Random subspacing for regression ensembles

Niall Rooney[1], David Patterson[1], Alexey Tsymbal[2], Sarab Anand[1]
[1]Northern Ireland Knowledge Engineering Laboratory (NIKEL)
University of Ulster
[2]Department of Computer Science, Trinity College Dublin
{nf.rooney,wd.patterson,ss.anand}@ulster.ac.uk, Alexey.Tsymbal@cs.tcd.ie

Abstract

In this work we present a novel approach to ensemble learning for regression models, by combining the ensemble generation technique of random subspace method with the ensemble integration methods of Stacked Regression and Dynamic Selection. We show that for simple regression methods such as global linear regression and nearest neighbours, this is a more effective method than the popular ensemble methods of Bagging and Boosting. We demonstrate that the approach can be effective even when the ensemble size is small.

Introduction

Regression is a classical learning problem, the goal of which is to build a learning model from training data that predicts the values of a continuous target variable of test instances, where both the training and test instances are drawn from the same population. Each instance consists of a target variable together with a number of numeric or categorical variables which may act as predictors to the target variable.

The purpose of ensemble learning is to build a learning model which integrates a number of diverse base learning models, so that the model gives better generalization performance on application to a particular data set than any of the individual base models. A popular theoretical consideration of the generalization error of a learning method is based on the bias-variance decomposition of the expected error. Informally the bias is a measure of how closely the model's average prediction, measured over all possible training sets of fixed size, matches the true prediction of a target instance. Variance is a measure of how the models' predictions will vary from the average prediction over all possible training sets of fixed size.

Ensemble learning consists of two problems; *ensemble generation*: how does one generate the base models? and *ensemble integration*: how does one integrate the base models' predictions to improve performance*?* Ensemble generation can be characterized as being *homogeneous* if each base learning model uses the same learning algorithm or *heterogeneous* if the base models can be built from a range of learning algorithms. More formally, an ensemble consists of a set of base models, $h_1..h_N$ where N is the size of the ensemble and each base model in the ensemble uses training instances from the corresponding training set $T_i, i = 1..N$. Ensemble integration can be addressed by either one of two mechanisms, either the predictions of the base models are combined in some fashion during the application phase to give an ensemble prediction (*combination approach*) or the prediction of one base model is selected according to some criteria to form the final prediction (*selection approach*). Some ensemble learning algorithms, such as Boosting, define both how ensembles are generated and how the base models are integrated. Theoretical and empirical work has shown the ensemble technique to be effective with the proviso that the base models are diverse (where the diversity is measured by the degree of correlation between their training errors) and accurate (Dietterich 00). These measures are however not necessarily independent of each other. If the prediction error of all base models is very low, then their learning hypothesis must be very similar to the true function underlying the data, and hence they must of necessity, be similar to each other i.e. they are unlikely to be diverse. In essence then there is often a trade-off between diversity and accuracy (Christensen 03).

There has been much empirical work on ensemble learning for regression in the context of neural networks, however there has been little research carried out in terms of using homogeneous ensemble techniques to improve the performance of simple regression algorithms. In this paper we look at improving the generalization performance of nearest neighbours (NN) and least-squares linear regression (LR). These methods were chosen as they are simple models with different approaches to learning and whereas linear regression is an eager learner which tries to approximate the true function by a globally linear function, k-nearest neighbours is a lazy learner which tries to approximate the true function locally.

Ensemble Generation

Ensemble generation for homogeneous learning is generally addressed by using different samples of the training data for each base model (this is intrinsic to bagging and boosting techniques as will be described later) or if the learning method has a set of learning parameters, adjusting them to have different values for each of the base models (for example in the case of neural networks initializing the base models with different random weights).

