Explanation and Prediction of nsSNP-Induced Pathology Using Association Mining, Transduction, and Active Learning

Nada Basit and Harry Wechsler

Department of Computer Science
George Mason University
4400 University Drive, Fairfax, Virginia 22030, USA
{nbasit, wechsler}@gmu.edu

Copyright © 2013 Nada Basit and Harry Wechsler. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

This paper is about robust and effective explanation and prediction of nsSNP-induced pathology. Towards that end, we propose a novel hybrid method that combines transduction (T)(using both labeled and unlabeled samples), association mining (AM) using either A priori or Ripper algorithms, and active learning (AL). The proposed method, called T-AM-AL, which also addresses the imbalance class problem using random over-sampling and stratification (for the purpose of double cross-validation), yields similar accuracy performance but much better positive predictive value (PPV) compared to state-of-the art methods. This is achieved using much less annotation for training (our method employs 53% training data for learning compared to more than 90% training data employed by competing state-of-the art methods trained using random forests or SVM). An additional comparative advantage from the use of association rules (compared to random forest or SVM) is the explanatory aspect (including confidence metrics), which can be useful for drug design and synthesis. The active learning component of our method serves to integrate explanation and prediction, and also helps in reducing the amount of annotation required.

Keywords: Active Learning, Association Rules, Transduction

1 Introduction

The most common reason for variations in DNA sequence is traced to Single-Nucleotide Polymorphisms (SNPs). There is much interest in determining whether such variations, leading to a single amino acid (or residue) replacement in the translated protein, are neutral or disease-related. SNPs occurring in coding regions may result in single amino acid polymorphisms (SAPs) that may affect protein function and lead to disease (pathology). This paper innovates as it proposes and validates concurrent explanations and predictions regarding SNP pathologies.

Statistical learning theory [26] discusses at length model selection and prediction in terms of complexity and generalization ability. Testing hypotheses is usually approached in terms of explanation or prediction [12]. This paper advances a hybrid approach where both explanation and prediction are made for reasons such as controlling events that affect outcomes (“explanatory purposes”) and classification (“prediction purposes”). The goal here is to (data) mine for association rules, i.e., explanations, which determine the possibility of SNP-induced pathology, coupled to predictions about SNP-induced pathology using transduction and active learning. The novel and hybrid approach, applied to the specific problem of determining SNP-induced pathology, is carried out using *in silico* (“virtual”) computational mutagenesis.

This paper expands on our previous work related to predicting enzyme mutant activity using computational mutagenesis and incremental transduction [7]. Here it involves using SNPs instead of amino acids, and mining for explanatory associations in addition to using transduction for prediction purposes. The expected outcomes include among others, less initial annotation, training on both labeled and unlabeled data samples, and overall generalization at a reduced cost and effort using active learning. The data mining association techniques considered here are the A priori and Ripper algorithms. Feasibility and effectiveness of the hybrid approach is duly made using proper performance evaluation, e.g., cross-validation, and performance metrics, (see Sect. 6). Comparison is then made between using a base classifier (A priori or Ripper) in a “one-shot” learning scheme and using, in an iterative fashion, a base classifier embedded in incremental transduction (with active learning) driven by the support and confidence measures provided by the association rules. Comparison is also made between our proposed method and alternative state-of-the art learning methods, e.g., Random Forest (RF) and Support Vector Machines (SVM), which are shown to yield similar performance [17] but lack in explanatory power and require additional annotation.

The applications of explanation and prediction of SNP-induced pathology are

broad and include drug design, protein engineering, and preventative medicine. The major benefit of conducting mutagenesis *in silico* is to minimize the cost associated with wet lab experiments by limiting experiments only to mutants of interest.

The outline of the paper is as follows. Sect. 2 gives a brief biological background on SNPs. Sect. 3 describes the SNP data and the features used for their representation. Sect. 4 discusses model selection and prediction, and specific learning methods used for comparative performance evaluation. Sect. 4 also discusses the one-shot and incremental (data stream) training strategies. Sect. 5 describes our novel hybrid approach in terms of transduction, association mining, and active learning. Performance evaluation (including cross-validation) and metrics are presented in Sect. 6, while experimental design and comparative results are presented in Sect. 7. Sect. 8 describes the explanatory aspect and its relation to drug design and protein engineering, while Sect. 9 concludes the paper.

