

## Comparative study of classification algorithms for immunosignaturing data

Kukreja *et al.*

---

RESEARCH ARTICLE

Open Access

# Comparative study of classification algorithms for immunosignaturing data

Muskan Kukreja, Stephen Albert Johnston and Phillip Stafford\*

## Abstract

**Background:** High-throughput technologies such as DNA, RNA, protein, antibody and peptide microarrays are often used to examine differences across drug treatments, diseases, transgenic animals, and others. Typically one trains a classification system by gathering large amounts of probe-level data, selecting informative features, and classifies test samples using a small number of features. As new microarrays are invented, classification systems that worked well for other array types may not be ideal. Expression microarrays, arguably one of the most prevalent array types, have been used for years to help develop classification algorithms. Many biological assumptions are built into classifiers that were designed for these types of data. One of the more problematic is the assumption of independence, both at the probe level and again at the biological level. Probes for RNA transcripts are designed to bind single transcripts. At the biological level, many genes have dependencies across transcriptional pathways where co-regulation of transcriptional units may make many genes appear as being completely dependent. Thus, algorithms that perform well for gene expression data may not be suitable when other technologies with different binding characteristics exist. The immunosignaturing microarray is based on complex mixtures of antibodies binding to arrays of random sequence peptides. It relies on many-to-many binding of antibodies to the random sequence peptides. Each peptide can bind multiple antibodies and each antibody can bind multiple peptides. This technology has been shown to be highly reproducible and appears promising for diagnosing a variety of disease states. However, it is not clear what is the optimal classification algorithm for analyzing this new type of data.

**Results:** We characterized several classification algorithms to analyze immunosignaturing data. We selected several datasets that range from easy to difficult to classify, from simple monoclonal binding to complex binding patterns in asthma patients. We then classified the biological samples using 17 different classification algorithms. Using a wide variety of assessment criteria, we found 'Naïve Bayes' far more useful than other widely used methods due to its simplicity, robustness, speed and accuracy.

**Conclusions:** 'Naïve Bayes' algorithm appears to accommodate the complex patterns hidden within multilayered immunosignaturing microarray data due to its fundamental mathematical properties.

**Keywords:** Immunosignature, Random peptide microarray, Data mining, Classification algorithms, Naive Bayes

## Background

Serological diagnostics have received increasing scrutiny recently [1,2] due to their potential to measure antibodies rather than low-abundance biomarker molecules. Antibodies avoid the biomarker dilution problem and are recruited rapidly following infection, chronic, or autoimmune episodes, or exposure to cancer cells. Serological diagnostics using antibodies have the potential to reduce medical costs and may be one of the few methods

that allow for true presymptomatic detection of disease. For this reason, our group has pursued immunosignaturing for its ability to detect the diseases early and with a low false positive rate. The platform consists of a peptide microarray with either 10,000 or 330,000 peptides per assay. This microarray is useful with standard mathematical analysis, but for a variety of reasons, certain methods of classification enable the best accuracy [3,4]. Classification methods differ in their ability to handle high or low numbers of features, the feature selection method, and the features' combined contribution to a linear, polynomial, or complex discrimination threshold. Expression microarrays

\* Correspondence: phillip.stafford@asu.edu  
Center for Innovations in Medicine, Biodesign Institute, Arizona State University, Tempe, AZ 85281, USA

are quite ubiquitous and relevant to many biological studies, and have been used often when studying classification methods. However, immunosignaturing microarrays may require that we change our underlying assumptions as we determine the suitability of a particular classifier.

In order to establish the question of classification suitability, we examine a basic classification algorithm, Linear Discriminant Analysis (LDA). LDA is widely used in analyzing biomedical data in order to classify two or more disease classes [5-8]. One of the most commonly used high-throughput analytical methods is the gene expression microarray. Probes on an expression microarray are designed to bind to a single transcript, splice variant or methy variant of that transcript. These one-on-one interactions provide relative transcript numbers and cumulatively help to define high-level biological pathways. LDA uses these data to define biologically relevant classes based on the contribution of differentially expressed genes. This method often uses statistically identified features (gene transcripts) that are different from one condition to another. LDA can leverage coordinated gene expression to make predictions based on a fundamental biological process. The advantage of this method is that relatively few features are required to make sweeping predictions. When features change sporadically or asynchronously, the discriminator predictions are adversely affected. This causes low sensitivity in exchange for occasionally higher discrimination. Tree-based methods use far more features to obtain a less biased but less sensitive view of the data. These methods can partition effects even if the effect sizes vary considerably. This approach can be more useful than frequentist approaches where it is important to maintain partitions in discreet groups.

Immunosignaturing has its foundations in both phage display and peptide microarrays. Most phage display methods that use random-sequence libraries also use fairly short peptides, on the order of 8–11 amino acids [9]. Epitope microarrays use peptides in the same size range, but typically far fewer total peptides, on the order of hundreds to thousands [10]. Each of these methods assumes that a single antibody binds to a single peptide, which is either detected by selection (phage display) or by fluorescent secondary antibody (epitope microarray). Immunosignaturing uses long 20-mer random-sequence peptides that have potentially 7 or more possible linear epitopes per peptide. Although immunosignaturing must make do with only 10,000 to ~300,000 peptides, the information content derived from partial binding makes these data useful in ways quite different from phage display [11-15].

The complexity in analysis arises from the many-to-many relationship between peptide and antibody (Figure 1). This relationship imposes a particular challenge for classification because a simple one-to-one relationship between

probe and target, idiomatic for gene expression microarrays, allows a coherent contribution of many genes that behave coordinately based on biological stimuli. That idiom is broken for immunosignaturing microarrays, where each peptide may bind a number of different antibodies and every antibody might bind a number of peptides. Unless disease-specific antibodies find similar groups of peptides across individuals, very little useful information is available to the classifier. The aim of this work is to assess the performance of various classification algorithms on immunosignaturing data.

We have considered 17 diverse data mining classification methods. For feature selection, we used a simple *t*-test when we examined two classes, and a fixed-effects 1-way ANOVA for multiple classes with no post-hoc stratification. We have assessed these algorithms' ability to handle increasing numbers of features by providing four different sets of peptides with increasing p-value cutoff. The four levels include from 10 (minimum) to >1000 (maximum) peptides. Each algorithm is thus tested under conditions that highlight either synergistic or antagonistic effects as the feature numbers increase.

## Methods

### Technology

A peptide microarray described previously [11-15] was used to provide data for analysis. Two different sets of 10,000 random peptide sequences are tested. The two peptide sets are non-overlapping and are known as CIM10Kv1 and CIM10Kv2. Peptides are printed as in [1].

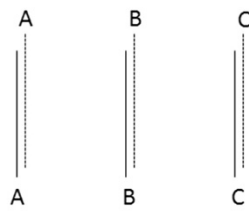
### Sample processing

Samples consist of sera, plasma or saliva – each produces a suitable distribution of signals upon detection with an anti-human secondary IgG-specific antibody. Samples are added to the microarray at 1:500 dilutions in sample buffer (1xPBS, 0.5% Tween20, 0.5% Bovine Serum Albumin (Sigma, St. Louis, MO)), IgG antibodies are detected through a biotinylated secondary anti human IgG antibody (Novus anti-human IgG (H+L), Littleton, CO), which binds the primary. Fluorescently labeled streptavidin is used to label the secondary antibodies and scanned with an Agilent 'C' laser scanner in single-color mode. 16-bit images are processed using GenePix Pro 8, which provides the tabular information for each peptide in a continuous value ranging from 0–65,000. Four unique data sets have been used in this analysis, 2 run on the CIM10Kv1 and 2 on the CIM10Kv2. Each individual sample was run in duplicate; replicates with >0.8 Pearson correlation coefficient were considered for analysis.

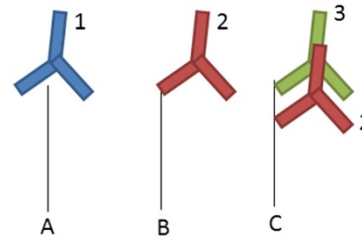
### Datasets

Center for Innovations in Medicine, Biodesign Institute, Arizona State University has an existing IRB

Binding in Gene expression



Binding in peptide expression



**Figure 1** One-to-one correspondence found in gene expression microarrays is not observed for the immunosignaturing arrays. We propose that a single peptide may bind numerous antibodies, and have shown that a single antibody can bind hundreds of different peptides.

0912004625, which allows analysis of blinded samples from collaborators.

- Type 1 diabetes data set: This dataset contains 80 sera samples (41 controls and 39 type 1 diabetes children ages 6 to 13). These samples were tested on the CIM10kV1 microarrays.
- Alzheimer's disease data set: This dataset contains 23 samples (12 controls and 11 Alzheimer's disease subjects). These were tested on the CIM10kV2 microarrays.
- Antibodies dataset: This dataset contains 50 samples and has 5 groups monoclonal antibodies, arbitrarily arranged. All monoclonals were raised in mouse, and use the same secondary detection antibody. Samples were run on the CIM10kV1 microarrays.
- Asthma dataset: This dataset consists of 47 unique samples containing serum from patients with 4 distinct classes, corresponding to the household environment. Condition A consists of 12 control subjects who had no environmental stimuli. Condition B consists of 12 subjects who had stimuli but no asthma-related symptoms. Condition C consists of 11 subjects who had no stimuli but with clinical asthma. Condition D consists of 12 subjects who have both stimuli and clinical asthma. Samples were tested on the CIM 10 kV2 microarrays. Asthma datasets were been analyzed by considering all four conditions using ANOVA in order to study the combined effect of stimuli and asthma on subjects and then by considering pair wise comparison of condition A vs. B, A vs. C, and B vs. D.

#### Data preprocessing, normalization and feature selection

The 16-bit tiff images from the scanned microarrays were imported into GenePix Pro 6.0 (Molecular Devices, Santa Clara, CA). Raw tabular data were imported into Agilent's GeneSpring 7.3.1 (Agilent, Santa Clara, CA). Data were median normalized per array and  $\log_{10}$  transformed. For feature selection we used Welch-corrected *T*-test with multiple tested (FWER = 5%). For multiple

groups (Antibody and Asthma datasets) we used 1-way fixed-effects ANOVA.

