

Aggregated Classifiers and Variable Selection to Improve Prediction Capability

Classificatori Aggregati e Selezione delle Variabili per Aumentare la Capacità di Previsione.

Simone Borra

Dip. SEFEMEQ, University of Rome “Tor Vergata”, via Columbia 2, 00133, Roma

Riassunto: Molti lavori comparativi, sia teorici sia sperimentali, hanno mostrato come i classificatori aggregati possono far aumentare significativamente la capacità di previsione rispetto a singoli modelli di classificazione. Negli ultimi anni sono stati proposti diversi metodi di costruzione di classificatori aggregati, basati sia su tecniche di ricampionamento dei dati, sia sull'estrazione di sottoinsiemi di variabili esplicative. In entrambi i casi, è stato dimostrato che una procedura di selezione delle variabili interna agli algoritmi stessi può far incrementare notevolmente la capacità di previsione del modello. Una procedura di selezione delle variabili può, inoltre, agevolare l'individuazione delle variabili più importanti. In questo lavoro descriviamo i più recenti metodi di selezione delle variabili per classificatori aggregati.

Keywords: variable selection; aggregated classifiers; bagging; boosting; random variable subspaces.

1. Introduction

In regression and classification problems with a response Y and p predictors $X = (X_1, \dots, X_p)$ we can choose many alternatives of model fitting procedures to find an estimate $\hat{y} = \hat{f}(X)$ of the true target function. These procedures could be referred to parametric or nonparametric approaches but, anyway, the criteria for evaluating the quality of a model will differ according to two different aspects: prediction performance and interpretability. The first aspect concerns the accuracy of prediction on new data, preferring models with higher capability to predict new observations. In many problems regarding different areas, for instance, finance, environment, medicine, accuracy of prediction is the most relevant aim in the building of a model. The second aspect concerns the interpretation of the model, a difficult task when the number of predictors is large. In fact, in this case, it is reasonable to expect that some predictors are irrelevant whereas others are highly correlated. Then, it is desirable to delete irrelevant and redundant variables since, with high dimensional data, analytic methods incur in the “curse of dimensionality”.

The variable selection procedures can be classified as:

- a) procedures not included as part of model selection, identifying the explanatory variables that have remarkable effects without assuming any model (*model-free*);
- b) procedures incorporated within the process of building the model.

In literature, many model-free variable selection procedures have been proposed both for regression and classification problems. For regression problems, for instance, L. Li et al. (2005) advanced new model-free approaches based on *sufficient dimension reduction theory*. In classification problems, many attribute evaluators have been proposed to reduce the dimensionality of a dataset. For example, *Info-gain* or *Gain-ratio* based on a measure of entropy evaluates the worth of an attribute X with respect to the class Y .

The use of a model-free approach is particularly useful for data mining, where data sets with a large number of variables are analyzed. In that case, we would like preliminarily to screen out noise or redundant variables thus making the subsequent model building more easy and accurate.

However, most of variable selection procedures are incorporated into model selection. For example, in linear regression models we can consider many criterion functions as in Lasso (Tibshirani, 1996), a penalized least squares method imposing an L_1 -penalty on the regression coefficients, or in LARS-EN, a recent modification of Lasso (Zou et al., 2005). Foster et al. (2004) proposed a modified version of stepwise model selection to build a classification model on data sets with several hundreds of variables. However, the “optimal” subset of variables depends on the chosen model selection procedure.

For a classification problem, let $C_1, C_2, \dots, C_M$ collections of classes of additive classifiers based respectively on 1, 2, ..., M variables and, for each collection C_k , let

$f_{n,k}^*$ minimize over C_k the training classification error (apparent error),

$L_n(f_{n,k}) = \frac{1}{n} \sum_{i=1}^n I(f_k(x_i) \neq y_i)$, based on a data set $D_n = \{y_i, x_{i1}, \dots, x_{ip}\}_{i=1}^n$. A penalty model

selection chooses the classifier $f_{n,\hat{k}}^*$ that minimizes:

$$L_n(f_{n,k}^*) + pen(n,k)$$

among all classes $\{C_k\}_1^M$. The penalization term $pen(n,k)$ is a positive quantity, possibly data-dependent, and it can be interpreted as an estimate of the optimism (the expected difference between prediction error and apparent error). Determining the right amount of penalization is crucial to the success of the minimization of penalized empirical error. Theoretical results produced several procedures to reach the right size of penalization. For instance, when a log-likelihood loss function as $L_n(f)$ is used, a gaussian error additive model is supposed and $f_{n,k}^*$ is obtained by a linear fitting with k