2 SNP Background

DNA sequence variations which occur when a single nucleotide is altered are called Single-Nucleotide Polymorphisms (SNPs). Similar to SNPs, Single Amino acid Polymorphisms, or SAPs, are changes in the amino acid sequence where the original amino acid at a given location is replaced by another amino acid. Protein sequence, structure, and function are closely related. Mutagenesis is the process by which a deliberate change in the genetic information (mutation) is made to create a mutant. An affected protein function may lead to disease (pathology). Computational mutagenesis carries out mutagenesis *in silico*. This virtual method is time and cost effective.

For a variation to be considered an SNP, it must occur in at least 1% of the population [15]. SNPs can occur in coding (gene) and noncoding regions of the genome. Because only about 3 to 5 percent of a person's DNA sequence codes for the production of proteins, most SNPs are found outside of coding regions [21]. There is particularly much interest in SNPs found within a coding region because they have an increased likelihood to alter the biological function of a protein. SNPs within a coding region do not necessarily change the amino acid sequence of the protein that is produced. Therefore, non-synonymous polymorphisms (nsSNPs) is the subject of interest here as they can be the leading cause of disease related pathology. Stenson et al. [22] have reported that more than half of all known mutations driven diseases come from nsSNPs.

3 Representation

This section describes the protein (feature) representation using computational geometry (Sect. 3.1), and the nsSNP data set (Sect. 3.2).

3.1. Protein representation using computational geometry

As mentioned in Sect. 2, nsSNPs result in a change in the amino acid sequence of the mutated protein. Given a protein and its amino acid sequence, one can represent it using methods drawn from computational geometry. Towards that end one considers each amino acid as a single point in 3D space using numerical coordinates (obtained from the Protein Data Bank (PDB)), with the whole protein then represented by a 3D graph where the nodes are the amino acids and the edges connect to the nearest amino acids. Once a protein is represented numerically/graphically one extracts features, which later on serve for classification of unlabeled mutants.

For the purpose of further processing, amino acids are abstracted in terms of their alpha carbon atomic coordinates. Each protein, that is, an amino acid sequence, is thus a sequence of corresponding alpha carbons (“C-alpha trace” or “backbone”). The sequence is subject to Delaunay tessellation, which yields Delaunay simplices in 3D and establishes nearest-neighborhood relationships between the amino acids making up the protein.

Delaunay tessellation of each protein structure yields an aggregate of non-overlapping, space-filling, irregular tetrahedral simplices (referred to as Delaunay simplices) whose vertices are the amino acid point representations. [16, 24] An amino acid vertex is simultaneously shared by multiple Delaunay simplices within a protein tessellation. The Quickhull algorithm [4] performs the Delaunay tessellations. Each Delaunay simplex in a protein structure Delaunay tessellation objectively defines the four nearest-neighbor amino acids for each given amino acid (which represents a fundamental topological property of 3D space). This is why Delaunay simplices are also known as quadruplets. For added assurance of biochemically feasible quadruplet interactions, Delaunay simplices were removed from every tessellation if at least one of the six edges has length greater than 12 Å. [17]

3.2. NsSNP data

A data sample represents an nsSNP with information on the mutation and contextual structural information surrounding the mutated position (see features below), and finally the resulting pathology of this mutation (the class/label). The data set consists of 1,790 mutants coming from 243 tessellated human protein structures corresponding to nsSNPs obtained from the Swiss-Prot database that are functionally categorized as either associated with a particular disease (belonging to the “disease-associated” class) or not (belonging to the “neutral” class). In particular, the data set consists of 458 neutral nsSNPs mapping to 184 protein structures and 1,332 disease-associated nsSNPs mapping to 102 protein structures. [2] Each nsSNP sample is selected here if it appears in both the Swiss-Prot database and the Protein Data Bank to ensure that both its class and 3D structure are available. In addition the mutation was selected if the position

undergoing mutation had at least six tessellation-based nearest neighbors. [2, 17] This data has been previously used [17], however, in this research the number of features used has been reduced (see below for a list of the features used) because deleted features, e.g., residual and environmental scores, were determined empirically to lack discriminative power.

The features that describe each nsSNP data sample are listed below:

- (1) Wild type nucleotide (letter), site number (position) of mutated (replacement) nucleotide, replacement nucleotide type (letter), e.g., T14A: wild type = T, position = 14, and replacement = A
- (2) Location (depth) of mutated position: surface (S), buried (B), undersurface (U)
- (3) The secondary structure that the mutated position is a part of: alpha helix (H), beta strand (R), coil (C), and turn (T)
- (4) Structure environment information – surrounding the point of mutation in 3D space
 - (a) Amino acid identities at the six nearest neighbors,
 - (b) Differences between the primary sequence amino acid positions of the nearest neighbors and the mutated position,
 - (c) Number of edge contacts that the mutated position has with surface positions.