#### Data mining classification algorithms

Four distinct peptide features are chosen for the comparison study. For each analysis, peptides are selected by *t*-test or ANOVA across biological classes, with 4 different p-value cutoffs. Cutoffs were selected to obtain roughly equivalent sized feature sets to assess the ability of each algorithm to process sparse to rich feature sets. Once the significant features were collected, data was imported into WEKA [16] for classification. The algorithms themselves spanned a wide variety of classifiers including Bayesian, regression based methods, meta-analysis, clustering, and tree based approaches.

We obtained accuracy from each analysis type using leave-one-out cross-validation. We obtained a list of *t*-test or ANOVA-selected peptides at each stringency level. The highest stringency uses peptides with p-values in the range of  $10^{-5}$  to  $10^{-10}$  and contains the least 'noise'. The less-stringent second set uses p-values approximately 10-fold higher than the most stringent. The third contains the top 200 peptides and the fourth contains ~1000 peptides at  $p < 0.05$ . Although different numbers of peptides are used for each dataset, each peptide set yields the same general ability to distinguish the cognate classes. The WEKA default setting of parameters were used for every algorithm to avoid bias and over fitting. These default parameters are taken from the cited papers listed below for each algorithm. Brief details of default parameters and algorithms are listed

- Naïve Bayes: Probabilistic classifier based on Bayes theorem. Numeric estimator precision values are chosen based on analysis of the training data. In the present study, normal distribution was used for numeric attributes rather than kernel estimator [17].
- Bayes net: Probabilistic graphical model that represents random variables and conditional dependencies in the form of a directed acyclic

- graph. A Simple Estimator algorithm has been used for finding conditional probability tables for Bayes net. A K2 search algorithm was used to search network structure [18,19].
- III. Logistic Regression (Logistic R): A generalized linear model that uses logistic curve modeling to fit the probabilistic occurrence of an event[20]. The Quasi-Newton method is used to search for optimization.  $1 \times 10^8$  has been used for ridge values in the log likelihood calculation [21].
  - IV. Simple Logistic: Classifier for building linear logistic regression models. For fitting the logistic model 'LogitBoost', simple regression functions are used. Automatic attribute selection is obtained by cross validation of the optimal number of 'LogitBoost' iterations [22]. Heuristic stop parameter is set at 50. The number of maximum iterations for LogitBoost has been set to 500.
  - V. Support Vector Machines (SVM): A non-probabilistic binary linear classifier that constructs one or more hyper planes to be can be used for classification. For training support vector classes, John Platt's sequential minimal optimization algorithm was used which replaces all missing values [23]. Here multiclass problems are used using pair-wise classification. The complexity parameter is set to 1. Epsilon for round off error is set to  $1 \times 10^{-12}$ . PolyKernel is the set to be kernel. The tolerance parameter is set to 0.001 [24,25].
  - VI. Multilayer Perceptron (MLP): A supervised learning technique with a feed forward artificial neural network through back-propagation that can classify non-linearly separable data [26,27]. The learning rate is set to 0.3 and momentum applied during updating weights is set to 0.2. The validation threshold use to terminate the validation testing is set to 20.
  - VII. K nearest neighbors (KNN): Instance based learning or lazy learning which trains the classifier function locally by majority note of its neighboring data points. Linear NN Search algorithm is used for search algorithm [28,29]. K is set to 3.
  - VIII. K Star: Instance based classifier that uses similarity function from the training set to classify test set. Missing values are averaged by column entropy curves and global blending parameter is set to 20 [30].
  - IX. Attribute Selected Classifier (ASC): 'Cfs subset' evaluator is used during the attribute selection phase to reduce the dimension of training and test data. The 'BestFit' search method is invoked after which J48 tree classifier is used [31].
  - X. Classification via clustering (K means): Simple k means clustering method is used where k is set to the number of classes in the data set [32]. Euclidean distance was used for evaluation with 500 iterations.
  - XI. Classification via Regression (M5P): Regression is a method used to evaluate the relationship between dependent and independent variables through an empirically determined function. The M5P base classifier is used which combines conventional decision tree with the possibility of linear regression at the nodes. The minimum number of instances per leaf node is set to 4 [33].
  - XII. Linear Discriminant Analysis (LDA): Prevalent classification technique that identifies the combination of features that best characterizes classes through linear relationships. Prior probabilities are set to uniform and the model as homoscedastic.
  - XIII. Hyper Pipes: Simple, fast classifier that counts internally defined attributes for all samples and compares the number of instances of each attribute per sample. Classification is based on simple counts. Works well when there are many attributes [34].
  - XIV. VFI: Voting feature interval classifier is a simple heuristic attribute-weighting scheme. Intervals are constructed for numeric attributes. For each feature per interval, class counts are recorded and classification is done by voting. Higher weight is assigned to more confident intervals. The strength of the bias towards more confident features is set to 0 [35].
  - XV. J48: Java implementation of C4.5 algorithm. Based on the Hunt's algorithm, pruning takes place by replacing internal node with a leaf node. Top-down decision tree/voting algorithm [36]. 0.25 is used for the confidence factor. No Laplace method for tree smoothing [37].
  - XVI. Random Trees: A tree is grown from data that has K randomly chosen attributes at each node. It does not perform pruning. K-value ( $\log_2$  (number of attributes) + 1) is set at zero. There is no depth restriction. The minimum total weight per leaf is set to 1 [34].
  - XVII. Random Forest (R. Forest): Like Random Tree, the algorithm constructs a forest of random trees [38] with locations of attributes chosen at random. It uses an ensemble of unprune decision trees by a bootstrap sample using training data. There is no restriction on the depth of the tree; number of tress used is 100.
- Time performance**
- CPU time was calculated for every algorithm at the four different significance levels. This time was measured on a standard PC (Intel dual core, 2.2 GHz 3 Gb RAM) that was completely dedicated to WEKA. To measure CPU

time, open source jar files from WEKA were imported to Eclipse where the function 'time ()' was invoked prior to running the classification including the time required for cross validation. Most Windows 7 services were switched off; the times reported were an average of 5 different measurements.

## Results

### Overall performance accuracy of classification algorithms over all data sets

For each dataset, accuracies are measured at four levels (top 10, 50,200, 1000 peptides) at various levels of significance. Overall average performance measure is calculated for each algorithm for a given data set. Table 1 shows the overall average percentage score for each algorithm calculated by averaging accuracy, specificity, sensitivity and area under ROC curve under all levels of significance. Scores >90% are marked in bold. MLP algorithm did not finish due to huge memory requirements on last level of significance and is averaged based on first three levels of significance. For type 1 diabetes, Alzheimer's and antibodies dataset, >6 algorithms scored >90% average score. Overall, Naïve Bayes had the highest average score (90.4%) and was always among top 3 algorithms among all datasets.

### Performance accuracy of classification algorithms at different levels of significance over all data sets

For each data set, different levels of significance are chosen to measure the performance accuracy of each

algorithm. These levels contain approximately equal number of peptides for each data set. The first level contains 10 peptides selected from the *t*-test (lowest p value) and hence contains the least noise. Next, approximately 50 peptides, 200 peptides and 1000 peptides were chosen for the other three levels.

Tables 2, 3, 4, 5, 6, 7, 8 shows 4 different performance measures (accuracy, specificity, sensitivity and area under ROC curve) at different levels of significance over 7 datasets. For the Asthma dataset, we considered all conditions A-D together, then performed the pair-wise comparisons of condition A and B, condition A and C, and condition B and D at three different levels of significance. Measures >90% are marked in bold. For the diabetes dataset, 9 algorithms achieved >90% score. For Alzheimer's and the Antibodies dataset, 6 algorithms achieved >90% score. Naïve Bayes scored 100% in all 4 measures at the first level of significance in the Alzheimer's dataset and scored 91.5% average score on the Antibodies dataset. For the Asthma datasets, the highest score was <80%. Only Naïve Bayes had >90% specificity for more than one level of significance. For two conditions in Asthma datasets, Naïve Bayes and VFI scored >90% average score.

### Comparative analysis of worst time performance of classification algorithms over data sets

The amount of time taken by each algorithm to build the model and perform cross validation was measured.

**Table 1 Overall performance measure of classification algorithms on datasets**

Algorithms	T1D	Az	Ab	Asthma	A & B	A & C	B & D	Avg.	Rank
Naïve Bayes	<b>92.0</b>	<b>93.4</b>	<b>91.5</b>	77.7	<b>90.8</b>	<b>93.5</b>	<b>93.6</b>	<b>90.4</b>	<b>1</b>
MLP	<b>90.1</b>	<b>92.7</b>	<b>90.2</b>	71.1	84.7	<b>92.7</b>	89.3	87.3	<b>2</b>
SVM	<b>91.6</b>	88.0	<b>90.7</b>	71.3	86.1	88.4	<b>93.1</b>	87.0	<b>3</b>
VFI	<b>90.5</b>	<b>92.2</b>	75.5	62.6	87.7	<b>93.4</b>	<b>92.7</b>	84.9	<b>4</b>
Hyper Pipes	89.8	89.7	81.3	62.3	82.0	86.6	87.8	82.8	<b>5</b>
R. Forest	<b>91.5</b>	82.4	<b>93.3</b>	62.8	80.6	81.4	81.1	81.9	<b>6</b>
Bayes Net	<b>90.3</b>	87.7	<b>92.5</b>	53.9	80.2	83.2	85.1	81.8	<b>7</b>
K-means	88.3	<b>91.8</b>	80.7	59.6	77.8	83.3	83.6	80.7	<b>8</b>
Logistic R.	<b>90.6</b>	<b>93.3</b>	60.4	50.7	81.5	84.8	<b>90.7</b>	78.9	<b>9</b>
SLR	<b>92.2</b>	71.8	<b>90.1</b>	72.2	65.0	68.5	84.7	77.8	<b>10</b>
KNN	<b>91.4</b>	81.5	52.5	55.8	87.5	75.7	89.0	76.2	<b>11</b>
K star	81.9	<b>90.7</b>	89.4	53.5	64.3	68.8	70.7	74.2	<b>12</b>
MSP	85.1	58.7	83.2	60.0	75.2	73.4	79.6	73.6	<b>13</b>
J48	80.3	69.7	78.4	48.7	70.6	68.4	76.7	70.4	<b>14</b>
Random Tree	83.8	71.7	76.2	52.9	69.3	60.8	75.0	70.0	<b>15</b>
ASC	76.8	70.0	77.9	43.1	72.0	63.1	76.7	68.5	<b>16</b>
LDA	69.7	52.0	89.1	70.8	62.8	69.7	52.6	66.7	<b>17</b>

T1D: Type 1 diabetes datasets, Az: Alzheimer's dataset, Ab: Antibodies dataset. Table showing algorithms overall performance in each datasets based on average score. Score >90% are marked in bold. Naïve Bayes scored the overall highest average score of 90.4%.

