

# An overview of concept drift applications

Indrė Žliobaitė<sup>1,2</sup>, Mykola Pechenizkiy<sup>3</sup>, and João Gama<sup>4</sup>

<sup>1</sup> Dept. of Computer Science, Aalto University, Finland

<sup>2</sup> Helsinki Institute for Information Technology, Finland

<sup>3</sup> Eindhoven University of Technology, The Netherlands

<sup>4</sup> LIAAD, INESC TEC and University of Porto, Portugal

indrė.zliobaite@aalto.fi, m.pechenizkiy@tue.nl, jgama@fep.up.pt

**Abstract.** In most challenging data analysis applications, data evolve over time and must be analyzed in near real time. Patterns and relations in such data often evolve over time, thus, models built for analyzing such data quickly become obsolete over time. In machine learning and data mining this phenomenon is referred to as concept drift. The objective is to deploy models that would diagnose themselves and adapt to changing data over time. This chapter provides an application oriented view towards concept drift research, with a focus on supervised learning tasks. First we overview and categorize application tasks for which the problem of concept drift is particularly relevant. Then we construct a reference framework for positioning application tasks within a spectrum of problems related to concept drift. Finally, we discuss some promising research directions from the application perspective, and present recommendations for application driven concept drift research and development.

**Keywords:** evolving data, adaptation, concept drift, data streams

*We dedicate this chapter to Dr. Alexey Tsymbal who passed away suddenly and unexpectedly in November 2014 at age of 39. Alexey contributed to the progress of data mining and medical informatics on several topics, including notable work on handling concept drift.*

---

## 1 Introduction

Realism of the perfect world assumptions in machine learning has been challenged years ago [30]. One of these challenges relates to an observation that in the real world the data tends to change over time. As a result, predictions of the models trained in the past may become less accurate as time passes or opportunities to improve the accuracy might be missed. Thus, learning models need

to have mechanisms for continuous diagnostics of performance, and be able to adapt to changes in data over time.

In machine learning, data mining and predictive analytics unexpected changes in underlying data distribution over time are referred to as concept drift [27, 71, 58, 73]. In pattern recognition the phenomenon is known as covariate shift or dataset shift [58]. In signal processing the phenomenon is known as non-stationarity [36]. Changes in underlying data occur due to changing personal interests, changes in population, adversary activities or they can be attributed to a complex nature of the environment.

The traditional supervised learning assumes that the training and the application data come from the same distribution, as illustrated in Figure 1 (a). In real life the predictions need to be made online, often in real time. An on-line setting brings additional challenges, since it may be expected for the data distribution to change over time. Thus, at any point in time the testing data may be coming from a different distribution than the training data has come, as illustrated in Figure 1.

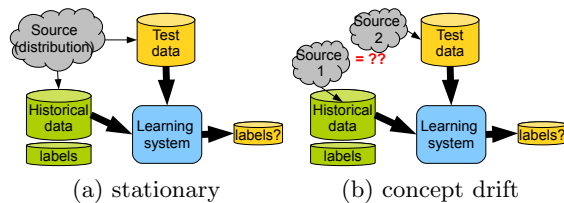


Fig. 1: Stationary supervised learning (a) and learning under concept drift (b).

The problem of concept drift is of increasing importance as more and more data is organized in the form of data streams rather than static databases, and it is unrealistic to expect that data distributions stay stable over a long period of time. It is not surprising that the problem of concept drift has been studied in several research communities including but not limited to pattern mining, machine learning and data mining, data streams, information retrieval, and recommender systems. Different approaches for detecting and handling concept drift have been proposed in research literature, and many of them have already proven their potential in a wide range of application domains.

One of the most illustrative cases, is learning against an adversary (e.g. spam filters, intrusion detection). A predictive model aims at identifying patterns characteristic of the adversary activity, while the adversary is aware that adaptive learning is used, and tries to change the behavior. Another context is learning in the presence of hidden variables. User modelling is one of the most popular learning tasks, where the learning system constructs a model of the user intentions, which of course are not observable and may change time to time. Drift also occurs in monitoring tasks and predictive maintenance. Learning the behaviour

of a system (e.g. the quality of products in industrial process) where degradation or corrosion of mechanical pieces occur over time.

Concept drift is used as a generic term to describe computational problems with changes over time. These changes may be of countless different types and there are different types of applications that call for different adaptation techniques. Thus, a solution “one-size-fits-all” is hardly possible and not desirable for handling concept drift. On the other hand, real application tasks being seemingly different from each other may share common properties and may have similar needs for adaptation. In order to transfer adaptive techniques from application to application we need to have means to characterize application tasks in a systematic manner.

The main aim and contribution of this chapter is to present tools for describing application tasks with concept drift in a systematic way, to position the existing application driven work using these tools, and define promising directions for future research. To keep the focus on applications we leave a detailed discussion of concept drift handling methods out of the scope of this paper, a reader is referred to existing reviews of the methods and techniques [27, 71, 58, 41]. Our study focuses on describing the *research tasks* driven by application needs.

The chapter is organized as follows. In Section 2 we discuss knowledge discovery process in the context of learning from streaming data and handling concept drift. Section 3 presents a reference framework of concept drift tasks and applications. This framework is intended to serve as a tool for describing an application oriented task in a systematic way. In Section 4 we survey application oriented published work on adaptive learning, focusing on task formulations, while leaving the techniques out of the scope of this study. Section 5 gives our recommendations towards promising and urgent future research directions from the concept drift application perspective, and concludes the study.

## 2 Knowledge discovery process and industry standards

In the era of big data, many data mining projects shift their emphasis towards evolving nature of the data that requires to study the automation of feedback loops more thoroughly. In the standard data mining and machine learning settings the majority of algorithmic techniques have been researched and developed under the assumption of identical and independent data distribution (IID). In big data applications data arrives in a stream, and patterns in the data are expected to evolve over time, therefore, it is not practical, and often is not feasible to involve a data mining expert to monitor the performance of the models and to retrain the models every time they become outdated. Therefore, interest towards automating development and update of predictive models in the streaming data settings has been increasing.

CRISP-DM model [12] describes the classical data mining process, where the life cycle of a data mining project spans over six phases: business understanding, data understanding, data preparation, modeling, evaluation and deployment.

Reinartz’s framework [65] follows CRISP-DM with some modifications, making modeling steps more explicit. The high-level process steps are summarized in Figure 2.

Business understanding phase aims at formulating business questions, and translating them into data mining goals. Data understanding phase aims at analyzing and documenting the available data and knowledge sources in the business according to the formulated goals, and providing initial characterization of data. Data preparation phase starts from target data selection that is often related to the problem of building and maintaining useful data warehouses. After selection, the target data is preprocessed in order to reduce the level of noise, pre-process the missing information, reduce data, and remove obviously redundant features. Next, data exploration phase aims at providing the first insight into the data, evaluate the initial hypotheses, usually, by means of descriptive statistics and visualization techniques. Data mining phase covers selection and application of data mining techniques, initialization and further calibration of their parameters to optimal values. Evaluation phase typically considers offline evaluation on historical data. In predictive modeling, one would typically analyze simulated performance of the data mining system with respect to some suitable measures of accuracy (such as precision, recall, or AUC, among others), or utility (for instance, expressed as cost-sensitive classification). Finally, the most promising predictive model is deployed in operational settings, and the performance is regularly followed up.

