# Effective Voting of Heterogeneous Classifiers

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas

Department of Informatics,
Aristotle University of Thessaloniki,
54124 Thessaloniki, Greece
{greg,katak,vlahavas}@csd.auth.gr

**Abstract.** This paper deals with the combination of classification models that have been derived from running different (heterogeneous) learning algorithms on the same data set. We focus on the Classifier Evaluation and Selection (ES) method, that evaluates each of the models (typically using 10-fold cross-validation) and selects the best one. We examine the performance of this method in comparison with the Oracle selecting the best classifier for the test set and show that 10-fold cross-validation has problems in detecting the best classifier. We then extend ES by applying a statistical test to the 10-fold accuracies of the models and combining through voting the most significant ones. Experimental results show that the proposed method, Effective Voting, performs comparably with the state-of-the-art method of Stacking with Multi-Response Model Trees without the additional computational cost of meta-training.

## 1 Introduction

A very active research area during the last years is the one involving methodologies and systems for the combination of multiple predictive models. It has attracted scientists from the fields of Statistics, Machine Learning, Pattern Recognition and Knowledge Discovery aiming at improving the predictive accuracy of a single classification or regression model. Within the Machine Learning community this area is commonly referred to as Ensemble Methods [6].

Models that have been derived from different executions of the same learning algorithm are often called Homogeneous. Such models can be induced by injecting randomness into the learning algorithm or through the manipulation of the training instances, the input attributes and the model outputs [7]. Homogeneous models are typically combined through weighted or unweighted voting. Models that have been derived from running different learning algorithms on the same data set are often called Heterogeneous.

This paper deals with the combination of Heterogeneous Classification Models and focuses on the Classifier Evaluation and Selection (ES) method. This method evaluates each of the models (typically using 10-fold cross-validation) and selects the best one. We examine the performance of ES in comparison with the Oracle selecting the best classifier for the test set and show that 10-fold cross-validation has problems in detecting the best classifier. We then extend

J.-F. Boulicaut et al. (Eds.): ECML 2004, LNAI 3201, pp. 465–476, 2004.

ES by applying a statistical test to the 10-fold accuracies of the models and combining through voting the most significant ones in an attempt to alleviate this problem. Extensive experimental results show that the proposed method, Effective Voting, performs comparably with the state-of-the-art Heterogeneous Ensemble Method of Stacking with Multi-Response Model Trees [8] without the additional computational cost of meta-training.

The rest of this paper is organized as follows. The next section presents related work on combining Heterogeneous Classification Models. Section 3 describes in detail the proposed approach. Section 4 presents the experimental methodology and Section 5 the results and observations. Finally, Section 6 summarizes this paper and discusses issues for further investigation.

## 2 Combining Heterogeneous Classification Models

Unweighted and Weighted Voting are two of the simplest methods for combining not only Heterogeneous but also Homogeneous models. In Voting, each model outputs a class value (or ranking, or probability distribution) and the class with the most votes (or the highest average ranking, or average probability) is the one proposed by the ensemble. Note that this type of Voting is in fact called Plurality Voting, in contrast to the frequently used term Majority Voting, as the latter formally implies that at least 50% (the majority) of the votes should belong to the winning class. In Weighted Voting, the classification models are not treated equally. Each model is associated with a coefficient (weight), usually proportional to its classification accuracy.

Another simple method is Evaluation and Selection. Each of the classification models is evaluated (typically using 10-fold cross-validation) on the training set and the best one is selected for application on the test set. In [10, 20], the accuracy of the models is estimated locally on the different examples that surround each test example. Such approaches belong to the family of Dynamic Classifier Selection introduced in [11], which was the first work discussing the idea of using a different function for classifier combination in different partitions of the training set.

Stacked Generalization [19], also known as Stacking in the literature, is a method that combines multiple classifiers by learning a meta-level model that predicts the correct class based on the decisions of the classifiers. This model is induced on a set of meta-level training data that are typically produced by a process similar to applying $k$-fold cross-validation on the training data. Specifically, $k$-1 folds are used for training the classifiers and one for recording their decisions along with the true class. This leads to a meta-level training data set of equal size to the original training data set. A meta-classifier is then induced from these data. When a new instance appears for classification, the output of all classifiers is first calculated and then propagated to the meta-level model, which outputs the final result. A recent study [9] has shown that Stacking with Multi-Response Model Trees [8] as the meta-level learning algorithm, is the most accurate heterogeneous classifier combining method of the Stacking family.