The amino acid identities at the six nearest neighbors are determined by the six nearest-neighbor positions (in 3D) that participate in simplices with the mutated position. A count is obtained for the number of edge contacts that the mutated position has with surface positions (derived from tessellating the 3D structure) [2, 5]; buried mutated positions have a count of zero by definition.

4 Model Selection, Explanation and Prediction, Learning Methods, and Training Strategies

Model selection is fundamental to scientific inquiry. A good model balances goodness of fit with simplicity for the purpose of robust prediction. That is, model selection seeks a model of the right complexity that is neither susceptible to overfitting (the model is too complex and thus undermines generalization) nor underfitting (the model is too simple to explain the training data). [23, 26, 27] Measuring the performance of the model on the test set (previously unseen data samples) provides an unbiased estimate of its generalization error. Comparing the relative performances of different classifiers on the same domain involves ranking their computed accuracies (or error rates) on the test set. Towards that end the class labels of the sequestered test data samples must be available, i.e., ground-truth is known. One way to evaluate the performance of a classifier and tune its performance is by using cross-validation (see Sect. 6).

The goal here is not only to find another method that helps to annotate nsSNPs regarding pathology, but should also explain / link inputs and outcomes. The approach used to determine nsSNP pathology seeks to find possible associations rules between features and diagnoses. Towards that end, association rule mining is coupled with transduction. Transduction, and active learning are discussed in Sect. 5 along with the description of our novel hybrid method. Association rule mining and the A priori [1] and Ripper [10] algorithms used are discussed here. The other learning methods that are used for comparison are Random Forest (RF) [20] and Support Vector Machine (SVM) [26].

A data set with categorical “item” features is known as a transactional data set, where each data sample is a “transaction.” Rules ($X \rightarrow Y$) can be derived from the transactional data set. The indicators “Support” (sup) and “Confidence” (conf) report on the strength of a given rule. Support and Confidence are formally defined as follows. Support is the percentage of the transactions that contain $X \cup Y$. That is, $\text{sup} = P(X \cup Y)$, where both items X and Y appear together in the same transaction. Confidence is the percentage of transactions that contain X also contain Y , that is, $\text{conf} = P(Y | X)$.

Normal association rule mining doesn’t have any target. It finds all possible rules that exist in data, i.e., any item can appear as a consequent or a condition of a rule. However, in some applications, one is interested in specific “consequent” targets. The SNP endeavor falls into such a category where features representing a single-nucleotide polymorphism (“annotation”) are linked (“associated”) with pathology (class label). The task here is thus to mine for (class) association rules. Class association rules are of the form ($X \rightarrow Y$) where X is one or several items/features (item set) from the transactional data set and Y is a class label. A class association rule is generated only if the set X is greater than or equal to the minimum support and confidence indicators specified. An example rule of this form is given below:

{location=buried; secondary structure=coil $\rightarrow$ class=disease-associated}
[sup=25%; conf=88%]

The methods used to learn association rules are, A priori [1] and Ripper [10].

Our hybrid method employs an incremental training strategy, characteristic of transduction, which is in contrast to a “one-shot” training strategy, characteristic of one round of cross validation. One-shot learning utilizes all labeled data (no data stream) and trains a classifier on one portion of the data set (training set) then tests on the remainder (test set). This strategy is called “one-shot” because the classifier is trained with all the data in the training set once and then tested (labeled) on the remaining test set. The incremental training strategy takes into account not only the (labeled) training set but also the (unlabeled) data that one wishes to classify. [28] The incremental strategy doesn’t label the test set all at once, rather it incorporates unlabeled (test) data samples in the classification process a little at a time. This creates a data stream whereby the training set is augmented after each iteration and each time the classifier is trained on the new training set. One method to choose which data samples augment the training set

is by using active learning (see Sect. 5). Our proposed hybrid method, that employs this incremental learning training strategy, is compared against one-shot methods to show the benefits of the former.

5 Transduction, Association Mining, and Active Learning (T-AM-AL)

A discussion of transduction and active learning, components of our hybrid method, are discussed here. The discussion of association mining has been discussed in Sect. 4. A description of our hybrid method will follow.