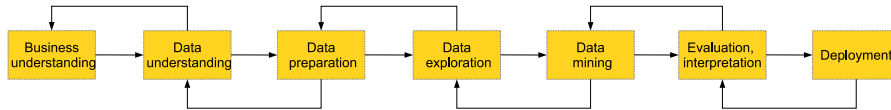


Fig. 2: Knowledge discovery process: from problem understanding to deployment. Arrows indicate the most important and frequent dependencies between the phases.

CRISP-DM model assumes that most of the data mining processes, including data cleaning, feature engineering, algorithm and parameter selection, and final evaluation, performed offline. If anything goes wrong with the deployed model, a data mining expert would analyze the problem, and try to fix it revisiting one or more steps in the process, and retraining the model.

In the streaming settings, it is common to expect changes in data and model applicability. Therefore, monitoring of model performance and model update or relearning becomes a natural and core part of the data mining process. Figure 3 presents our view towards adaptive data mining process. The main difference with the standard process is that now data preparation, mining, and evaluation steps are automated, there is no manual data exploration, and there is automated monitoring of performance, including change detection and alert services, after deployment.

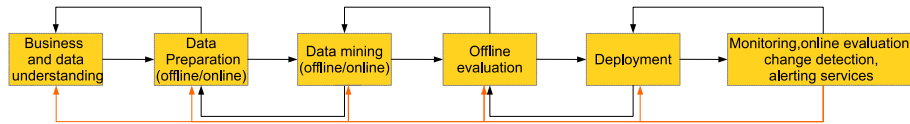


Fig. 3: Towards CRISP for Adaptive Data Mining.

Different strategies for updating learning models have been developed. Two main strategies can be distinguished. Learning models may evolve continuously, for instance, models can be periodically retrained using a sliding window of a fixed size over the past data (e.g. FLORA1 [73]). Alternatively, learning models may use trigger mechanisms, to initiate a model update. Typically, statistical change detection tests are used as triggers (e.g. [26]). Incoming data is continuously monitored, if changes are suspected, the trigger issues an alert, and adaptive actions are taken. When a change is signalled, the old training data is dropped and the model is updated using the latest data.

Learning systems can use single models or ensembles of models. Single model algorithms employ only one model for decision making at a time. Once the model is updated, the old one is permanently discarded. Ensembles, on the other hand, maintain some memory of different concepts. The prediction decisions are made either fusing the votes casted by different models or nominating the most suitable model for the time being from the pool of existing models.

Ensembles can be evolving or have trigger mechanisms as well. Evolving ensembles build and validate new models as new data arrives, the rule for model combination is dynamically updated based on the performance (e.g. [55]). Ensembles with triggers proactively assign the most relevant models for decision making based on the context (e.g. [72]).

Table 1 summarizes the taxonomy of adaptive learning strategies.

Table 1: Adaptive learning strategies.

	with triggers	evolving
single model	detectors	forgetting
multiple models	contextual	dynamic ensembles

An important aspect with respect to evaluation of performance of adaptive learning models relates to data collection. An adaptive system collects data, which is biased with respect to adaptations performed. For example, consider a recommender system, where so called “rich-gets-richer” phenomenon boosts the popularity of already popular items. In such situations relying on learning and evaluation of models on offline data is particularly dangerous, since within-system data does not give an unbiased view towards outside world. Consequently, it is important to develop techniques allowing for online evaluation and online adaptation.

Overall, we are not aware of fully automated and functioning adaptive learning system. It could be that well functioning fragments of such systems already exist in industry, especially in big data (web sized) data analysis, where manual attendance to all the running models is simply infeasible. In academia, except for some isolated cases (e.g. [10]), there has been little attention towards automating data mining process for big data, and we anticipate seeing more of such research efforts in the future.

In the following section we first categorize different big data applications where handling of concept drift is important and then refer to different data mining techniques that are suitable for data preprocessing, predictive modeling and evaluation in the streaming settings.

### 3 Categorization of concept drift tasks and applications

We start this section by describing relations between concept drift tasks and applications. We analyze application tasks in three steps:

- (a) properties of tasks,
- (b) landscape of applications,
- (c) links between tasks and applications.

The following subsections describe each component.

#### 3.1 Characterization of application tasks

Real application tasks, where concept drift is expected, can be mapped into three dimensions: (i) a type of the learning task, (ii) environment from which data comes, and (iii) online operational settings.

**Data and task.** Different types of tasks may be required depending on the intended application (even using the same data source): regression, ranking, classification, novelty detection, clustering, itemset mining.

Prediction makes assertions about the future, or about unknown characteristics of the present. Predictions is probably the most common use of data mining, it covers regression and classification tasks. Regression is typically considered in demand planning, resource scheduling optimization, user modelling, and, generally, in applications, in which the main objective is to anticipate future behavior of customers. Ranking is a special form of prediction, where partial ordering of alternative choices is required. Classification is a typical task in diagnosis and decision support, for example, antibiotic resistance prediction, e-mail spam classification, or news categorization. Ranking is a common task in recommendation, information retrieval, credit scoring and preference learning systems domains. Regression, ranking and classification are supervised learning tasks, where models are trained on examples, where the ground truth is available.

Novelty detection is a common task in fault, fraud detection applications, or identifying abnormal behavior. Faults in machines, frauds in credit cards

transactions, intrusion detection in computer networks, emergent topics in text news, requires some sort of outlier or anomaly detection, which is a basic form of novelty detection. Novelty detection is a semi-supervised, or unsupervised learning task. Typically, normal examples are available, but abnormal examples are unknown.

Clustering produces a grouping of people or objects, and is a popular task, for instance, in marketing. Itemset mining aims at finding items that commonly appear together, the task is relevant, for instance, in analyzing shopping baskets in retail. Patterns may evolve in those groups, new groups may appear or disappear due to changes in the data generating process. Clustering and itemset mining belong to unsupervised learning tasks, the ground truth is not known.

Orthogonally to different learning tasks, input data may have different forms. Data can be single or multi-relational, sequential, time series, general graph or particular complex structure, bags of instances or a mix. Instances can be noisy or highly accurate. Relational data can be of low or high dimensionality, have a few or lots of missing value, be almost complete or very sparse, have binary, categorical, ordered or numerical attributes.

Moreover, input data can be organized in different ways in terms of its accessibility. Data can come as a stream of instances or batches, or it can arrive in time-stamped batches. Data re-access can be allowed, or a single pass over the data may need to be strictly enforced. There might be randomly or systematically missing values in the incoming data.

