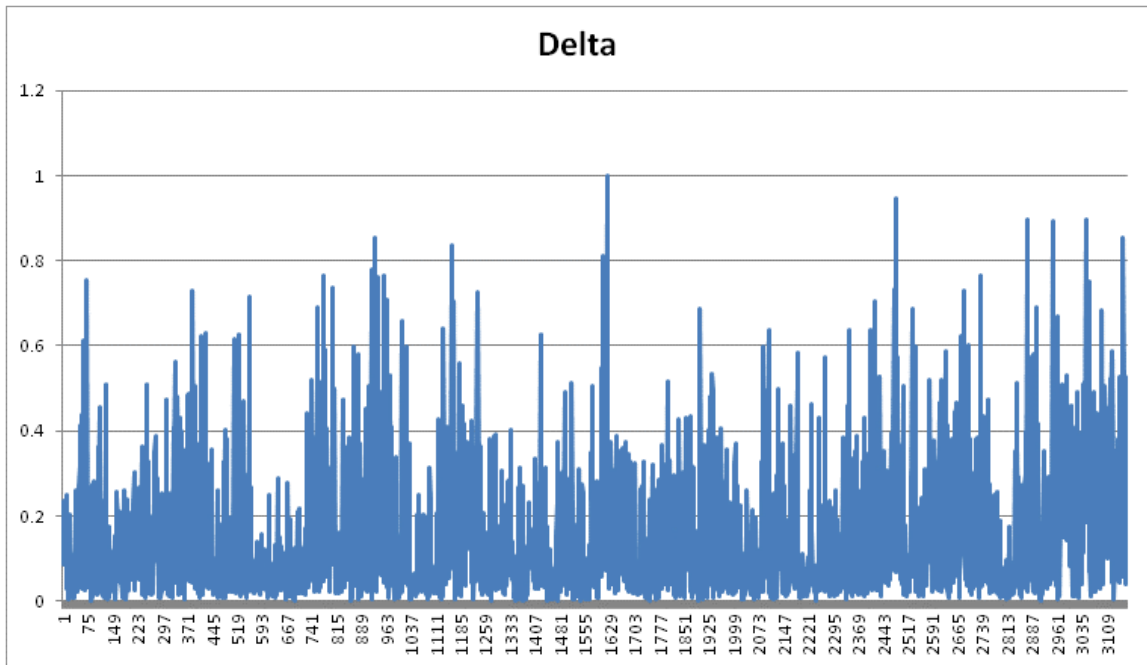
The Theta rhythm is measured between 6 and 10 Hz. It is not well understood, but may be linked to exploration, learning, memory, or motor cortex function.

Theta
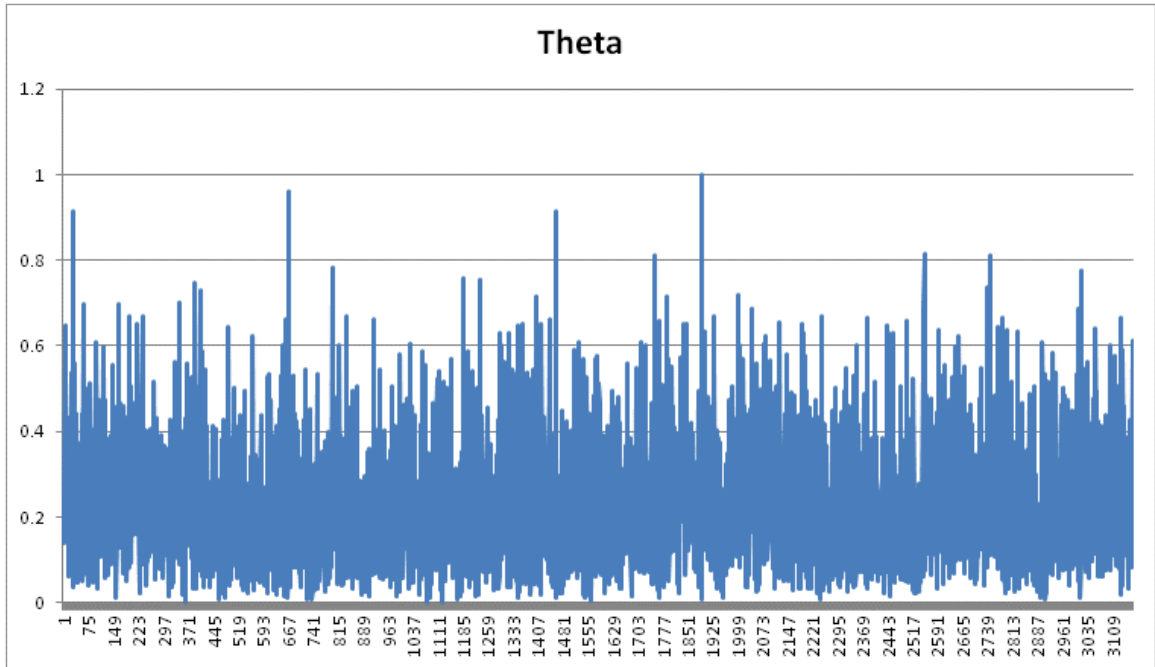
Meditation is a metric produced via unknown combination and weighting from the proprietary NeuroSky sensor. It has not been validated, but has been tested against 30 expert meditators. This metric has been able to differentiate between problem-solving tasks and previously-validated psychological batteries (Crowley et al. 2010). In theory, high measures show when someone is meditating.
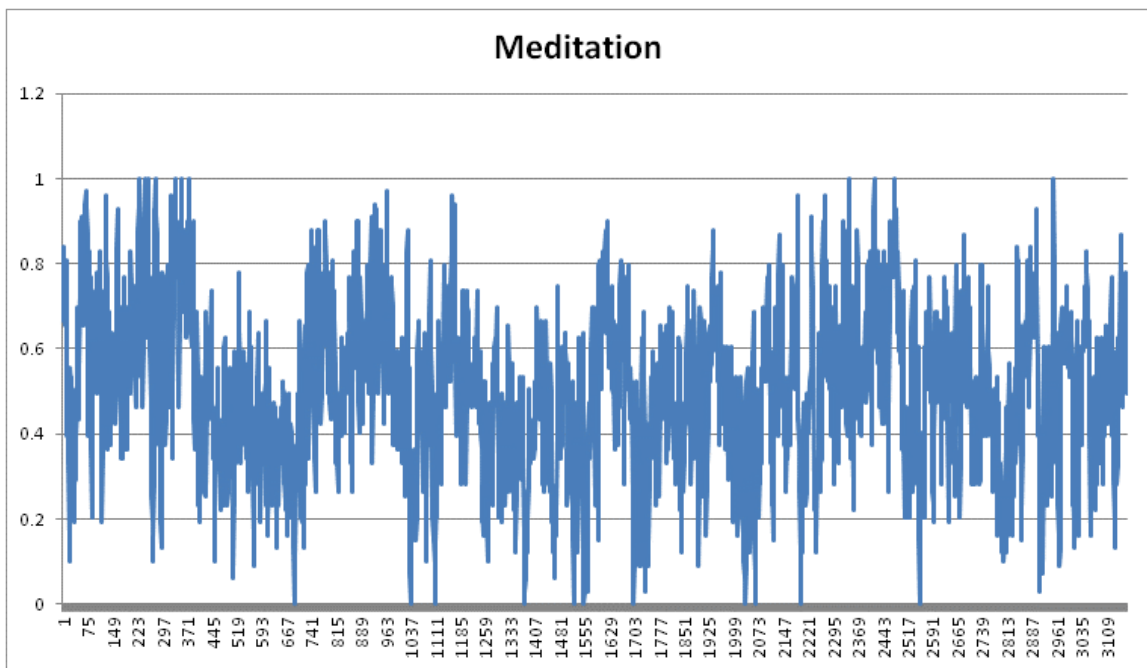
Attention is a metric produced via unknown combination and weighting from the proprietary NeuroSky sensor. It has not been validated, but has been tested against 30 expert meditators. This metric has been able to differentiate between problem-solving tasks and previously-validated psychological batteries (Crowley et al. 2010). In theory, high measures show when someone is dedicating cognitive resources.
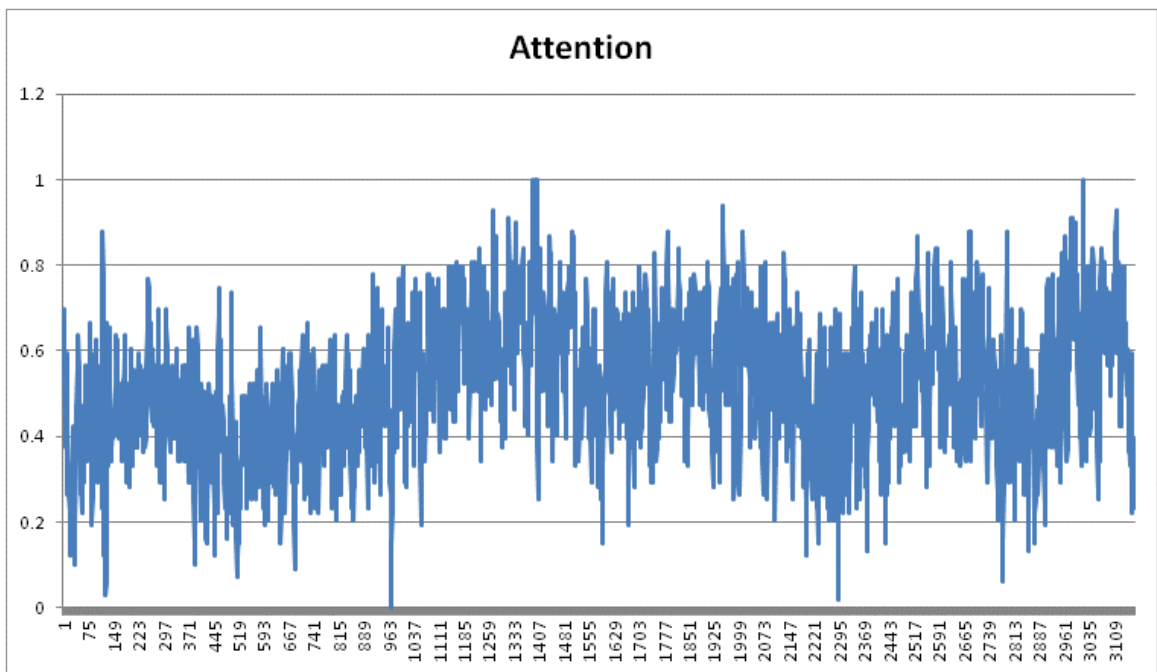
Appendix A-2   Zephyr Heart Measurements for Participant 4102

The Heart measure measures heart rate over time through heartbeat detection methods from the Zephyr Heart sensor. High measures correlate with higher heart rate which correlates with higher levels of bloodflow, stress, excitement, and psychological arousal (Anderson and Brown 1984).
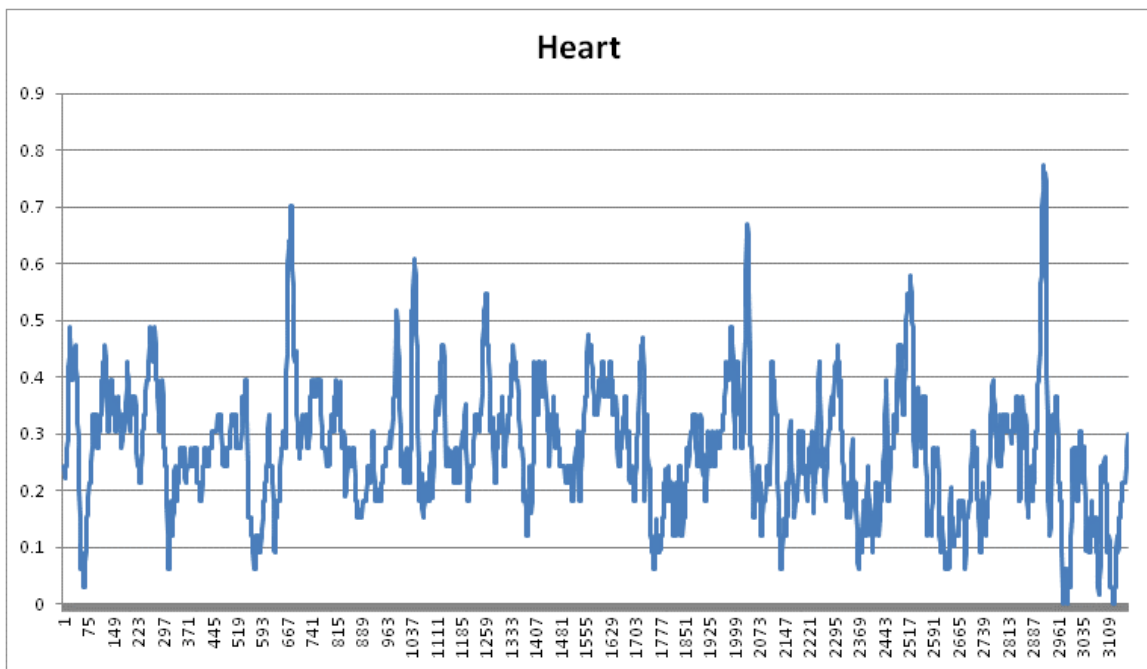
Appendix A-3   Sonar Distance Sensor Measurements for Participant 4102

The Motion measure from the sonar senor record how far a participant was from the computer. Higher measures indicate that the participant was further away while lower measures indicate closeness. These behaviors generally mean different things for different individuals.
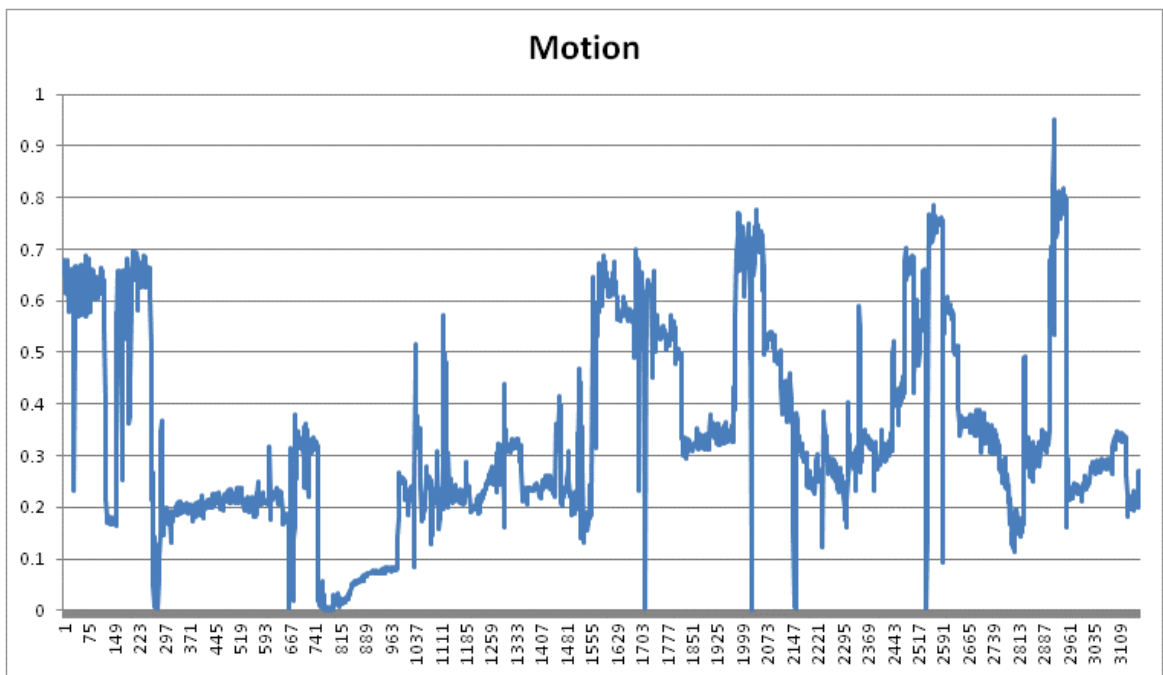
Appendix A-4   Sensor Chair Measurements for Participant 4104

The measurements from the sensor chair correspond to the pressure on each of the eight sensors. Sensors numbered one through four were placed on the back of the chair and generally show little variability from any participants. Sensors numbered five through eight show significantly more variability. It is unknown how these measures correlate with cognitive and emotional states, aside from that they are used in the Linear Regression models used by the original experimenters. These measures generally mean different things for different individuals.

Sensors 1 through 4 measured the amount of pressure on the back of the chair. These did not always result in a non-zero reading. Note that Chair Sensors 1-4 are the *only* feature of Dataset #1 not used in any cognitive or affective model. See Table 18 for more information.

Sensor1

Sensor2

Sensor3

Sensor4

Sensor5

Sensor6

Sensor7

Sensor8

287
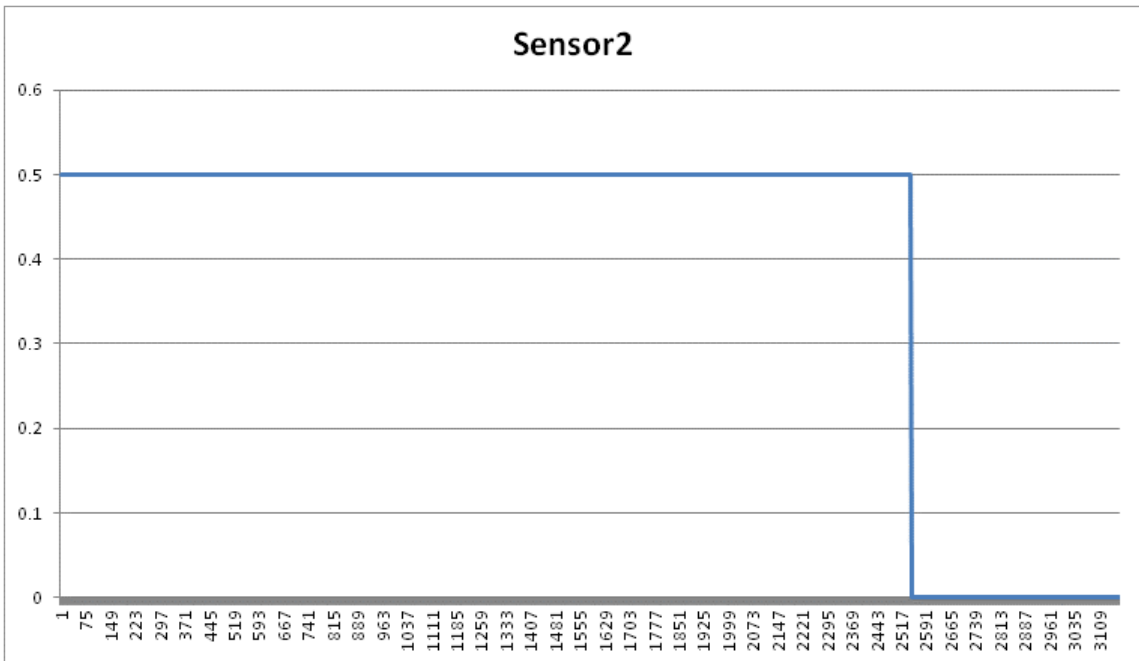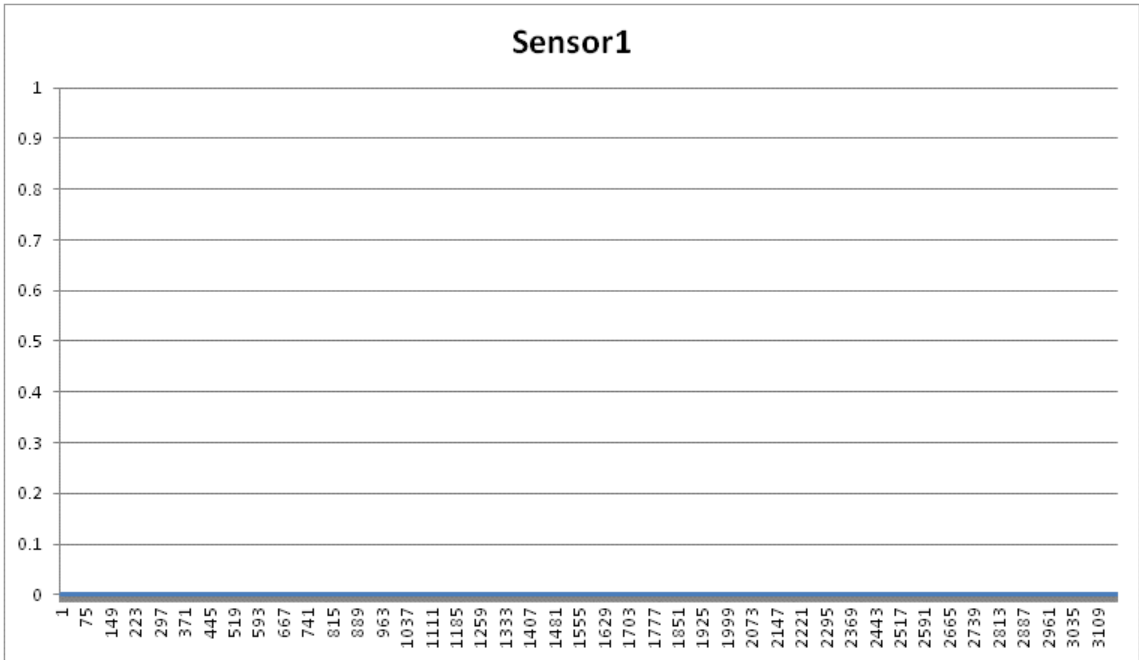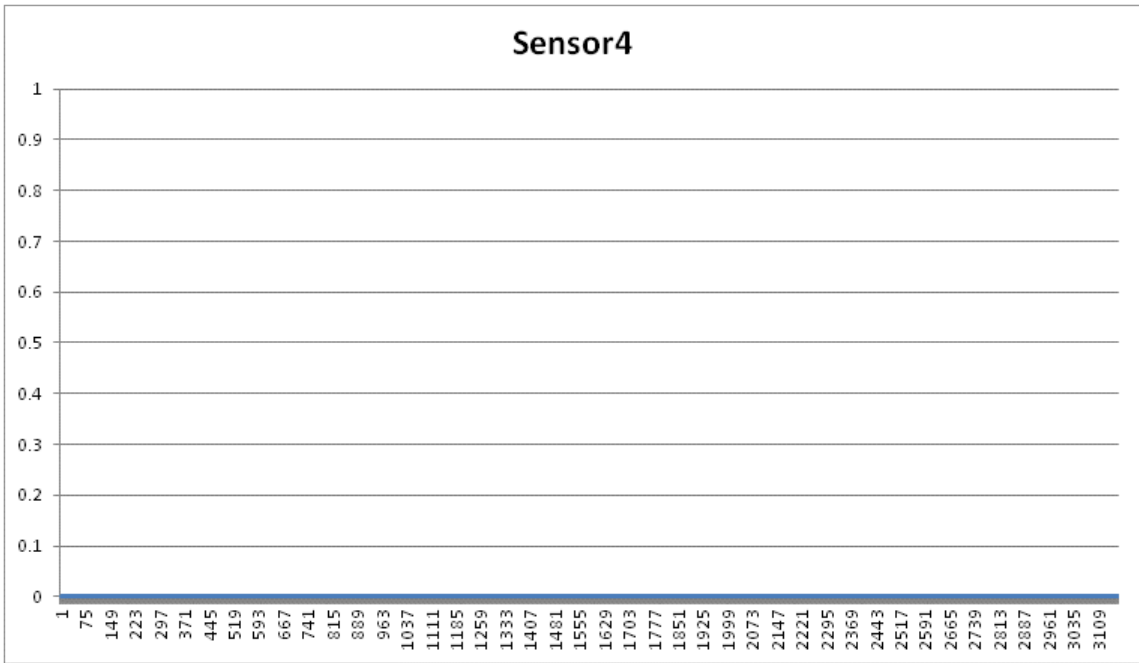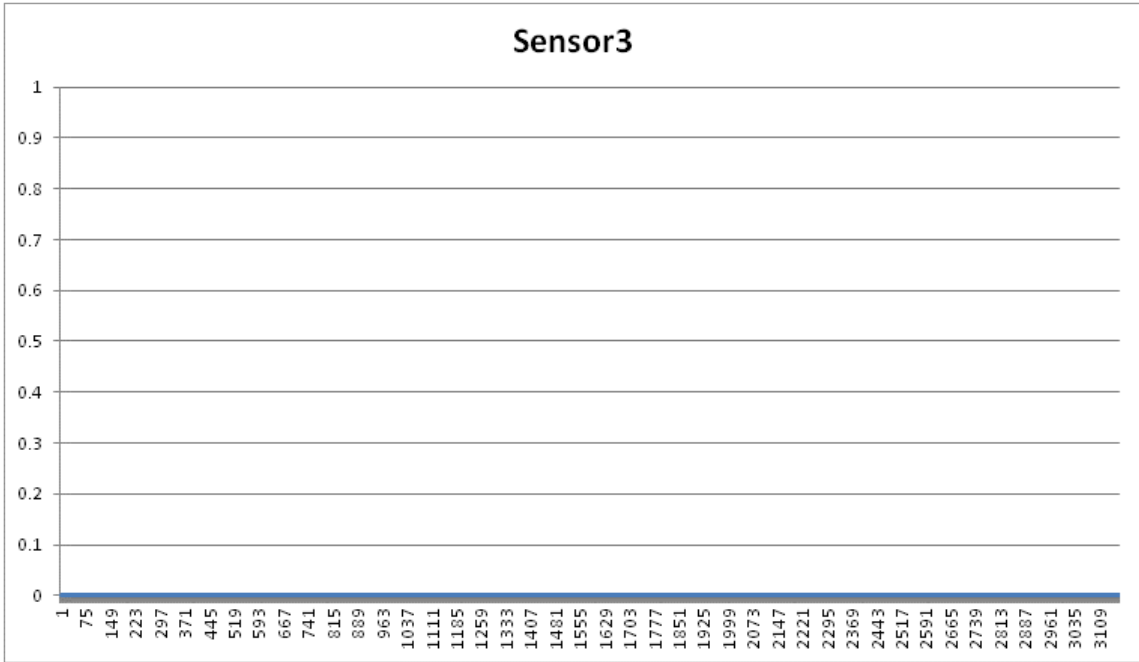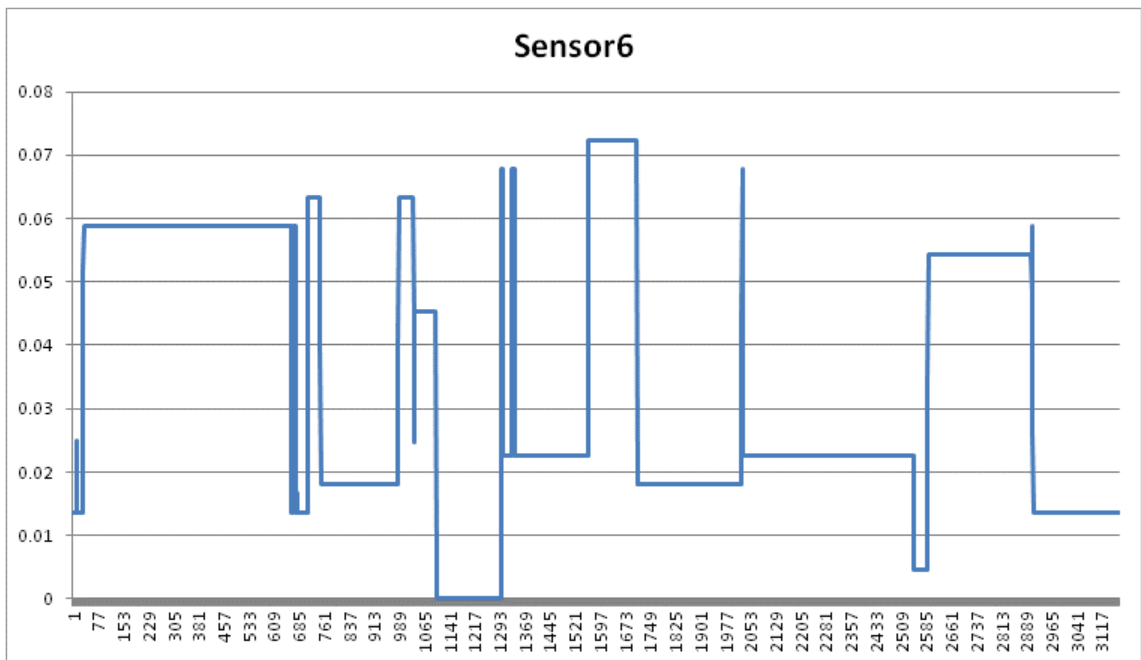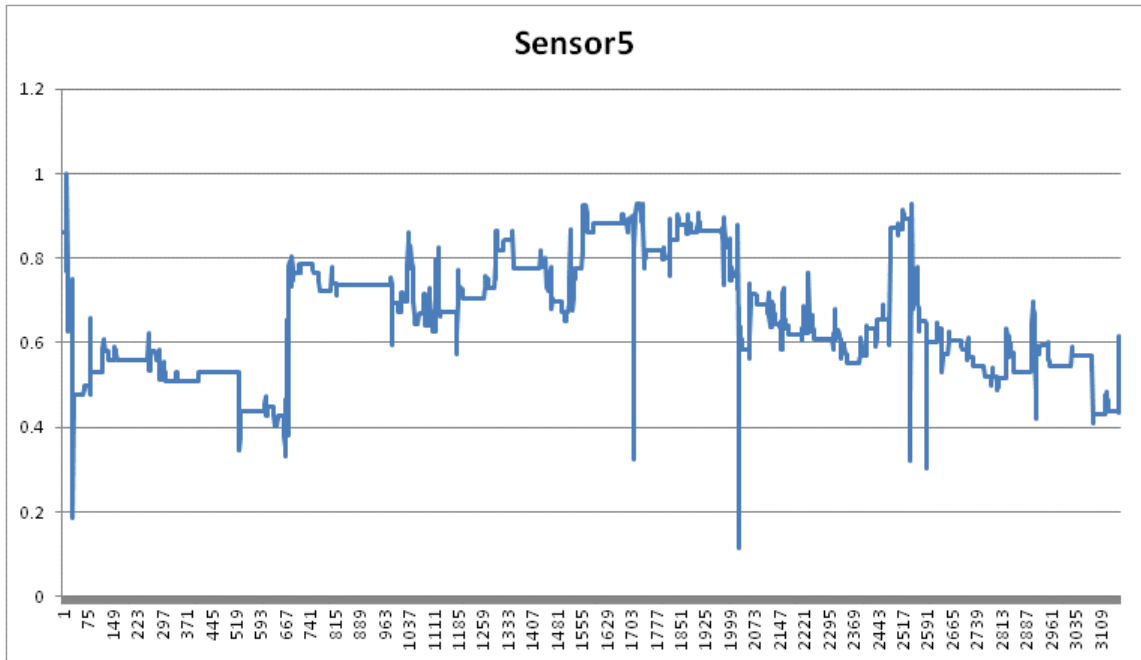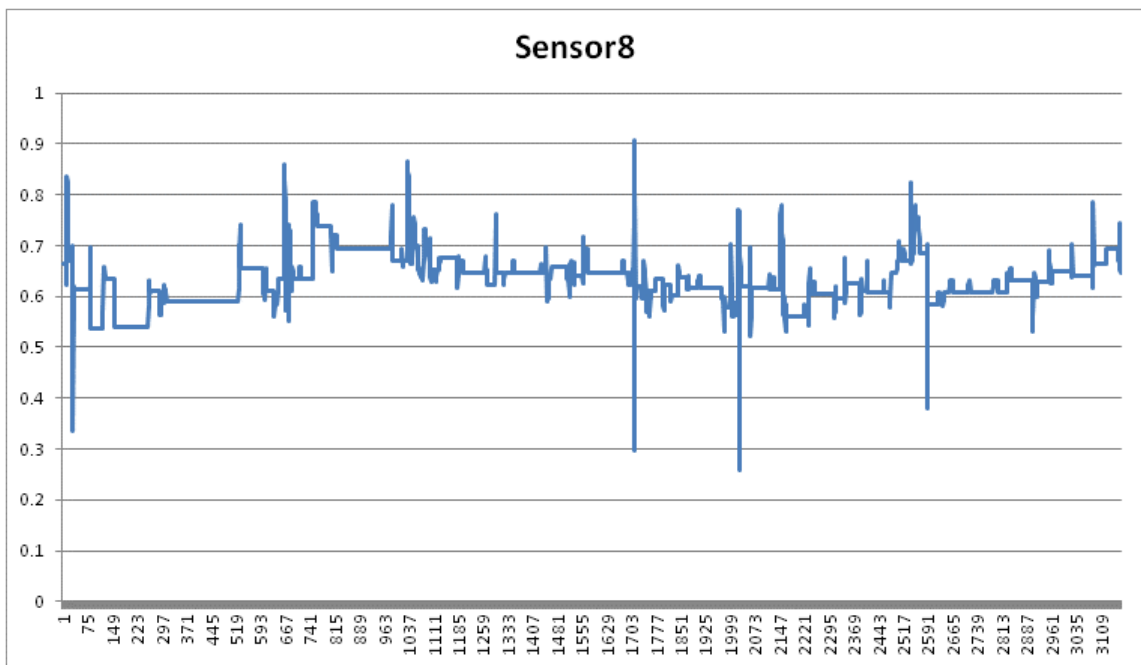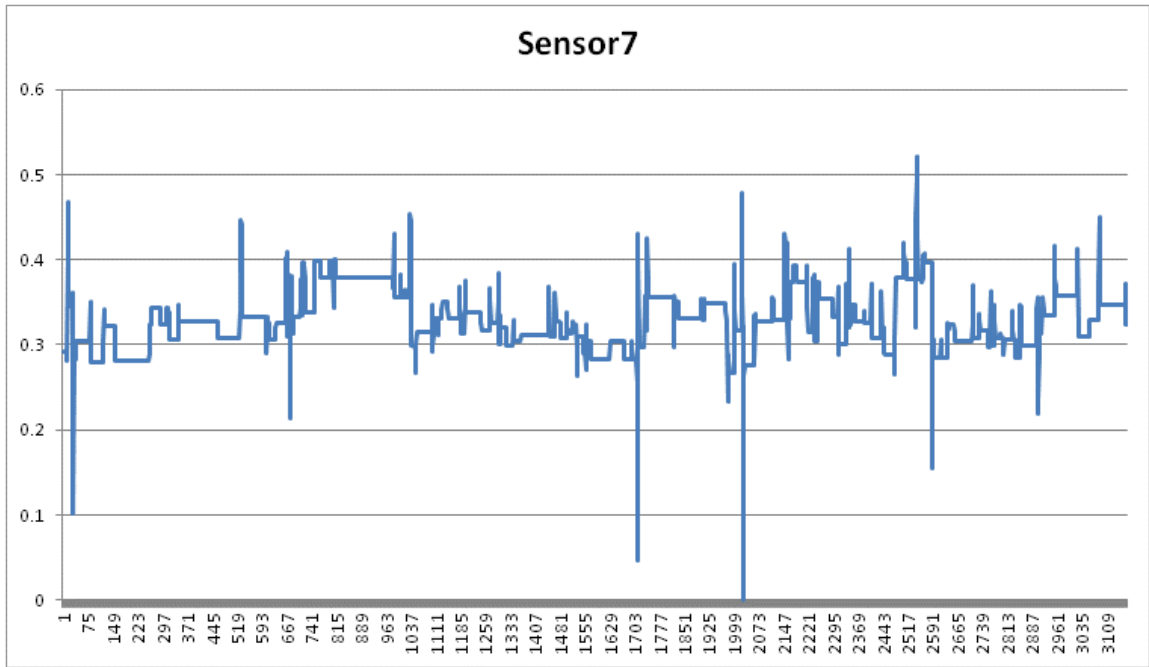
Appendix A-5   Eye Sensor Measurements for Participant 4102

The measurement of left eye pupil diameter is taken via the customized sensor for this experiment.   Pupil diameter has been shown to be correlated with memory (Kahneman and Beatty 1966) and other cognitive states (Marshall 2007).
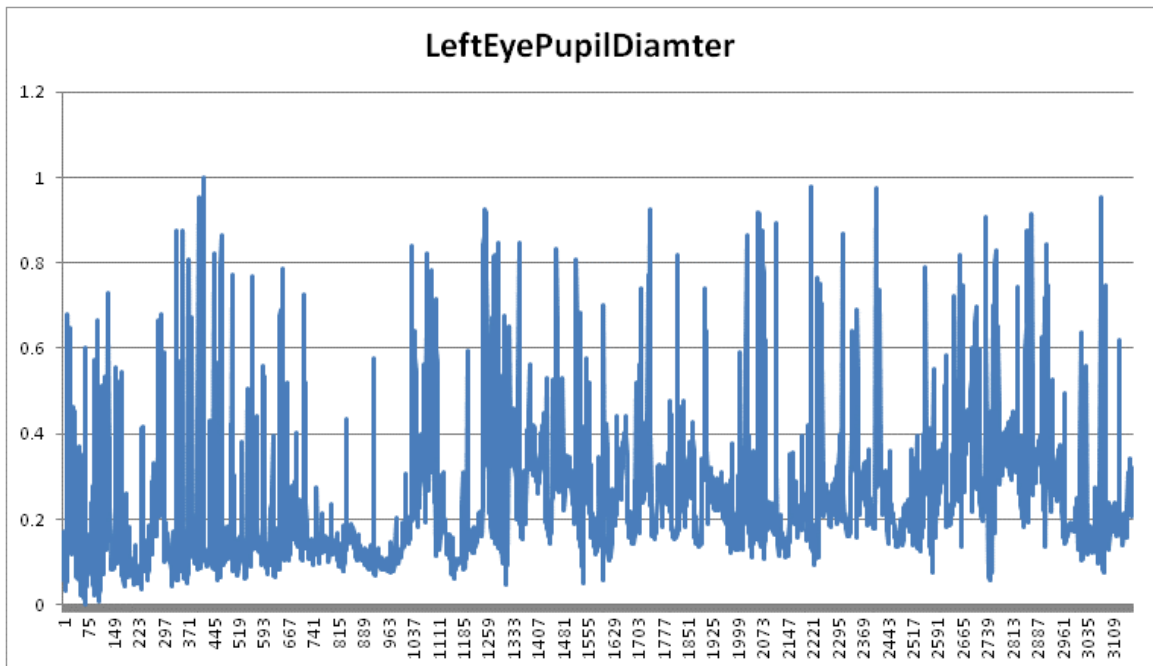
Appendix A-6   Derived Measurements for Participant 4102

The difference measures associated with these data are not known to be associated with any specific state. There are taken in order to ease the burden on the machine learning methods, and were used by the original experimenters.
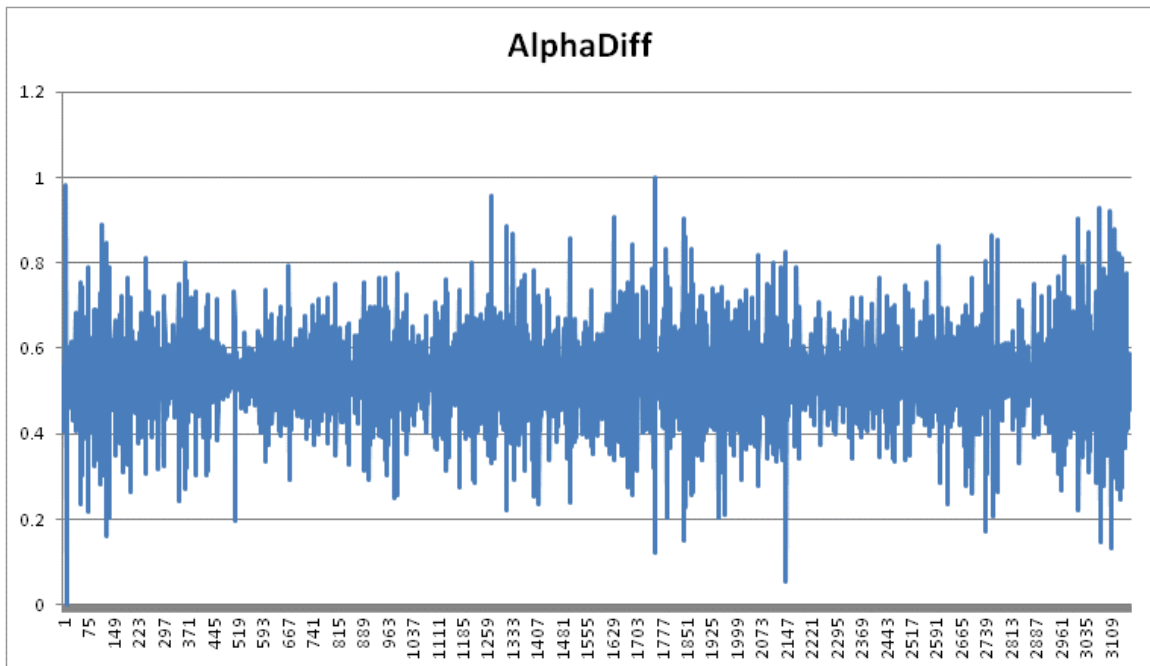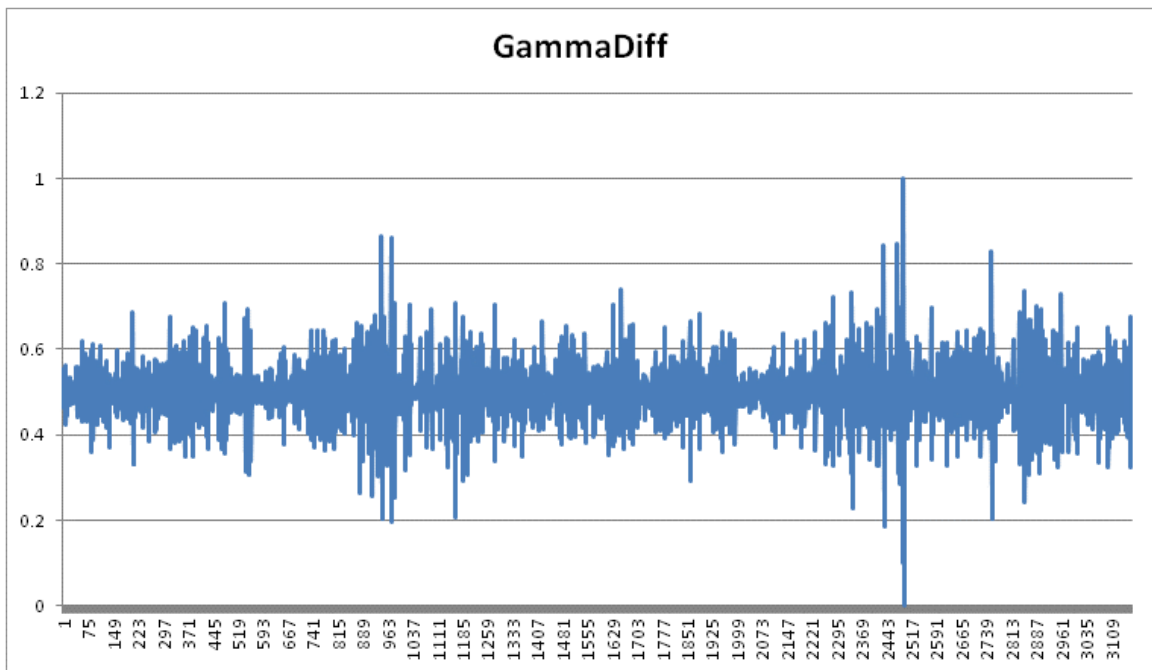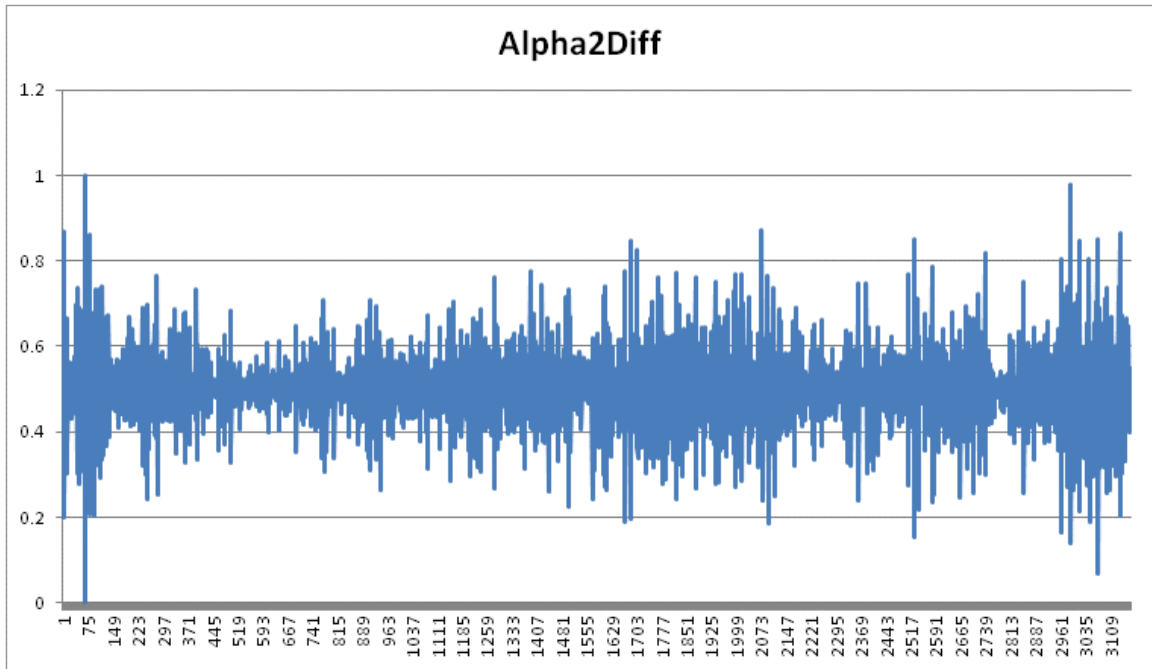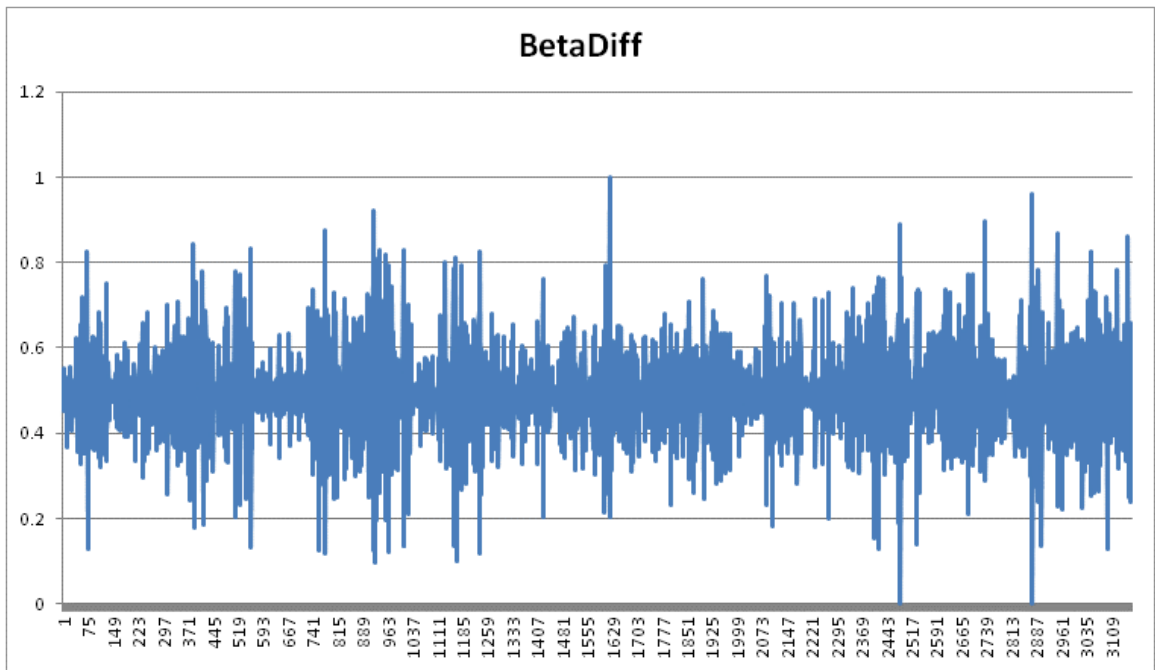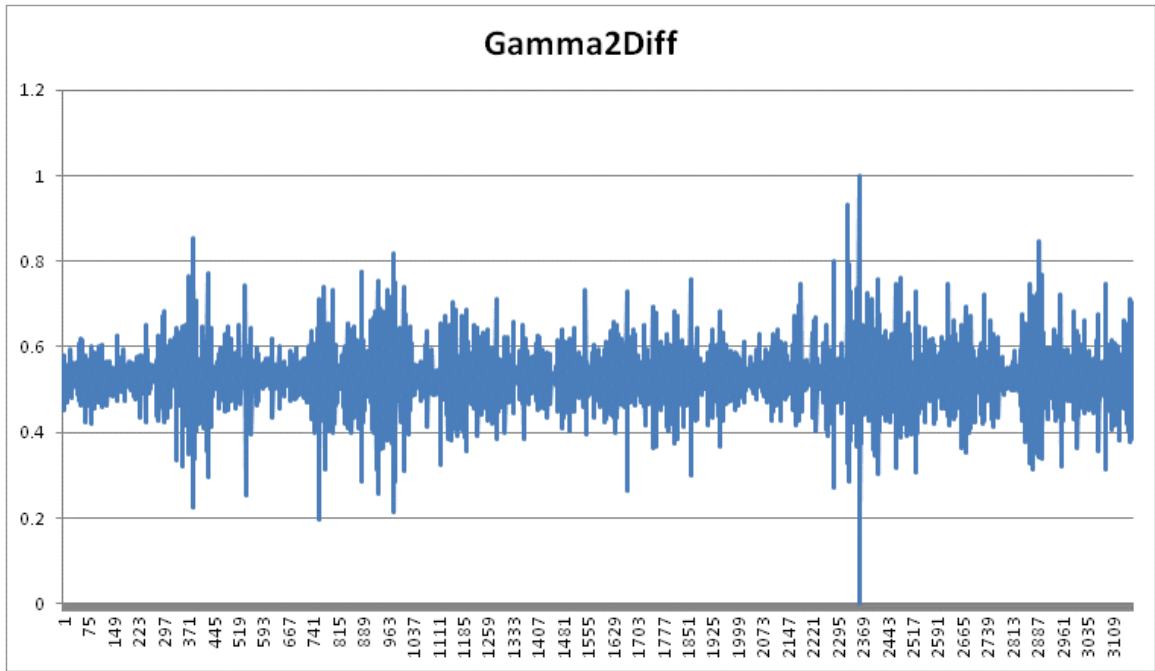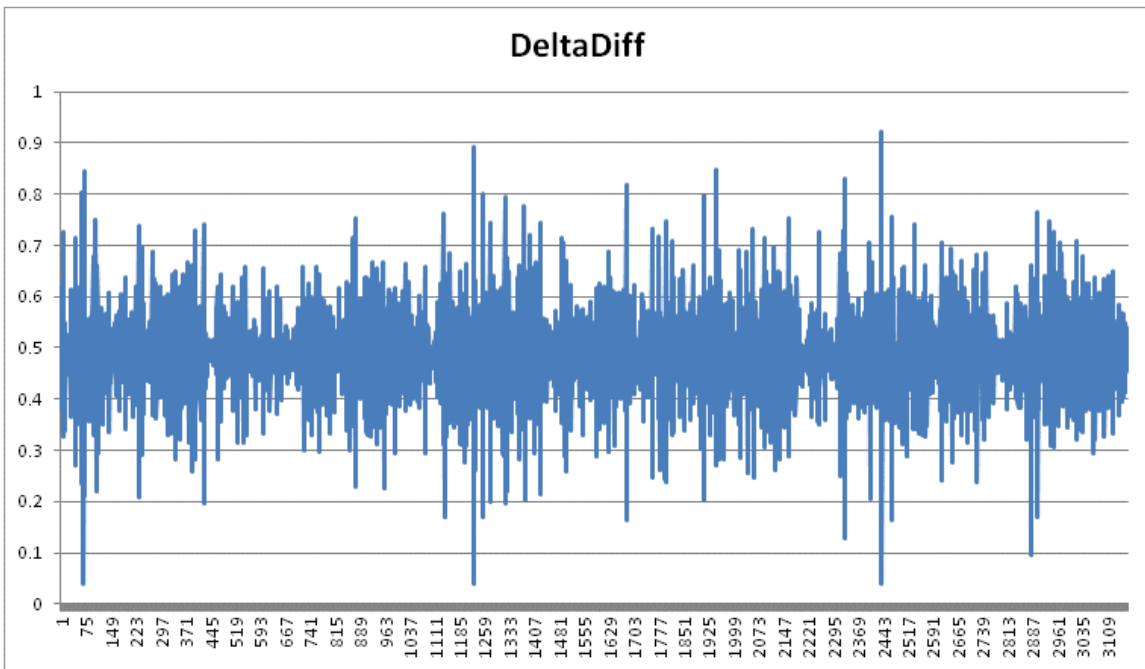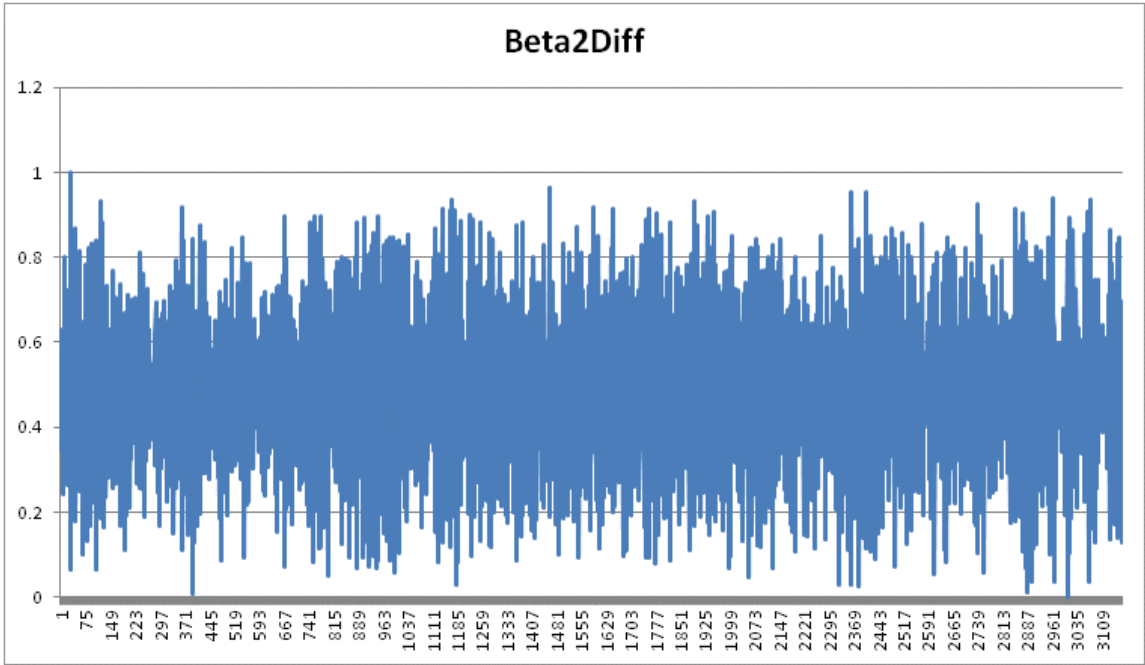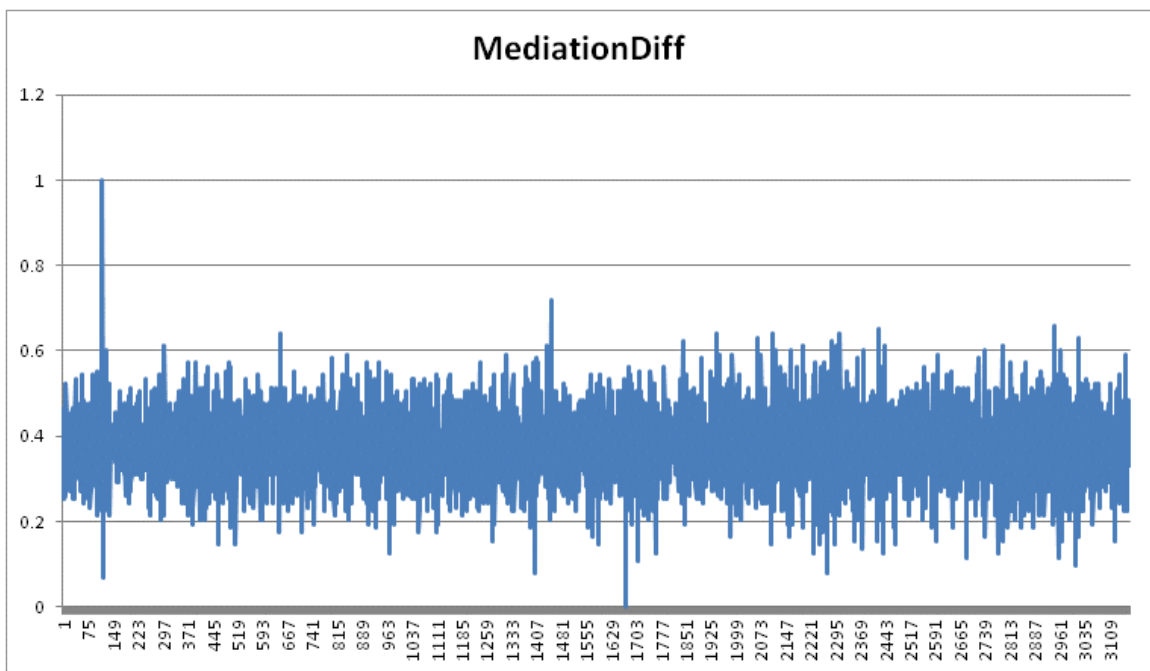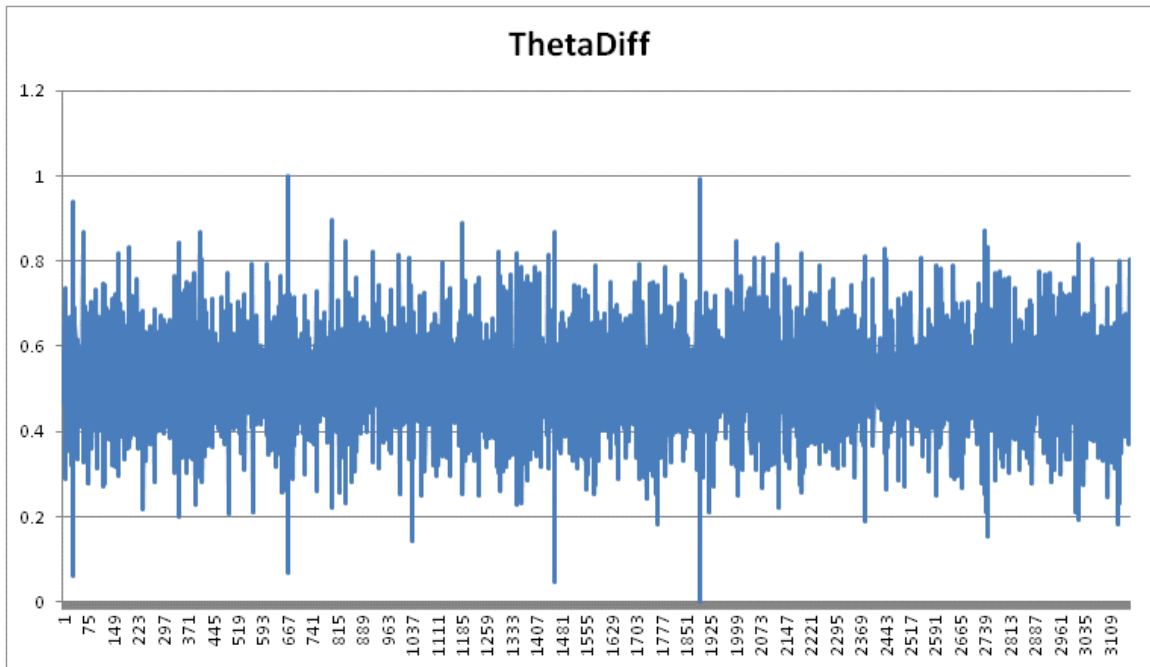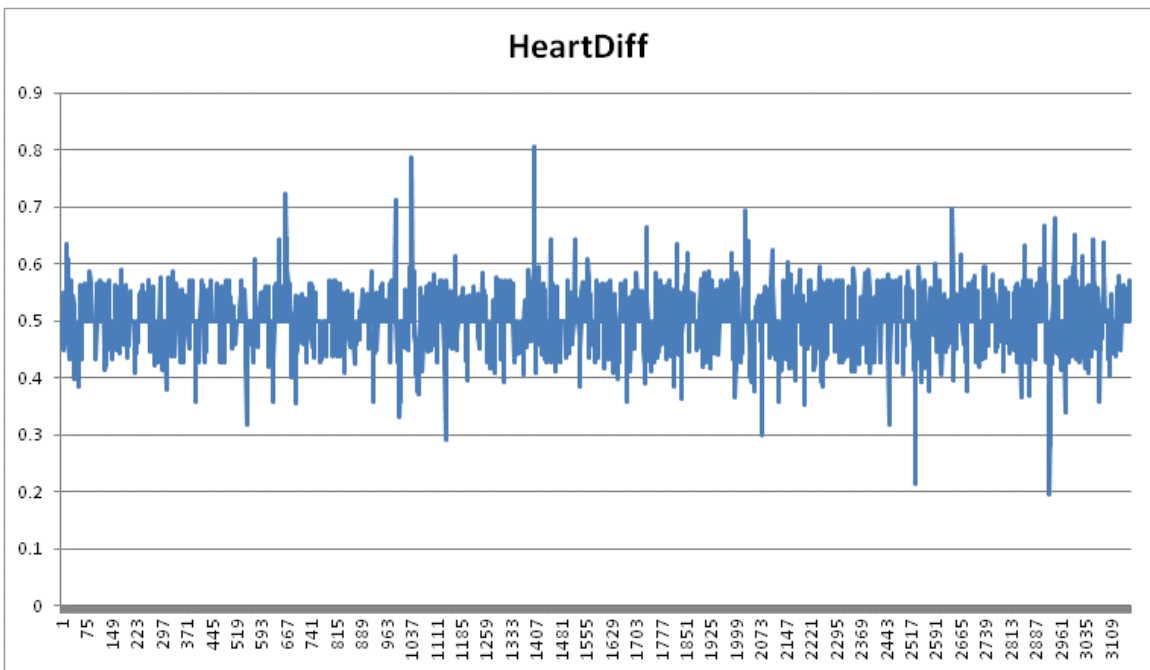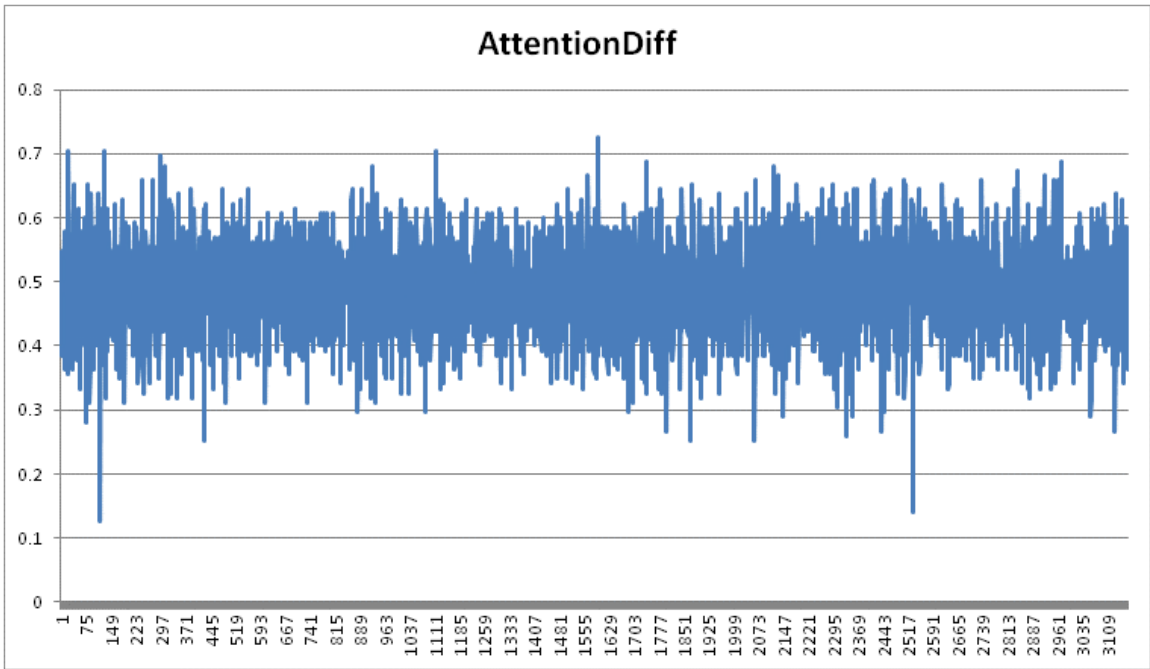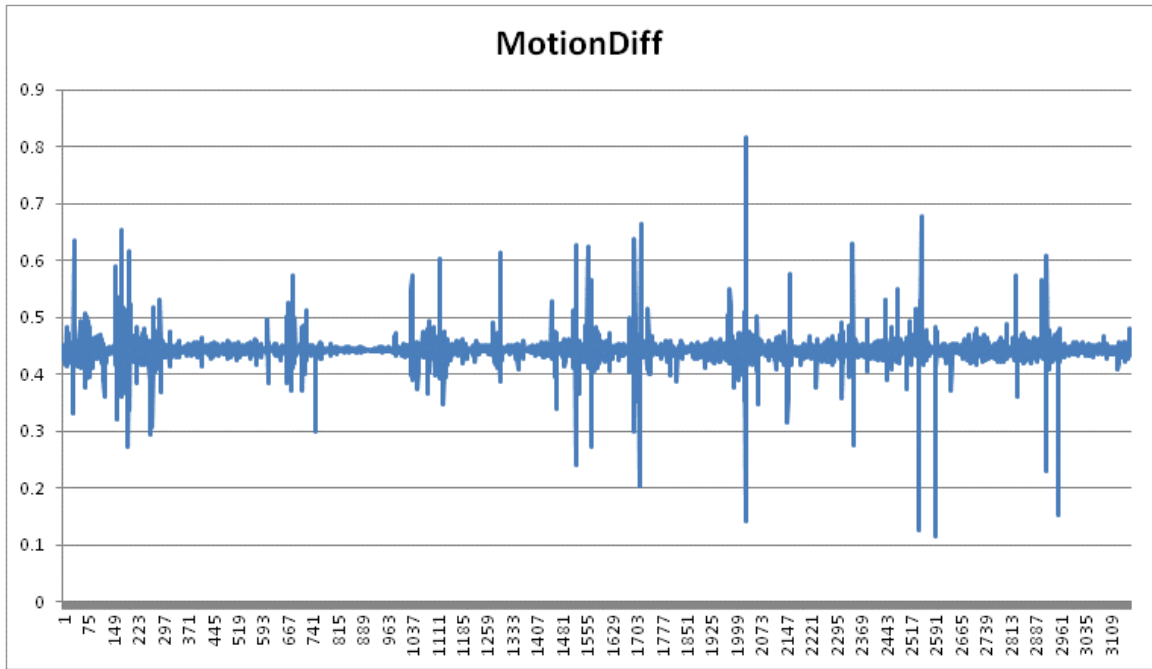
Alpha2Diff