Transduction is local inference (“estimation”) that moves from particular(s) to particular(s). [9, 29] In contrast to inductive inference, one now directly infers (using transduction) the values of the function only at the points of interest from the training data. [25, 26] Inference takes place using both labeled and unlabeled data, which are complementary to each other for the purpose of prediction. Transduction incorporates unlabeled data, characteristic of test (“unlabeled”) samples, in the classification process responsible to label them for the purpose of prediction. It further seeks for a consistent and stable labeling across both (near-by) training (“labeled”) and test (“unlabeled”) data. [13] Transduction seeks here to authenticate mutations whose pathology (i.e. disease-related or neutral) is most consistent with the given pathologies driven by known and similar protein nsSNPs. The search for putative labels (for unlabeled samples) seeks to make the labels for both training and test data compatible or equivalently to make the training and test error consistent.

Active learning helps with the effective and efficient use of computational resources to enhance learning and yield overall better performance. Towards that end, active learning determines which data to acquire next for the purpose of learning by iteratively seeking out the most informative new data samples. The selective aspect of active learning can be one of many criteria; it is realized here using “maximum curiosity,” a criteria that selects those examples that maximize the cross-validated classifier accuracy under either assumed mutant activity. [6, 11]

Our proposed hybrid method *T-AM-AL*, employs transduction (T), association mining (AM) and active learning (AL). The motivation is to combine explanation and prediction, and to provide for robust prediction and effective training using less annotation. Regarding operation, let P be the labeled training set and let Q be the (unlabeled) test set. P is further divided into two sets, a learning set L and a validation set V . Ground truth for the test set and the validation set are known but withheld and used here for evaluation purposes only. The A priori and Ripper association methods serve as the base classifier C for experiments using transduction. Note that A priori and Ripper are not used in conjunction but rather as individual components for comparison purposes. The AM methods associate reliability indicators with the output rules generated.

Once the stopping criterion has been reached the “classifier” is run on the training set L and high support and confidence rules are reported. Prediction accuracies are then reported by applying these rules to the test set Q . T-AM-AL works as follows:

- (1) Train the classifier C on L
- (2) Amongst the rules generated allow only those with high support and confidence to annotate the unlabeled data in V . This requires minimum support of 10% and a minimum confidence of 70% (for both A priori and Ripper), thresholds which were empirically determined. Examples that get annotated with these rules are selected and fed for further learning via AL
- (3) $L = L \cup \text{labeled}(V)$. The learning set L is augmented with validation samples from V whose labels are found with high support and confidence. Note that newly labeled samples that augment the learning set are deleted from the validation set
- (4) Iterate until no further validation samples can be classified (labeled) with high support and confidence or until the maximum number of iterations is reached. During each iteration, a new (augmented) learning set L is accessed to train the AM classifier.

Note that AL using maximum curiosity is used to select only those (strong) association mining rules which have a minimum support of 10% and a minimum confidence of 70%, thereby preventing weaker rules from annotating the unlabeled examples (see step (2)).

6 Performance Evaluation and Metrics

Cross-validation is an enhanced version of the holdout method for performance evaluation and it is used here. The data set is divided into k disjoint partitions (folds), with one fold used for testing while the other folds are used for training (the training folds themselves are divided into a learning fold and the other folds are used for tuning / validation during double cross-validation). This is repeated k times so that each fold is used for testing. The training folds make up the training set. Overall performance is derived by averaging performance metrics computed over the k splits.[27]

The learning fold is used to train the base classifier / learner (A priori or Ripper) in order to learn association rules while tuning takes place over the validation folds. The final association rules learned are run on the (sequestered) test fold partition and results are tabulated.

The metrics used to evaluate the results are as follows. Assuming that TP (TN) stand for the total number of correctly predicted disease-associated (neutral) mutants, and that FP (FN) stand for the total number of misclassified disease-associated (neutral) mutants, the overall (percentage) accuracy %Acc of a

given method is calculated as $\%Acc = (TP + TN) / (TP + FN + TN + FP)$. The following metrics are also computed given their robustness with respect to unbalanced class distributions [17, 27] as well as for comparisons against other methods. The balanced error rate (BER) = $0.5 \times [FN/(FN + TP) + FP/(FP + TN)]$, the Matthew's correlation coefficient (MCC) = $(TP \times TN - FP \times FN) / [(TP + FN)(TP + FP)(TN + FN)(TN + FP)]^{1/2}$ and the positive predictive value (PPV) = $TP / (TP + FP)$, which is similar to precision. The role of PPV is more important (than NPV) since we wish to learn about what causes disease rather than what corresponds to a neutral condition. As 5-fold cross-validation is performed the value for each of the metrics listed is the average over five iterations.

