Combining diverse classifiers using precision index functions

Jose Bird* and Daijin Ko

Department of Management and Statistics,
The University of Texas at San Antonio,
One UTSA Circle, San Antonio, Texas 78249, USA
E-mail: jose.bird@utsa.edu
E-mail: daijin.ko@utsa.edu
*Corresponding author

Abstract: This paper introduces a combining classifier using two proposed precision indexes: precision index (PIN) and class specific precision index (PIC). Comparison of combining methods typically fails to consider high precision performance. This new combining method generates predictions with higher precision and recall than other methods. The proposed method is especially useful for efficient screening of predictions where actual verification is time consuming and costly. The performance of the proposed method is compared to majority voting, stacking, and cluster-selection for two well-known datasets:

1. vowel recognition (Hastie et al., 2009)
2. yeast protein localisation (Frank and Asuncion, 2010).

The precisions obtained exceeded results previously reported for protein localisation data (Horton and Nakai, 1997; Chen, 2010) and for vowel recognition data (Hastie et al., 2009). A weighted precision index using PIC and PIN indexes outperformed all combining methods at higher precisions.

Keywords: precision index; PIN; class-specific precision index; PIC; combined classifiers; precision; recall; pattern recognition.


Biographical notes: Jose Bird received his PhD in Applied Statistics at the University of Texas at San Antonio. He works at Valero Energy as Director Energy System and Metrics and is Lecturer at the Department of Management Science and Statistics, University of Texas at San Antonio. His research interests include application of data mining methods to pattern recognition problems in engineering and bioinformatics fields.

Daijin Ko received his PhD in Statistics from the University of Washington, Seattle, WA. He is a Professor at the Department of Management Science and Statistics, University of Texas at San Antonio. His research interests include applied machine learning, bioinformatics, and pattern recognition in information security management, and neurostatistics.
1 Introduction

The field of supervised statistical learning has experienced significant growth over the last 30 years as new and improved classification techniques have been developed. Some widely used supervised statistical learning methods applied to classification problems include: classification and regression trees (CART), random forest (RF) method, support vector machines (SVM), multivariate adaptive regression splines (MARS), K-nearest neighbours (KNN), linear discriminant analysis (LDA), logistic regression, and neural networks (NNs). Each of these methods has its own set of strengths and weaknesses when applied to classification problems (Hastie et al., 2009), where the discrete outcome class of an observation is predicted given a set of input variables known as the feature or domain space.

Recently much attention has been given to the development of combining methods that aggregate the results of multiple classifiers. It has been shown that combining classifiers can improve prediction accuracy beyond that which would be obtained with any single classifier (Lam and Krzyzak, 1995; Breiman, 1996; Sollich and Krogh, 1996; Lam and Suen, 1996; Clarke, 2004; Kuncheva, 2004; Witten and Eibe, 2005; Monteith et al., 2011). Combination techniques have been divided into two main categories:

1 classifier fusion
2 classifier selection (Kuncheva, 2004).

Classifier fusion techniques assume each individual classifier has knowledge of the whole feature space and use combining functions such as majority voting and average. Classifier selection techniques, on the other hand, assume that individual classifiers are experts on different regions of the feature space and the classifier with the highest estimated prediction accuracy for a given region is selected. Examples of combination techniques following the fusion strategy include majority vote, weighted majority vote, Naive Bayes (NB) combination, and multinomial methods. Examples of techniques following the selection strategy include static cluster-selection and the PIN combining method.

An example of the use of a weighted voting scheme to predict the gene function using gene expression data is discussed (Lan et al., 2007). Predictions from the following base classifiers were combined in their work: logistics regression, quadratic discriminant analysis (QDA), KNN, LDA, and NB. Their paper demonstrates that the combined classifier method outperformed all of the individual classifiers in predicting whether a gene responded to stress or not. Lan et al. used a weighted majority scheme with weights representing the area under the receiver operating characteristic curve up to 50 false positives (ROC50) for each classifier.

An example of a technique that combines the results of diverse classifiers using a weighted voting scheme is stacked generalisation or stacking (Wolpert, 1992; Witten and Eibe, 2005). In stacking, two levels of learners are used: level-0 and level-1 models. The level-0 models are base classifiers that use the training data. The predicted classes of the level-0 classifiers are used as inputs to the level-1 model or meta-learner. Cross-validated predictions generated by the level-0 classifiers are used to train the level-1 meta-learner. Stacking determines how to best combine the predicted classes of the base classifiers using a weighted voting scheme represented by the meta-learner which was trained using
Combining diverse classifiers using precision index functions

the cross-validated predictions. Stacking can also be formulated to include other inputs such as the class posterior probabilities of the individual classifiers and the weights can vary depending on the input location \( x \) (Hastie et al., 2009).

The second major type of combining strategy is classifier selection. The two types of classifier selection methods are:

1. static classifier selection


In static classifier selection, the feature space is partitioned into \( K \) regions and the classifier with the highest estimated prediction accuracy for a given region is selected. An example of the static classifier selection method is the clustering and selection algorithm (Kuncheva, 2000). The algorithm first divides the training data into \( K \) clusters using the \( K \)-means clustering method and then selects the most competent classifier for each cluster based on estimated classification accuracy.

The second form of classifier selection is known as dynamic classifier selection (Woods et al., 1997). In this method, a measure of the overall local accuracy or local class accuracy for each individual classifier is used for a given input \( x \). The overall local accuracy estimate uses the \( K \)-nearest points of the training data to calculate the classifier prediction accuracy. The local class accuracy estimate uses the \( K \)-nearest predictions of a classifier for a given class to calculate the classifier local class prediction accuracy.

Combinations of classifier fusion and classifier selection strategies based on the performance of the classifiers in the different regions of the feature space have been used (Kuncheva, 2002). Kuncheva applied a selection strategy to regions of the feature space in which one classifier dominates strongly over others and the fusion strategy to the remaining regions of the feature space.