GammaDiff

Gamma2Diff

BetaDiff

291

Beta2Diff

DeltaDiff

293

AttentionDiff

HeartDiff

MotionDiff

The ABM EEG headset produces three outputs measures: Engagement, Distractions, and Workload. These measures are derived from Power Spectral Density (PSD) absolute and relative signals in the 1-4 Hz, 5-7 Hz, 8-13 Hz, 14-24 Hz, and 25-40 Hz bands from eight key sites around the cranial area across a large population of individuals. The Workload metric is correlated with task load, memory, complex operations. The Engagement metric is correlated with drowsiness/alertness in driving tasks, attention to simulations, verbal processing in simple/complex environments, and verbal reasoning tasks. The Distraction metric is a measurement of whether the individual is "on task". See (Berka et al. 2007) for more information.
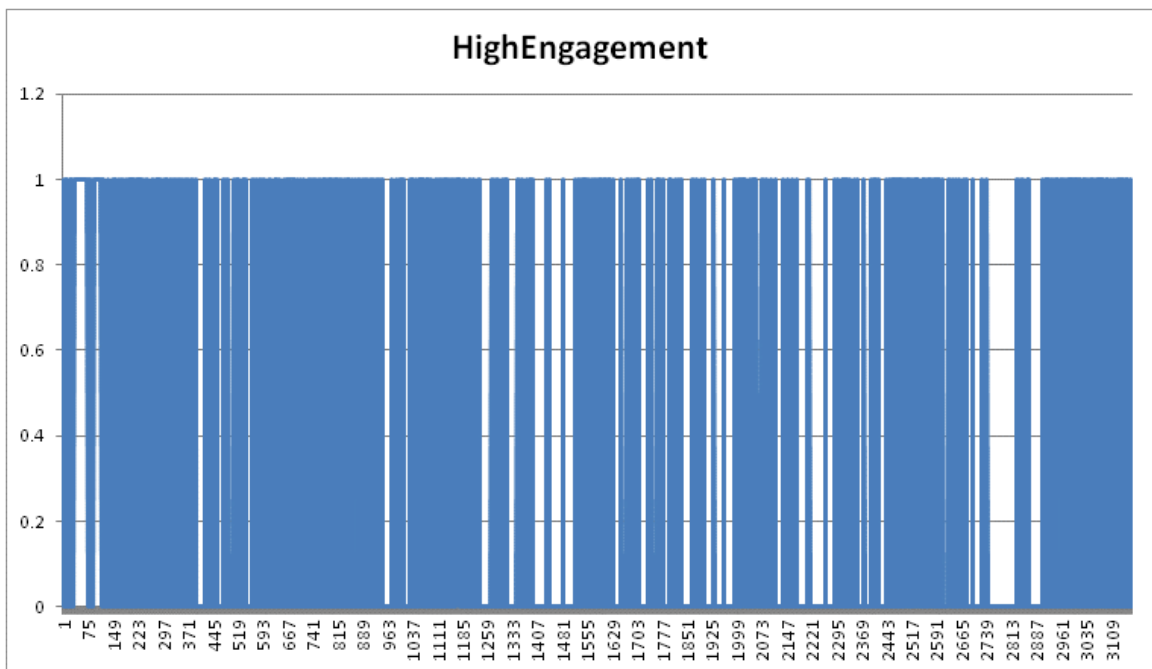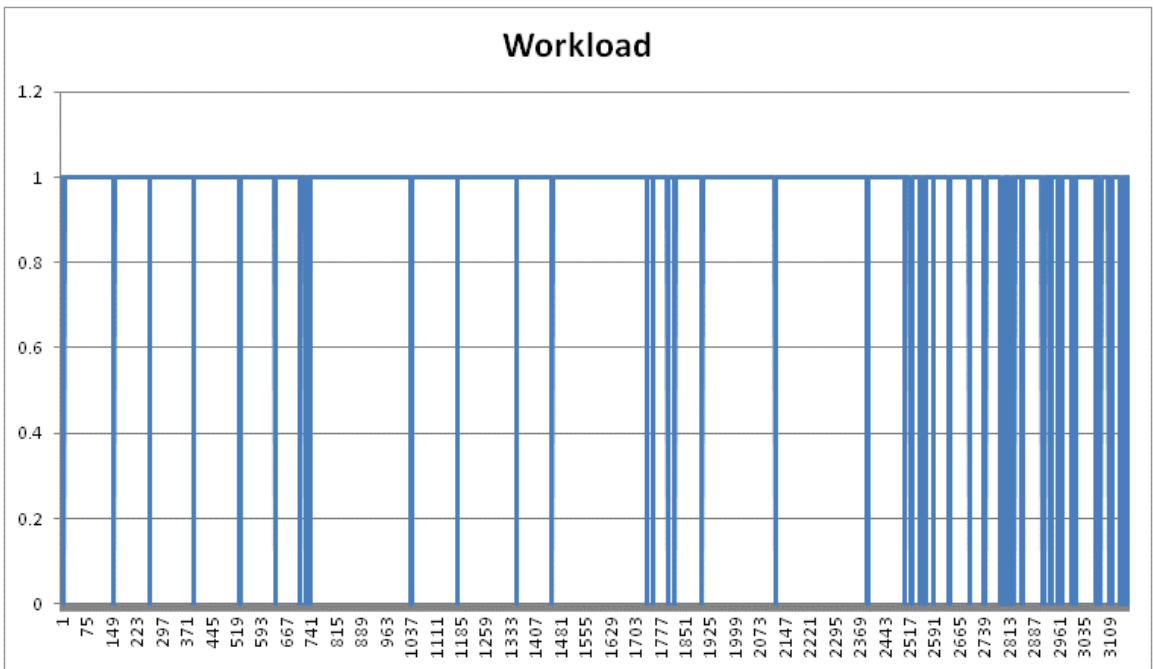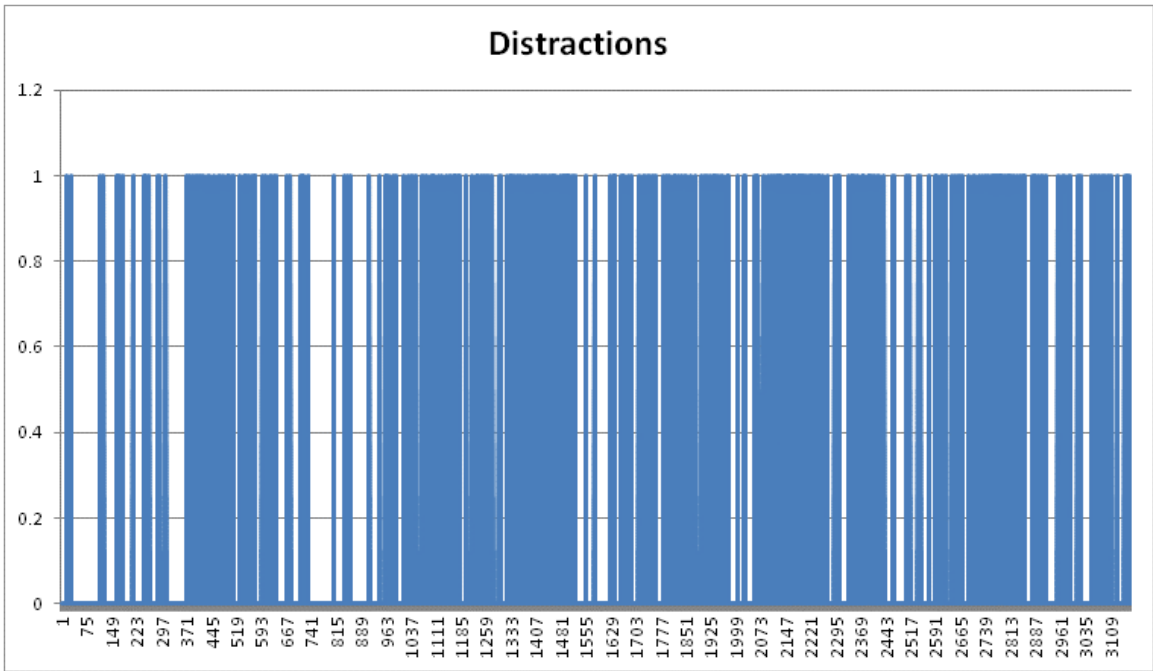
Distractions

Workload

The EmoPro® measurement tool produces three outputs measures used in this study: Anger, Boredom, and Fear. These measures are derived from direct user query with an emoticon-based interface.   While the three measures have not been validated, the measures have face validity, as the user selects the emoticon closest to the emotion that they are experiencing. It has been used in other recent studies (Jones et al. 2012; Kokini et al. 2012), and is commercially available. Its use is consistent with other user feedback reporting mechanisms.

Anger

Boredom

Fear
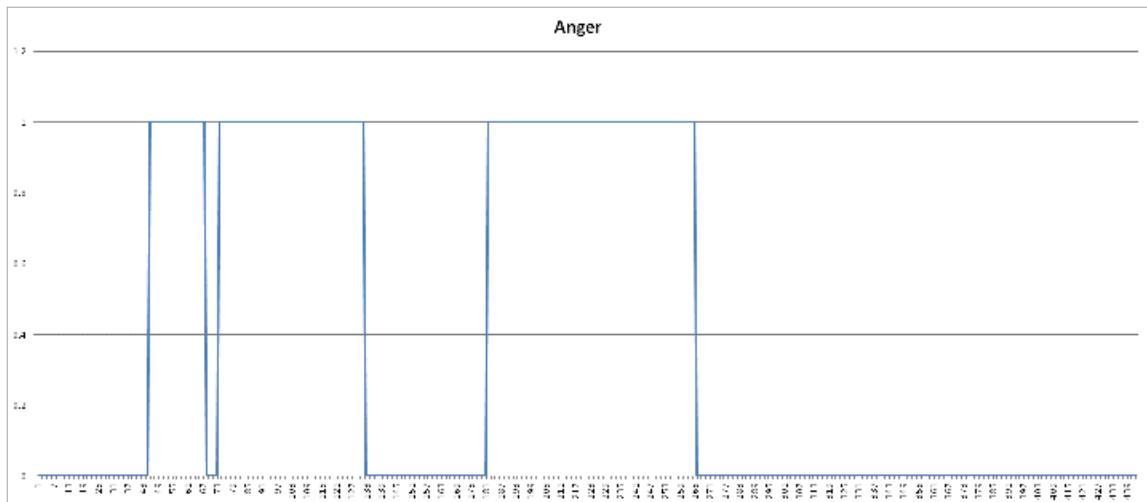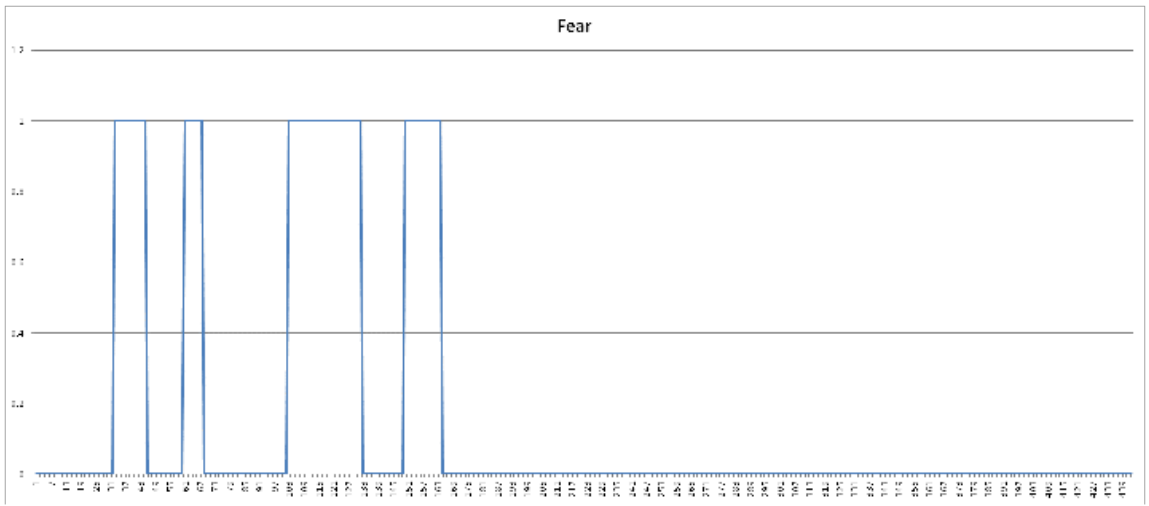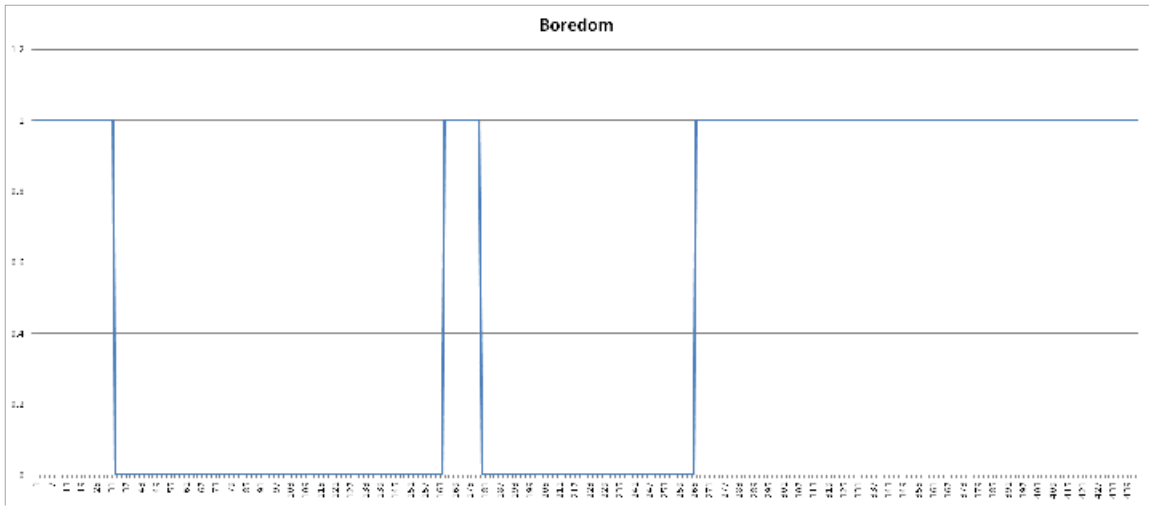
Table 65 – Example of a single data point from Dataset #1 (point 1, participant 1) shown

DateTime	14:56.0
Alpha	0.0192
Alpha2	0.01152
Gamma	0.02402
Gamma2	0.06282
Beta	0.90774
Beta2	0.0745
Delta	0.02695
Theta	0.03727
Meditation	0.50505
Attention	0.34343
Heart	0.69156
Motion	0.45775
Sensor1	0
Sensor2	0
Sensor3	0
Sensor4	0
Sensor5	0.75
Sensor6	0.52108
Sensor7	0.76461
Sensor8	0.79086
LeftEyePupilDiamter	0.55969
AlphaDiff	0.51934
Alpha2Diff	0.44642
GammaDiff	0.49582
Gamma2Diff	0.50694
BetaDiff	0.45627
Beta2Diff	0.69313
DeltaDiff	0.55585
ThetaDiff	0.19525
MediationDiff	0.43284
AttentionDiff	0.54878
HeartDiff	0.65885
MotionDiff	0.51036
ParticipantID	4101
HighEngagement	1
Distractions	0

WorkloadFBDS	0
Anger	0
Boredom	0
Fear	0

APPENDIX B   MEASUREMENTS FOR DATASET #2

The Fixations measurement corresponds to how long the participant remained starting at a point on the screen. The longer a participant remained staring, the higher the reported fixation. If the participant was not staring at a point on the screen, this measurement reported '0'. The high variability of fixations data corresponds to the participant looking around and focusing on different items during the conduct of the experiment. This data was normalized prior to running machine learning experiments. This is shown via the large jump in total number of usable models in the final section of this Appendix.
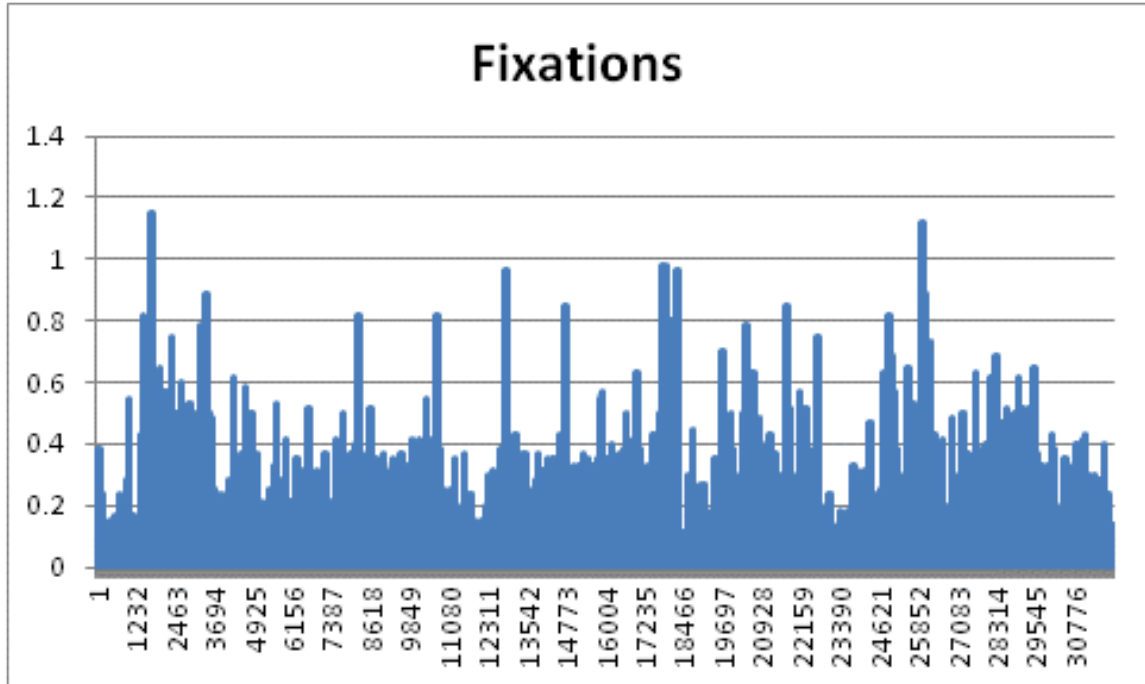
Pupil Diameter refers to the number of millimeters the radius of the pupil of the left eye.

For a more full description, see Appendix A-5.

Pupil Diameter
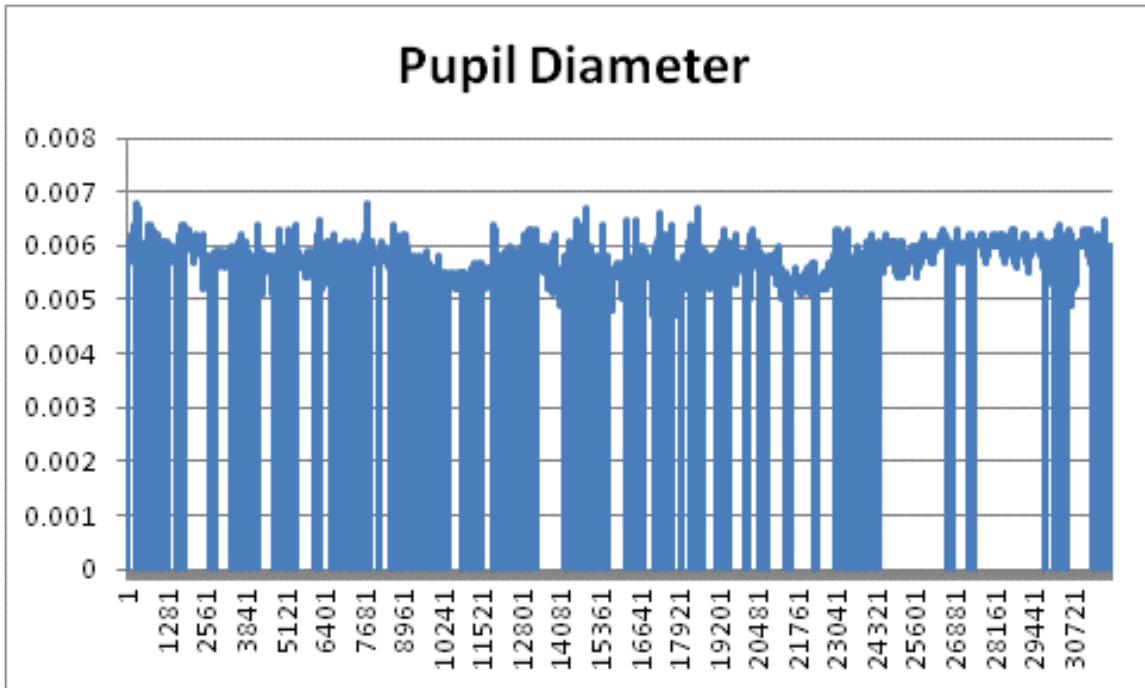
Graphs of Labeled Measurements from the Facelab System (5%

Of Total Data)

The Index of Cognitive Activity is a measure of cognitive workload produced from eyetracking data (Marshall 2002).   It has been validated in high and low light environments, and against EEG measures of workload. It remains to be commonly used in the psychology domain for identification of workload (Demberg et al. 2013; Marshall 2007).
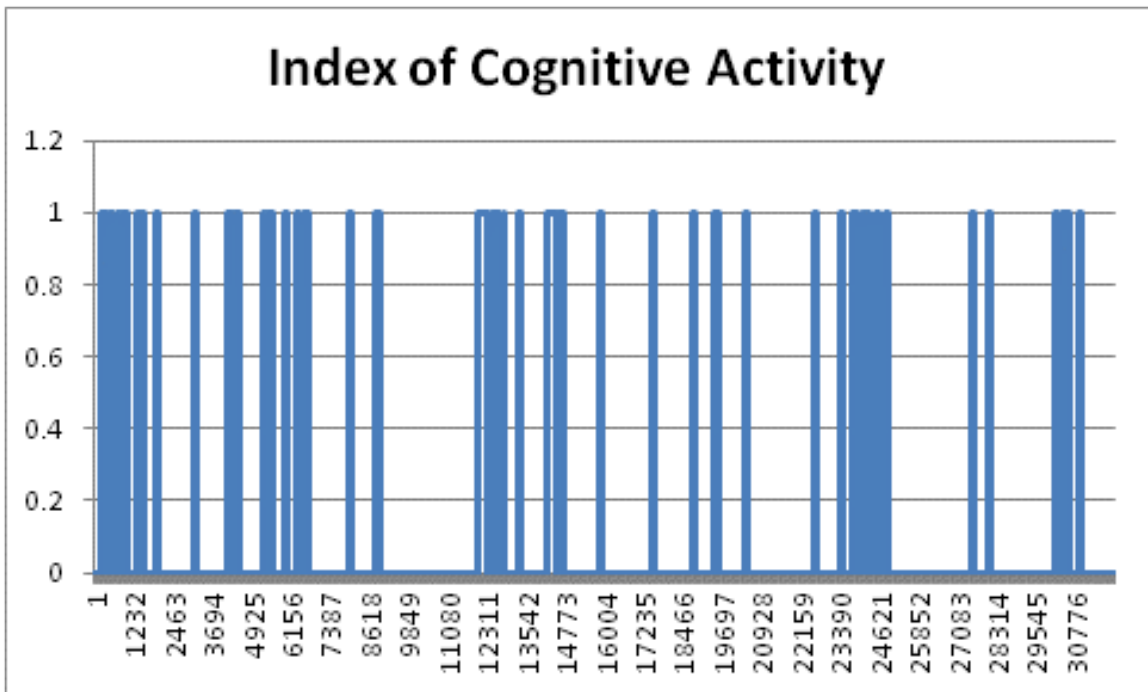
Table 66 shows the effect that downsampling, as discussed in Section 6.3.3, has on overall data collection. The read will note that there is still a large amount of repeated data, and that no significant information was destroyed in the process. Data was downsampled from 14000 Hz to 3500 Hz.

Table 66 – Downsampled Dataset #2, 3500 Hz, few changes observed.

Time	ParticipantID	Fixations	Pupil Diameter	Index Of Cognitive Activity
11.20.41.22.075	32	0	0.0051	0
11.20.41.22.075	32	0	0.0051	0
11.20.41.22.075	32	0	0.0051	0
11.20.41.22.075	32	0	0.0051	0
11.20.41.22.075	32	0	0.0051	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.005	0
11.20.41.22.075	32	0	0.0051	0
11.20.41.22.075	32	0	0.0051	0

APPENDIX C   COMPLETE RESULTS OF ALL ALGORITHMS ON

ALL DATASETS

A dissertation should present all complete results sets. This dissertation presents the results from several batches of model creation. The first set of results, hereafter referred to as Results Set #1, is created from default algorithmic parameter settings on the total set of cognitive and affective data from Dataset #1. The second set of results, hereafter referred to as Results Set #2, uses Dataset #1 and Dataset #2 with altered parameter settings believed to produce models with more accuracy, as earlier in this dissertation. The third set of results, hereafter referred to as Results Set #3, is created through the use of an abbreviated set of Dataset #1 models, using only the input features which have already been determined to be useful in the previous studies.

Each algorithm has three sets of graph per set of model which corresponds to one for each scheme of labeling (unsupervised, supervised, or semi-supervised). Each model is divided into type. The first type is the Dataset #1 Cognitive models of distraction, engagement, and workload. The second type is the Dataset #1 affective models of anger, boredom, and fear. The third type is the Dataset #2 cognitive models of the Index of Cognitive Activity. In order to facilitate a more in-depth discussion of the impact of semi-supervision on overall algorithm performance in Chapter 0, the semi-supervision of all algorithms and all models are graphed together. A brief summary of the presented graphs is shown in Table 67.

Table 67 – Preview of upcoming results graphs

Method	Supervision	Type of Model	Graphed Performance Data
ART	Unsupervised	Cognitive	Distraction, Engagement, Workload
ART	Supervised	Cognitive	Distraction, Engagement, Workload
ART	Semi-supervised	Cognitive	Distraction, Engagement, Workload
ART	Unsupervised	Affective	Anger, Boredom, Fear
ART	Supervised	Affective	Anger, Boredom, Fear
ART	Semi-supervised	Affective	Anger, Boredom, Fear
…	…	…	…
Other methods	Un-/semi-/fully-supervised	Both	All
…	…	…	…
All	Semi-Supervised	Cognitive	Distraction, Engagement, Workload
All	Semi-Supervised	Affective	Anger, Boredom, Fear

The primary item of interest to realtime model creation is the goodness of fit of the model over time. The x-axis of each graph presented in the results section is time, with each

line corresponding to a measured evaluation. All evaluations are measured with the AUC ROC metric. Three types of AUC ROC measures are taken: "all", "next", and "prev". The "all" ROC measure represents the ability of the model to correctly predict all of the data that has so far been presented. The "next" and "prev" measures represent the ability of the model to correctly predict the unseen next 10% of total data and the recently presented previous 10% of total data, respectively.
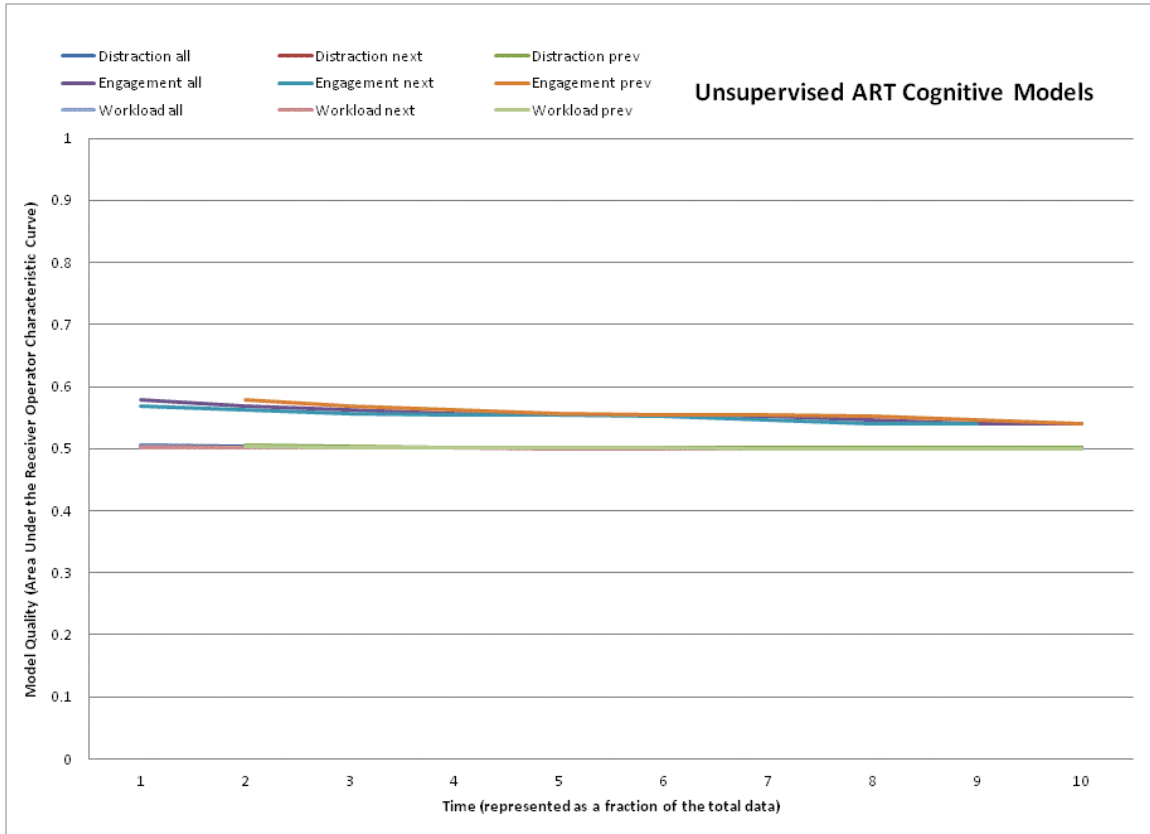
Figure 47 – Performance of unsupervised ART for cognitive modeling
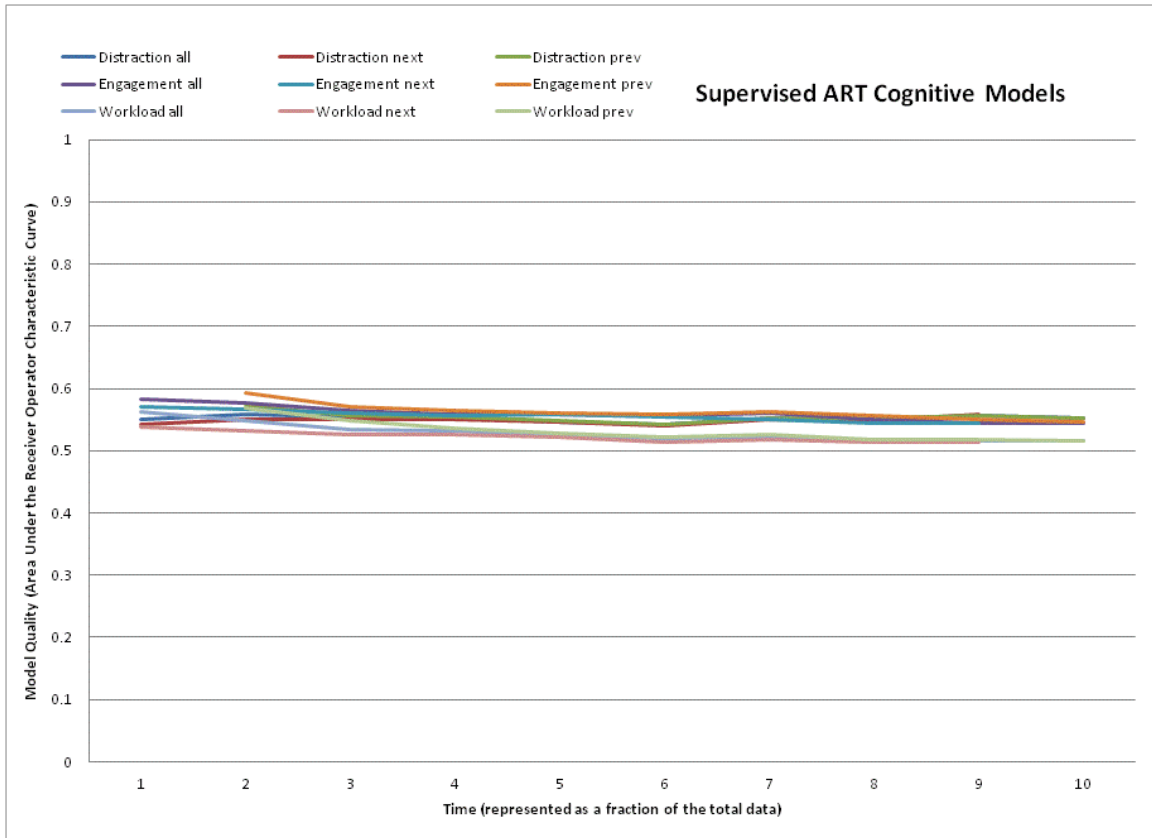
Figure 48 – Performance of supervised ART for cognitive modeling
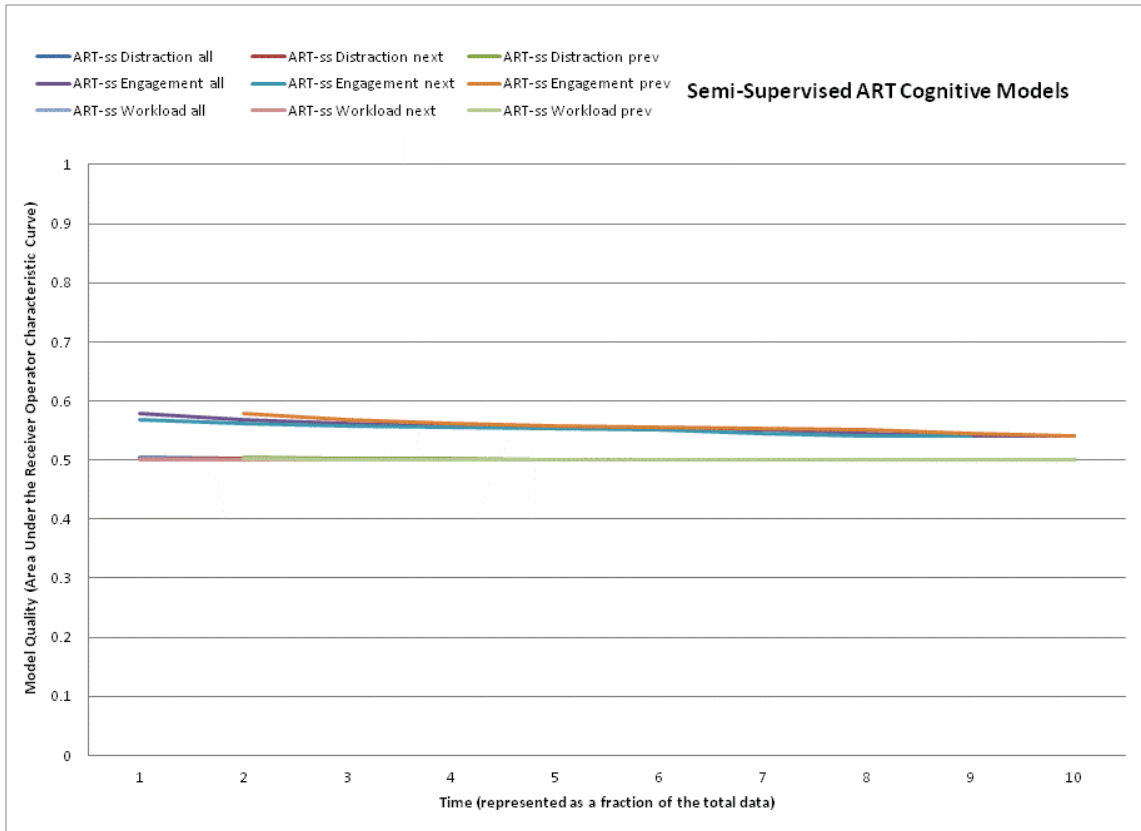
Figure 49 – Performance of semi-supervised ART for cognitive modeling
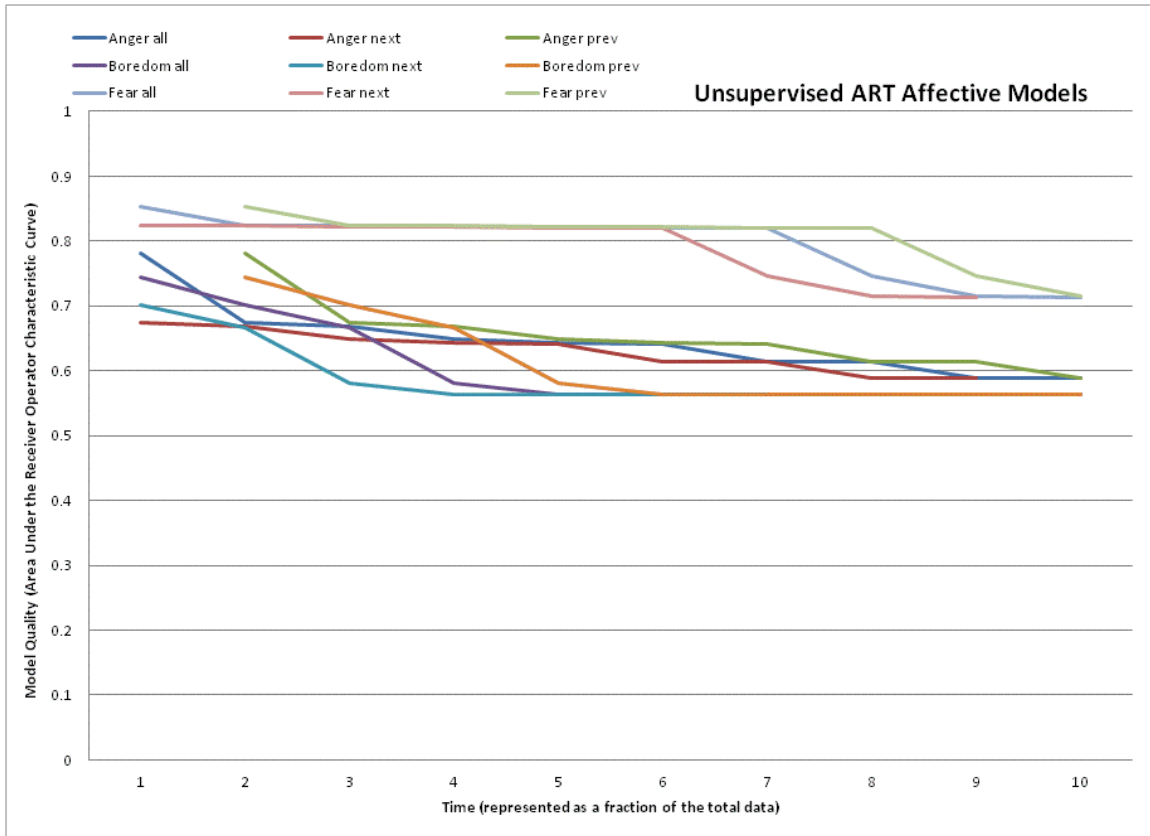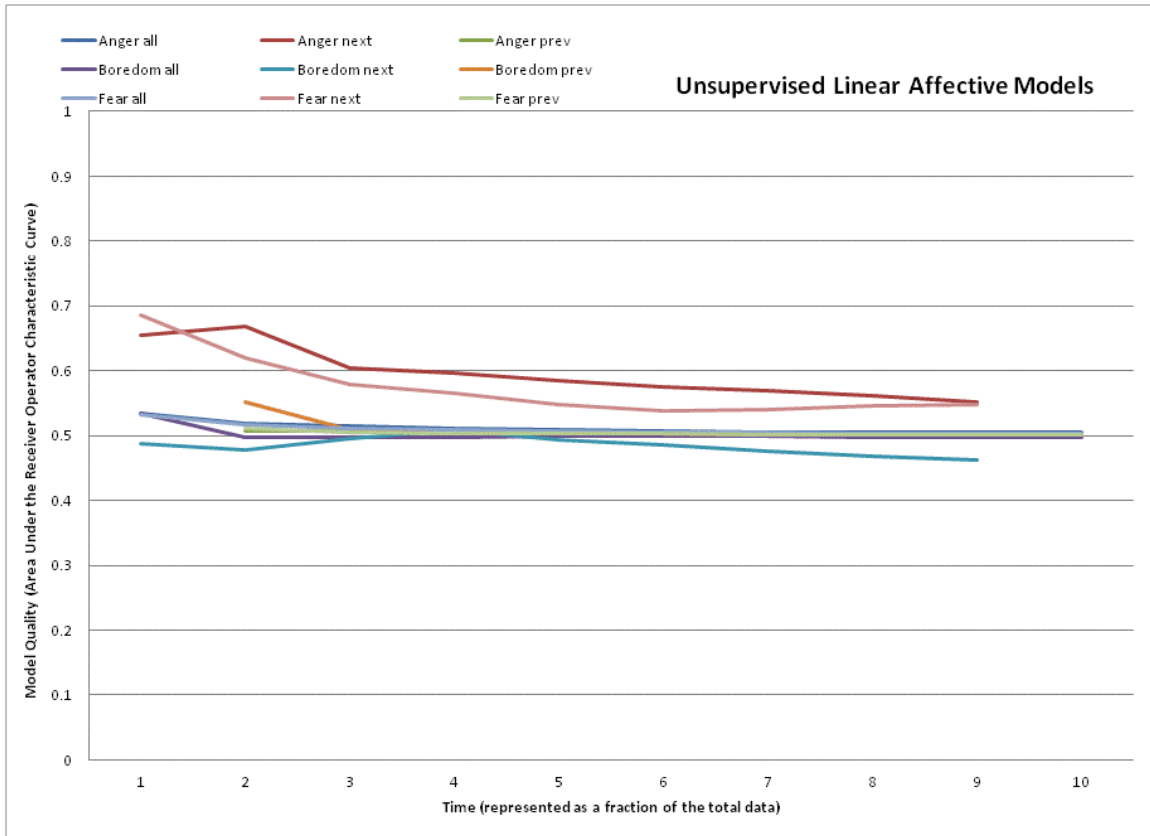
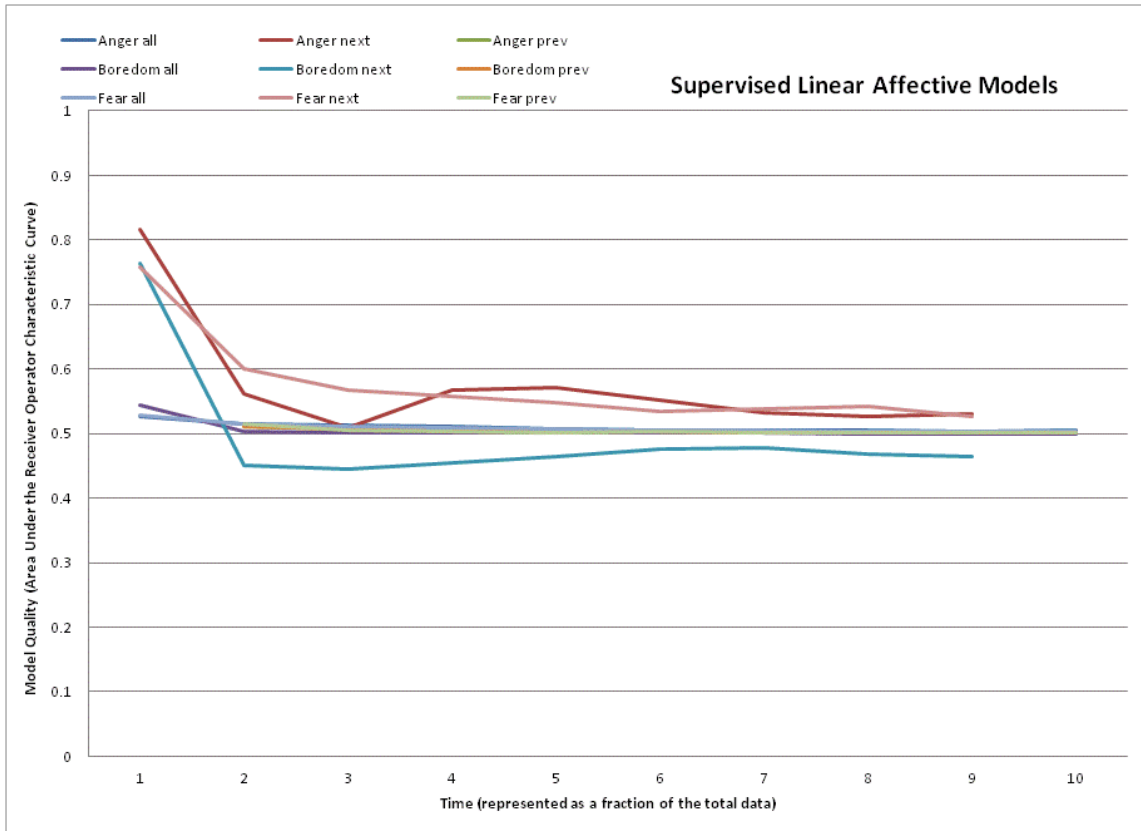**Figure 50 – Performance of unsupervised ART for affective modeling**

314
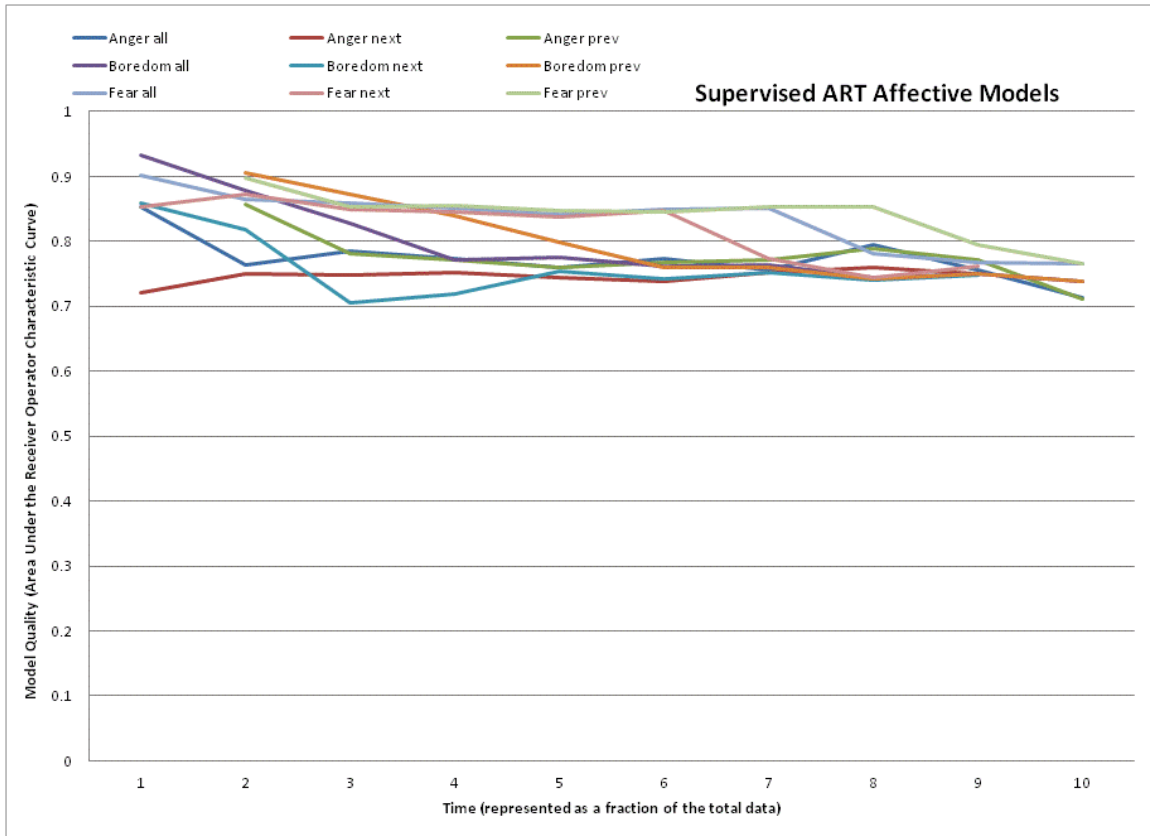
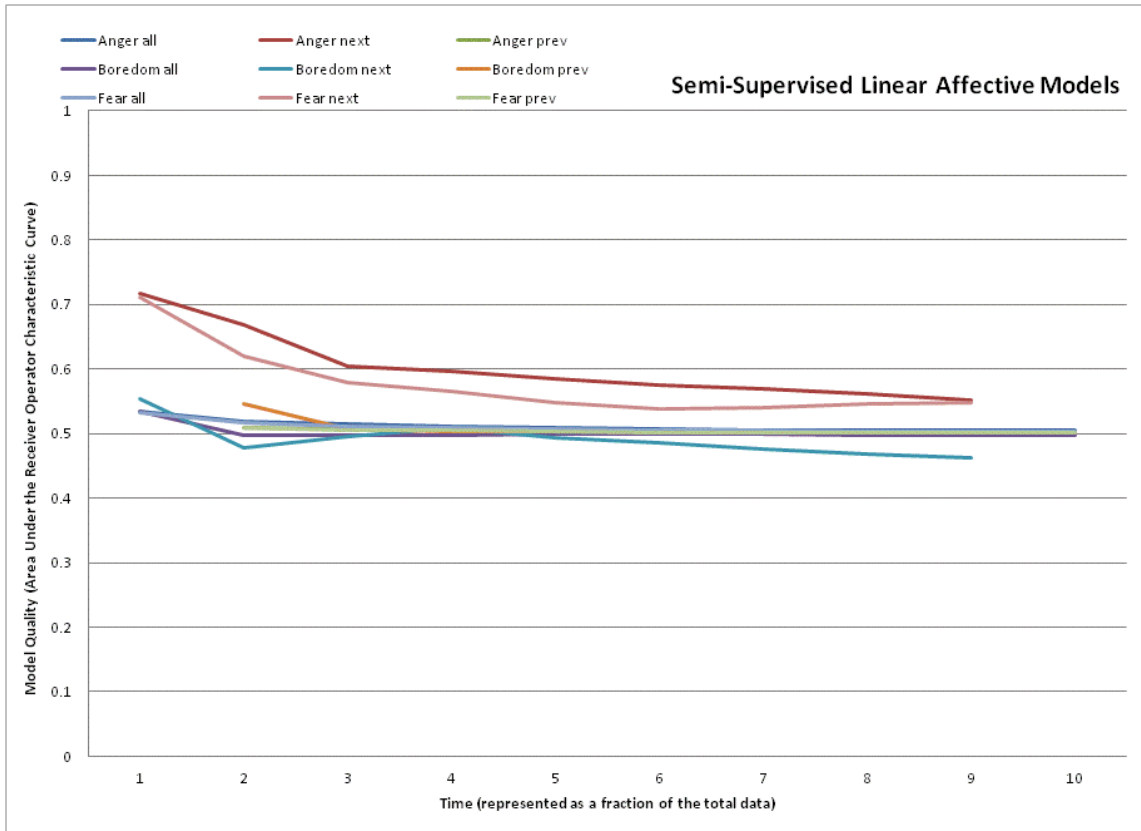**Figure 51 – Performance of supervised ART for affective modeling**

315
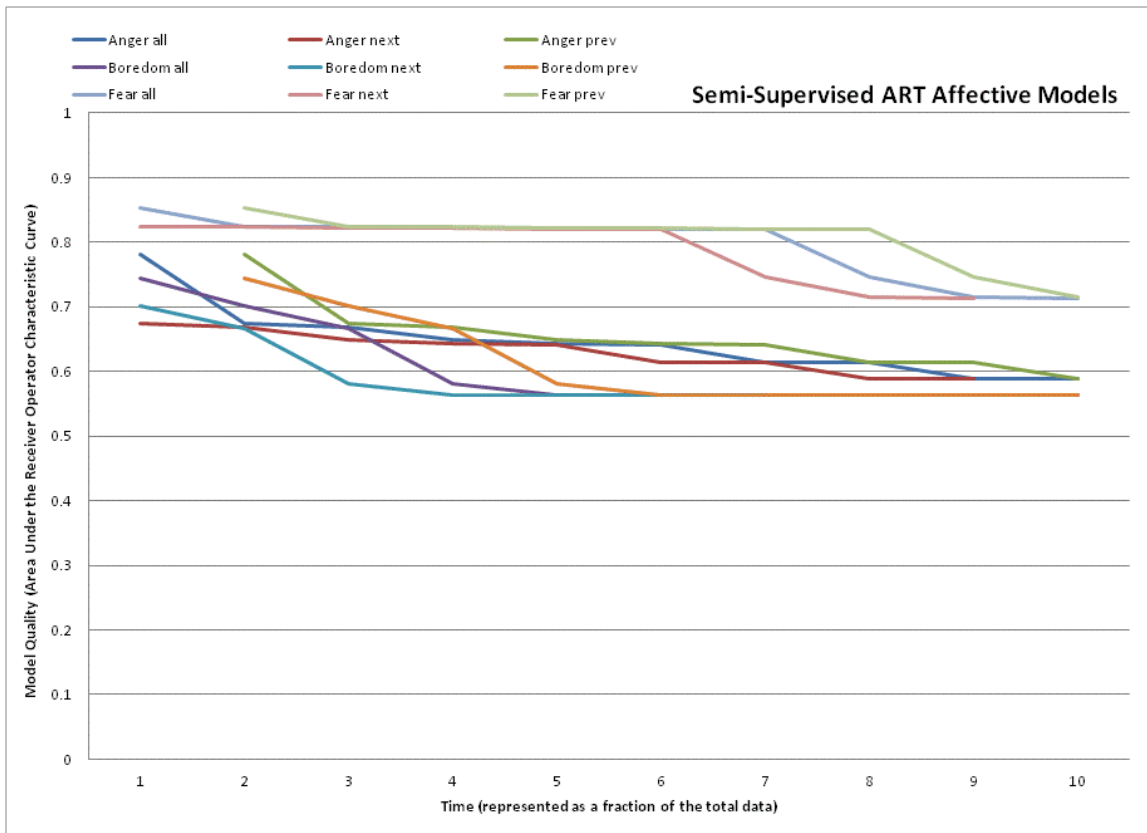
Figure 52 – Performance of semi-supervised ART for affective modeling
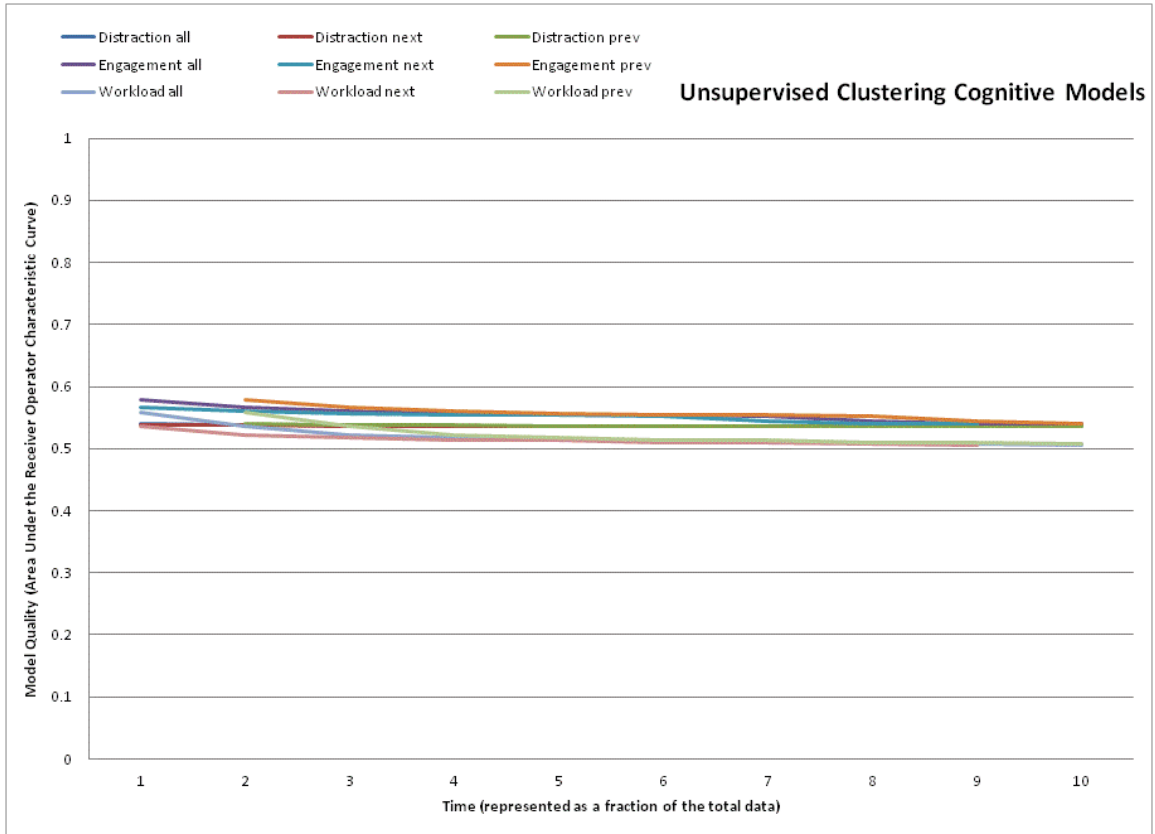
Figure 53 – Performance of unsupervised K-Means clustering for cognitive modeling
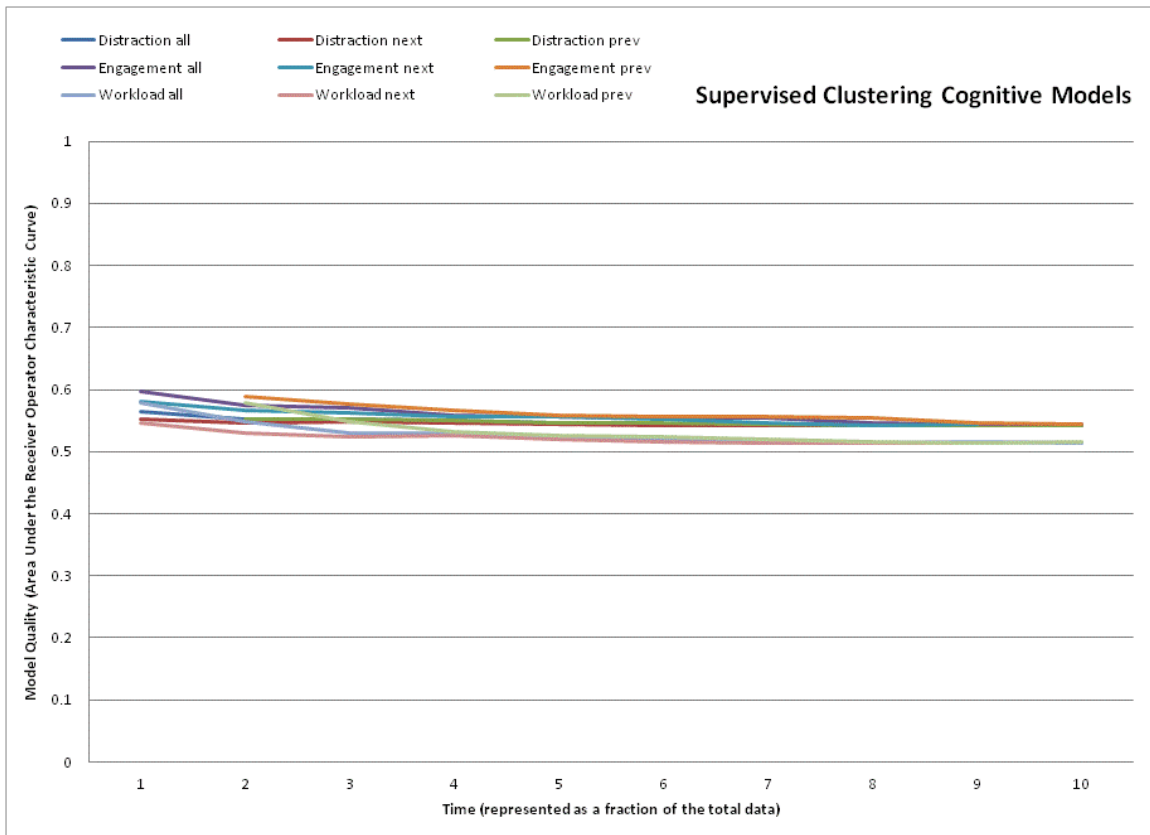
Figure 54 – Performance of supervised K-Means clustering for cognitive modeling