**Characteristics of changes.** When designing adaptive learning systems one needs to consider, what is the source of drift in data, as different adaptive learning algorithms may be better suited for handling different types of changes. Data may change due to evolution in individual preferences (a person used to like accordion and jazz music earlier, but does not like it any more), a population change (in time of a crisis everybody tend to get lower salaries), adversary actions (new actions are tried to overcome the security system aiming to commit credit card frauds), or complexity of the environment (in automated vehicle navigation the environment is so complex that it is not feasible to take into account all possibilities of landscape deterministically, thus the environment is assumed to be changing).

In addition to types of drifts, it is important to consider, in which patterns changes are expected to occur in the future. Patterns of changes can be categorized according to the transition speed from one concept to another into sudden, or gradual. A drift can include a combination of multiple changes, for instance incremental drift features small steps of sudden changes, resulting in a trend. In terms of reoccurrence drifts can be categorized into novel, or reoccurring concepts.

Finally, it is advisable to consider, to what extent future changes may be predictable in a particular application. Concept drift can be completely unpredictable (e.g. evolution of the financial markets), somewhat predictable or identifiable (e.g. an upcoming financial crisis may be anticipated using a signal from

external early warning systems), or the environment might be well identifiable due to seasonality, or reoccurring contexts (e.g. an increase sales of ice-cream in summer).

**Operational settings** determine availability of the ground truth in an online operation, such as, arrival of true labels in classification, or true target values in regression tasks. Labels may become known immediately in the next time step after casting the prediction (e.g. food sales prediction). Labels may arrive within a fixed or variable time lag (in credit scoring typically the horizon of bankruptcy prediction is fixed, for instance, to one year, thus true labels become known after one year has passed). Alternatively, the setting may allow to obtain labels on demand (e.g. in spam categorization we can ask the user the true status of a given message).

Requirements for the speed of decision making need to be considered when selecting, which algorithms to deploy. In some applications prediction decisions may be required immediately (fraud detection), the sooner the better, while for other analytical decisions timing may be more flexible (e.g. credit scoring decision may reasonably take one-two weeks).

The cost of errors is an aspect to consider when selecting an evaluation metric for monitoring of performance. In traditional supervised learning different types of errors (e.g. false positives, false negatives) may resolve to different losses. In some applications prediction accuracy may be the main performance metric (e.g. in online mass flow prediction), in other applications accurate and timely identification of changes as well as accurate prediction are important (e.g. in demand prediction). In the online setting, discrepancies in time may as well have associated error costs (for instance, too early prediction of a peak in food sales would still allow to sell the extra products later, but too late prediction would lead to throwing away the excess products).

Finally, the ground truth labels may be objective based on clearly defined and accepted rules (e.g. bankrupt or not bankrupt company) or subjective, based on a personal opinion (e.g. interesting or not interesting article). Alternatively, the true labels may not be available at all being impossible or too costly to measure or define in a direct way.

Table 2 summarizes the identified properties of the concept drift application tasks. The identified properties are relevant for describing the type of task, the associated environment and the operational settings of an application under consideration. This information is essential to determine the characteristics that the adaptive learning system needs to possess, the properties that must be prioritized when designing such a system and the evaluation criteria of the system performance.

### 3.2 A landscape of concept drift application areas

Now as we have identified the properties that characterize concept drift application tasks, our next goal is to categorize application areas, and present typical applications for each category.



Table 2: Summary of properties of concept drift applications.

Data and task	<p><i>task</i>: prediction, classification, detection, clustering, itemset mining;  <i>input data type</i>: time series, relational, graph, bags or mix;  <i>incoming data</i>: stream, batches, collection iterations on demand;  <i>complexity</i>: volume; multiple scans; dimensionality;  <i>missing values</i>: unlikely, random, systematic;</p>
Characteristics of changes	<p><i>change source</i>: adversary activities, changes of preferences, population change, complex environment;  <i>change type</i>: sudden, incremental, gradual, reoccurring;  <i>change expectation</i>: unpredictable, predictable, identifiable (meta);  <i>change visibility</i>: direct/indirect; visual inspection, ground truth, external source, result of statistical hypothesis testing;</p>
Operational settings	<p><i>label availability</i>: real time, on demand, fixed lag, variable lag;  <i>decision speed</i>: real time, analytical;  <i>costs of mistakes</i>: balanced, unbalanced;  <i>true labels</i>: objective, subjective;</p>

We recall application domains, where data mining already plays an important role, or it has a high potential to be deployed. For surveying and summarizing the application domains we combine the taxonomies from the ACM classification<sup>5</sup> and KDnuggets polls<sup>6</sup>.

Table 3 presents our categorization of applications within the identified industries. We group different application areas into three application blocks:

- (a) monitoring and control,
- (b) information management, and
- (c) analytics and diagnostics.

For a compact representation each industry (rows) is assigned a group of applications that share common supervised learning tasks. As it can be seen from the table, for each of the industries or groups of industries, more than one application type can be relevant.

The *monitoring and control* block mostly relates to the detection tasks, where an abnormal behavior needs to be signaled. It includes such tasks as detection of adversary activities on the web, computer networks, telecommunications, financial transactions. In most of these tasks the normal behavior is modeled and the goal is to alarm when an abnormal behavior is observed.

<sup>5</sup> <http://www.acm.org/about/class/ccs98-html>

<sup>6</sup> <http://www.kdnuggets.com/polls/2010/analytics-data-mining-industries-applications.html>

Table 3: Categorization of applications by type and industry.

<b>Appl.</b> <b>Indust.</b>	<b>Monitoring and control</b>	<b>Information management</b>	<b>Analytics and diagnostics</b>
<b>Security, Police</b>	fraud detection, insider trading detection, adversary actions detection	next crime place prediction	crime volume prediction
<b>Finance, Banking, Telecom, Insurance, Marketing, Retail, Advertising</b>	monitoring & management of customer segments, bankruptcy prediction	product or service recommendation, including complimentary, user intent or information need prediction	demand prediction, response rate prediction, budget planning
<b>Production industry</b>	controlling output quality	–	predict bottlenecks
<b>Education (e-Learning, e-Health), Media, Entertainment</b>	gaming the system, drop out prediction	music, VOD, movie, news, learning object personalized search & recommendations	player-centered game design, learner-centered education

The *information management* applications address personalized learning, they include (web) search, recommender systems, categorization and organization of textual information, customer profiling for marketing, personal mail categorization and spam filtering.

The *analytics and diagnostics* block includes predictive analytics and diagnostics tasks, such as evaluation of creditworthiness, demand prediction, drug resistance prediction.

After identifying three blocks of application areas, we now assign the most likely properties to the respective application areas based on our subjective judgement. Table 4 presents the assignment of the properties.

We acknowledge that contradictory examples within each area are always possible to find, yet we believe that the identified properties are the most common for given areas.