**Table 2 Performance measures of data mining algorithm at different levels of significance over Type 1 diabetes dataset**

SIGNIFICANCE	$p < 5 \times 10^{-13}$				$p < 5 \times 10^{-10}$				$p < 5 \times 10^{-7}$				$p < 5 \times 10^{-4}$				Avg.
	Algorithm	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	
SLR	87.5	85.0	89.7	<b>0.93</b>	<b>92.5</b>	<b>90.2</b>	<b>94.9</b>	<b>0.97</b>	<b>92.5</b>	<b>92.0</b>	<b>92.0</b>	<b>0.96</b>	<b>92.5</b>	<b>90.0</b>	<b>94.9</b>	<b>0.96</b>	<b>92.2</b>
Naïve Bayes	<b>90.0</b>	85.4	<b>95.0</b>	<b>0.97</b>	<b>91.3</b>	<b>90.2</b>	<b>92.3</b>	<b>0.98</b>	<b>92.5</b>	<b>90.2</b>	<b>95.0</b>	<b>0.96</b>	89.0	85.4	<b>92.3</b>	<b>0.92</b>	<b>92.0</b>
SVM	88.8	82.9	<b>94.9</b>	0.89	<b>90.0</b>	82.9	<b>97.4</b>	<b>0.90</b>	<b>93.8</b>	<b>90.2</b>	<b>97.4</b>	<b>0.93</b>	<b>93.8</b>	<b>92.7</b>	<b>94.9</b>	<b>0.94</b>	<b>91.6</b>
R. Forest	87.5	87.8	87.2	<b>0.96</b>	<b>92.5</b>	<b>90.2</b>	<b>94.9</b>	<b>0.97</b>	<b>91.5</b>	87.8	<b>94.9</b>	<b>0.97</b>	88.8	85.4	<b>92.3</b>	<b>0.94</b>	<b>91.5</b>
KNN	<b>92.5</b>	<b>90.2</b>	<b>94.9</b>	<b>0.95</b>	<b>95.0</b>	<b>92.7</b>	<b>97.4</b>	<b>0.96</b>	<b>90.0</b>	85.4	<b>94.9</b>	<b>0.93</b>	85.0	80.5	89.7	<b>0.90</b>	<b>91.4</b>
Logistic. R	86.3	87.8	84.6	0.82	<b>92.5</b>	<b>90.2</b>	<b>94.9</b>	<b>0.97</b>	<b>92.5</b>	<b>92.7</b>	<b>97.4</b>	<b>0.97</b>	87.5	<b>92.7</b>	82.1	<b>0.92</b>	<b>90.6</b>
VFI	87.5	82.9	<b>92.3</b>	<b>0.95</b>	<b>92.5</b>	<b>90.2</b>	<b>94.9</b>	<b>0.97</b>	88.8	85.4	<b>92.3</b>	<b>0.95</b>	87.5	82.9	<b>92.3</b>	<b>0.92</b>	<b>90.5</b>
Bayes Net	<b>91.3</b>	<b>90.2</b>	<b>92.3</b>	<b>0.97</b>	<b>90.0</b>	85.4	<b>94.9</b>	<b>0.98</b>	<b>90.0</b>	85.4	<b>94.9</b>	<b>0.95</b>	83.8	78.0	89.7	0.89	<b>90.3</b>
MLP	80.0	80.5	79.5	0.89	<b>91.3</b>	<b>90.2</b>	<b>92.3</b>	<b>0.98</b>	<b>93.8</b>	<b>90.2</b>	<b>97.4</b>	<b>0.99</b>	dnf	dnf	dnf	dnf	<b>90.1*</b>
Hyper Pipes	87.5	<b>90.2</b>	84.6	<b>0.96</b>	<b>91.3</b>	<b>90.2</b>	<b>92.3</b>	<b>0.97</b>	<b>90.0</b>	<b>90.2</b>	89.7	<b>0.95</b>	83.8	<b>92.7</b>	74.4	<b>0.92</b>	89.8
K-means	<b>91.3</b>	82.9	<b>100</b>	<b>0.92</b>	<b>90.0</b>	82.9	<b>97.4</b>	<b>0.90</b>	86.3	78.0	<b>94.9</b>	0.87	85.0	75.6	<b>94.9</b>	0.85	88.3
M5P	88.8	85.4	<b>92.3</b>	<b>0.94</b>	85.0	80.5	89.7	<b>0.94</b>	81.3	78.0	84.6	0.87	78.8	73.2	84.6	0.85	85.1
Random Tree	85.0	87.8	82.1	0.85	78.8	75.6	82.1	0.79	87.5	85.4	89.7	0.88	83.8	85.4	82.1	0.84	83.8
K star	87.5	87.8	87.2	<b>0.96</b>	<b>91.3</b>	85.4	<b>97.4</b>	0.98	<b>90.0</b>	85.4	<b>94.9</b>	<b>0.97</b>	53.8	<b>100</b>	5.1	0.54	81.9
J48	86.3	85.4	87.2	0.79	81.3	82.9	79.5	0.83	78.8	82.9	74.4	0.72	80.0	85.4	74.4	0.73	80.3
ASC	86.3	85.4	87.2	0.79	80.0	82.9	76.9	0.80	80.0	87.8	71.8	0.78	66.3	80.5	51.3	0.55	76.8
LDA	88.8	82.9	<b>94.9</b>	<b>0.96</b>	<b>91.3</b>	85.4	<b>97.4</b>	0.95	40.0	<b>96.7</b>	15.8	0.68	21.3	<b>94.4</b>	0.0	0.48	69.7

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, dnf: "Did Not Finish", \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.

**Table 3 Performance measures of data mining algorithm at different levels of significance over Alzheimer's dataset**

SIGNIFICANCE	$p < 5 \times 10^{-5}$				$p < 5 \times 10^{-4}$				$p < 5 \times 10^{-3}$				$p < 5 \times 10^{-2}$				Avg.
	Algorithm	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	
Naïve Bayes	<b>100</b>	<b>100</b>	<b>100</b>	<b>1.00</b>	<b>91.3</b>	82.0	<b>100</b>	<b>0.96</b>	<b>91.3</b>	82.0	<b>100</b>	<b>0.96</b>	86.5	<b>91.0</b>	84.0	<b>0.94</b>	<b>93.4</b>
Logistic. R	<b>95.0</b>	<b>90.0</b>	<b>100</b>	<b>0.99</b>	<b>95.7</b>	<b>90.0</b>	<b>100</b>	<b>0.97</b>	<b>91.3</b>	<b>90.0</b>	<b>91.7</b>	<b>0.90</b>	<b>91.3</b>	<b>90.0</b>	<b>91.7</b>	<b>0.90</b>	<b>93.3</b>
MLP	<b>91.3</b>	<b>90.9</b>	<b>91.7</b>	<b>0.97</b>	<b>95.6</b>	<b>90.9</b>	<b>100</b>	<b>0.97</b>	87.0	<b>90.9</b>	83.3	<b>0.97</b>	dnf	dnf	dnf	dnf	<b>92.7*</b>
VFI	<b>91.3</b>	<b>90.9</b>	<b>91.7</b>	0.87	<b>95.7</b>	<b>90.9</b>	<b>100</b>	<b>0.92</b>	<b>91.3</b>	81.8	<b>100</b>	0.89	<b>91.3</b>	81.8	<b>100</b>	<b>1.00</b>	<b>92.2</b>
KNN	<b>91.3</b>	<b>90.9</b>	<b>91.7</b>	<b>0.93</b>	<b>95.6</b>	<b>90.9</b>	<b>100</b>	<b>0.93</b>	86.9	<b>90.9</b>	83.3	<b>0.95</b>	<b>91.3</b>	<b>90.9</b>	<b>91.7</b>	<b>0.92</b>	<b>91.8</b>
K-means	82.6	<b>100</b>	66.7	0.83	<b>91.3</b>	<b>90.9</b>	<b>100</b>	<b>0.91</b>	<b>95.7</b>	<b>90.9</b>	<b>100</b>	<b>0.96</b>	<b>91.3</b>	81.8	<b>100</b>	<b>0.90</b>	<b>90.7</b>
Hyper Pipes	<b>91.3</b>	81.8	<b>100</b>	<b>0.98</b>	<b>95.7</b>	<b>90.9</b>	<b>100</b>	<b>0.97</b>	<b>91.3</b>	81.8	<b>100</b>	<b>0.95</b>	73.9	81.8	66.7	<b>0.90</b>	89.7
SVM	87.0	<b>90.9</b>	83.3	0.87	<b>95.7</b>	<b>90.9</b>	<b>100</b>	<b>0.95</b>	82.6	81.8	83.3	0.83	87.0	81.8	<b>91.7</b>	0.87	88.0
Bayes Net	<b>91.3</b>	81.8	<b>100</b>	<b>0.96</b>	<b>91.3</b>	<b>90.9</b>	<b>91.7</b>	<b>0.95</b>	87.0	81.8	<b>91.7</b>	0.86	78.3	81.8	75.0	0.84	87.7
R. Forest	86.9	81.8	<b>91.7</b>	<b>0.94</b>	82.6	81.8	83.3	<b>0.93</b>	73.9	72.7	75.0	0.89	72.6	81.8	75.0	0.84	82.4
K star	<b>95.7</b>	<b>90.9</b>	<b>100</b>	<b>0.98</b>	<b>91.3</b>	<b>90.9</b>	<b>91.7</b>	<b>0.94</b>	78.2	81.8	75.0	0.86	56.5	18.2	<b>91.7</b>	0.64	81.5
SLR	86.9	81.8	<b>91.7</b>	<b>0.96</b>	73.9	72.7	75.0	0.82	60.9	63.6	58.3	0.80	52.2	54.5	50.0	0.69	71.8
Random Tree	78.3	72.7	83.3	0.78	60.9	54.5	66.7	0.61	73.9	63.6	83.3	0.74	73.9	81.8	66.7	0.74	71.7
ASC	73.9	63.6	83.3	0.61	68.9	63.6	58.3	0.56	73.9	81.8	66.7	0.75	78.2	63.9	<b>91.7</b>	0.61	70.0
J48	73.9	63.6	83.3	0.61	60.9	63.6	58.3	0.56	73.9	81.8	70.0	0.75	78.3	63.6	<b>91.7</b>	0.61	69.7
M5P	69.5	54.5	83.3	0.80	52.2	45.5	58.3	0.73	56.5	45.5	66.7	0.43	56.5	36.4	75.0	0.44	58.7
LDA	69.6	72.7	66.7	0.81	34.8	40.0	75.0	0.45	34.8	0.0	<b>100</b>	0.30	30.4	<b>100</b>	0.0	0.52	52.0

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, dnf: Did not Finish, \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.