318
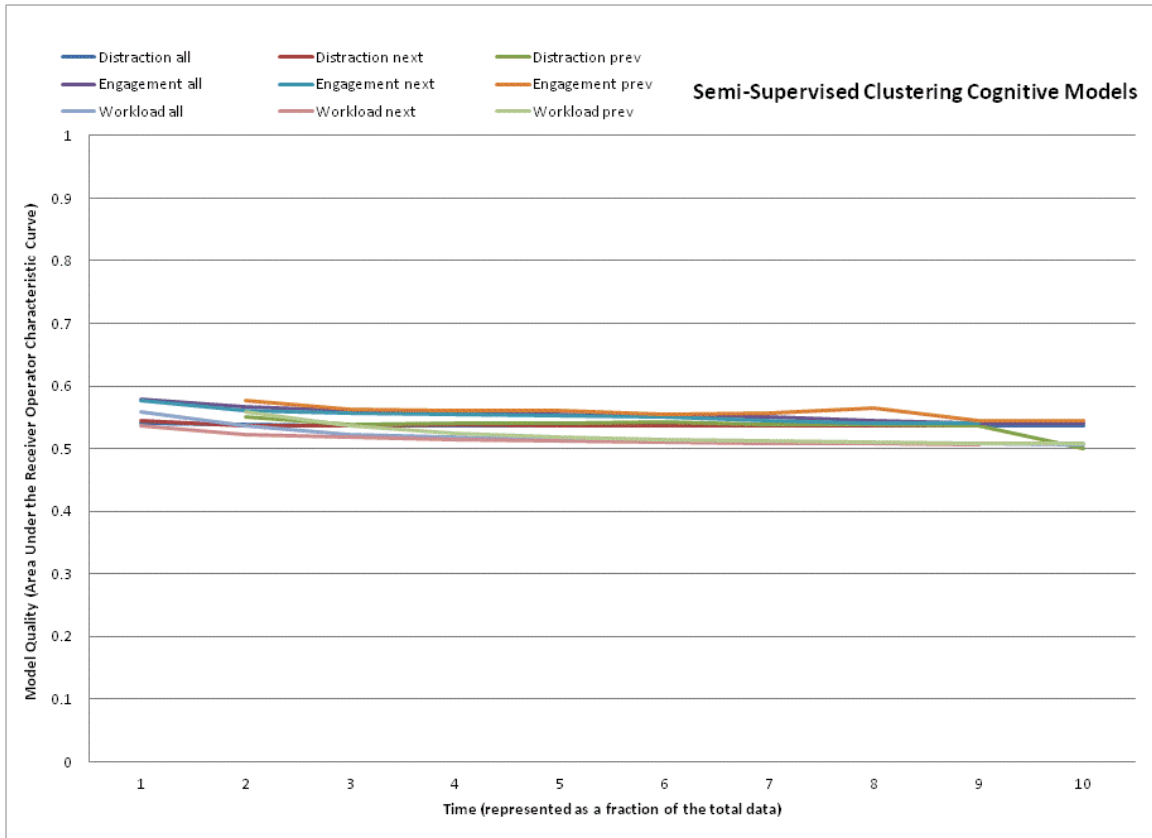
Figure 55 – Performance of semi-supervised K-Means clustering for cognitive modeling
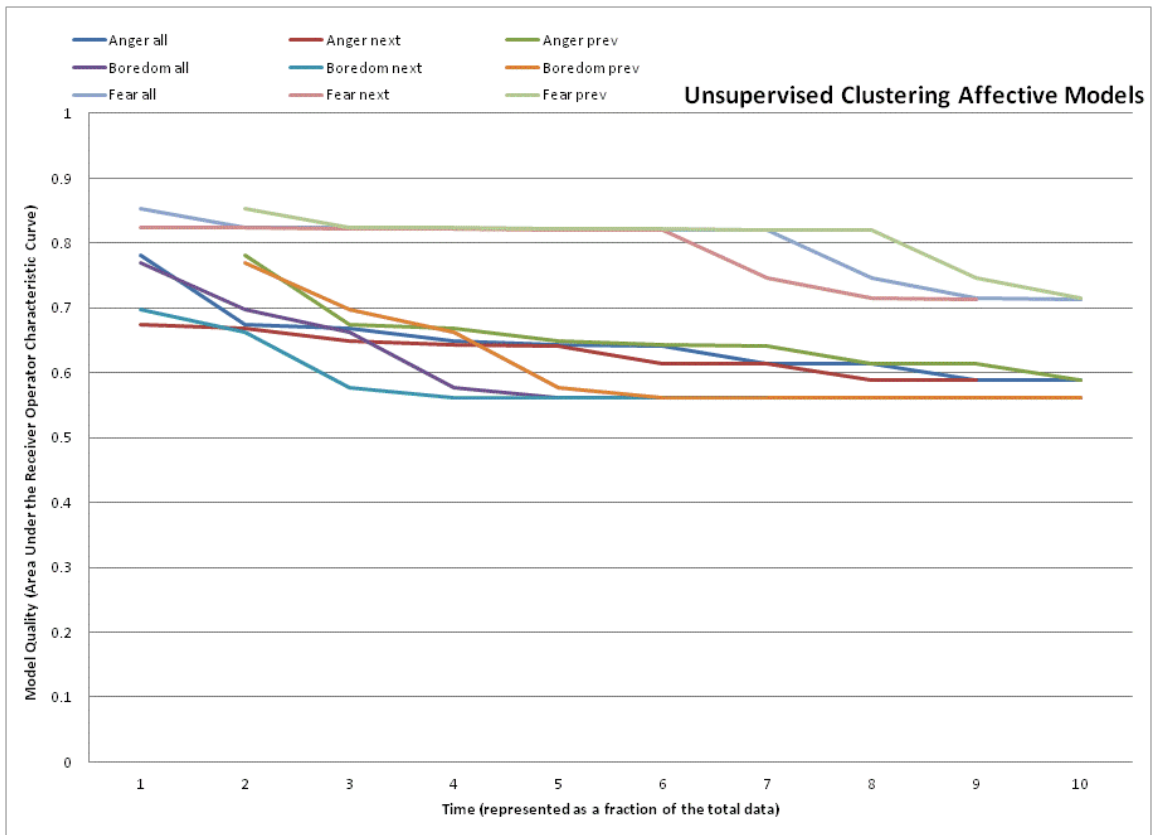
Figure 56 – Performance of unsupervised K-Means clustering for affective modeling
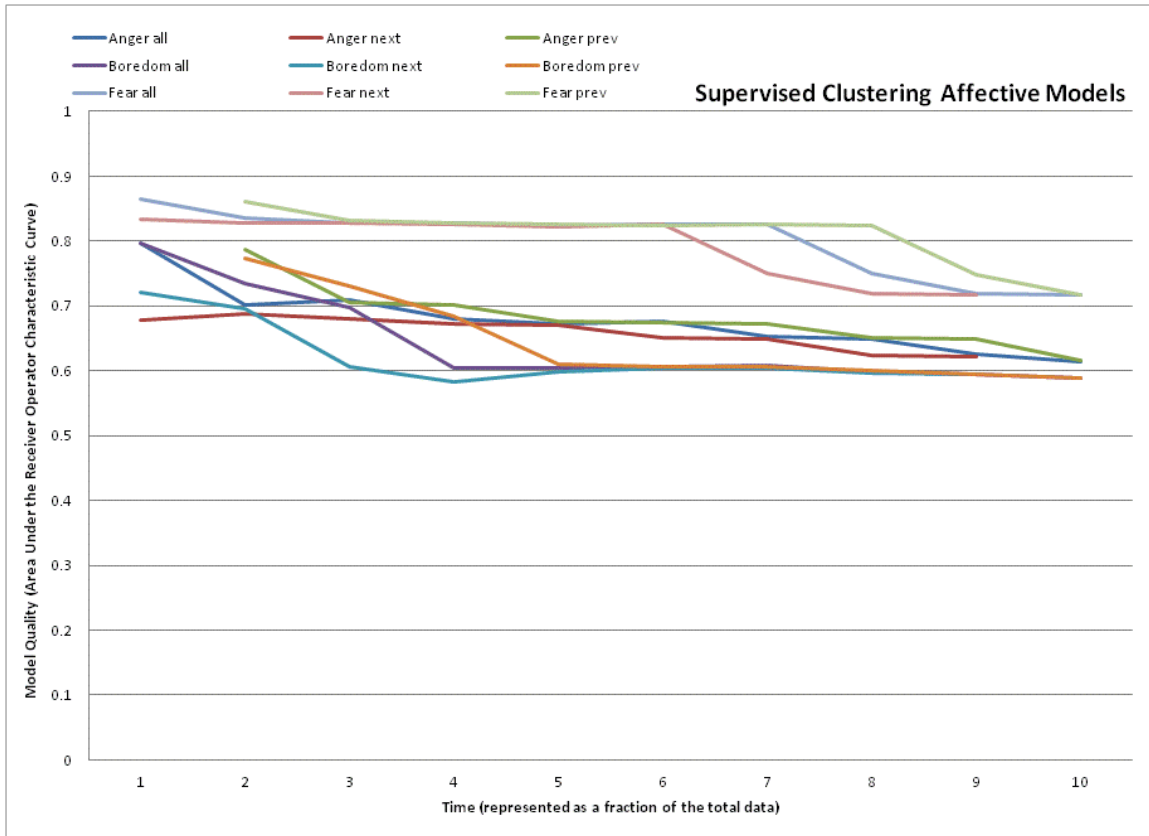
Figure 57 – Performance of supervised K-Means clustering for affective modeling
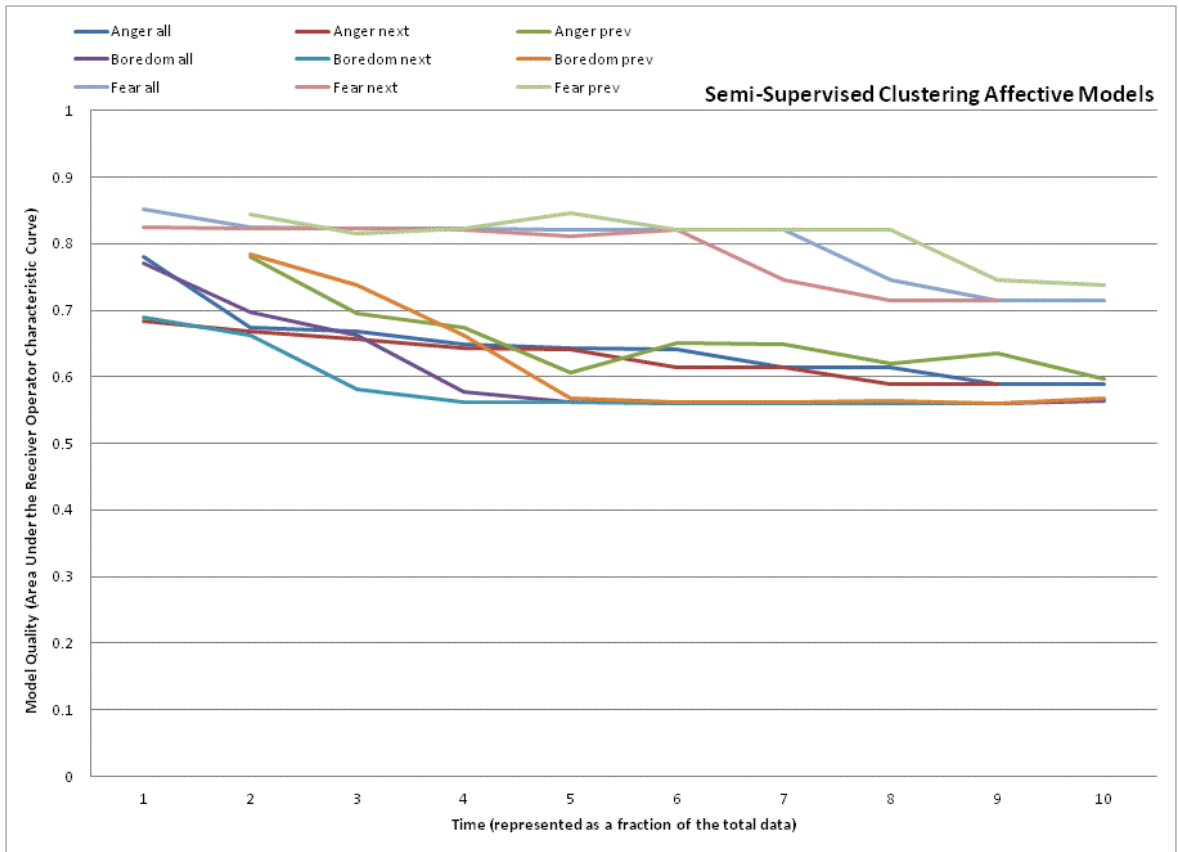
Figure 58 – Performance of semi-supervised K-Means clustering for affective modeling

Appendix C-1-3   Growing Neural Gas
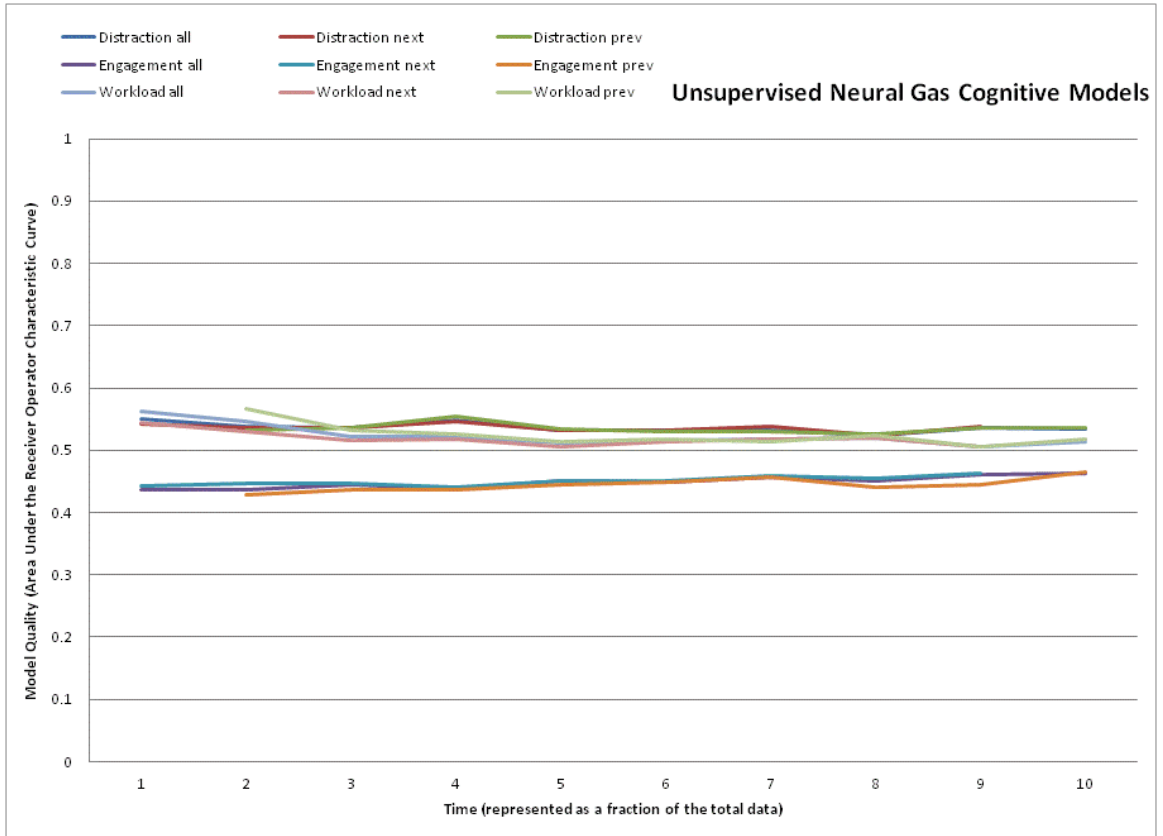
Figure 59 – Performance of unsupervised Growing Neural Gas for cognitive modeling
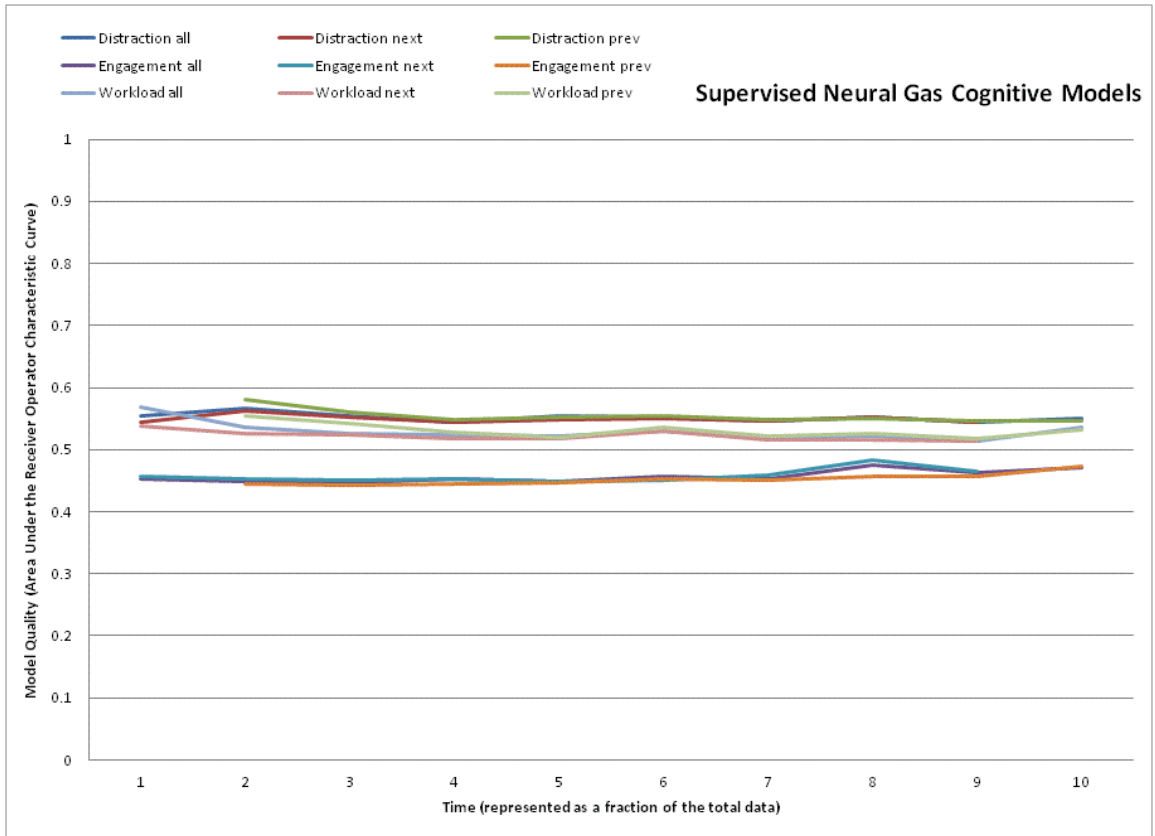
Figure 60 – Performance of supervised Growing Neural Gas for cognitive modeling
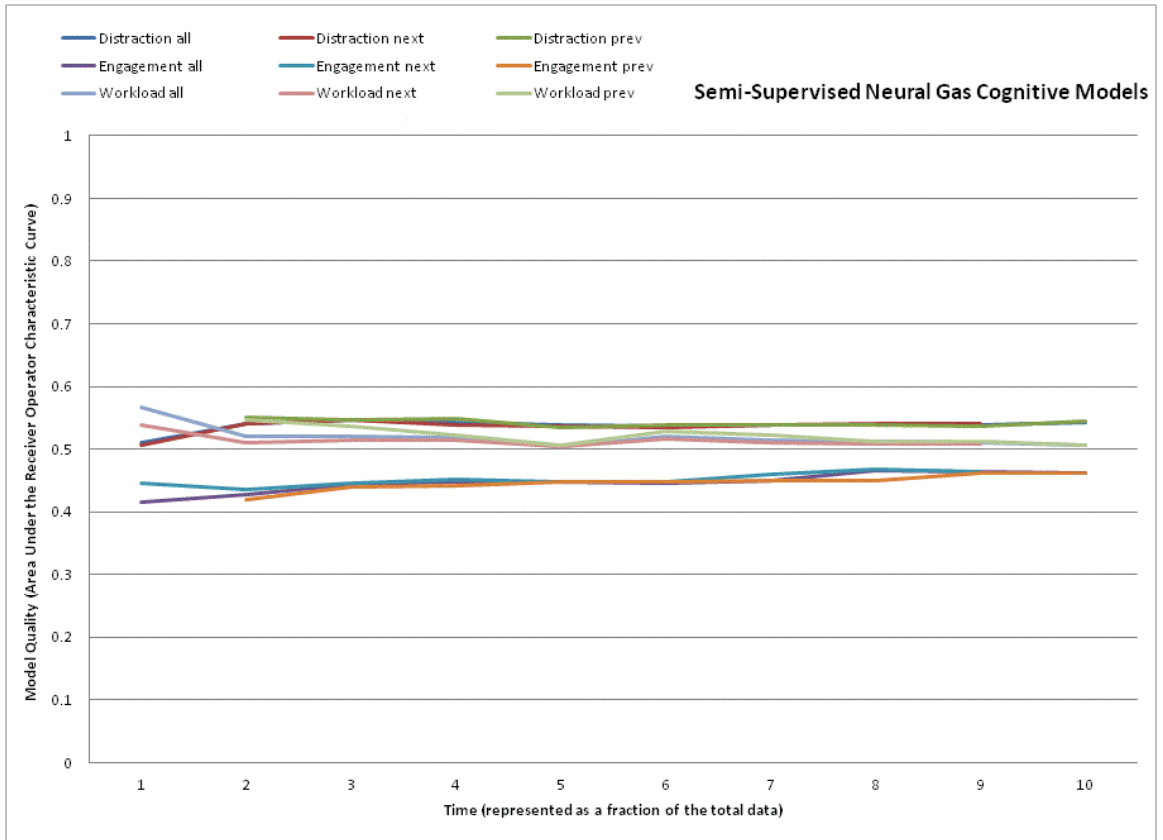
Figure 61 – Performance of semi-supervised Growing Neural Gas for cognitive modeling
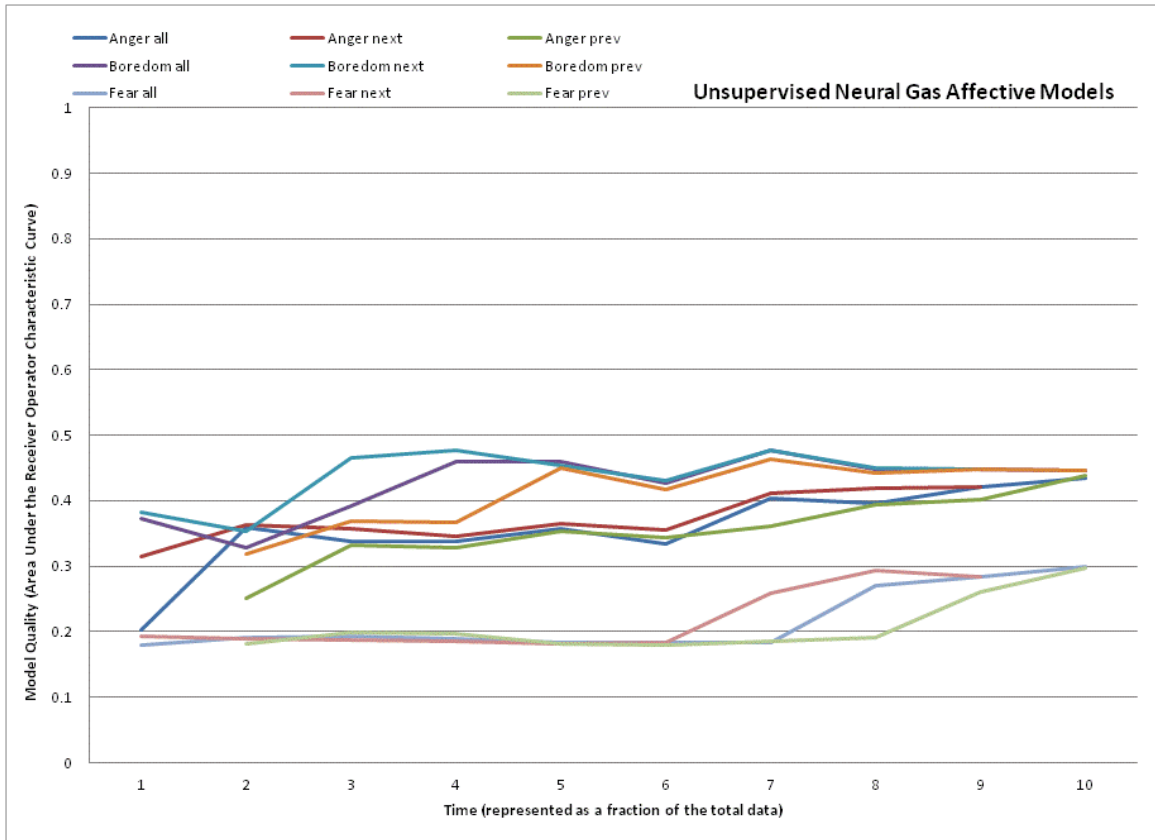
Figure 62 – Performance of unsupervised Growing Neural Gas for affective modeling
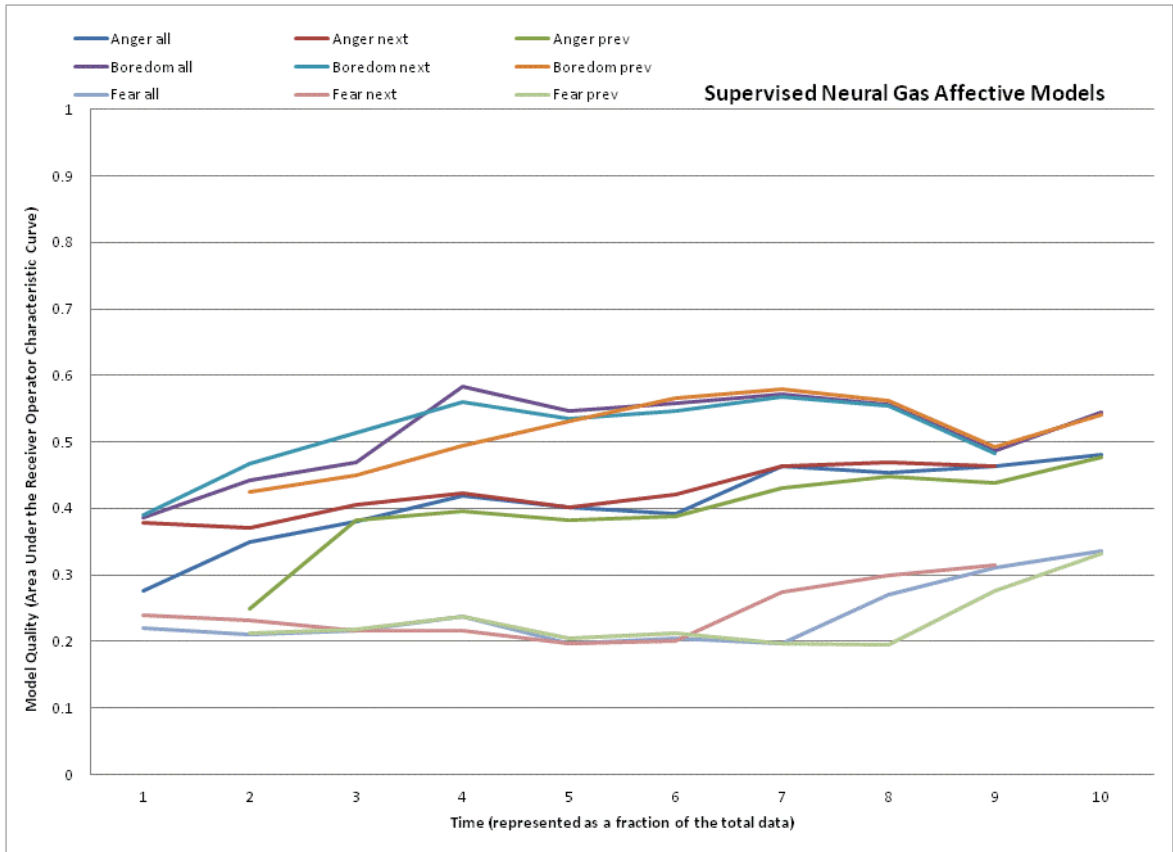
Figure 63 – Performance of supervised Growing Neural Gas for affective modeling
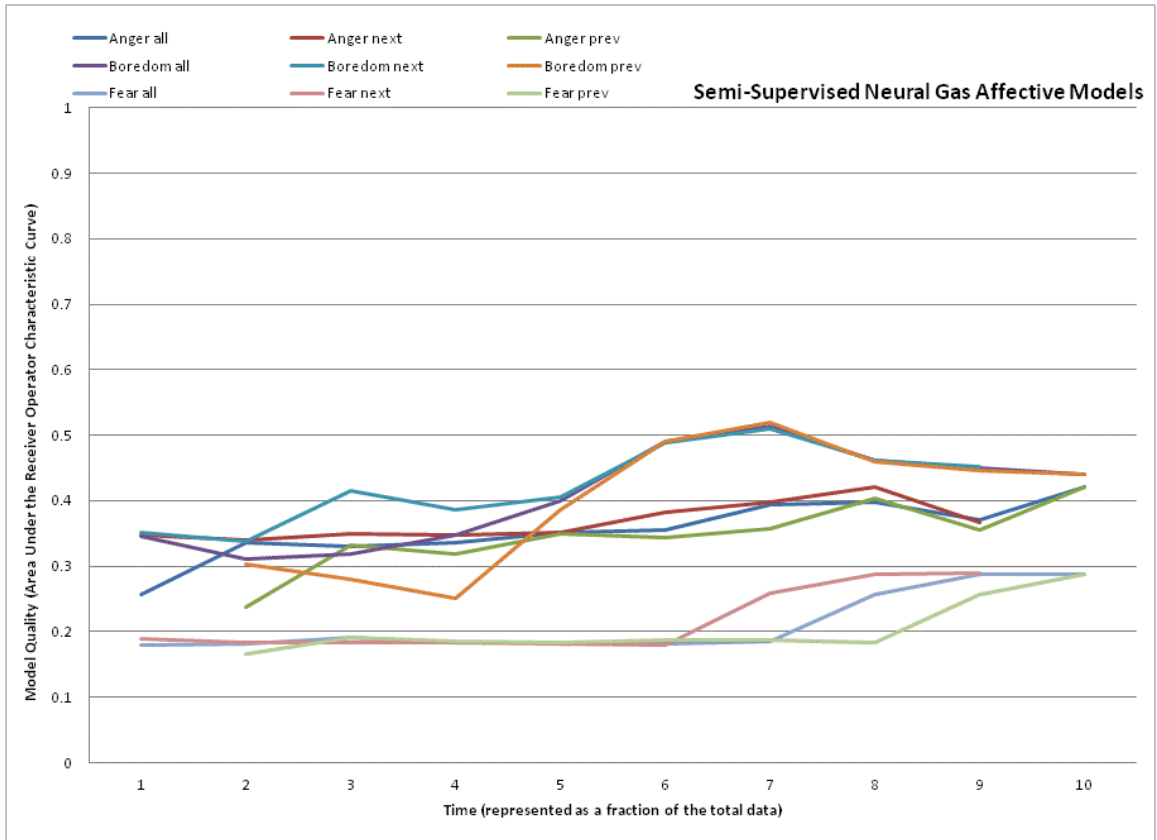
Figure 64 – Performance of semi-supervised Growing Neural Gas for affective modeling

Appendix C-1-4   Vowpal Wabbit
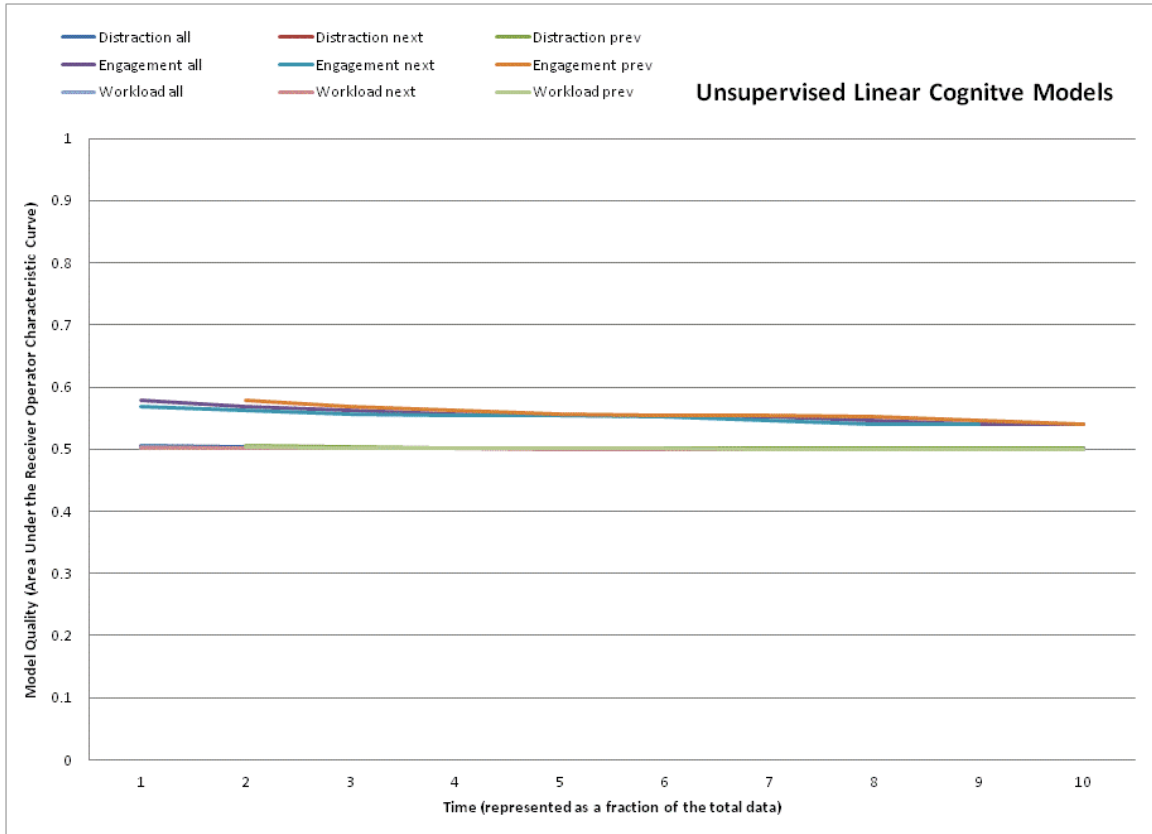
Figure 65 – Performance of unsupervised VW for linear cognitive modeling

329
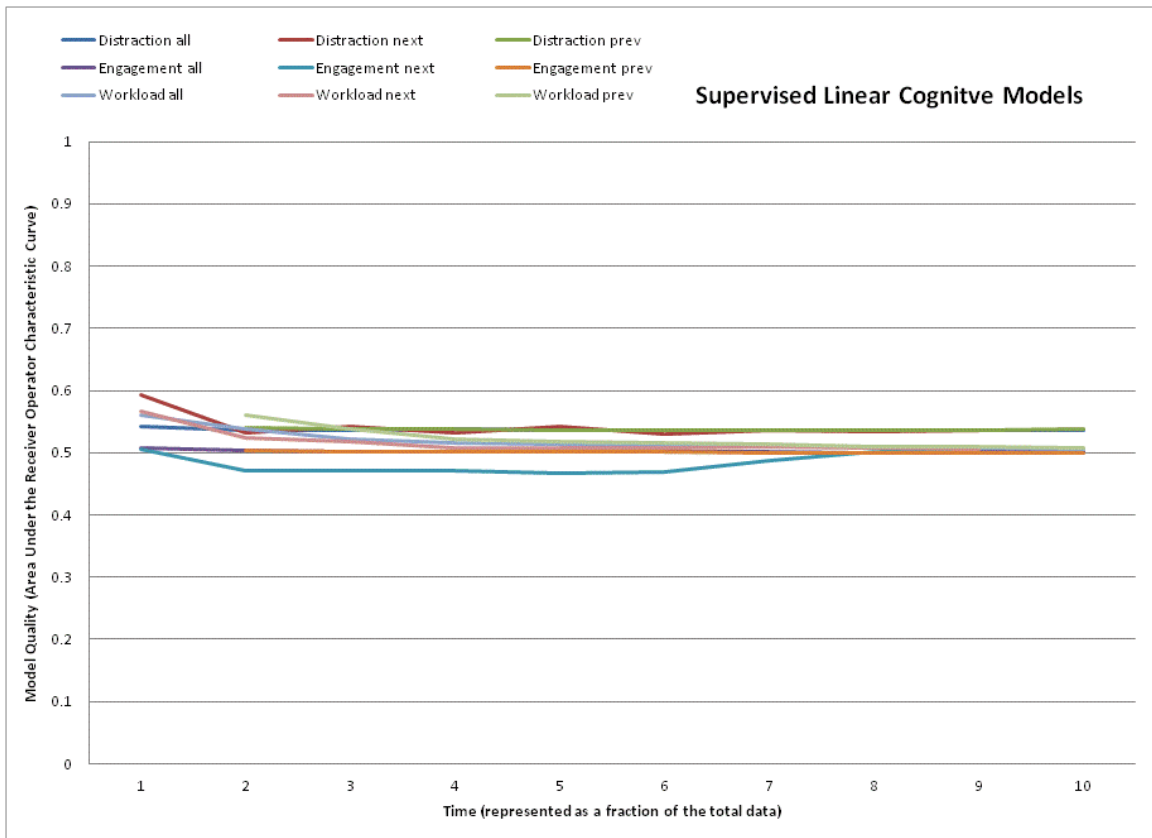
Figure 66 – Performance of supervised VW for linear cognitive modeling
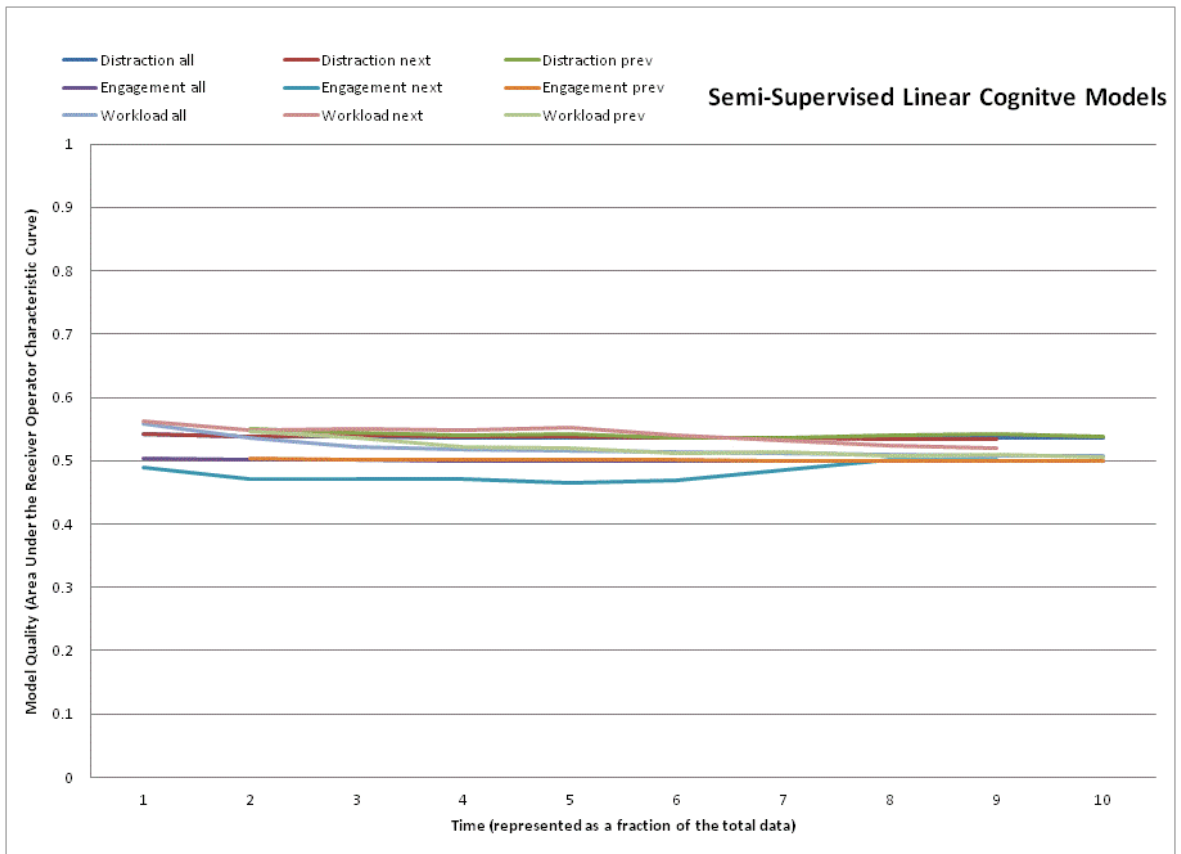
Figure 67 – Performance of semi-supervised VW for linear cognitive modeling

331
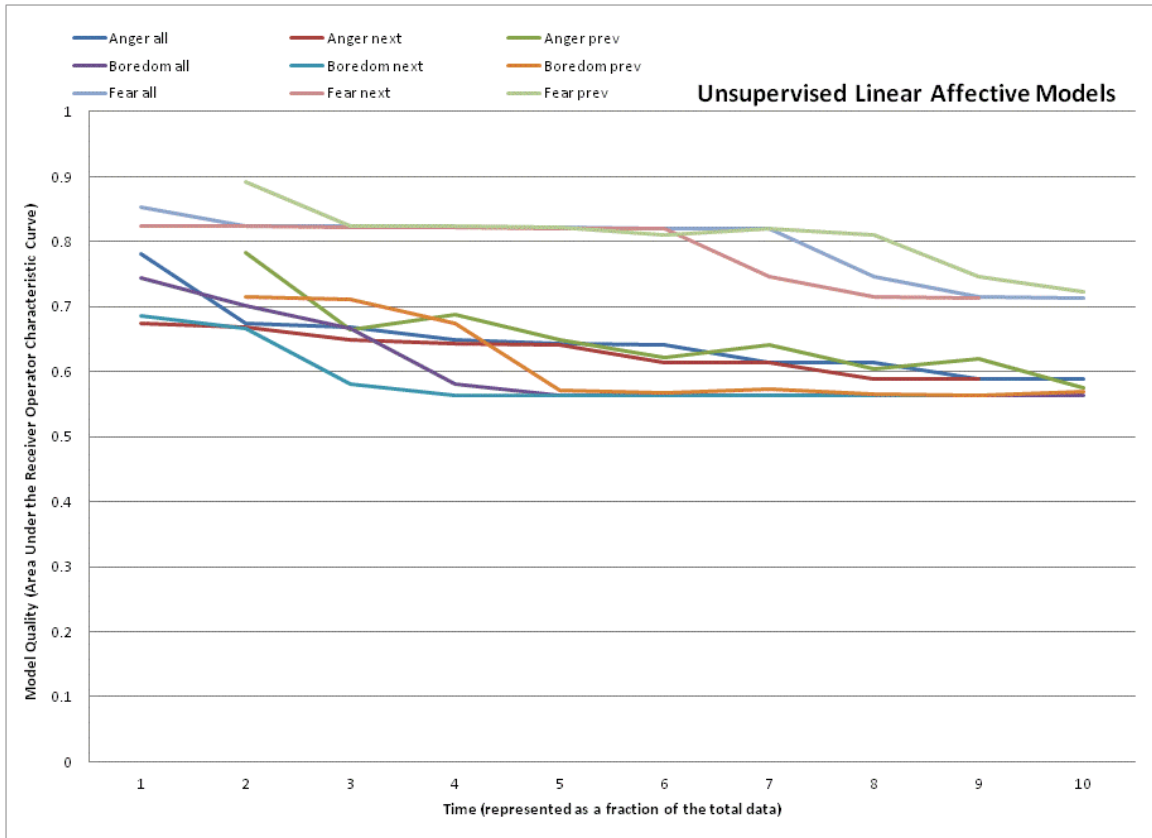
Figure 68 – Performance of unsupervised VW for linear affective modeling
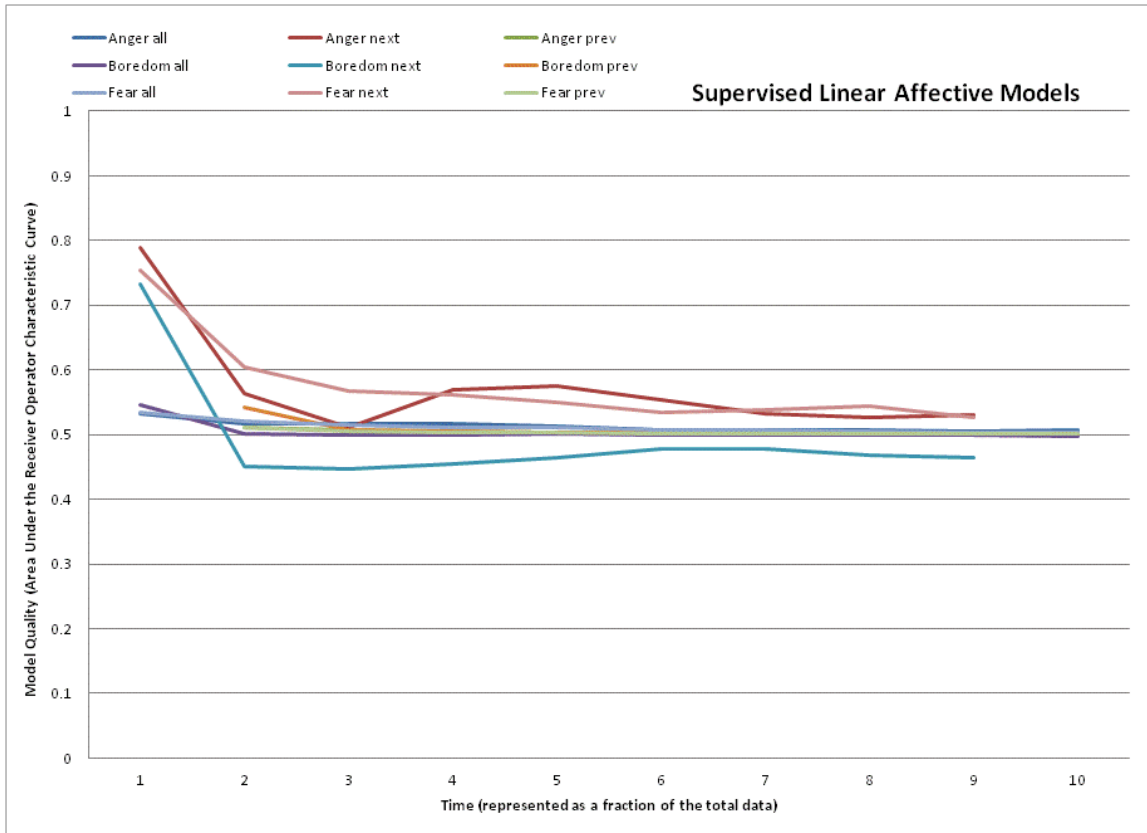
Figure 69 – Performance of supervised VW for linear affective modeling
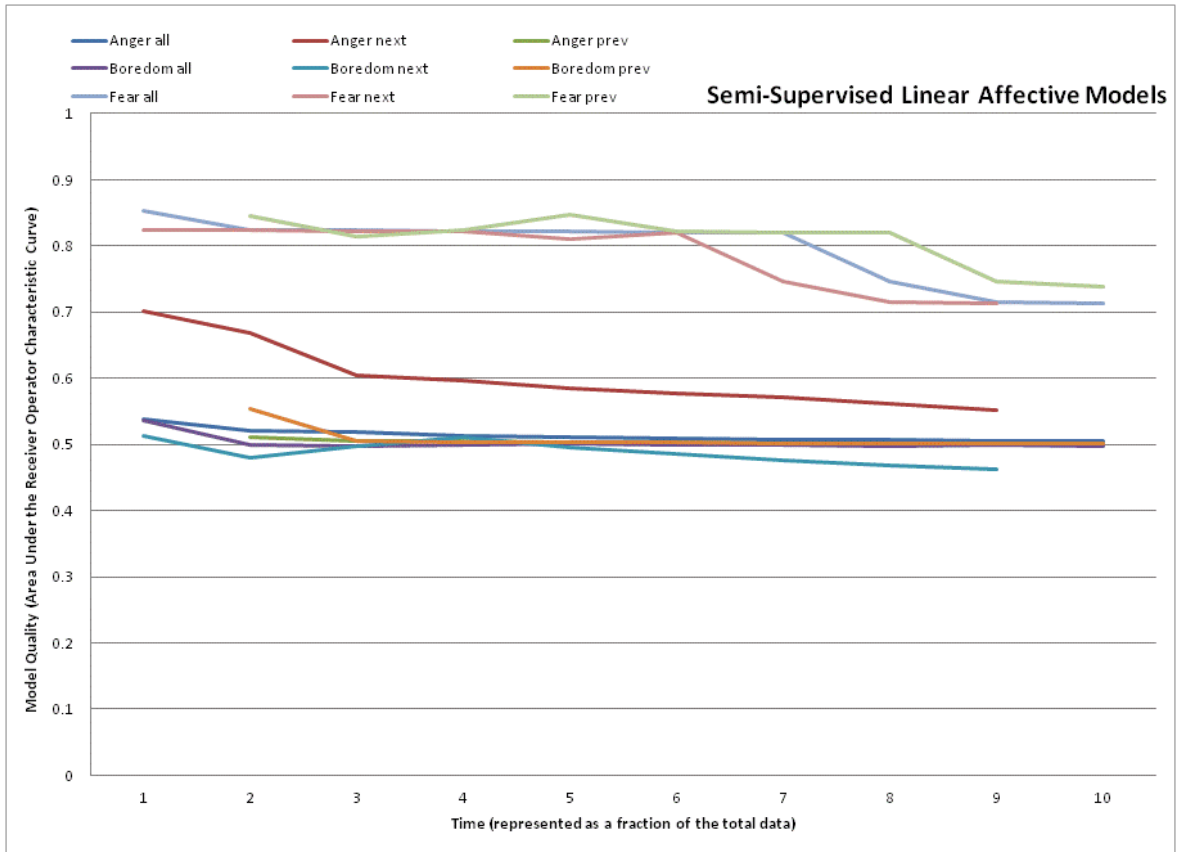
Figure 70 – Performance of semi-supervised VW for linear affective modeling

334
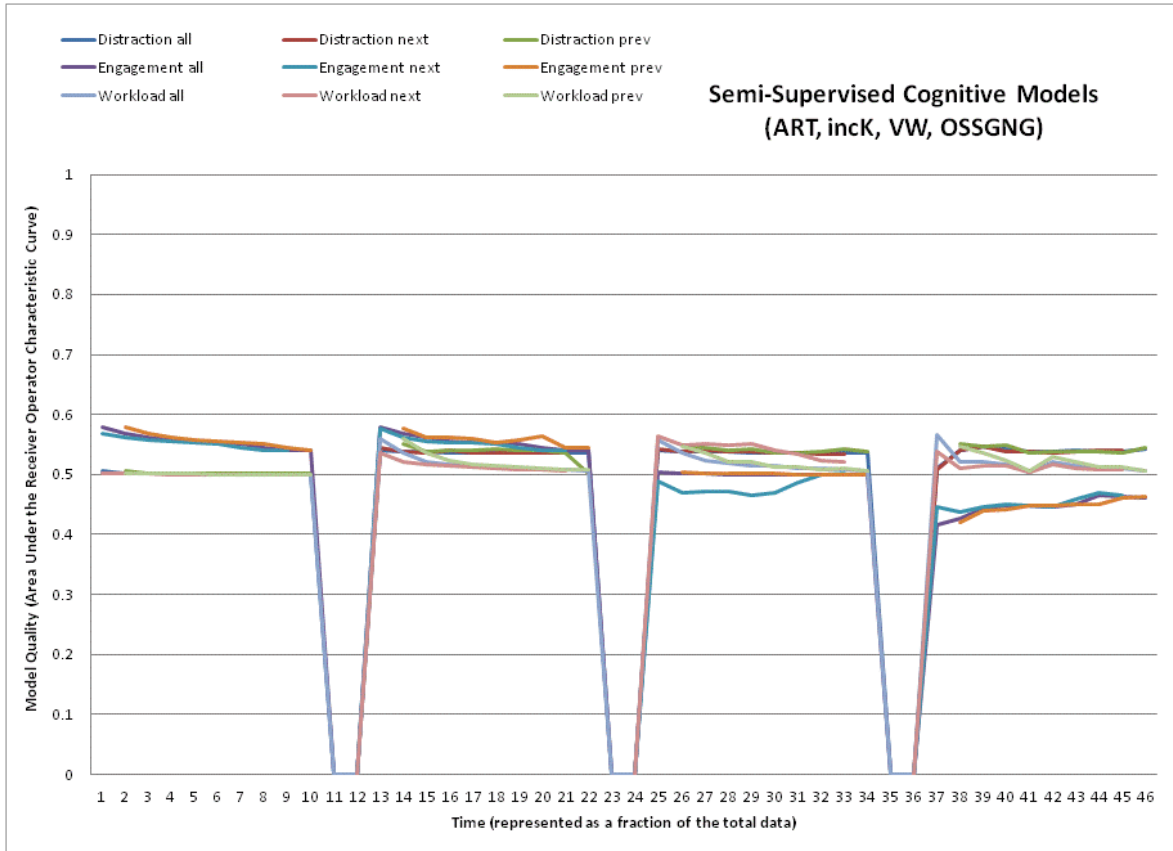
Figure 71 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for cognitive
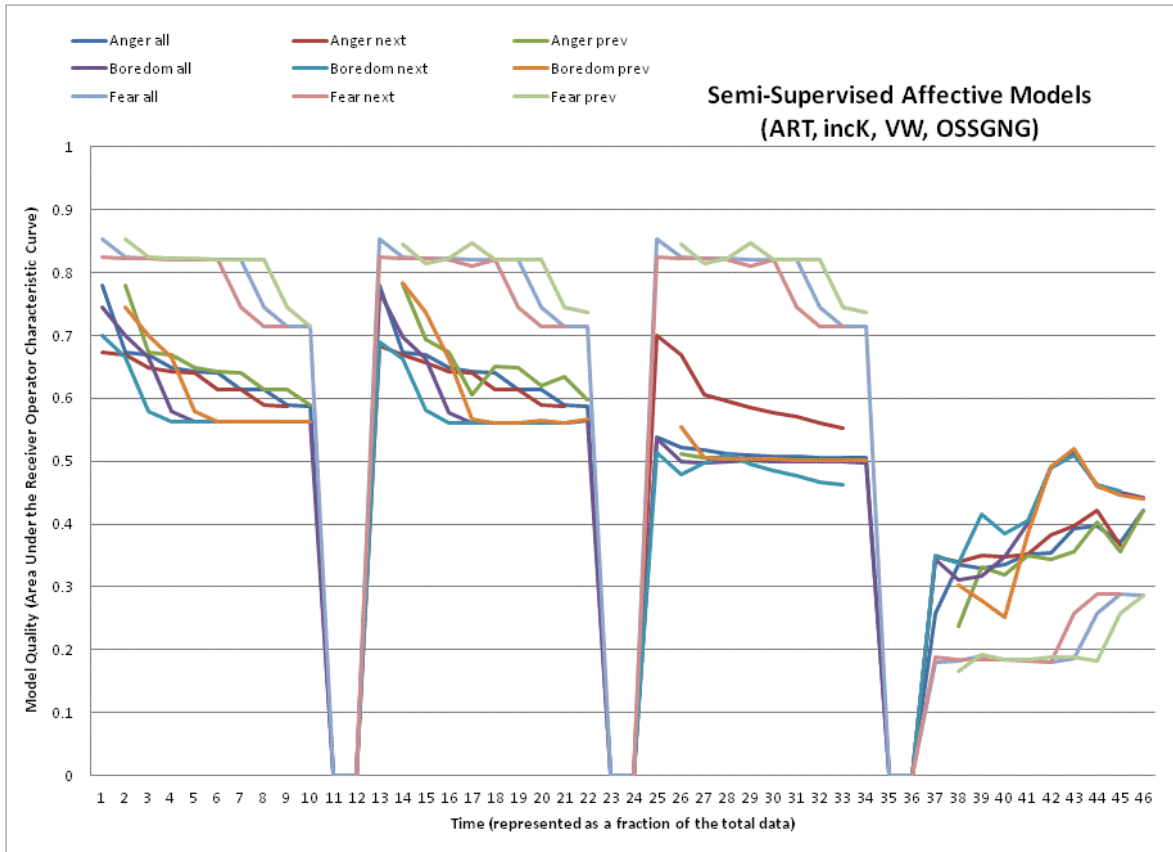
modeling

Figure 72 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for affective modeling