variables or basis functions, we obtain Akaike information criterion (AIC, with $pen(n,k) = 2 \frac{k}{n}$) (Akaike, 1973), Bayesian information criterion (BIC, with $pen(n,k) = \log(n)k$) (Schwarz, 1978) and Risk inflation criterion (RIC, with $pen(n,k) = 2 \log(p)$) (Foster et al., 1994).

The assessment of these criteria requires knowledge of the empirical process. This partly explains the diffusion of re-sampling techniques like cross-validation or bootstrap (Hastie et al., 2001) particularly in data mining context where often the data generation process is unknown.

This work is concentrated on variable selection methods in supervised classification problems to improve interpretability and prediction capability of single and aggregated nonparametric classifiers. In Section 2, we briefly review variable selection techniques

for single classifiers. In Section 3, we describe some methods to estimate the importance of predictors. In section 4, we illustrate several methods to obtain aggregated classifiers and how to apply variable selection.

2. Variable selection in nonparametric classification

In machine learning literature, variable selection techniques for classification can be divided into two broad categories: filters and wrappers. Filters use general characteristics of the training data to evaluate attributes and operate independently of any learning algorithm. Applying a filter, we rank the variables and we can select the more important predictors. Differently, wrappers require one particular learning algorithm and use its performance to evaluate and decide the variable to select. A wrapper tends to find predictors more appropriated to the given learning algorithm, resulting in good prediction capability, but it is computationally more expensive than a filter (Liu et al., 2004). Another distinction is possible between methods that evaluate individual variables and methods that evaluate subsets of variables.

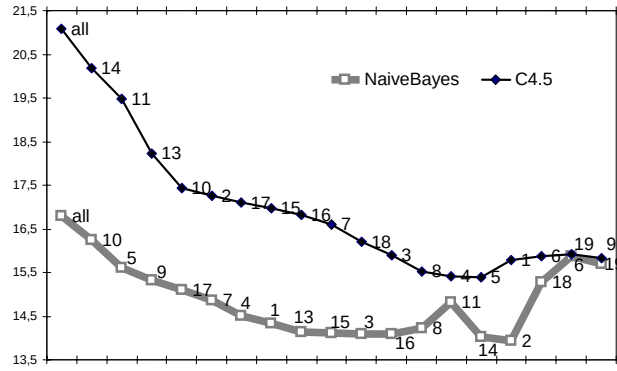
The simplest and fastest filter is the *Information Gain* (IG) based on the reduction of entropy of nominal response variable after observing one predictor. The filter *ReliefF* (Kononenko, 1994) ranks variables according to idea that a useful variable should differentiate between instances of different classes having similar value for instances of the same class. The filter *Correlation-based Feature Selection* (CFS) (Hall et al., 2003) evaluates subsets of variables and the heuristic index of merit for a subset S of k predictors is based on the average of correlations between each predictor and the response variable and on the average of correlations between predictors. *Consistency-based Subset Evaluation* (CSE) filter looks for subsets of predictors whose values divide the data into subsets containing a strong single class majority. Benchmark comparisons, as in Hall and Holmes (2003), show there is no best approach for all situations. However, CFS is faster and chooses fewer predictors, but in presence of strong predictor interactions, it is better use CSE or *ReliefF*. When we use data sets with large dimensionality, to avoid dramatic computational time consuming, we could apply filters only on a subsets of instances (generally a random or a stratified sample). A selective sampling approach to improve the capability of filters is proposed by Liu et al. (2004). They use a *kd*-tree (a generalization of the simple binary tree) to partition data then they draw a sample of instances from the partitions and finally the *ReliefF* algorithm is applied to select the predictors. A comparison on a large set of artificial and real data sets proves that selective sampling applied to a filter method can be effective in reduction of the number of training instances without performance deterioration.