A combination technique recently developed using a precision index (PIN) function (Ko and Windle, 2011) is an example of a class selection strategy. This technique combines classifiers using the PIN index which is a PIN function that represents the observed overall prediction accuracy of each classifier. According to Ko and Windle, the PIN is a unifying measure that has the properties of a probability measure and can be used to select the best classifier among a group of classifiers. The PIN is defined as the precision of a set containing all observations meeting a certain maximum posterior probability (MaxP). The MaxP is the maximum posterior probability \( \max_j p(j \mid x) \) where \( p(j \mid x) \) represents the posterior probability of \( x \) to class \( j \).

In this paper, a new class specific precision index (PIC) is introduced which estimates the local class prediction accuracy of a classifier and a combining method based on PIC is evaluated. We show that the newly proposed PIC-based combining method compares the class-specific performance of individual classifiers and leverages the strengths of the different classifiers, generating much higher recall and precision sets of predictions. This paper applies the PIC and other combining methods to two well-known datasets, the vowel recognition data (Hastie et al., 2009) which is a class balanced dataset, and the yeast protein localisation data (Frank and Asuncion, 2010), which is an class imbalanced dataset. This paper also evaluates the use of a weighted PIN/PIC index to leverage the strengths of both methods when combining the results of diverse classifiers.
2 Methods

2.1 PIN functions

Let \( (\hat{y}, y, x, MaxP(x)) \) be a prediction point from a classifier where \( \hat{y} \) is the predicted class, \( y \) the true class, \( x \) is a vector of input variables, and \( MaxP(x) \) is the maximum posterior probability \( max_j (p(j|x)) \) where \( p(j|x) \) represents the posterior probability of \( x \) to class \( j \). Define the PIN of a subset \( A \) of prediction points as the number of correct predictions (points with \( \hat{y} = y \)) divided by the number of points in \( A \). The PIN function at \( a \), \( PIN(a) \) is defined as the PIN of the set \( A = \{ (\hat{y}, y, x, MaxP(x)) | MaxP(x) \geq a \} \) and the PIC of class \( k \) at \( a \), \( PIC_k(a) \) is defined as the precision of the set \( A_k = \{ (\hat{y}, y, x, MaxP(x)) | MaxP(x) \geq a \text{ and } \hat{y} = k \} \). Ko and Windle (2011) define \( PIN(x) \) as the precision of the set \( A = \{ (\hat{y}, y, x, MaxP(x)) | MaxP(x) \geq MaxP(x) \} \).

This definition is extended to a PIC point \( x_k \), \( PIC_k(x) \) for a single classifier as \( PIC_k(x) = prec( (\hat{y}, y, x, MaxP(x), MaxP(x)) \} ) \) where \( \hat{y} = k \), i.e., \( PIC_k(x) \) is precision of the set \( A = \{ (\hat{y}, y, x, MaxP(x)) | MaxP(x) \geq MaxP(x) \} \). Both the PIN \( PIN(x) \) and the PIC \( PIC_k(x) \) are assumed to be monotonically non-decreasing functions of \( MaxP(x) \) so that as \( MaxP(x) \) increases so does the precision for the prediction set. For simplicity of notation, \( PIN(x) \) is used to represent \( PIN(MaxP(x)) \) and \( PIC_k(x) \) to represent \( PIC_k(MaxP(x)) \). The observed pairs of values of \( \{ (MaxP(x), PIN(x)) \}, i = 1, ..., n \} \) are used to generate a monotonically non-decreasing PIN(.) function, which represent the relationship between prediction probability (the maximum posterior probability) and PIN. Similarly, the observed pairs of values of \( \{ (MaxP(x), PIC_k(x)) \}, i = 1, ..., n \} \) for each class \( k \) are used to generate a monotonically non-decreasing PIC(.) function relating the maximum posterior probability and PIC.

To generate PIN(.) and PIC(.) for a single classifier, a set prediction points \( (\hat{y}, y, x, MaxP(x)) \) are needed. The single ten-fold cross-validation scheme is used to generate a set of prediction points for a given training set. In single ten-fold cross validation, the training data is partitioned into ten folds, nine folds are used to train the individual classifiers and the remaining one fold to generate cross-validated predictions. This process is repeated ten times by alternating the 1 fold used to generate predictions and the nine folds used to training the learning methods.

To maintain the original class size, class-wise partitioning or stratified sampling (Japkowicz and Shah, 2011) is used where each class is partitioned into subsets of approximately equal size. The size of the prediction set is usually not large enough so 20–100 of the ten-fold random partitions are created to generate the cross-validated predictions. For each of the cross-validated predictions, the maximum posterior probability is calculated. Figure 1 shows a schematic of the repeated single ten-fold cross-validation scheme.

The cross-validated predictions \( \{ (\hat{y}_j, y_j, x_j, MaxP(x_j)) \}, j = 1, ..., t*n \} \) are used in calculating \( PIN(x) \) and \( PIC_k(x) \) for \( x \) where \( t \) is the number of random partitions. Note that \( y_j \), the true class, is not used in estimating \( \hat{y}_j \). For each class \( k \), the set of prediction points \( \{ (\hat{y}_j, y_j, x_j, MaxP(x_j)) | \hat{y}_j = k \} \) are used in calculating \( PIC_k(a) \) for any threshold \( a \) and hence \( PIC_k(MaxP(x)) \) for the data \( x \).
For each predicted class $k$, let $I_k$ be the number of prediction points whose prediction class is $k$. From the observed $\{(\text{MaxP}(x_i)), \text{PIC}_k(\text{MaxP}(x_i)) \mid i = 1,\ldots,I_k\}$, we use a monotone regression (antitonic regression) and linear interpolation to interpolate/extrapolate $\text{PIC}_k(a)$ for all values $a$, $0 \leq a \leq 1$. $\text{PIN}(a)$ is similarly generated based on the set of all predictions $\{(\text{MaxP}(x_i)), \text{PIN}(\text{MaxP}(x_i)) \mid i = 1,\ldots,t*n\}$.

Once the PIN and PIC curves have been constructed using the cross-validated predictions and monotone regressions, the curves are used to generate the PIN and PIC.
values for test data. For each test observation \((\hat{y}, x, MaxP(x))\), the PIN or PIC curve is evaluated at the value \(MaxP(x)\) to generate the PIN or PIC values at \(x\). Recall-precision curves are then be used to evaluate the performance of a classifier applied to the test data using the predicted class, the true class, and the PIN\((x)\) or PIC\((x)\) values. The recall-precision curve evaluates the average performance of a classifier over the entire prediction set and the class-specific recall-precision curve evaluates the performance of a classifier for each individual class.