It should be noted also that this summary is aimed to cover the majority of cases that would be traditionally associated with applications of machine learning, data mining, and pattern recognition, in which the term concept drift was originally coined and studied most. More recent examples of big data applications in web information retrieval and recommender systems also fit well to our categorization. However, the wider adoption of the big data perspective in other research areas and application domains may bring new interesting aspects. Thus, e.g. handling concept drift has been recognized as an important problem in process mining research dealing with the different kinds of analysis of (business)

Table 4: Mapping between properties and application areas.

	<b>Monitoring and control</b>	<b>Information management</b>	<b>Analytics and diagnostics</b>
	<i>Task</i>		
<b>task</b>	detection, prediction	prediction ranking	prediction classification
<b>input data</b>	sequential	relational transactional	time series sequential relational
<b>incoming</b>	stream	batches	stream iterations
<b>volume</b>	high	moderate	moderate
<b>multiple scans</b>	no/yes	yes	yes
<b>missing values</b>	random	unlikely	systematic
	<i>Environment</i>		
<b>change source</b>	adversary complex	preferences contextual	population
<b>change type</b>	sudden	gradual incremental	incremental reoccurring
<b>expectations</b>	unpredictable	unpredictable predictable	identifiable unpredictable
	<i>Operational settings</i>		
<b>label speed</b>	fixed lag	on demand	real time
<b>ground labels</b>	objective	subjective	objective

processes by extracting information from event logs recorded by an information system [11, 8]

In the following section we overview application oriented studies on learning from evolving data and through considered examples illustrate peculiarities of handling concept drift under different application settings.

#### 4 An overview of application oriented studies on learning from evolving data

Following the categorization of applications, we distinguish three main groups of application tasks: monitoring and control, information management, and diagnostics. Besides having different goals, the groups also differ in data types. Monitoring and control applications typically use streaming sensory data as inputs, concept drift typically happens fast and suddenly. Information management applications work with time-stamped documents, concept drift happens slower than in the previous case, changes can be sudden or gradual. Diagnostics applications typically use relational data tables, where observations are time-stamped. Concept drift, also known as population drift, typically happens slowly. Changes are typically incremental, or evolving. Sudden shifts are not very typical in these applications.

Table 5: Summary of monitoring and control studies.

Goal	Domain	Application task	References
<b>Monitoring for management</b>	transportation	traffic management	[14, 57]
	remote sensing	place, activity recognition	[77, 52, 51]
	production industry	production quality control	[61, 40]
	telecom. network	telecommunication monitoring	[60]
<b>Automated control</b>	mobile systems	controlling robots, vehicles	[70, 63, 48]
	smart home	intelligent appliances	[64, 2]
	virtual reality	computer games, flight simulators	[13, 34]
<b>Anomaly detection</b>	computer security	intrusion detection	[50]
	telecommunications	intrusion detection, fraud	[54, 37]
	finance	fraud, insider trading	[7, 19]

In this section we briefly characterize each group, overview application studies that fall within each group and touch upon the issue of concept drift, and present three studies in more detail, illustrating how the prediction task is formulated, and how concept drift is handled. We discuss research challenges, and highlight interesting aspects of these application tasks from concept drift handling perspective

We do not claim that this is an exhaustive list of concept drift applications. Our goal is to include examples from a wide range of application tasks.

#### 4.1 Monitoring and Control

The first group of concept drift application tasks aim at real-time monitoring or control of some automated activity, for example, operation of a chemical plant. Input data typically consists of streaming sensory readings, and the target is often related to describing the quality of the activity or process. The goal of such monitoring could be to oversee operation of the system (without interfering, unless something goes wrong), to control the system, or to detect abnormal behaviour (possibly due to adversary actions). Concept drift typically happens fast (in the order of seconds or minutes), and changes are sudden. Table 5 summarizes example studies related to handling concept drift in monitoring and control applications.

**Monitoring for management.** Monitoring for management tasks are often found in production industry and transportation domains. Concept drift is typically observed due to complexity of the process, or human (operators) factors. So many factors are affecting the process, that it is not possible to take all of them in the predictive model. When some of those factors, that have been fixed for a while, suddenly change - a concept drift is observed. For example, production quality in a chemical plant may be different depending on the supplier of raw materials. If we make a model when one supplier is used, such a model may not be as accurate when the supplier changes, and some adaptation may be required.

In transportation, traffic control centers use data driven traffic management systems for predicting traffic conditions [14], such as car density in a particular area, or anticipating traffic accidents. Public transportation travel time prediction [57] is used for scheduling and human resource (drivers) planning purposes. In remote sensing relevant application tasks include place recognition [52], activity recognition [51], interactive road segmentation [77]. In production industry relevant tasks include monitoring the output quality, for example, in chemical production [40], or the process itself, for example, boilers producing heat [61]. Monitoring models in production industry are called *soft sensors* [41]. In *service monitoring* detection of defects or faults in telecommunication network [60] present relevant tasks.

**Automated control.** In automated control applications the problem of concept drift is often referred to as the dynamically changing environment. The objects learn how to interact with the environment and since the environment is too complex to take all the playing factors into a predictive model, therefore predictive models need to be adaptive.

Examples of application domains in automated control include: mobile systems and robotics, smart homes, and virtual reality. Ubiquitous knowledge discovery deals with distributed and mobile systems, operating in a complex, dynamic and unstable environment. The word 'ubiquitous' means distributed at a time. Relevant tasks include navigation systems [70], soccer playing robots [48], vehicle monitoring, household management systems, music mining are examples. Intelligent systems, or smart home systems [64] aim to develop intelligent household appliances [2]. Virtual reality includes application tasks in computer game design [13], where adversary actions of the players (cheating) or improving skills of a player, may be cause concept drift. Virtual reality is also used in flight simulators, where skills and strategies change from user to user [34].

**Anomaly detection.** Anomaly detection is often tackled as one class classification task, where the properties of a normal behavior are well defined, while the properties of abnormal behavior may be changing. Concept drift happens due to changes in behavior, characteristics of legitimate users, or new creative adversary actions.

Anomaly detection is very relevant for computer security domain, in particular, network intrusion detection [50]. In telecommunications fraud prevention [37] or mobile masquerade detection [54] are the relevant tasks. In finance data mining techniques are employed to monitor streams of financial transactions (credit cards, internet banking) to alert for possible frauds or insider trading [68, 3, 31].

**Credit scoring.** In retail banking, credit risk assessment often relies in credit scoring models developed with supervised learning methods used to evaluate a persons credit worthiness. The output of these models is a score that translates a probability of a customer becoming a defaulter, usually in a fixed future period,

so-called scoring or PD models. Nowadays, these models are at the core of the banking business, because they are imperative in credit decision-making, in price settlement, and to determine the cost of capital. Moreover, central banks and international regulation have dramatically evolved to a structure where the use of these models is implicit, to achieve sound standards for credit risk valuation in the banking system.