## 3 Effective Voting of Heterogeneous Classifiers

This paper presents a simple, yet effective extension to Classifier Evaluation and Selection. On top of the 10-fold cross-validation for the evaluation of the models we apply a paired t-test with a significance level of 0.05 for each pair of models to evaluate the statistical significance of their relative performance. We then combine the most significant ones through 3 different strategies. The whole process of the proposed method, called Effective Voting, is described in the following paragraphs.

Consider a set of classification models $c_i$, $i$=1..$N$. For each pair of models $i$,$j$ we initially perform the paired t-test:

$$test(c_i, c_j) = \begin{cases} 1 & \text{if } c_i \text{ significantly better than } c_j \\ -1 & \text{if } c_j \text{ significantly better than } c_i \\ 0 & \text{otherwise} \end{cases}$$

Then for each model we calculate the overall significance index:

$$Sig(c_i) = \sum_{j=1}^{N} test(c_i, c_j)$$

Finally we try the following 3 strategies:

- EV$_1$: We select the model(s) with the highest significance index and combine their decisions through Weighted Voting (if more than one).
- EV$_2$: We select the model with the lowest error rate along with any others that are not significantly worse than this and combine their decisions through Weighted Voting (if more than one).
- EV$_3$: We select the three models with the highest significance index and combine their decisions through Voting. If there are ties, we brake them by selecting the most accurate ones.

Note that all of the above proposed strategies aim at selecting the most significantly accurate models, yet in a different way. The first gives priority to models with the highest statistical significance index taking into account their accuracies through the Weighted Voting process. The second gives priority to the most accurate model but also considers those that are not significantly worse than the best one. The last strategy always chooses the three models with the highest statistical significance index and combines them through simple Voting. However, it also takes into account the accuracy of the models, by selecting the most accurate ones when there are draws.

Table 1 exemplifies the operation of the three different strategies, based on a sample of the error rates and significance tests of 10 models from the experimental results. For each model in each row, the first 10 columns show the result of the paired t-test. The next two columns show the significance index and the average error rate. The last 3 columns show which models are selected by each strategy.

**Table 1.** Example of the operation of the three Effective Voting strategies

| | $c_1$ | $c_2$ | $c_3$ | $c_4$ | $c_5$ | $c_6$ | $c_7$ | $c_8$ | $c_9$ | $c_{10}$ | *Sig* | Err. (%) | EV$_1$ | EV$_2$ | EV$_3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $c_1$ | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 23.53 | | √ | |
| $c_2$ | -1 | 0 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -9 | 71.08 | | | |
| $c_3$ | -1 | 1 | 0 | 0 | 0 | -1 | 0 | 0 | -1 | -1 | -3 | 28.44 | | | |
| $c_4$ | -1 | 1 | 0 | 0 | -1 | -1 | 0 | 0 | -1 | -1 | -4 | 29.42 | | | |
| $c_5$ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 24.51 | | √ | |
| $c_6$ | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 5 | 20.59 | √ | √ | √ |
| $c_7$ | -1 | 1 | 0 | 0 | 0 | -1 | 0 | 0 | -1 | -1 | -3 | 28.93 | | | |
| $c_8$ | 0 | 1 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 27.97 | | | |
| $c_9$ | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 22.55 | | √ | √ |
| $c_{10}$ | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 21.57 | | √ | √ |

We notice that EV$_1$ selects just one model, $c_6$, as it is the only one with the highest significance index, equal to 5. It is also the model with the lowest error rate. However, if we look at the line of $c_6$, we will see that models $c_1$, $c_5$, $c_9$ and $c_{10}$ are not significantly worse. Therefore the decisions of all these models along with $c_5$ are combined by EV$_2$ through Weighted Voting. Finally EV$_3$ selects $c_6$ and then has to choose another two models from $c_1$, $c_9$ and $c_{10}$ as they tie with a significant index of 4. From these it drops $c_1$, as it has the highest error rate and combines $c_6$ $c_9$ and $c_{10}$ with Voting.