Generally, when we use nonparametric classifiers as CART (Breiman et.al., 1984), MARS (Friedman, 1991) or Artificial Neural Networks all predictors are considered as input variables and a procedure to avoid the overfitting problem is usually adopted obtaining a simplified model. For example, with classification tree we will stop the growth of the tree according to some stopping criteria before it has become a full-grown tree (pre-pruning methods). Alternatively, we can use methods to prune branches of tree according to some criteria after the tree has grown fully (post-pruning methods).

With a wrapper technique combined with a search strategy it is possible to select the most relevant subset of predictors and to remove irrelevant and redundant information increasing the prediction capability of a classifier. An extensive review of variable

selection algorithms can be found in Kudo et al. (2000). For example, we have applied Naïve Bayes and C4.5 to the Hepatitis data-set, in which 155 hepatitis patients are classified as “survival” or “non survival” (19 covariates, 6 quantitative, include missing values). We have adopted a backward procedure for the variable selection based on prediction error estimated by Holdout Random Resampling (Borra et al., 2004). In Fig.1 we show the trend of the classifier’s error-rate when we drop recursively one variables at a time.

Figure 1: *Backward selection (NaiveBayes and C4.5 on Hepatitis data-set).*



In both cases, it is evident that using the backward procedure we can reduce the number of predictors.

Wrapper methods generally give better results than filters because of the interaction between the variable selection and the classification algorithm, but the improved performance comes at the cost of higher computational time consuming.

3. Importance of predictors

Applying filters or wrappers it is possible to obtain more parsimonious and powerful models but not necessarily it is simple to distinguish the relative importance of each input variable in predicting the response. Now we will concentrate our attention on classification trees and artificial neural networks, being the most used methods to build aggregated classifiers.

Given a classification tree T , let $\Delta i(s_{x_i}, t)$ the best improvement of an impurity measure $i(t)$ obtained splitting the node t with respect to the values of X_i . For a given node t , the variable X_i is named *primary variable* if provides the largest improvement over all variables, $\Delta i(s_{x_i}^*, t) = \max_{1 \leq h \leq p} \Delta i(s_{x_h}, t)$, and $s_{x_i}^*$ is called the *primary split*. The

variable which closely imitates the primary variable is defined *surrogate variable*. With the surrogate variable, we find a *surrogate split* $\tilde{s}$ producing two subsets of units most similar to those obtained with primary split. We remark that surrogate split $\tilde{s}$ does not correspond to the competing split, which provides the second largest improvement.

From the estimated classification tree, the importance of each variable could be calculated in several ways. A measure of importance of the variable X_h with respect to a given tree T (Breiman et al., 1984) is:

$$I(X_h) = \sum_{t \in T} \Delta i(\tilde{s}_{x_h}, t)$$

where $\tilde{s}_{x_h}$ could be a primary or a surrogate split or null over all nodes t of the tree T .

We can observe that this concept of variable importance is related to the method of surrogate splitting, so it is possible for a variable to be important even though it might not have been used for any split.

An alternative measure is:

$$I'(X_h) = \sum_{t \in T} \Delta i(s_{x_h}, t)$$

computed by summing, over all nodes in the tree T , the improvement of impurity given by X_h . The difference is that the former measure $I(X_h)$ considers only the values of improvement for each primary and surrogate split, but not the competing splits. Suppose there is one variable that for all nodes defines the competing split. Using $I(X_h)$ as importance measure, that variable could be entirely ignored, remaining unidentified as important variable. The meaning of importance in this case pertains to a specific final solution (tree), and if a predictor is never actually chosen for a split (or a surrogate), it is, indeed, not an important predictor in the specific final (chosen) tree. At the other hand the use of $I'(X_h)$ could lead to a potential inflation of importance for the predictors that were never chosen.

For artificial neural networks, there are a series of methods available to aid in interpretation of connection between weights and variable contributions (Olden et al. 2002). Simple visual methods are proposed as, for example, *neural interpretation diagram* (Özesmi et al., 1999) based on a visual interpretation of the connection weights among neurons. However, the effectiveness of this approach is reduced when the number of connections and hidden neurons increase. Garson's algorithm (1991) and its modifications are based on the absolute values of the input-hidden-output connection weights and the importance of an input variable is obtained summing the relative contribution of this variable to the outgoing signal of each hidden neuron. Recently several types of sensitivity analysis have been proposed to determine the spectrum of input variable contributions.