The precision \(P(a)\) of a classifier at a value \(a\) is defined as the number of correct predictions in set \(A = \{(\hat{y}, y, x, PIN(x)) | PIN(x) \geq a\}\) divided by the size of the set \(A\). The recall \(R(a)\) is defined as the number of correct predictions in the set \(A\) divided by the size of the whole set of predictions in the test dataset. The set of points \(\{(P(a), R(a)) | 0 \leq a \leq 1\}\) defines the recall-precision curve and is used to assess the performance of a classifier. To generate the recall-precision curve when using the PIC, the \(PIC(x)\) values for the test data are used to define the set \(A = \{(\hat{y}, y, x, PIC(x)) | PIC(x) \geq a\}\) used in the calculations.

The class specific precision \(Pd(a)\) for class \(k\) and subset \(A_k = \{(\hat{y}, y, x, PIC(x)) | PIC(x) \geq a\}\) for \(0 \leq a \leq 1\), is defined as the number of correct predictions in set \(A_k\) divided by the size of the subset \(A_k\). The class specific recall \(Rd(a)\) is defined as the number of correct predictions in set \(A_k\) divided by the size of the whole set of predictions whose true class is \(k\). Both the recall \(R(a)\) and the class-specific recall \(Rd(a)\) are assumed to be non-increasing functions of \(a\) and the precision \(P(a)\) and the class specific precision \(Pd(a)\) are assumed to be non-decreasing functions of \(a\). A monotone regression (antitonic regression) and simple linear interpolation are used in estimating curves \(\{(P(a), R(a)) | 0 \leq a \leq 1\}\) and \(\{(P(a), Rd(a)) | 0 \leq a \leq 1\}\) from the observed \(a_i = PIC(x), i = 1, \ldots, J\).

Let \(M_1, M_2, \ldots, M_L\) be a set of \(L\) classifiers and the PIN\(_{MC}(a)\) and PIC\(_{MC}(a)\) for \(0 \leq a \leq 1\) be the PIN and PIC curves for classifier \(M_c, c = 1, \ldots, L\). The recall \(R_{MC}(a)\), precision \(P_{MC}(a)\), class \(k\) recall \(R_{k MC}(a)\), and class \(k\) precision \(P_{k MC}(a)\) are defined similarly for classifier \(M_c, c = 1, \ldots, L\).

When multiple classifiers are considered, the PIN combining algorithm selects one prediction from the predictions of the multiple classifiers as follows. Let \(M_1, M_2, \ldots, M_L\) be the set of classifiers and PIN\(_{MC}(x)\), PIN\(_{M2}(x)\), \ldots, PIN\(_{ML}(x)\) be the corresponding PIN values for a point \(x\) and \(\hat{y}_{MC}(x), \hat{y}_{M2}(x), \ldots, \hat{y}_{ML}(x)\) be the corresponding predicted classes. The PIN combined classifier (PIN.Com) selects the predicted class of the classifier with the highest PIN value (Ko and Windle, 2011). The combined classifier is known as the PIN.Com (Ko and Windle, 2011). In the event of a PIN tie, where more than one classifier has the highest PIN value, the predicted class is selected randomly from the classes predicted by the classifiers with the highest PIN value.

A new classifier PIC.Com is defined as follows. For notational simplicity, PIC\(_{MC}(x)\) is used for PIC\(_{MC}(x)\) where \(kMC = \hat{y}_{MC}(x)\), the predicted class of \(x\) for classifier \(MC\). Let \(M_1, M_2, \ldots, M_L\) be the set of classifiers and PIC\(_{M1}(x)\), PIC\(_{M2}(x)\), \ldots, PIC\(_{ML}(x)\) be the corresponding predicted classes. The predicted class of PIC combined classifier PIC.Com is defined as the class predicted by the classifier with the highest PIC value. The PIC combined classifier PIC.Com defines the predicted class as the class predicted by the
classifier with the highest PIC value. In the event of a tie, the predicted class is selected randomly from the classes predicted by the classifiers with the highest PIC value. The multiple classifiers considered could be of different learning types such as RF and SVM or of the same type such as a NNs with different number of hidden nodes.

**Figure 2** Repeated double ten-fold cross validation scheme
When test data is not available, a repeated double ten-fold cross validation scheme can be used from the training data to evaluate the performance of the PIN and PIC combining algorithms. Figure 2 illustrates the repeated double ten-fold cross validation scheme. In this process, 20–50 random partitions of the original training data are typically created. The cross-validated predictions generated through the double cross-validation scheme and known true classes are used to build the PIN and PIC curves. The cross-validated predictions generated via the single ten-fold cross-validation scheme described in Figure 3 are used to evaluate the performance of the PIN and PIC algorithms using recall-precision curves.

A new PIN is defined as a weighted average of the PIN and PIC indexes with a corresponding combining classifier who selects the predicted class as the class that maximises the new PIN. This approach can be viewed as combining the opinions of two experts where the PIN represents the overall prediction accuracy of a classifier for a given prediction point and the PIC represents the local class prediction accuracy of the classifier. By using a PIN than combines the PIN and PIC indexes, the ability of the combining method to discriminate between high and low precision predictions is improved.

Let \((\text{PIN}(x), \text{PIC}(x))\) be a prediction point where \(\text{PIN}(x)\) is the overall PIN, and \(\text{PIC}(x)\) the PIC for the predicted class \(\hat{y}(x)\). The weighted PIN/PIC index for the prediction point is defined as follows:

\[
\text{PIN}_{w}\text{PIC}_{1-w}(x) = w\text{PIN}(x) + (1-w)\text{PIC}(x)
\]

where \(w\) is a number between 0 and 1, \(x\) is the input vector, and \(\text{PIN}(x)\) and \(\text{PIC}(x)\) the values of the overall and class specific precision indexes for the given input vector \(x\). Note that when \(w = 1\), the index reduces to \(\text{PIN}(x)\) and when \(w = 0\), the index reduces to \(\text{PIC}(x)\). The combined classifier using the \(\text{PIN}_{w}\text{PIC}_{1-w}\) weighted index selects the predicted class of the classifier with the highest \(\text{PIN}_{w}\text{PIC}_{1-w}(x)\) value for the input \(x\).