Effective Voting aims to stand in between methods that combine all models, such as Voting, Weighted Voting and Stacking and methods that just select a single model, like Evaluation and Selection. The former category of methods has the advantage of error correction through the contribution of different biases but also has the disadvantage of letting some models with potentially inferior predictive performance participate in the combination process. On the other hand the latter category excludes all models but one, which might not always be the most accurate one. Effective Voting attempts to first select the most significant models with the aid of statistical tests and then combine them through a voting process. It can therefore be considered as a pre-processing method, rather than an actual combination method.

## 4 Experimental Setup

This section provides information on the data sets, combining methods, participating algorithms and evaluation methodology that were used for the experiments. The WEKA machine learning software [18] was used as the platform for all the experiments.

### 4.1 Data Sets

The predictive performance of the combining methods was evaluated on 40 data sets from the UCI Machine Learning repository [3]. Table 2 presents the details

**Table 2.** Details of the data sets used in the experiments: Folder in UCI server, number of instances, classes, continuous and discrete attributes, (%) percentage of missing values

| UCI Folder | Inst | Cls | Cnt | Dsc | MV |
|---|---|---|---|---|---|
| annealing | 898 | 6 | 6 | 32 | 64.98 |
| audiology | 226 | 24 | 0 | 69 | 2.03 |
| autos | 205 | 7 | 15 | 10 | 1.15 |
| balance-scale | 625 | 3 | 4 | 0 | 0.00 |
| breast-cancer | 286 | 2 | 0 | 9 | 0.35 |
| breast-cancer-wisconsin | 699 | 2 | 9 | 0 | 0.25 |
| car | 1728 | 4 | 0 | 6 | 0.00 |
| chess (kr-vs-kp) | 3196 | 2 | 0 | 36 | 0.00 |
| cmc | 1473 | 3 | 2 | 7 | 0.00 |
| dermatology | 366 | 6 | 1 | 33 | 0.01 |
| ecoli | 336 | 8 | 7 | 0 | 0.00 |
| glass | 214 | 7 | 9 | 0 | 0.00 |
| heart-disease (cleveland) | 303 | 5 | 6 | 7 | 0.18 |
| heart-disease (hungary) | 294 | 5 | 6 | 7 | 20.46 |
| heart-disease (switzerland) | 123 | 5 | 6 | 7 | 17.07 |
| heart-disease (va) | 200 | 5 | 6 | 7 | 26.85 |
| hepatitis | 155 | 2 | 6 | 13 | 5.67 |
| horse-colic | 368 | 2 | 7 | 15 | 23.80 |
| image | 2310 | 7 | 19 | 0 | 0.00 |
| ionosphere | 351 | 2 | 34 | 0 | 0.00 |
| iris | 150 | 3 | 4 | 0 | 0.00 |
| labor | 57 | 2 | 8 | 8 | 35.75 |
| lymphography | 148 | 4 | 3 | 15 | 0.00 |
| pima-indians-diabetes | 768 | 2 | 8 | 0 | 0.00 |
| primary-tumor | 339 | 22 | 0 | 17 | 3.90 |
| soybean | 683 | 19 | 0 | 35 | 9.78 |
| statlog (australian) | 690 | 2 | 6 | 9 | 0.65 |
| statlog (german) | 1000 | 2 | 7 | 13 | 0.00 |
| statlog (heart) | 270 | 2 | 13 | 0 | 0.00 |
| statlog (satimage) | 6435 | 6 | 36 | 0 | 0.00 |
| statlog (segment) | 2310 | 7 | 19 | 0 | 0.00 |
| statlog (vehicle) | 846 | 4 | 18 | 0 | 0.00 |
| thyroid-disease | 3772 | 4 | 7 | 22 | 5.54 |
| tic-tac-toe | 958 | 2 | 0 | 9 | 0.00 |
| undocumented (sonar) | 208 | 2 | 60 | 0 | 0.00 |
| undocumented (vowel-context) | 990 | 11 | 10 | 3 | 0.00 |
| voting-records | 435 | 2 | 0 | 16 | 5.63 |
| waveform | 5000 | 3 | 40 | 0 | 0.00 |
| wine | 178 | 3 | 13 | 0 | 0.00 |
| zoo | 101 | 7 | 1 | 16 | 0.00 |

of the data sets (Folder in UCI server, number of instances, classes, continuous and discrete attributes, (%) percentage of missing values).