**Table 4 Performance measures of data mining algorithm at different levels of significance over Antibodies dataset**

SIGNIFICANCE	$p < 5 \times 10^{-8}$				$p < 5 \times 10^{-7}$				$p < 5 \times 10^{-6}$				$p < 5 \times 10^{-5}$				Avg.
	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	
<b>R. Forest</b>	<b>90.0</b>	<b>93.0</b>	<b>90.0</b>	<b>0.96</b>	<b>90.0</b>	<b>91.0</b>	<b>90.0</b>	<b>0.97</b>	<b>92.0</b>	<b>94.0</b>	<b>92.0</b>	<b>0.96</b>	<b>94.0</b>	<b>96.0</b>	<b>94.0</b>	<b>0.97</b>	<b>93.3</b>
<b>Bayes Net</b>	88.0	<b>92.0</b>	88.0	<b>0.96</b>	88.0	<b>91.0</b>	88.0	<b>0.96</b>	<b>94.0</b>	<b>95.0</b>	<b>94.0</b>	<b>0.95</b>	<b>92.0</b>	<b>95.0</b>	<b>92.0</b>	<b>0.96</b>	<b>92.5</b>
<b>Naïve Bayes</b>	88.0	<b>94.0</b>	88.0	<b>0.96</b>	88.0	<b>94.0</b>	88.0	<b>0.96</b>	88.0	<b>94.0</b>	88.0	<b>0.96</b>	88.0	<b>94.0</b>	88.0	<b>0.96</b>	<b>91.5</b>
<b>SVM</b>	80.0	86.6	80.0	0.86	86.0	89.9	86.0	0.89	<b>94.0</b>	<b>96.6</b>	<b>97.0</b>	<b>0.95</b>	<b>96.0</b>	<b>96.9</b>	<b>96.0</b>	<b>0.96</b>	<b>90.7</b>
<b>MLP</b>	80.0	89.8	80.0	<b>0.91</b>	86.0	89.9	86.0	<b>0.96</b>	<b>94.0</b>	<b>96.6</b>	<b>94.0</b>	<b>0.99</b>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<b>90.2*</b>
<b>SLR</b>	84.0	<b>91.6</b>	84.0	0.89	86.0	83.2	86.0	<b>0.92</b>	<b>90.0</b>	<b>93.5</b>	<b>90.0</b>	<b>0.97</b>	<b>92.0</b>	<b>95.0</b>	<b>92.0</b>	<b>0.96</b>	<b>90.1</b>
<b>KNN</b>	82.0	<b>90.7</b>	82.0	<b>0.92</b>	84.0	88.7	84.0	<b>0.94</b>	86.0	<b>91.2</b>	86.0	<b>0.95</b>	<b>92.0</b>	<b>96.4</b>	<b>92.0</b>	<b>0.95</b>	89.4
<b>Logistic R.</b>	72.0	85.3	72.0	<b>0.92</b>	84.0	<b>90.1</b>	84.0	<b>0.93</b>	<b>92.0</b>	<b>96.4</b>	<b>92.0</b>	<b>0.98</b>	<b>90.0</b>	<b>96.1</b>	<b>90.0</b>	<b>0.98</b>	89.1
<b>M5P</b>	80.0	<b>91.5</b>	80.0	<b>0.92</b>	76.0	87.4	76.0	<b>0.90</b>	78.0	89.4	78.0	<b>0.91</b>	74.0	85.4	74.0	0.89	83.2
<b>Hyper Pipes</b>	64.0	83.6	64.0	<b>0.90</b>	72.0	84.9	72.0	<b>0.90</b>	80.0	87.5	<b>80.0</b>	<b>0.92</b>	80.0	87.1	80.0	<b>0.93</b>	81.3
<b>K star</b>	88.0	<b>93.4</b>	88.0	<b>0.94</b>	<b>94.0</b>	<b>97.2</b>	<b>94.0</b>	<b>0.95</b>	82.0	<b>91.8</b>	<b>82.0</b>	<b>0.93</b>	20.0	<b>90.2</b>	20.8	0.68	80.7
<b>J48</b>	80.0	<b>92.5</b>	80.0	0.86	72.0	87.0	72.0	0.87	70.0	87.6	70.0	0.79	64.0	86.1	64.0	0.77	78.4
<b>ASC</b>	82.0	<b>91.7</b>	82.0	0.87	72.0	82.9	72.0	0.82	70.0	87.8	70.0	0.76	64.0	88.5	64.0	0.75	77.9
<b>Random Tree</b>	72.0	<b>90.3</b>	72.0	0.81	64.0	82.1	64.0	0.73	68.0	87.7	68.0	0.78	74.0	89.7	74.0	0.82	76.2
<b>VFI</b>	72.0	88.5	72.0	0.86	64.0	<b>91.9</b>	64.0	0.85	58.0	<b>94.7</b>	58.0	0.86	52.0	<b>94.5</b>	52.0	0.89	75.5
<b>LDA</b>	68.0	84.5	68.0	0.88	40.0	81.1	40.0	0.71	42.0	89.7	48.8	0.54	20.0	88.4	25.0	0.58	60.4
<b>K means</b>	46.0	68.7	46.0	0.57	46.0	68.7	46.0	0.57	40.0	68.1	40.0	0.54	40.0	68.1	40.0	0.54	52.5

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, *dnf*: Did not Finish\*, \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.

**Table 5 Performance measures of data mining algorithm at different levels of significance over Asthma dataset 4 classes**

SIGNIFICANCE	$p < 5 \times 10^{-5}$				$p < 5 \times 10^{-4}$				$p < 5 \times 10^{-3}$				$p < 5 \times 10^{-2}$				Avg.
	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	
<b>Naïve Bayes</b>	61.7	87.2	61.7	0.82	68.1	89.3	68.1	0.86	72.3	<b>90.8</b>	72.3	0.87	70.2	<b>90.0</b>	70.2	0.86	77.7
<b>SLR</b>	57.5	85.8	57.4	0.80	57.4	85.6	57.4	0.81	72.3	<b>90.7</b>	72.3	0.85	55.3	86.1	55.3	0.76	72.2
<b>SVM</b>	55.3	86.2	55.3	0.77	55.3	86.2	55.3	0.77	61.7	87.2	61.7	0.82	66.0	87.6	66.0	0.81	71.3
<b>MLP</b>	55.3	86.1	55.3	0.82	53.2	84.6	53.2	0.80	63.8	87.8	63.8	0.88	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	71.1*
<b>Logistic R.</b>	48.9	87.0	48.9	0.78	53.2	84.4	53.2	0.79	59.6	86.4	59.6	0.84	68.0	89.2	68.1	0.86	70.8
<b>R. Forest</b>	48.9	86.9	48.9	0.77	48.9	86.9	48.9	0.77	46.8	81.1	46.8	0.75	40.4	80.0	40.4	0.71	62.8
<b>VFI</b>	48.9	82.8	48.9	0.66	48.9	82.9	48.9	0.67	51.0	83.6	51.1	0.69	46.8	81.9	46.8	0.77	62.6
<b>Hyper Pipes</b>	51.1	83.4	51.1	0.72	53.2	84.0	53.2	0.70	46.8	71.8	46.8	0.74	42.6	80.3	42.0	0.75	62.3
<b>M5P</b>	48.9	82.8	48.9	0.79	55.3	86.1	55.3	0.81	42.5	81.0	42.6	0.68	27.6	75.8	27.7	0.57	60.0
<b>KNN</b>	42.5	87.1	42.6	0.69	46.8	86.6	46.8	0.67	44.6	88.0	44.7	0.69	36.2	79.7	36.2	0.67	59.6
<b>K means</b>	40.4	81.9	40.4	0.60	46.8	82.2	46.8	0.65	42.6	80.7	42.6	0.62	34.0	78.0	34.0	0.56	55.8
<b>Bayes Net</b>	38.3	79.3	38.3	0.56	36.2	77.8	36.2	0.56	44.7	81.4	44.7	0.63	36.2	77.6	36.2	0.60	53.9
<b>K star</b>	48.9	83.0	48.9	0.70	38.3	79.4	38.3	0.63	36.2	79.4	36.2	0.62	23.4	76.4	23.4	0.49	53.5
<b>Random Tree</b>	29.8	76.6	29.8	0.53	40.4	80.2	40.4	0.60	38.3	79.5	38.3	0.59	40.4	80.2	40.4	0.60	52.9
<b>LDA</b>	53.2	84.4	53.2	0.80	27.7	80.0	32.5	0.57	8.5	86.5	16.7	0.56	14.9	83.6	23.3	0.53	50.7
<b>J48</b>	27.7	75.4	27.7	0.52	27.7	75.9	27.7	0.49	42.6	80.8	42.6	0.58	31.9	77.1	31.9	0.52	48.7
<b>ASC</b>	27.7	76.0	27.7	0.52	19.2	71.8	19.1	0.46	29.8	76.7	29.8	0.52	21.2	74.8	21.3	0.45	43.1

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, *dnf*: Did not Finish\*, \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.