Developing and implementing a credit scoring model can be time and resources consuming easily ranging from 9 to 18 months, from data extraction until deployment. Hence, it is not rare that banks use an unchanged credit scoring model for several years (a 5 year period is commonly exceeded). Bearing in mind that models are built using a sample file frequently comprising 2 or more years of historical data, in the best case scenario, data used in the models are shifted 3 years away from the point they will be used. An 8-years shift is frequently exceeded. Should conditions remain unchanged, then this would not significantly affect the accuracy of the model, otherwise, its performance can greatly deteriorate over time. The recent financial crisis came to confirm that financial environment greatly fluctuates, in an unexpected manner, posing renewed attention regarding scorecards built upon frames that are by far outdated. By 2007-2008, many financial institutions were using stale scorecards built with historical data of the early-decade. The degradation of stationary credit scoring models is an issue with empirical evidence in the literature[15, 32], however research is still lacking application oriented solutions.

**Example study: online mass flow estimation.** Industrial boilers are used for heating buildings in winter times. Some boilers operate on biofuel, which is a mix tree branches, peat and plants; the mix is not necessarily uniform and the proportions may vary. The authors of the first example study [61] consider the problem of online mass flow estimation in boiler operation. During burning phase the mass of fuel inside the boiler container decreases, and as new fuel is added to the container, while at the same time the burning process continues, the fuel feeding phase starts that is reflected by a rapid mass increase.

Input data comes from physical sensors with a negligible lag. The task is to estimate the current mass flow (similarly to fuel consumption indicators in passenger cars), and detect the points of phase switch in real time.

There are three main sources of drifts in the signal (an example is depicted in Figure 4). First, fuel feeding is manual and non-standardized process, which is not necessarily smooth and may have short interruptions. Second, rotation of the feeding screw adds noise to the measured signal. Finally, there is a low amplitude rather periodic noise, which is caused by the mechanical rotation of the system parts, the magnitude of this noise depends on operational setting.

The main focus is on constructing a learning system that can deal with two types of change points: an abrupt change to feeding and slower but still abrupt switch to burning, and asymmetric outliers, which in online settings can be easily mixed with the changes to feeding. These change points need to be identified in real time, and they should not be mixed with noise. When these regime switch

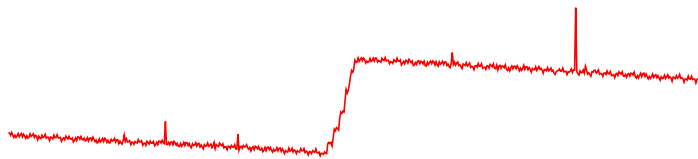


Fig. 4: An example of boiler data.

points are known, a new predictive model can be incrementally started after each feed to reflect the most recent fuel characteristics.

The optimization criteria for change detection is to minimize the detection delay (from the actual change point to detection), and minimize the number of false alarms, when an outlier is singled as a change. All true change points have to be detected, no misses are allowed. In addition, the final performance indicator is the mean square error (MSE) of the mass flow estimation. It is critical for algorithm design to understand, how different types of errors in detection affect the overall accuracy of classification. Such sensitivity analysis can be performed by varying the detection thresholds.

Evaluation of the performance of the algorithms is challenging, since there is no ground truth available. The authors construct an approximation to the ground truth, and use it for the evaluation purposes in online settings (only in the experiments, but not in real operational setting). Absence of ground truth is a common problem in monitoring applications, since, if it was easily available, there would be no need for the predictive model, which is being designed.

## 4.2 Information management

These tasks aim at organizing, and personalizing information. Typically, data is comes as time stamped entities, for instance, web documents, and the goal is to characterize each entity. Information management application tasks can be further split into personal assistance, marketing, and management tasks. Concept drift happens not so fast (in the order of days or weeks), changes could be sudden or gradual. Table 6 summarizes example studies related to handling concept drift for information management.

**Personal assistance.** Personal assistance applications aim at user modeling. The goal is to personalize information flow, the process is often referred to as information filtering. A rich technical presentation on user modeling can be found in [28]. One of the primary applications of user modeling is representation of queries, news, blog entries with respect to *current* user interests. Changing user interests over time is the main cause of concept drift.

Large part of personal assistance applications are related to handling *textual data*, example tasks include news story classification [74, 4], or document categorization [49, 59]. In web search, detecting changes in user satisfaction has been

Table 6: Summary of information management studies.

Goal	Domain	Application task	References
<b>Personal assistance</b>	text processing	news categorization,	[74, 4, 49, 59]
		document classification	
	web mining	spam filtering	[17, 21]
		web personalization	[67, 16, 9]
		library, media search	[35, 22]
<b>Marketing</b>	customer segmentation	customer segmentation,	[14, 5, 47, 66]
		profiling	
<b>Management</b>	recommender systems	movie recommendations	[45, 18]
	project management	software project mgmt.	[20]
	archiving	article, mail organization	[76, 43]

recognized to be important [42]. Personal assistance tasks relate to other types of data, such as networked multimedia, music, video, as well as digital libraries [35]. Large body of applications relate to web personalization and dynamics [67, 16, 9], where interim system data (logs) is mined.

**Marketing.** Customer profiling applications use aggregated data from many users. The goal is to segment customers based on their interests and needs. Concept drift happens due to changing individual interests and behavior over time.

Relevant tasks include direct marketing, based on product preferences, for example cars [14], or service usage, for example telecommunications [5], identifying and analyzing shopping baskets [66], social network analysis for customer segmentation [47], recommender systems [45].

**Management.** A number of studies aim at adaptive organization or categorization of web documents, e-mails, news articles [76, 43]. Concept drift happens due to evolving nature of the content. In business software project management, careful planning may be inaccurate if concept drift is not taken into account [20].

**Example study: movie recommendation.** Interest of data mining community in recommender systems domain has been boosted by Netflix competition<sup>7</sup>. One of the lessons learnt from it was that taking temporal dynamics is important for building accurate models. Handling concept drift has another set of peculiarities here. Both items and users are changing over time. Item-side effects include first of all changing product perception and popularity. The popularity of some movies is expected to follow seasonal patterns. User-side effects include changing tastes and preferences of customers, some of which may be short-term or contextual and therefore likely reoccurring (mood, activity, company, etc), changing

<sup>7</sup> [www.netflixprize.com](http://www.netflixprize.com)



Table 7: Summary of diagnostics studies.

Goal	Domain	Application task	References
<b>Forecasting</b>	banking	bankruptcy prediction	[69, 38]
	economics	forecasting	[29, 44, 33]
<b>Medicine</b>	drug research	antibiotic resistance	[72, 39, 53]
		drug discovery	[23]
	clinical research	disease monitoring	[46, 24, 6]
<b>Security</b>	biometrics	authentication	[75, 62]

perception of rating scale, possible change of rater within household and alike problems.

As suggested in [45] popular windowing and instance weighing approaches for handling concept drift are not the best choice simply because in collaborative filtering the relations between ratings is of the main importance for predictive modeling.