### 4.2 Combining Methods and Participating Algorithms

We compared the following classifier combining methods that were described in Sections 2 and 3: Stacking with Multi-Response Model Trees (SMT), Voting (V), Weighted Voting (WV), Effective Voting with the 3 different strategies (EV) and Evaluation and Selection (ES).

The comparison also includes the Oracle classifier selection method (ORA) that a-posteriori selects the classification algorithm with the highest accuracy on the test-set. This is not an actual combining method, but it is used to evaluate the ability of (ES) to select the best classifier on the test set. The predictive performance of ORA is the best performance that ES could achieve, if it always selected the best classifier for the test-set.

All the above methods are used in conjunction with the WEKA implementations of the following 10 base-level classification algorithms, which are run with default parameter values unless otherwise stated:

- DT: the decision table algorithm of [13].
- JRip: the RIPPER rule learning algorithm [5].
- PART: the PART rule learning algorithm [17].
- J48: the decision tree learning algorithm C4.5 [15].
- IBk: the $k$ nearest neighbor algorithm [1].
- K*: an instance-based learning algorithm with entropic distance measure [4].
- NB: the Naive Bayes algorithm [12] using the kernel density estimator rather than assume normal distributions for numeric attributes.
- SMO: the sequential minimal optimization algorithm for training a support vector classifier using polynomial kernels [14].
- RBF: an algorithm for training a radial basis function network [2].
- MLR: the multi-response linear regression algorithm, as used in [16].

The meta-level training data for Stacking are produced using 10-fold stratified cross-validation on the training set. The same procedure is used for estimating the accuracy of the above base-level algorithms and their significance index through the paired t-tests. All combining methods operate on probability distributions of the base-level classifiers.

### 4.3 Evaluation Methodology

For the evaluation of the combining methods we perform a 10-fold stratified cross-validation experiment. In each of the 10 repetitions, the same 9 folds are used for training the different combining methods and 1 fold for evaluating their performance. The error rates are averaged over the 10 folds in order to obtain the average error $err_m(d_i)$ of each method $m$ in each data set $d_i$.

A first indicator for pairwise comparisons, is the geometric mean of the average error of each method over all the data sets. Comparing the geometric means of each method we can get a general impression of the relationship between the methods. If the geometric mean of a method $m_1$ is greater than another method $m_2$ then this implies that method $m_1$ performs worse than method $m_2$, and vice versa. The ratio of their means gives a measure of how much better/worse is their relative performance.

In addition, we apply a paired t-test to the errors of the methods in the 10 folds of the cross-validation experiment with a significance level of 0.05. From the outcome of this test on all data sets we report the statistically significant wins and losses for each pair of methods.

## 5 Results and Discussion

Table 3 shows the significant wins and losses for each pair of combining methods on all data sets. Table 4, shows the average 10-fold cross-validation error rate of the combining methods for each data set. The geometric mean is presented in the last line.

**Table 3.** A symmetric matrix that presents the significant wins and losses (w:l) for each pair of combining methods

| | SMT | V | WV | $EV_1$ | $EV_2$ | $EV_3$ | ES | ORA |
|---|---|---|---|---|---|---|---|---|
| SMT | | 7:1 | 7:1 | 5:2 | 3:3 | 3:4 | 8:2 | 3:18 |
| V | 1:7 | | 0:1 | 2:5 | 0:7 | 0:6 | 5:6 | 0:23 |
| WV | 1:7 | 1:0 | | 2:5 | 0:4 | 0:5 | 5:6 | 0:22 |
| $EV_1$ | 2:5 | 5:2 | 5:2 | | 0:1 | 1:4 | 0:0 | 0:31 |
| $EV_2$ | 3:3 | 7:0 | 4:0 | 1:0 | | 1:1 | 3:1 | 0:17 |
| $EV_3$ | 4:3 | 6:0 | 5:0 | 4:1 | 1:1 | | 3:2 | 0:23 |
| ES | 2:8 | 6:5 | 6:5 | 0:0 | 1:3 | 2:3 | | 0:29 |
| ORA | 18:3 | 23:0 | 22:0 | 31:0 | 17:0 | 23:0 | 29:0 | |

### 5.1 The Performance of Evaluation and Selection

The results show that ORA achieves by far the significantly best performance compared to the rest. However, ORA is a control method for comparison purposes. The ES method which should approximate it, actually performs much worse. This reveals the fact that the average accuracy of a classification algorithm based on a 10-fold cross-validation experiment on a training set is often an unsuccessful indicator of the algorithm's accuracy on a test set.