7 Experimental Results

In this section the software and data sets used are discussed briefly in Sect. 7.1. We summarize the methods used for comparison in Sect. 7.2. The experimental design is discussed in Sect. 7.3, with the results of our method and its variants reported in Sect. 7.4. Sect. 7.5 shows and discusses comparative results between our best methods and other competitive methods. We also compare the methods in terms of the amount of annotation (training / learning) used.

7.1. Software & data sets

The A priori and Ripper implementation are provided by WEKA [14]. The required format of the data for each algorithm is different. A priori requires data to be text (no numbers) whereas Ripper requires data to be numeric. The transductive experiments employ MATLAB [18] and utilize the association results made available by WEKA. The data set consists of 1,790 human nsSNPs that are functionally categorized as belonging to one of two classes – disease-associated or neutral.

7.2. Comparative methods

The methods used for evaluation and comparison are by Capriotti et al. [8], Bao and Cui [3], and Masso and Vaisman [5, 17]. The method by Masso and Vailsman is the one we expanded on to include knowledge (explanation) by means of association rule mining.

7.3. Experimental design

The first set of experiments employs 5-fold cross-validation with four folds used for training (using A priori or Ripper for mining associations) and the remaining fold used for testing, with folds denoted as A, B, C, D, and E. This corresponds

to one-shot learning using association mining. The second set of experiments, which integrates association mining (AM) (using A priori or Ripper as base learners) and transduction, employs also 5-fold cross-validation. Due to an imbalance in the class distribution, the experimental design used ensures equal class representation for the cross-validation and a balanced data set by means of stratified sampling and random over-sampling. Now double cross-validation is used where the folds in the training set are further segregated into Learning and Validate sets. The Learning set starts at 20% of the entire data set, with the remaining training data assigned to the Validate set (assumed to be unlabeled despite its ground truth being known). The Learning set trains as usual the base learner (A priori or Ripper) to learn association rules and tests on the Validate set. Confidently labeled samples in the Validate set augment the Learning set. This process iterates, with the (labeled) Learning set growing, while revised association rules are learned and relearned. Process stops when no new rules are generated and/or no additional (high support, high confidence) labeled samples from the Validate set are found. The final association rules (once process stops) are run on the sequestered test data partition / fold and the results are tabulated.

7.4. Results

One-shot learning corresponds to associative mining (AM), while the hybrid method, T-AM-AL, combines association mining (AM) (employing either A priori or Ripper) with Transduction (T) and its implicit use of active learning (AL).

A number of methods are carried out and the results are summarized in the following table. We first carry out one-shot association mining and these methods are called “One-shot A priori” and “One-shot Ripper.” We continue with our hybrid method (T-AM-AL) that uses transduction to replace the one-shot learning paradigm as well as the use of active learning. These methods are called “T-A priori-AL” and “T-Ripper-AL.” For an additional variation we paired association mining with transduction but omitted active learning. These methods are called “T-A priori” and “T-Ripper.” Table 1 summarizes the results of all methods presented so far and includes the average percentage accuracy (%Acc) and its standard deviation (St. Dev.) as well as the performance metrics BER, MCC, PPV, and the percentage of the data used for training (%Use).

Table 1. Summary of Our Methods

Method	%Acc	St. Dev.	BER	MCC	PPV	%Use
One-shot A priori	0.72	1.21	0.43	0.28	0.70	80
One-shot Ripper	0.73	0.99	0.40	0.30	0.72	80
T-A priori	0.73	1.16	0.38	0.31	0.72	80
T-Ripper	0.73	1.10	0.38	0.32	0.74	80
T-A priori-AL	0.73	0.96	0.37	0.31	0.75	53
T-Ripper-AL	0.73	1.44	0.36	0.33	0.77	53

One can observe from Table 1 that our proposed hybrid method, T-AM-AL, where AM=Ripper, performed the best among all the variants, particularly in comparison to the one-shot methods. This improvement can be seen relative to the performance metrics used. A marked improvement is also the reduction in the amount of training data that is used once active learning is activated (80% annotation goes all the way down to 53%).

The realization that the class imbalance for our data set can impact the performance of our classifiers and methods we implemented stratified sampling for the purpose of constructing data folds that are identical in their class distribution for our (double) cross-validation implementation (see Sect. 6). We run all the variants of our method again, this time implementing also stratified sampling. We further expand on our method by carrying out random over-sampling to balance the proportions of the classes in the data set. We once again we repeat all the variants of our method (one-shot, transduction minus active learning, and transduction with active learning) this time implementing stratified sampling as well as random over-sampling to balance the data set.