4. Aggregated classifiers and variable selection

A method to improve, often considerably, the accuracy in the prediction obtained with a single classifier is to train several different classifiers on a set of training examples and then to combine their predictions. The ensemble of classifiers is named aggregated classifier. Therefore, given a set S of training examples and a collection of L classifiers $h_1, h_2, \dots, h_L$ based on S , an aggregated classifier combines their individual decisions in some way (typically by weighted or unweighted voting) to classify new examples.

Theoretical and empirical researches (Quinlan, 1996; Bauer et al., 1999; Dietterich, 2000) have demonstrated that an aggregated classifier is often more accurate than any of the single classifiers in the ensemble.

To construct an aggregated classifier we have to take into account of several aspects: a) which base of classifiers to use to assemble an aggregated classifier; b) how choose training examples; c) which set of variables submit to single classifiers; d) the choice of integration method of predictions of the base classifiers.

Many studies have shown that an efficient aggregated classifier should consist of single classifiers both accurate and diverse in the sense that their errors occur in different regions of the variables. However, some techniques of aggregation as Bagging or Boosting show that assembling low accurate classifiers (weak learners) can be effective as well. The low accuracy of single classifiers is compensated by their diversity.

Measure of diversity

Many indices to evaluate the diversity of the base of classifiers have been considered (Tsymbal et al., 2003, 2005; Bacauskiene et al. 2004; Kuncheva et al., 2003). For instance, the Q -statistic, an index measuring diversity between couples of classifiers:

$$Q_{ij} = \frac{n^{11}n^{00} - n^{10}n^{01}}{n^{11}n^{00} + n^{10}n^{01}}$$

where n^{ab} is the number of examples in the data set classified correctly or incorrectly ($a=1$ or $a=0$) by the i -th classifier and correctly or incorrectly ($b=1$ or $b=0$) by the j -th classifier; or k -statistic:

$$k = \frac{\theta_1 - \theta_2}{1 - \theta_2} \quad \text{with} \quad \theta_1 = \frac{\sum_{i=1}^C n_{ii}}{n} \quad \text{and} \quad \theta_2 = \sum_{i=1}^C \left(\frac{n_{i\cdot}}{n} \cdot \frac{n_{\cdot i}}{n} \right)$$

where n_{ii} is the number of examples classified in class i by both classifiers; $n_{i\cdot}$ and $n_{\cdot i}$ are the number of examples classified in class i by the first and second classifier respectively; n is the total number of examples and C the number of classes.

Many methods generating base classifiers obtain single classifiers accurate in their predictions, but with high diversity.

Integration methods

A heterogeneous ensemble of classifiers is not a sufficient condition for the accuracy of an aggregated classifier. It is necessary a good integration method to utilize the diversity between single classifiers. There are many integration methods generally associated to specific techniques of subsampling (for example, bagging and boosting) or variable selection (for example, random subspace method).

Considering the classification of each single classifier, in *majority vote* the class value that receives the highest number of votes is selected as the final classification (ties are resolved in several ways). In *weighting vote*, to each vote a weight is assigned, which is normally related to the estimated generalization performance of the corresponding classifier. In the *cross-validation majority*, the accuracy of each classifier is estimated by cross-validation on the training set and then the classifier with the highest accuracy is selected. Alternatively, it is possible to use more sophisticated integration approaches (dynamic integration methods) where the integration procedure depends on the characteristics of each particular example to classify. For example, Tsymbal et al. (2003) show several approaches based on the weighted nearest neighbor classification procedure. Using this procedure, firstly, for each example of the training set the

classification error of each single classifier is estimated by cross-validation. Then, for each example in the test sample, a weighted k-nearest neighbor procedure based on the training data set is used to predict the local classification errors of each single classifier. In the last decade, many methods to build aggregated classifiers have been developed and they mainly differ for the choice of training examples or the variable selection method. Comparative studies have indicated that there is not preferable aggregated classifier, as their performance depends on the characteristics of the classification problem.