A more profound look into which algorithms are selected as best by ES against the true best as selected by ORA is given by the confusion matrix of Table 5. The total number of selections in the last column, shows how many times the algorithm of each row was selected by ES. The total number of selections at the bottom of the table, shows how many times the algorithm of each column was truly best. The number in row $i$ and column $j$ represents the number of times that the algorithm of row $i$ was found to be the best by ES while the true best was the algorithm of column $j$. Note that the total number of selections is 400 (40 data sets * 10 folds of the cross-validation).

**Table 4.** Folder in UCI server, average error rate of each combining method on each of the 40 data sets and geometric mean of each combining method over all data sets

| UCI Folder | SMT | V | WV | EV$_1$ | EV$_2$ | EV$_3$ | ES | ORA |
|---|---|---|---|---|---|---|---|---|
| annealing | 1.56 | 2.78 | 2.67 | 1.67 | 1.67 | 2.00 | 1.67 | 1.67 |
| audiology | 20.34 | 16.80 | 17.69 | 19.49 | 17.31 | 18.16 | 20.36 | 14.60 |
| autos | 15.48 | 16.02 | 16.00 | 19.83 | 18.40 | 16.45 | 21.76 | 14.57 |
| balance-scale | 5.44 | 10.24 | 9.92 | 8.00 | 8.32 | 9.91 | 8.00 | 8.00 |
| breast-cancer | 24.47 | 24.11 | 24.47 | 26.88 | 23.41 | 25.91 | 29.01 | 19.94 |
| breast-cancer-wisconsin | 3.01 | 3.00 | 3.00 | 2.86 | 3.00 | 3.00 | 2.72 | 2.00 |
| car | 1.74 | 5.09 | 4.92 | 4.75 | 4.74 | 4.46 | 5.32 | 3.88 |
| chess (kr-vs-kp) | 0.63 | 0.78 | 0.72 | 0.56 | 0.47 | 0.59 | 0.59 | 0.47 |
| cmc | 45.15 | 47.05 | 46.98 | 48.21 | 46.37 | 47.12 | 48.68 | 44.20 |
| dermatology | 1.64 | 1.91 | 2.18 | 3.00 | 2.46 | 2.46 | 2.73 | 0.82 |
| ecoli | 14.31 | 12.83 | 13.13 | 13.39 | 13.71 | 13.40 | 12.79 | 11.31 |
| glass | 22.40 | 23.79 | 23.31 | 23.81 | 23.29 | 24.29 | 27.51 | 17.77 |
| heart-disease (cleveland) | 16.46 | 18.19 | 18.19 | 15.84 | 17.19 | 14.85 | 16.81 | 11.57 |
| heart-disease (hungary) | 16.34 | 18.01 | 17.68 | 14.63 | 14.97 | 14.61 | 13.95 | 11.89 |
| heart-disease (switzerland) | 65.71 | 58.21 | 59.74 | 63.21 | 60.64 | 63.14 | 61.67 | 47.63 |
| heart-disease (va) | 68.50 | 66.50 | 68.00 | 69.50 | 66.00 | 64.00 | 64.50 | 57.50 |
| hepatitis | 19.46 | 14.88 | 14.88 | 14.25 | 14.88 | 14.25 | 14.21 | 8.38 |
| horse-colic | 16.29 | 14.67 | 14.67 | 16.00 | 16.82 | 15.46 | 16.00 | 12.21 |
| image | 2.08 | 1.90 | 1.86 | 2.21 | 1.90 | 1.65 | 2.25 | 1.73 |
| ionosphere | 7.12 | 7.68 | 7.97 | 8.83 | 7.12 | 5.13 | 7.99 | 4.28 |