Table 2. Comparison of performance and metrics of stratified and balanced methods

Method	Strat	Bal	%Acc	St. Dev.	BER	MCC	PPV	%Use
One-shot A priori	Y		0.73	1.20	0.42	0.29	0.73	80
One-shot Ripper	Y		0.73	0.98	0.39	0.30	0.73	80
T-A priori	Y		0.72	1.03	0.35	0.32	0.77	80
T-Ripper	Y		0.74	0.91	0.36	0.34	0.79	80
T-A priori-AL	Y		0.75	0.69	0.32	0.36	0.83	53
T-Ripper-AL	Y		0.76	0.65	0.31	0.38	0.85	53
One-shot A priori	Y	Y	0.73	1.14	0.41	0.29	0.74	80
One-shot Ripper	Y	Y	0.73	0.88	0.38	0.30	0.74	80
T-A priori	Y	Y	0.72	0.93	0.35	0.33	0.78	80
T-Ripper	Y	Y	0.74	0.77	0.36	0.34	0.81	80
T-A priori-AL	Y	Y	0.75	0.66	0.31	0.36	0.85	53
T-Ripper-AL	Y	Y	0.76	0.60	0.31	0.38	0.87	53

Table 2 summarizes the results of stratified (Strat) and balanced methods (Bal). Results representing stratified methods will have a “Y” (representing “Yes”) in the Strat column, and results representing both stratified and balanced methods will have a “Y” in both the Strat and Bal columns. An empty entry in the Strat and Bal columns indicate the corresponding implementations were not used. Performance is evaluated by prediction accuracy and its standard deviation (St. Dev.) as well as the performance metrics BER, MCC, PPV, and %Use.

The method that performed the best (see Table 1 and Table 2) is the hybrid method using transduction with active learning (where AM=Ripper as the base classifier) and has access to stratified and random over-sampling for balancing

purposes. This is the specific method we will, from now on, refer to as our best method and call it T-AM-AL for comparative purposes.

T-AM-AL not only performs the best in terms of percentage accuracy but also in the other performance metrics as well. It has the lowest BER score, the highest MCC score, and highest PPV (precision) score. Additionally, T-AM-AL uses the least amount of training data (53%) enabling it to tune the classifier faster (much less annotation) therefore enabling efficient use of computational resources, which also makes it competitive when compared to other methods.

7.5. Comparison of T-AM-AL with other learning methods

We compare now our hybrid method, T-AM-AL (where AM=Ripper and our method is both stratified and balanced), against three studies reported by other research groups, two of whom utilized only Swiss-Prot annotated mutants for training, and the third, as in our study (and to which our study is a direct expansion on), who utilized both Swiss-Prot and PDB databases (to ensure that both its class and 3D structure are available). These studies explicitly reported values for %Acc, BER, and MCC, thereby allowing for comparison with our approach. While we utilize an additional performance metric (PPV) to evaluate our approach, %Acc, BER, and MCC is the largest set of metrics shared by all studies [27]. A comparative assessment of all these methods is shown in Table 3, where we also report the size of the data used (Size of DB) as well as the percentage of the data used for training (%Use) so that an easy comparison of these factors can be made. Our hybrid method, T-AM-AL, is shown in the last row of Table 3.

Our results are similar to Masso and Vaisman [2, 17] who report results on the same task of determining nsSNP-induced pathology and using the same data set. We obtain similar performance but use much less annotation (53% compared to 95%). While our method is comparable to these methods in terms of percent accuracies and metrics, it has some comparative advantages.

Table 3. Comparison of performance and metrics with other methods

Method	Size of DB	%Acc	BER	MCC	%Use
Masso and Vaisman, RF	1790	0.76	0.29	0.40	95
Capriotti et al., SVM	21,000+	0.70	0.35	0.34	95
Capriotti et al., SVM-hybrid	21,000+	0.74	0.27	0.46	95
Bao and Cui, RF	4013	0.77	0.29	0.32	90
Bao and Cui, SVM	4013	0.76	0.32	0.27	90
T-AM-AL	1790	0.76	0.31	0.38	53

In particular, T-AM-AL has (1) the combined benefits of using less data for training (less annotation), and is thus able to consider a larger body of examples for possible training; (2) employs less features (as mentioned in Sect. 3.2, the number of features we use are a subset of those used by Masso and Vaisman); and

(3) the explanatory aspect of using association rules (see also Sect. 8).