- *Subsampling methods*: a classifier is applied several times, each time with a different subset of the training sample. For instance:
 - Bagging (Breiman, 1996) use a collection of samples drawn randomly with replacement from the original training sample and the predictions are aggregated with majority vote.
 - Dagging, is similar to bagging but uses disjoint samples rather than bootstrapping.
 - Half & Half bagging (Breiman, 1998) uses iterative random sampling to collect new training samples that are half filled of examples correctly classified by the classifiers already constructed and half filled of misclassified examples.
 - Boosting (Freunde et al. 1996) uses a sequence of samples where the cases are differently weighted and the aggregated classifier is constructed by a weighted vote of the individual classifiers. In Hastie et. al. (2001) it is shown that Boosting, for a two classes problem, where the response variable y assume values $\{-1, +1\}$, is equivalent to forward stagewise additive modelling using the loss function $L(y, f(x)) = \exp(-yf(x))$.
 - A generalization of tree boosting is MART proposed by Friedman (2001). Many comparative studies using simulated and real data sets showed as Boosting classifiers are better than Bagging classifiers in prediction capability.
- *Manipulating the response variable*: This technique is based on a manipulation of the response variable. Dietterich et al. (1995) and Schapire (1997) have shown how boosting can be combined with a manipulation of response values to yield an excellent aggregated classification method calls Adaboost.OC.
- *Variable subspace methods*: The idea is to vary the variable subsets used to generate the base of classifiers to obtain higher diversity between the classifiers. In the Random subspace method, proposed by Ho (1998), for each step of aggregation all training examples are projected onto a randomly chosen variables subspace in which a classifier is derived. The number of variables drawn to form the subset can be fixed or let to vary as proposed in Tsymbal et al. (2005). These aggregated classifiers work well when the variables are highly redundant as well as the classification ability is spread over many informative variables (Skurichina et al., 2001). When the number of training examples is small as compared to the number of variables, constructing classifiers in several variable subspaces can be a way to reduce the dimensionality of data.
- *Injecting randomness*: A dose of randomness can be injected into the learning algorithm. For example, choosing randomly the initial weights of neural networks, or

including random subset of variables at each internal node in a classification tree (Random tree). In Random Forest, proposed by Breiman (2001), bagging is used in tandem with random feature selection at each node to grow each tree, so it combines both subsampling and random subspace methods. The resulting aggregated classifier (or forest) gives a prediction performance comparable to boosting and it is faster than bagging or boosting. Latine et al. (2000) proposed to combine bagging and random subspace method.

- *Heterogeneous classifier ensembles*: In this class of algorithms, different types of classifiers make up ensemble architecture. For example, Hothorn et al. (2003) using a double-bagging combines two extreme models as linear discriminant analysis and classification trees. With a stacked generalization approach (Wolpert, 1992), Ting et al. (1997) proposed the use of a learning method instead of a majority vote, to combine single classifiers obtained with bagging or dagging techniques.

Variable selection

In literature, works on variable selection for aggregated classifiers are very rare. However, we may distinguish between two different approaches:

- variable selection methods applied conjointly to variable subspace methods, where the purpose of selection is mainly to improve the prediction capability of aggregated classifier;
- variable selection methods applied conjointly to subsampling methods, oriented to improve the prediction capability and to find salient variables for classification.

In the first approach, the subsets of variables obtained from a variable subspace method play an important role in the diversity of a classifier ensemble and consequently in the rise of prediction capability of the aggregated classifier. However, a better performance may be achieved by discarding redundant or irrelevant variables with a variable selection procedure. An example of this approach is the work of Tsymbal et al. (2003). The authors present EFS_SBC algorithm for creating ensembles of simple Bayesian classifiers with variable selection using random subspace method. This algorithm use a refinement cycle based on a hill-climbing search that improves the accuracy and diversity of the base classifiers built on random variable subsets. For all variable subsets, each variable can be included or deleted. If the resulting variable subset allows the corresponding classifier to obtain better performance, the change is kept and the process continue until no further improvements are possible. The performance of i -th classifier is measured with a fitness function:

$$Fitness_i = acc_i + \alpha \cdot div_i$$

Therefore, the rise of performance takes into account of both accuracy and diversity and α is a parameter indicating the influence of diversity on the fitness. In a more recent work, Tsymbal et al. (2005) have extended EFS_SBC algorithm to other three search strategies for variable selection: genetic search, ensemble forward and backward sequential selection. They used also several measures of diversity and integration approaches and they carried on experiments on a large number of datasets observing that: genetic and hill-climb search was better than a simple random subspace method; the ensemble variable selection process can be sensitive to the choice of the diversity measure.

In the second approach, based on subsampling of examples, the whole variables set is submitted to each classifier and a selection procedure search, for each classifier, the most important variables.