| iris | 6.00 | 4.67 | 4.67 | 4.67 | 4.67 | 5.33 | 3.33 | 2.00 |
| labor | 7.00 | 5.33 | 5.33 | 8.33 | 5.33 | 7.00 | 10.33 | 0.01 |
| lymphography | 21.52 | 14.24 | 14.24 | 18.90 | 14.24 | 12.90 | 14.81 | 8.14 |
| pima-indians-diabetes | 22.91 | 23.30 | 23.04 | 22.91 | 23.04 | 23.17 | 23.56 | 20.83 |
| primary-tumor | 58.38 | 53.96 | 53.96 | 51.92 | 51.91 | 49.26 | 52.80 | 47.49 |
| soybean | 6.59 | 7.02 | 7.02 | 6.59 | 6.73 | 7.02 | 6.30 | 4.39 |
| statlog (australian) | 14.35 | 13.19 | 13.19 | 14.35 | 13.91 | 13.62 | 14.78 | 11.74 |
| statlog (german) | 22.80 | 24.00 | 23.90 | 25.00 | 24.60 | 24.50 | 25.20 | 22.40 |
| statlog (heart) | 14.81 | 15.93 | 16.30 | 15.19 | 15.19 | 15.56 | 15.19 | 11.85 |
| statlog (satimage) | 7.79 | 8.38 | 8.27 | 9.20 | 8.80 | 8.50 | 9.71 | 8.78 |
| statlog (segment) | 1.82 | 1.52 | 1.47 | 2.12 | 1.90 | 1.77 | 2.25 | 1.69 |
| statlog (vehicle) | 20.33 | 25.53 | 25.30 | 23.76 | 23.88 | 23.40 | 24.95 | 21.63 |
| thyroid-disease | 0.40 | 2.78 | 2.41 | 0.48 | 0.42 | 0.37 | 0.58 | 0.32 |
| tic-tac-toe | 0.83 | 3.13 | 3.13 | 1.56 | 1.67 | 1.67 | 0.94 | 0.94 |
| undocumented (sonar) | 17.79 | 15.90 | 15.43 | 14.31 | 11.50 | 11.50 | 15.29 | 8.62 |
| undocumented (vowel-context) | 0.91 | 1.21 | 0.71 | 0.71 | 0.71 | 0.71 | 0.71 | 0.51 |
| voting-records | 3.91 | 4.37 | 4.37 | 3.91 | 3.92 | 4.14 | 4.13 | 2.77 |
| waveform | 13.58 | 16.30 | 16.14 | 13.58 | 13.58 | 13.92 | 13.42 | 13.24 |
| wine | 1.70 | 0.56 | 0.56 | 2.78 | 1.11 | 1.70 | 2.78 | 0.01 |
| zoo | 5.91 | 2.91 | 2.91 | 6.00 | 2.91 | 5.00 | 7.00 | 2.00 |
| geometric mean | 7.93 | 8.60 | 8.45 | 8.58 | 7.81 | 8.00 | 8.57 | 4.59 |

**Table 5.** Confusion matrix: number of times each algorithm was selected by ES against true times being best

| | DT | IBk | J48 | JRip | K* | MLR | NB | PART | RBF | SMO | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|
| DT | 11 | 2 | 1 | 0 | 2 | 2 | 2 | 2 | 0 | 0 | 22 |
| IBk | 2 | 9 | 1 | 4 | 9 | 0 | 1 | 1 | 0 | 6 | 33 |
| J48 | 5 | 0 | 7 | 5 | 2 | 3 | 2 | 5 | 1 | 3 | 33 |
| JRip | 4 | 6 | 9 | 21 | 8 | 5 | 4 | 9 | 1 | 2 | 69 |
| K* | 0 | 5 | 2 | 7 | 4 | 1 | 4 | 2 | 0 | 6 | 31 |
| MLR | 1 | 2 | 2 | 7 | 3 | 9 | 13 | 5 | 1 | 7 | 50 |
| NB | 5 | 7 | 8 | 6 | 4 | 10 | 18 | 3 | 1 | 11 | 73 |
| PART | 1 | 4 | 4 | 3 | 0 | 1 | 0 | 3 | 0 | 3 | 19 |
| RBF | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 2 |
| SMO | 5 | 5 | 5 | 4 | 4 | 13 | 7 | 5 | 0 | 20 | 68 |
| Total | 34 | 40 | 39 | 57 | 36 | 44 | 52 | 35 | 5 | 58 | |
| Ratio | 0.32 | 0.23 | 0.18 | 0.37 | 0.11 | 0.16 | 0.35 | 0.09 | 0.20 | 0.34 | |