To determine the best weight \(w\) for a given dataset, if the ‘true’ class is known for the test data, the recall-precision curves can be examined for different values of \(w\). If the ‘true’ class for the test data is not known, a repeated double ten-fold cross validation scheme can be used to determine the weight \(w\) that yields the best prediction performance for the combining algorithm. The weighting of PIN and PIC is done at the individual classifier level and the resulting weighted PIN/PIC index values are used to generate the combined classifier prediction just like in the PIN or PIC algorithm.

### 2.2 Theoretical justification

It can be shown that the classifier that maximises the PIN or PIC index (or any weighted PIN/PIC index) is the optimal choice by an almost identical approach and proof as used in the decision theoretic optimal class selection (Ripley, 1996). The optimality proof for the combining method based on PIC index (or weighted PIN/PIC index) is almost identical to the PIN.Com case presented here.

Let \(M_1, \ldots, M_L\) be the classifiers used to classify a random feature vector \(X\) and \(M_1(X), \ldots, M_L(X)\) be the predicted class by the classifiers. The objective is to select the classifier that has the highest precision for the prediction. For a given \(X\) in a suitable feature space \(X\), let \(\hat{S}\) be the best classifier selection according to the highest precision and \(\hat{S} : X \rightarrow \{1, 2, \ldots, L\}\) be a selection procedure of a classifier among \(\{M_1, \ldots, M_L\}\). Assume
that the proportion of classifier \( M_c \), being the best classifier in the population under study is \( \pi_c \). Feature vectors from which \( M_c \) is the best classifier are distributed as \( p_c(\mathbf{x}) \). The task is to select the best classifier on the basis of the observed value \( X = \mathbf{x} \); decision \( c \) corresponding to claiming \( M_c \) is the best classifier. The probability of a wrong selection assuming that there is always one best classifier for each \( x \) can be defined as

\[
pws(c) = \Pr\{\hat{S}(\mathbf{x}) \neq c \mid S(\mathbf{x}) = c\}
\]

Consider the following loss function

\[
L(c, l) = \begin{cases} 
0 & \text{if } l = c \\
1 & \text{if } l \neq c 
\end{cases}
\]

for \( l = 1, 2, \ldots, L \).

The risk function for selecting \( \hat{S} \) is the expected loss when using this selection procedure.

\[
R(\hat{S}, c) = \mathbb{E}[L(c, \hat{S}(X)) \mid S = c] = \sum_{l=1}^{L} L(c, l) p(\hat{S}(\mathbf{x}) = l \mid S = c) = pws(c)
\]

The total risk is

\[
R(\hat{S}) = \mathbb{E}[R(\hat{S}, S)] = \sum_{c=1}^{L} \pi_c pws(c)
\]

Let \( p(c \mid x) = \Pr\{S = c \mid X = \mathbf{x}\} \) be the posterior probability of correctly selecting \( M_c \) using PIN index given \( X = \mathbf{x} \). The following shows that the selection rule which minimises the total risk under the above loss function is \( \hat{S}(\mathbf{x}) = c \) if \( p(c \mid x) = \max_{l=1, \ldots, L} p(l \mid x) \).

**Proof:**

\[
p(c \mid x) = \Pr\{S = c \mid X = \mathbf{x}\} = \frac{\pi_c \text{PIN}_c(\mathbf{x})}{\sum_{j=1}^{L} \pi_j \text{PIN}_j(\mathbf{x})}
\]

and

\[
R(\hat{S}) = \mathbb{E}[R(\hat{S}, S)] = \mathbb{E}\left[ \mathbb{E}\left[ L(S, \hat{S}(X)) \mid X \right] \right] = \int \mathbb{E}\left[ L(S, \hat{S}(X)) \mid X = x \right] p(x)dx
\]

where \( p(x) = \sum_{j=1}^{L} \pi_j \text{PIN}_j(\mathbf{x}) \) is the marginal precision density for \( X \). It is sufficient to minimise the conditional expectation, which can be written as \( \sum_{l=1}^{L} L(l, c) p(l \mid x) \), with respect to \( c \), for each \( x \). Under the given loss, the minimum becomes \( 1 - p(1 \mid x), \ldots, 1 - p(L \mid x) \), when \( \hat{S}(\mathbf{x}) = 1, \ldots, \) and \( L \) respectively. This is equivalent to finding the maximum of \( p(1 \mid x), \ldots, p(L \mid x) \) which is the solution given above.
Now to illustrate how the PIC method can perform better than the PIN method, assume a binary classification problem with two classifiers with sufficient data available to construct accurate PIC curves for both classes. Let $\text{PIN}_1(x)$ and $\text{PIN}_2(x)$ be the PIN values of classifier 1 and 2 at input $x$, respectively. Assume classifier 1 predicts class 1 with $\text{PIC}_1(x)$ value and classifier 2 predicts class 2 with $\text{PIC}_2(x)$ value at input $x$. Let $\text{PIN}_1(x) > \text{PIN}_2(x)$ and $\text{PIC}_2(x) > \text{PIC}_1(x)$ which states that classifier 1 has a higher average precision than classifier 2 at input $x$ and classifier 2 has a higher precision for its predicted class than classifier 1.

Using the maximum PIN decision rule would predict class 1 from classifier 1 while using PIC decision rule would predict class 2 from classifier 2. Since the PIC decision rule selects the classifier whose predicted class has the highest precision it gives the correct answer.

3 Analysis and results

3.1 Application of PIN and PIC methods to vowel recognition data

The performance of the combining methods based on PIN and PIC functions was compared to the performance of several combining methods for the vowel recognition data (Hastie et al., 2009). The vowel recognition data represents recorded sounds of 11 words of British English spoken by 15 speakers. The recorded vowel sounds were processed using speech processing methods and the resulting data from eight speakers was used as training data and from seven speakers as test data. The distribution of classes and number of observations for training and test datasets are shown in Table 1. The dataset contains 11 classes and ten predictor variables. The vowel recognition dataset is a class balanced dataset where all classes are represented equally.