8 Explanatory Dimension for T-AM-AL

The ability of our method, T-AM-AL, to provide explanations for the causes leading to pathology due to nsSNPs is important on its own. It allows for better understanding of the circumstances (the state of the features) that result in the protein becoming disease-associated. Association mining provides a simple and straightforward style of explanation (in the form of rules) that describes the links between the features and the outcome (pathology). The major benefit of conducting mutagenesis *in silico* is being able to accumulate knowledge and understanding of nsSNPs at a fraction of the time and cost.

A few significant rules discovered by T-AM-AL pertaining to the disease-associated class follow. Listed below are some of the interesting item sets found suggesting nsSNP induced pathology (such results have also been confirmed in literature [17, 19]):

{location = buried; secondary structure=coil}	[sup = 25%; conf = 88%],
{location=buried; #edges=low; secondary structure=coil}	[sup=25%;conf=88%],
{location = buried; secondary structure=helix}	[sup = 36%; conf = 76%],
{location=buried; #edges=low}	[sup = 77%; conf = 79%].

The explanatory aspect of using association rules, an important advantage of our method, is briefly described next and is representative for all the experiments conducted. Analyzing the samples that augment the training set in our method shows that the following features make a significant contribution. Over 80% of the correct predictions made are associated with the feature {location=buried}, and belong to the disease-associated class. They also had confidence values above 85% with only a few between 70-80%. Item sets including {location=buried} also significantly contribute to achieving similar classification. No rules were generated with the features {#edges=medium} or {#edges=high}, suggesting buried mutations have a greater impact on pathology than ones closer to the surface. Mutations at buried positions generally reflecting adverse protein functional effects, have also been confirmed in literature [17, 19].

An nsSNP feature present only in the rules generated using transduction (with active learning) but not for one-shot AM learning, was the amino acid nearest neighbors (aannX, where X is the position) (see feature 4a in Sect. 3.2). The motivation for this observation is better performance for transduction compared to one-shot AM learning. These specific results (involving the amino acids glycine (“G”) and leucine (“L”)) are not yet confirmed in literature but are examples of candidate classification rules for further study including wet lab experimentation. An example of such a rule is:

{aann1=G; #edges=low → class=disease-associated} [sup = 10%; conf = 93%].

Our data set representation employs a reduced set of features but is flexible

enough to accommodate additional features of interest. Towards that end, one can expand on our original data representation (to include data features of interest) and run the same algorithms as we did. This would enable different studies geared to combine computing, e.g., computational mutagenesis, and drug design and synthesis related to nsSNP induced pathology.

9 Conclusions

This paper presented a novel hybrid method, T-AM-AL, which combines association mining (using either A priori or Ripper) with transduction and active learning for the purpose of determining nsSNP-induced pathology. One of the novelties of our method comes from combining explanation and prediction through selectivity when choosing candidates for classification rules bearing on nsSNP induced pathology. Our method T-AM-AL (using Ripper for association mining) was found to be most effective compared against other competing learning methods vis-à-vis the amount of annotation / training required to learn the classification rules (less annotation), and is thus able to consider a larger body of examples for possible training. It is compact in data representation as it employs fewer features, and compared to abstract classifiers such as RF and SVM it explains the biomolecular “reasons” behind the association rules learned for the purpose of classification. Venues for future research include considering specific diseases (“multi-class”) rather than binary labels in order to determine specific associations rules or patterns relevant to nsSNP induced pathology, different choices for active learning including “Bayesian Surprise,” [11] and investigating the effects of sets of nsSNPs (rather than one) within an organism and their effects on pathology.

References

- [1] R. Agrawal and R. Srikant, Fast Algorithms for Mining Association Rules in Large Databases, in Proceedings of the 20th International Conference on Very Large Data Bases, Santiago, Chile, (1994), 487–499.
- [2] AUTO-MUTE: Human nsSNPs [Online], Available: proteins.gmu.edu/automute/AUTO-MUTE_nsSNPs.html.
- [3] L. Bao and Y. Cui, Prediction of the phenotypic effects of non-synonymous single nucleotide polymorphisms using structural and evolutionary information, *Bioinformatics*, 21(10) (2005), 2185–2190.
- [4] C. Barber, D. Dobkin, and H. Huhdanpaa, The Quickhull algorithm for convex hulls, *ACM Transactions on Mathematical Software*, 22(4) (1996), 469–483.
- [5] M. Barenboim, M. Masso, I. Vaisman, and D. Jamison, Statistical geometry based prediction of nonsynonymous SNP functional effects using random forest and neuro-fuzzy classifiers, *Proteins*, 71(4) (2008), 1930–1939.