We firstly notice that RBF is only 5 times the best classifier out of the 400. This shows that in general it is not a very accurate classification learning algorithm with the default parameters as used in the experiments. The most accurate classifiers are JRip, NB and SMO. These are also the algorithms that are most of the times correctly identified by ES as it is given by the ratio of correct selections by ES over the actual selections by ORA, in the last line of Table 5. PART and K* are two algorithms that although together they are 71 times the best algorithms, yet ES only correctly identifies them 7 times.

### 5.2 The Performance of Effective Voting

The results in Table 3 show that the proposed approach manages to alleviate the problem of predicting the most accurate classification algorithm on a test set based on the average accuracies of a 10-fold cross-validation experiment on the training set. Strategy $EV_1$ is comparable to ES and has a lower geometric mean error. More importantly, strategies $EV_2$ and $EV_3$ have respectively two and one significant wins more than ES and a lower geometric mean error.

In addition, $EV_2$ is comparable to the state-of-the-art SMT method with a lower geometric mean error and $EV_3$ has one significant win more but a little higher geometric mean error. Moreover, Effective Voting is much faster than SMT as it does not involve any computational cost for meta-training. In fact, SMT is a very complex method since not only does it require meta-training, but it also learns one model for each class at the meta-level. It so creates a second Ensemble of homogeneous models at the meta-level, which in part justifies its increased performance.

It is interesting to mention the actual data sets where the significant wins/losses of the Effective Voting variants occur compared to the state-of-the-art SMT method. SMT is significantly better than all three methods in the data sets of *balance-scale* and *car* (which is in accordance with the results of [9]). In addition, in the *cmc* domain it wins all but $EV_2$ and in *statlog-satimage* all but $EV_3$. All variants of EV win SMT in the *primary-tumor* domain. In addition $EV_1$ wins SMT in the *hepatitis* domain, $EV_2$ in the *hepatitis* and *sonar* domains and $EV_3$ in the *lymphography*, *ionosphere* and *sonar* domains. This shows that the three different strategies of EV actually offer different merits.

### 5.3 Evaluating the Effect of Model Removal

In this part we evaluate the effect that the removal of a model has on the performance of the methods. We will remove in turn the different models, replacing them back to the ensemble before removing the next. The geometric mean error of the combining methods is presented in Table 6. Each column corresponds to one run and the header of the columns states the classification algorithm not used.

**Table 6.** Geometric mean error of combining methods with different ensemble compositions

| | (ALL) | DT | IBk | J48 | JRIP | K* | MLR | NB | PART | RBF | SMO |
|---|---|---|---|---|---|---|---|---|---|---|---|
| SMT | 7.93 | 7.91 | 7.78 | 7.85 | 7.82 | 7.88 | 8.23 | 7.95 | 7.90 | 7.92 | 7.88 |
| V | 8.60 | 8.69 | 8.85 | 8.92 | 8.86 | 9.04 | 8.47 | 8.65 | 9.03 | 8.50 | 8.77 |
| WV | 8.45 | 8.47 | 8.63 | 8.48 | 8.74 | 8.78 | 8.27 | 8.40 | 8.70 | 8.28 | 8.43 |
| $EV_1$ | 8.58 | 8.56 | 8.71 | 8.57 | 8.52 | 8.45 | 8.47 | 8.47 | 8.59 | 8.52 | 8.55 |
| $EV_2$ | 7.81 | 7.81 | 7.69 | 7.84 | 7.98 | 7.92 | 7.89 | 8.01 | 7.93 | 7.84 | 7.76 |
| $EV_3$ | 8.00 | 8.03 | 8.26 | 8.12 | 8.33 | 8.46 | 8.20 | 8.26 | 8.08 | 8.06 | 8.12 |
| ES | 8.57 | 8.88 | 8.62 | 8.59 | 8.81 | 8.48 | 8.71 | 8.79 | 8.74 | 8.74 | 8.70 |
| ORA | 4.59 | 5.34 | 4.77 | 4.73 | 4.75 | 4.68 | 4.64 | 5.38 | 4.72 | 4.56 | 4.66 |