Table 1  Vowel recognition data class distribution class vowel word training observations test observations

<table>
<thead>
<tr>
<th>Class</th>
<th>Vowel</th>
<th>Word</th>
<th>Training observations</th>
<th>Test observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>i</td>
<td>Heed</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>2</td>
<td>I</td>
<td>Hid</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>3</td>
<td>E</td>
<td>Head</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>Had</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>5</td>
<td>a:</td>
<td>Hard</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>6</td>
<td>Y</td>
<td>Hud</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>7</td>
<td>O</td>
<td>Hod</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>8</td>
<td>C:</td>
<td>Hoard</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>9</td>
<td>U</td>
<td>Hood</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>10</td>
<td>u:</td>
<td>Who’d</td>
<td>48</td>
<td>42</td>
</tr>
<tr>
<td>11</td>
<td>3:</td>
<td>Heard</td>
<td>48</td>
<td>42</td>
</tr>
</tbody>
</table>
The following individual classifiers were considered in the analysis: RF method with 1,000 trees with three variables selected randomly at each node, NN with one hidden layer containing ten nodes, SVM with radial basis function kernel, KNN with $k = 6$, and the MARS with 15 maximum number of terms.

For the RF method, the randomForest function in the R randomForest library (Breiman, 2002) was used. The NN method results were generated using the R nnet library (Venables and Ripley, 2002). To generate the SVM method results, the svm function in the R e1071 library (Chang and Lin, 2001, 2009; Cortes and Vapnik, 1995; Vapnik, 1998; Hsu and Lin, 2002) was used. The KNN method results were generated using the knn function in the R class library. The MARS method results were generated using mars and fda functions in the R mda library.

The following combining methods were considered: majority voting, two stacked generalisation schemes, two cluster-selection schemes, the PIN method, the PIC method, and a weighted combination of PIN and PIC. For the stacked generalisation method, two level-1 trainers were considered (meta-learners) (Ko and Windle, 2011). The first meta-learner (stage-1 model) used the class posterior probabilities for each prediction point as inputs to a multi-response linear regression (MLR) model which represents each class outcome as a 0/1 outcome discrete variable (Ko and Windle, 2011). The class for the linear model predicting the highest response value is selected as the combining method predicted class. The second meta-learner (stage-2 model) used the class posterior probabilities and the predicted classes of the level-0 base classifiers as inputs to a RF model.

The static cluster-selection algorithm (Kuncheva, 2000) was used as one of the cluster-selection methods. The K-means algorithm (MacQueen, 1967; Hartigan and Wong, 1979; Forgy, 1965; Hastie et al., 2009) was used to generate the different regions. The K-means function in the R stats library was used (Reinhardt and Hubbard, 1998). Single ten-fold cross-validated predictions were used to estimate the prediction accuracy of each individual classifier for the different regions. The KNN algorithm with parameter $k = 1$ was used to assign the single ten-fold cross-validated predictions to a corresponding region. Scenarios with 10, 20, and 30 regions were considered.

An extension of the static cluster-selection algorithm to the class-specific level was also developed where the class prediction accuracies of the individual classifiers in the different regions were computed. The combining method selects the predicted class of the classifier yielding the highest prediction accuracy for its predicted class. Scenarios with 10, 20, and 30 regions were considered. The case with ten regions gave the best performance for the standard cluster selection method and with 30 regions for the class-specific method.

For the PIN and PIC combining methods, a repeated single ten-fold cross validation scheme with 100 random partitions was used to generate the PIN and PIC curves. Figure 3 shows the resulting recall-precision curves for the individual classifiers and the PIC.Com classifier and Figure 4 for the individual classifiers and the PIN.Com classifier. Note that in terms of overall precision, which typically has been the primary criteria considered when evaluating combining methods (Japkowicz and Shah, 2011), both the PIC.Com and PIN.Com classifiers yielded better performance than the individual classifiers. However, in terms of the goal of this work which is to generate high recall high precision prediction sets for screening of predictions, the PIN.Com met it generating maximum precision of 95% while the PIC.Com did not generating a more limited
maximum precision of 71%. When examining the individual classifiers performance, it appears that the inability of the RF method to generate high precisions using PIC values limited the PIC combining method.

Since our main motivation for using a PIC is to generate prediction sets with high recall and precision for screening of predictions where actual verification is time consuming and costly, we examined a couple of ways to extend the PIC.Com classifier results to higher precisions. First, we evaluated the addition of the second best predicted class and corresponding posterior probability in the construction of the PIC curves. The original PIC curves were constructed using only the individual classifier predicted class with the highest class posterior probability ($\text{MaxP}$). The reasoning behind this approach is that by adding the second predicted class, more data would be available to generate the PIC curves and improve the ability of the method to differentiate between good and bad predictions.

Figure 5 provides the recall-precision curves for the individual classifiers and the PIC.Com classifier using the modified PIC curves. The PIC.Com generated higher overall precision when compared to the individual classifiers and precisions were extended from 71% to 78% with the addition of the second class in the construction of PIC curves. Even with this improvement, the PIC.Com classifier did not generate precisions as high as those obtained with the PIN.Com classifier and apparently was limited by the inability of the KNN and SVM methods to generate high precisions.

**Figure 3** Recall-precision curves for PIC method – vowel data
Combining diverse classifiers using precision index functions

Figure 4  Recall-precision curves for PIN method – vowel data

Figure 5  Recall-precision curves for modified PIC method – vowel data
The second method considered to extend the performance of the PIC.Com classifier to higher precisions used a weighted PIN based on a combination of PIC and PIN values to leverage the strengths of both methods. Different weighted combinations of PIN and PIC indexes were evaluated to examine the impact of PIC weight on the performance of the combining method. The modified PIC curves which used the addition of the second best predicted class were used to generate the PIC values for these weighted combinations. PIC weights ranging from 0% to 100% at 20% increments were evaluated. Figure 6 provides the recall-precision curves for the different PIN/PIC weighted combinations. Note that when compared to the PIN.Com classifier, the combining method using the 20% PIN/80% PIC weighted index yielded lower recalls up to 72% precisions followed by higher recalls in the 72% to 85% precision range and about the same performance for precisions exceeding 85%. The trade-off between overall precision and mid-precision performance can be clearly seen as we changed the PIC weight.