**Table 6 Performance measures of data mining algorithm at different levels of significance on A & B conditions**

SIGNIFICANCE	$p < 5 \times 10^{-4}$				$p < 5 \times 10^{-3}$				$p < 5 \times 10^{-2}$				Avg.
	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	
Naïve Bayes	87.5	83.3	<b>91.7</b>	0.84	<b>91.7</b>	83.3	<b>100</b>	<b>0.97</b>	<b>91.7</b>	83.3	<b>100</b>	<b>0.96</b>	<b>90.8</b>
VFI	79.2	75.0	83.3	<b>0.93</b>	<b>91.7</b>	83.3	<b>100</b>	<b>0.95</b>	87.5	75.0	<b>100</b>	<b>0.90</b>	87.7
K means	87.5	83.3	<b>91.7</b>	0.88	<b>91.7</b>	83.3	<b>100</b>	<b>0.92</b>	83.3	75.0	<b>91.7</b>	0.83	87.5
SVM	83.3	83.3	83.3	0.83	87.5	<b>91.7</b>	83.3	0.87	87.5	83.3	<b>91.7</b>	0.88	86.1
MLP	79.2	83.3	75.0	0.70	<b>91.7</b>	<b>91.7</b>	<b>91.7</b>	<b>0.95</b>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	84.7*
Hyper Pipes	83.3	75.0	<b>91.7</b>	<b>0.91</b>	83.3	83.3	83.3	<b>0.93</b>	70.8	83.3	58.3	0.88	82.0
Logistic R.	66.7	83.3	50.0	0.76	<b>95.8</b>	<b>91.7</b>	<b>100</b>	<b>0.92</b>	79.2	83.3	75.0	0.85	81.5
Random Forest	79.2	83.3	75.0	<b>0.91</b>	79.2	75.0	83.3	0.86	79.2	75.0	83.3	0.78	80.6
Bayes Net	83.3	75.0	<b>91.7</b>	0.87	83.3	83.3	83.3	0.83	75.0	75.0	75.0	0.67	80.2
KNN	75.0	83.3	66.7	0.85	75.0	<b>91.7</b>	58.3	<b>0.90</b>	75.0	<b>91.7</b>	58.3	0.84	77.8
M5P	75.0	83.3	66.7	0.74	75.0	75.0	75.0	0.79	75.0	75.0	75.0	0.74	75.2
ASC	62.5	66.7	58.3	0.65	79.2	83.3	75.0	0.85	70.8	75.0	66.7	0.76	72.0
J48	62.5	66.7	58.3	0.65	79.2	83.3	75.0	0.85	66.7	75.0	58.3	0.72	70.6
Random Tree	70.8	75.0	66.7	0.70	70.8	75.0	66.7	0.70	66.7	66.7	66.7	0.67	69.3
SLR	70.8	75.0	66.7	0.80	66.7	75.0	58.3	0.77	50.0	50.0	50.0	0.60	65.0
K star	66.7	<b>91.7</b>	41.7	0.83	58.3	<b>100</b>	46.7	0.83	50.0	0.0	<b>100</b>	0.50	64.3
LDA	79.2	83.3	75.0	0.84	61.2	64.5	54.5	0.52	29.2	14.3	<b>100</b>	0.56	62.8

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, *dnf*: Did not Finish\*, \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.

Table 9 shows the time in milliseconds for each algorithm at the lowest level of significance when the number of peptides nears 1000. Random Tree was the

fastest, at ~1000 milliseconds (average) to complete the task, while MLP was the worst which did not finish due to high memory requirements. Random tree, Hyper

**Table 7 Performance measures of data mining algorithm at different levels of significance on A & C conditions**

SIGNIFICANCE	$p < 5 \times 10^{-4}$				$p < 5 \times 10^{-3}$				$p < 5 \times 10^{-2}$				Avg.
	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	
Naïve Bayes	<b>91.3</b>	<b>91.7</b>	<b>91.0</b>	<b>0.94</b>	<b>96.0</b>	<b>100</b>	<b>90.9</b>	<b>0.99</b>	<b>91.3</b>	<b>100</b>	81.8	<b>0.95</b>	<b>93.5</b>
VFI	<b>95.6</b>	<b>100</b>	<b>90.0</b>	<b>0.97</b>	<b>95.6</b>	<b>100</b>	<b>90.0</b>	<b>0.97</b>	87.0	83.3	<b>90.0</b>	<b>0.95</b>	<b>93.4</b>
MLP	86.9	<b>91.7</b>	81.8	<b>0.97</b>	<b>95.6</b>	<b>100</b>	<b>90.9</b>	<b>0.98</b>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<b>92.7*</b>
SVM	<b>95.6</b>	<b>100</b>	<b>90.9</b>	<b>0.96</b>	<b>95.7</b>	<b>100</b>	<b>90.9</b>	<b>0.96</b>	73.9	75.0	72.7	0.74	88.4
Hyper Pipes	<b>95.7</b>	<b>100</b>	<b>90.9</b>	<b>0.99</b>	82.6	<b>91.7</b>	72.7	<b>0.90</b>	78.2	83.3	72.7	0.83	86.6
Logistic R.	86.0	<b>91.7</b>	81.8	<b>0.96</b>	<b>95.7</b>	<b>100</b>	<b>90.9</b>	<b>0.92</b>	69.6	83.3	54.5	0.76	84.8
KNN	<b>91.3</b>	<b>100</b>	81.8	<b>0.92</b>	<b>91.3</b>	<b>100</b>	81.8	<b>0.94</b>	65.2	66.7	63.6	0.72	83.3
Bayes Net	<b>95.7</b>	<b>100</b>	<b>90.9</b>	<b>0.99</b>	82.6	83.3	81.8	<b>0.92</b>	69.6	66.7	72.7	0.64	83.2
Random Forest	87.0	83.3	<b>90.9</b>	<b>0.93</b>	82.6	83.3	81.8	<b>0.91</b>	69.5	66.7	72.7	0.75	81.4
K means	69.6	83.3	54.5	0.69	<b>95.7</b>	<b>100</b>	<b>90.9</b>	<b>0.95</b>	60.9	63.6	63.6	0.63	75.7
M5P	<b>91.3</b>	<b>91.7</b>	<b>90.9</b>	0.86	65.2	58.3	72.7	0.72	65.2	58.3	72.7	0.56	73.4
LDA	<b>91.3</b>	<b>100</b>	81.8	<b>0.97</b>	65.2	71.7	58.6	0.77	17.4	25.0	<b>100</b>	0.52	69.7
K star	73.9	<b>91.7</b>	54.5	<b>0.93</b>	78.2	<b>100</b>	54.5	0.82	47.8	0.0	<b>100</b>	0.50	68.8
SLR	87.0	83.3	<b>90.9</b>	0.89	73.9	75.0	72.7	0.74	43.5	41.7	45.5	0.45	68.5
J48	69.6	66.7	72.7	0.76	69.6	58.3	81.8	0.77	60.9	58.3	63.6	0.66	68.4
ASC	65.6	66.7	72.7	0.76	69.6	66.7	72.7	0.76	47.8	66.7	27.3	0.49	63.1
Random Tree	73.9	<b>91.7</b>	54.5	0.73	73.9	66.7	81.8	0.74	34.8	33.3	36.4	0.35	60.8

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, *dnf*: Did not Finish\*, \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.

**Table 8 Performance measures of data mining algorithm at different levels of significance on B & D conditions**

SIGNIFICANCE	$p < 5 \times 10^{-4}$				$p < 5 \times 10^{-3}$				$p < 5 \times 10^{-2}$				Avg.
	Algorithm	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	AUC	Acc.	Sp	Sn	
Naïve Bayes	<b>91.7</b>	<b>100</b>	83.3	<b>0.95</b>	<b>91.7</b>	<b>91.7</b>	<b>91.7</b>	<b>0.92</b>	<b>95.8</b>	<b>91.7</b>	<b>100</b>	<b>0.98</b>	<b>93.6</b>
SVM	<b>91.7</b>	<b>100</b>	83.3	<b>0.92</b>	<b>91.7</b>	<b>91.7</b>	<b>91.7</b>	<b>0.92</b>	<b>95.8</b>	<b>100</b>	<b>91.7</b>	<b>0.96</b>	<b>93.1</b>
VFI	87.5	<b>100</b>	75.0	<b>0.93</b>	<b>91.7</b>	<b>100</b>	83.3	<b>0.94</b>	<b>95.8</b>	<b>100</b>	<b>91.7</b>	<b>1.00</b>	<b>92.7</b>
Logistic R.	79.1	83.3	75.0	<b>0.92</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>1.00</b>	87.5	<b>91.7</b>	83.3	<b>0.97</b>	<b>90.7</b>
MLP	87.5	<b>91.7</b>	83.3	<b>0.94</b>	87.5	83.3	<b>91.7</b>	<b>0.96</b>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	89.3*
K means	87.5	<b>91.7</b>	83.3	0.88	<b>91.4</b>	<b>91.7</b>	<b>91.7</b>	<b>0.92</b>	87.5	83.3	<b>91.7</b>	0.88	89.0
Hyper Pipes	87.5	83.3	<b>91.7</b>	0.89	<b>91.7</b>	<b>91.7</b>	<b>91.7</b>	0.87	83.3	75.0	<b>91.7</b>	<b>0.90</b>	87.8
Bayes Net	83.3	83.3	83.3	0.89	87.5	<b>91.7</b>	83.3	0.86	83.3	83.3	83.3	0.84	85.1
SLR	83.3	83.3	83.3	0.88	79.2	66.7	<b>91.7</b>	<b>0.90</b>	87.5	<b>100</b>	75.0	0.89	84.7
KNN	79.2	75.0	83.3	0.80	83.3	83.3	83.3	0.83	87.5	<b>91.7</b>	83.3	<b>0.90</b>	83.6
Random Forest	83.3	83.3	83.3	0.83	79.2	83.3	75.0	0.84	79.2	83.3	75.0	0.81	81.1
M5P	87.5	<b>91.7</b>	83.3	0.88	79.2	83.3	75.0	0.73	75.0	83.3	66.7	0.69	79.6
ASC	<b>91.7</b>	<b>100</b>	83.3	0.83	75.0	83.3	66.7	0.61	70.8	75.0	66.7	0.64	76.7
J48	<b>91.7</b>	<b>100</b>	83.3	0.83	75.0	83.3	66.7	0.61	70.8	75.0	66.7	0.64	76.7
Random Tree	83.3	<b>91.7</b>	75.0	0.83	70.8	66.7	75.0	0.71	70.8	66.7	75.0	0.71	75.0
K star	70.8	66.7	75.0	0.83	79.2	75.0	83.3	0.82	58.3	<b>100</b>	16.7	0.58	70.7
LDA	62.5	72.3	60.9	0.75	50.0	65.0	48.0	0.71	20.8	42.6	18.6	0.45	52.6

Acc: Accuracy, Sp: Specificity, Sn: Sensitivity, AUC: Area under ROC curve, Avg: Average score in % for each algorithms, *dnf*: Did not Finish", \* denotes Avg. from 3 significance levels. Measures >90% are marked in bold.