A simple algorithm, proposed by Van de Laar et al. (2000), implements a variable selection based on aggregated neural networks. For each classifier in the ensemble the data set is randomly divided into a training and a test set. The algorithm provides for each size p (dimension of variable subset) an ensemble of classifiers where each classifier is based on the same number of input variables. A partial retraining algorithm is used to estimate the performance of a single neural network on a given set of variables. Then it is possible to summarize the results with a graph and to identify the most relevant variables for each dimension of the variable subset.

Bekaškiene et al. (2004) proposed a more sophisticated algorithm for identifying salient variables for classification in ensembles of neural networks. As in EFS_SBC algorithm, the variable selection is based on accuracy and diversity of neural networks included in the ensemble but to create different networks, they use a subsampling method as, for example, bagging or half & half bagging. In this algorithm, the number of classifiers aggregated increases during the learning process until reach an accuracy threshold for aggregated classifier. Therefore, for j -th classifier the least salient variable k is eliminated, according to the rule: $k = \arg \min_{i \in S} \bar{k}_i$, where $\bar{k}_i$ is the average k -statistic

calculated for the ensemble of classifiers when the i -th variable is eliminated from the input variable set of the j -th classifier. S indicates the subset of variables less important in terms of accuracy and then the variables to delete. In this way, the variable selection procedure for the j -th classifier deletes variables belonging to S whose elimination get the highest diversity. Experimental investigations show the capability of this algorithm to remove a large number of variables from the original data set without reducing the classification accuracy of the aggregated classifier.

Variable importance

When the final aggregated classifier is obtained, we may rank variables using an appropriated measure of importance. The measure of importance depends on the choice of the classifiers and on the methods used for constructing the aggregated classifier.

When we consider a subsampling method based on T bootstrapping samples, T importance measures can be calculated for each variable and they can be combined to obtain an aggregated measure. For instance, using bagging and classification trees as single predictors we can obtain the aggregated importance for h -th variable averaging values of $I_j(x_h)$ for $j = 1, 2, \dots, T$.

Another method to obtain an importance measure with bagging is to use out-of-bag examples. For example, to obtain the importance of k -th variable in a Random Forest, Breiman (2001) proposed to randomly permute its values in the out-of-bag examples and run down the corresponding tree on the out-of-bag data. The out-of-bag classes predicted by the aggregated classifier with the h -th variable noised up are compared with the true classes label to give a misclassification error. The importance of h -th variable is the percent increase in misclassification rate as compared to the out-of-bag rate with all variables intact.

References

- Akaike H. (1973), Information theory and an extension of the maximum likelihood principle, *Second International Symposium on Information Theory*, eds B.N. Petrov and Csàki F., 267-281.
- Bacauskiene M., Verikas A. (2004) Selecting salient features for classification based on neural network committees, *Pattern Recognition Letters*, 25, 1879-1891.
- Bauer E., Kohavi R. (1999) An empirical comparison of voting classification algorithms: Bagging, Boosting and variants, *Machine Learning*, 36, 105-139.
- Borra S., Di Ciaccio A. (2004), Methods to compare nonparametric classifiers and to select the predictors, in *New Developments in Classification and Data Analysis*, M. Vichi, P. Monari, S. Mignani, A. Montanari editors, Springer-Verlag, Berlin, pp.11-20.
- Breiman L. (1996) Bagging predictors, *Machine Learning*, 24, 123-140.
- Breiman L. (1998) Half & Half bagging and hard boundary points, *Tech. Rep. 534*, Statistics Department, Univ. of California, Berkeley.
- Breiman L. (2001) Random Forests, *Machine Learning*, 45, 5-32.
- Breiman L., Friedman J., Olshen R., Stone C. (1984) *Classification and Regression Trees*, Wadsworth.
- Dietterich T. G. (2000) An experimental comparison of three methods for constructing ensembles of decision tree: bagging, boosting, and randomization, *Machine Learning*, 40, 139-157.
- Dietterich T. G., Bakiri G. (1995) Solving multiclass learning problems via error-correcting output codes, *Journal of Artificial Intelligence Research*, 2, 263-286.
- Foster D. P., George E. I. (1994), The risk inflation criterion for multiple regression, *Ann. Statist.*, 22, 1947-1975.
- Foster P., Stine R. A. (2004) Variable selection in data mining: building a predictive model for bankruptcy, *J. A. S. A.*, vol.99, n.466, 303-313.
- Freund Y., Schapire R. E. (1996) Experiments with a new boosting algorithm, in *Proceedings of the Thirteenth National Conference on Machine Learning*, Morgan Kaufmann, 148-156.
- Friedman J. (1991) Multivariate adaptive regression splines, *Annals of Statistics*, 19, 1-141.
- Friedman J. (2001) Greedy function approximation: the gradient boosting machine, *Annals of Statistics*,
- Garzon G. D. (1991) Interpreting neural-network connection weights, *Artif. Intell. Expert*, 6, 47-51
- Hall M. A., Holmes G. (2003) Benchmarking attribute selection techniques for discrete class data mining, *IEEE Transaction on knowledge and data engineering*, vol.15, n.3.
- Hastie T., Tibshirani R., Friedman J. (2001), *The element of statistical learning: data mining, inference and prediction*, Springer.
- Ho T. K. (1998) The random subspace method for constructing decision forests, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, 832-844.
- Hothorn T., Lausen B. (2003) Double-bagging: combining classifiers by bootstrap aggregation, *Pattern Recognition*, 36, 1303-1309.