Table 2 summarises the recalls at different precisions for the combining methods and Figure 7 provides the recall-precision curves for these methods. Note that by using the modified PIC curves instead of the original PIC curves, the overall precision increased from 62.3% to 63.4% and the maximum precision from 71% to 78%. The combining method using the 20% PIN/80% PIC weighted index also gave better performance when using the modified PIC curves. Note that the PIC method by itself is outperformed by several methods but its performance is significantly improved when combined with the PIN index. Note that the combined classifier using the 20% PIN/80% PIC weighted index outperformed most combining methods at precisions higher than 72%. Just like the PIC.Com classifier, the class-specific cluster-selection method with 30 regions (CSC30) failed to generate high precision prediction sets. The overall precisions obtained with the PIN methods exceeded the 61% overall precision previously reported using MARS method (Hastie et al., 2009).

Figure 6  Recall-precision curves for PIN/PIC combinations – vowel data
Combining diverse classifiers using precision index functions

Table 2 Combining methods recalls for different precisions – vowel data precision overall

<table>
<thead>
<tr>
<th>Precision</th>
<th>Overall</th>
<th>60</th>
<th>65</th>
<th>70</th>
<th>75</th>
<th>80</th>
<th>85</th>
<th>90</th>
<th>95</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOTE</td>
<td>62.6</td>
<td>58.2</td>
<td>46.5</td>
<td>32.5</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>STK1</td>
<td>57.7</td>
<td>53.3</td>
<td>45.9</td>
<td>40.9</td>
<td>32.0</td>
<td>17.0</td>
<td>9.5</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>STK2</td>
<td>61.5</td>
<td>53.7</td>
<td>42.3</td>
<td>33.8</td>
<td>24.4</td>
<td>13.2</td>
<td>9.7</td>
<td>8.3</td>
<td>0.0</td>
</tr>
<tr>
<td>CS10</td>
<td>59.5</td>
<td>54.7</td>
<td>17.3</td>
<td>10.2</td>
<td>6.6</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>CSC30</td>
<td>63.6</td>
<td>61.3</td>
<td>43.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>PIC</td>
<td>62.3</td>
<td>51.3</td>
<td>45.8</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>PIC_mod</td>
<td>63.4</td>
<td>52.2</td>
<td>46.3</td>
<td>39.1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>PIN</td>
<td>65.6</td>
<td>48.8</td>
<td>37.5</td>
<td>30.0</td>
<td>24.7</td>
<td>19.0</td>
<td>14.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIN2PIC8</td>
<td>62.6</td>
<td>50.1</td>
<td>43.7</td>
<td>38.3</td>
<td>31.0</td>
<td>24.3</td>
<td>19.3</td>
<td>14.4</td>
<td></td>
</tr>
<tr>
<td>PIN2PIC8_mod</td>
<td>64.1</td>
<td>53.6</td>
<td>43.9</td>
<td>39.9</td>
<td>32.3</td>
<td>25.8</td>
<td>19.8</td>
<td>14.6</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7 Recall-precision curves for combining methods – vowel data

3.2 Application of PIN and PIC methods to protein localisation data

To illustrate the application of combining methods using precision indexes to a class imbalanced dataset, a dataset consisting of yeast proteins classified into sub-cellular localisation sites based on amino acid sequence attributes was considered (Frank and Asuncion, 2010). Knowing a protein’s intra-cellular localisation helps in the understanding of its function in cellular processes, its role in healthy processes and onset
of diseases, and its potential use as a drug target. Experimental characterisation of protein localisation is a time consuming and labour intensive process so machine learning techniques can aid in the screening of potential strong candidates (Reinhardt and Hubbard, 1998; Chen, 2008, 2010).

The dataset contains proteins belonging to ten different sub-cellular localisation sites. Table 3 provides the distribution of protein localisation sites in the dataset. Note that the first four classes have a significant number of observations while the last six classes are under-represented in varying degrees. Since a separate test dataset was not available for the yeast protein localisation data, a repeated double ten-fold cross-validation scheme was used to evaluate the performance of the PIN and PIC methods. A total of 20 outer and 20 inner random partitions were used in the repeated double ten-fold cross-validation scheme (see Figure 2).

The following individual classifiers were used to evaluate the performance of the PIC.Com classifier for this dataset: RF method with 1,000 trees and five variables selected randomly at each node split, NN with one hidden layer containing ten nodes, SVM with radial basis function kernel, KNN with \( k = 7 \), and the MARS with 15 maximum number of terms.

<table>
<thead>
<tr>
<th>Class</th>
<th>Site</th>
<th>Description</th>
<th>Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CYT</td>
<td>Cytosolic or cytoskeletal</td>
<td>463</td>
</tr>
<tr>
<td>2</td>
<td>NUC</td>
<td>Nuclear</td>
<td>429</td>
</tr>
<tr>
<td>3</td>
<td>MIT</td>
<td>Mitochondrial</td>
<td>244</td>
</tr>
<tr>
<td>4</td>
<td>ME3</td>
<td>Membrane protein, no N-terminal signal</td>
<td>163</td>
</tr>
<tr>
<td>5</td>
<td>ME2</td>
<td>Membrane protein, uncleaved signal</td>
<td>51</td>
</tr>
<tr>
<td>6</td>
<td>ME1</td>
<td>Membrane protein, cleaved signal</td>
<td>44</td>
</tr>
<tr>
<td>7</td>
<td>EXC</td>
<td>Extracellular</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>VAC</td>
<td>Vacuolar</td>
<td>30</td>
</tr>
<tr>
<td>9</td>
<td>POX</td>
<td>Peroxisomal</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>ERL</td>
<td>Endoplasmic reticulum lumen</td>
<td>5</td>
</tr>
</tbody>
</table>