In this application labels are soft, data comes in batches, and the rating matrix is high-dimensional and extremely sparse containing only about 1% of non-zero elements (that makes the application of most machine learning predictors inapplicable and boost the development of advanced collaborative filtering approaches).

### 4.3 Analytics and diagnostics

Analytics and diagnostics tasks aim at characterizing health, well-being, or a state of humans, economies, or entities. Data typically comes as time stamped relational data. Concept drift often happens due to population drift, and changes are typically slow (in the order of months or years) and incremental.

Analytics and diagnostics tasks can be further split into forecasting, medicine, or security applications. Table 7 summarizes example studies related to handling concept drift in diagnostics.

Changes happen due to changing environment, such as economic situation, which includes a large number of influencing factors.

**Forecasting.** Forecasting applications typically relate to analytics tasks in economics and finance, such as macroeconomic forecasting, demand prediction, travel time predictions, event prediction (e.g. crime maps, epidemic outbreaks). Changes over time often happen due to population drift, which typically happens much slower, than, for instance, changes in personal preferences in information management applications, or adversary actions in monitoring applications.

In finance relevant tasks include bankruptcy prediction or individual credit scoring [69, 38], in economics concept drift appears in making macroeconomic forecasts [29], predicting phases of a business cycle [44], or stock price prediction [33].

**Security.** In biometric authentication [75, 62] concept drift can be caused by changing physiological factors, e.g. growing a beard.

**Medicine.** Medicine applications, such as antibiotic resistance prediction, or predicting epidemic outbreaks or nosocomial infections, may be subject to concept drift due to adaptive nature of microorganisms [72, 39, 53]. Clinical studies and systems need adaptivity mechanisms to changes caused by human demographics [46, 24].

**Example study: predicting antibiotic resistance.** Antibiotic resistance is an important problem and it is an especially difficult problem with nosocomial infections in hospitals because pathogens attack critically ill patients who are more vulnerable to infections than the general population and therefore require more antibiotics.

Prediction model is based on information about patients, hospitalization, pathogens and antibiotic themselves. The data arrives in batches, the labels become available with a variable lag depending on the size of the hospital and intensiveness of the patients flow. The size of the data is relatively small both in number of instances and the number of features to be considered.

The peculiarity of concept drift is that it may happen for various reasons particularly because pathogens may develop resistance and share this information with peers in different ways. Consequently, the type and severity of changes may depend on the location in the instance space. Furthermore, the drift is expected to be local and reflect e.g. a pathway in the hospital where the resistance was taking place and spread around. This calls for the direct or indirect identification of the regions or subgroups in which concept drift is occurring. Handling concept drift with dynamic integration of classifiers that takes this peculiarity into account was shown to be effective [72].

## 5 Discussion and Conclusions

The main lesson in this study is related to the evolving characteristic of data and the implications in data analysis. Nowadays, digital data collection is easy and cheap. Data analytics in applications where data is collected over time, must take into account the evolving nature of data.

The problem of concept drift has been recognized in different application domains. Interest in different research communities has been reinforced by several recent competitions including e.g. controlling driverless cars at the DARPA challenge, credit risk assessment competition at PAKDD'09), and Netflix movie recommendation.

However, concept drift research field is still in an early stage. The research problems, although motivated by a belief that handling concept drift is highly important for practical data mining applications, have been formulated and addressed often in artificial and somewhat isolated settings. Approaches for handling concept drift are rather diverse and have been developed from two sides –

theory-oriented and applications-oriented. Recent studies however do highlight the peculiarities of particular applications and give intuition and/or empirical evidence why traditional general-purpose concept drift handling techniques are not expected to perform well and suggest tailored or more focused techniques suitable for a particular application type.

In this work we categorized the applications, where handling concept drift is known or expected to be an important component of any learning system. We identified three major types of applications, identified key properties of the corresponding settings, and provided a discussion emphasizing the most important application oriented aspects. Summarizing those we can speculate that the concept drift research area is likely to refocus further from studying general methods to detect and handle concept drift to designing more specific, application oriented approaches that address various issues like delayed labeling, label availability, cost-benefit trade-off of the model update and other issues peculiar to a particular type of applications.

Most of the work on concept drift assumes that the changes happen in hidden context that is not observable to the adaptive learning system. Hence, concept drift is considered to be unpredictable and its detection and handling is mostly reactive. However, there are various application settings in which concept drift is expected to reappear along the time line and across different objects in the modeled domain. Seasonal effects with vague periodicity for a certain subgroup of object would be common e.g. in food demand prediction [78]. Availability of external contextual information or extraction of hidden contexts from the predictive features may help to better handle recurrent concept drift, e.g. with use of a meta-learning approach [25]. Temporal relationships mining can be used to identify related drifts, e.g. in the distributed or peer-to-peer settings in which concept drift in one peer may precede another drift in related peer(s) [1]. Thus, we can expect that for many applications more accurate, more proactive and more transparent change detection mechanisms may become possible.

Moving from adaptive algorithms towards adaptive systems that would automate full knowledge discovery process and scaling these solutions to meet the computational challenges of big data applications is another important step for bringing research closer to practice. Developing open-source tools like SAMOA [56] certainly facilitates this.

Domain experts play an important role in acceptance of big data solutions. They often want to go away from non interpretable black-box models and to develop trust in underlying techniques, e.g. to be certain that a control system is really going to react to changes when they happen and to understand how these changes are detected and what adaptation would happen. Therefore we anticipate that there will be also a change in the focus from change detection to change *description*, from *when a change happen* to *how and why it happened* as such research would be helpful in improving utility, usability and trust in adaptive learning systems being developed for many of the big data applications.

**Acknowledgements.** This work was partially supported by European Commission through the project MAESTRA (Grant number ICT-2013-612944).