The results show that from the three proposed strategies for Effective Voting, the most competitive one is $EV_2$. For all different model compositions of the ensemble it has a much lower geometric error mean. In addition the geometric mean of $EV_2$ is comparable with that of SMT.

## 6 Conclusions and Future Work

This paper has focused on the method of Classifier Evaluation and Selection (ES). It demonstrated that one of the weaknesses of this method derives from the inaccurate evaluation of the models' accuracies. To remedy this problem we proposed an extension to ES that makes use of statistical significance tests in order to select and combine the most accurate models. Through a large and thorough experimental study we showed that the proposed method, Effective Voting, is comparable in accuracy to recent state-of-the-art heterogeneous classifier combining methods, such as Stacking with Multi-Response Model Trees having at the same time reduced computational cost.

As a general conclusion, we believe that it is worth researching more into advancing simple heterogeneous ensemble methods such as evaluation and selection instead of complex methods that require a lot more computational cost. This conclusion is reinforced by the very good performance of the ORA method. On the other hand, ORA was significantly worse than SMT in the domains of *balance-scale* and *car*, which shows that in some cases more complex methods are definitely required to achieve better performance.

For future work, we intend to research into alternative methods for classifier performance evaluation, in order to further improve the Evaluation and Selection framework. In addition we intend to investigate the applicability of the proposed ideas to more complex Classifier Evaluation methods such as those that are based on local accuracy estimates.

## References

1. D. Aha, D.W. Kibler, and M.K. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.
2. C.M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.
3. C.L. Blake and C.J. Merz. UCI repository of machine learning databases. http://www.ics.uci.edu/∼mlearn/MLRepository.html.
4. J.G. Cleary and L.E. Trigg. K*: An instance-based learner using an entropic distance measure. In *Proceedings of the 12th International Conference on Machine Learning*, pages 108–114. Morgan Kaufmann, 1995.
5. W.W. Cohen. Fast effective rule induction. In *Proceedings of the 12th International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.
6. T. G. Dietterich. Machine-learning research: Four current directions. *AI Magazine*, 18(4):97–136, 1997.
7. T. G. Dietterich. Ensemble Methods in Machine Learning. In *Proceedings of the 1st International Workshop in Multiple Classifier Systems*, pages 1–15, 2000.
8. S. Dzeroski and B. Zenko. Stacking with multi-response model trees. In *Proceedings of the 3d International Workshop in Multiple Classifier Systems*. Springer, 2002.
9. S. Dzeroski and B. Zenko. Is Combining Classifiers with Stacking Better than Selecting the Best One? *Machine Learning*, 54:255–273, 2004.
10. G. Giacinto and F. Roli. Adaptive selection of image classifiers. In *Proceedings of the 9th International Conference on Image Analysis and Processing*, pages 38–45, 1997.
11. T.K. Ho, J.J. Hull, and S.N. Srihari. Decision combination in multiple classifier systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(1):66–75, 1994.
12. G.H. John and P. Langley. Estimating continuous distributions in bayesian classifiers. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Francisco, 1995. Morgan Kaufmann.
13. R. Kohavi. The power of decision tables. In *Proceedings of the 12th European Conference on Machine Learning*, pages 174–189, 1995.
14. J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Scholkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1998.

15. R.J. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufman, San Mateo, 1993.
16. K.M. Ting and I.H. Witten. Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289, 1999.
17. I.H. Witten and E. Frank. Generating accurate rule sets without global optimization. In *Proceedings of the 15th International Conference on Machine Learning, ICML98*, pages 144–151, 1998.
18. I.H. Witten and E. Frank. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, 1999.
19. D. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.
20. K. Woods, W. P. Kegelmeyer, and K. Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE Transanctions on Pattern Analysis and Machine Intelligence*, 19(4):405–410, 1997.