The following combining methods were considered: majority voting, one stacked generalisation scheme, two cluster-selection schemes, the PIN method, the PIC method, and a weighted combination of PIN and PIC. The stage 2 stacking scheme was considered which used the predicted classes and the class posterior probabilities of the individual classifiers as inputs to a RF meta-learner model. The stacking combining scheme used cross-validated predictions generated with double cross-validation scheme to train the meta-learner and corresponding cross-validated predictions generated with single cross-validation scheme to evaluate the performance of the combining algorithm. Static cluster-selection and class-specific extension methods were considered with 10, 20, and 30 regions. Prediction accuracies for both cluster-selection methods were estimated using cross-validated predictions generated with ten-fold double-cross validation scheme.
The recall-precision curves for the individual classifiers and the PIC.Com classifier are shown in Figure 8. The PIC.Com classifier yielded the same overall precision as the RF method, lower recalls than the RF method between 65% to 75% precisions, and higher
recalls between 75% to 100% precisions than all of the individual classifiers. The use of modified PIC curves was considered to improve the mid-precision range performance of the PIC.Com classifier but virtually no change was observed which apparently indicates the adequacy of the original PIC curves.

Figure 9 provides the overall recall-precision curves for the individual classifiers and the PIN.Com classifier. Note that the PIN.Com essentially generated the same results obtained with the RF method which was the best individual classifier.

Figure 10 shows the recall-precision curves for different PIN/PIC weighted combinations. Note that the PIN.Com outperformed the PIC.Com in the mid precision range while the PIC.Com outperformed the PIN.Com in the high precision range. The combined classifier using the 80% PIN/20% PIC weighted index outperformed the PIN.Com over the entire precision range demonstrating the benefits of combining the PIN and PIC indexes.

Table 4 summarises the recalls at different precisions for the combining methods and Figure 11 provides the recall-precision curves for these methods. The class-specific cluster-selection method with 30 regions (CSC30) outperformed all other methods including the PIN.Com classifier. The combining method using the 20% PIN/80% PIC weighted index outperformed the class-specific cluster selection method at precisions exceeding 90%.

The overall precisions reported above compared favourably to previously reported results (Chen, 2010) where NNs, decision trees, NB, and several Bayesian network methods were applied to this dataset. The best performing classifier reported by Chen consisting of a Bayesian model averaging method yielded a 59.5% overall precision which was exceeded by the PIN methods, the cluster-selection methods, and the stacking
Combining diverse classifiers using precision index functions

method. The results obtained here were also better than the 59.5% overall precision previously reported using KNN (Horton and Nakai, 1997).

Table 4 Combining methods recalls for different precisions – yeast data

<table>
<thead>
<tr>
<th></th>
<th>Overall</th>
<th>55</th>
<th>60</th>
<th>65</th>
<th>70</th>
<th>75</th>
<th>80</th>
<th>85</th>
<th>90</th>
<th>95</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOTE</td>
<td>50.2</td>
<td>41.6</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>STK2</td>
<td>62.3</td>
<td>59.6</td>
<td>50.0</td>
<td>34.4</td>
<td>19.1</td>
<td>11.2</td>
<td>5.5</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>CS10</td>
<td>61.5</td>
<td>53.5</td>
<td>29.5</td>
<td>19.6</td>
<td>4.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>CSC30</td>
<td>64.1</td>
<td>62.4</td>
<td>49.2</td>
<td>35.4</td>
<td>24.0</td>
<td>15.9</td>
<td>7.8</td>
<td>3.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC</td>
<td>61.6</td>
<td>55.7</td>
<td>43.7</td>
<td>22.6</td>
<td>18.2</td>
<td>14.3</td>
<td>10.0</td>
<td>3.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIC_mod</td>
<td>61.6</td>
<td>56.4</td>
<td>43.9</td>
<td>22.6</td>
<td>18.2</td>
<td>14.4</td>
<td>10.0</td>
<td>3.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIN</td>
<td>61.3</td>
<td>55.6</td>
<td>46.8</td>
<td>35.5</td>
<td>23.7</td>
<td>12.8</td>
<td>5.2</td>
<td>0.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIN2PIC8</td>
<td>61.9</td>
<td>57.1</td>
<td>45.7</td>
<td>22.8</td>
<td>18.2</td>
<td>14.5</td>
<td>9.5</td>
<td>4.7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 11 Recall-precision curves for combining methods – yeast data

These results suggest that if the goal is to screen a set of candidate proteins with high recalls and precisions for expensive experimental confirmation, use of a high PIC weight would be desirable. The combined classifier using the 20% PIN/80% PIC weighted index would choose almost twice as many proteins when compared to using the PIN.Com only (10% vs. 5.2% – Table 3) at 90% precisions.

Figure 12 compares the class distributions of the PIC.Com and the PIN.Com classifiers for 90% precision prediction sets. The 90% precision sets were constructed by selecting predictions with PIC and PIN indexes values equal to or greater than 0.9 which yielded prediction sets with precisions of approximately 90% for both methods. Note that
the PIC.Com generated a higher number of predictions than the PIN.Com classifier for several classes with the improvement being more significant for several of the smaller minority classes (6, 7, 9, 10). These results demonstrate the ability of the PIC method to generate higher prediction accuracy for minority classes than the PIN method when PIC curves are well defined.

Table 5 provides the confusion matrix for the PIC.Com classifier shown as percent of the number of observations of the given true class. Note that the confusion matrix shows that most of the error is associated with confusing the two major classes, cytoplasmic proteins and nuclear proteins (class 1 vs. class 2). These results are consistent with previous results (Horton and Nakai, 1997) where difficulty was reported with predicting vacuolar proteins (class 8).