Appendix C-2   Results Set #2

The results in this section will be presented similar to the previous section, as summarized in Table 67. It will be broken into a section for the algorithm, the method of label assignment, and the type of model created. In each of these results graphs, the measures of classification quality, previous model quality, and predictive accuracy for each of the model types is shown. Results Set #2 additionally introduces workload

models produced from Dataset #2 analysis, and the altered parameter settings from Results Set #1 experimentation.
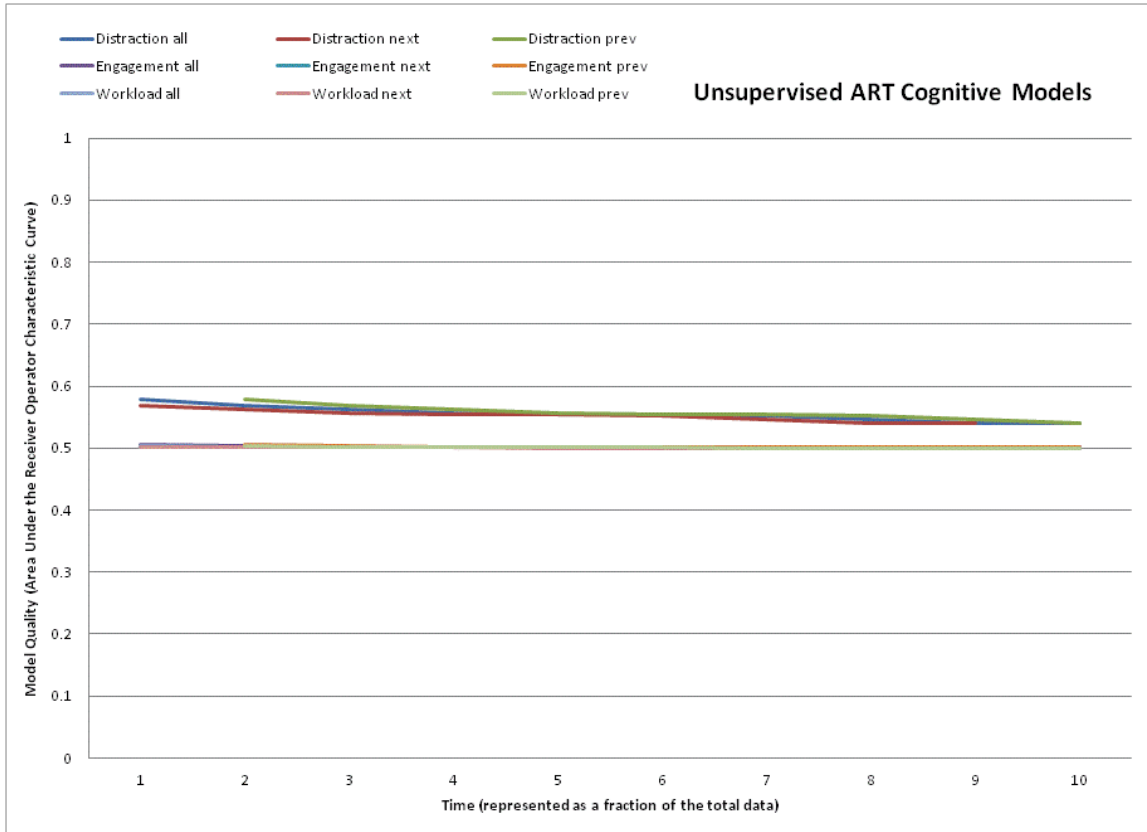
Appendix C-2-1   ART (Dataset #1)
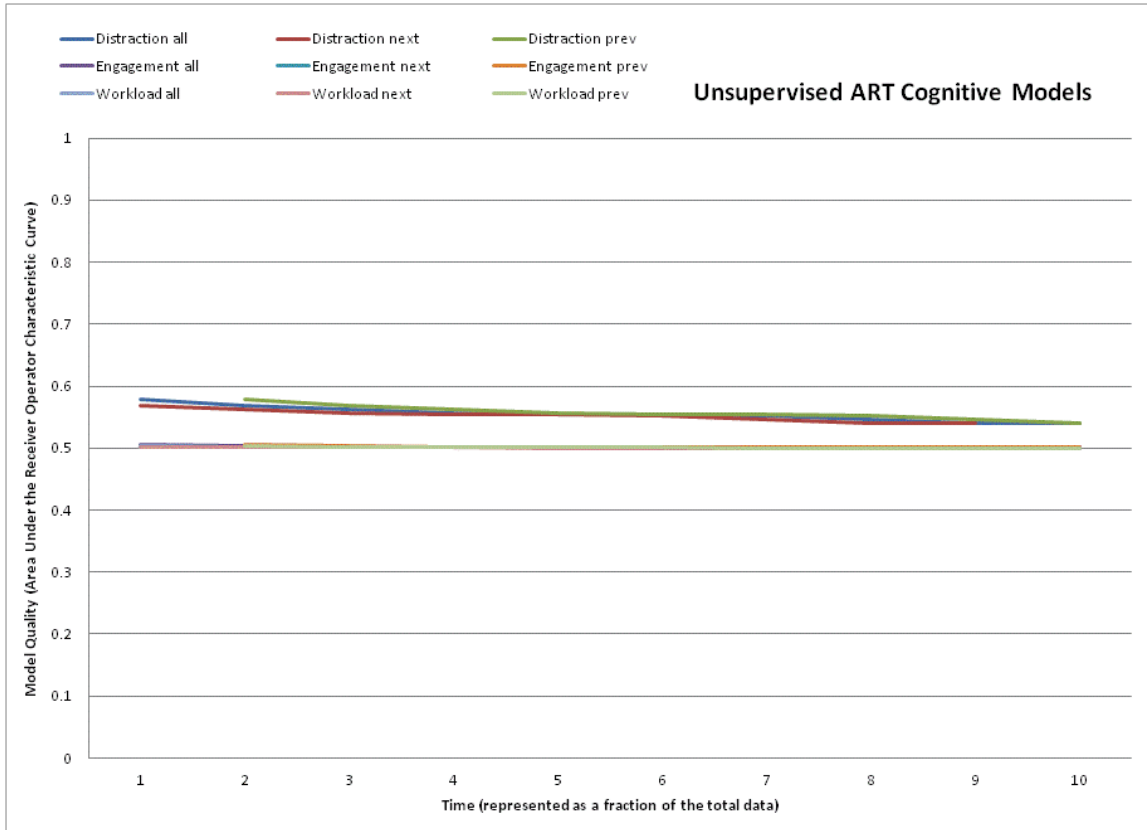
Figure 73 – Performance of unsupervised ART for cognitive modeling
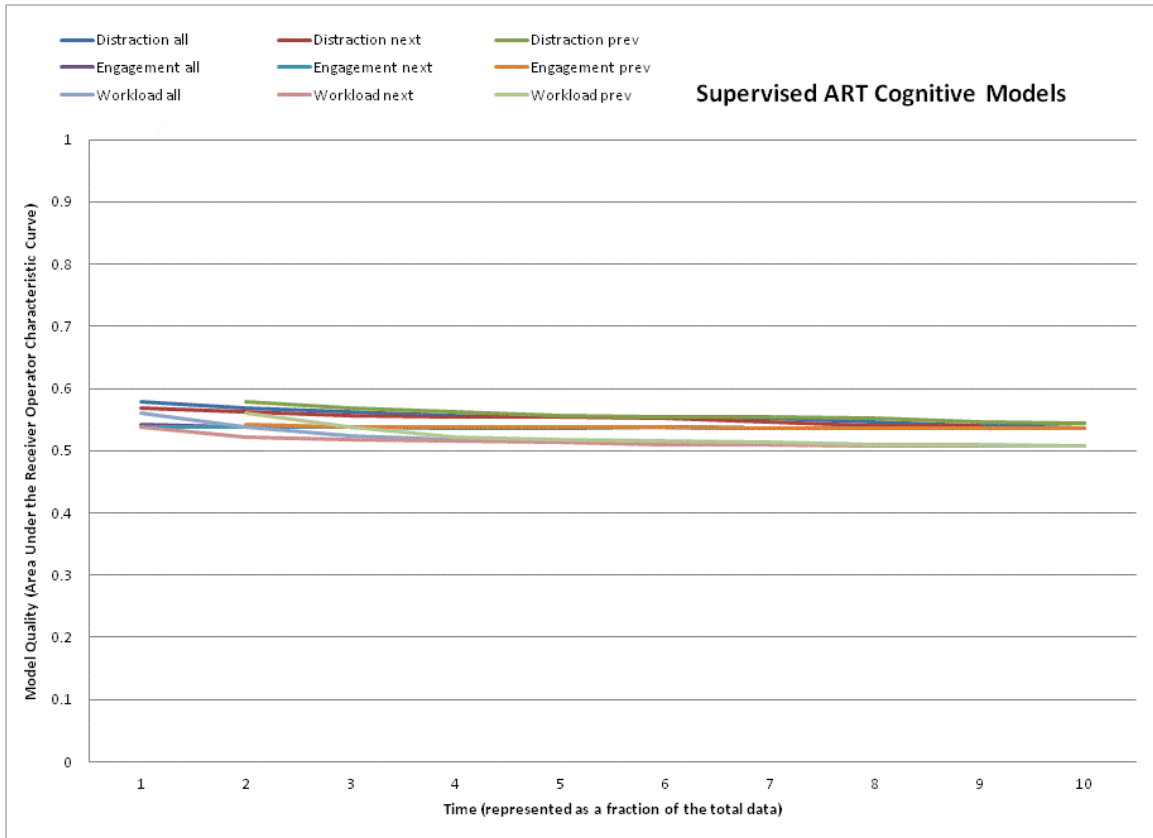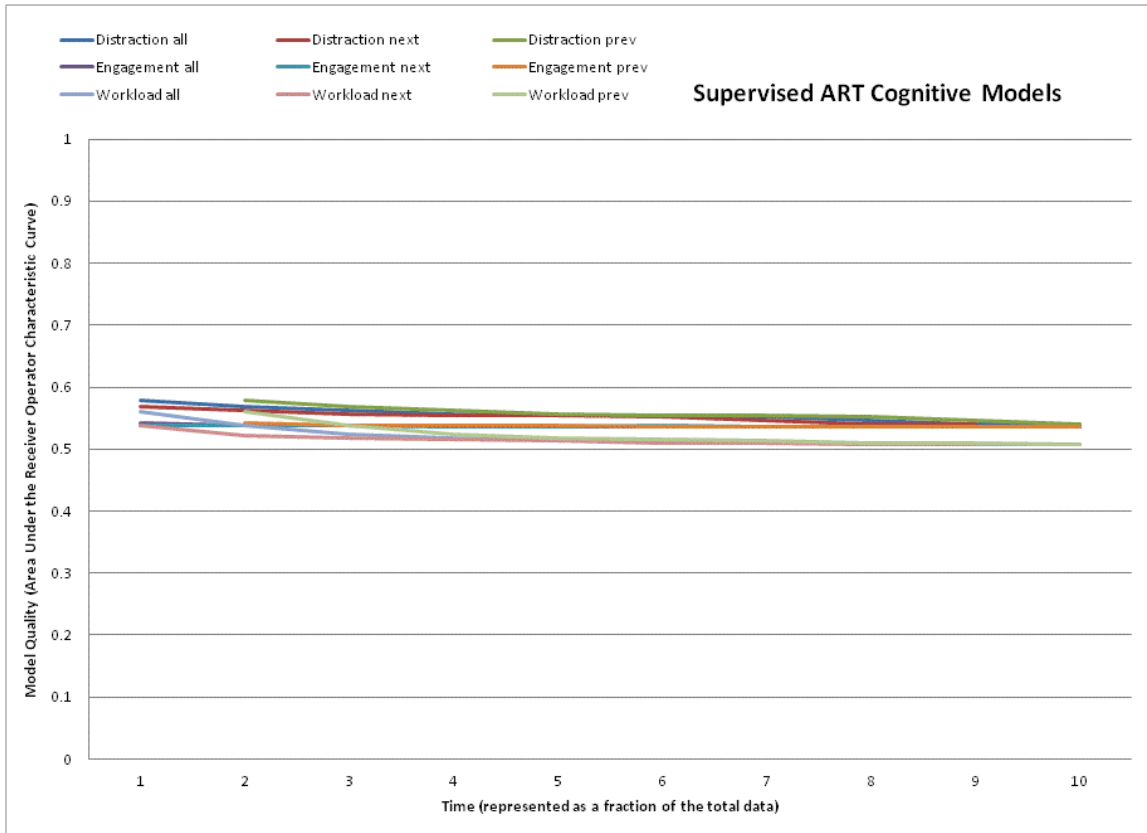
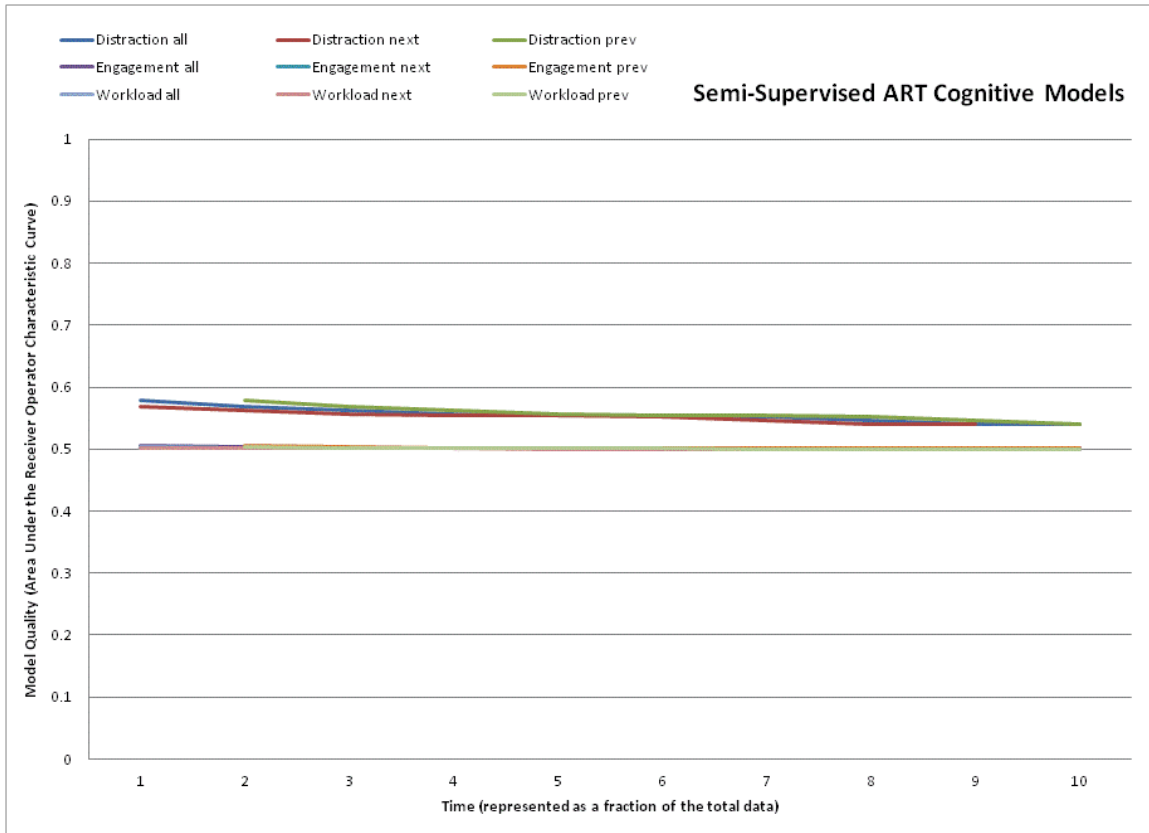**Figure 74 – Performance of supervised ART for cognitive modeling**
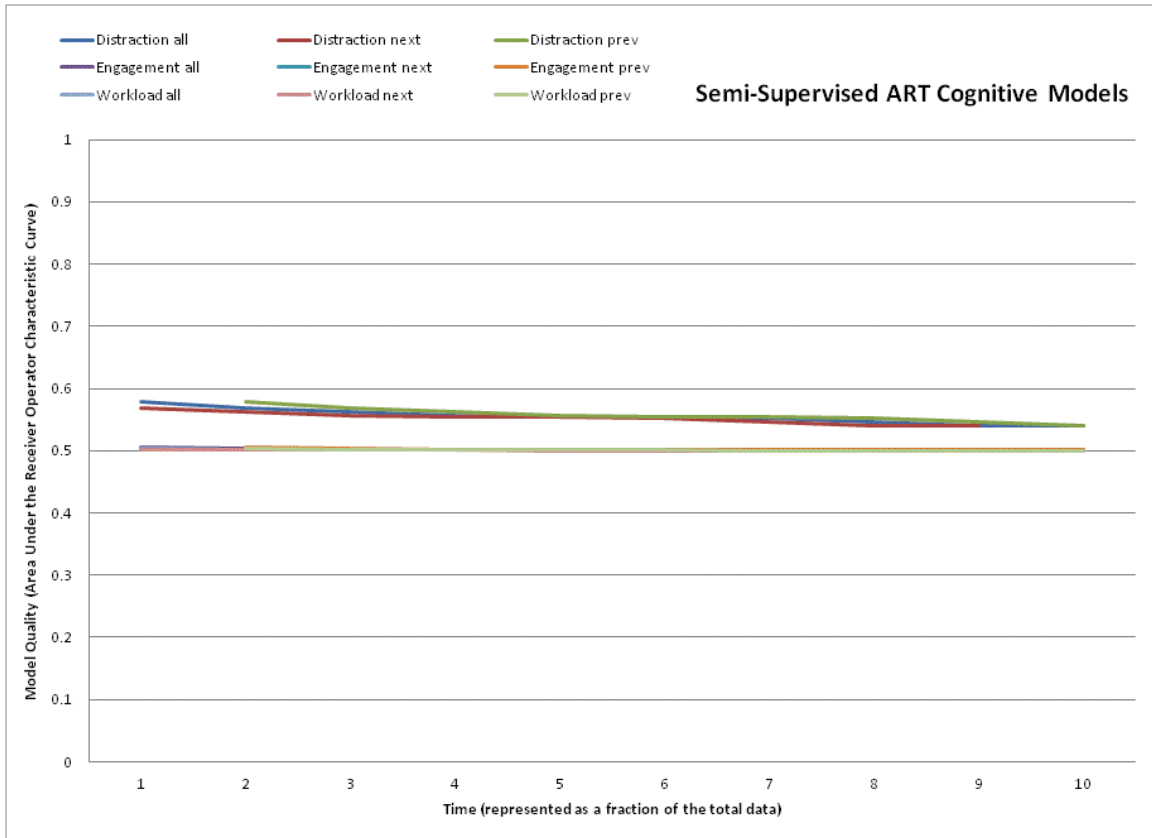
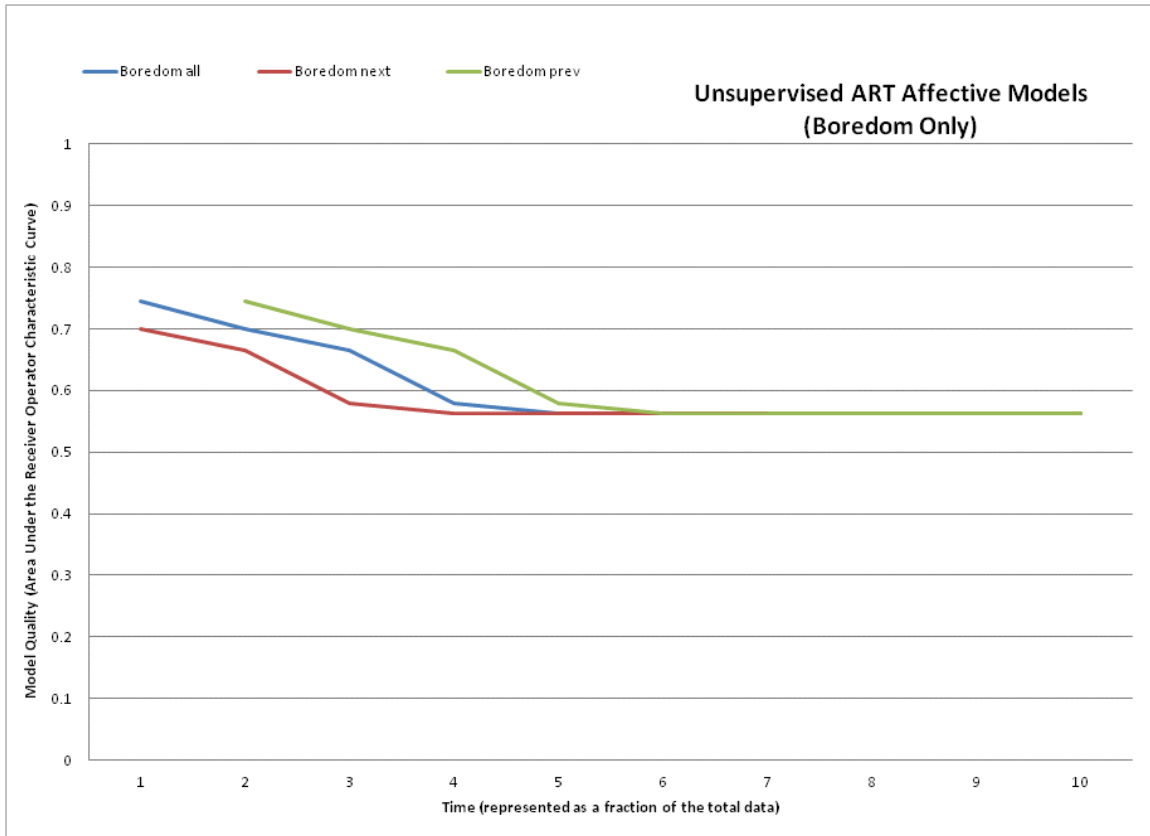**Figure 75 – Performance of semi-supervised ART for cognitive modeling**
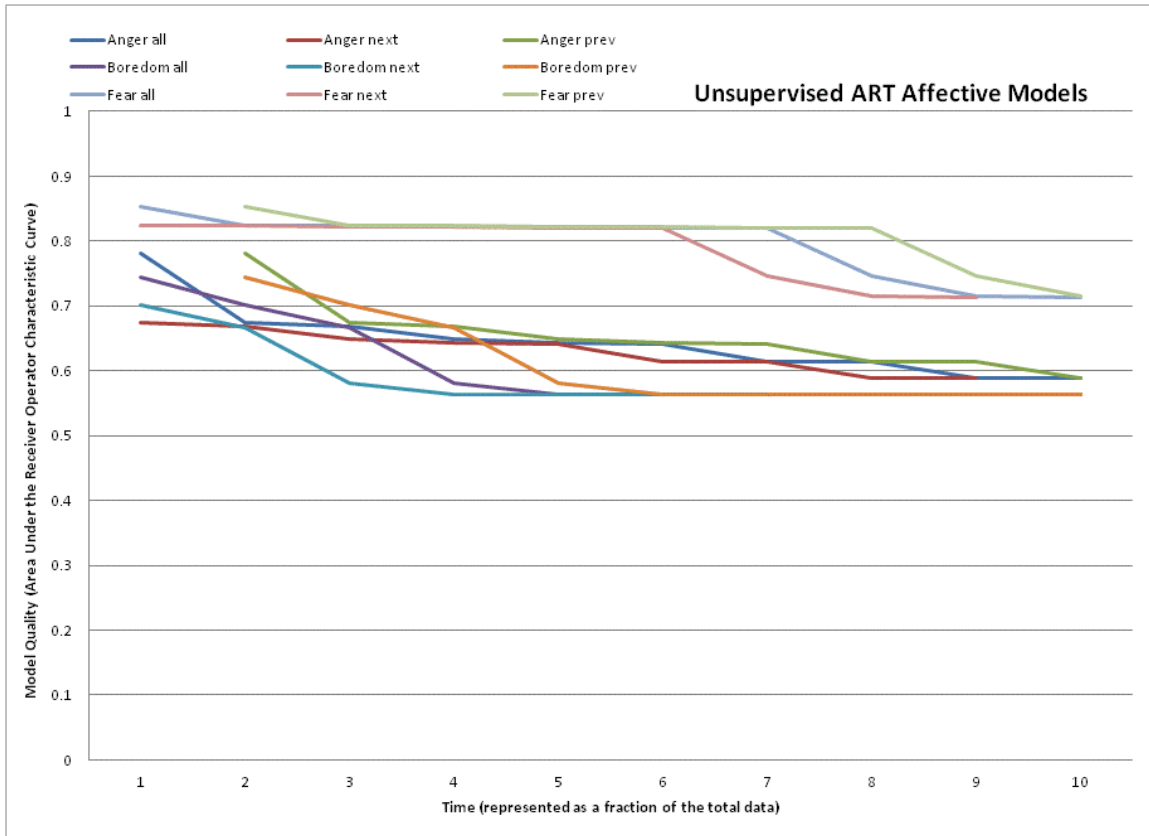
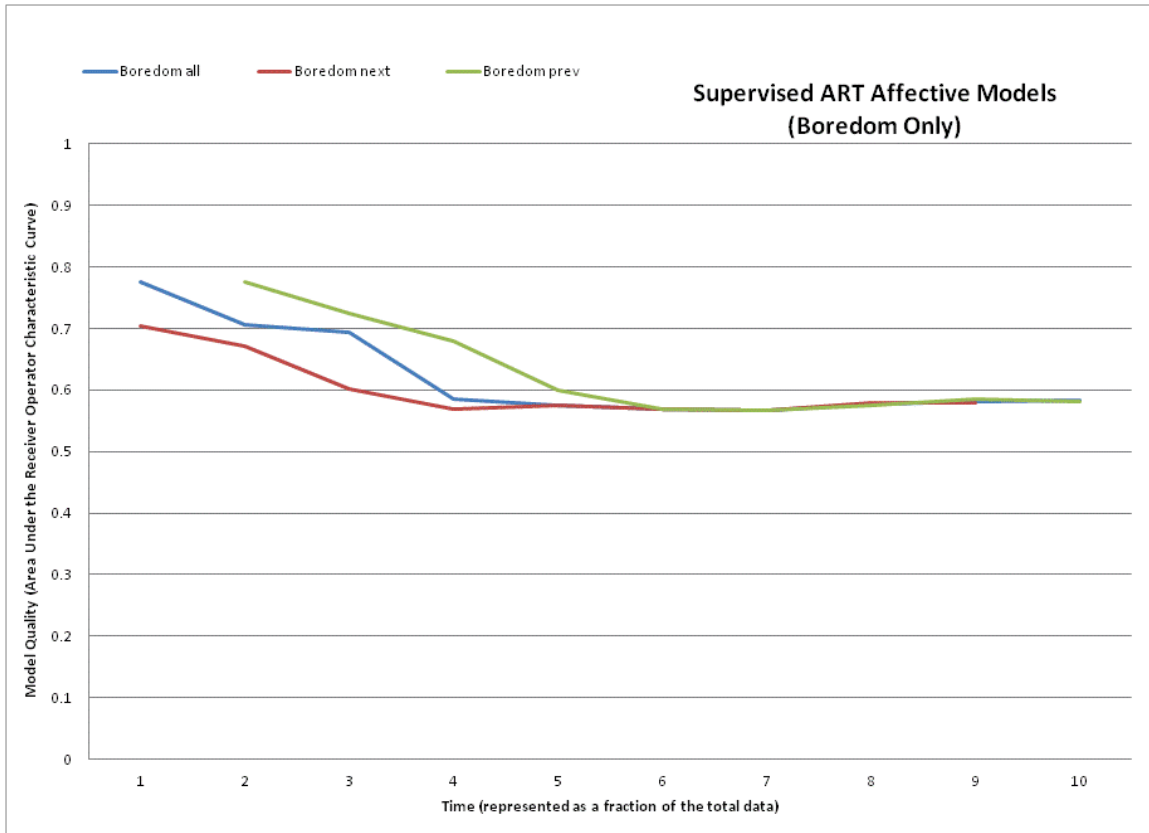**Figure 76 – Performance of unsupervised ART for affective modeling**
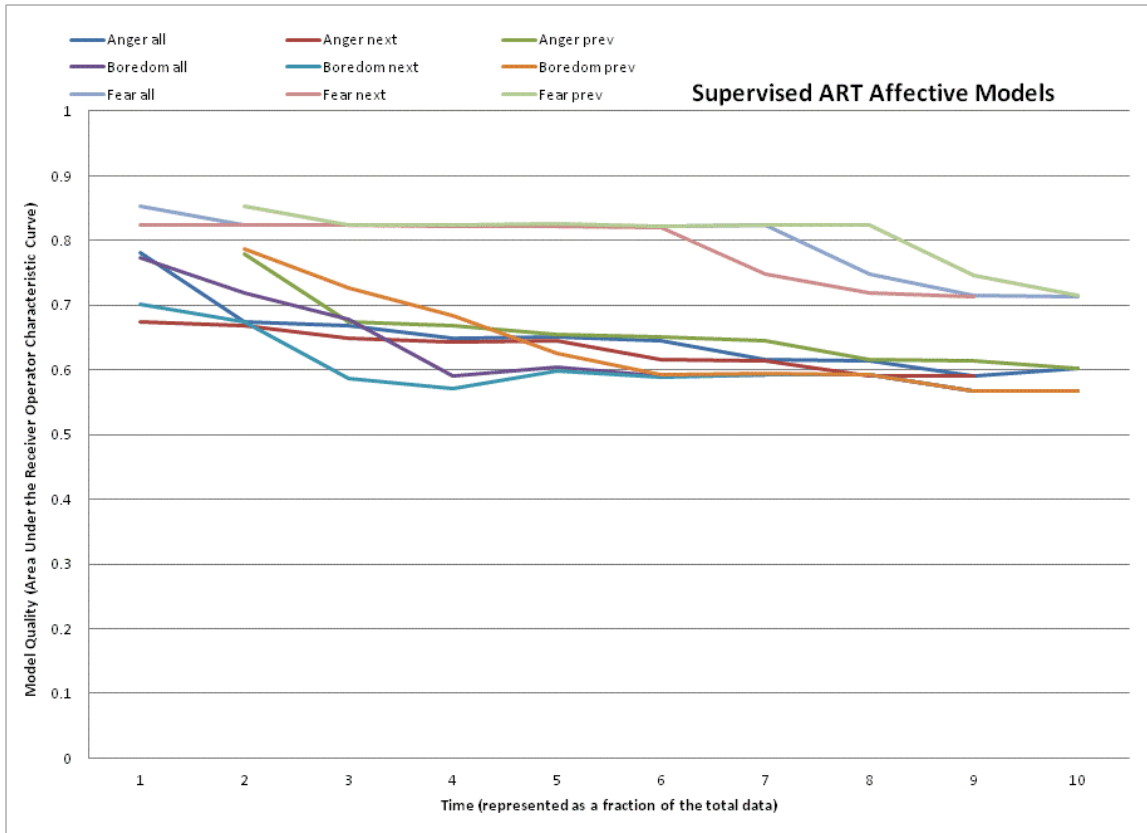
Figure 77 – Performance of supervised ART for affective modeling
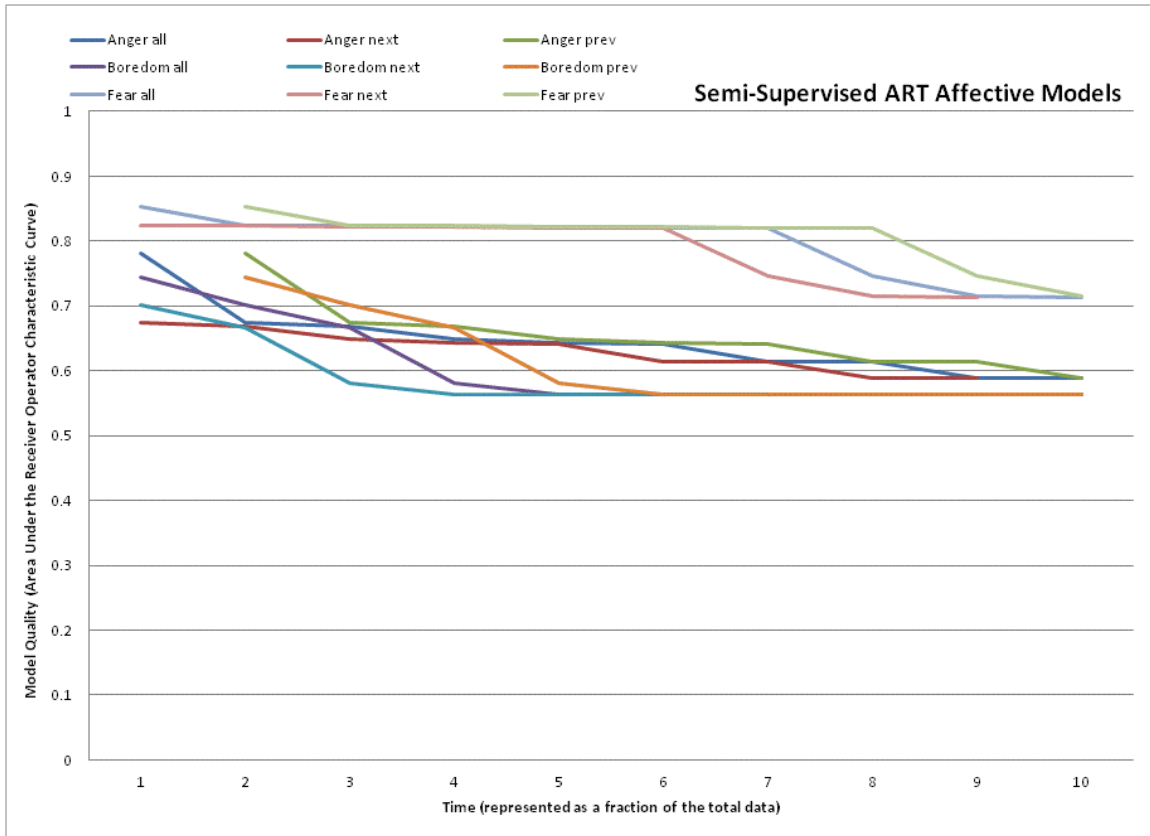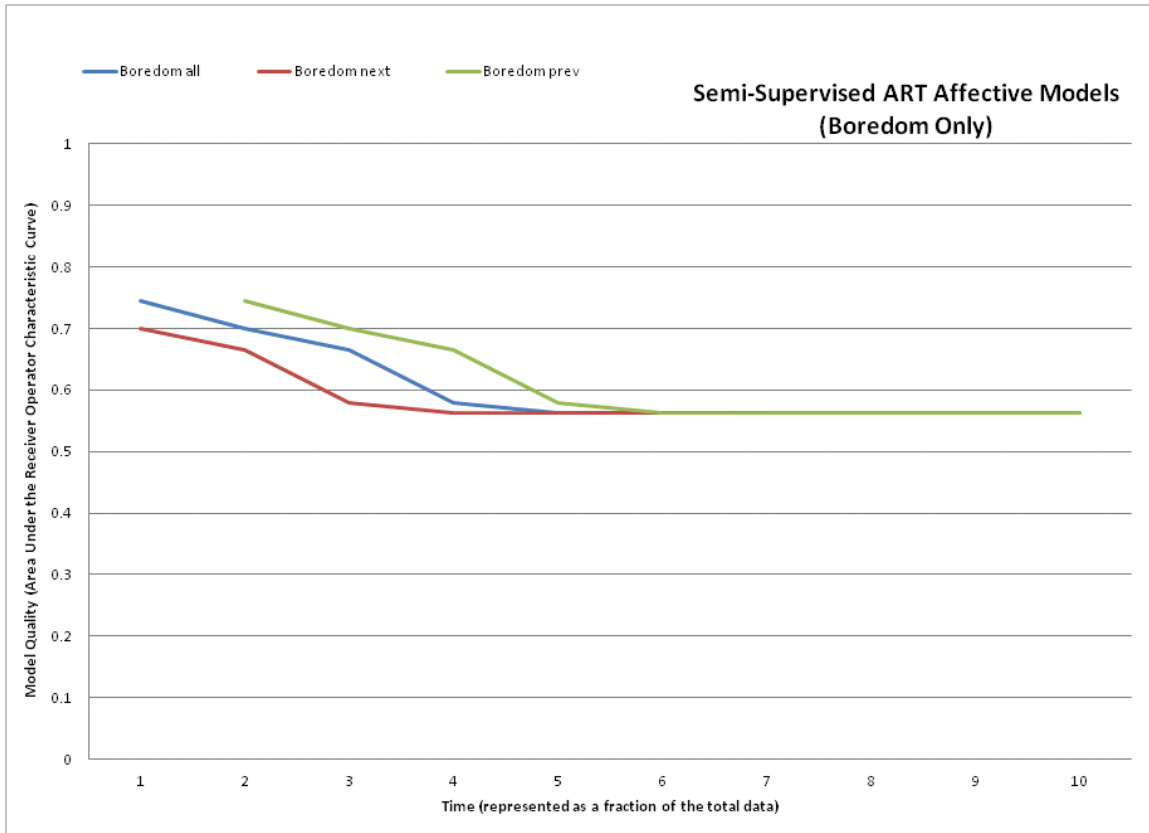
Figure 78 – Performance of semi-supervised ART for affective modeling

Appendix C-2-2   K-Means (Dataset #1)
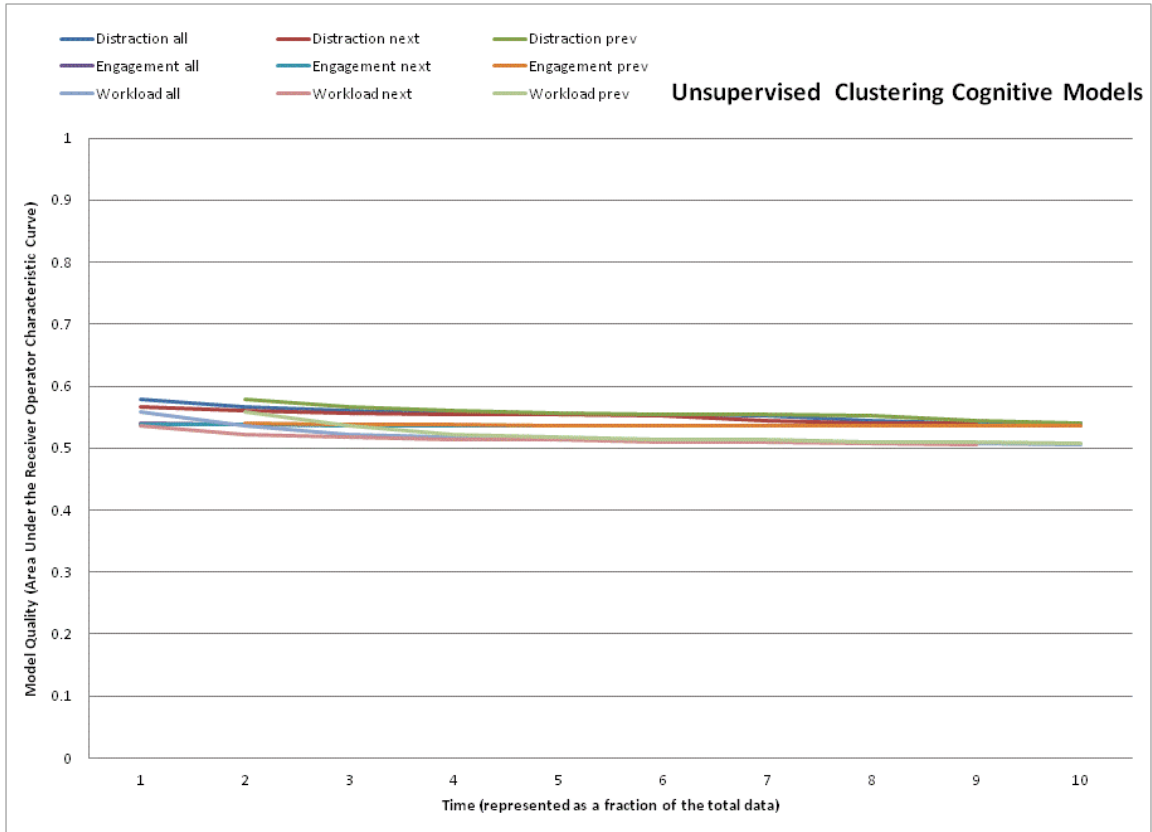
Figure 79 – Performance of unsupervised K-Means clustering for cognitive modeling
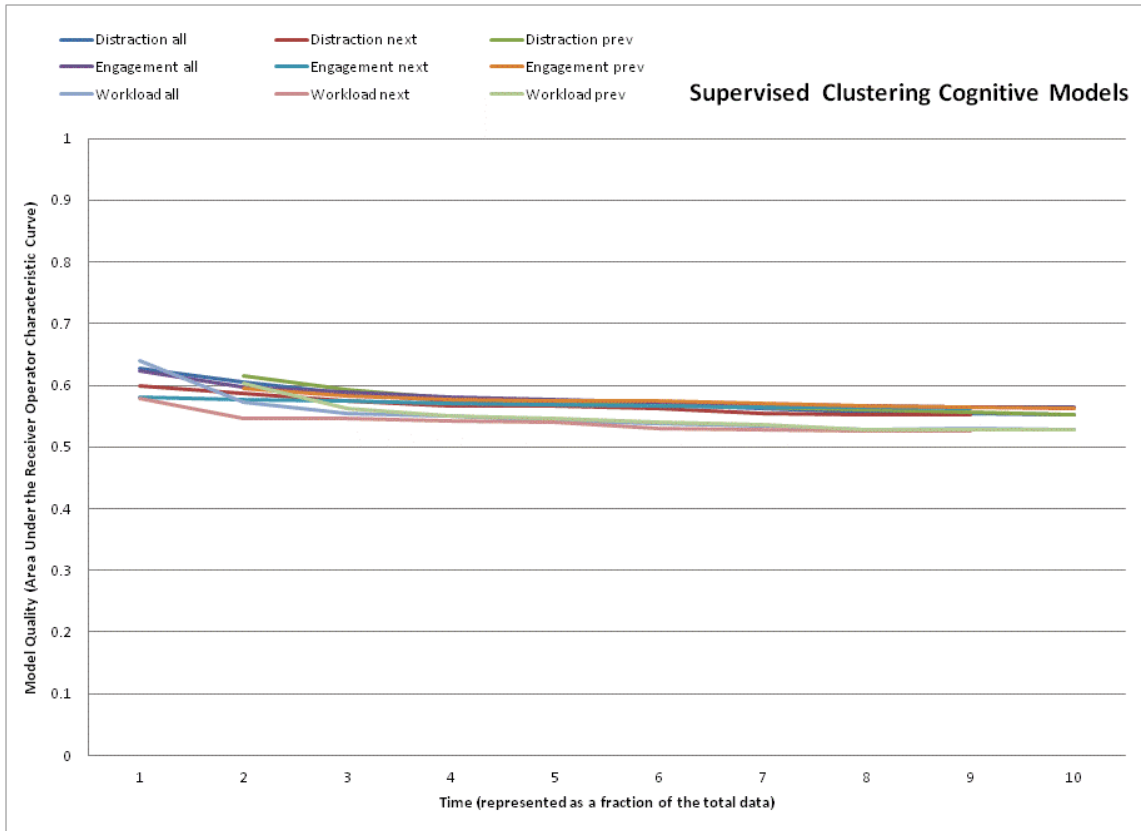
Figure 80 – Performance of supervised K-Means clustering for cognitive modeling
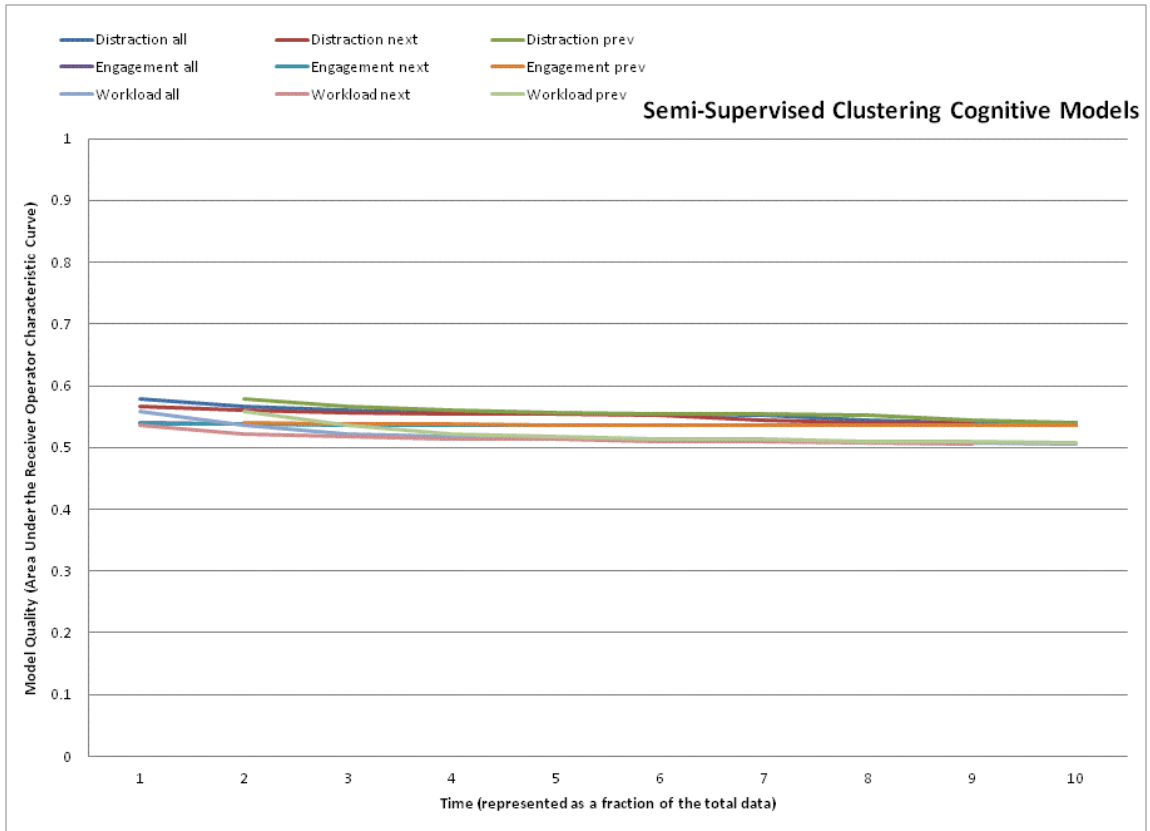
Figure 81 – Performance of semi-supervised K-Means clustering for cognitive modeling
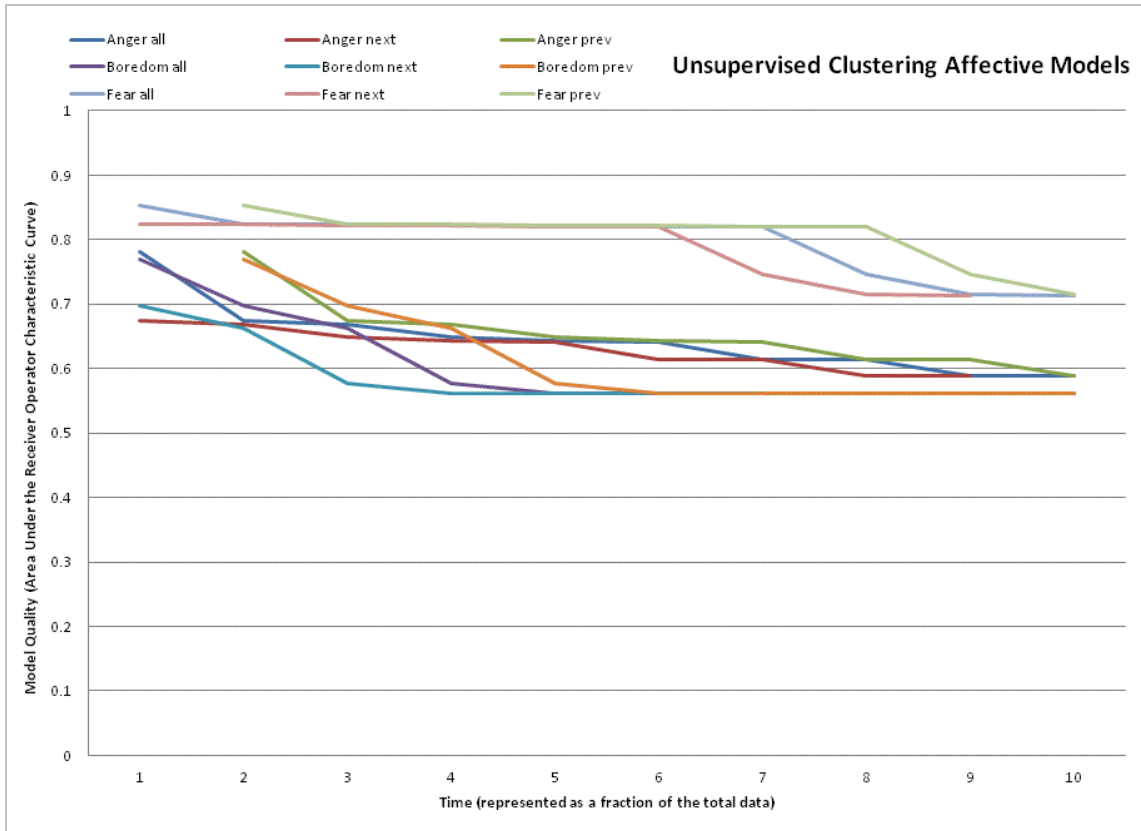
Figure 82 – Performance of unsupervised K-Means clustering for affective modeling
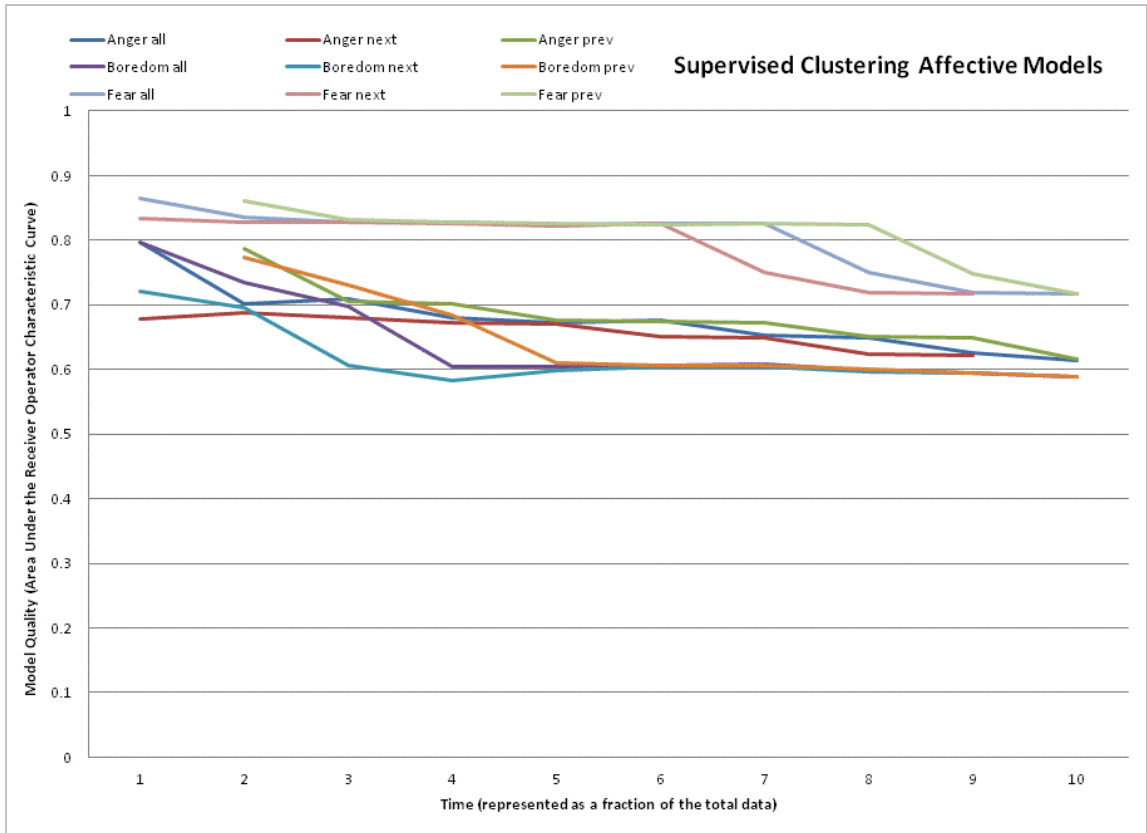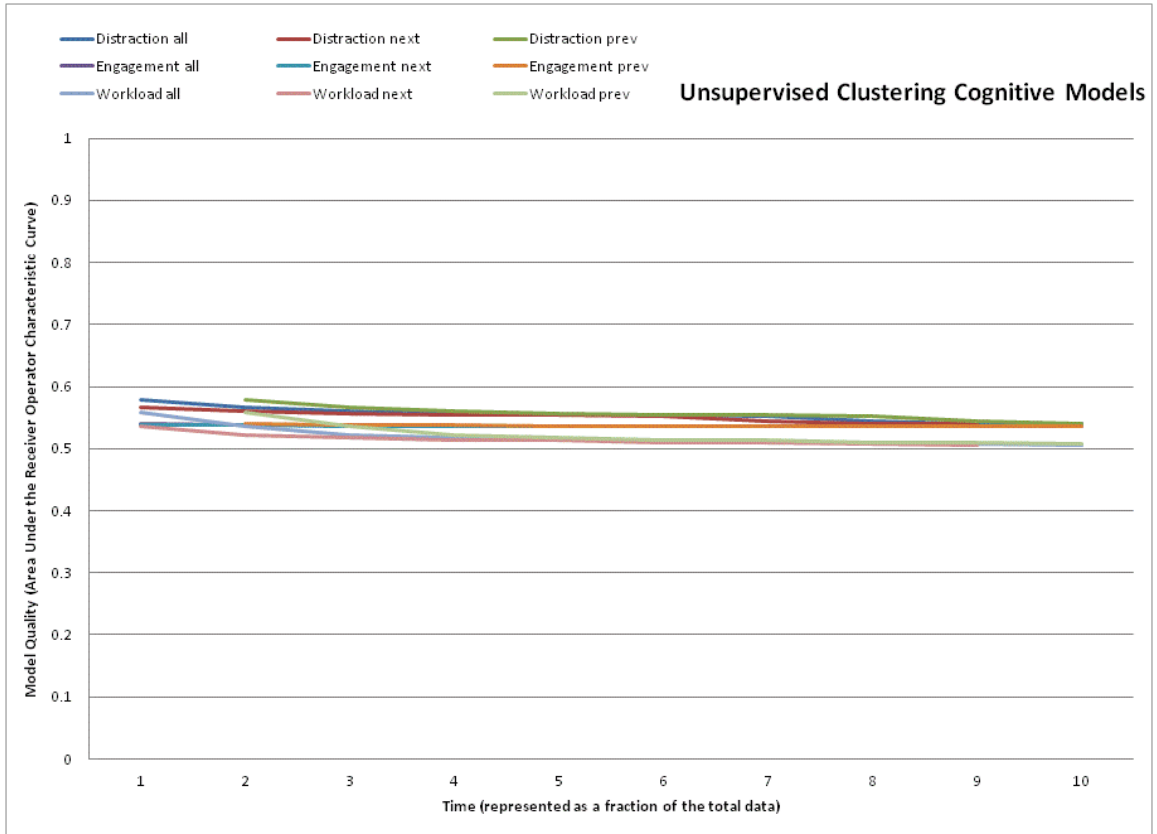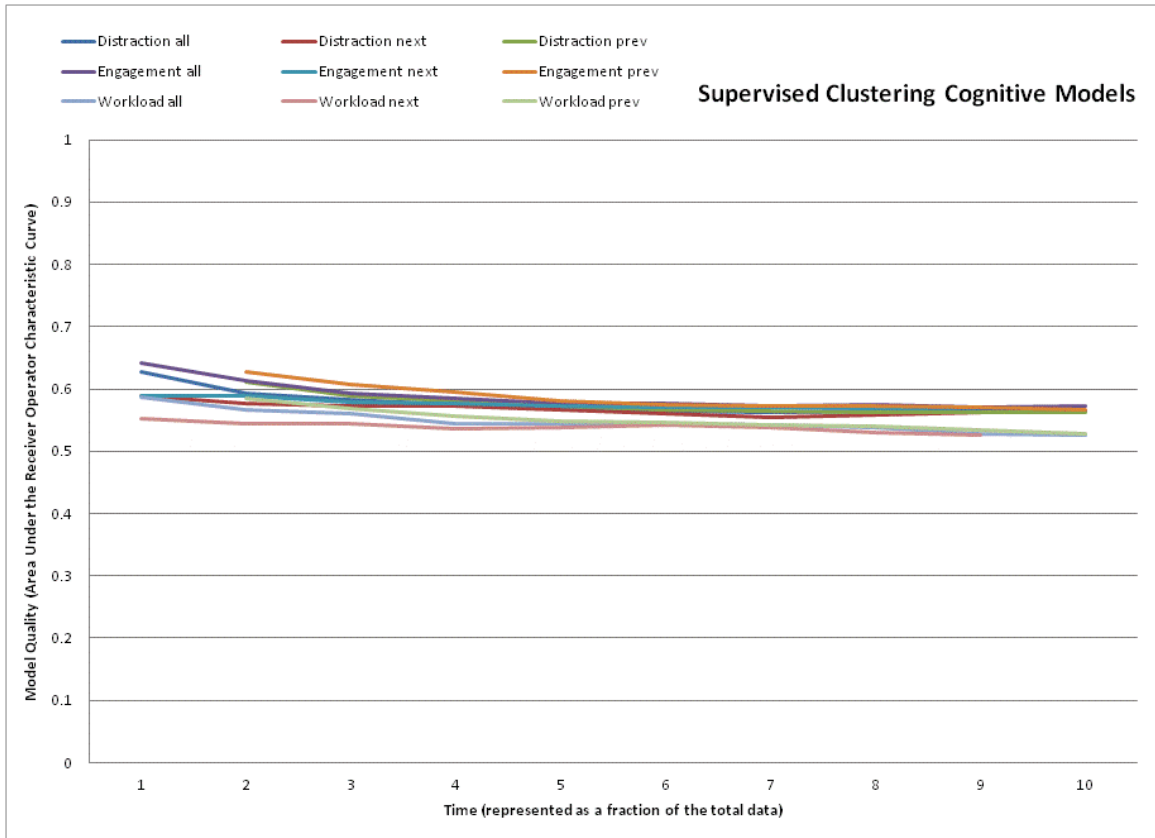
Figure 83 – Performance of supervised K-Means clustering for affective modeling
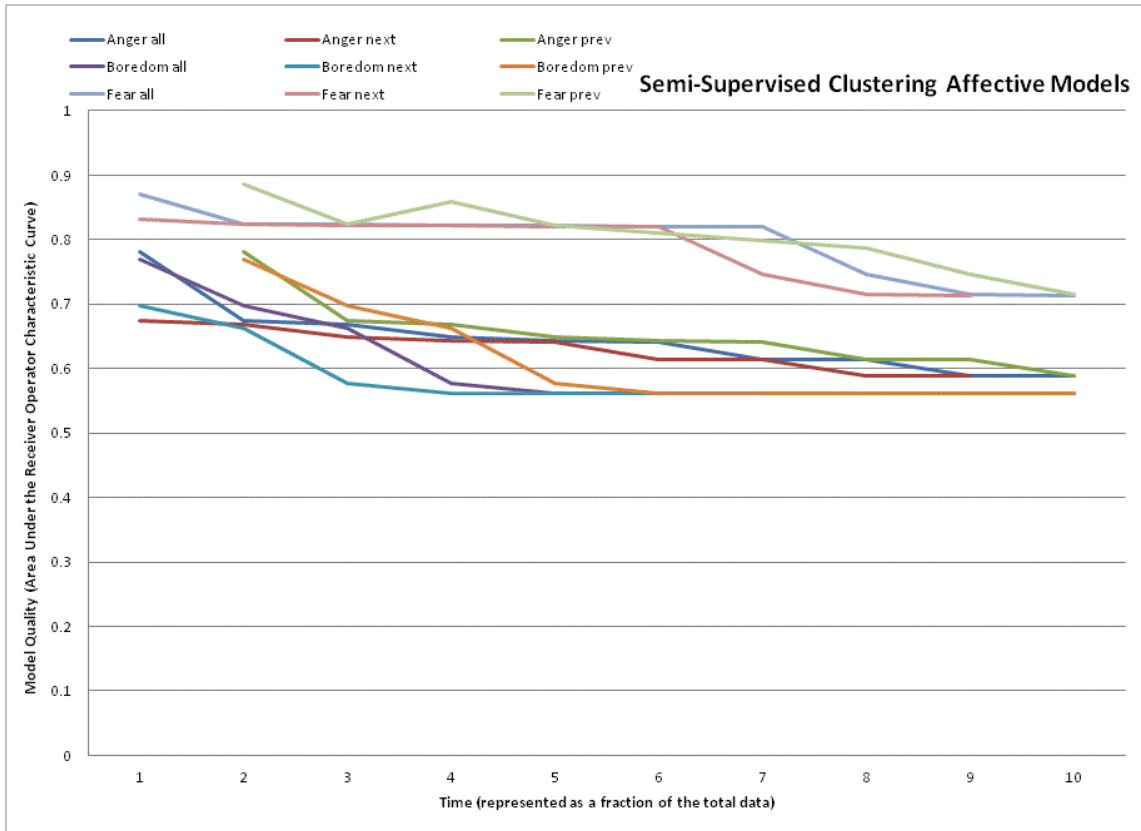
Figure 84 – Performance of semi-supervised K-Means clustering for affective modeling