## References

1. H. H. Ang, V. Gopalkrishnan, I. Zliobaite, M. Pechenizkiy, and S.C.H. Hoi. Predictive handling of asynchronous concept drifts in distributed environments. *IEEE Trans. on Knowl. and Data Eng.*, 25:2343–2355, 2013.
2. D. Anguita. Smart adaptive systems: State of the art and future directions of research. In *Proc. of the 1st European Symp. on Intelligent Technologies, Hybrid Systems and Smart Adaptive Systems*, EUNITE, 2001.
3. R. A. Becker, C. Volinsky, and A. R. Wilks. Fraud detection in telecommunications: History and lessons learned. *Technometrics*, 52(1):20–33, 2010.
4. D. Billsus and M. Pazzani. A hybrid user model for news story classification. In *Proc. of the 7th int. conf. on User modeling*, UM, pages 99–108, 1999.
5. M. Black and R. Hickey. Classification of customer call data in the presence of concept drift and noise. In *Proc. of the 1st Int. Conf. on Computing in an Imperfect World*, pages 74–87, 2002.
6. M. Black and R. Hickey. Detecting and adapting to concept drift in bioinformatics. In *Proc. of Knowledge Exploration in Life Science Informatics, International Symposium*, pages 161–168, 2004.
7. R. Bolton and D. Hand. Statistical fraud detection: A review. *Statistical Science*, 17(3):235–255, 2002.
8. R. P. Jagadeesh Chandra Bose, Wil M. P. van der Aalst, Indre Zliobaite, and Mykola Pechenizkiy. Dealing with concept drift in process mining. *IEEE Trans. on Neur. Net. and Lear. Syst.*, accepted:–, 2013.
9. P. De Bra, A. Aerts, B. Berden, B. de Lange, B. Rousseau, T. Santic, D. Smits, and N. Stash. AHA! the adaptive hypermedia architecture. In *Proc. of the 14th ACM conf. on Hypertext and hypermedia*, pages 81–84, 2003.
10. M. Budka, M. Eastwood, B. Gabrys, P. Kadlec, M. Martin-Salvador, S. Schwan, A. Tsakonas, and I. Zliobaite. From sensor readings to predictions: On the process of developing practical soft sensors. In *Proc. of the 13th Int. Symp. on Intelligent Data Analysis*, pages 49–60, 2014.
11. J. Carmona and R. Gavaldà. Online techniques for dealing with concept drift in process mining. In *Proc. of the 11th int. symposium on Intelligent Data Analysis*, pages 90–102, 2012.
12. P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth. CRISP-DM 1.0 step-by-step data mining guide. Technical report, The CRISP-DM consortium, 2000.
13. D. Charles, A. Kerr, M. McNeill, M. McAlister, M. Black, J. Kucklich, A. Moore, and K. Stringer. Player-centred game design: Player modelling and adaptive digital games. In *Proc. of the Digital Games Research Conference*, pages 285–298, 2005.
14. F. Crespo and R. Weber. A methodology for dynamic data mining based on fuzzy clustering. *Fuzzy Sets and Systems*, 150:267–284, 2005.
15. J. Crook, R. Hamilton, and T. L. C. The degradation of the scorecard over the business cycle. *IMA Journal of Management Mathematics*, 4:111–123, 1992.
16. A. da Silva, Y. Lechevallier, F. Rossi, and F. de Carvalho. Construction and analysis of evolving data summaries: An application on web usage data. In *Proc. of the 7th Int. Conf. on Intelligent Systems Design and Applications*, pages 377–380, 2007.

17. S. Delany, P. Cunningham, and A. Tsybal. A comparison of ensemble and case-base maintenance techniques for handling concept drift in spam filtering. In *Proc. of Florida Artificial Intelligence Research Society Conference*, pages 340–345, 2006.
18. Y. Ding and X. Li. Time weight collaborative filtering. In *Proc. of the 14th ACM int. conf. on Information and knowledge management*, pages 485–492, 2005.
19. S. Donoho. Early detection of insider trading in option markets. In *Proc. of the 10th ACM SIGKDD int. conf. on Knowledge discovery and data mining*, pages 420–429, 2004.
20. J. Ekanayake, J. Tappolet, H. C. Gall, and A. Bernstein. Tracking concept drift of software projects using defect prediction quality. In *Proc. of the 6th IEEE Int. Working Conf. on Mining Software Repositories*, pages 51–60, 2009.
21. F. Fdez-Riverola, E. Iglesias, F. Diaz, J. Mendez, and J. Corchado. Applying lazy learning algorithms to tackle concept drift in spam filtering. *Expert Syst. Appl.*, 33(1):36–48, 2007.
22. O. Flasch, A. Kaspari, K. Morik, and M. Wurst. Aspect-based tagging for collaborative media organization. In *Proc. of Workshop on Web Mining, From Web to Social Web: Discovering and Deploying User and Content Profiles*, pages 122–141, 2007.
23. G. Forman. Incremental machine learning to reduce biochemistry lab costs in the search for drug discovery. In *Proc. of the 2nd workshop on Data Mining in Bioinformatics*, pages 33–36, 2002.
24. P. Gago, A. Silva, and M. Santos. Adaptive decision support for intensive care. In *Proc. of 13th Portuguese Conference on Artificial Intelligence*, pages 415–425, 2007.
25. J. Gama and P. Kosina. Learning about the learning process. In *Proc. of the 10th int. conf. on Advances in intelligent data analysis, IDA*, pages 162–172, Germany, 2011. Springer.
26. J. Gama, P. Medas, G. Castillo, and P. Rodrigues. Learning with drift detection. In *Proc. of the 17th Brazilian Symposium on Artificial Intelligence*, pages 286–295, 2004.
27. J. Gama, I. Zliobaite, A. Bifet, M. Pechenizkiy, and A. Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):44:1–44:37, 2014.
28. S. Gauch, M. Speretta, A. Chandramouli, and A. Micarelli. User profiles for personalized information access. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web*, pages 54–89. Springer-Verlag, 2007.
29. R. Giacomini and B. Rossi. Detecting and predicting forecast breakdowns. Working Paper 638, ECB, 2006.
30. D. Hand. Classifier technology and the illusion of progress. *Statistical Science*, 21(1):1–14, 2006.
31. D. J. Hand. Fraud detection in telecommunications and banking: Discussion of Becker, Volinsky, and Wilks (2010) and Sudjianto et al. (2010). *Technometrics*, 52(1):34–38, 2010.
32. D. J. Hand and N. M. Adams. Selection bias in credit scorecard evaluation. *JORS*, 65(3):408–415, 2014.
33. M. Harries and K. Horn. Detecting concept drift in financial time series prediction using symbolic machine learning. In *In Proc. of the 8th Australian joint conf. on artificial intelligence*, pages 91–98, 1995.
34. M. Harries, C. Sammut, and K. Horn. Extracting hidden context. *Mach. Learn.*, 32(2):101–126, 1998.