- [6] N. Basit, and H. Wechsler, Function prediction for in silico protein mutagenesis using transduction and active learning, in IEEE International Conference on Bioinformatics and Biomedicine, Atlanta, Georgia (2011).
- [7] N. Basit, and H. Wechsler, Prediction of Enzyme Mutant Activity Using Computational Mutagenesis and Incremental Transduction, *Advances in Bioinformatics*, Article ID 958129, (2011).
- [8] E. Capriotti, R. Calabrese, and R. Casadio, Predicting the insurgence of human genetic diseases associated to single point protein mutations with support vector machines and evolutionary information, *Bioinformatics*, 22 (2006), 2729–2734.
- [9] V. Cherkassky, and F. Mulier, *Learning from Data: Concepts, Theory, and Methods*, John Wiley & Sons (2007).
- [10] W. Cohen, Fast Effective Rule Induction, in *Proceedings of the Twelfth International Conference on Machine Learning*, (1995), 115–123.
- [11] S. Danziger, J. Zeng, Y. Wang, R. Brachmann, and R. Lathrop, Choosing Where to Look Next in a Mutation Sequence Space: Active Learning of Informative P53 Cancer Rescue Mutants, *Bioinformatics*, 23(13) (2007), 104–114.
- [12] R. Frazier, *Explanation, control and generality*, (2009), [Online], Available: kant1.chch.ox.ac.uk/rlfrazier/EPC/.
- [13] A. Gammerman, V. Vovk, and V. Vapnik, Learning by Transduction, in *In Uncertainty in Artificial Intelligence*, (1998), 148–155.
- [14] M. Hal, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten, The WEKA Data Mining Software: An Update, *SIGKDD Explorations*, 11(1) (2009).
- [15] Human Genome Project (HGP) [Online], Available: genomics.energy.gov/.
- [16] M. Masso, and I. Vaisman, Accurate Prediction of Enzyme Mutant Activity Based on a Multibody Statistical Potential, *Bioinformatics*, 23 (2007), 3155–161.
- [17] M. Masso, and I. Vaisman, Knowledge-based computational mutagenesis for predicting the disease potential of human non-synonymous single nucleotide polymorphisms, *J. Theor. Biol.*, 266(4) (2010), 560–568.
- [18] MATLAB version 6.5.0/7.10.0. Available: mathworks.com.
- [19] H. Rangwala, and G. Karypis (eds.), *Introduction to Protein Structure Prediction: Methods and Algorithms*, John Wiley & Sons, (2010).
- [20] S. Russell, and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed, Prentice Hall, New Jersey (2009).
- [21] SNPs: Variations on a Theme, National Center for Biotechnology Information, a division of the National Library of Medicine (NLM) at the National Institutes of Health (NIH) [Online], Available: ncbi.nlm.nih.gov/About/primer/snps.html.
- [22] P. Stenson, M. Mort, E. Ball, K. Howells, A. Phillips, N. Thomas, and D. Cooper, The Human Gene Mutation Database: 2008 update, *Genome Med*, 1 (2009), 13.

- [23] P. Tan, M. Steinbach, and V. Kumar, Introduction to Data Mining, 1st ed, Addison Wesley, (2005).
- [24] I. Vaisman, A. Tropsha, and W. Zheng, Compositional Preferences in Quadruplets of Nearest Neighbor Residues in Protein Structures: Statistical Geometry Analysis, in Proc. of the IEEE Symp. on Intel. & Syst., (1998), 163–68.
- [25] V. Vapnik, Estimation of Dependences Based on Empirical Data, 1st ed, Springer, New York (1982).
- [26] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York (1998).
- [27] M. Vihinen, How to evaluate performance of prediction methods? Measures and their interpretation in variation effect analysis, *BMC Genomics*, 13 (2012).
- [28] J. Weston, F. Pérez-Cruz, O. Bousquet, O. Chapelle, A. Elisseeff, and B. Schölkopf, Feature selection and transduction for prediction of molecular bioactivity for drug design, *Bioinformatics*, 19(6) (2003), 764–771.
- [29] X. Zhu, *Semi-supervised learning literature survey*, Department of Computer Sciences, University of Wisconsin, Madison, (2005).

Received: February 7, 2013