Appendix C-2-3   GNG (Dataset #1)
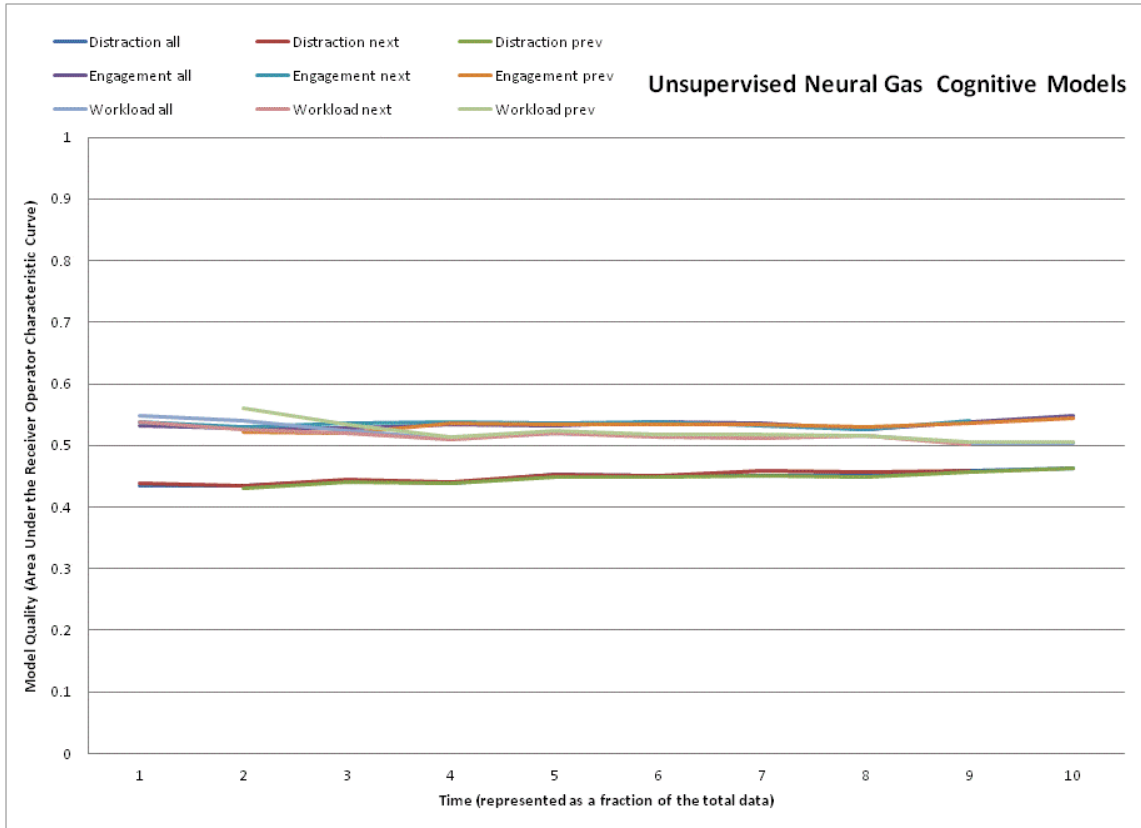
Figure 85 – Performance of unsupervised Growing Neural Gas for cognitive modeling
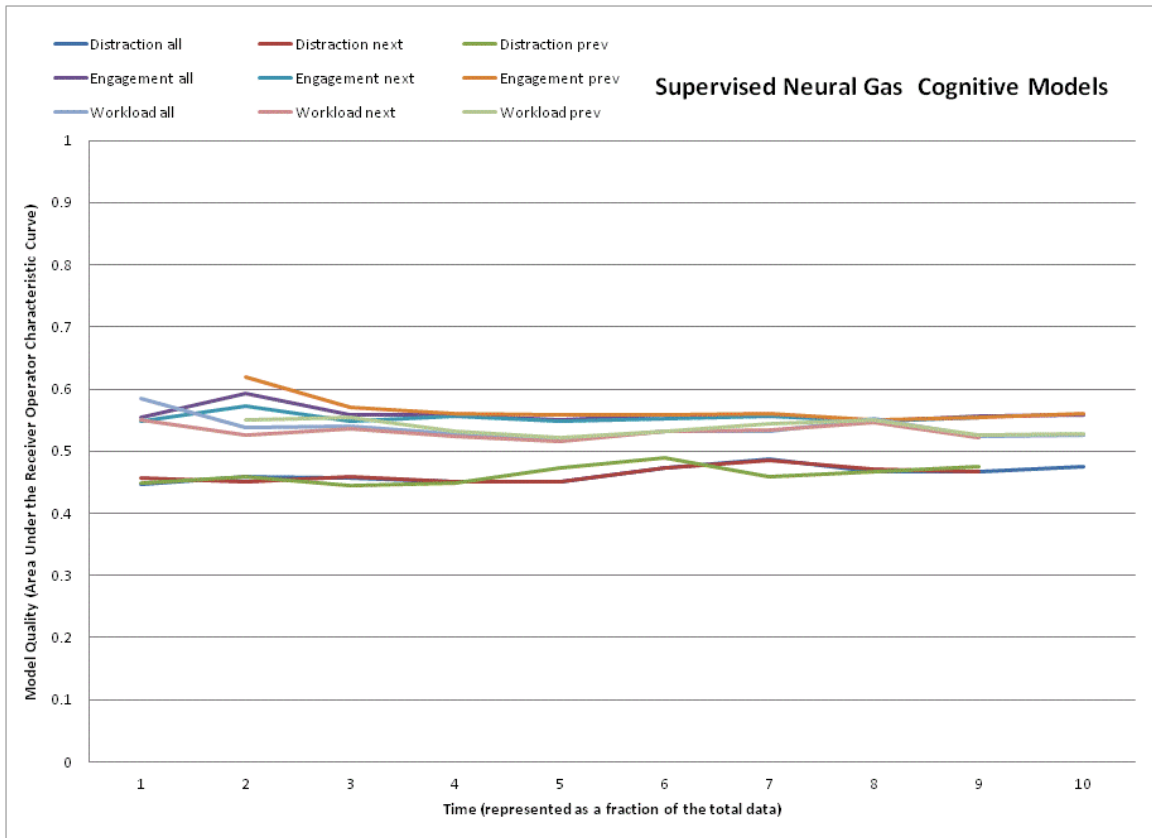
Figure 86 – Performance of supervised Growing Neural Gas for cognitive modeling
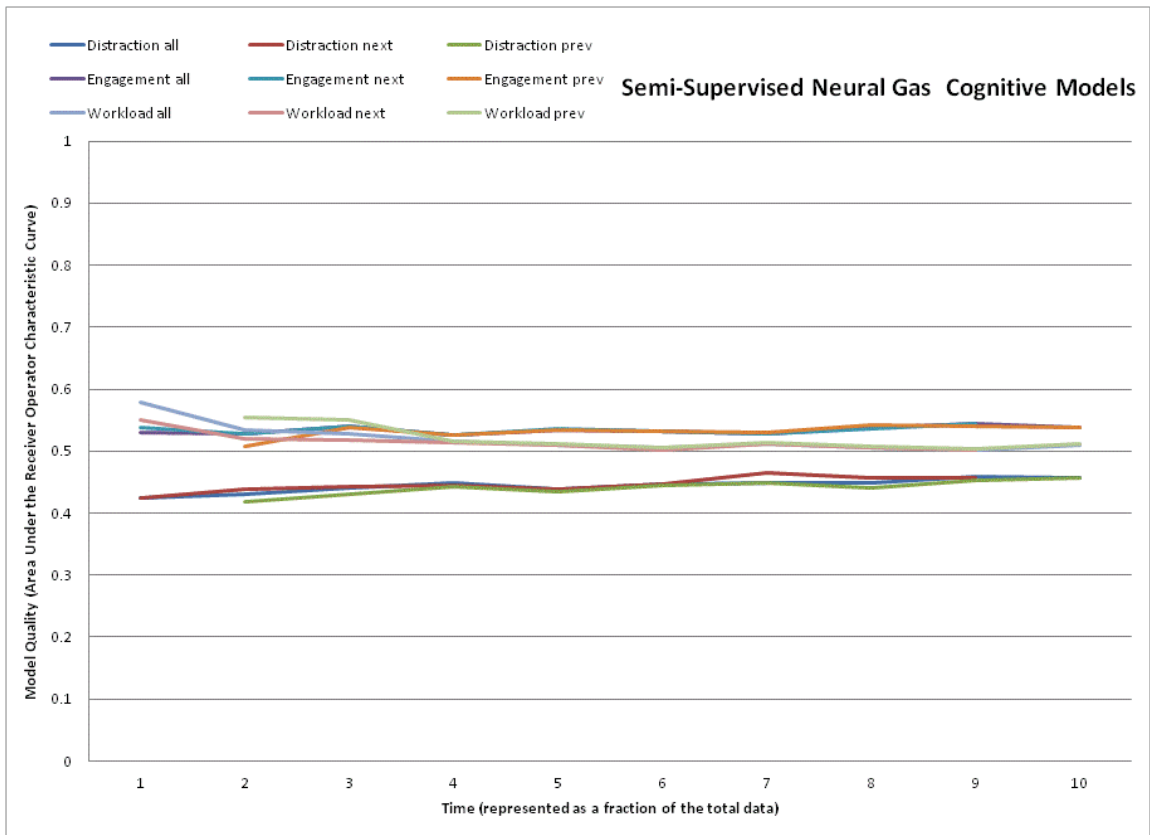
Figure 87 – Performance of semi-supervised Growing Neural Gas for cognitive modeling
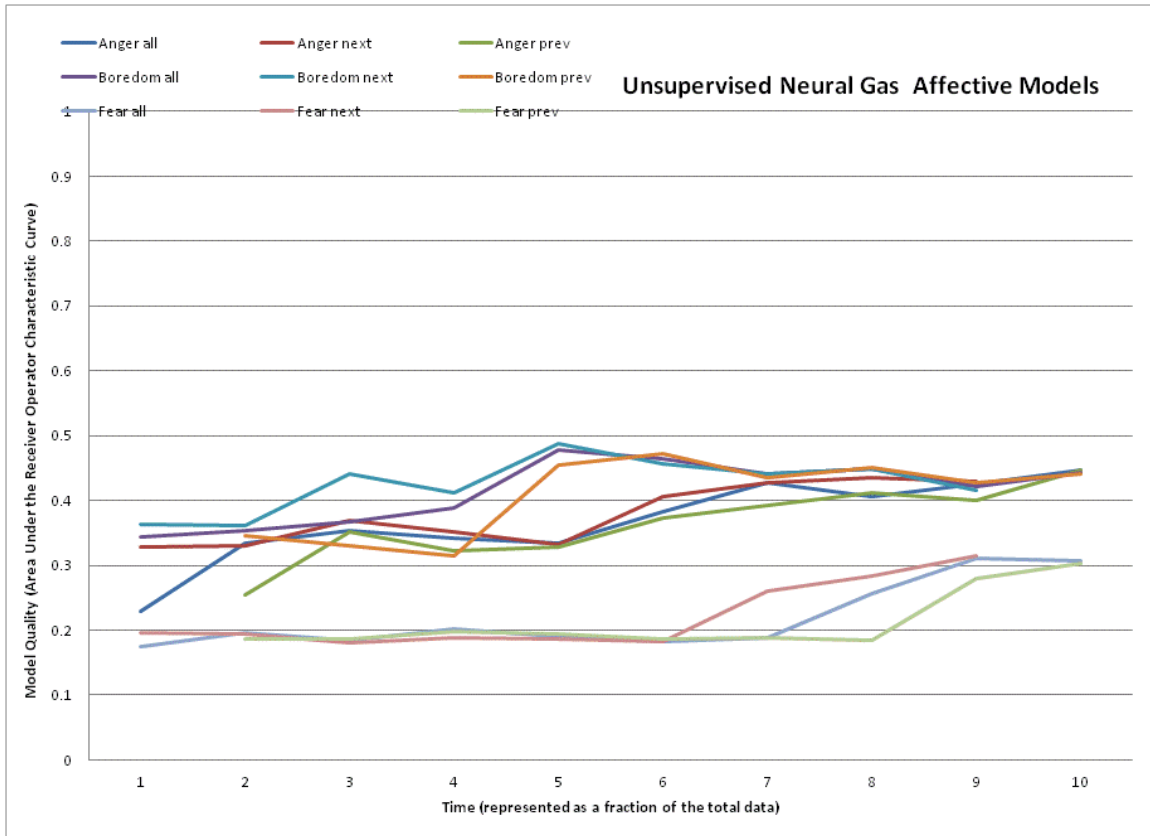
Figure 88 – Performance of unsupervised Growing Neural Gas for affective modeling
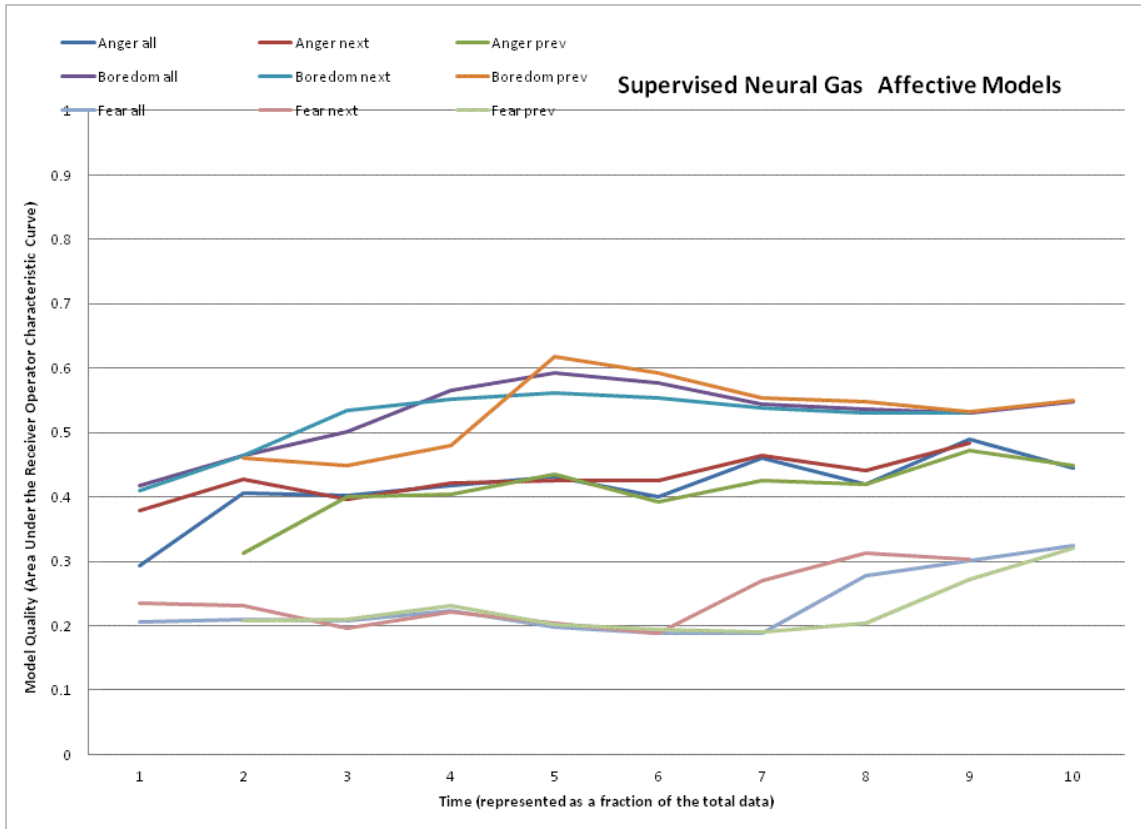
Figure 89 – Performance of supervised Growing Neural Gas for affective modeling
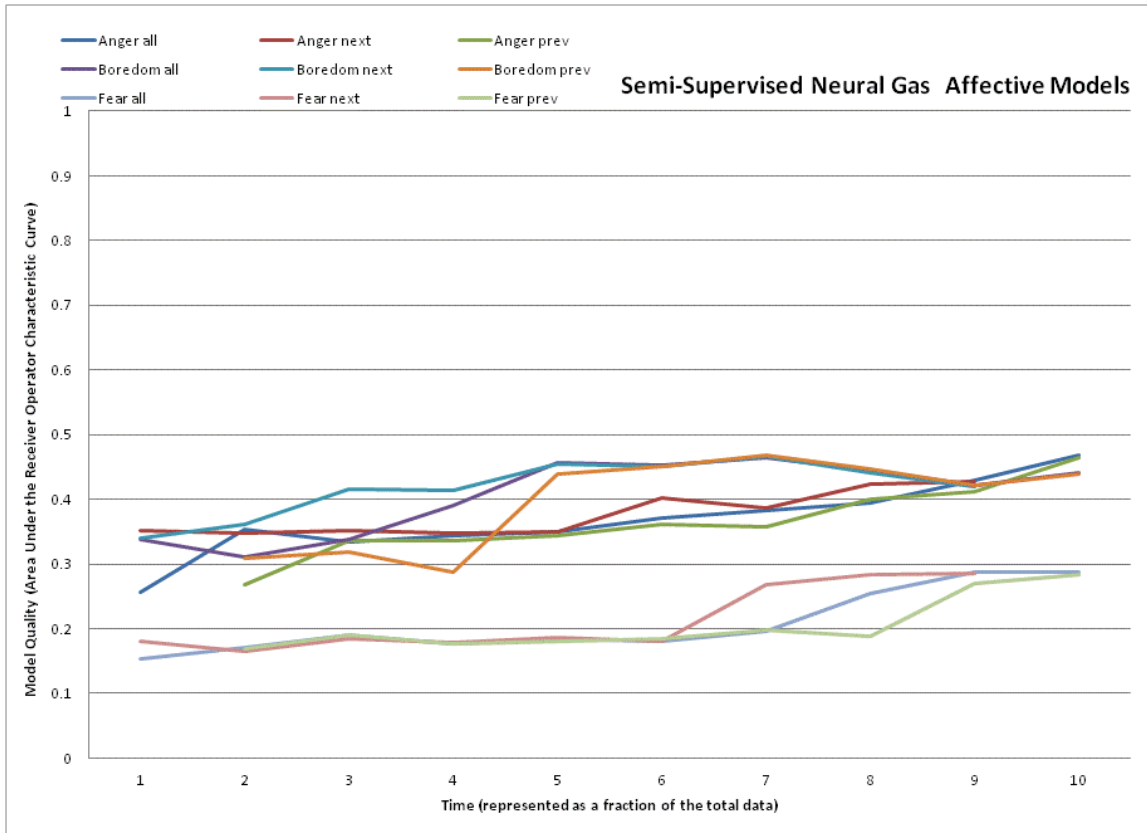
Figure 90 – Performance of semi-supervised Growing Neural Gas for affective modeling

Appendix C-2-4   Vowpal Wabbit (Dataset #1)
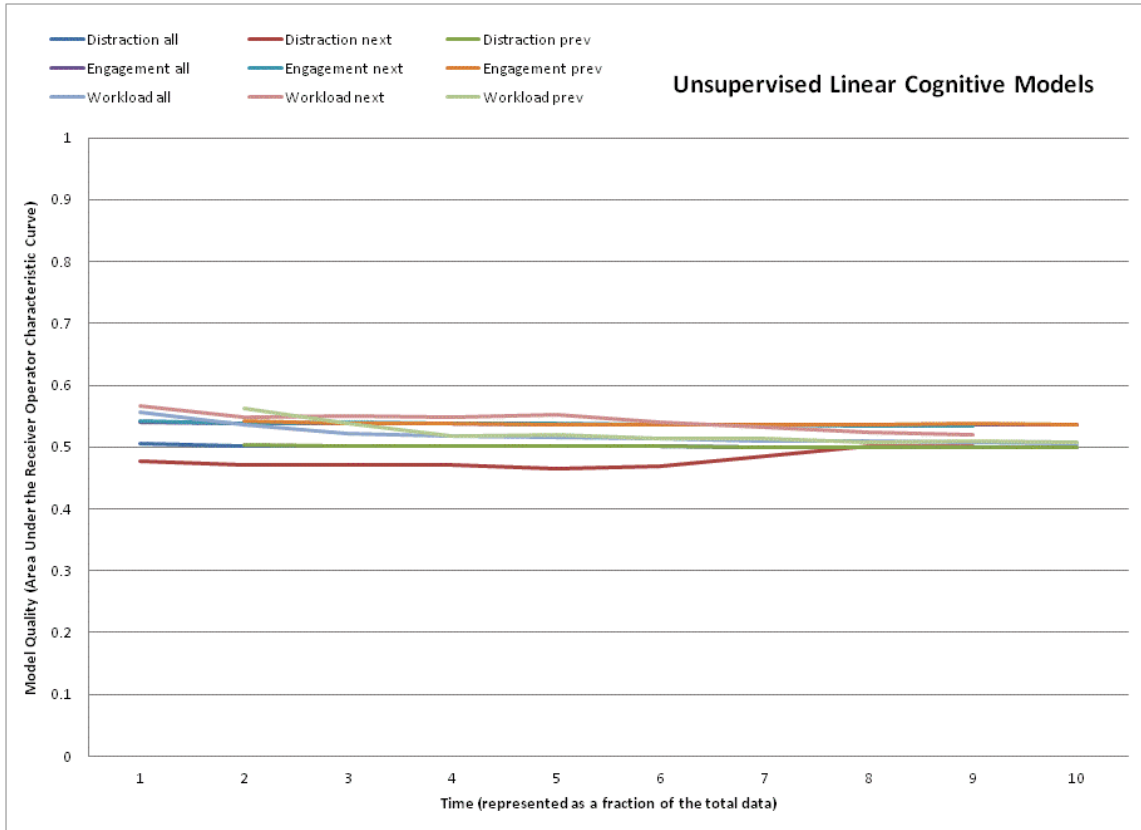
Figure 91 – Performance of unsupervised VW for linear cognitive modeling
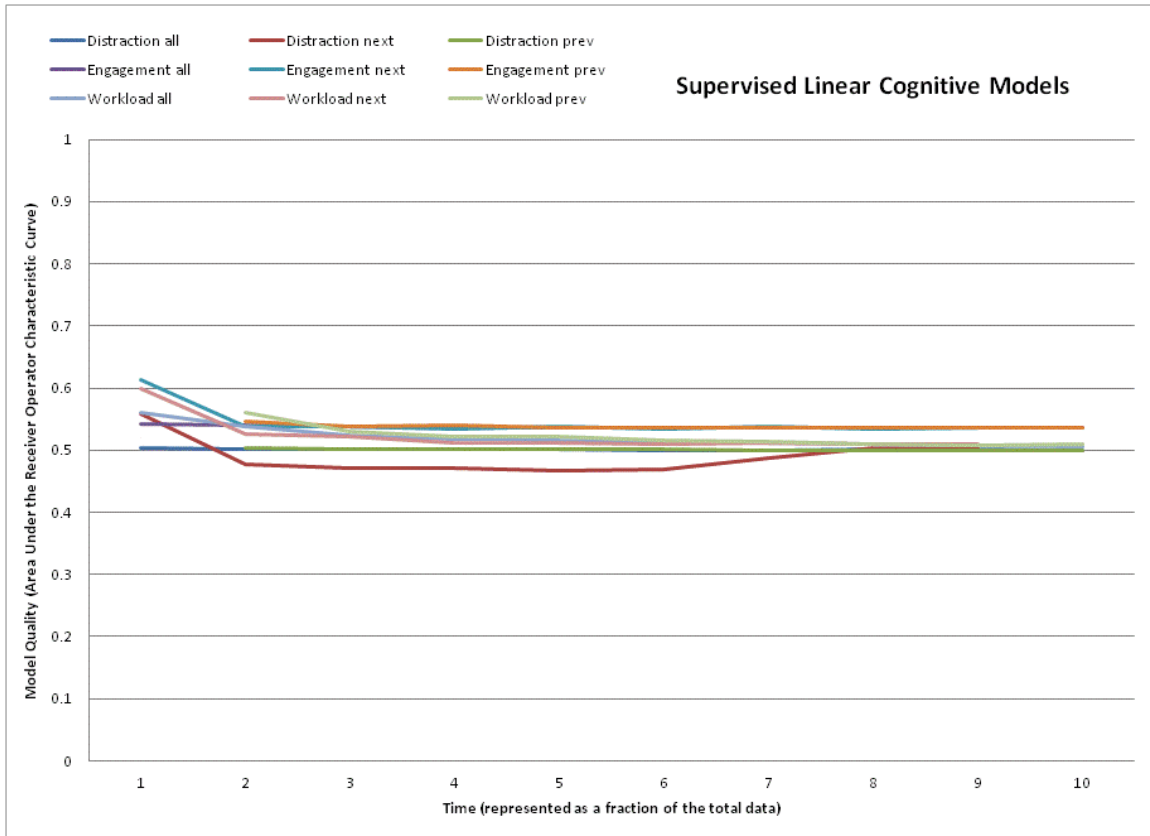
Figure 92 – Performance of supervised VW for linear cognitive modeling
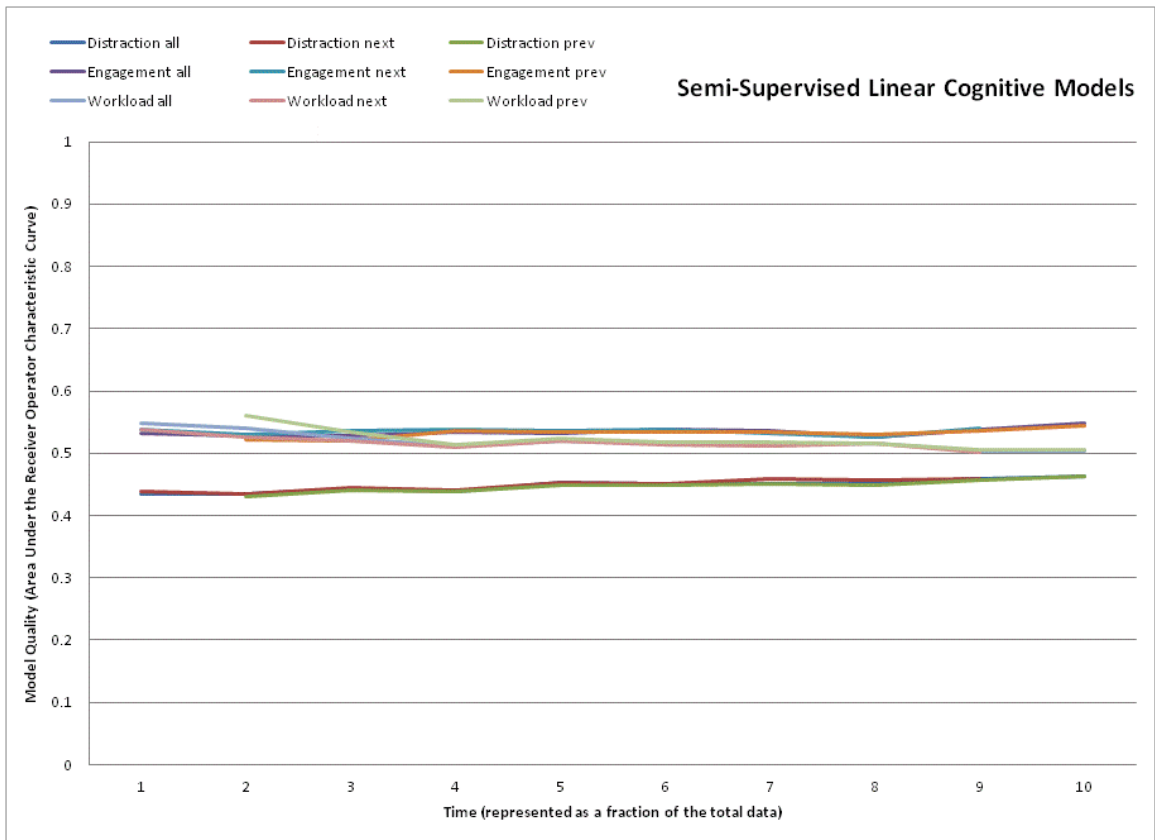
Figure 93 – Performance of semi-supervised VW for linear cognitive modeling

357

Figure 94 – Performance of unsupervised VW for linear affective modeling

358

Figure 95 – Performance of supervised VW for linear affective modeling

Figure 96 – Performance of semi-supervised VW for linear affective modeling

Appendix C-2-5   ART (Dataset #2)
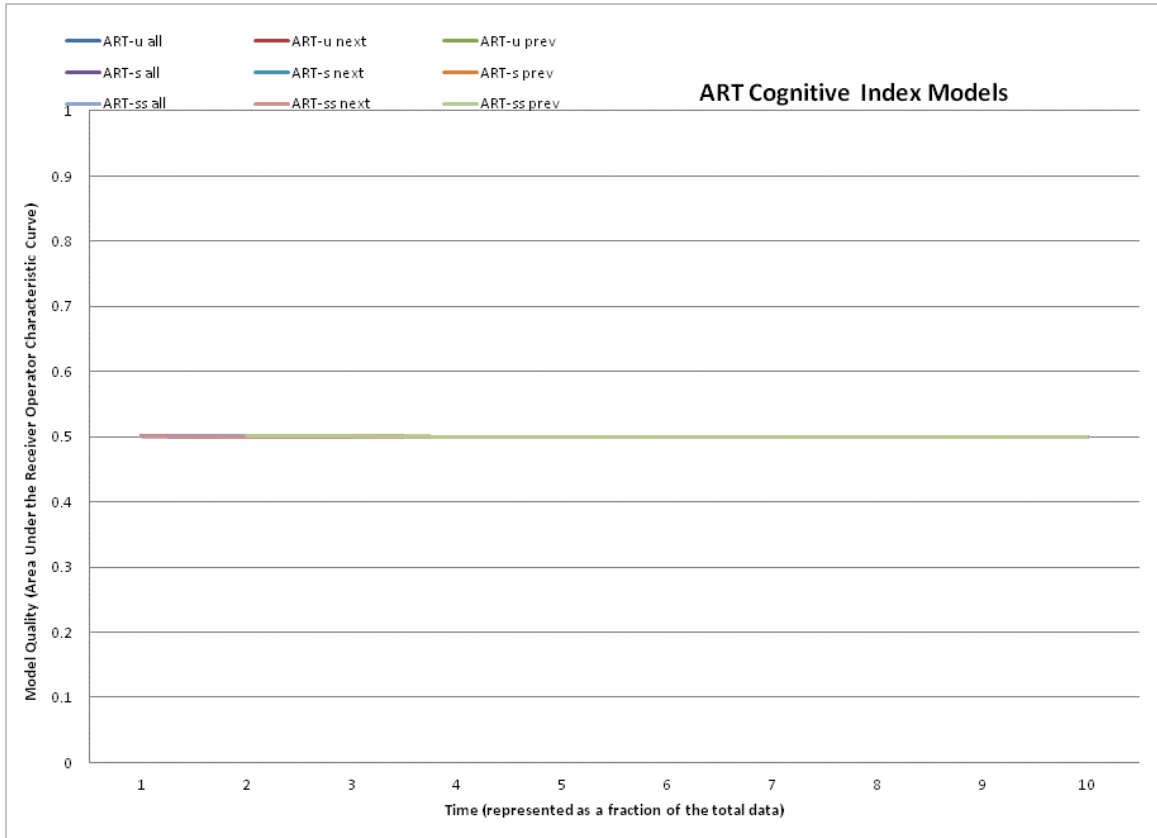
Figure 97 – Performance of ART for cognitive index modeling

Appendix C-2-6   *Growing Neural Gas (Dataset #2)*
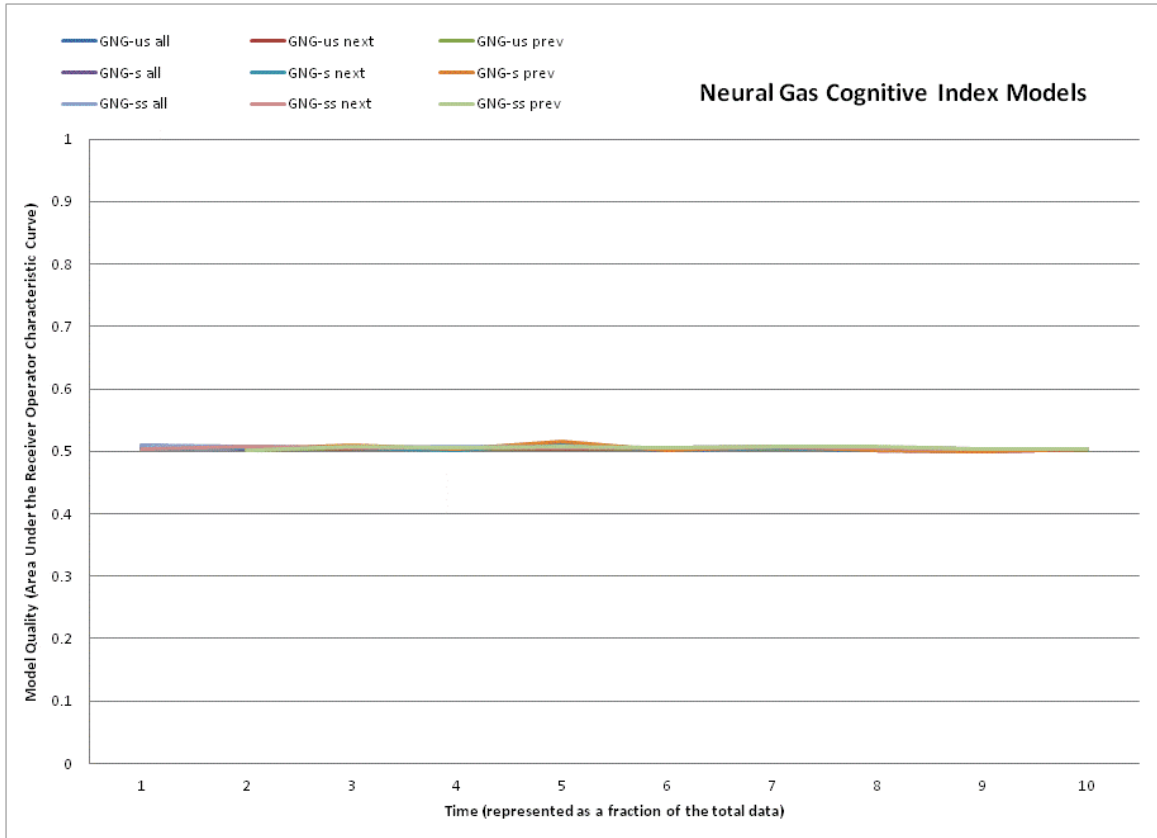
Figure 98 – Performance of GNG for cognitive index modeling

Appendix C-2-7   Vowpal Wabbit (Dataset #2)
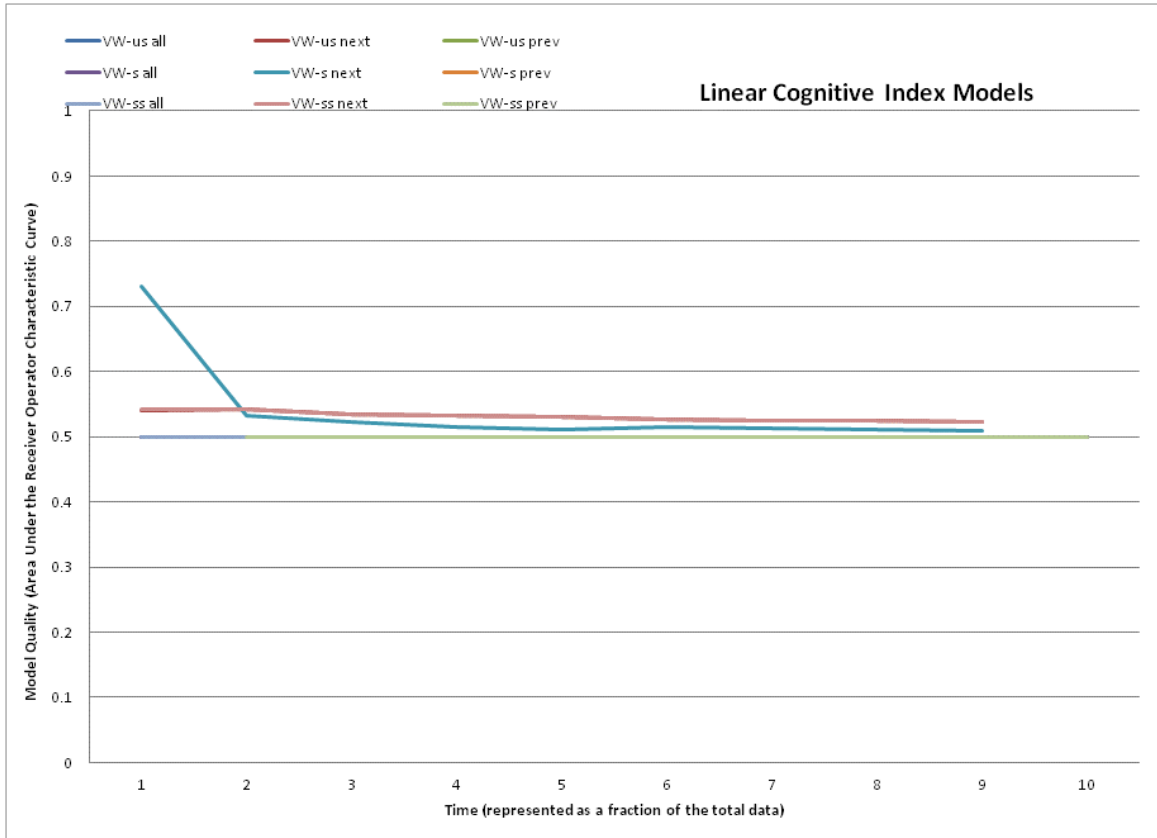
Figure 99 – Performance of VW for cognitive index modeling

Figure 100 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for cognitive modeling
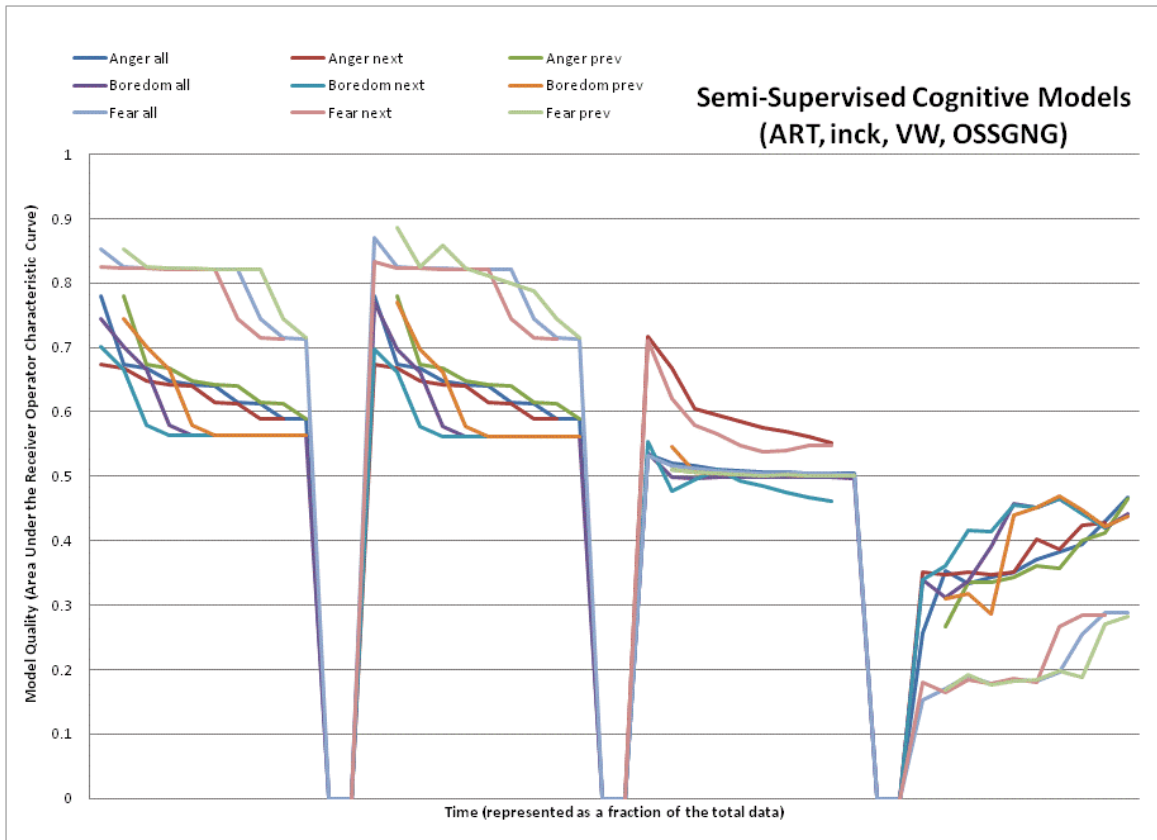
Figure 101 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for affective modeling

Appendix C-2-9   Total Results Set #2 Semi-Supervised Modeling Ability

(Dataset #2)
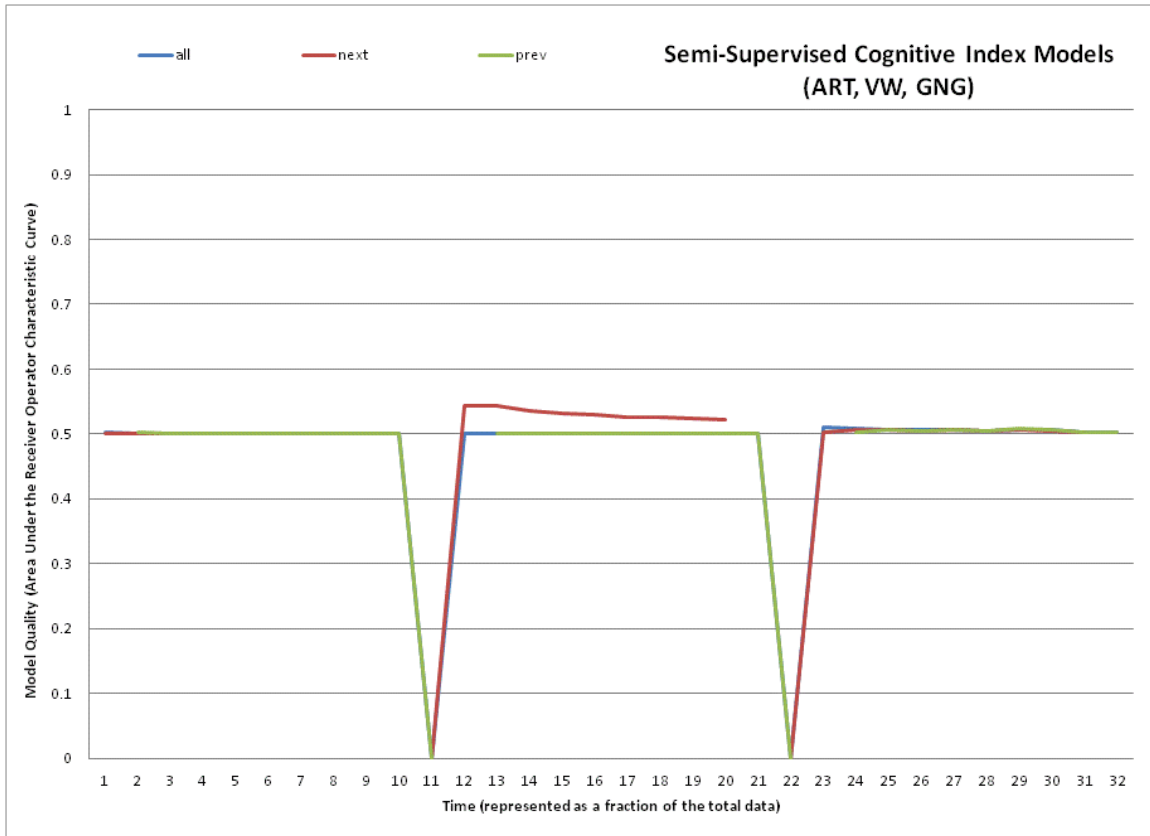
Figure 102 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for cognitive

index modeling

Appendix C-3   Results Set #3

The results in this section will be presented similar to the previous section. It will be broken into a section for the algorithm, the method of label assignment, and the type of model created. In each of these results graphs, the measures of classification quality, previous model quality, and predictive accuracy for each of the model types is shown. Results Set #3 differs from Results Set #1 and #2 in that the created affective and cognitive models were given a significantly reduced input feature set, as found in the previous research study.

Figure 103 – Performance of unsupervised ART for cognitive modeling

Figure 104 – Performance of supervised ART for cognitive modeling

369

Figure 105 – Performance of semi-supervised ART for cognitive modeling

Figure 106 – Performance of unsupervised ART for affective modeling

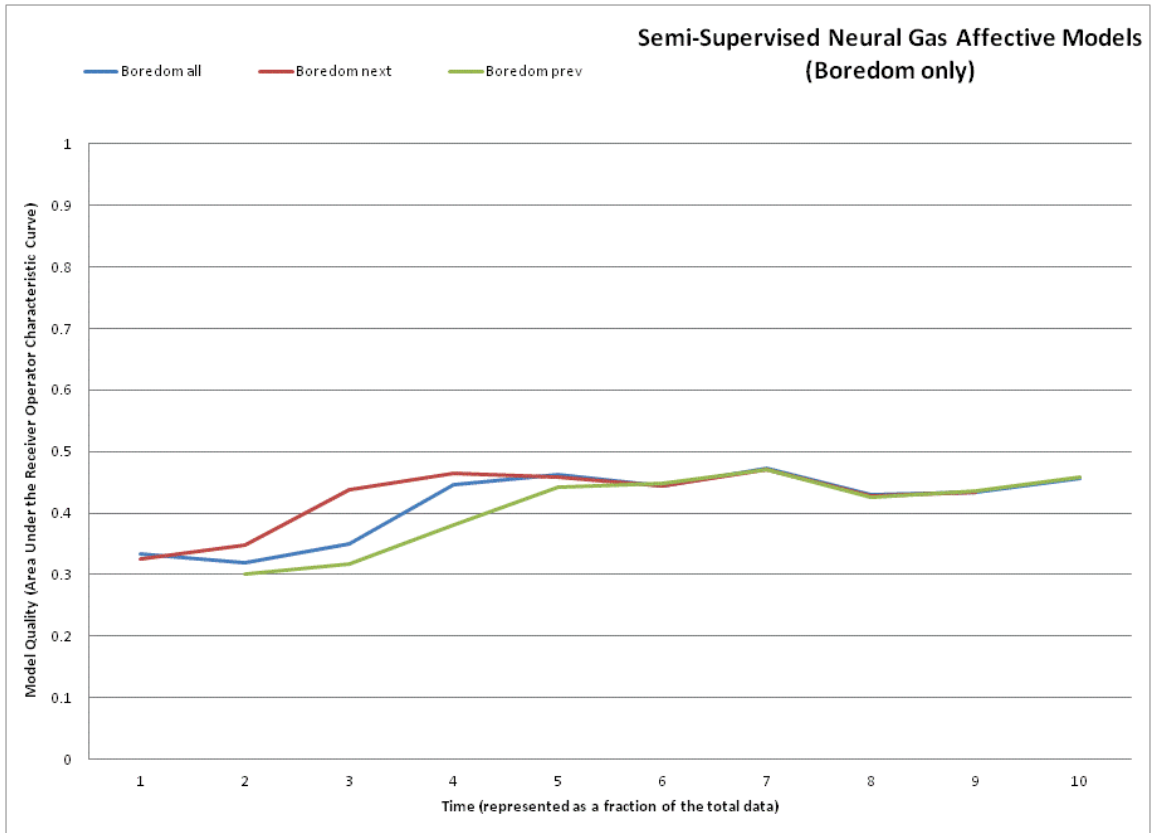**Figure 107 – Performance of supervised ART for affective modeling**

Figure 108 – Performance of semi-supervised ART for affective modeling

Appendix C-3-2   K-Means (Dataset #1)

Figure 109 – Performance of unsupervised K-Means clustering for cognitive modeling

Figure 110 – Performance of supervised K-Means clustering for cognitive modeling
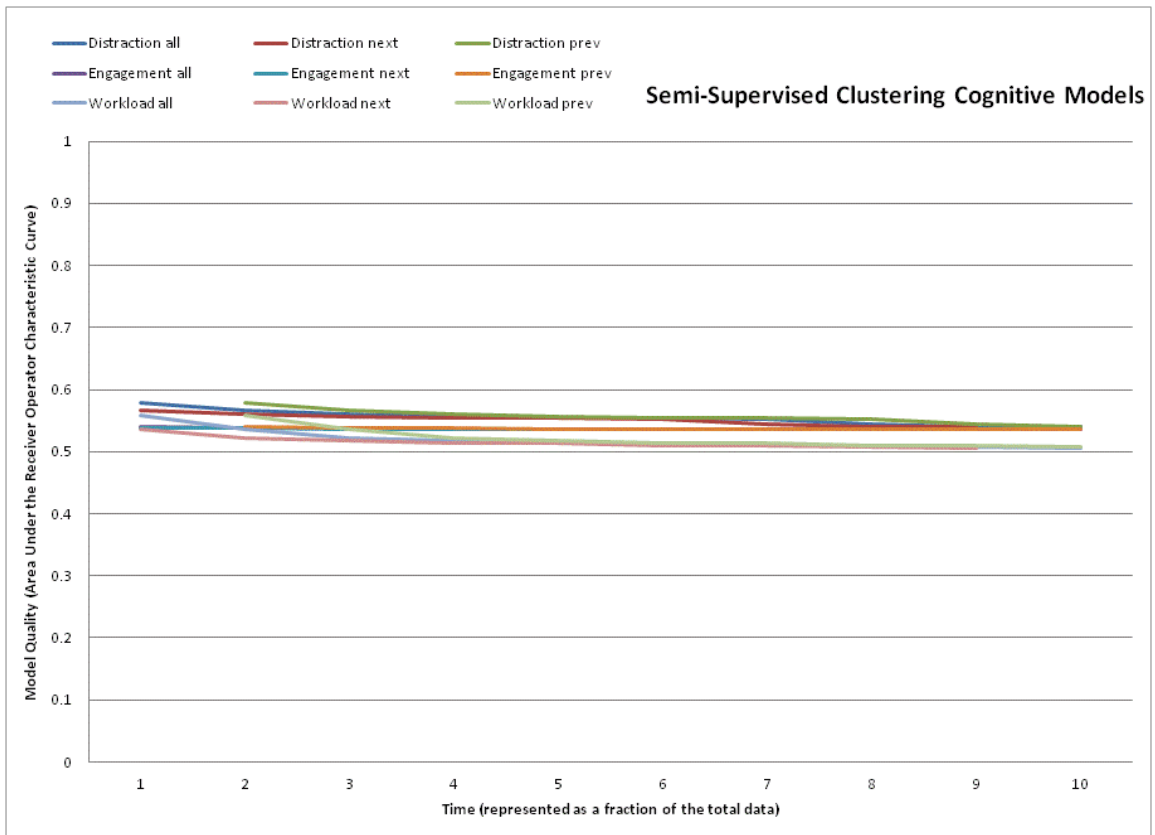
Figure 111 – Performance of semi-supervised K-Means clustering for cognitive modeling
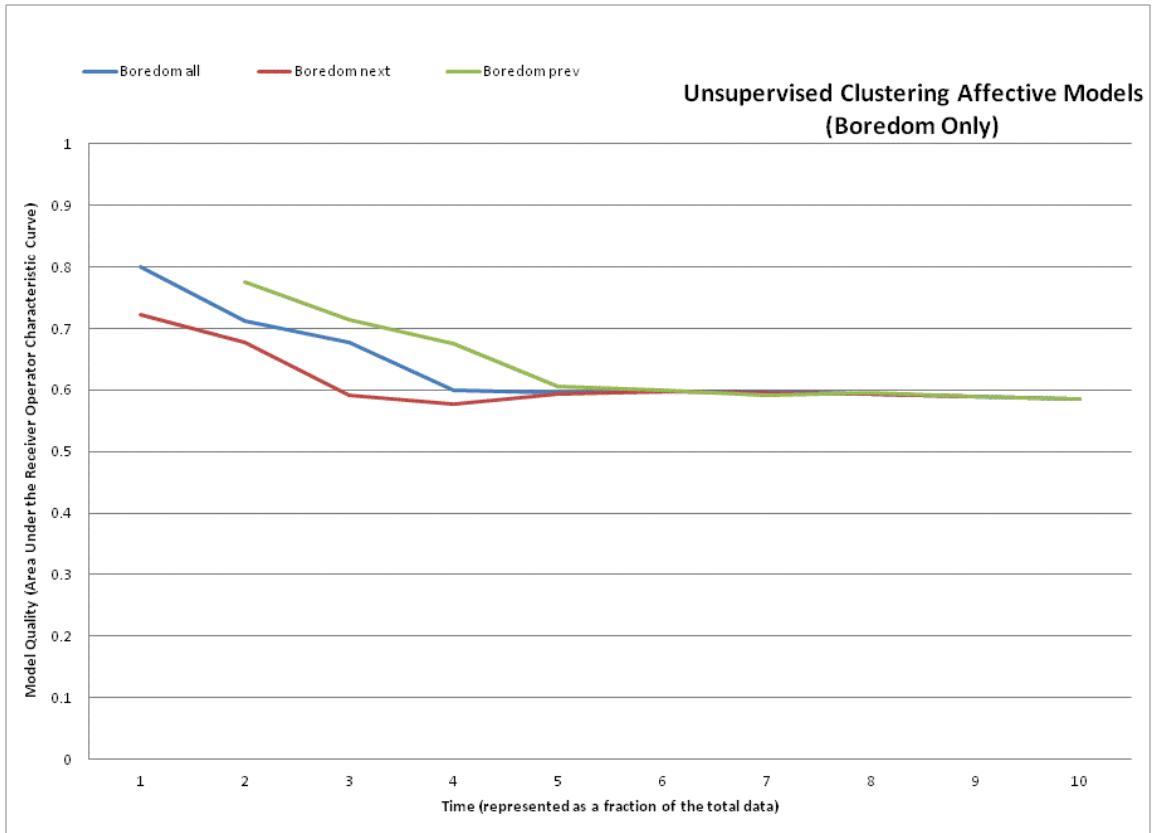
Figure 112 – Performance of unsupervised K-Means clustering for affective modeling
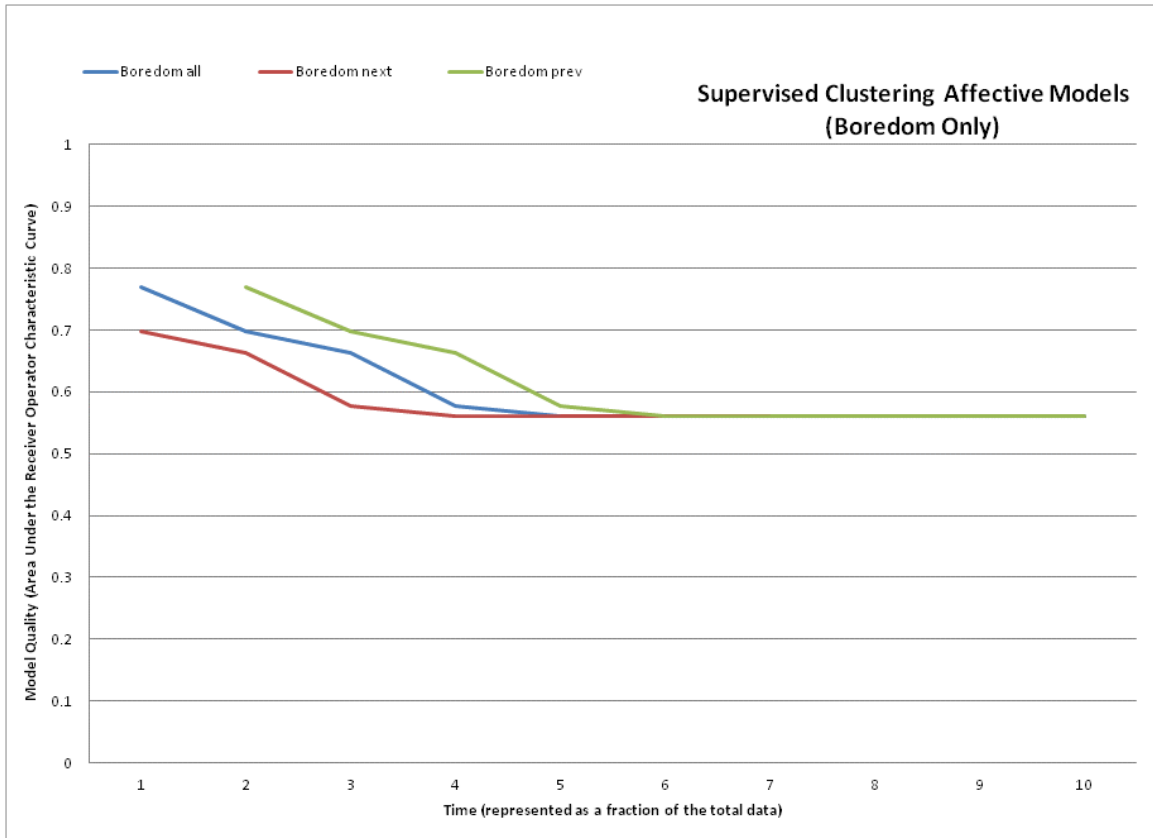
Figure 113 – Performance of supervised K-Means clustering for affective modeling
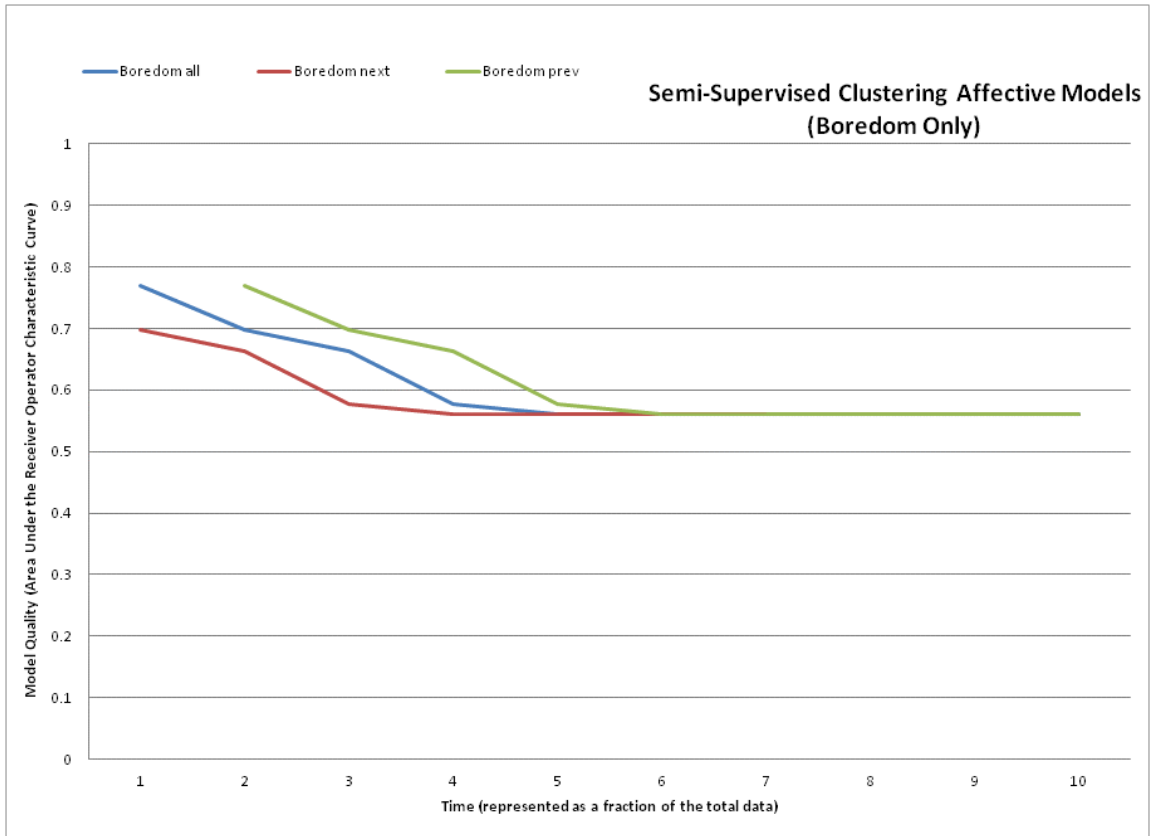
Figure 114 – Performance of semi-supervised K-Means clustering for affective modeling
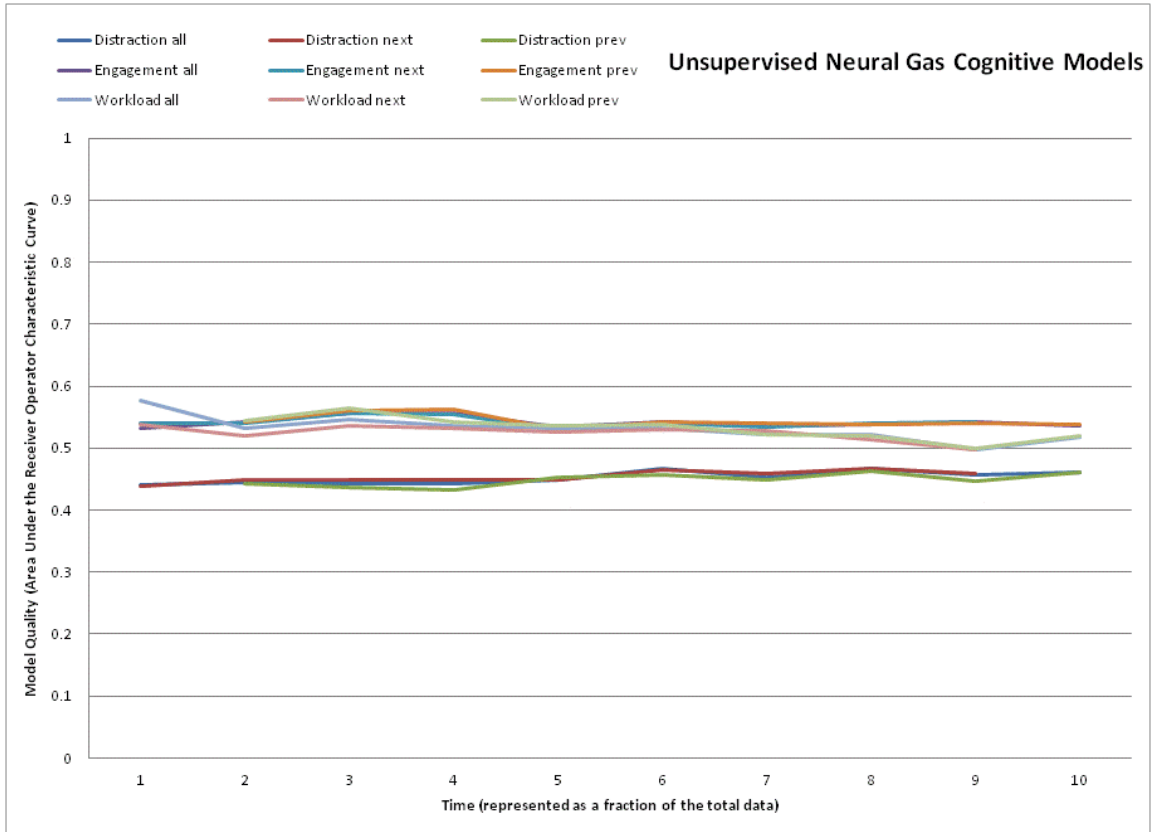
Figure 115 – Performance of unsupervised Growing Neural Gas for cognitive modeling
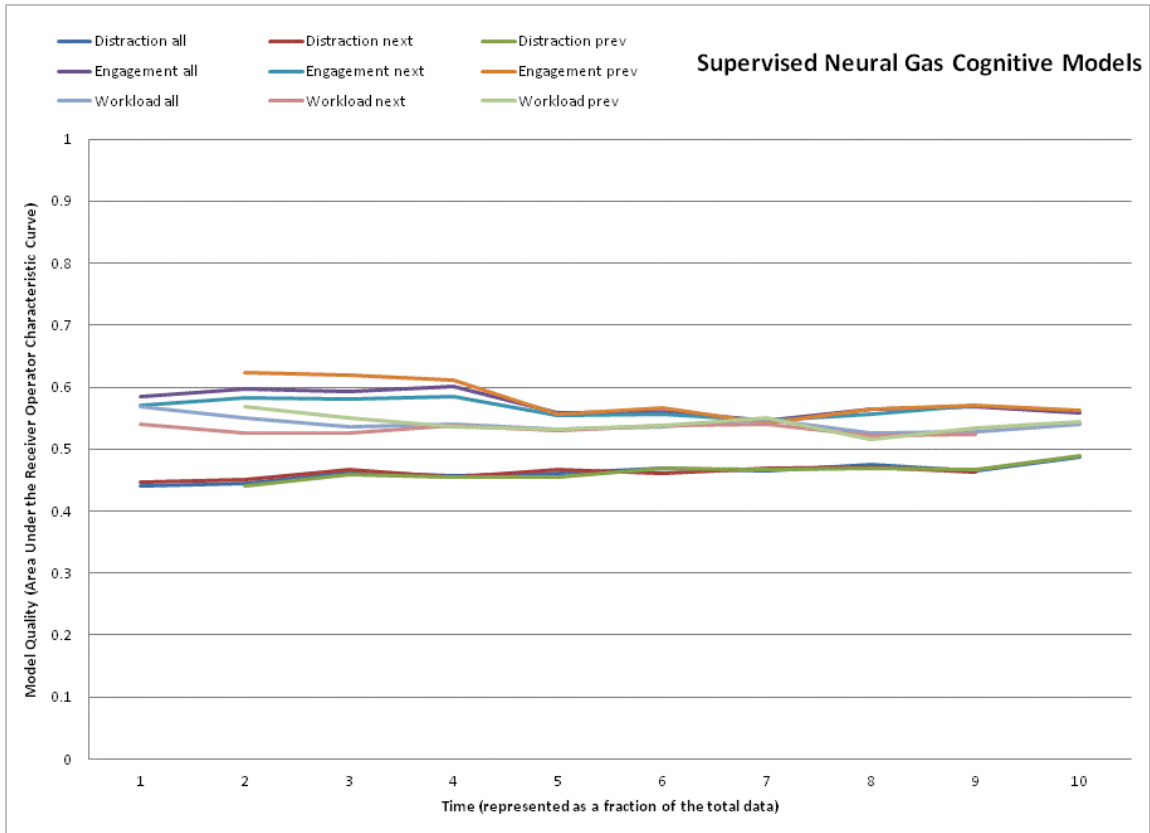
Figure 116 – Performance of supervised Growing Neural Gas for cognitive modeling
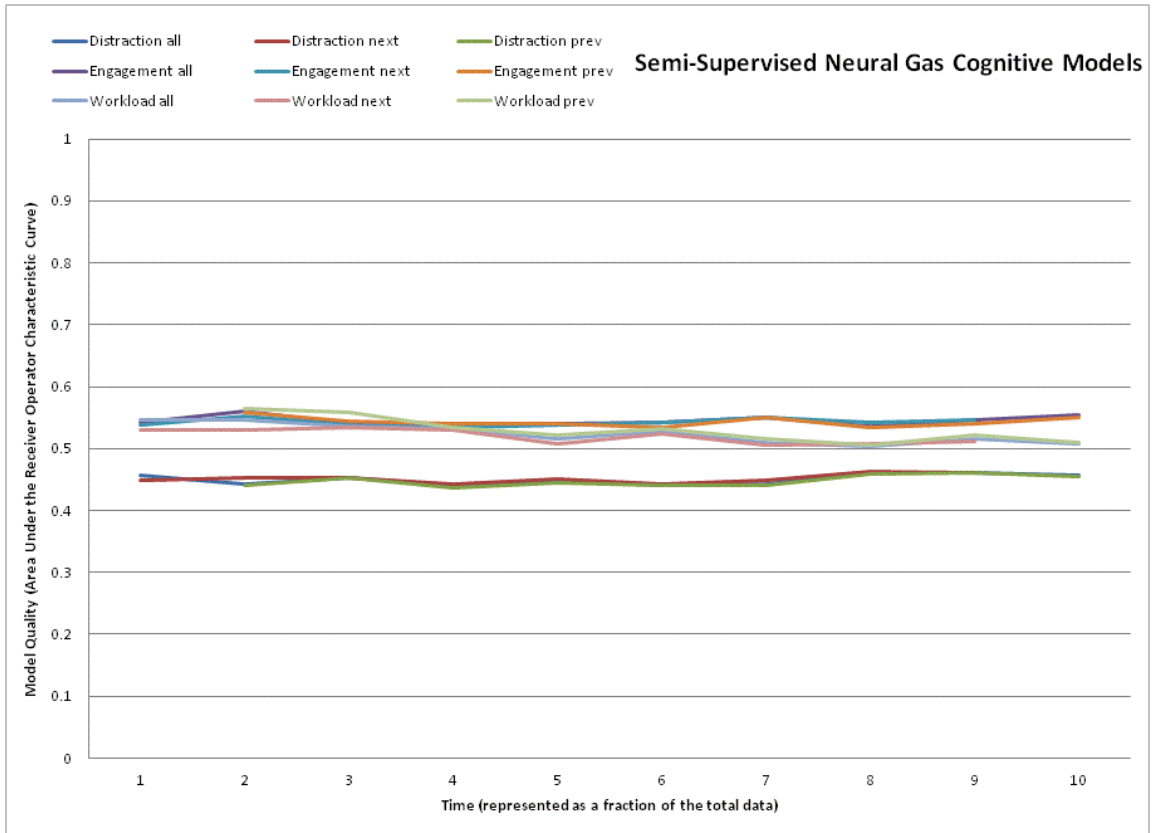
Figure 117 – Performance of semi-supervised Growing Neural Gas for cognitive modeling
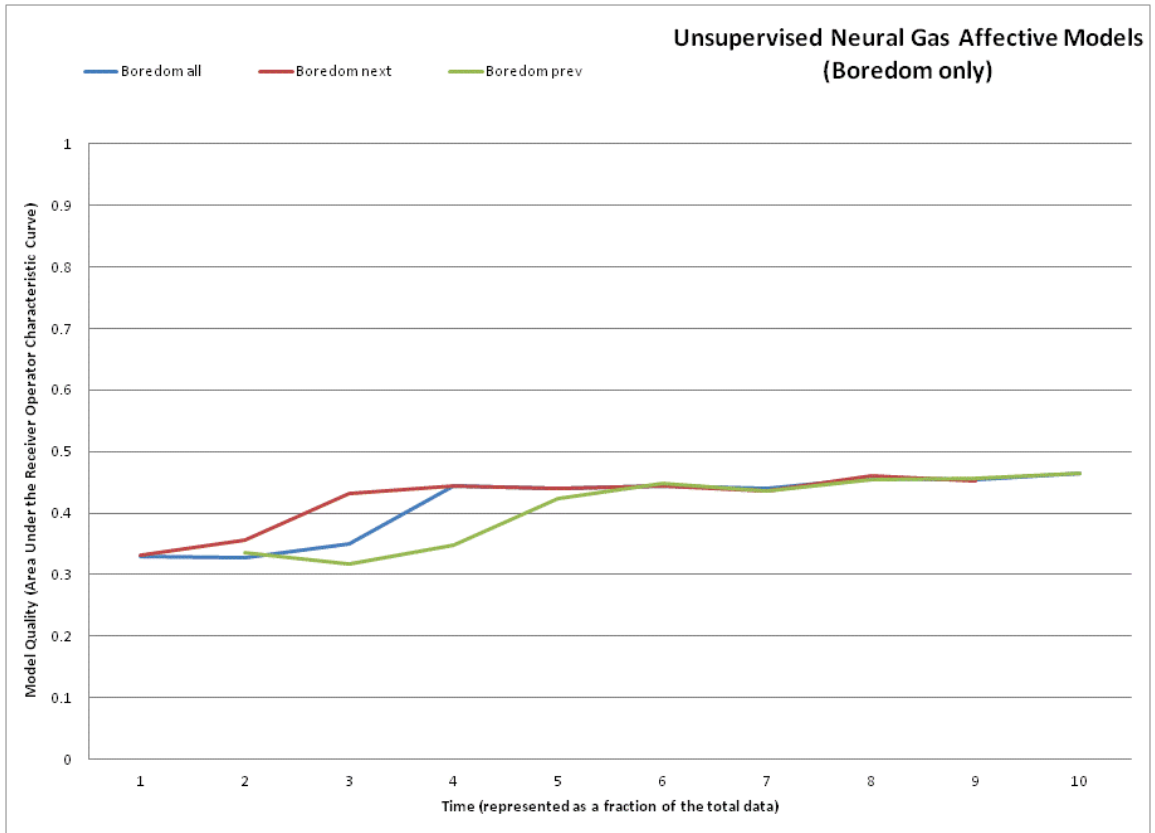
Figure 118 – Performance of unsupervised Growing Neural Gas for affective modeling
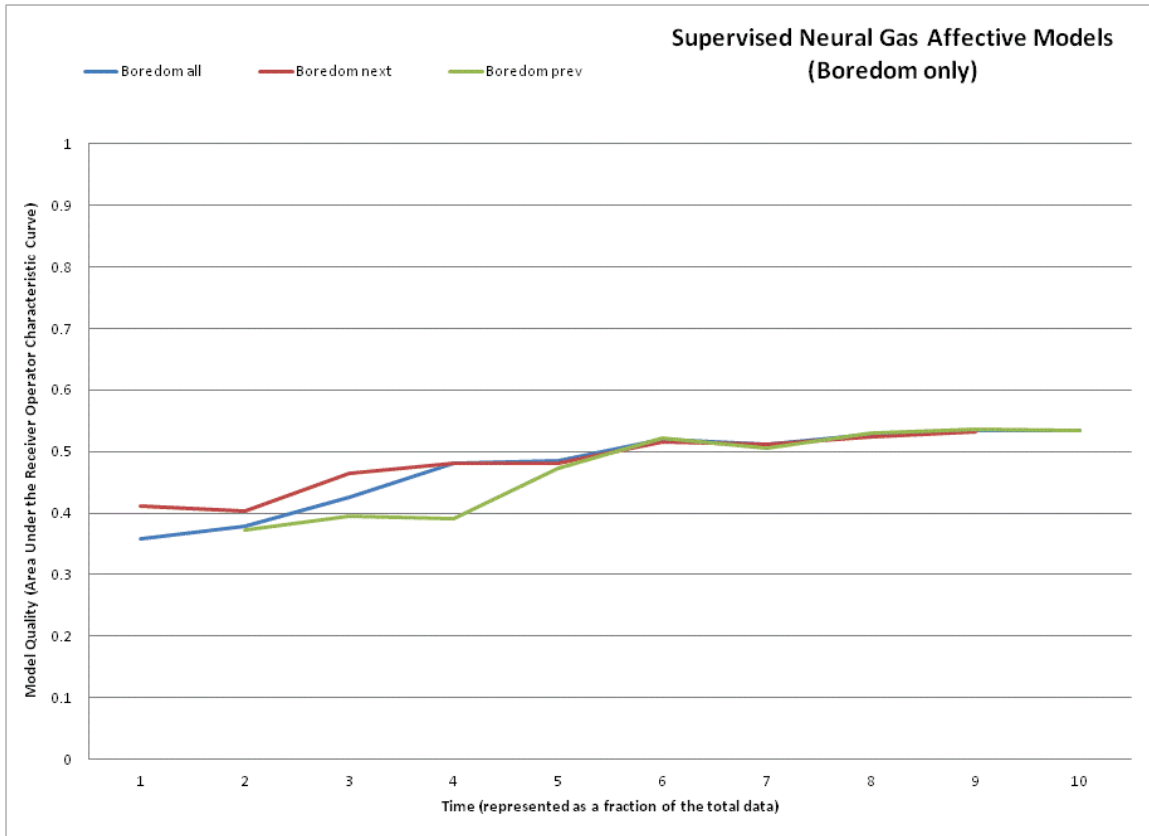
Figure 119 – Performance of supervised Growing Neural Gas for affective modeling