35. M. Hasan and E. Nantajeewarawat. Towards intelligent and adaptive digital library services. In *Proc. of the 11th Int. Conf. on Asian Digital Libraries*, pages 104–113, 2008.
36. S. Haykin and Liang Li. Nonlinear adaptive prediction of nonstationary signals. *IEEE Trans. on Signal Processing*, 43(2):526–535, 1995.
37. C. Hilar. Designing an expert system for fraud detection in private telecommunications networks. *Expert Syst. Appl.*, 36(9):11559–11569, 2009.
38. R. Horta, B. de Lima, and C. Borges. Data pre-processing of bankruptcy prediction models using data mining techniques, 2009.
39. C. Jermaine. Data mining for multiple antibiotic resistance. online, 2008.
40. P. Kadlec and B. Gabrys. Local learning-based adaptive soft sensor for catalyst activation prediction. *AIChE Journal*, 57(5):1288–1301, 2011.
41. P. Kadlec, R. Grbic, and B. Gabrys. Review of adaptation mechanisms for data-driven soft sensors. *Computers & Chemical Engineering*, 35:1–24, 2011.
42. J. Kiseleva, E. Crestan, R. Brigo, and R. Dittel. Modelling and detecting changes in user satisfaction. In *Proc. of the 23rd ACM Int. Conf. on Conference on Information and Knowledge Management*, pages 1449–1458, 2014.
43. J. Kleinberg. Bursty and hierarchical structure in streams. In *Proceedings of the 8th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 91–101. ACM, 2002.
44. R. Klinkenberg. Meta-learning, model selection and example selection in machine learning domains with concept drift. In *Proc. of annual workshop of the Special Interest Group on Machine Learning, Knowledge Discovery, and Data Mining*, pages 64–171, 2005.
45. Y. Koren. Collaborative filtering with temporal dynamics. *Commun. ACM*, 53(4):89–97, 2010.
46. M. Kukar. Drifting concepts as hidden factors in clinical studies. In *Proc. of the 9th Conf. on Artificial Intelligence in Medicine in Europe*, pages 355–364, 2003.
47. N. Lathia, S. Hailes, and L. Capra. kNN CF: a temporal social network. In *Proc. of the ACM Conf. on Recommender Systems*, pages 227–234, 2008.
48. A. Lattner, A. Miene, U. Visser, and O. Herzog. Sequential pattern mining for situation and behavior prediction in simulated robotic soccer. In *Proc. of Robot Soccer World Cup IX*, pages 118–129, 2006.
49. G. Lebanon and Y. Zhao. Local likelihood modeling of temporal text streams. In *Proc. of the 25th int. conf. on Machine learning*, pages 552–559, 2008.
50. W. Lee, S. J. Stolfo, and K. W. Mok. Adaptive intrusion detection: A data mining approach. *Artif. Intell. Rev.*, 14(6):533–567, 2000.
51. L. Liao, D. Patterson, D. Fox, and H. Kautz. Learning and inferring transportation routines. *Artif. Intell.*, 171(5-6):311–331, 2007.
52. J. Luo, A. Pronobis, B. Caputo, and P. Jensfelt. Incremental learning for place recognition in dynamic environments. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 721–728, 2007.
53. M. T. Martin, T. B. Knudsen, R. S. Judson, R. J. Kavlock, and D. J. Dix. Economic benefits of using adaptive predictive models of reproductive toxicity in the context of a tiered testing program. *Syst. Biol. Reprod. Med.*, 58:3–9, 2012.
54. O. Mazhelis and S. Puuronen. Comparing classifier combining techniques for mobile-masquerader detection. In *Proc. of the The 2nd Int. Conf. on Availability, Reliability and Security*, pages 465–472, 2007.
55. L. L. Minku, A. P. White, and X. Yao. The impact of diversity on online ensemble learning in the presence of concept drift. *IEEE Trans. Knowl. Data Eng.*, 22(5):730–742, 2010.

56. G. De Francisci Morales and A. Bifet. SAMOA: Scalable advanced massive online analysis. *Journal of Machine Learning Research*, 16:149–153, 2015.
57. J. Moreira. *Travel time prediction for the planning of mass transit companies: a machine learning approach*. PhD thesis, University of Porto, 2008.
58. J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera. A unifying view on dataset shift in classification. *Pattern Recogn.*, 45(1):521–530, 2012.
59. F. Mourao, L. Rocha, R. Araujo, T. Couto, M. Goncalves, and W. Meira. Understanding temporal aspects in document classification. In *Proc. of the int. conf. on Web search and web data mining*, pages 159–170, 2008.
60. A. Pawling, N. Chawla, and G. Madey. Anomaly detection in a mobile communication network. *Comput. Math. Organ. Theory*, 13(4):407–422, 2007.
61. M. Pechenizkiy, J. Bakker, I. Zliobaite, A. Ivannikov, and T. Karkkainen. Online mass flow prediction in CFB boilers with explicit detection of sudden concept drift. *SIGKDD Explorations*, 11(2):109–116, 2009.
62. N. Poh, R. Wong, J. Kittler, and F. Roli. Challenges and research directions for adaptive biometric recognition systems. In *Proc. of the 3rd Int. Conf. on Advances in Biometrics*, pages 753–764, 2009.
63. M. Procopio, J. Mulligan, and G. Grudic. Learning terrain segmentation with classifier ensembles for autonomous robot navigation in unstructured environments. *J. Field Robot.*, 26(2):145–175, 2009.
64. P. Rashidi and D. Cook. Keeping the resident in the loop: Adapting the smart home to the user. *IEEE Trans. on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 39(5):949–959, 2009.
65. Thomas P. Reinartz. *Focusing Solutions for Data Mining: Analytical Studies and Experimental Results in Real-World Domains*, volume 1623 of *Lecture Notes in Computer Science*. Springer, 1999.
66. A. Rozsygal and M. Kubat. Association mining in time-varying domains. *Intell. Data Anal.*, 9(3):273–288, 2005.
67. J. Scanlan, J. Hartnett, and R. Williams. DynamicWEB: Adapting to concept drift and object drift in cobweb. In *Proc. of the 21st Australasian Joint Conf. on Artificial Intelligence*, pages 454–460, 2008.
68. A. Sudjianto, S. Nair, M. Yuan, A. Zhang, D. Kern, and F. Cela-Diaz. Statistical methods for fighting financial crimes. *Technometrics*, 52(1):5–19, 2010.
69. T. Sung, N. Chang, and G. Lee. Dynamics of modeling in data mining: interpretive approach to bankruptcy prediction. *J. Manage. Inf. Syst.*, 16(1):63–85, 1999.
70. S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Winning the darpa grand challenge. *Journal of Field Robotics*, 23(9):661–692, 2006.
71. A. Tsymbal. The problem of concept drift: Definitions and related work. Technical report, Department of Computer Science, Trinity College Dublin, Ireland, 2004.
72. A. Tsymbal, M. Pechenizkiy, P. Cunningham, and S. Puuronen. Dynamic integration of classifiers for handling concept drift. *Information Fusion*, 9(1):56–68, 2008.
73. G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Mach. Learn.*, 23(1):69–101, 1996.
74. D. Widyantoro and J. Yen. Relevant data expansion for learning concept drift from sparsely labeled data. *IEEE Trans. on Knowl. and Data Eng.*, 17(3):401–412, 2005.

75. R. Yampolskiy and V. Govindaraju. Direct and indirect human computer interaction based biometrics. *Journal of computers*, 2(10):76–88, 2007.
76. Y. Yang, X. Wu, and X. Zhu. Mining in anticipation for concept change: Proactive-reactive prediction in data streams. *Data Min. Knowl. Discov.*, 13(3):261–289, 2006.
77. J. Zhou, L. Cheng, and W. Bischof. Prediction and change detection in sequential data for interactive applications. In *Proc. of the 23rd AAAI Conf. on Artificial Intelligence*, pages 805–810, 2008.
78. I. Zliobaite, J. Bakker, and M. Pechenizkiy. Beating the baseline prediction in food sales: How intelligent an intelligent predictor is? *Expert Syst. Appl.*, 31(1):806–815, 2012.