**Table 9 Worst case time performance (in ms) of classification algorithms**

Data set	Diabetes	Alzheimer's	Antibodies	Avg. (in ms)	Rank
Random Tree	<b>1809</b>	<b>491</b>	<b>1478</b>	<b>1260</b>	<b>1</b>
KNN	<b>3016</b>	<b>607</b>	<b>910</b>	<b>1511</b>	<b>2</b>
Hyper Pipes	<b>2486</b>	<b>602</b>	<b>2180</b>	<b>1756</b>	<b>3</b>
Naïve Bayes	<b>4780</b>	<b>1158</b>	<b>2480</b>	<b>2806</b>	<b>4</b>
VFI	<b>7440</b>	<b>1357</b>	<b>3000</b>	<b>3932</b>	<b>5</b>
J48	16581	<b>1385</b>	11731	<b>9899</b>	<b>6</b>
K star	25974	<b>2348</b>	6341	11555	<b>7</b>
SVM	10496	<b>2722</b>	29008	14076	<b>8</b>
R. Forest	50087	<b>8032</b>	21452	26524	<b>9</b>
M5P	50290	<b>8563</b>	23452	27435	<b>10</b>
Bayes Net	55672	<b>9031</b>	25000	29901	<b>11</b>
K-means	85955	12405	29658	42672	<b>12</b>
SLR	632840	48215	605365	428806	<b>13</b>
LDA	658668	869523	632983	720391	<b>14</b>
Logistic R.	1589092	1146783	1315256	1350377	<b>15</b>
ASC	5444533	2465021	4565896	4158483	<b>16</b>
MLP	<i>dnf</i>	<i>dnf</i>	<i>dnf</i>	NA	<b>17</b>

Table showing time performance in milliseconds over >1000 peptides for three datasets. Random Tree, KNN, Hyper Pipes and VFI were among the fastest. MLP were among the slowest with *dnf*: "Did not finish". Time measurements less than 10 seconds are marked in bold.

Pipes, Naïve Bayes, VFI and KNN were the five fastest algorithms; each took less than ~4000 milliseconds to complete classification of >1,000 peptides. Logistic Regression and Attribute Selected Classifier, MLP were among the slowest algorithms taking more than 20 minutes to perform classification of >1,000 peptides. The absolute ranking for every algorithm was consistent per dataset; only three datasets have been considered to measure time performance.

#### Comparative analysis of time performance of classification algorithms at different levels of significance over three data sets

For each level of significance, time was measured for each algorithm to build the model and for cross validation. At the highest level of significance (about 10 peptides), each algorithm were fast enough to complete the task in under 25 seconds. Execution times increased as the level of significance was lowered due to the higher number of features and increased difficulty in constructing the model. Table 10 shows classification algorithms time performance at various levels of significance.

#### Results summary

We have explored several disparate classifiers using a relatively new type of microarray data: immunosignaturing data. The tested algorithms come from a broad family of

**Table 10 Time performance (in ms) of classification algorithms on datasets**

p value<	Diabetes dataset				Alzheimer's dataset				Antibodies dataset			
	5x10 <sup>-13</sup>	5x10 <sup>-10</sup>	5x10 <sup>-7</sup>	5x10 <sup>-4</sup>	5x10 <sup>-5</sup>	5x10 <sup>-4</sup>	5x10 <sup>-3</sup>	5x10 <sup>-2</sup>	5x10 <sup>-8</sup>	5x10 <sup>-7</sup>	5x10 <sup>-6</sup>	5x10 <sup>-5</sup>
<b>R. Tree</b>	<b>337</b>	<b>408</b>	<b>571</b>	<b>1809</b>	<b>184</b>	<b>200</b>	<b>218</b>	<b>491</b>	<b>250</b>	<b>265</b>	<b>608</b>	<b>1478</b>
<b>KNN</b>	<b>265</b>	<b>333</b>	<b>585</b>	<b>3016</b>	<b>130</b>	<b>156</b>	<b>239</b>	<b>607</b>	<b>187</b>	<b>234</b>	<b>414</b>	<b>910</b>
<b>Hyper Pipes</b>	<b>226</b>	<b>274</b>	<b>630</b>	<b>2486</b>	<b>119</b>	<b>259</b>	<b>423</b>	<b>602</b>	<b>281</b>	<b>312</b>	<b>736</b>	<b>2180</b>
<b>Naïve Bayes</b>	<b>250</b>	<b>456</b>	<b>1120</b>	<b>4780</b>	<b>182</b>	<b>340</b>	<b>500</b>	<b>1158</b>	<b>265</b>	<b>362</b>	<b>892</b>	<b>2480</b>
<b>VFI</b>	<b>299</b>	<b>561</b>	<b>1384</b>	<b>7440</b>	<b>187</b>	<b>337</b>	<b>623</b>	<b>1357</b>	<b>280</b>	<b>368</b>	<b>1379</b>	<b>3000</b>
<b>J48</b>	<b>415</b>	<b>833</b>	<b>3718</b>	16581	<b>166</b>	<b>256</b>	<b>712</b>	<b>1385</b>	<b>468</b>	<b>880</b>	<b>3011</b>	11731
<b>K star</b>	<b>468</b>	<b>1387</b>	<b>4150</b>	25974	<b>187</b>	<b>260</b>	<b>666</b>	<b>2349</b>	<b>299</b>	<b>562</b>	<b>2340</b>	<b>6341</b>
<b>SVM</b>	<b>3313</b>	<b>3635</b>	<b>5304</b>	10496	<b>1054</b>	<b>1108</b>	<b>1389</b>	<b>2722</b>	18297	18372	23712	29009
<b>R. Forest</b>	<b>5717</b>	11889	18254	50087	<b>952</b>	<b>1852</b>	<b>4843</b>	<b>8032</b>	<b>5004</b>	<b>6749</b>	13848	21452
<b>M5P</b>	<b>701</b>	<b>2583</b>	<b>7717</b>	50290	<b>290</b>	<b>524</b>	<b>2324</b>	<b>8563</b>	<b>2632</b>	<b>4711</b>	12033	23452
<b>Bayes Net</b>	<b>718</b>	<b>2087</b>	<b>5653</b>	55672	<b>334</b>	<b>662</b>	<b>4996</b>	<b>9031</b>	<b>733</b>	<b>1140</b>	<b>3394</b>	25000
<b>K means</b>	<b>2618</b>	<b>6651</b>	11876	85955	<b>593</b>	<b>1123</b>	<b>7212</b>	12405	<b>850</b>	<b>908</b>	<b>3442</b>	29658
<b>SLR</b>	11215	26380	79308	632840	<b>1330</b>	<b>3413</b>	22625	48215	17389	20649	89107	605365
<b>LDA</b>	<b>683</b>	<b>1044</b>	<b>7994</b>	658668	<b>402</b>	<b>699</b>	35568	869523	<b>1512</b>	<b>2018</b>	17373	632983
<b>Logistic R.</b>	<b>1204</b>	<b>2592</b>	24687	1589092	<b>629</b>	<b>1651</b>	48659	1146783	<b>1654</b>	<b>9379</b>	255103	1315256
<b>ASC</b>	<b>864</b>	<b>3504</b>	32836	5444533	<b>518</b>	<b>1859</b>	36849	2465021	<b>1217</b>	<b>1763</b>	25496	4565896
<b>MLP</b>	23759	314076	4572305	dnf	<b>2057</b>	30342	2789485	dnf	22916	156905	3277395	dnf

Table showing time performance in milliseconds on all level of significance for three datasets. MLP were among the slowest with dnf: "Did not finish". Time measurements less than 10 seconds are marked in bold.

approaches to classify data. We chose algorithms from Bayesian, regression, trees, multivariate and meta analysis and we believe we have sampled sufficiently that the results are relevant. From Table 2 we found that Naïve Bayes had a higher average performance than all other algorithms tested. Naïve Bayes achieved > 90% average for 2 classes datasets where there is a clear distinction between two classes. For the multi-class the Antibodies dataset, where there is a clear difference between different types of antibodies, Naïve Bayes scored 88% average accuracy and was ranked third, close to the 93.3% accuracy of random forest. On the Asthma dataset, containing four classes, none of the algorithms were able to achieve more than 75% accuracy. This matches the biological interpretation very well. Naïve Bayes outperformed all algorithms for speed and accuracy, achieving 77.7% average score overall. Naïve Bayes was one of the top five fastest algorithms, ~500 times faster than the logistic regression. A summary of the all algorithms performance measures and time is given in below and described in Table 11. Distance metrics have been defined to access performance measures for all algorithms compared to the highest scoring algorithm on a given dataset.

- I. Naïve Bayes: Naïve Bayes performed best overall with > 90% overall average score. It was always among the top 3 algorithms in all 7 comparisons. It ranked first 5 out of 7 times when comparing all

**Table 11 Summary of performance and time measures of classification algorithms**

	# Rank 1	# Rank 2	# >90%	Distance	Time
<b>Naïve Bayes</b>	5	1	6	<b>-0.3</b>	<b>2X</b>
<b>MLP</b>	0	0	4	<b>-3.4</b>	7615X
<b>SVM</b>	0	1	3	<b>-3.6</b>	11X
<b>VFI</b>	0	2	4	-5.7	<b>3X</b>
<b>Hyper Pipes</b>	0	0	0	-7.9	<b>1X</b>
<b>R. Forest</b>	1	0	2	-8.8	21X
<b>Bayes Net</b>	0	1	2	-8.8	24X
<b>K-means</b>	0	0	1	-9.9	34X
<b>Logistic R.</b>	0	1	3	-11.8	1072X
<b>SLR</b>	1	1	2	-12.9	340X
<b>KNN</b>	0	0	1	-14.4	<b>1X</b>
<b>K star</b>	0	0	1	-16.5	9X
<b>M5P</b>	0	0	0	-17.0	22X
<b>J48</b>	0	0	0	-20.2	8X
<b>Random Tree</b>	0	0	0	-20.7	<b>1X</b>
<b>ASC</b>	0	0	0	-22.1	3300X
<b>LDA</b>	0	0	0	-24.0	572X

#Rank 1, Rank 2: No. of times algorithm ranked 1<sup>st</sup> and 2<sup>nd</sup> on 7 datasets, # > 90%: No. of times algorithm scored overall average score >90% on 7 datasets, Distance: magnitude an algorithm trails behind on average from the Rank 1 for the datasets (5% or less distance are marked in bold). Time: performance slower with respective to fastest algorithm. Time performances slower by 5 folds to fastest algorithm are marked in bold.