Figure 120 – Performance of semi-supervised Growing Neural Gas for affective modeling
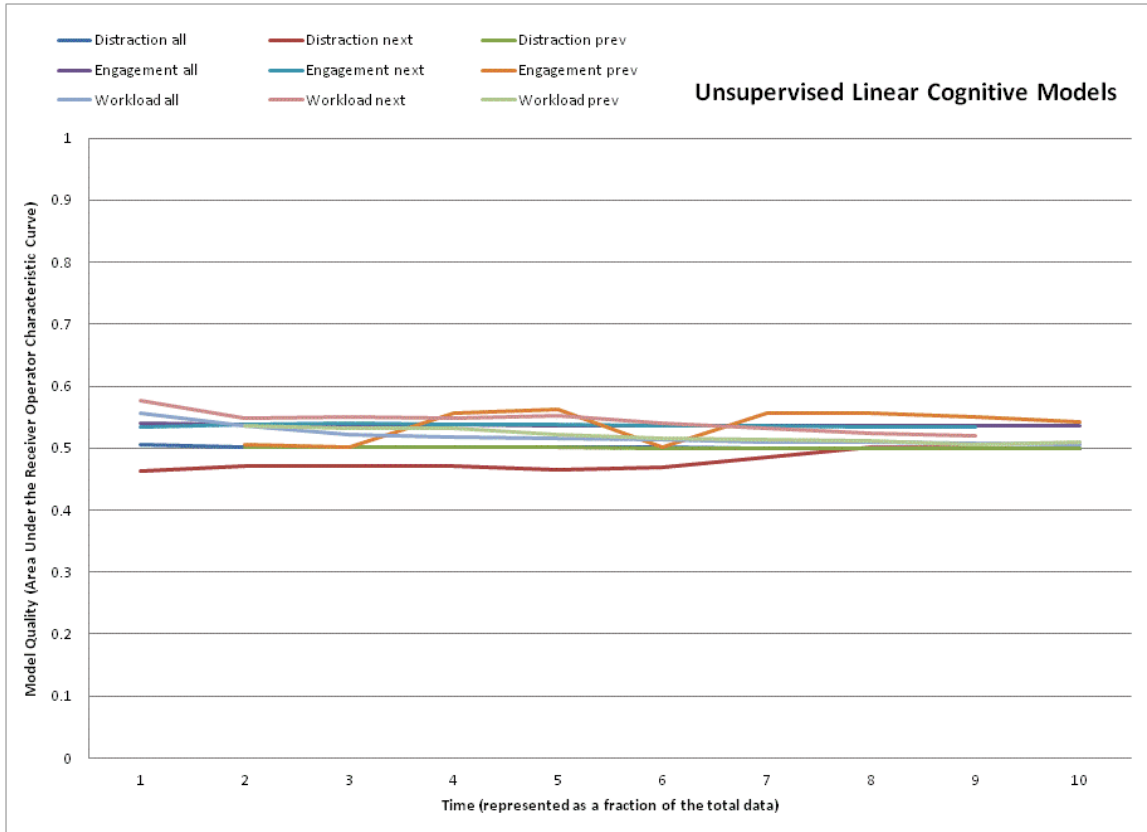
Figure 121 – Performance of unsupervised VW for linear cognitive modeling
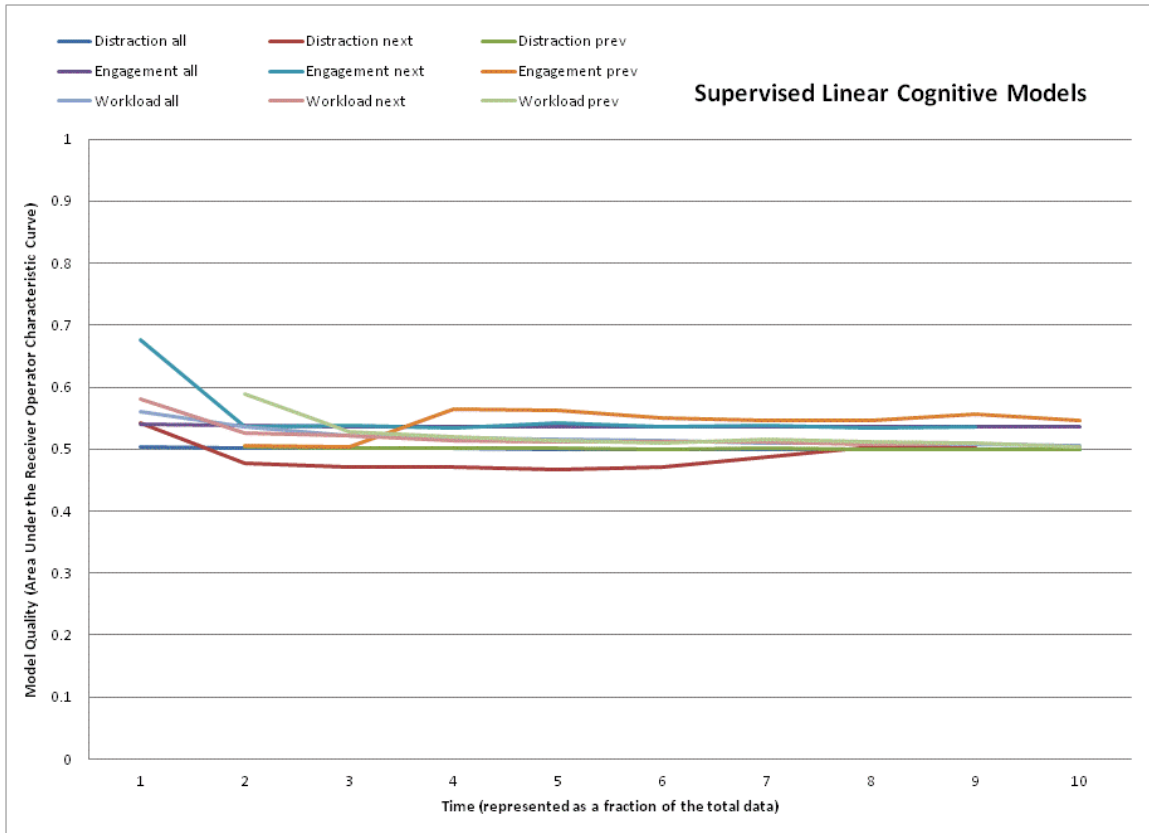
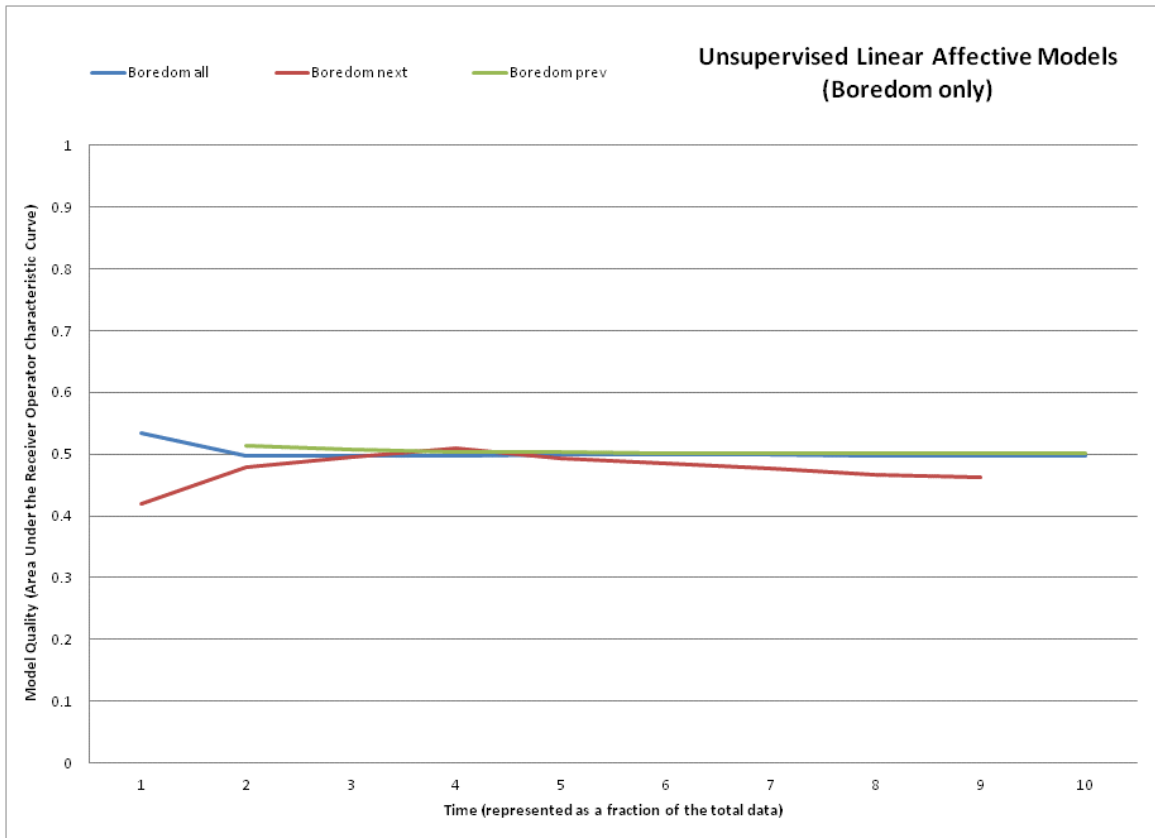**Figure 122 – Performance of supervised VW for linear cognitive modeling**
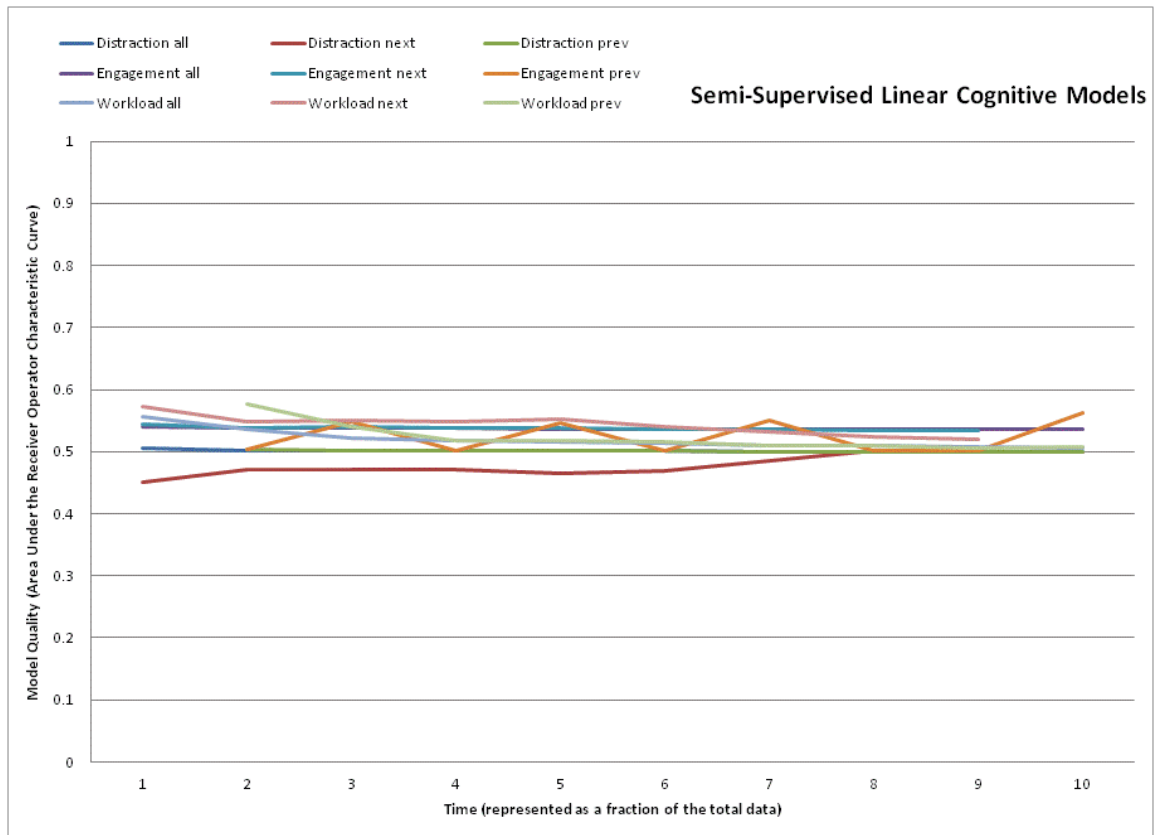
Figure 123 – Performance of semi-supervised VW for linear cognitive modeling

388

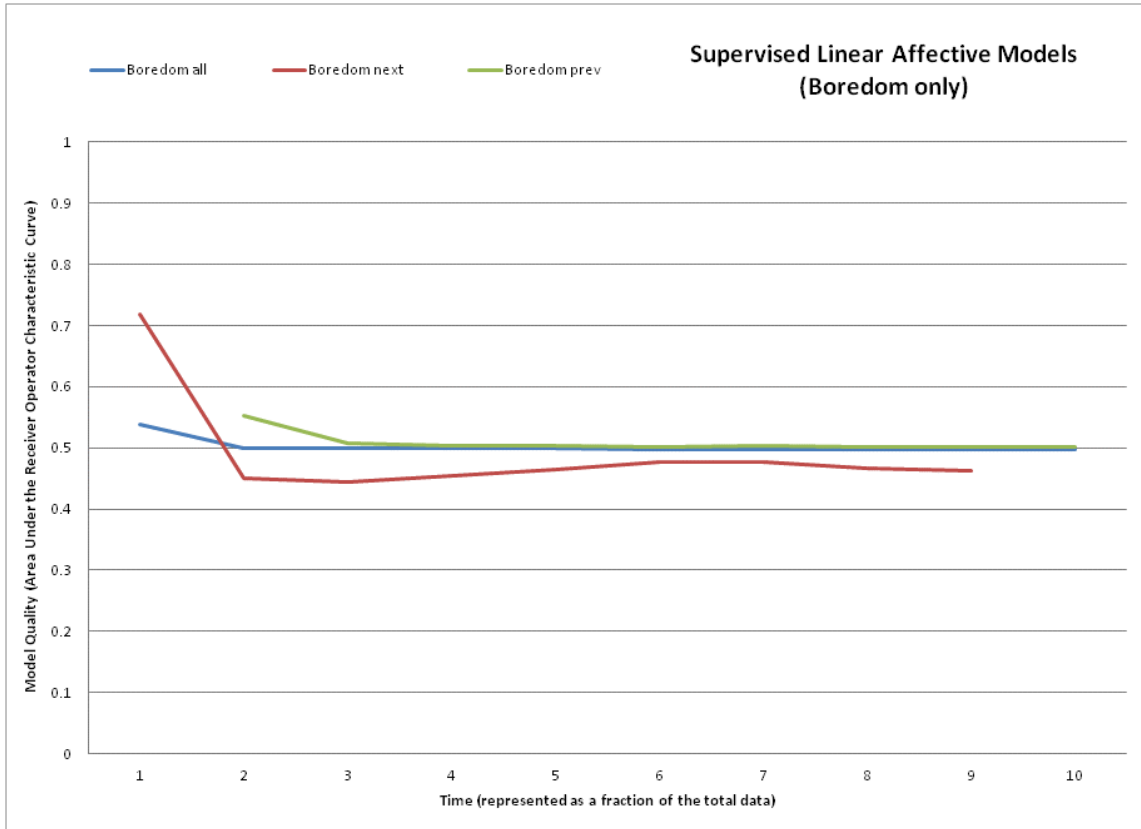**Figure 124 – Performance of unsupervised VW for linear affective modeling**

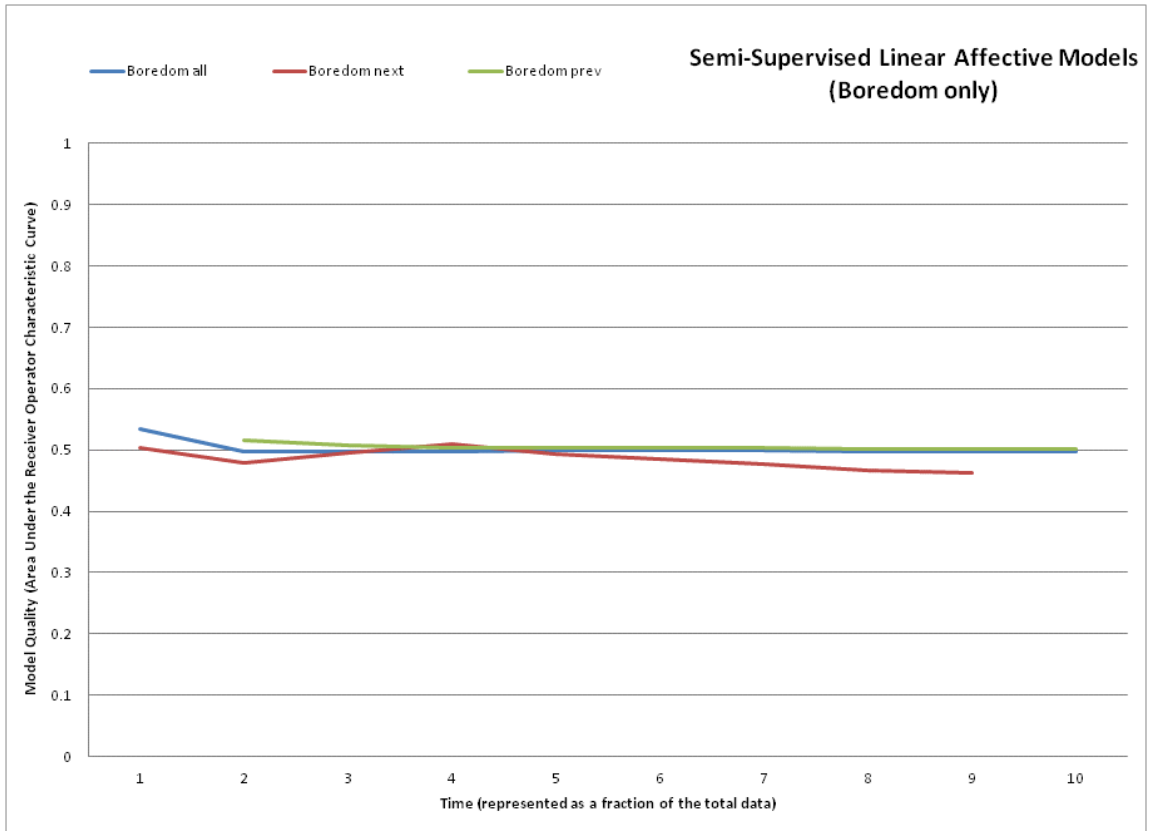**Figure 125 – Performance of supervised VW for linear affective modeling**

Figure 126 – Performance of semi-supervised VW for linear affective modeling
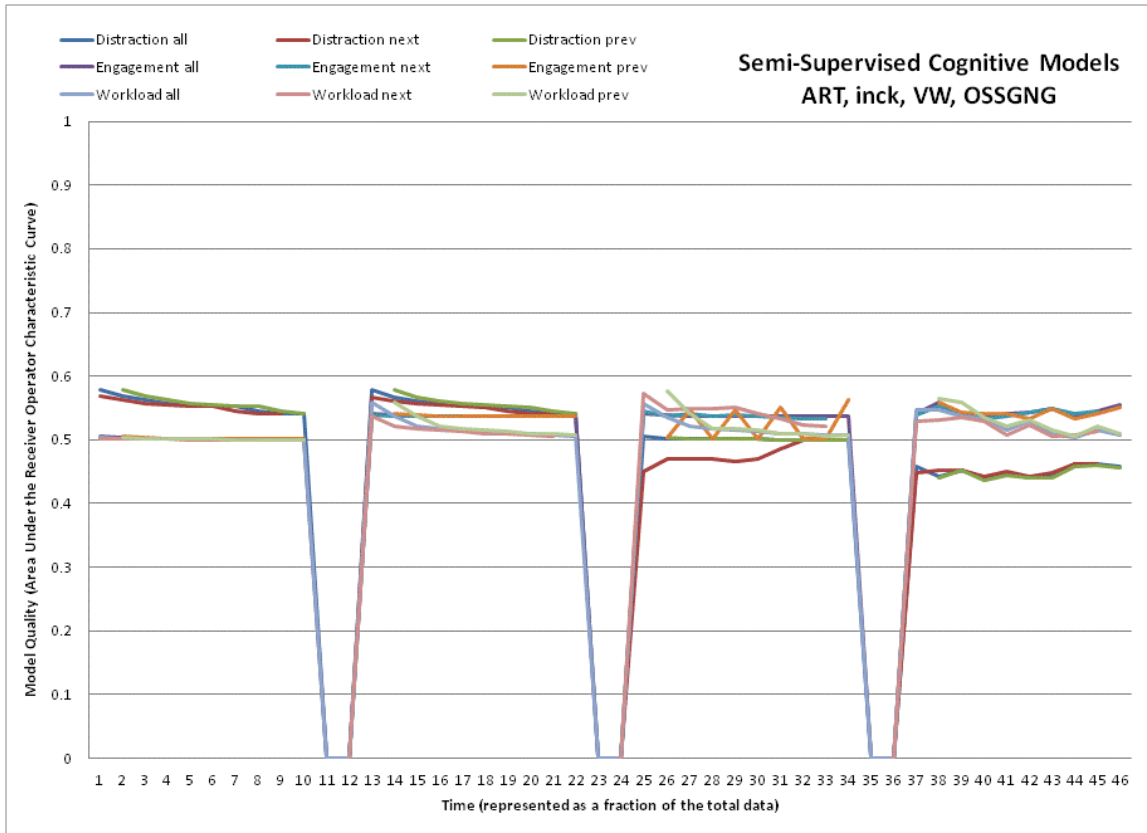
Figure 127 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for cognitive modeling
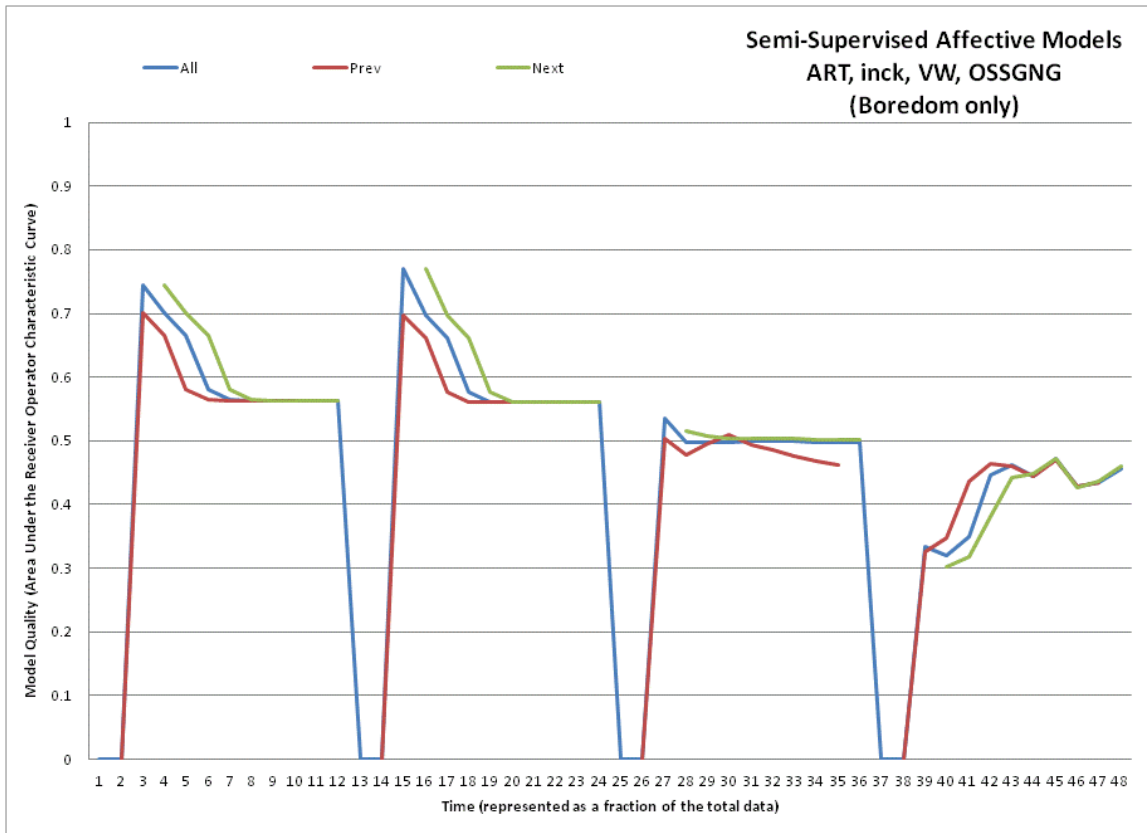
Figure 128 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for affective

modeling

Appendix C-4   Results Set #4

The results in this section will be presented similar to the previous sections. It will be broken into a section for the algorithm, the method of label assignment, and the type of model created. In each of these results graphs, the measures of classification quality, previous model quality, and predictive accuracy for each of the model types is shown. Results Set #4 differs from Results Set #1, #2, and #3 in so far as 22 new features were introduced into the dataset. This was performed through incorporation of a 5-second average of each of the previous 21 features, resulting in 42 total features.

It is not appropriate to compare models created on this new dataset directly to models produced with the other datasets, but was performed to shed light on whether a simple historical statistical measure introduced into the datastream would be enough to stabilize models of cognition or produce superior models of affect.   This was not observed in comparisons of the C-2-1 appendix to the C-1-1 appendix.
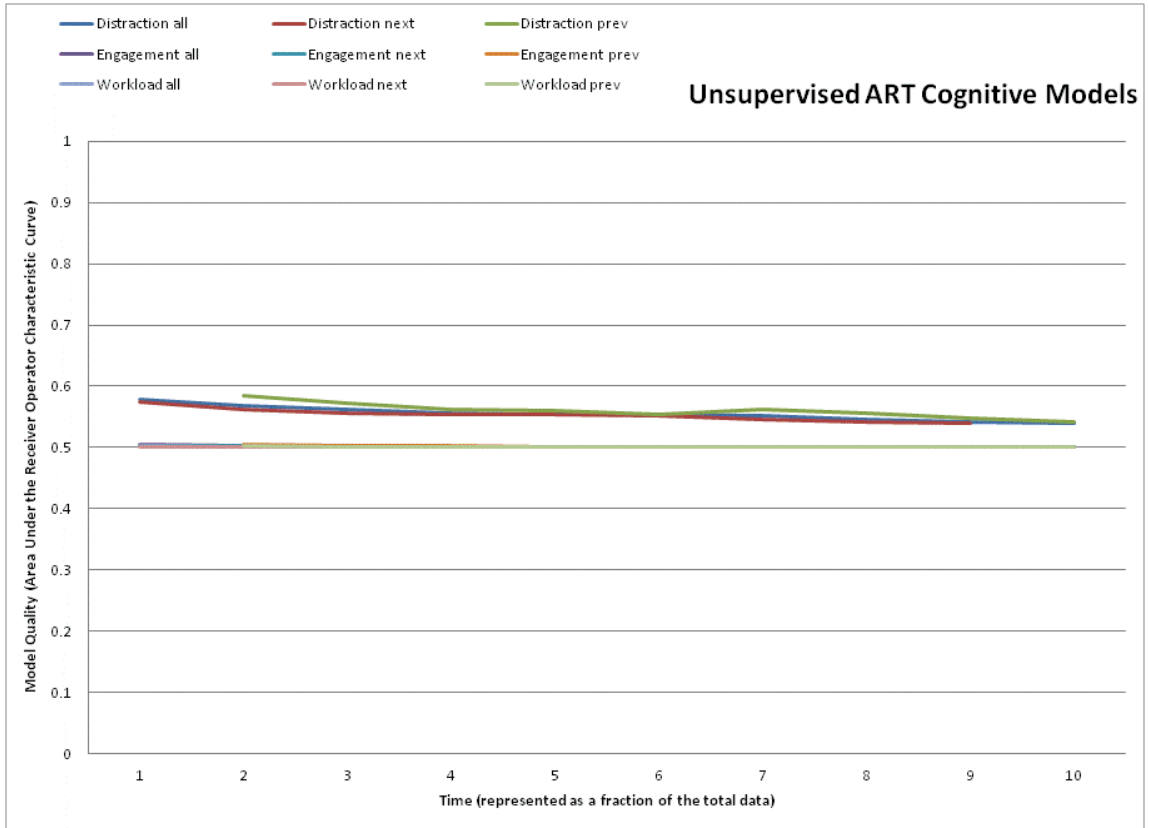
Figure 129 – Performance of unsupervised ART for cognitive modeling for Results Set #4
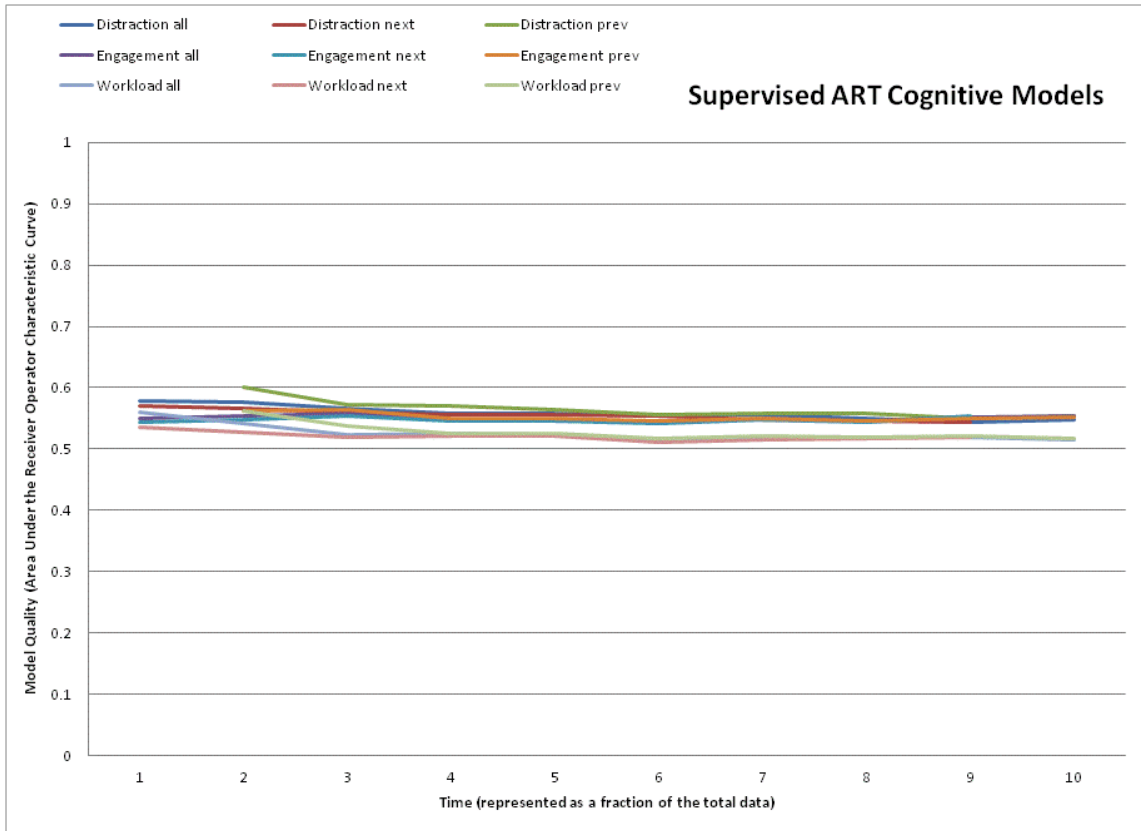
Figure 130 – Performance of supervised ART for cognitive modeling for Results Set #4
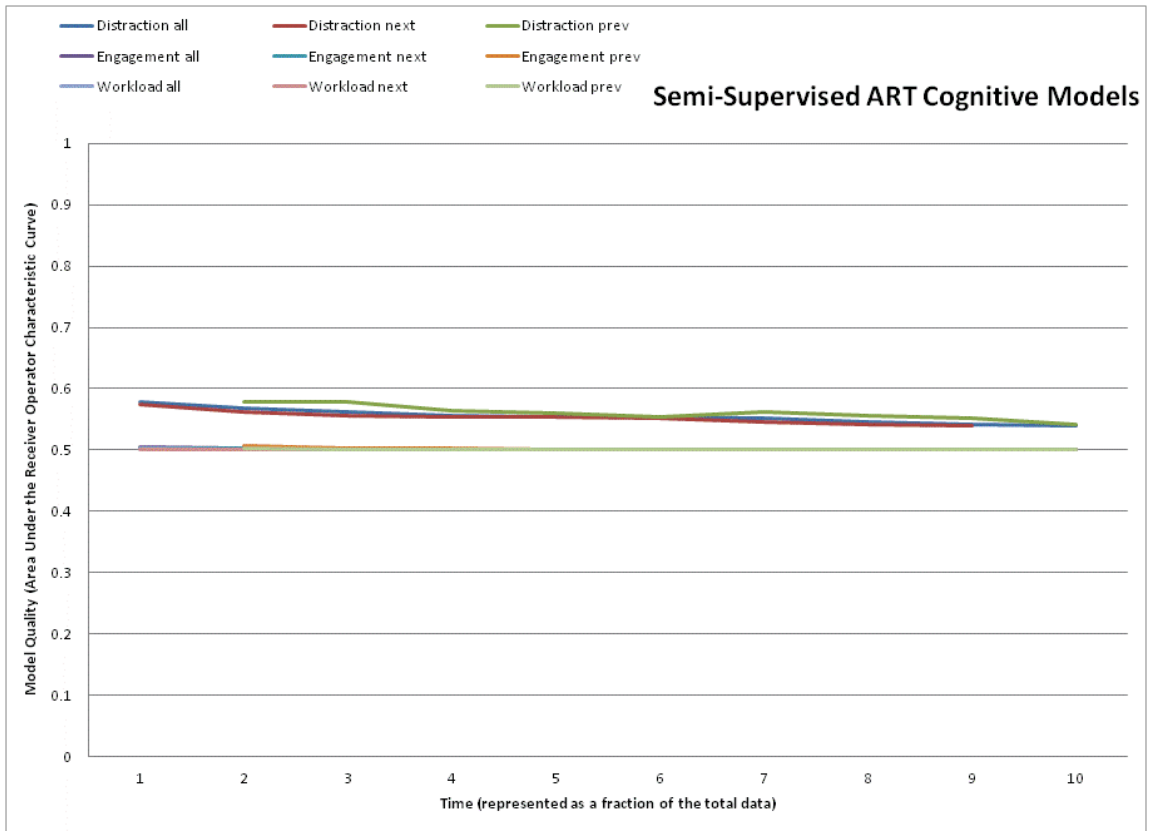
Figure 131 – Performance of supervised ART for cognitive modeling for Results Set #4

Figure 132 – Performance of unsupervised ART for affective modeling for Results Set #4
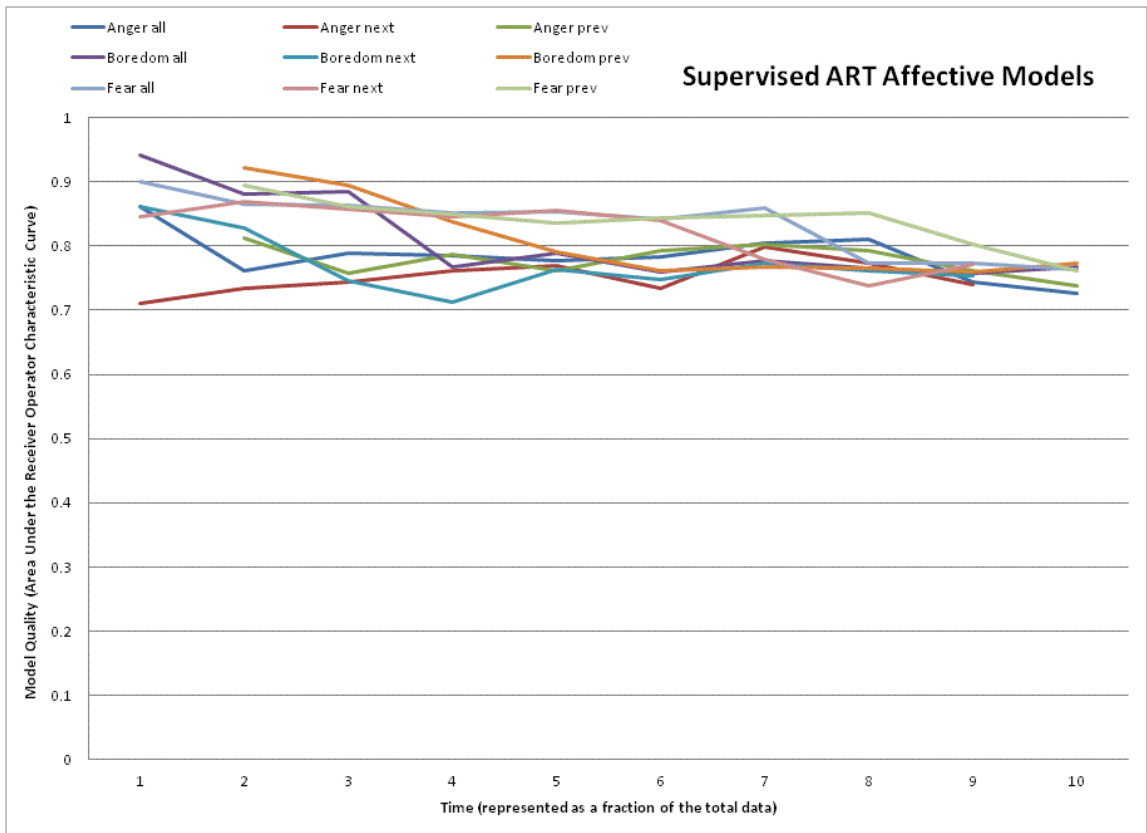
Figure 133 – Performance of supervised ART for affective modeling for Results Set #4

Figure 134 – Performance of semi-supervised ART for affective modeling for Results Set #4

Appendix C-4-2   K-Means

Figure 135 – Performance of unsupervised clustering for cognitive modeling for Results Set #4

401

Figure 136 – Performance of supervised clustering for cognitive modeling for Results Set #4

Figure 137 – Performance of semisupervised clustering for cognitive modeling for Results Set #4
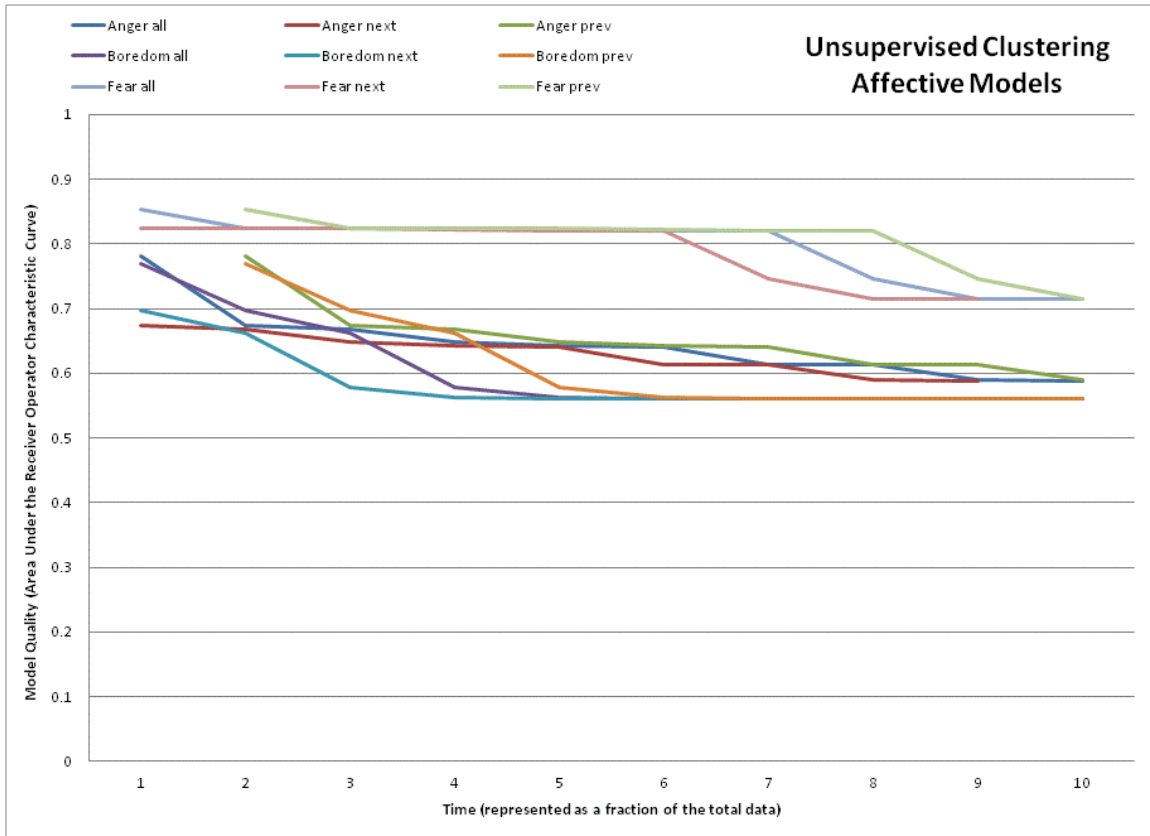
Figure 138 – Performance of unsupervised clustering for affective modeling for Results Set #4
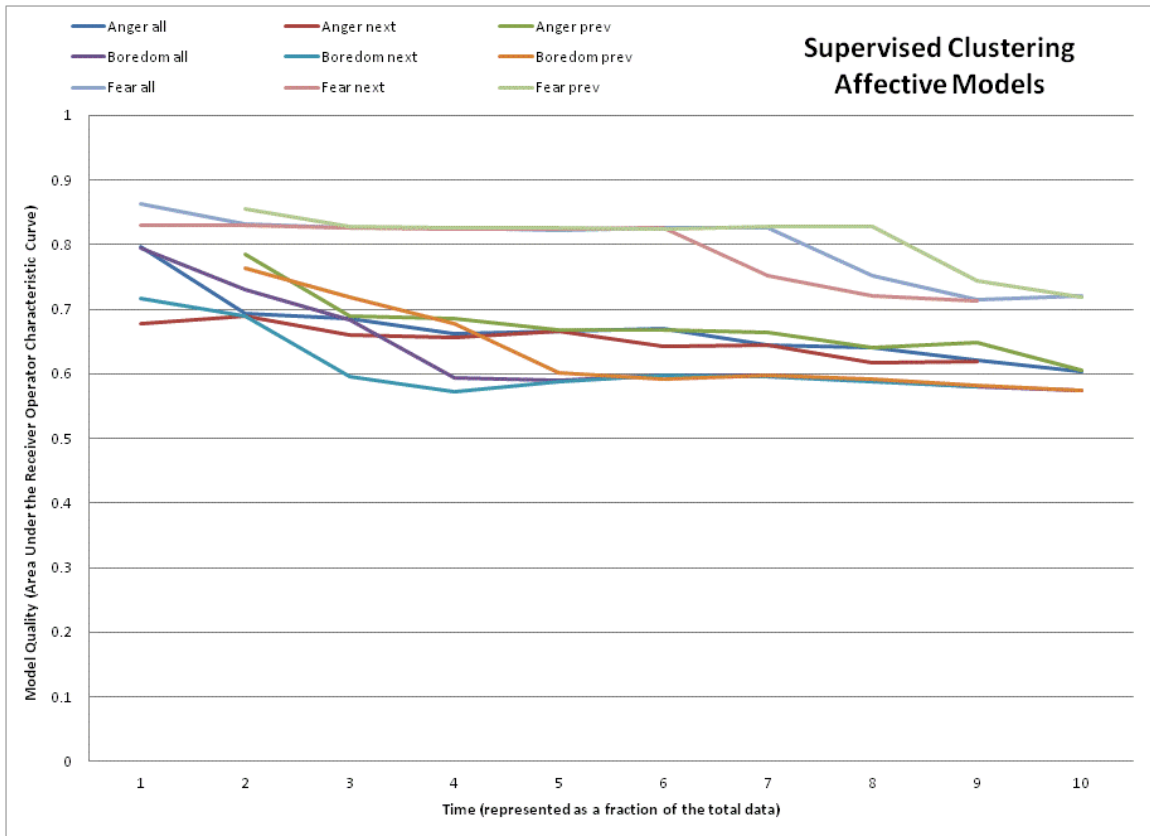
Figure 139 – Performance of supervised clustering for cognitive modeling for Results Set #4
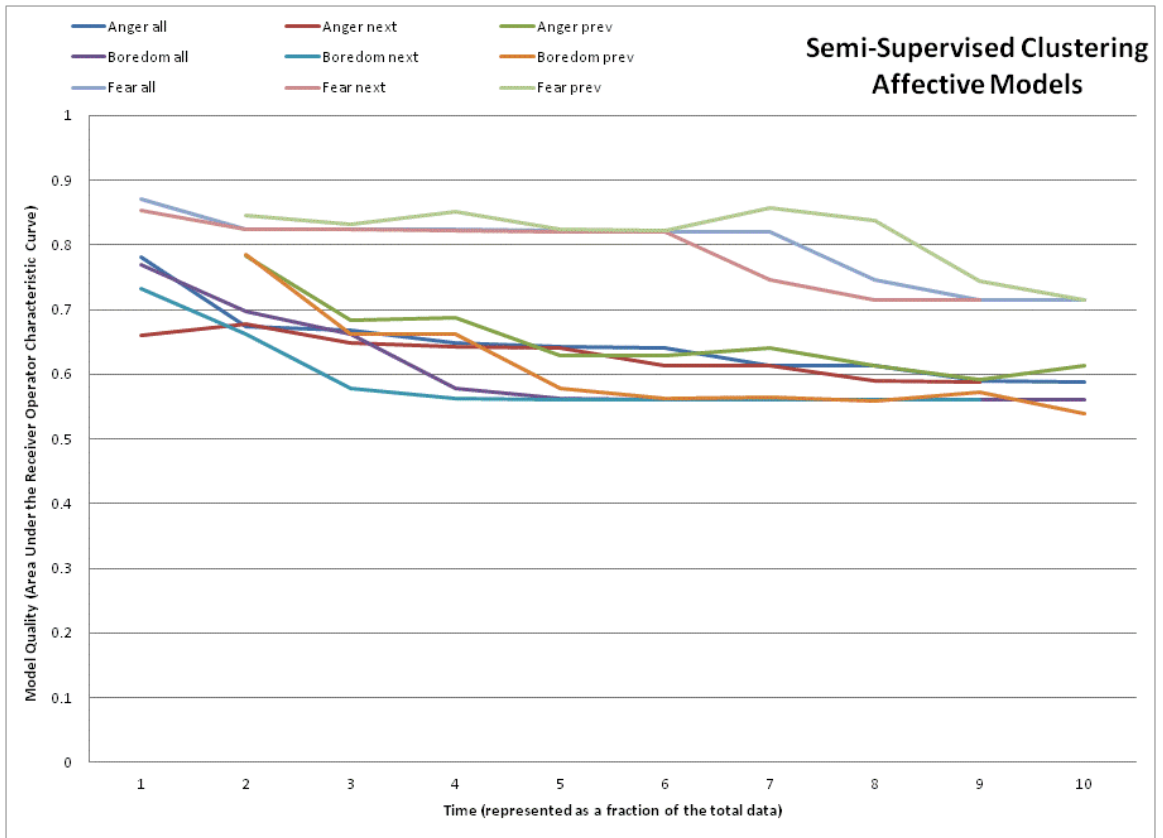
Figure 140 – Performance of semi-supervised clustering for cognitive modeling for Results Set #4
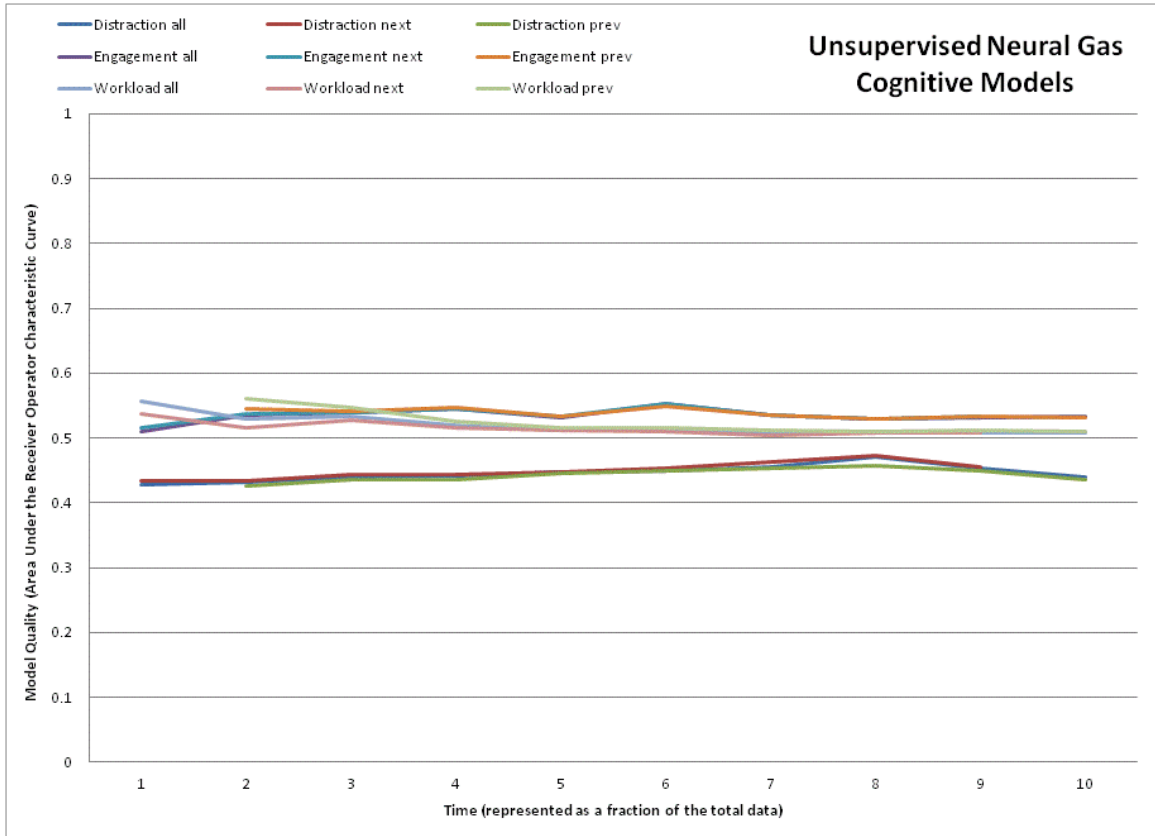
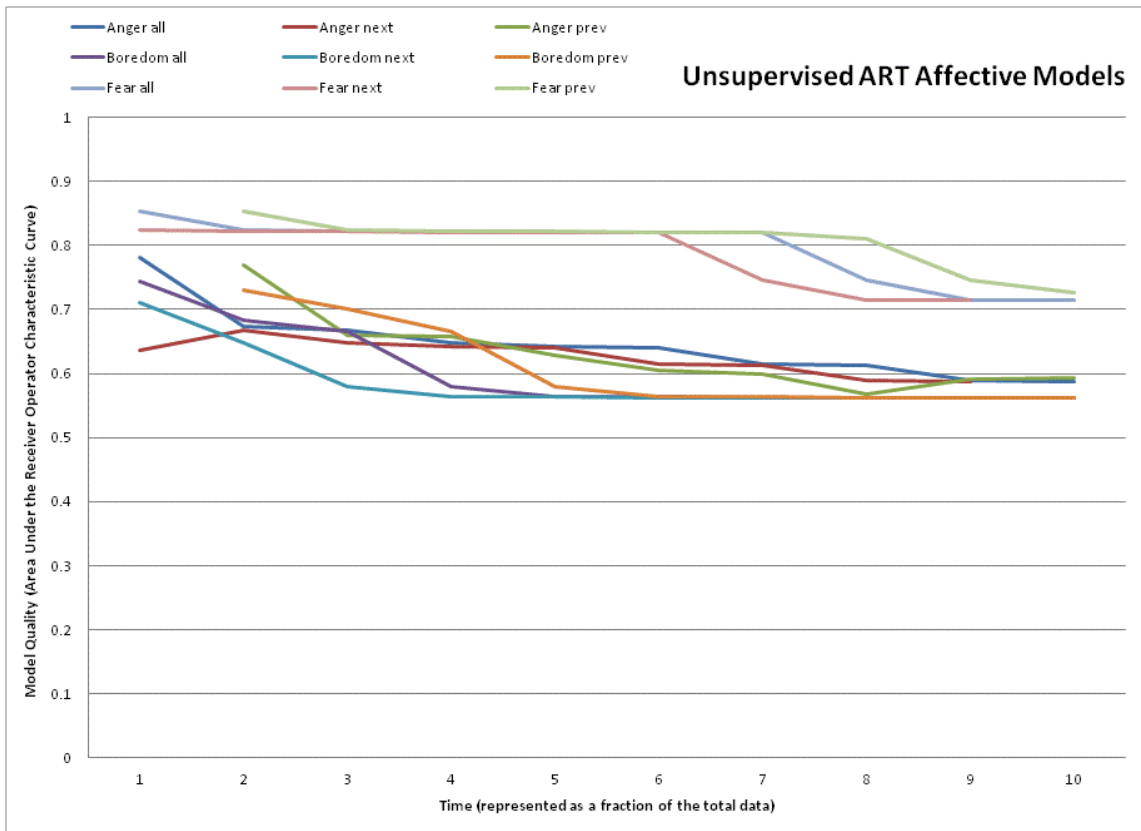**Figure 141 – Performance of unsupervised neural gas for cognitive modeling for Results Set #4**
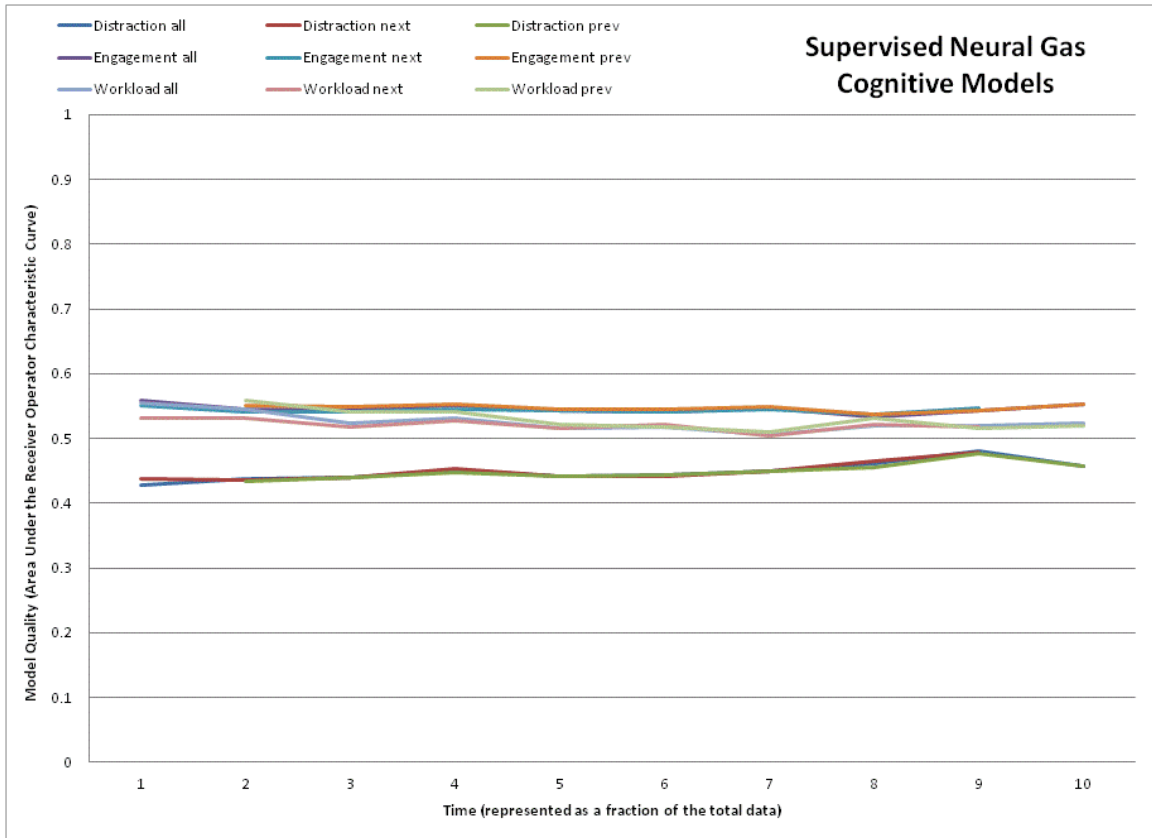
Figure 142 – Performance of supervised neural gas for cognitive modeling for Results Set #4
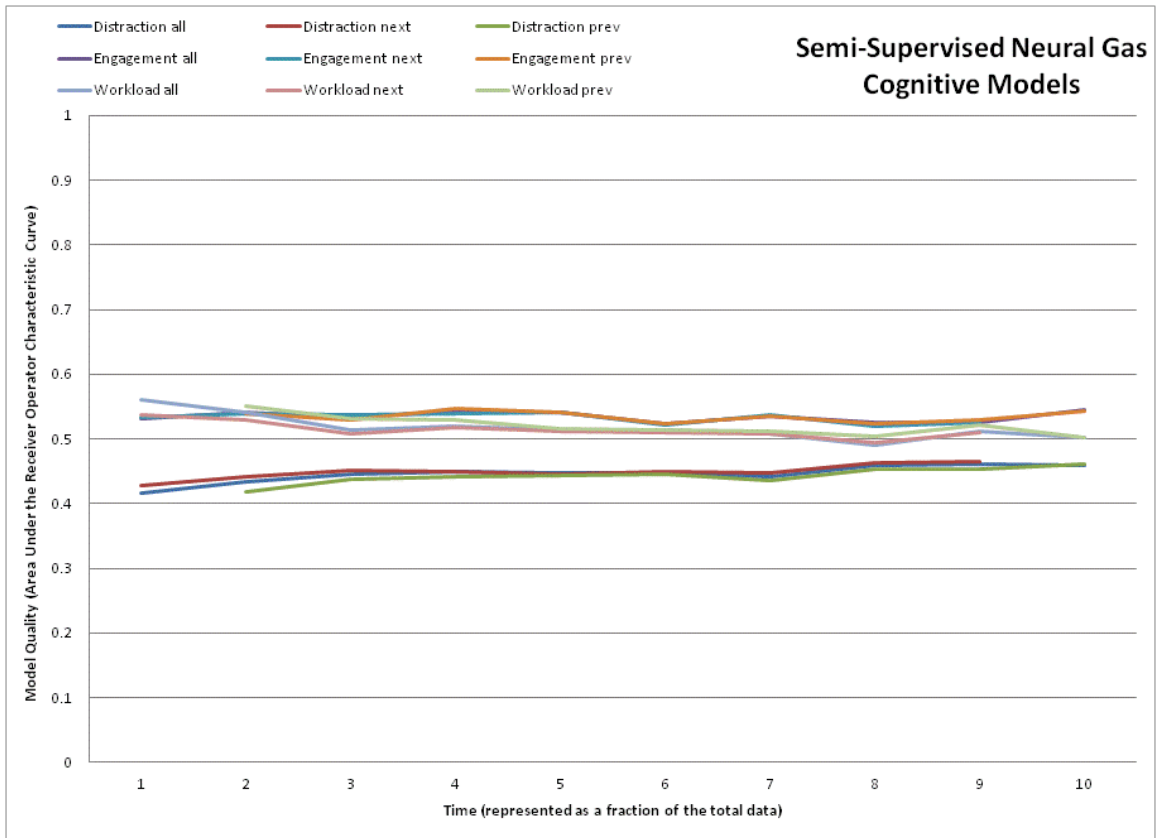
Figure 143 – Performance of semi-supervised neural gas for cognitive modeling for Results Set #4
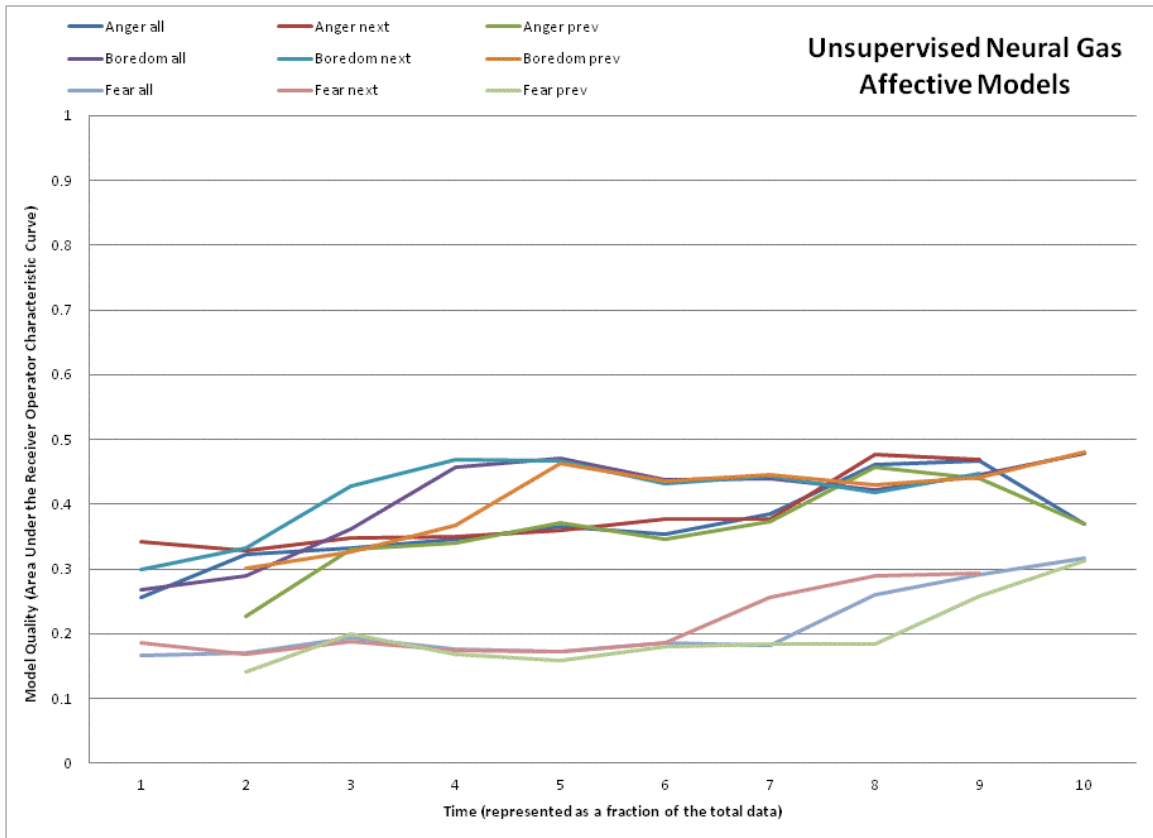
Figure 144 – Performance of unsupervised neural gas for affective modeling for Results Set #4
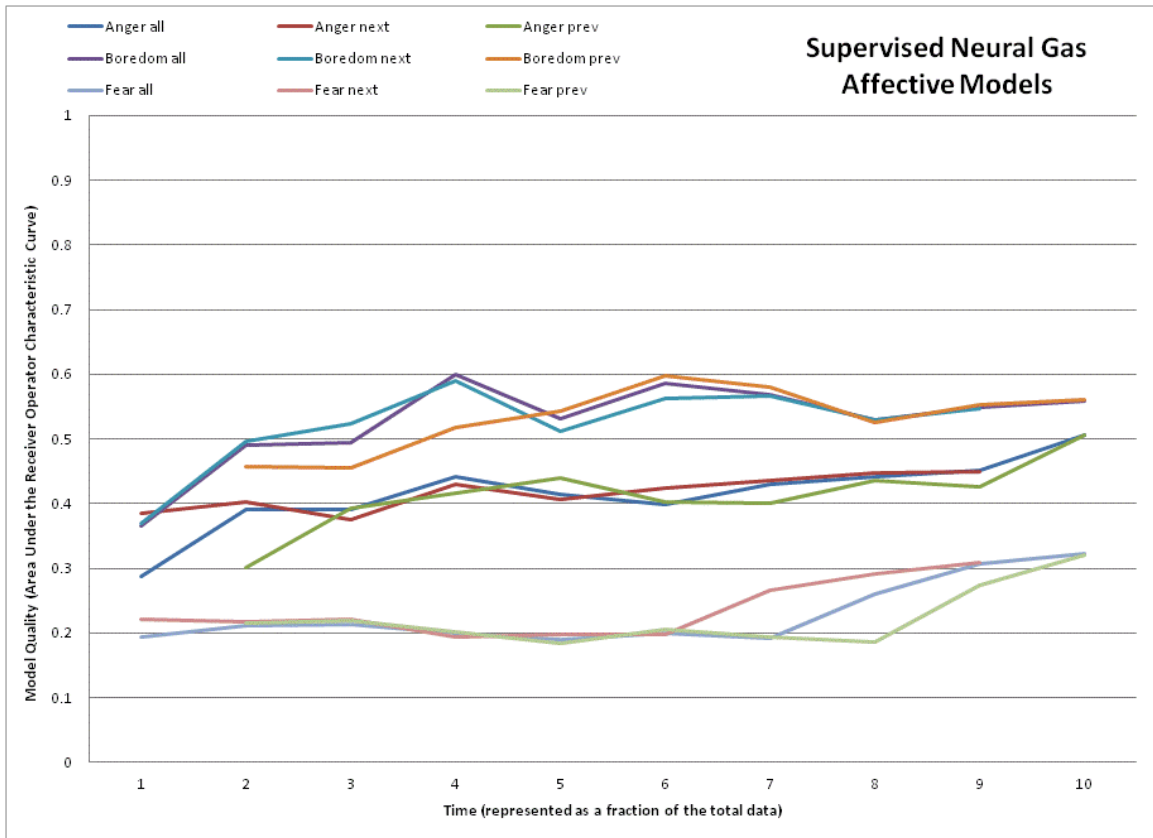
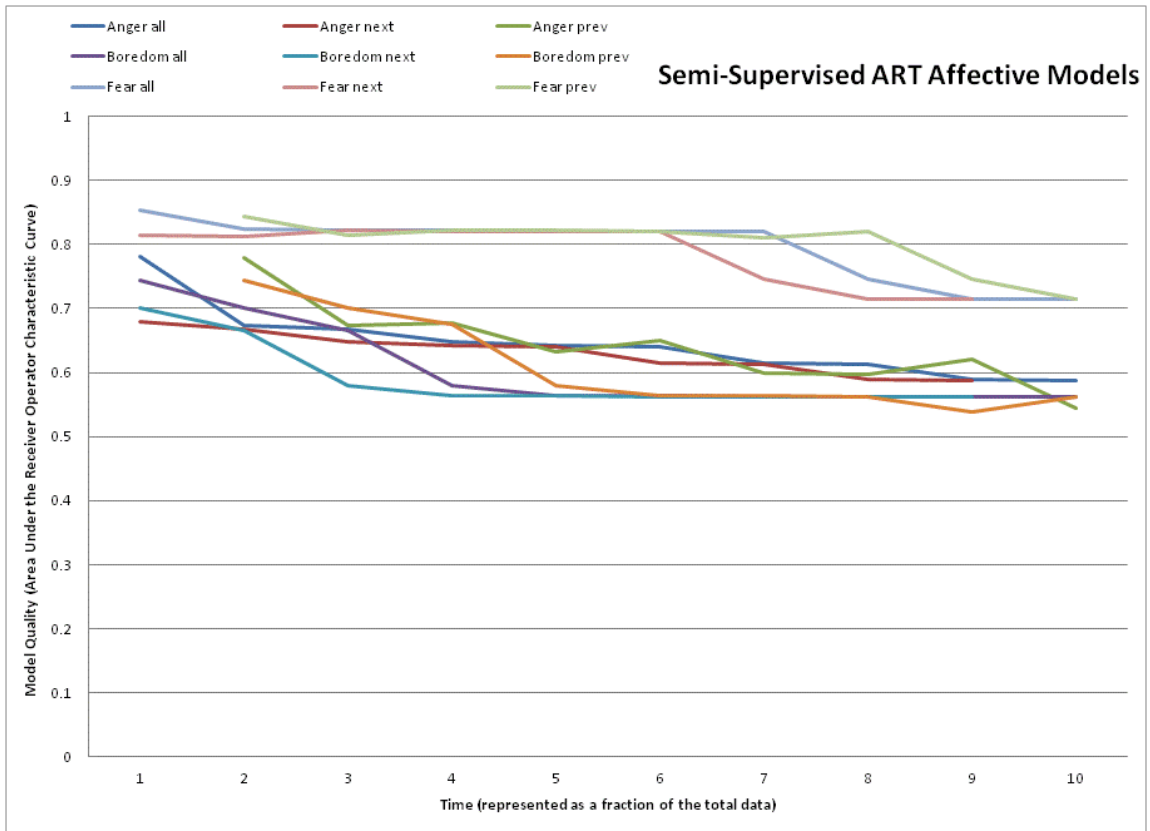**Figure 145 – Performance of supervised neural gas for affective modeling for Results Set #4**
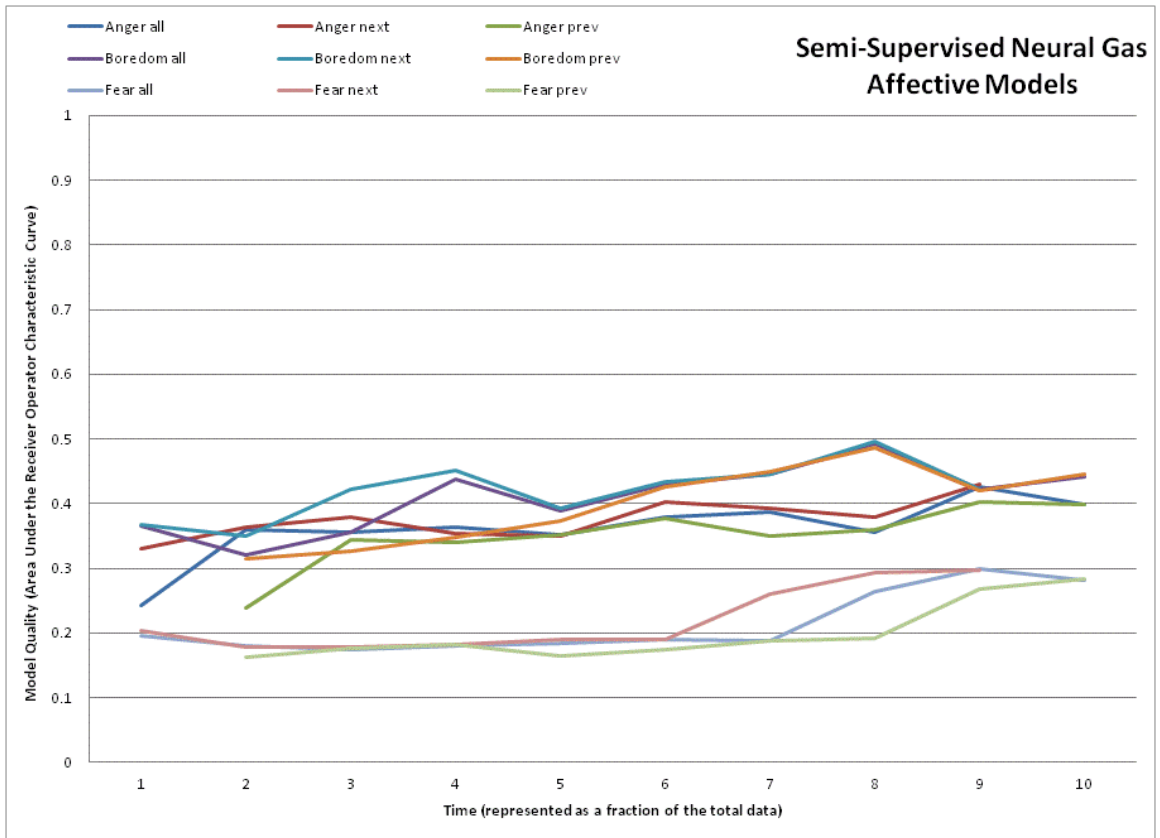
Figure 146 – Performance of semi-supervised neural gas for affective modeling for Results Set #4
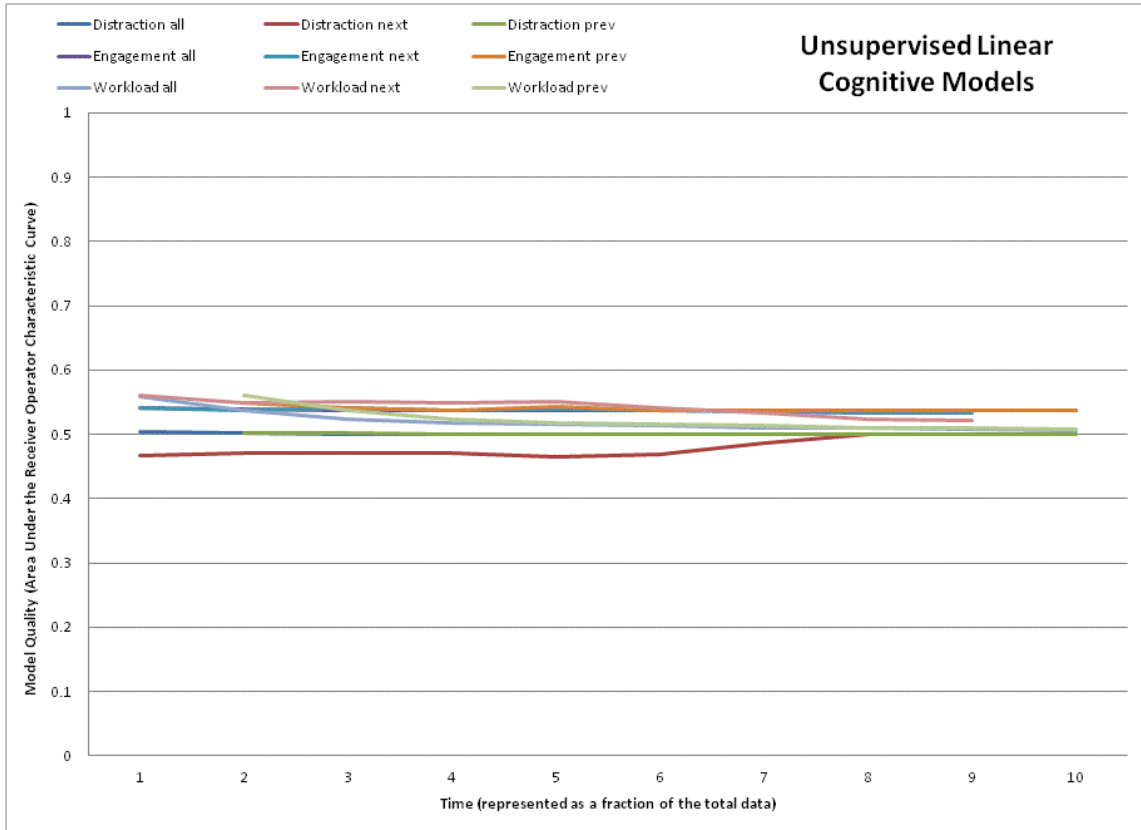
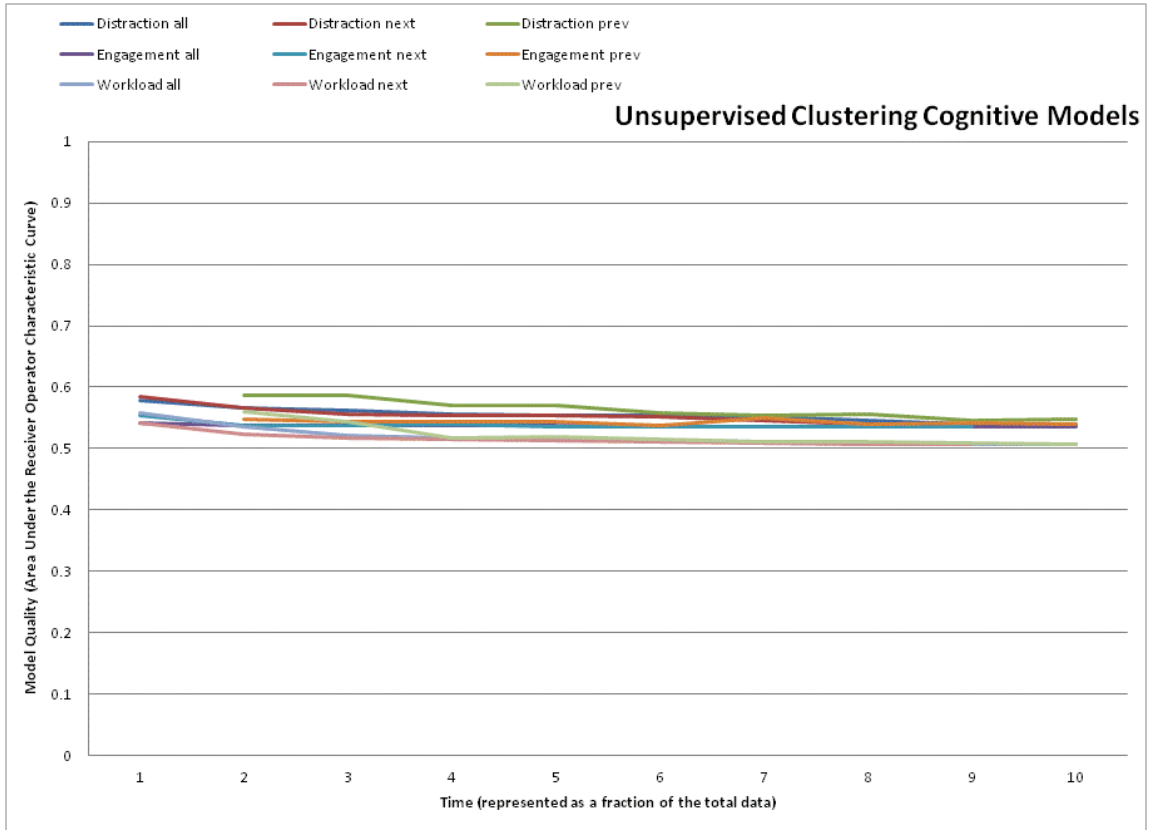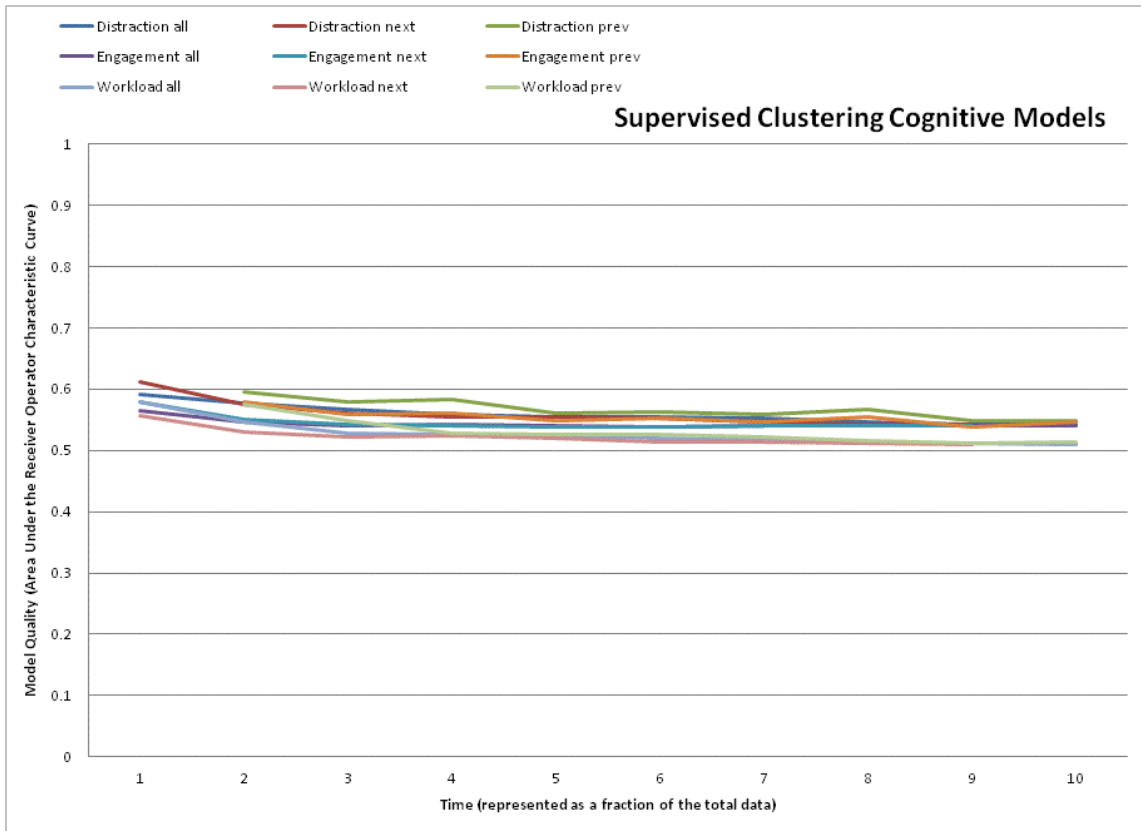**Figure 147 – Performance of unsupervised VW for cognitive modeling for Results Set #4**