Figure 12  90% precision PIN.Com and PIC.Com class frequency histograms

<table>
<thead>
<tr>
<th>Class</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>59%</td>
<td>27%</td>
<td>11%</td>
<td>2%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>2</td>
<td>26%</td>
<td>59%</td>
<td>9%</td>
<td>5%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>3</td>
<td>18%</td>
<td>9%</td>
<td>64%</td>
<td>5%</td>
<td>2%</td>
<td>1%</td>
<td>1%</td>
<td>0%</td>
<td>1%</td>
<td>0%</td>
</tr>
<tr>
<td>4</td>
<td>2%</td>
<td>6%</td>
<td>1%</td>
<td>91%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>5</td>
<td>10%</td>
<td>8%</td>
<td>14%</td>
<td>19%</td>
<td>23%</td>
<td>15%</td>
<td>11%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>6</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>4%</td>
<td>2%</td>
<td>84%</td>
<td>8%</td>
<td>0%</td>
<td>1%</td>
<td>0%</td>
</tr>
<tr>
<td>7</td>
<td>6%</td>
<td>1%</td>
<td>17%</td>
<td>0%</td>
<td>3%</td>
<td>16%</td>
<td>56%</td>
<td>0%</td>
<td>1%</td>
<td>0%</td>
</tr>
<tr>
<td>8</td>
<td>36%</td>
<td>13%</td>
<td>9%</td>
<td>24%</td>
<td>3%</td>
<td>4%</td>
<td>11%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>9</td>
<td>21%</td>
<td>7%</td>
<td>11%</td>
<td>2%</td>
<td>0%</td>
<td>7%</td>
<td>3%</td>
<td>0%</td>
<td>50%</td>
<td>0%</td>
</tr>
<tr>
<td>10</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>100%</td>
</tr>
</tbody>
</table>
4 Conclusions and future work

This paper shows that the PIC combined classifier can improve the performance of individual classifiers and the PIN combined classifier by better leveraging the differences in class-specific performance of the individual classifiers. The use of a weighted combination of the PIC and PIN indexes can yield larger prediction sets (recall) with higher precisions when compared to using PIN alone. The results suggest that if the goal of a researcher is to construct a high recall prediction set with high precisions (90%), a PIN-based method with high PIC weight would be preferred when compared to other combining methods.

When the PIC method was applied to the vowel recognition data, the maximum precision of the best subset selected by thresholding PIC index was 71% which was below the maximum of 95% precision of the best subset selected by thresholding the PIN index. The maximum precision of the PIC method was increased from 71% to 78% with the addition of the second best predicted class in the construction of the PIC curves. Even though the PIC method by itself was outperformed by several combining methods, its performance was significantly improved when combined with the PIN index. The combined classifier using the 20% PIN/80% PIC weighted index outperformed most combining methods with higher recalls for subsets selected whose precisions were higher than 72%. Just like the PIC combining classifier, the class-specific cluster-selection method with 30 regions (CSC30) developed as part of this work, failed to generate high precision prediction sets. The overall precisions obtained with the PIN combining methods exceeded the 61% overall best precision previously reported using other methods including MARS (Hastie et al., 2009).

When applied to the yeast protein localisation dataset, the PIC method was able generate prediction subsets with precisions up to 100%. When the recalls of the prediction subsets thresholded by the PIN index are compared to those of prediction subsets thresholded by the PIC index, the PIN combining method outperformed the PIC method for subsets selected in the mid precision range while the PIC method outperformed the PIN method for subsets selected in the high precision range. The combined classifier using the 80% PIN/20% PIC weighted index outperformed the PIN method over the entire precision range demonstrating the benefits of combining the PIN and PIC indexes. The class specific cluster selection method, developed as part of this work, performed better than most combining methods over the entire precision range for this dataset.

Note that the PIC method performed very well at high precisions for the class imbalanced protein localisation dataset but not as well for the class balanced vowel recognition data. It is our view that the main reason for this difference in performance is the characterisation of the PIC curves for the two corresponding datasets. As this work demonstrates, by adding the second best predicted class in the construction of the PIC curves, the PIC combining method was better able to differentiate between good and bad predictions for the vowel recognition dataset and precisions of prediction subsets were extended to a higher range. Only a small improvement was observed for the protein localisation data which seems to indicate that the PIC curves were already well defined for this dataset. Therefore, the difference in the PIC method performance should be attributed to the characterisation of the PIC curves in both datasets and not to the degree of balance of the classes in the respective datasets. As demonstrated, by combining the
PIN and PIC indexes the strengths of both methods can be leveraged and shortcomings of the PIC curves can be somewhat mitigated.

In terms of the performance of some of the minority classes in the protein localisation data, the results showed that some minority classes performed very well (classes 6 and 10) whereas others did not (classes 5 and 8). We believe that the difference in performance between these minority classes is a reflection of the degree of class separation for the given feature space between the classes. Previous studies (Horton and Nakai, 1997) also showed difficulty in predicting class 8. The PIC method outperformed the PIN method for certain minority classes demonstrating the potential for improvement in class-specific performance associated with the PIC method.

This work also demonstrated how to determine the best PIC weight for combining PIC and PIN when the class of the test data is known. Here we tried only a few values of weights, namely 0%, 20%, 40%, 60%, 80%, 100% because the differences in performance were not too large but it could be extended to a finer percentage increment. When the class of the test data is unknown, the best PIC weight can be determined using a double cross validation scheme with training data. As illustrated in this work, the best PIC weight can vary depending on the range of precisions of the prediction subset the experimenter is seeking for further experimentation.

Future work associated with the PIC algorithm includes evaluating the performance of the algorithm when more classes and their associated posterior probabilities are considered in the construction of the PIC curves. The use of generalised linear models such as beta distribution model and non-parametric models such as smoothing splines to represent the PIC curves is an area of future work. Further study of the class-specific cluster selection scheme developed as part of this work in the determination of the optimum number of clusters for improved local class prediction accuracy should be considered. The use of a weighted index using PIN, PIC, and class-specific cluster-selection prediction accuracy is another area worthy of consideration. Evaluation of a stacking combining scheme using the PIC values, PIN values, and class-specific cluster-selection prediction accuracies along with the associated predicted classes as inputs to a meta-learner is another area of future work. Finally, we are processing a much larger dataset to apply the PIC/PIN methods. Our initial analysis shows that the larger dataset allows us to estimate the PIC curves better and the resulting PIC/PIN methods clearly are benefited from the large class sizes. The result will be reported in the near future.

Acknowledgements

This work was supported in part by NIH SNRP grant NS060658.

References

Combining diverse classifiers using precision index functions