- datasets. It was on an average just 0.3% behind the rank 1 algorithm in overall comparison. It is 2X slower than the fastest algorithm due to its mathematical properties. It would be feasible to perform large-scale classification studies using Naïve Bayes.
- II. Multilayer Perceptron (MLP): It ranked second with overall score of 87.3% and was very close to SVM. The overall score is biased since MLP did not finish for level containing ~1000 peptides and hence scored was averaged from just the three levels. It was the slowest algorithm and infeasible to perform large-scale classification.
  - III. Support Vector Machines (SVM): Although it ranked third, it was not significantly different from the MLP in terms of performance measures. It was 700X faster than MLP and achieved >90% measured accuracy 3 times. Both MLP and SVM were <5% behind the rank 1 algorithm on average.
  - IV. VFI: VFI ranked fourth in overall performance measures and was among the top 5 fastest algorithms due to its voting method. Four times it obtained >90% average overall accuracy and ranked 2<sup>nd</sup> twice.
  - V. Hyper Pipes: Hyper pipes ranked fifth overall in performance measures and was among the fastest of the tested algorithms, likely due to its inherently simplistic ranking method. It was <8% from first place 6 times.
  - VI. Random Forest: Random forest ranked sixth in overall performance measures and performed better on datasets having multiple classes (Antibodies and Asthma). It was 21 times slower than the fastest algorithm due to bootstrapping.
  - VII. Bayes net: Ranked in the middle for overall accuracy and time. It scored >90% overall measures twice. It was slower than the Naïve Bayes due to construction of networks in the form of an acyclic graph and it is relatively inefficient compared to Naïve Bayes due to the change in network topology during assessment of probability.
  - VIII. K means: K-means ranked eighth in overall performance measures and was 34X slower than the fastest algorithm in time performance due to the multiple iterations required to form clusters. It performed far better for 2 classes compared to multiple classes because guaranteed convergence, scalability and linear separation boundaries are more easily maintained.
  - IX. Logistic Regression: Logistic regression ranked ninth in overall accuracy. It was >90% three times. It was among the worst in time performance, being ~1000 times slower than the fastest algorithm as it needs to regress on high number of features. It is efficient for small numbers of features and sample sizes > 400.
  - X. Simple Logistic: It ranked tenth in overall performance measures and ranked first on the diabetes dataset. It ranked second in multiclass Asthma dataset. It was slow in time performance due to LogitBoost iterations.
  - XI. K nearest neighbors: It performed well on the 2 classes dataset but didn't perform as well for multi class datasets. It was >90% performance for only rather difficult Diabetes dataset. This may be related to evenly defined but diffuse clusters related to the subtle differences between the Asthma patients.
  - XII. K star: It performed >90% for only the Diabetes dataset and was 9 times slower than the fastest algorithm. This algorithm may also be sensitive to the even and diffuse clusters described by this dataset.
  - XIII. M5P: It did not perform well on either time performance or accuracy. It never achieved >90% average score and was 22 times slower than the fastest algorithm due to formation of comprehensive linear model for every interior node of the unpruned tree.
  - XIV. J48: Top 5 fastest algorithm due to rapid construction of trees. It was >20% behind from the rank 1 algorithm on an average; its lower performance may possibly be due to formation of empty/insignificant branches which often leads to overtraining.
  - XV. Random Trees: It was the fastest algorithm since it builds trees of height  $\log(k)$  where  $k$  is the number of attributes, however it achieves poor accuracy since it performs no pruning.
  - XVI. Attribute Selected Classifier (ASC): One of the slowest algorithms as it had to evaluate attributes prior to classification. It underperformed in performance measures due to the C4.5 classifier limitations that prevent overtraining.
  - XVII. Linear Discriminant Analysis (LDA): Its performance accuracy decreased as the number of features increased due to its inability to deal with highly variant data. It was slow (>500X slower than the fastest algorithm) since it tries to optimize class distinctions but the variance covariance matrix increases dramatically as the number of features increased.

## Discussion

The comparisons provided in this article provide a glimpse into how existing classification algorithms handle data with intrinsically different properties than traditional microarray expression data. Immunosignaturing provides a means to quantify the dispersion of serum (or saliva) antibodies that result from disease or other immune

challenge. Unlike most phage display or other panning experiments, fewer but longer random-sequence peptides are used. Rather than converging to relatively few sequences, the immunosignaturing microarray provides data on the binding affinity of all 10,000 peptides with high precision. Classifiers in the open-source program WEKA were used to determine whether any algorithm stood out as being particularly well suited for these data. The 17 classifiers, which were tested, are readily available and represent some of the most widely used classification methods in biology. However, they also represent classifiers that are diverse at the most fundamental levels. Tree methods, regression, and clustering are inherently different; the grouping methods are quite varied and top-down or bottom-up paradigms address data structures in substantially different ways. Given this, we present and interpret the results from our tests, which we believe will be applicable to any dataset with target-probe interactions similar to immunosignaturing microarrays.

From the comparisons above, Naïve Bayes was the superior analysis method in all aspects. Naïve Bayes assumes a feature independent model, which may account for its superior performance. It relies on the degree of correlation of the attributes in the dataset; for immunosignaturing, the number of attributes can be quite large. In gene expression data, where genes are connected by gene regulatory networks, there is a direct and significant correlation between hub genes and dependent genes. This relationship affects the performance of Naïve Bayes by limiting its efficiency through multiple containers of similarly - connected features [39-41]. In peptide-antibody arrays, where the signals that arise from the peptides are multiplexed signals of many antibodies attaching to many peptides, there is no direct correlation between peptides, but there is a general trend. Moreover, there is a competition of antibodies attaching to a single peptide, which makes it difficult for multiple mimotopes to show significant correlation with each other. Thus, the 10,000 random peptides have no direct relationships to each other each contributes partially to defining the disease state. This makes the immunosignaturing technology a better fit for the assumption of strong feature independence employed by the Naïve Bayes technique, and the fact that reproducible data can be had at intensity values down to 1 standard deviation above background enables enormous numbers of informative, precise, and independent features. Presence or absence of a few high- or low-binding peptides on the microarray will not impact the binding affinity for any other peptide, since the kinetics ensures that the antibody pool is not limiting. This is important when building microarrays with >300,000 features per physical assay, as in our newest microarray. More than 90% of the peptides on either microarray demonstrate normal distribution for binding signals. This is important since feature selection methods

used in this analysis (*t*-test and one way ANOVA) and the Naïve Bayes classifier all assume normal distribution of features.

The Naïve Bayes approach requires relatively little training data, which makes it a very good fit for the biomarker field. The sample sizes usually range from  $N = 20-100$  for the training set. Naïve Bayes has other advantages as well: it can train well on a small but high feature data set and still yield good prediction accuracy on a large test set. Any microarray with more than a few thousand probes succumbs to the issue of dimensionality. Since Naïve Bayes independently estimates each distribution instead of calculating a covariance or correlation matrix, it escapes relatively unharmed from problems of dimensionality.

The data used here for evaluating the algorithms were generated using an array with 10,000 different features, almost all of which contribute information. We have arrays with >300,000 peptides per assay (current microarrays are available from [www.peptidearraycore.com](http://www.peptidearraycore.com)) which should provide for less sharing between peptide and antibody, effectively spreading out antibodies over the peptides with more specificity. This presumably will allow resolving antibody populations with finer detail. This expansion may require a classification method that is robust to noise, irrelevant attributes and redundancy. Naïve Bayes has an outstanding edge in this regard as it is robust to noisy data since such data points are averaged out when estimating conditional probabilities. It can also handle missing values by ignoring them during model building and classification. It is highly robust to irrelevant and redundant attributes because if  $Y_i$  is irrelevant then  $P(\text{Class}|Y_i)$  becomes uniformly distributed. This is due to that fact that the class conditional probability for  $X_i$  has no significant impact on the overall computation of posterior probability. Naïve Bayes will arrive at a correct classification as long as the correct classes are even slightly more predictable than the alternative. Here, class probabilities need not be estimated very well, which corresponds to the practical reality of immunosignaturing: signals are multiplexed due to competition, affinity, and other technological limitation of spotting, background and other biochemical effects that exist between antibody and mimotope.

#### Time efficiency

As the immunosignaturing technology is increasingly used for large-scale experiments, it will result in an explosion of data. We need an algorithm that is accurate and can process enormous amounts of data with low memory overhead and fast enough for model building and evaluation. One aim for next-generation immunosignaturing microarrays is to monitor the health status of a large population on an on-going basis. The number

of selected attributes will no longer be limited in such a scenario. For risk evaluation, complex patterns must be normalized against themselves at regular intervals. This time analysis would require a conditional probabilistic argument along with the capacity of accurately predicting the risk with low computational cost. The slope of Naïve Bayes on time performance scale is extremely small, allowing it to process a large number of attributes.

## Conclusion

Immunosignaturing is a novel approach which aims to detect complex patterns of antibodies produced in acute or chronic disease. This complex pattern is obtained using random peptide microarrays where 10,000 random peptides are exposed to antibodies in sera/plasma/saliva. Antibody binding to the peptides is not one-to-one but a more complicated and multiplexed process. The quantity and appearance of this data appears numerically, distributionally, and statistically the same as gene expression microarray data, but is fundamentally quite different. The relationships between attributes and functionality of those attributes are not the same. Hence, traditional classification algorithms used in gene expression data might be suboptimal for analyzing immunosignaturing results. We investigated 17 different kinds of classification algorithm spanning Bayesian, regression, tree based approaches and meta-analysis and compared their leave-one-out cross-validated accuracy values using various numbers of features. We found that the Naïve Bayes classification algorithm outperforms the majority of the classification algorithms in classification accuracy and in time performance, which is not the case for expression microarrays [42]. We also discussed its assumptions, simplicity, and fitness for immunosignaturing data. More than most, these data provide access to the information found in antibodies. Deconvoluting this information was a barrier to using antibodies as biomarkers. Pairing immunosignaturing with Naïve Bayes classification may open up the immune system to a more systematic analysis of disease.