413
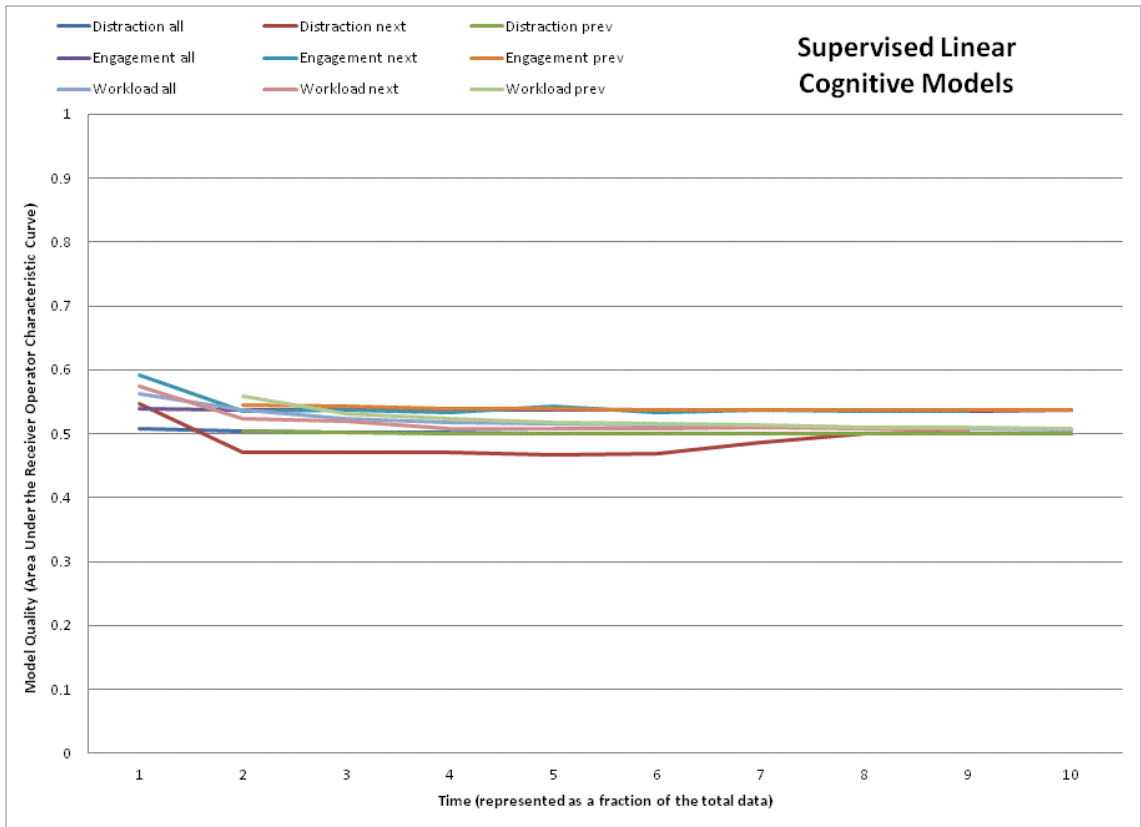
Figure 148 – Performance of supervised VW for cognitive modeling for Results Set #4

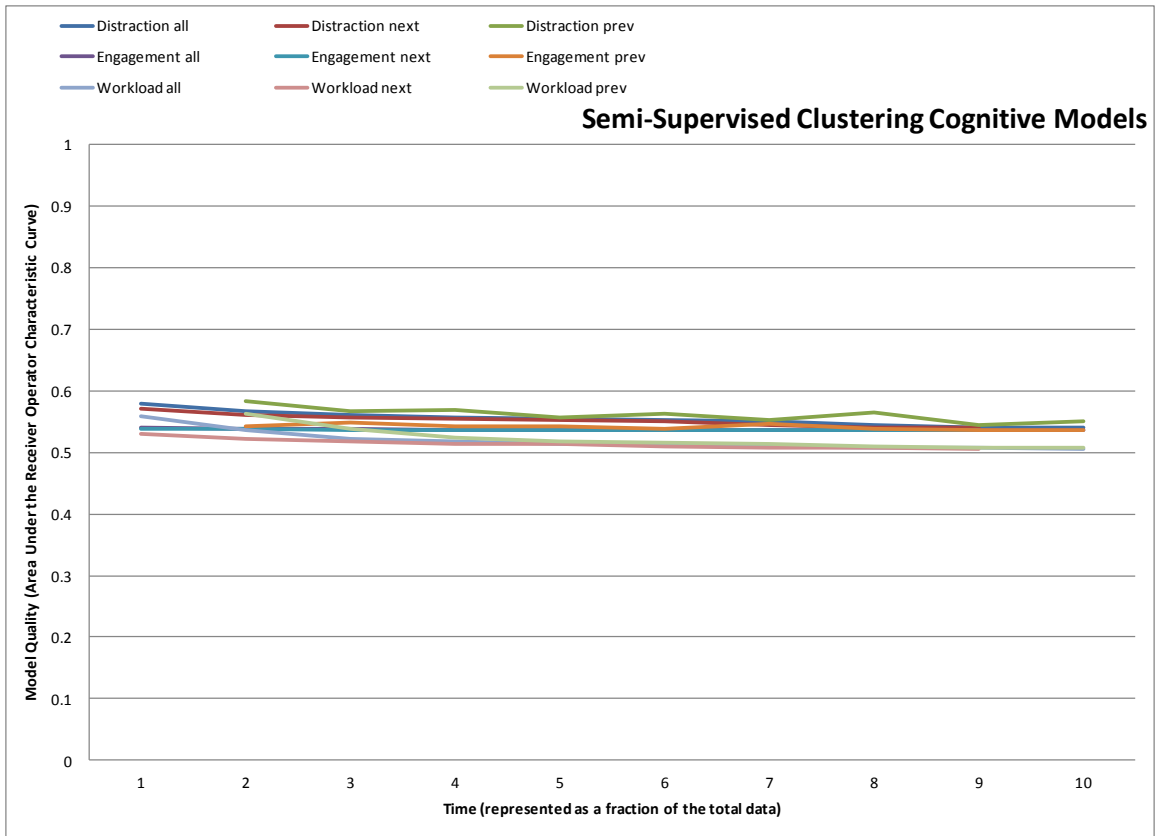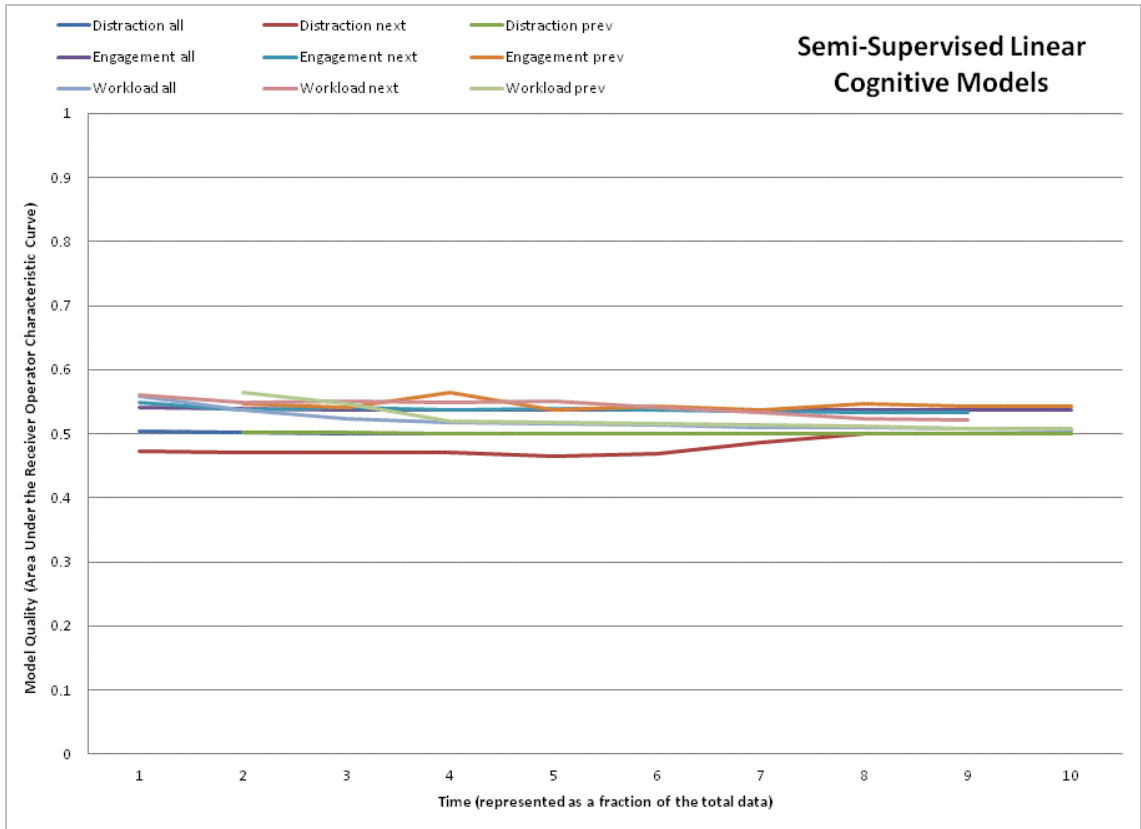**Figure 149 – Performance of semi-supervised VW for cognitive modeling for Results Set #4**

415
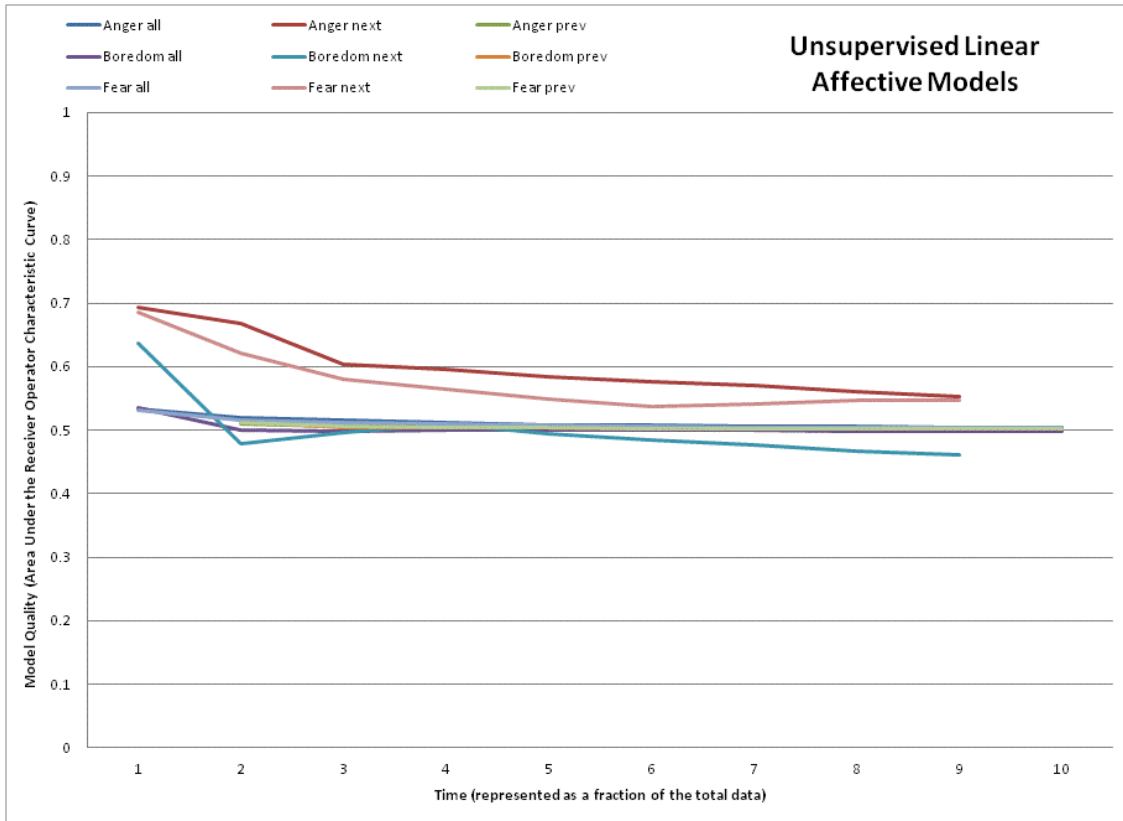
Figure 150 – Performance of unsupervised VW for affective modeling for Results Set #4
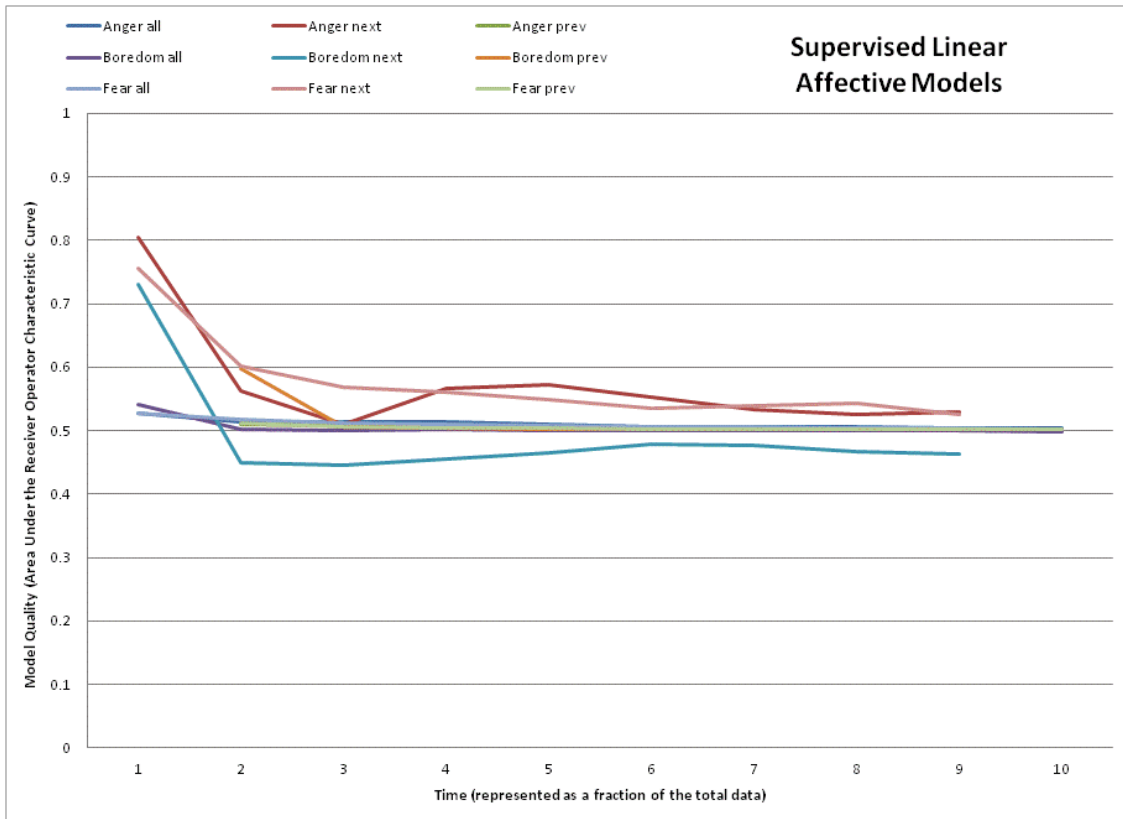
Figure 151 – Performance of supervised VW for affective modeling for Results Set #4
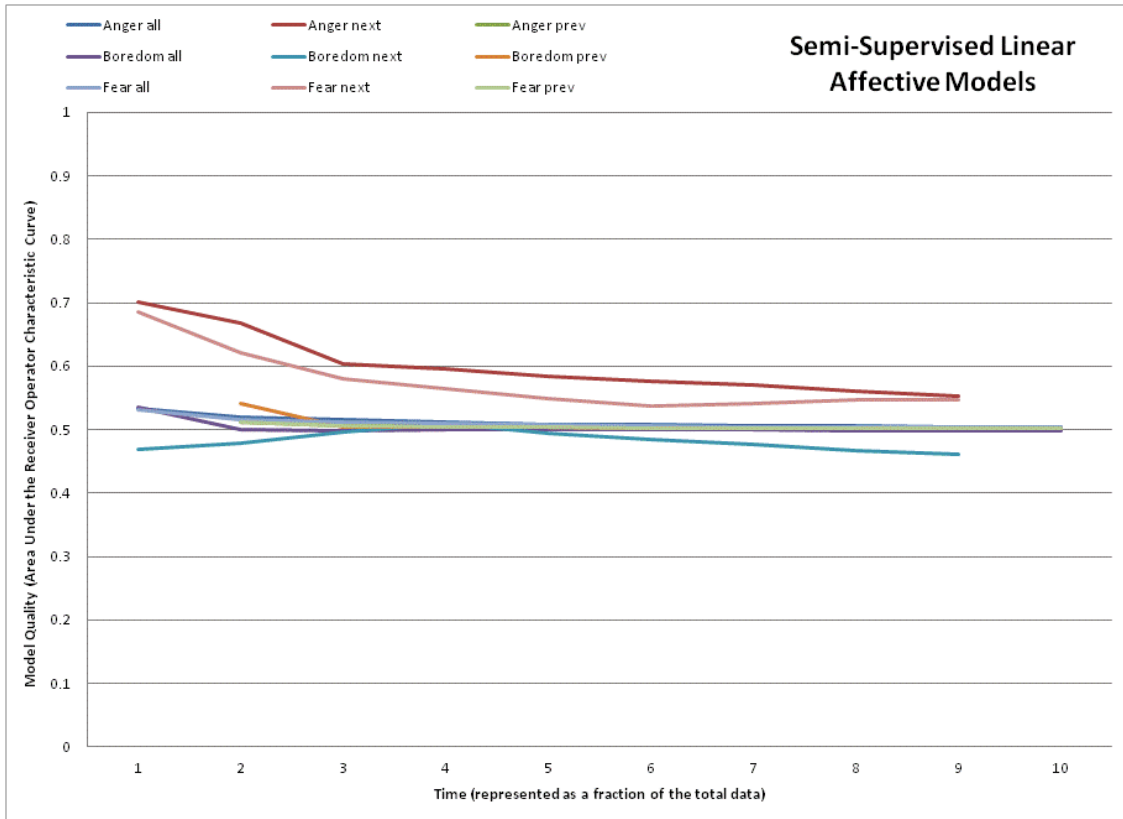
Figure 152 – Performance of semi-supervised VW for affective modeling for Results Set #4

Appendix C-4-5   Total Results Set #4 Semi-Supervised Modeling Ability
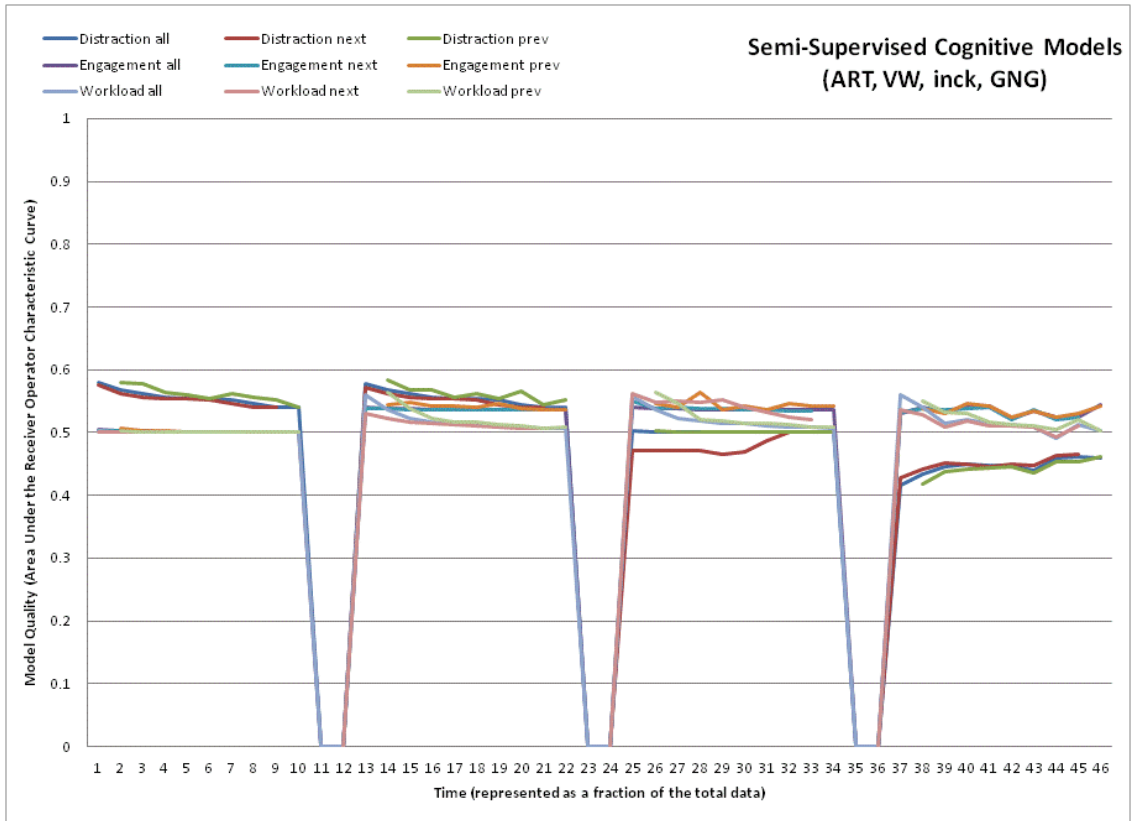
Figure 153 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for cognitive modeling for Results Set #4
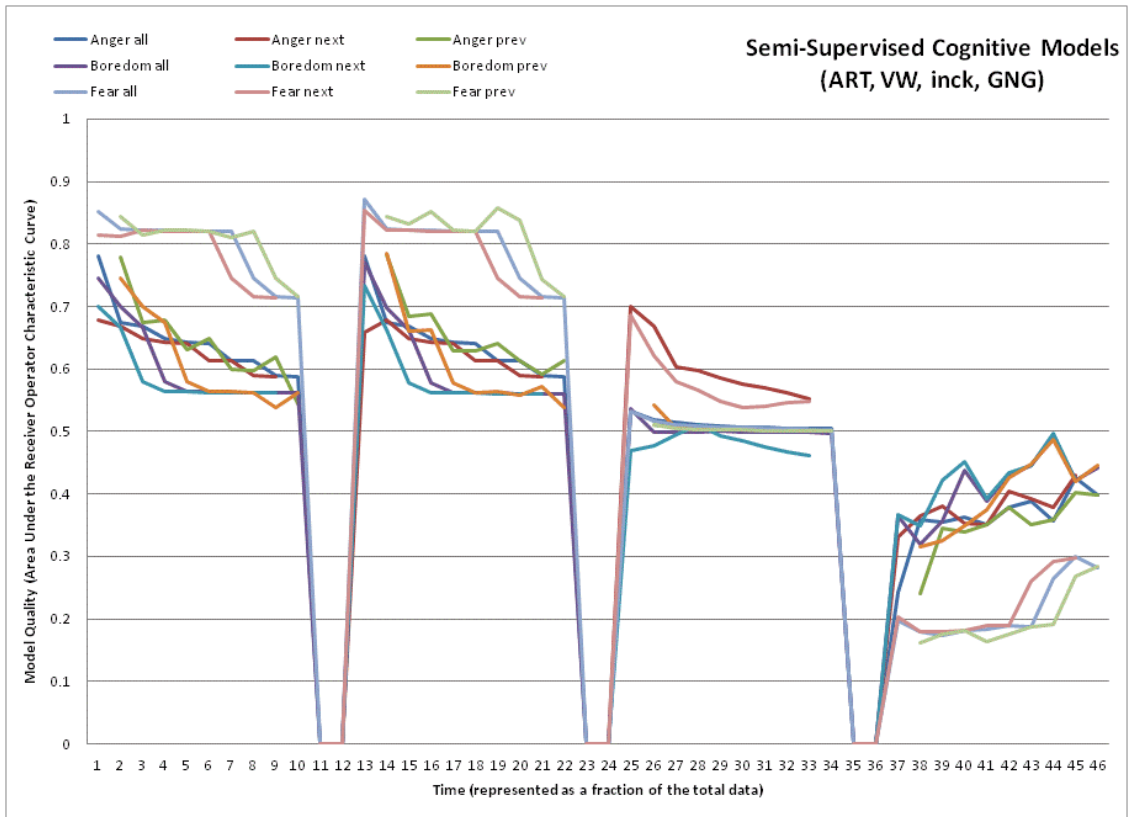
Figure 154 – Performance of semi-supervised methods (ART, K-Means, VW, OSSGNG) for affective modeling for Results Set #4

APPENDIX D   VARIATION OF PARAMETERS OF THE

ADAPTIVE RESONANCE THEORY ALGORITHM

Additional attempts to tune parameter settings on the ART algorithm were attempted in order to recommend parameter settings for field use. Vigilance parameter values of 01., 0.3, 0.5, 0.7, and 0.9 were tested with full supervision and examined with various results presented below. Generally, the 0.75 parameter setting value which was initially attempted based on literature recommendations was found to have acceptable performance.

The reader should note that a vigilance parameter setting of 0.9 is a very large vigilance parameter. As a result, nearly every datapoint is given its own input category, which classifies a very small amount of the total data. This leads to higher overall accuracy, but at the cost of practicality. For a practical Intelligent Tutoring System to make use of learner data, it must have stable categories over an area of instruction. Extremely high number of classification categories do not allow for this, but provide an estimate of the levels of vigilance which must be selected.
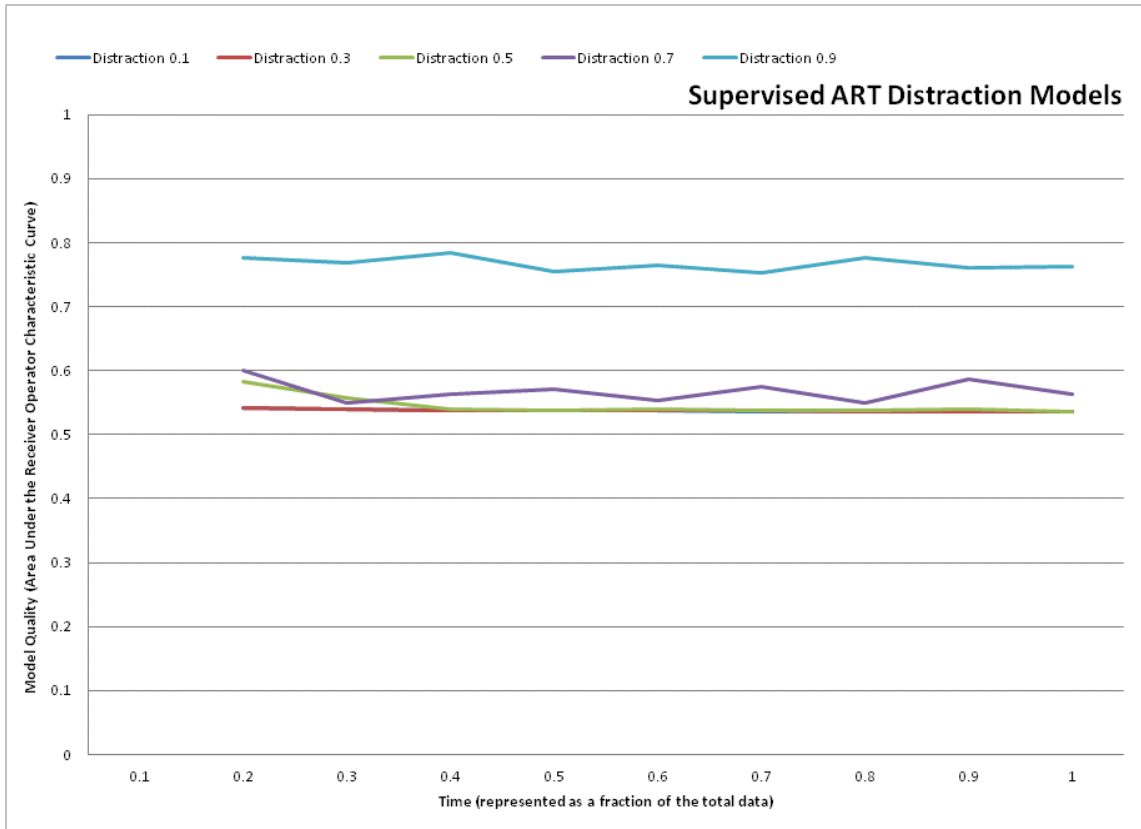
Figure 155 – Performance of various ART parameters for modeling Distraction
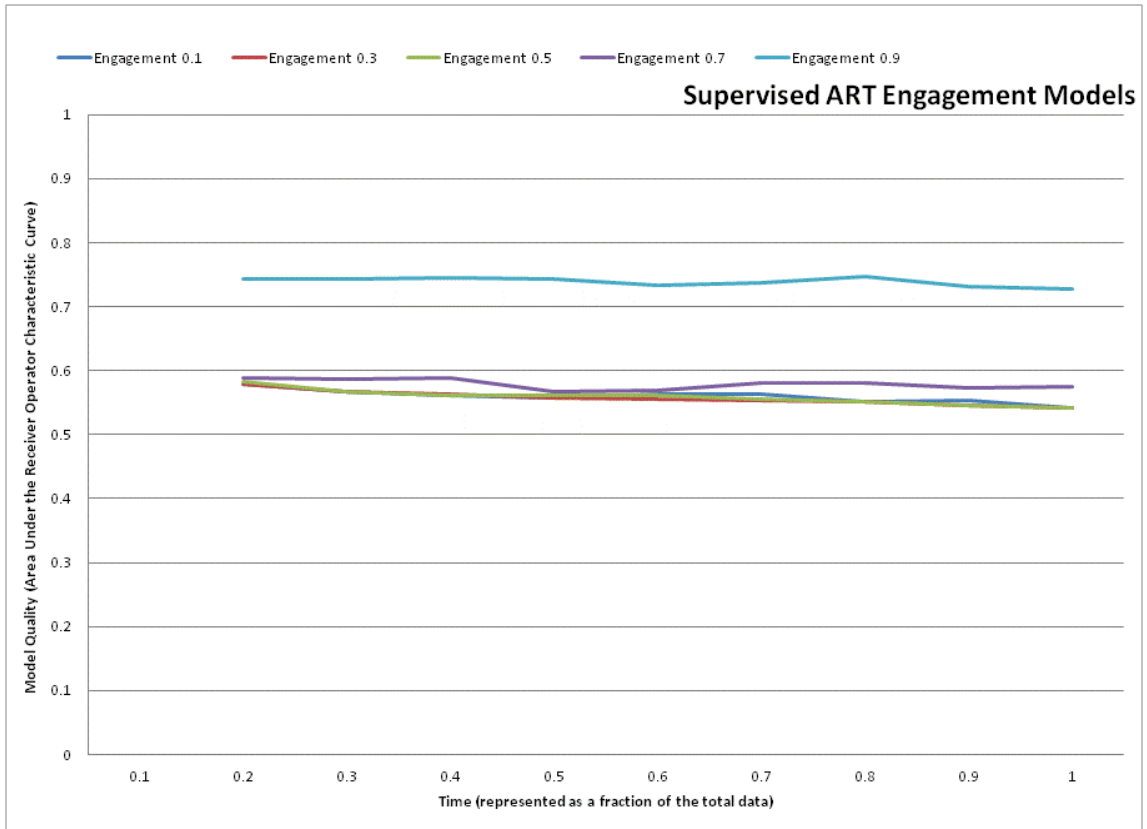
Figure 156 – Performance of various ART parameters for modeling Engagement
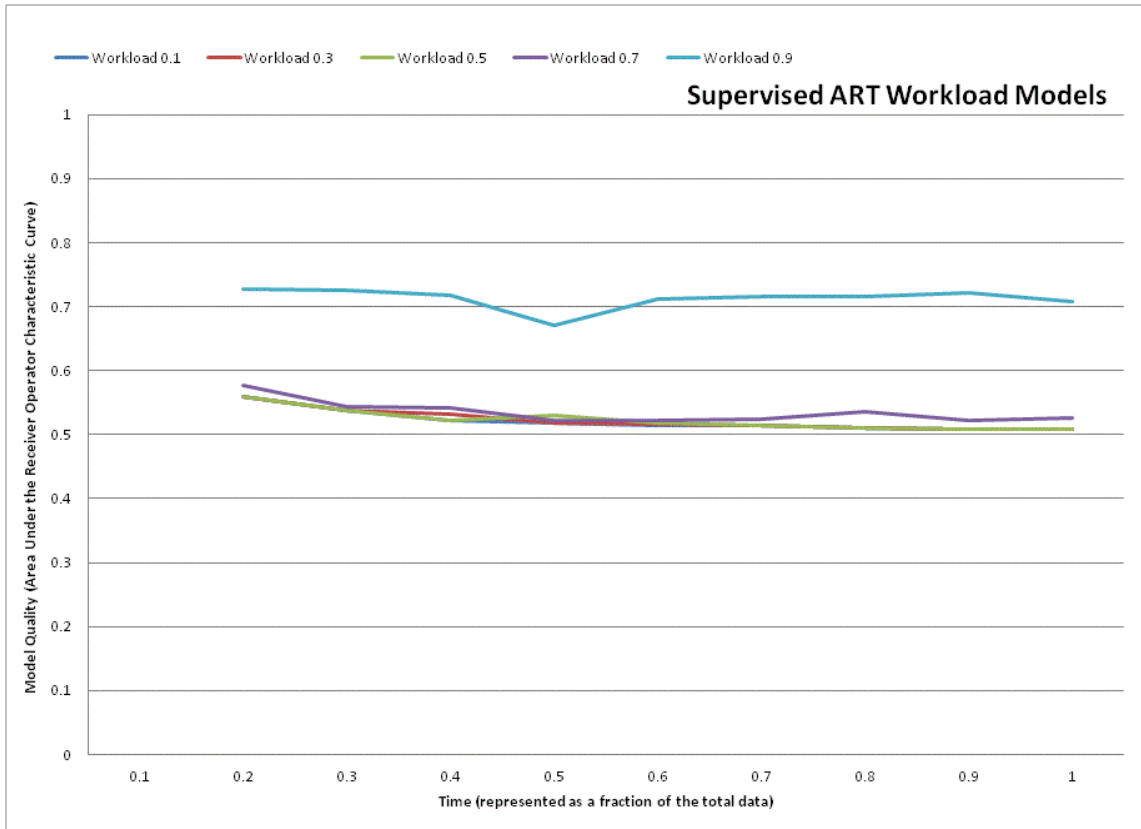
Figure 157 – Performance of various ART parameters for modeling Workload
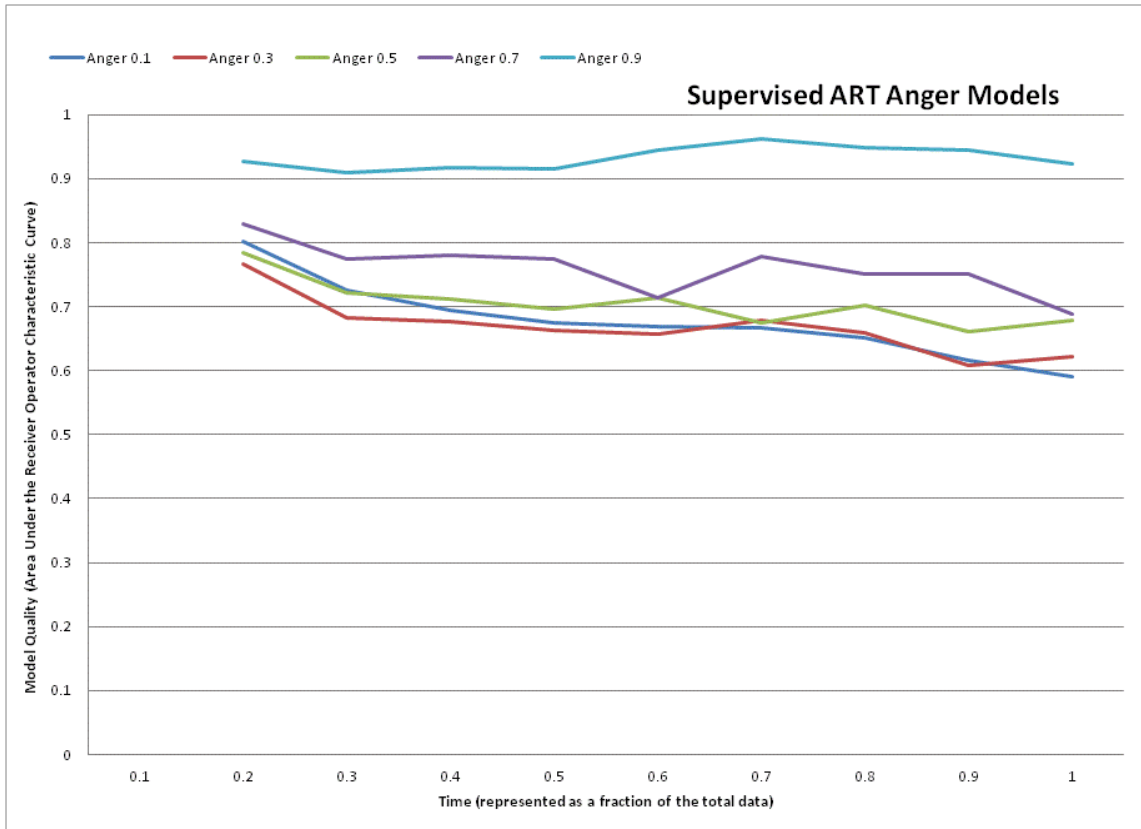
Figure 158 – Performance of various ART parameters for modeling Anger
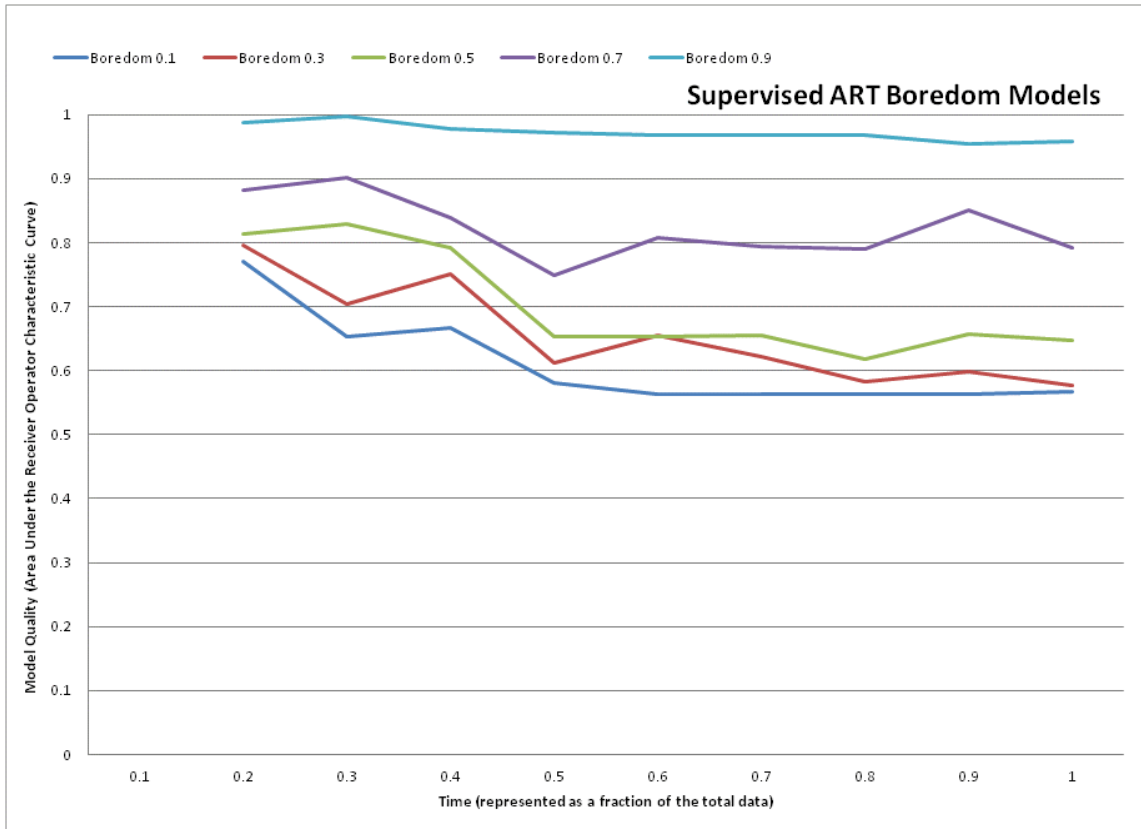
Figure 159 – Performance of various ART parameters for modeling Boredom
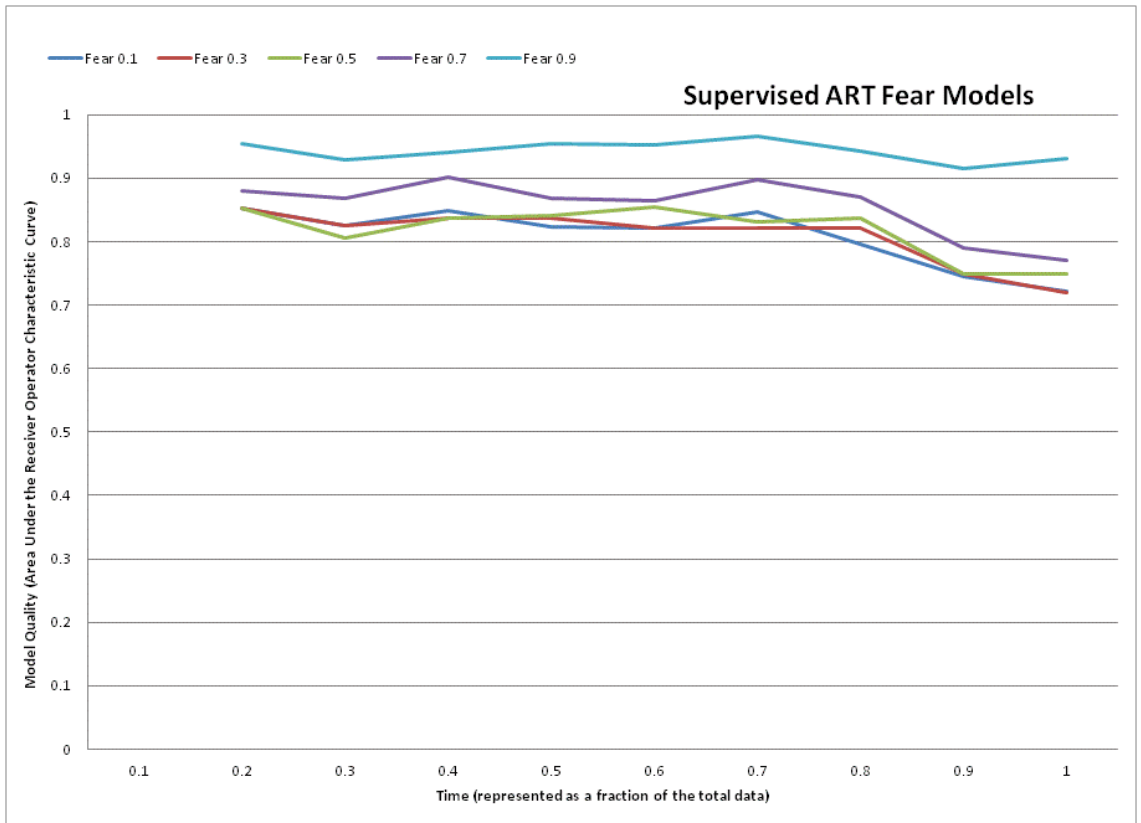
Figure 160 – Performance of various ART parameters for modeling Fear

Table 68 – Quality values for various parameter settings using supervised ART

Model	Quality Metric				
	0.1	0.3	0.5	0.7	0.9
Anger	0.677	0.668	0.705	0.760	0.932
Boredom	0.610	0.655	0.702	0.823	0.973
Fear	0.809	0.809	0.818	0.857	0.943
Distraction	0.538	0.538	0.545	0.568	0.767
Engagement	0.560	0.557	0.559	0.579	0.739
Workload	0.522	0.523	0.523	0.535	0.713

Table 69 – Percentage of usable models for various parameter settings using supervised ART

Model	Percentage Usable				
	0.1	0.3	0.5	0.7	0.9
Anger	58%	63%	74%	89%	100%
Boredom	32%	63%	89%	100%	100%
Fear	89%	74%	95%	100%	100%
Distraction	14%	7%	14%	7%	100%
Engagement	21%	14%	14%	14%	100%
Workload	0%	0%	0%	0%	100%

LIST OF REFERENCES

Abdelbar, A. M., and Hedetniemi, S. M. (1998). "Approximating MAPs for belief networks is NP-hard and other theorems." *Artificial Intelligence*, 102(1), 21-38.