- Kononenko I. (1994) Estimating attributes: analysis and extensions of relief, in *Proceedings of the Seventh European Conference on Machine Learning*, Springer-Verlag, 171-182.
- Kudo M., Sklansky J. (2000) Comparison of algorithms that select features for pattern classifiers, *Pattern Recognition*, 33, 24-41.
- Kuncheva L. I., Whitaker C. J. (2003) Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, *Machine Learning*, 51, 181-207.
- Latinne P., Debeir O., Decaestecker C. (2000) Different ways of weakening decision trees and their impact on classification accuracy of DT combination, in *Multiple classifier systems: first international workshop*, eds Kittler J., Roli F., Springer, 200-209.
- Li L., Cook R.D., Nachtsheim C.J. (2005) Model-free variable selection, *J.R.Statist. Soc. B*, 67, Part 2, pp.285-299.
- Liu H., Motoda H., Yu L. (2004) A selective sampling approach to active feature selection, *Artificial Intelligence*, 159, 49-74.
- Olden J. D., Jackson D. A. (2002) Illuminating the “black box”: a randomization approach for understanding variable contributions in artificial neural networks, *Ecological Modelling*, 154, 135-150.
- Özesmi S. L., Özesmi U. (1999) An artificial neural network approach to spacial habitat modeling with interspecific interaction, *Ecological Modelling*, 116, 15-31.
- Quinlan J. R. (1996) Bagging, Boosting, and C4.5, in *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, AAAI Press and the MIT Press, 725-730.
- Schapire R. E. (1997) Using output codes to boost multiclass learning problems, in *Proceedings of the Fourteenth National Conference on Machine Learning*, Morgan Kaufmann, 313-321.
- Schwarz G. (1978), Estimating the dimension of a model, *Ann. Statist.*, 6, 461-464.
- Skurichina M., Duin R. P. W. (2001) Bagging and random subspace method for redundant feature spaces, in *Multiple classifier systems: second international workshop*, eds Kittler J., Roli F., Springer, 1-10.
- Tibshirani R. (1996) Regression shrinkage and selection via the lasso. *J. R. Statist. Soc. B*, 58, 267-288.
- Ting K. M., Witten I. H. (1997) Stacking bagged and dagged models, in *Proceedings of the Fourteenth National Conference on Machine Learning*.
- Tsymbal A., Pechenizkiy M., Cunningham P. (2005) Diversity in search strategies for ensemble feature selection, *Information Fusion*, 6, pp.83-98.
- Tsymbal A., Puuronen S., Patterson D. (2003) Ensemble feature selection with the simple Bayesian classification, *Information Fusion*, 4 (2), pp.87-100.
- Van de Laar P., Heskes T. (2000) Input selection based on an ensemble, *Neurocomputing*, 34, 227-238.
- Wolpert D. H. (1992) Stacked generalization, *Neural Networks*, 5, 241-259.
- Zou H., Hastie T. (2005), Regularization and variables selection via the elastic net, *J. R. Statist. Soc. B*, 67, 301-320.