## Competing interests

US Patent Compound Arrays for Sample profiling: 61218890, US Patent 'Naïve Bayes Classification for Immunosignaturing M12-104L, SAJ is cofounder of HealthTell Diagnostics which owns the patent to immunosignaturing.

## Authors' contributions

MK completed the analysis of all data, and the original manuscript draft. PS completed all revisions and consulted on analysis. SAJ/PS co-invented immunosignaturing, SAJ funded the project. All authors read and approved the revised manuscript.

## Ethics statement

Consent was obtained for every sample in this manuscript and was approved by ASU IRB according to protocol number 0912004625 entitled "Profiling Human Sera for Unique Antibody Signatures". Humans were consented by the retrieving institution and a materials transfer agreement

was signed between the Biodesign Institute and the collaborating institute. The collaborating institutes' protocols were current and each human subject signed an approved consent form and released their sera.

## Acknowledgements

We are thankful to Dr. AbnerNotkins, NIH, National Institute of Dental and Craniofacial Research for providing type 1 diabetes sample, DrLucusRestrepo for providing Alzheimer's dataset. Dr Bart Legutki and Dr Rebecca Halperin for providing Antibodies dataset, University of Arizona, department of Pharmacy and Pharmacology (Serine Lau, Donata Vercelli, Marilyn Halonen) for providing Asthma samples, Peptide Microarray Core (DrZbigniewCichacz) for providing Asthma datasets, PradeepKanwar for implementing the time function in JAVA and ValentinDinu for invaluable discussion regarding algorithm selection. This work was supported by an Innovator Award from the DoD Breast Cancer Program to SAJ.

Received: 20 January 2012 Accepted: 15 May 2012

Published: 21 June 2012

## References

1. Haab BB: **Methods and applications of antibody microarrays in cancer research.** *Proteomics* 2003, **3**:2116–2122.
2. Whiteaker JR, Zhao L, Zhang HY, Feng L-C, Piening BD, Anderson L, Paulovich AG: **Antibody-based enrichment of peptides on magnetic beads for mass-spectrometry-based quantification of serum biomarkers.** *Anal Biochem* 2007, **362**:44–54.
3. Reimer U, Reineke U, Schneider-Mergener J: **Peptide arrays: from macro to micro.** *Curr Opin Biotechnol* 2002, **13**:315–320.
4. Merbl Y, Itzchak R, Vider-Shalit T, Louzoun Y, Quintana FJ, Vadai E, Eisenbach L, Cohen IR: **A systems immunology approach to the host-tumor interaction: large-scale patterns of natural autoantibodies distinguish healthy and tumor-bearing mice.** *PLoS One* 2009, **4**:e6053.
5. Braga-Neto UM, Dougherty ER: **Is cross-validation valid for small-sample microarray classification?** *Bioinformatics* 2004, **20**:374–380.
6. Hua J, Xiong Z, Lowey J, Suh E, Dougherty ER: **Optimal number of features as a function of sample size for various classification rules.** *Bioinformatics* 2004, **21**:1509–1515.
7. Sima C, Attoor S, Brag-Neto U, Lowey J, Suh E, Dougherty ER: **Impact of error estimation on feature selection.** *Pattern Recognit* 2005, **38**:2472–2482.
8. Braga-Neto U, Dougherty E: **Bolstered error estimation.** *Pattern Recognit* 2004, **37**:1267–1281.
9. Cwirla SE, Peters EA, Barrett RW, Dower WJ: **Peptides on phage: a vast library of peptides for identifying ligands.** *Proc Natl Acad Sci U S A* 1990, **87**:6378–6382.
10. Nahtman T, Jernberg A, Mahdaviavir S, Zerweck J, Schutkowski M, Mauerer M, Reilly M: **Validation of peptide epitope microarray experiments and extraction of quality data.** *J Immunol Methods* 2007, **328**:1–13.
11. Boltz KW, Gonzalez-Moa MJ, Stafford P, Johnston SA, Svarovsky SA: **Peptide microarrays for carbohydrate recognition.** *Analyst* 2009, **134**:650–652.
12. Brown J, Stafford P, Johnston S, Dinu V: **Statistical Methods for Analyzing Immunosignatures.** *BMC Bioinforma* 2011, **12**:349.
13. Halperin RF, Stafford P, Johnston SA: **Exploring antibody recognition of sequence space through random-sequence peptide microarrays.** *Mol Cell Proteomics* 2011, **10**:110–000786.
14. Legutki JB, Magee DM, Stafford P, Johnston SA: **A general method for characterization of humoral immunity induced by a vaccine or infection.** *Vaccine* 2010, **28**:4529–4537.
15. Restrepo L, Stafford P, Magee DM, Johnston SA: **Application of immunosignatures to the assessment of Alzheimer's disease.** *Ann Neurol* 2011, **70**:286–295.
16. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH: **The WEKA data mining software: an update.** *SIGKDD Explor News* 2009, **11**:10–18.
17. John GH, Langley P: **Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence.** In *Estimating Continuous Distributions in Bayesian Classifiers.* San Mateo: Morgan Kaufmann; 1995:338–345.
18. Friedman N, Geiger D, Goldszmidt M: **Bayesian Network Classifiers.** *Mach Learn* 1997, **29**:131–163.

19. Yu J, Chen X: **Bayesian neural network approaches to ovarian cancer identification from high-resolution mass spectrometry data.** *Bioinformatics* 2005, **21**(Suppl 1):i487-i494.
20. Friedman J, Hastie T, Tibshirani R: **Additive logistic regression: a statistical view of boosting.** *Ann Stat* 2000, **28**:337-407.
21. Cessie SL, Houwelingen JCV: **Ridge Estimators in Logistic Regression.** *J R Stat Soc Ser C (Appl Stat)* 1992, **41**:191-201.
22. Landwehr N, Hall M, Frank E: **Logistic Model Trees.** *Mach Learn* 2005, **59**:161-205.
23. Platt J: **Fast Training of Support Vector Machines using Sequential Minimal Optimization.** Book *Fast Training of Support Vector Machines using Sequential Minimal Optimization.* City: MIT Press; 1998.
24. Hastie T, Tibshirani R: **Classification by Pairwise Coupling.** Book *Classification by Pairwise Coupling.* City: MIT Press; 1998.
25. Keerthi SS, Shevade SK, Bhattacharyya C, Murthy KKR: **Improvements to Platt's SMO Algorithm for SVM Classifier Design.** *Neural Comput* 2001, **13**:637-649.
26. Chaudhuri BB, Bhattacharya U: **Efficient training and improved performance of multilayer perceptron in pattern classification.** *Neurocomputing* 2000, **34**:11-27.
27. Gardner MW, Dorling SR: **Artificial neural networks (the multilayer perceptron), a review of applications in the atmospheric sciences.** *Atmos Environ* 1998, **32**:2627-2636.
28. Aha DW, Kibler D, Albert MK: **Instance-based learning algorithms.** *Mach Learn* 1991, **6**:37-66.
29. Weinberger K, Blitzer J, Saul L: **Distance metric learning for large margin nearest neighbor classification.** *J Mach Learn Res* 2009, **10**:207-244.
30. Cleary J, Trigg L: **Proceedings of the 12th International Conference on Machine Learning.** In *K\*: An Instance-based Learner Using an Entropic Distance Measure.* Morgan Kaufmann; 1995:108-114.
31. Hall MA: **Correlation-based Feature Subset Selection for Machine Learning,** PhD Thesis, University of Waikato. New Zealand: Hamilton; 1998.
32. Hartigan JA: **Statistical theory in clustering.** *J Classif* 1985, **2**:63-76.
33. Quinlan JR: **Proceedings of the 5th Australian Joint Conference on Artificial Intelligence.** In *Learning with continuous classes.* World Scientific; 1992:343-348.
34. Witten IH, Eibe F, Hall MA: **Data Mining: Practical Machine Learning Tools and Techniques.** Third edition. San Francisco: Morgan Kaufmann; 2011.
35. Güvenir HA: **Voting features based classifier with feature construction and its application to predicting financial distress.** *Expert Syst Appl* 2010, **37**:1713-1718.
36. Salzberg SL: **C4.5: Programs for Machine Learning by J. Ross Quinlan.** Morgan Kaufmann Publishers, Inc., 1993. *Mach Learn* 1994, **16**:235-240.
37. Quinlan J: **Bagging, Boosting and C4.** *AAAI/IAAI* 1996, **5**:1.
38. Breiman L: **Random Forests.** *Mach Learn* 2001, **45**:5-32.
39. Hedenfalk I, Duggan D, Chen Y, Radmacher M, Bittner M, Simon R, Meltzer P, Gusterson B, Esteller M, Raffeld M, et al: **Gene-Expression Profiles in Hereditary Breast Cancer.** *New England J Med* 2001, **344**:539-548.
40. Li T, Zhang C, Ogihara M: **A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression.** *Bioinformatics* 2004, **20**:2429-2437.
41. Liu H, Li J, Wong L: **A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns.** *Genome Inform* 2002, **13**:51-60.
42. Stafford P, Brun M: **Three methods for optimization of cross-laboratory and cross-platform microarray expression data.** *Nucleic Acids Res* 2007, **35**:e72.

doi:10.1186/1471-2105-13-139

**Cite this article as:** Kukreja et al.: Comparative study of classification algorithms for immunosignaturing data. *BMC Bioinformatics* 2012 **13**:139.

**Submit your next manuscript to BioMed Central and take full advantage of:**

- Convenient online submission
- Thorough peer review
- No space constraints or color figure charges
- Immediate publication on acceptance
- Inclusion in PubMed, CAS, Scopus and Google Scholar
- Research which is freely available for redistribution

Submit your manuscript at  
www.biomedcentral.com/submit