Abraham, A., Corchado, E., and Corchado, J. M. (2009). "Hybrid learning machines."

Agarwal, A., Chapelle, O., Dudík, M., and Langford, J. (2011). "A reliable effective terascale linear learning system." *arXiv preprint arXiv:1110.4198*.

Ahissar, M., and Hochstein, S. (2002). "The Role of Attention in Learning Simple Visual Tasks", in M. Fahle and T. Poggio, (eds.), *Perceptual Learning*. Cambridge, MA: The MIT Press, pp. 253-272.

Ahlstrom, U., and Friedman-Bern, F. J. (2006). "Using eye movement activity as a correlate of cognitive workload." *International Journal of Industrial Ergonomics*, 36(7), 623-636.

Alexander, T., Goldberg, S., Magee, L., Sottilare, R., Andrews, D., and Roessingh, J. J. "Enhancing Human Effectiveness through Embedded Virtual Simulation." *Presented at The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*.

AlZoubi, O., Calvo, R., and Stevens, R. (2009). "Classification of EEG for Affect Recognition: An Adaptive Approach." *AI 2009: Advances in Artificial Intelligence*, 52-61.

Alzoubi, O., Hussain, S., D'Mello, S., and Calvo, R. A. (2011). "Affective Modeling from Multichannel Physiology: Analysis of Day Differences", in S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 4-13.

AlZoubi, O., Koprinska, I., and Calvo, R. A. "Classification of Brain-Computer Interface Data."

Anderson, G., and Brown, R. I. F. (1984). "Real and laboratory gambling, sensation-seeking and arousal." *British Journal of Psychology*, 75(3), 401-410.

Anderson, J. R. (1987). "Production systems, learning and tutoring", in D. Klahr, P. Langley, and R. Neches, (eds.), *Production System Models of Learning and Development* London: MIT Press,, pp. 437-458.

Anderson, J. R., Boyle, C. F., Farrell, R., and Reiser, B. J. (1987). "Cognitive Principles in the Design of Computer Tutors", in P. Morris, (ed.), *Modelling Cognition*. John Wiley & Sons Ltd., pp. 93-133.

Army, D. o. t. (2011). *The U.S. Army Learning Concept for 2015*. TRADOC.

Arroyo, I., Cooper, D. G., Burleson, W., Woolf, B. P., Muldner, K., and Christopherson, R. (2009). "Emotion sensors go to school"*International Conference on Artificial Intelligence in Education*. City: IOS Press, pp. 17-24.

Arroyo, I., and Woolf, B. P. "Inferring learning and attitudes from a Bayesian Network of log file data."

Arroyo, I., Woolf, B. P., and Beal, C. R. (2006). "Addressing Cognitive Differences and Gender During Problem Solving." *Technology, Instruction, Cognition and Learning*, 4, 31-63.

Baker, R., Gowda, S., Corbett, A., and Ocumpaugh, J. "Towards automatically detecting whether student learning is shallow."

Baker, R. S., Corbett, A. T., Koedinger, K. R., and Wagner, A. Z. "Off-task behavior in the cognitive tutor classroom: when students game the system." *Presented at Proceedings of the SIGCHI conference on Human factors in computing systems*.

Baker, R. S. J., D'Mello, S. K., Rodrigo, M. M. T., and Graesser, A. C. (2010). "Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive-affective states during interactions with three different computer-based learning environments." *International Journal of Human-Computer Studies*, 68(4), 223-241.

Baker, R. S. J., Kalka, J., Aleven, V., Rossi, L., Gowda, S. M., Wagner, A. Z., Kusbit, G. W., Wixon, M., Salvi, A., and Ocumpaugh, J. (2012b). "Towards Sensor-Free Affect Detection in Cognitive Tutor Algebra."

Baker, R. S. J. d. (2010). "Mining data for student models", in R. Nkmabou, R. Mizoguchi, and J. Bourdeau, (eds.), *Advances in Intelligent Tutoring Systems, Studies in Computational Intelligence*. Heidelberg: Springer Verlag, pp. 323-337.

Banda, N., and Robinson, P. (2011). "Multimodal Affect Recognition in Intelligent Tutoring Systems", in S. D. Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 200-207.

Barber, D., and Hudson, I. (2011). "Distributed logging and synchronization of physiological and performance measures to support adaptive automation strategies." *Foundations of Augmented Cognition. Directing the Future of Adaptive Systems*, 559-566.

Barr, A., and Feigenbaum, E. A. (1982). *The Handbook of Artificial Intelligence*, Los Altos, CA: Kaufmann.

Bartels, M., and Marshall, S. P. "Measuring cognitive workload across different eye tracking hardware platforms." *Presented at Proceedings of the Symposium on Eye Tracking Research and Applications*.

Beck, J., Stern, M., and Haugsjaa, E. (1996). "Applications of AI in Education." *ACM Crossroads*, 3(1), 11-15.

Benasich, A. A., Gou, Z., Choudhury, N., and Harris, K. D. (2008). "Early cognitive and language skills are linked to resting frontal gamma power across the first 3 years." *Behavioural brain research*, 195(2), 215-222.

Beni, G., and Wang, J. (1993). "Swarm intelligence in cellular robotic systems." *Robots and Biological Systems: Towards a New Bionics?*, 703-712.

Beringer, J., and Hüllermeier, E. (2006). "Online clustering of parallel data streams." *Data Knowl. Eng*, 58(2), 180-204.

Berka, C., Levendowski, D. J., Cvetinovic, M., Petrovic, M. M., Davis, G. F., Lumicao, M. N., and al., e. (2004). "Real-time analysis of EEG indices of alertness, cognition and memory acquired with a wireless EEG headset." *International Journal of Human-Computer Interaction*, 17(2), 151-170.

Berka, C., Levendowski, D. J., Lumicao, M. N., Yau, A., Davis, G., Zivkovic, V. T., Olmstead, R. E., Tremoulet, P. D., and Craven, P. L. (2007). "EEG Correlates of Task Engagement and Mental Workload in Vigilance, Learning, and Memory Tasks." *Aviation Space and Environmental Medicine*, 78(5), B231-B244.

Bersak, D., McDarby, G., Augenblick, N., McDarby, P., McDonnell, D., McDonal, B., and Karkun, R. "Biofeedback using an Immersive Competitive Environment." *Presented at Ubicomp 2001, Designing Ubiquitous Computing Games Workshop*.

Beyer, O., and Cimiano, P. "Online semi-supervised growing neural gas." *Presented at Workshop New Challenges in Neural Computation 2011*.

Beygelzimer, A., Hsu, D., Karampatziakis, N., Langford, J., and Zhang, T. "Efficient active learning." *Presented at ICML 2011 Workshop on On-line Trading of Exploration and Exploitation*.

Beygelzimer, A., Hsu, D., Langford, J., and Zhang, T. (2010a). "Agnostic active learning without constraints." *arXiv preprint arXiv:1006.2588.*

Beygelzimer, A., Hsu, D., Langford, J., and Zhang, T. (2010b). "Agnostic Active Learning Without Constraints"*Neural Information Processing Systems*. City.

Blanchard, E., Chalfoun, P., and Frasson, C. (2007). "Towards advanced learner modeling: Discussion on quasi real-time adaptation with physiological data", *Proceedings of the 7th IEEE International Conference on Advanced Learning Technologies*. Montreal, Quebec, pp. 809-813.

Blanchard, E. G., Volfson, B., Hong, Y.-J., and Lajoie, S. P. (2009). "Affective Artificial Intelliegnce in Education: From Detection to Adaptation", V. D. R. Mizoguchi, B. d. Boulay, and A. Grasser, (eds.), *International Conference on Artificial Intelligence in Education*. City: IOS Press, pp. 81-88.

Bloom, B. S. (1984). "The 2-Sigma Problem: The search for methods of group instruction as effective as one-to-one tutoring,." *Educational Researcher*, 13(6), 4-16.

Bohl, O., Scheuhase, J., Sengler, R., and Winand, U. "The sharable content object reference model (SCORM)-a critical review."

Bonnet, A. (1985). *Artificial Intelligence: Promise and Performance.*, London: Prentice Hall.

Bostanov, V. (2004). "BCI competition 2003-data sets Ib and IIb: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram." *Biomedical Engineering, IEEE Transactions on*, 51(6), 1057-1061.

Boulay, B. d., and Luckin, R. (2001). "Modelling Human Teaching Tactics and Strategies for Tutoring Systems." *International Journal of Artificial Intelligence in Education*, 12, 235-256.

Bower, G. H. (1992). "How might emotions affect learning?", in S. A.Christianson, (ed.), *The handbook of emotion and memory*. Hillsdale, NJ: Erlbaum, pp. 3 –31.

Bradley, M. M., Greenwald, M. K., Petry, M. C., and Lang, P. J. (1992). "Remembering Pictures: Pleasure and Arousal in Memory." *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18(2), 379-390.

Bransford, J. D., Brown, A. L., Cocking, R. R., Donovan, M. S., and Pellegrino, J. W. (2000). *How People Learn: Brain, Mind, Experience, and School*, Washington, D.C.: National Academy Press.

Brawner, K., and Goldberg, B. "Real-Time Monitoring of ECG and GSR Signals during Computer-Based Training."

Brawner, K., Sottilare, R., and Gonzalez, A. "Semi-Supervised Classification of Realtime Physiological Sensor Datastreams for Student Affect Assessment in Intelligent Tutoring."

Brawner, K. W., and Gonzalez, A. J. (2011). "Realtime Clustering of Unlabeled Sensory Data for User State Assessment"*Proceedings of International Defense & Homeland Security Simulation Workshop of the I3M Conference*. City: Rome, Italy.

Brawner, K. W., Holden, H. K., Goldberg, B. S., and Sottilare, R. A. "Understanding the Impact of Intelligent Tutoring Agents on Real-Time Training Simulations."

Brown, J. S., and VanLehn, K. (1980). "Repair theory: A generative theory of bugs in procedural skills." *Cognitive Science*, 4(4), 379-426.

Brusilovsky, P., Eklund, J., and Schwarz, E. (1998). "Web-based education for all: A tool for developing adaptive courseware", *Computer Networks and ISDN Systems (Proceedings of Seventh International World Wide Web Conference, 14-18 April 1998)*. pp. 291-300.

Burns, H. L., and Capps, C. G. (1988). "Foundations of intelligent tutoring systems: an introduction", in M. C. Poison and J. J. Richardson, (eds.), *Foundations of Intelligent Tutoring Systems*. London: Lawrence Erlbaum Associates, Inc., pp. 1-19.

Burton, R., and Brown, J. (1976). "A tutoring and student modelling paradigm for gaming environments"*SIGCSE-SIGCUE Joint Symposium on Computer Science Education. .* City.

Calvo, R. A., and D'Mello, S. (2012). "Frontiers of Affect-Aware Learning Technologies." *Intelligent Systems, IEEE*, 27(6), 86-89.

Campbell, J. "Theorising habits of mind as a framework for learning." *Presented at Proceedings of the Australian Association for Research in Education Conference*.

Candes, E., Demanet, L., Donoho, D., and Ying, L. (2006). "Fast discrete curvelet transforms." *Multiscale Modeling & Simulation*, 5(3), 861-899.

Cannady, J., and Garcia, R. (2001). "The application of fuzzy ARTMAP in the detection of computer network attacks." *Artificial Neural Networks—ICANN 2001*, 225-230.

Capuano, N., Marsella, M., and Salerno, S. "ABITS: an agent-based intelligent tutoring system for distance learning." *Presented at International Workshop on Adaptive and Intelligent Web-based Educational Systems, International Conference on Intelligent Tutoring Systems (ITS 2000)*, Montreal, Canada.

Carole, R., and Hyokyeong, L. (2005). "Creating a pedagogical model that uses student self reports of motivation and mood to adapt ITS instruction."

Carpenter, G. A., and Grossberg, S. (1987). "A massively parallel architecture for a self-organizing neural pattern recognition machine." *Computer vision, graphics, and image processing*, 37(1), 54-115.

Carpenter, G. A., and Grossberg, S. (1995). "Adaptive resonance theory (ART)", in M. Arbib, (ed.), *The handbook of brain theory and neural networks*. Cambridge, MA: MIT press, pp. 79-82.

Carpenter, G. A., Grossberg, S., and Reynolds, J. H. (1991a). "ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network." *Neural networks*, 4(5), 565-588.

Carpenter, G. A., Grossberg, S., and Rosen, D. B. (1991b). "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system." *Neural networks*, 4(6), 759-771.

Carroll, M., Kokini, C., Champney, R., Sottilare, R., and Goldberg, B. (2011). "Modeling Trainee Affective and Cognitive State Using Low Cost Sensors", *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*. Orlando, FL.

Carroll, M., Kokini, C., Champney, R., Sottilare, R., and Goldberg, B. "Pending..." *Presented at Interservice/Interindustry Training Simulation and Education Conference 2012*, Orlando, FL.

Castro, J., Georgiopoulos, M., Demara, R., and Gonzalez, A. "A Partitioned Fuzzy ARTMAP Implementation for Fast Processing of Large Databases on Sequential Machines." *Presented at FLAIRS Conference*.

Cha, H., Kim, Y., Park, S., Yoon, T., Jung, Y., and Lee, J. H. "Learning styles diagnosis based on user interface behaviors for the customization of learning interfaces in an intelligent tutoring system."

Chalfoun, P., and Frasson, C. (2012). "Cognitive Priming: Assessing the Use of Non-conscious Perception to Enhance Learner's Reasoning Ability"*Intelligent Tutoring Systems 2012*. City: Springer: Crete, Greece, pp. 84-89.

Champney, R. K., and Stanney, K. M. "Using emotions in usability."

Chaouachi, M., Chalfoun, P., Jraidi, I., and Frasson, C. (2010). "Affect and mental engagement: towards adaptability for intelligent systems.", in H.W. Guesgen and R. C. Murray, (eds.), *Proceedings of the 23rd International Florida Artificial Intelligence Research Society Conference* Menlo Park, CA: AAAI Press., pp. 355-360.

Chaouachi, M., and Frasson, C. (2010). "Exploring the Relationship between Learner EEG Mental Engagement and Affect"*10th International Conference on Intelligent Tutoring Systems*. City: Springer Verlag: Pittsburgh, PA.

Charniak, E. (1991). "Bayesian Networks without tears." *AI Magazine*, 12(4), 50.

Chi, M. T. H. (1996). "Constructing self-explanations and scaffolded explanations in tutoring." *Applied Cognitive Psychology*, 10(7), 33-49.

Clarke, R. J., and Macrae, R. (1988). *Coffee: Physiology*: Kluwer Academic Pub.

Clearinghouse, W. W. (2008). "WWC procedures and standards handbook". City: Dec.

Cocea, M., Hershkovitz, A., and Baker, R. S. J. (2009). "The impact of off-task and gaming behaviors on learning: immediate or aggregate?".

Cohen, J. (1992). "Quantitative Methods in Psychology: A Power Primer." *Psychological Bulletin*, 112(1), 155-159.

Coles, M. G. H. (1989). "Modern mind-brain reading: psychophysiology, physiology, and cognition." *Psychophysiology*, 26(3), 251-269.

Conati, C. (2002). "Probabilistic assessment of user's emotions in educational games." *Journal of Applied Artificial Intelligence*, 16, 555-575.

Conati, C. (2010). "Bayesian Student Modeling. Studies in Computational Intelligence, 2010." *Advances in Intelligent Tutoring Systems*, 308, 281-299.

Conati, C. (2011). "Combining cognitive appraisal and sensors for affect detection in a framework for modeling user affect." *New Perspectives on Affect and Learning Technologies*, 71-84.

Conejo, R., Guzmán, E., Millán, E., Trella, M., Pérez-De-La-Cruz, J. L., and Ríos, A. (2004). "SIETTE: A Web–Based Tool for Adaptive Testing." *International Journal of Artificial Intelligence in Education*, 14, 1-33.

Considine, J., Li, F., Kollios, G., and Byers, J. "Approximate aggregation techniques for sensor databases." *Presented at 20th IEEE International Conference on Data Engineering*.

Cooper, D., Muldner, K., Arroyo, I., Woolf, B., and Burleson, W. (2010). "Ranking feature sets for emotion models used in classroom based intelligent tutoring systems." *User Modeling, Adaptation, and Personalization*, 135-146.

Cooper, D. G., Arroyo, I., and Woolf, B. P. (2011). "Actionable affective processing for automatic tutor interventions." *New Perspectives on Affect and Learning Technologies*, 127-140.

Corbett, A. T. (2001). "Cognitive Computer Tutors: Solving the Two-Sigma Problem", in M. Bauer, P. J.Gmytrasiewicz, __, and J. Vassileva., (eds.), *Proceedings of the 8th International Conference on User Modeling 2001 (UM '01)*. London, UK: Springer-Verlag pp. 137-147.

Craig, S. D., Graesser, A. C., Sullins, J., and Gholson, B. (2004). "Affect and learning: An exploratory look into the role of affect in learning with AutoTutor." *Journal of Educational Media*, 29(3), 241-250.

Craik, F. I., Govoni, R., Naveh-Benjamin, M., and Anderson, N. D. (1996). "The effects of divided attention on encoding and retrieval processes in human memory." *Journal of Experimental Psychology General*, 152(2), 159-180.

Crowder, N. A. (1959). "Automatic tutoring by means of intrinsic programming." *Automatic teaching: The state of the art*, 116.

Crowley, K., Sliney, A., Pitt, I., and Murphy, D. "Evaluating a brain-computer interface to categorise human emotional response." *Presented at Advanced Learning Technologies (ICALT), 2010 IEEE 10th International Conference on*.

Curtin, M. (1998). "Write once, run anywhere: Why it matters." *Technical Article. http://java. sun. com/features/1998/01/wo*.

Cuseo, J. (2007). "The empirical case against large class size: adverse effects on the teaching, learning, and retention of first-year students." *The Journal of Faculty Development*, 21(1), 5-21.

D'Mello, S., Graesser, A., and Picard, R. W. (2007). "Toward an affect-sensitive AutoTutor." *Intelligent Systems, IEEE*, 22(4), 53-61.

D Mello, S., and Graesser, A. (2007). "Mind and Body: Dialogue and posture for affect detection in learning environments." *Frontiers in Artificial Intelligence and Applications*, 158, 161.

437

D'Mello, S., and Graesser, A. (2007). "Mind and Body: Dialogue and Posture for Affect Detection in Learning Environments", in R. Luckin, K. Koedinger, and J. Greer, (eds.), *Proceedings of the 13th International Conference on Artificial Intelligence in Education (AIED 2007)*. Amsterdam, The Netherlands: IOS Press, pp. 161-168.

D'Mello, S. K., Taylor, R., and Graesser, A. C. (2007). "Monitoring Affective Trajectories during Complex Learning", in D. S. McNamara and J. G. Trafton, (eds.), *Proceedings of the 29th Annual Cognitive Science Society*. Austin, TX: Cognitive Science Society, pp. 203-208.

Dagum, P., and Luby, M. (1993). "Approximating probabilistic inference in Bayesian belief networks is NP-hard." *Artificial Intelligence*, 60(1), 141-153.

Dal Seno, B., Matteucci, M., and Mainardi, L. (2010). "Online detection of P300 and error potentials in a BCI speller." *Computational intelligence and neuroscience*, 2010, 11.

Dasgupta, S., Hsu, D., and Monteleoni, C. (2007). "A general agnostic active learning algorithm." *Advances in neural information processing systems*, 20, 353-360.

Dasgupta, S., and Langford, J. (2009). "Active Learning Tutorial, ICML 2009."

Davidson, R. J., Scherer, K. R., and Goldsmith, H. H. (2003). *Handbook of Affective Sciences*, New York: Oxford University Press.

Davis, L. (1991). *Handbook of genetic algorithms*: Van Nostrand Reinhold New York.

Demberg, V., Kiagia, E., and Sayeed, A. (2013). "The Index of Cognitive Activity as a Measure of Linguistic Processing." *reading time*, 500, 1500.

Dennerlein, J., Becker, T., Johnson, P., Reynolds, C., and Picard, R. W. "Frustrating computer users increases exposure to physical factors."

Derakhshan, N., and Eysenck, M. (2010). "Emotional states, attention, and working memory: a special issue of cognition & emotion." *Recherche*, 67, 02.

Dolan, B., and Behrens, J. (2012). "Five Aspirations for Educational Data Mining." *Proceedings of the 5th International Conference on Educational Data Mining*.

Donchin, E., Spencer, K. M., and Wijesinghe, R. (2000). "The mental prosthesis: assessing the speed of a P300-based brain-computer interface." *Rehabilitation Engineering, IEEE Transactions on*, 8(2), 174-179.

Dorigo, M., and Di Caro, G. "Ant colony optimization: a new meta-heuristic." *Presented at Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*.

Dorneich, M. C., Ververs, P. M., Mathan, S., and Whitlow, S. D. (2007). *Defense Advanced Research Projects Agency (DARPA) Improving Warfighter Information Intake under Stress: Augmented Cognition - Phases 2, 3, and 4. - Final rept. Jun 2003-Jan 2007*. Honeywell, Inc., Honeywell Laboratories., Minneapolis, MN.

Dragon, T., Arroyo, I., Woolf, B. P., Burleson, W., el Kaliouby, R., and Eydgahi, H. (2008). "Viewing Student Affect and Learning through Classroom Observation and Physical Sensors", in B. Woolf, E. Aimeur, R. Nikambou, and S. Lajoie, (eds.), *Intelligent Tutoring Systems: Proceedings of the 9th International Conference on Intelligent Tutoring Systems. LNCS*. Berlin: Springer-Verlag, pp. 29-39.

Duchi, J., Hazan, E., and Singer, Y. (2010). "Adaptive subgradient methods for online learning and stochastic optimization." *Journal of Machine Learning Research*, 12, 2121-2159.

Durlach, P. J. (1998). "The effects of a low dose of caffeine on cognitive performance." *Psychopharmacology*, 140(1), 116-119.

Eberhart, R., and Shi, Y. "Comparison between genetic algorithms and particle swarm optimization." *Presented at Evolutionary Programming VII*.

El Kaliouby, R., and Robinson, P. "Mind reading machines: Automated inference of cognitive mental states from video."

Engler, J., and Schnel, T. (2012). "Classifying Workload using Discrete Deterministic Nonlinear Models Across Subject Populations and for Extended Time." *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*.

Eysenck, M. W., and Calvo, M. G. (1992). "Anxiety and performance: The processing efficiency theory." *Cognition & Emotion*, 6(6), 409-434.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R. (1996). *Advances in knowledge discovery and data mining*.

Fayyad, U. M., Reina, C., and Bradley, P. S. (1998). "Refining Initialization of Expectation Maximization Clustering Algorithms"*4th International Conference on Knowledge Discovery and Data Mining (KDD-98)*. City: New York City.

Feng, M., Heffernan, N., and Koedinger, K. (2010). "Student Modeling in an Intelligent Tutoring System." *Intelligent Tutoring Systems in E-Learning Environments: Design, Implementation and Evaluation*, 208.

Fischer, G. (2001). "User modeling in human–computer interaction." *User Modeling and User-Adapted Interaction*, 11(1), 65-86.

Fisher, C. D. (1993). "Boredom at work: A neglected concept." *Human Relations*, 46, 395– 417.

Fletcher, J. D. (2011). *DARPA Education Dominance Program: April 2010 and November 2010 Digital Tutor Assessments*. Institute for Defense Analyses.

Folsom-Kovarik, J. T. (2012). *Leveraging Help Requests in POMDP Intelligent Tutoring Systems*, University of Central Florida.

Frasson, C., and Chalfoun, P. (2010). "Managing Learner's Affective States in Intelligent Tutoring Systems", in R. Nkambou, R. Mizoguchi, and J. Bourdeau, (eds.), *Advances in Intelligent Tutoring Systems*. Berlin-Heidelberg: Springer, pp. 339– 358.

Fredrickson, B. (1998). "What good are positive emotions?" *Review of General Psychology*, 2(3), 300-319.

Fritzke, B. (1995). "A growing neural gas network learns topologies." *Advances in neural information processing systems*, 7, 625-632.

Gagliolo, M., and Schmidhuber, J. (2006). "Learning dynamic algorithm portfolios." *Annals of Mathematics and Artificial Intelligence*, 47(3-4), 295-328.

García-Rodríguez, J., Flórez-Revuelta, F., and García-Chamizo, J. "Image compression using growing neural gas." *Presented at Neural Networks, 2007. IJCNN 2007. International Joint Conference on*.

Goldberg, B., Brawner, K., Sottilare, R., Tarr, R., Billings, D. R., and Malone, N. "Use of Evidence-based Strategies to Enhance the Extensibility of Adaptive Tutoring Technologies." *Presented at The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*.

Gonzalez, C. (2005). "The relationship between task workload and cognitive abilities in dynamic decision making." *Human Factors*, 47(1), 92-101.

Gowda, S., Pardos, Z., and Baker, R. "Content learning analysis using the moment-by-moment learning detector."

Graesser, A., Chipman, P., Haynes, B., and Olney, A. (2005). "AutoTutor: An intelligent tutoring system with mixed-initiative dialogue." *IEEE Transactions on Education*, 48(4), 612-618.

Graesser, A., Chipman, P., King, B., McDaniel, B., and D'Mello, S. (2007). "Emotions and learning with AutoTutor", in R. Luckin, K. Koedinger, and J. Greer, (eds.),

Proceedings of the 13th International Conference on Artificial Intelligence in Education (AIED 2007). Amsterdam, The Netherlands: IOS Press, pp. 569-571.

Graesser, A. C., Conley, M. W., and Olney, A. (2012). "Intelligent tutoring systems." *APA handbook of educational psychology. Washington, DC: American Psychological Association*.

Graesser, A. C., and D'Mello, S. (2012). "Moment-To-Moment Emotions During Reading." *The Reading Teacher*, 66(3), 238-242.

Graesser, A. C., Jackson, G. T., Mathews, E. C., Mitchell, H. H., Olney, A., Ventura, M., Chipman, P., Franceschetti, D., Hu, X., and Louwerse, M. M. "Why/AutoTutor: A test of learning gains from a physics tutor with natural language dialog."

Graesser, A. C., Person, N. K., Harter, D., and Group, T. R. (2001). "Teaching tactics and dialog in AutoTutor." *International Journal of Artificial Intelligence in Education*, 12(3), 257-279.

Graesser, A. C., Person, N. K., and Magliano, J. P. (1995). "Collaborative dialogue patterns in naturalistic one-to-one tutoring." *Applied Cognitive Psychology*, 9(6), 495-522.

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., and Kreuz, R. (1999). "AutoTutor: A simulation of a human tutor." *Cognitive Systems Research*, 1(1), 35-51.

Gross, R., Matthews, I., Cohn, J., Kanade, T., and Baker, S. (2010). "Multi-pie." *Image and Vision Computing*, 28(5), 807-813.

Guedalia, I. D., London, M., and Werman, M. (1998). "An on-line agglomerative clustering method for non-stationary data." *Neural Computation*.

Guo, H., and Hsu, W. "A survey of algorithms for real-time Bayesian network inference." *Presented at AAAI/KDD/UAI02 Joint Workshop on Real-Time Decision Support and Diagnosis Systems*.

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A. (2006). *Feature extraction: foundations and applications*: Springer.

Haddad, W. D. (1978). "Educational Effects of Class Size", *World Bank Staff Working Paper, No 280*. Washington D.C: World Bank.

Halverson, T., Estepp, J., Christensen, J., and Monnin, J. "Classifying Workload with Eye Movements in a Complex Task." *Presented at Proceedings of the Human Factors and Ergonomics Society Annual Meeting*.

Hanley, J. A. (1989). "Receiver operating characteristic (ROC) methodology: the state of the art." *Critical reviews in diagnostic imaging*, 29(3), 307.

Hanley, J. A., and McNeil, B. J. (1983). "A method of comparing the areas under receiver operating characteristic curves derived from the same cases." *Radiology*, 148(3), 839-843.

Harriott, C. E., Buford, G. L., Zhang, T., and Adams, J. A. "Assessing workload in human-robot peer-based teams." *Presented at Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*.

Hartley, J. R., and Sleeman, D. H. (1973). "Towards more intelligent teaching systems." *International Journal of Man-Machine Studies*, 5(2), 215-236.

Hassell, L. (2005). "Affect and trust." *Trust Management*, 271-284.

Haupt, R. L., and Haupt, S. E. (2004). *Practical genetic algorithms*: Wiley-Interscience.

He, Y., Hui, S. C., and Quan, T. T. (2009). "Automatic summary assessment for intelligent tutoring systems." *Computers & Education*, 53, 890-899.

Healey, J. (2011). "Recording Affect in the Field: Towards Methods and Metrics for Improving Ground Truth Labels ", in S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 107-116.

Heift, T. (2004). "Corrective feedback and learner uptake in CALL." *ReCALL*, 16(2), 416-431.

Hernandez, J., Morris, R. R., and Picard, R. W. (2011). "Call Center Stress Recognition with Person-Specific Models", S. D. Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. City: Springer-Verlag: Berlin Heidelberg, pp. 125-134.

Hewig, J., Hagemann, D., Seifert, J., Gollwitzer, M., Naumann, E., and Bartussek, D. (2005). "A revised film set for the induction of basic emotions." *Cognition and Emotion*, 19(7), 1095.

Hockenberry, M. "Simple tutors for hard problems: understanding the role of pseudo-tutors."

Hoens, T. R., Polikar, R., and Chawla, N. V. (2012). "Learning from streaming data with concept drift and imbalance: an overview." *Progress in Artificial Intelligence*, 1-13.

Hoffman, M. D., Blei, D. M., and Bach, F. (2010). "Online learning for latent dirichlet allocation." *Advances in neural information processing systems*, 23, 856-864.

Hofmann, T. (2001). "Unsupervised learning by probabilistic latent semantic analysis." *Machine Learning*, 42(1), 177-196.

Holland, J. H. (1992). "Complex adaptive systems." *Daedalus*, 17-30.

Holland, P. C., and Gallagher, M. (2006). "Different Roles for Amygdala Central Nucleus and Substantia Innominata in the Surprise-Induced Enhancement of Learning." *The Journal of Neuroscience*, 26(14), 3791-3797.

Hollenstein, T., McNeely, A., Eastabrook, J., Mackey, A., and Flynn, J. (2012). "Sympathetic and parasympathetic responses to social stress across adolescence." *Developmental Psychobiology*.

Holmstrom, J. (2002). "Growing neural gas". City: Uppsala University.

Holt, P., Dubs, S., Jones, M., and Greer, J. (1994). "The state of student modelling." *NATO ASI SERIES F COMPUTER AND SYSTEMS SCIENCES*, 125, 3-3.

Hothi, J., and Hall, W. "An evaluation of adapted hypermedia techniques using static user modelling."

Hu, X., Cai, Z., Han, L., Craig, S. D., and Wang, T. (2009). "Autotutor Lite", *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling*. Amsterdam, The Netherlands: IOS Press.

Hulten, G., Spencer, L., and Domingos, P. "Mining time-changing data streams."

Ingleton, C. (2000). "Emotion in learning - a neglected dynamic", in R. James, J. Milton, and R. Gabb, (eds.), *Research and Development in Higher Education, Cornerstones of Higher Education*. Melbourne, pp. 86-99.

IST. (2012). "MIX Testbed Project Description". City: www.active.ist.ucf.edu.

Jackson, P. (1990). *Introduction to expert systems*: Addison-Wesley Longman Publishing Co., Inc.

Jackson, T., Mathews, E., Lin, K., Olney, A., and Graesser, A. (2003). "Modeling student performance to enhance the pedagogy of autotutor.", in P. Brusilovsky, A.

Corbett, and F. d. Rosis, (eds.), *Proceedings of the 9th International Conference on User modeling (UM'03)*. Berlin, Heidelberg: Springer-Verlag, pp. 368-372.

Jain, A. K. (2008). "Data clustering: 50 years beyond k-means", *ECML PKDD '08: Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases - Part I,*. Berlin Heidelberg: Springer-Verlag, pp. 3-4.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). "Data clustering: a review." *ACM computing surveys (CSUR)*, 31(3), 264-323.

James, W. (1884). "What is Emotion?" *Mind, Brain, and Education*, 9, 188–205.

Jaques, P. A., Vicari, R., Pesty, S., and Martin, J.-C. (2011). "Evaluating a Cognitive-Based Affective Student Model", in S. D. Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 599-608.

Jennings, N. R. (2000). "On agent-based software engineering." *Artificial Intelligence*, 117(2), 277-296.

Johnson, R. R., Popovic, D. P., Olmstead, R. E., Stikic, M., Levendowski, D. J., and Berka, C. (2011). "Drowsiness/alertness algorithm development and validation using synchronized EEG and cognitive performance to individualize a generalized model." *Biological Psychology*.

Jones, D., Hale, K., Dechmerowski, S., and Fouad, H. "Creating Adaptive Emotional Experience During VE Training." *Presented at The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*.

Jones, E., Oliphant, T., and Peterson, P. (2001). "SciPy: Open source scientific tools for Python." *http://www. scipy. org/*.

Kahneman, D., and Beatty, J. (1966). "Pupil diameter and load on memory." *Science*.

Kapoor, A., and Picard, R. W. (2005a). "Multimodal affect recognition in learning environments." *ACM Multimedia*, 2005, 677-682.

Kapoor, A., and Picard, R. W. "Multimodal affect recognition in learning environments." *Presented at Proceedings of the 13th annual ACM international conference on Multimedia*.

Katz, S., Lesgold, A., Eggan, G., and Gordin, M. (1992). "Modelling the student in Sherlock II." *Journal of Artificial Intelligence in Education*, 3, 495-495.

Kim, Y., and Baylor, A. (2006). "A Social-Cognitive Framework for Pedagogical Agents as Learning Companions." *Educational Technology Research and Development,*, 54(6), 569-596.

Kirschner, P., Sweller, J., and Clark, R. E. (2006). "Why unguided learning does not work: An analysis of the failure of discovery learning, problem-based learning, experiential learning and inquiry-based learning." *Educational Psychologist*, 41(2).

Klašnja-Milićevića, A., Vesina, B., Ivanovićb, M., and Budimac, Z. (2011). "E-Learning personalization based on hybrid recommendation strategy and learning style identification." *Computers & Education*, 56(3), 885-899.

Kleinsmith, L. J., and Kaplan, S. (1963). "Paired-associate learning as a function of arousal and interpolated interval." *Journal of Experimental Psychology General*, 65(2), 190-193.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., and Mark, M. A. (1997). "Intelligent Tutoring Goes To School in the Big City." *International Journal of Artificial Intelligence in Education*, 8, 30-43.

Kohonen, T. (1982). "Self-organized formation of topologically correct feature maps." *Biological cybernetics*, 43(1), 59-69.

Kokini, C., Carroll, M., Ramirez-Padron, R., Hale, K., Sottilare, R., and Goldberg, B. "Quantification of trainee affective and cognitive state in real-time." *Presented at The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*.

Koranne, S. (2011). "Artificial Intelligence and Optimization." *Handbook of Open Source Tools*, 391-408.

Koren, Y. "Factorization meets the neighborhood: a multifaceted collaborative filtering model."

Kort, B., Reilly, R., and Picard, R. W. "An affective model of interplay between emotions and learning: Reengineering educational pedagogy-building a learning companion."

Kotthoff, L., Gent, I. P., and Miguel, I. "A preliminary evaluation of machine learning in algorithm selection for search problems." *Presented at Fourth Annual Symposium on Combinatorial Search*.

Kraiger, K., Ford, J. K., and Salas, E. (1993). "Application of cognitive, skill-based, and affective theories of learning outcomes to new methods of training evaluation." *Journal of Applied Psychology*, 78(2), 311.

Lalonde, M., Byrns, D., Gagnon, L., Teasdale, N., and Laurendeau, D. "Real-time eye blink detection with GPU-based SIFT tracking." *Presented at Computer and Robot Vision, 2007. CRV'07. Fourth Canadian Conference on*.

Lane, C., Noren, D., Auerbach, D., Birch, M., and Swartout, W. (2011). "Intelligent Tutoring Goes to the Museum in the Big City: A Pedagogical Agent for Informal Science Education," *Lecture Notes in Computer Science*. pp. 155-162.

Langford, J., Karampatziakis, N., Hsu, D., and Hoffman, M. (2010). *Vowpal Wabbit 5.0*.

Langford, J., Li, L., and Strehl, A. (2007). *Vowpal wabbit online learning*. Technical report.

Langford, J., Li, L., and Zhang, T. (2009). "Sparse online learning via truncated gradient." *The Journal of Machine Learning Research*, 10, 777-801.

Laparra-Hernández, J., Belda-Lois, J., Medina, E., Campos, N., and Poveda, R. (2009). "EMG and GSR signals for evaluating user's perception of different types of ceramic flooring." *International Journal of Industrial Ergonomics*, 39(2), 326-332.

LeCun, Y., Bottou, L., Orr, G., and Müller, K. (1998). "Efficient backprop." *Neural networks: Tricks of the trade*, 546-546.

Lepper, M., and Hodell, M. (1989). "Intrinsic motivation in the classroom", in C. Ames and R. E. Ames, (eds.), *Research on Motivation in Education Vol. 3*. New York: Academic Press, pp. 73-105.

Lepper, M., and Woolverton, M. (2002). "The Wisdom of Practice: Lessons Learned from the Study of Highly Effective Tutors", in J. Aronson, (ed.), *Improving academic achievement: impact of psychological factors on education* New York: Academic Press, pp. 135-158.

Lepper, M. R., Woolverton, M., Mumme, D. L., and Gurtner, J. (1993). "Motivational techniques of expert human tutors: Lessons for the design of computer-based tutors", in S. P. Lajoie and S. J. Derry, (eds.), *Computers as cognitive tools*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc., pp. 75-105.

Lesta, L., and Yacef, K. (2002). "An Intelligent Teaching-Assistant System for Logic", in S. Cerri and F. Paraguo, (eds.), *Proceedings of Intelligent Tutoring Systems*. Biarritz, France.

Lester, J. C. (2011). "Affect, Learning, and Delight", in S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 2.

Lester, J. C., Towns, S. G., and Fitzgerald, P. J. (1999). "Achieving affective impact: Visual emotive communication in lifelike pedagogical agents." *International Journal of Artificial Intelligence in Education*, 10(3-4), 278–291.

Likert, R. (1932). "A technique for the measurement of attitudes." *Archives of psychology*.

Luger, G. F. (2005). *Artificial intelligence: Structures and strategies for complex problem solving*: Addison Wesley Longman.

Lyster, R., and Ranta, L. (1997). "Corrective feedback and learner uptake." *Studies in second language acquisition*, 19(01), 37-66.

Madria, S., Bhowmick, S., Ng, W. K., and Lim, E. (1999). "Research issues in web data mining." *DataWarehousing and Knowledge Discovery*, 805-805.

Majumdar, A., and Ochieng, W. Y. (2002). "Factors affecting air traffic controller workload: Multivariate analysis based on simulation modeling of controller workload." *Transportation Research Record: Journal of the Transportation Research Board*, 1788(-1), 58-69.

Malik, M., Bigger, J., Camm, A., Kleiger, R., Malliani, A., and Moss, A. (1996). "Heart rate variability." *Circulation*, 93(5), 1043-1065.

Marshall, S. P. "The index of cognitive activity: Measuring cognitive workload." *Presented at Human factors and power plants, 2002. proceedings of the 2002 IEEE 7th conference on*.

Marshall, S. P. (2007). "Identifying cognitive state from eye metrics." *Aviation, space, and environmental medicine*, 78(Supplement 1), B165-B175.

Martens, D., Baesens, B., and Fawcett, T. (2011). "Editorial survey: swarm intelligence for data mining." *Machine Learning*, 82(1), 1-42.

Martinetz, T. (1993). "Competitive Hebbian learning rule forms perfectly topology preserving maps."

Martinetz, T., and Schulten, K. (1991). *A" neural-gas" network learns topologies*: University of Illinois at Urbana-Champaign.

447

Mason, B. J., and Bruning, R. (2001). "Providing Feedback in Computer-Based Instruction: What the Research Tells Us". City: Center of Instructional Innovation: University of Nebraska-Lincoln.

McMahan, H. B., and Streeter, M. (2010). "Adaptive bound optimization for online convex optimization." *arXiv preprint arXiv:1002.4908.*

McQuiggan, S., Lee, S., and Lester, J. (2007). "Early prediction of student frustration." *Affective Computing and Intelligent Interaction*, 698-709.

Medina, J. (2008). *Brain Rules: 12 Principles for Surviving and Thriving at Work, Home, and School*: Pear Press.

Meireles, M. R., Almeida, P. E., and Simões, M. G. (2003). "A comprehensive review for industrial applicability of artificial neural networks." *Industrial Electronics, IEEE Transactions on*, 50(3), 585-601.

Miller, G. A., Levin, D. N., Kozak, M. J., Cook III, E. W., McLean Jr, A., and Lang, P. J. (1987). "Individual differences in imagery and the psychophysiology of emotion." *Cognition and Emotion*, 1(4), 367-390.

Mitrovic, A., Martin, B., and Suraweera, P. (2007). "Intelligent tutors for all: Constraint-based modeling methodology, systems and authoring." *IEEE Intelligent Systems*, 22, 38-45.

Mitrovic, A., and Ohlsson, S. (1999). "Evaluation of a constraint-based tutor for a database language."

Mitrovic, A., Suraweera, P., Martin, B., Zakharov, K., Milik, N., and Holland, J. "Authoring constraint-based tutors in ASPIRE."

Monajati, M., Abbasi, S. H., Shabaninia, F., and Shamekhi, S. (2012). "Emotions States Recognition Based on Physiological Parameters by Employing of Fuzzy-Adaptive Resonance Theory." *International Journal of Intelligence Science*, 2(24), 166-176.

Mone, G. (2011). "2011 Invention Awards: A Mirror That Monitors Vital Signs"*Popular Science*. City.

Moon, T. K. (1996). "The expectation-maximization algorithm." *Signal Processing Magazine, IEEE*, 13(6), 47-60.

Morrison, J. G., Kobus, D. A., and Brown, C. M. (2006). *DARPA Improving Warfighter Information Intake Under Stress –Augmented Cognition, Phase II: The Concept*

Validation Experiment (Technical Report 1940). SPAWAR Systems Center, San Diego, California.

Mota, S., and Picard, R. W. "Automated posture analysis for detecting learner's interest level." *Presented at Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on*.

Muldner, K., Burleson, W., Van de Sande, B., and VanLehn, K. (2011). "An analysis of students' gaming behaviors in an intelligent tutoring system: predictors and impacts." *User Modeling and User-Adapted Interaction*, 21(1), 99-135.

Murray, T., and Arroyo, I. (2002). "Toward Measuring and Maintaining the Zone of Proximal Development in Adaptive Instructional Systems"*International Conference on Intelligent Tutoring Systems*. City: Biarritz, France.

Murray, T., and Arroyo, I. "Toward an operational definition of the zone of proximal development for adaptive instructional software." *Presented at 25th Annual Meeting of the Cognitive Science Society* Boston, MA.

Navarro, D. J., Griffiths, T. L., Steyvers, M., and Lee, M. D. (2006). "Modeling individual differences using Dirichlet processes." *Journal of mathematical Psychology*, 50(2), 101-122.

Neches, R., Langley, P., and Klahr, D. (1987). *Learning, development, and production systems*: The MIT Press.

NeuroSky. (2007). City.

Nkambou, R. (2006). "Towards Affective Intelligent Tutoring System, Workshop on Motivational and Affective Issues in ITS", *8th International Conference on ITS*. pp. 5-12.

Nkambou, R. (2010). *Advances in Intelligent Tutoring Systems*: Springer Verlag.

Nurminen, M. L., Niittynen, L., Korpela, R., and Vapaatalo, H. (1999). "Coffee, caffeine and blood pressure: a critical review." *European journal of clinical nutrition*, 53(11), 831.

Nwana, H. S. (1990). "Intelligent Tutoring Systems: An Overview." *Artificial Intelligence Review* 4(4).

Ohlsson, S. (1994). "Constraint-based student modeling." *NATO ASI SERIES F COMPUTER AND SYSTEMS SCIENCES*, 125, 167-167.

Ok-choon, P., Ray, S. P., and Seidel, R., J. . (1987). "Intelligent CAI: old wine in new bottles or a new vintage?", *Artificial Intelligence and Instruction: Instruction and Methods*. Reading, MA: Addison-Wesley, pp. 11-43.

Olney, A., Graesser, A. C., and Person, N. K. (2010). "Tutorial Dialog in Natural Language", in R. Nkambou, J. Bourdeau, and R. Mizoguchi, (eds.), *Advances in Intelligent Tutoring Systems, Studies in Computational Intelligence*. Berlin: Springer-Verlag, pp. 181-206.

Pajares, F., and Miller, M. D. (1994). "Role of self-efficacy and self-concept beliefs in mathematical problem solving: a path analysis." *Journal of Educational Psychology*, 86, 193-203.

Palinko, O., Kun, A. L., Shyrokov, A., and Heeman, P. "Estimating cognitive load using remote eye tracking in a driving simulator." *Presented at Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*.

Palinscar, A. S., and Brown, A. L. (1984). "Reciprocal teaching of comprehension-fostering and comprehension-monitoring activities." *Cognition and instruction*, 1(2), 117-175.

Pan, J., and Tompkins, W. J. (1985). "A real-time QRS detection algorithm." *IEEE Trans. Biomed. Eng. BME*, 32(3), 230-236.

Parasuraman, R., and Caggiano, D. (2002). "Mental workload." *Encyclopedia of the human brain*, 3, 17-27.

Parasuraman, R., Cosenzo, K. A., and De Visser, E. (2009). "Adaptive automation for human supervision of multiple uninhabited vehicles: Effects on change detection, situation awareness, and mental workload." *Military Psychology*, 21(2), 270-297.

Patil, A. S., and Abraham, A. (2010). "Intelligent and Interactive Web-Based Tutoring System in Engineering Education: Reviews, Perspectives and Development", in F. Xhafa, S. Caballe, A. Abraham, T. Daradoumis, and A. J. Perez, (eds.), *Computational Intelligence for Technology Enhanced Learning. Studies in Computational Intelligence*. Berlin: Springer-Verlag., pp. 79-97.

Person, N., Klettke, B., Link, K., Kreuz, R., and Group, T. R. "The integration of affective responses into AutoTutor."

Person, N. K., and Graesser, A. C. (2003). "Fourteen Facts about Human Tutoring: Food for Thought for ITS Developers", in V. Aleven, U. Hoppe, J. Kay, R. Mizoguchi, H. Pain, F. Verdejo, and K. Yacef, (eds.), *Artificial Intelligence in Education 2003 Workshop Proceedings on Tutorial Dialogue Systems: With a View Toward the Classroom* Sydney, Australia pp. 335-344.

Pfurtscheller, G., and Klimesch, W. (1992). "Event-related synchronization and desynchronization of alpha and beta waves in a cognitive task", *Induced rhythms in the brain*. Springer, pp. 117-128.

Picard, R. (2006). "Building an Affective Learning Companion."*Keynote address at the 8th International Conference on Intelligent Tutoring Systems*. City: Jhongli, Taiwan.

Picard, R. (2011). "Measuring Affect in the Wild", in S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 3.

Pintrich, P. R., and De Groot, E. V. (1990). "Motivational and self-regulated learning components of classroom academic performance." *Journal of Educational Psychology*, 82, 33-40.

Pollock, V. E., Teasdale, T., Stern, J., and Volavka, J. (1981). "Effects of caffeine on resting EEG and response to sine wave modulated light." *Electroencephalography and clinical neurophysiology*, 51(5), 470-476.

Prudent, Y., and Ennaji, A. "An incremental growing neural gas learns topologies." *Presented at Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*.

Quah, J. T., and Sriganesh, M. (2008). "Real-time credit card fraud detection using computational intelligence." *Expert Systems with Applications*, 35(4), 1721-1732.

Raley, C., Stripling, R., Kruse, A., Schmorrow, D., and Patrey, J. "Augmented Cognition overview: Improving information intake under stress."

Rauh, R., Burkert, M., Siepmann, M., and Mueck-Weymann, M. (2006). "Acute effects of caffeine on heart rate variability in habitual caffeine consumers." *Clinical physiology and functional imaging*, 26(3), 163-166.

Reinerman-Jones, L., Barber, D., Lackey, S. J., and Nicholson, D. "Developing methods for utilizing physiological measures." *Presented at Applied Human Factors and Ergonomics Society Conference 2010*.

Rice, J. R. (1975). "The algorithm selection problem."

Rich, E., and Knight, K. (1991). "Artificial intelligence." *International Student Edition, MacGraw-Hill, London*.

Rickel, J. W. (1989). "Intelligent computer-aided instruction: A survey organized around system components." *Systems, Man and Cybernetics, IEEE Transactions on*, 19(1), 40-57.

Ridgway, J. (1988). "Of course ICAI is impossible, . . worse though, it might be seditious.", in J. A. Self, (ed.), *Artificial Intelligence and Human Learning: Intelligent computer-aided instruction*. London: Chapman & Hall, pp. 28-48.

Robison, J., McQuiggan, S., and Lester, J. "Evaluating the consequences of affective feedback in intelligent tutoring systems."

Robison, J., McQuiggan, S., and Lester, J. "Developing empirically based student personality profiles for affective feedback models."

Rodrigo, M. M. T., Baker, R., D'Mello, S., Gonzalez, M. C. T., Lagud, M. C. V., Lim, S. A. L., Macapanpan, A. F., Pascua, S. A. M. S., Santillano, J. Q., Sugay, J. O., Tep, S., and Viehland, N. J. B. (2007). "Comparing Learners' Affect While Using an Intelligent Tutoring System and a Simulation Problem Solving Game"*9th International Conference on Intelligent Tutoring Systems*. City: Montreal, Canada, pp. 40-49.

Rodrigo, M. M. T., and Baker, R. S. J. d. (2009). "Coarse-grained detection of student frustration in an introductory programming course.", *Proceedings of the Fifth International Workshop on Computing Education Research (ICER '09)*. New York, NY: ACM, pp. 75-80.

Roll, I., Aleven, V., McLaren, B. M., and Koedinger, K. R. I. p. (2011). "Metacognitive Practice Makes Perfect: Improving Students' Self-Assessment Skills with an Intelligent Tutoring System", *Artificial Intelligence in Education* Berlin / Heidelberg: Springer, pp. 288-295.

Romero, C., and Ventura, S. (2007). "Educational data mining: A survey from 1995 to 2005." *Expert Systems with Applications*, 33(1), 135-146.

Rosasco, L., Vito, E. D., Caponnetto, A., Piana, M., and Verri, A. (2004). "Are loss functions all the same?" *Neural Computation*, 16(5), 1063-1076.

Rowe, J., Shores, L., Mott, B., and Lester, J. C. (2010a). "Integrating Learning and Engagement in Narrative-Centered Learning Environments", in V. Aleven, J. Kay, and J. Mostow, (eds.), *Intelligent Tutoring Systems: Proceedings of the 10th International Conference on Intelligent Tutoring Systems. LNCS* Berlin: Springer, pp. 166-177.

Rowe, J. P., Shores, L. R., Mott, B. W., and Lester, J. C. (2010b). "A framework for narrative adaptation in interactive story-based learning environments.",

Proceedings of the Intelligent Narrative Technologies III Workshop (INT3 '10). New York, NY: ACM.

Ryu, K., and Myung, R. (2005). "Evaluation of mental workload with a combined measure based on physiological indices during a dual task of tracking and mental arithmetic." *International Journal of Industrial Ergonomics*, 35(11), 991-1009.

Sabourin, J., Mott, B., and Lester, J. C. (2011). "Generalizing Models of Student Affect in Game-Based Learning Environments", in S. D. Mello, A. Graesser, B. Schuller, and J.-C. Martin, (eds.), *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction (ACII 2011), LNCS*. Berlin Heidelberg: Springer-Verlag, pp. 588-597.

Sabourin, J., Rowe, J., Mott, B., and Lester, J. "Exploring inquiry-based problem-solving strategies in game-based learning environments."

Sabourin, J., Shores, L., Mott, B., and Lester, J. "Predicting student self-regulation strategies in game-based learning environments."

Scandura, J. M. (2011). "What TutorIT Can Do Better Than a Human and Why: Now and in the Future."

Schachter, J. (1991). "Corrective feedback in historical perspective." *Second Language Research*, 7(2), 89-102.

Schohn, G., and Cohn, D. "Less is more: Active learning with support vector machines." *Presented at MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*.

Schulze, R. (2004). *Meta-analysis: A comparison of approaches*: Hogrefe & Huber Publishers.

SeeingMachines. (2012). "faceLAB Brochure". City.

Shalev-Shwartz, S., Singer, Y., and Ng, A. Y. "Online and batch learning of pseudo-metrics." *Presented at Proceedings of the twenty-first international conference on Machine learning*.

Shute, V., and Glaser, R. (1991). "An intelligent tutoring system for exploring principles of economics." *Improving inquiry in social science: A volume in honor of Lee J. Cronbach*, 333-336.

Shute, V. J., and Psotka, J. (1994). *Intelligent Tutoring Systems: Past, Present, and Future*. DTIC Document.

Sidney, K. D., Craig, S. D., Gholson, B., Franklin, S., Picard, R., and Graesser, A. C. "Integrating affect sensors in an intelligent tutoring system."

Skinner, B. F. (1954). *The science of learning and the art of teaching*: Cambridge, Mass, USA.

Skinner, B. F. (1958). "Teaching machines." *Science*.

Slavin, R. E. (2002). "Evidence-based education policies: Transforming educational practice and research." *Educational Researcher*, 31(7), 15-21.

Small, R. V. (1996). "Dimensions of Interest and Boredom in Instructional Situations."

Soller, A. (2001). "Supporting social interaction in an intelligent collaborative learning system." *International Journal of Artificial Intelligence in Education (IJAIED)*, 12, 40-62.

Soriano, J., Rodrigo, M., Baker, R., Ogan, A., Walker, E., Castro, M., Genato, R., Fontaine, S., and Belmontez, R. "A cross-cultural comparison of effective help-seeking behavior among students using an ITS for math."

Sottilare, R. "Making a case for machine perception of trainee affect to aid learning and performance in embedded virtual simulations." *Presented at Proceedings of the NATO HFM-169 Research Workshop on the Human Dimensions of Embedded Virtual Simulation. Orlando, Florida*.

Sottilare, R. (2010). "Challenges in the development of intelligent tutors for adaptive military training systems", *International Training and Education Conference 2010*. London, England.

Sottilare, R., Goldberg, S., and Durlach, P. J. "Research Gaps for Adaptive and Predictive Computer-Based Tutoring Systems." *Presented at International Defense and Homeland Security Simulation Workshop (DHSS)*, Rome, Italy.

Sottilare, R. A., Brawner, K., Goldberg, B., and Holden, H. (2012a). "A Modular Framework to Support the Authoring and Assessment of Adaptive Computer-Based Tutoring Systems"*Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*. City: Orlando, FL.

Sottilare, R. A., Brawner, K. W., Goldberg, B. S., and Holden, H. K. (2012b). "The Generalized Intelligent Framework for Tutoring (GIFT)."

Sottilare, R. A., Holden, H. K., Brawner, K. W., and Goldberg, B. S. "Challenges and Emerging Concepts in the Development of Adaptive, Computer-based Tutoring Systems for Team Training."

Srivastava, J., Cooley, R., Deshpande, M., and Tan, P. N. (2000). "Web usage mining: Discovery and applications of usage patterns from web data." *ACM SIGKDD Explorations Newsletter*, 1(2), 12-23.

Stanley, K. O., and Miikkulainen, R. (2002). "Evolving neural networks through augmenting topologies." *Evolutionary computation*, 10(2), 99-127.

Steinhaus, H. (1957). "Sur la division des corps materiels en parties (in French)." *Bull. Acad. Polon. Sci.*, 4(12), 801-804.

Stevens, R., and Galloway, T. (2013). "Towards the Development of a Quantitative Descriptions of the Neurodynamic Rhythms and Organizations of Teams"*Human Factors and Ergonomic Society*. City.

Stevens, R. H., Galloway, T. L., Berka, C., Johnson, R., and Sprang, M. "Assessing Student's Mental Representations of Complex Problem Spaces with EEG Technologies."

Su, F., Xia, L., Cai, A., and Ma, J. "Evaluation of recording factors in EEG-based personal identification: A vital step in real implementations."

Subramaniam, S., Palpanas, T., Papadopoulos, D., Kalogeraki, V., and Gunopulos, D. "Online outlier detection in sensor data using non-parametric models." *Presented at Proceedings of the 32nd international conference on Very large data bases*.

Suppes, P. (1966). *The uses of computers in education*: Freeman.

Sutton, R. S., and Barto, A. G. (1998). *Reinforcement learning: An introduction*: Cambridge Univ Press.

Sykes, J., and Brown, S. "Affective gaming: measuring emotion through the gamepad."

Teoh, T.-T., Cho, S.-Y., and Nguwi, Y.-Y. "Emotional prediction using time series multiple-regression genetic algorithm for autistic syndrome disorder." *Presented at Computer Science & Education (ICCSE), 2012 7th International Conference on*.

Tononi, G., and Cirelli, C. (2006). "Sleep function and synaptic homeostasis." *Sleep medicine reviews*, 10(1), 49-62.

Trinh, V. (2009). *Contextualizing observational data for modeling human performance*: ProQuest.

Tsai, C.-F., Hsu, Y.-F., Lin, C.-Y., and Lin, W.-Y. (2009). "Intrusion detection by machine learning: A review." *Expert Systems with Applications*, 36(10), 11994-12000.

Tvarožek, J., and BIeliková, M. "The Friend: Socially-Intelligent Tutoring and Collaboration."

Uhr, L. "Teaching machine programs that generate problems as a function of interaction with students."

Van Der Linden, W. J., and Glas, C. A. W. (2000). *Computerized adaptive testing: Theory and practice*: Springer.

Vandewaetere, M., Desmet, P., and Clarebout, G. (2011). "The contribution of learner characteristics in the development of computer-based adaptive learning environments." *Computers in Human Behavior*, 27(1), 118-130.

VanLehn, K. (2011). "The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems." *Educational Psychologist*, 46(4), 197-221.

VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., and Wintersgill, M. (2005). "The Andes Physics Tutoring System: Five Years of Evaluations", G. McCalla and C. K. Looi, (eds.), *International Conference on Artificial Intelligence in Education*. City: IOS Press: Amsterdam, pp. 678-685.

Verdú, E., Regueras, L. M., Verdú, M. J., De Castro, J. P., and Pérez, M. A. (2008). "Is Adaptive Learning Effective? A Review of the Research", in L. Qing, S. Y. Chen, A. Xu, and M. Li, (eds.), *Proceedings of the 7th WSEAS International Conference on Applied Computer & Applied Computational Science (ACACOS '08)*. Stevens Point, Wisconsin: WSEAS Press, pp. 710-715.

Vogel-Walcutt, J., and Abich, J. (2011). "Using neurophysiological data to inform feedback timing: a pilot study." *Foundations of Augmented Cognition. Directing the Future of Adaptive Systems*, 265-274.

Vygotsky, L. S. (1978). *Mind in Society: The development of higher psychological processes*, Cambridge, MA: Harvard University Press.

Weiss, D. J., and Kingsbury, G. (1984). "Application of computerized adaptive testing to educational problems." *Journal of Educational Measurement*, 21(4), 361-375.

Welch, G., and Bishop, G. (1995). "An introduction to the Kalman filter". City.

Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge*, Los Altos, CA: Morgan Kaufmann Publishers, Inc.

Widrow, B., and Lehr, M. A. (1990). "30 years of adaptive neural networks: Perceptron, madaline, and backpropagation." *Proceedings of the IEEE*, 78(9), 1415-1442.

Wiemer-Hastings, P., Graesser, A., and Harter, D. "The foundations and architecture of AutoTutor."

Wikipedia. (2012). "Intelligent tutoring system"*Wikipedia*. City: http://en.wikipedia.org/wiki/Intelligent_tutoring_system.

Wiley, J., and Bailey, J. (2006). "Effects of collaboration and argumentation on learning from web pages." *Collaborative learning, reasoning, and technology*, 297-321.

Wine, J. (1971). "Test anxiety and direction of attention." *Psychological Bulletin*, 76(2), 92.

Winston, P. H. (1992). "Artificial intelligence." *Reading, Addison Wesley*.

Wixon, M., Baker, R., Gobert, J., Ocumpaugh, J., and Bachmann, M. (2012). "WTF? detecting students who are conducting inquiry without thinking fastidiously." *User Modeling, Adaptation, and Personalization*, 286-296.

Woeginger, G. (2003). "Exact algorithms for NP-hard problems: A survey." *Combinatorial Optimization—Eureka, You Shrink!*, 185-207.

Wolpert, D. H., and Macready, W. G. (1997). "No free lunch theorems for optimization." *Evolutionary Computation, IEEE Transactions on*, 1(1), 67-82.

Woods, P., and Hartley, J. (1971). "Some learning models for arithmetic tasks and their use in computer based learning." *British Journal of Educational Psychology*, 41(1), 38-48.

Woolf, B. (2009a). *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing e-Learning*, Burlington, MA: Elsevier.

Woolf, B., Burelson, W., and Arroyo, I. "Emotional intelligence for computer tutors." *Presented at Supplementary Proceedings of the 13th International Conference of Artificial Intelligence in Education*.

Woolf, B., Burleson, W., Arroyo, I., Dragon, T., Cooper, D., and Picard, R. (2009). "Affect-Aware Tutors: Recognising and Responding to Student Affect." *International Journal of Learning Technology*, 4(3/4), 129-164.

Woolf, B. P. (2009b). *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning*, Burlington, MA: Morgan Kaufmann.

Woolf, B. P. (2010). *A Roadmap for Education Technology*. National Science Foundation.

Yuval-Greenberg, S., Tomer, O., Keren, A. S., Nelken, I., and Deouell, L. Y. (2008). "Transient induced gamma-band response in EEG as a manifestation of miniature saccades." *Neuron*, 58(3), 429-441.

Zaïane, O. R., Xin, M., and Han, J. "Discovering web access patterns and trends by applying OLAP and data mining technology on web logs."

Zaki, S. M., and Yin, H. (2008). "A semi-supervised learning algorithm for growing neural gas in face recognition." *Journal of Mathematical Modelling and Algorithms*, 7(4), 425-435.

Zander, T. O., Kothe, C., Jatzev, S., and Gaertner, M. (2010). "Enhancing human-computer interaction with input from active and passive brain-computer interfaces." *Brain-Computer Interfaces*, 181-199.

Zhang, T., and Oles, F. "The value of unlabeled data for classification problems." *Presented at Proceedings of the Seventeenth International Conference on Machine Learning,(Langley, P., ed.)*.

Zhang, Z. (2012a). "Microsoft Kinect Sensor and Its Effect." *IEEE MultiMedia*, 19(2), 4-10.

Zhang, Z. (2012b). "Microsoft Kinect Sensor and Its Effect." *Multimedia, IEEE*, 19(2), 4-10.

Zhu, X. (2005). "Semi-supervised learning literature survey."

Zwaan, R. A., and Singer, M. (2003). "Text comprehension."