An alternative approach for ensemble generation for homogeneous learning is the method of Random Subspacing was first proposed by Ho (Ho 98a, Ho 98b) for classification problems. Random subspace method (RSM) is a generation method where each base model is built from the training data which has been transformed to contain different random subsets of the variables. The RSM as proposed by Ho used a method of majority voting to combine the classifications of the base classifiers. Of course, such as the case with ensemble generation via sampling data, the ensemble generation of random subspacing can be combined with a variety of more sophisticated ensemble integration techniques. Ho has shown RSM to be both effective for unstable learners such as decision trees and for nearest neighbours if the data set size is small relative to its dimensionality. Skurichina et al. (Skurichina et al. 2003) have shown the conditions of training set size and data dimensionality where bagging, boosting and RSM may be effective for a range of linear classifiers. Tsymbal et al. (Tsymbal et al. 03) investigated the effect of RSM with more sophisticated ensemble generation techniques than majority voting.

Ensemble Integration

The initial approaches to ensemble *combination* for regression were based on the linear combination of the base models according to the function:

$$\sum_{i=1}^{n} \alpha_i f_i(x)$$

The simplest approach to determining the values of α_i is to set them to the same value. This is known as the Base Ensemble Method (BEM). Merz and Pazzani (Merz and Pazzani 99) provide an extensive description of more advanced techniques for determining the values of alpha.

The generic approach of Stacking was introduced by Wolpert (Wolpert 92), and was shown theoretically by LeBlanc and Tibshirani (LeBlanc and Tibshirani 92) to be a bias reducing technique. In the approach of Stacked Regression (SR), the base models produce meta-instances consisting of the target value and the base models' predictions, created by running a cross validation over the training data. The meta-data is used to build a meta-model, based on a learning algorithm and the base models are built using the whole training data. Ensemble prediction is made by a 2-stage process. The test instance is passed to the base models whose output is composed as a meta-instance. The meta-instance is passed to the meta-model which makes the final prediction. Typically the generation of the base models is heterogeneous or homogeneous but built with different training parameters. Breiman (Breiman 96a) investigated the use of linear regression to form the meta-model and found that linear regression is a suitable meta-model so long as the coefficients of regression are constrained to be non-negative.

Model *selection* simply chooses the best "base" model to make a prediction. This can be either done in a static fashion using cross-validation majority (Schaffer 93) where the best model is the one that has the lowest training error. Alternatively it can be done in a dynamic fashion (Merz 96, Puuronen et al. 99) where based on finding "close" instances in the training data to a test instance, a base model is chosen which has the lowest training error. The advantage of this approach is based on the rationale that a learner may perform better than other learning models in a localised region of the instance space even if, on average over the whole instance space, it performs no better than the others. Dynamic Selection (DS) was one of a number of Dynamic Integration techniques developed by Puuronen and Tsymbal (Puuronen et al. 99, Tsymbal et al. 03) Similar to Stacking; it performs a cross-validation history to determine the errors in the models. The errors for each training instance and the instance itself form meta-data that allows a lazy learner based on weighted nearest neighbours to dynamically select which base model should make a prediction for a test instance by assessing which model had the lowest cumulative error for the neighbouring instances and selecting it to make a prediction.

Two well-known approaches for homogeneous learning are bagging and adaptive boosting (AdaBoost). Both approaches combine the problem of ensemble generation with that of integration. Bagging (bootstrap aggregation) (Breiman 96b) draws instances from the training data using bootstrap sampling with replacement to create base models. On each draw, each training instance has an equal probability of being drawn. Ensemble prediction is based in terms of classification by majority vote and in terms of regression, by averaging. AdaBoost (Freund and Schapire 96) builds its base models iteratively. In each iteration, a new training set is generated using sampling with replacement and a predictor is trained using this training set. The difference to Bagging is that the probability of instances being drawn depends on the previous training errors of the previous cycle. The technique of adaptive boosting was adapted for regression by Drucker (Drucker 97), so that the weighted median of predictions made by the base regressors is used to form the ensemble prediction. This technique is known as AdaBoost.R1. Drucker showed this technique to be effective for regression trees. In general, Boosting and Bagging have shown to be good candidates for ensemble learning when the base models are unstable i.e. a small change in the input data can lead to a very different model being built (Opitz and Maclin 99).

In this paper, we focus on ensemble comprising of simple homogenous learning models, which are generated using the feature sampling technique of RSM. We take as our ensemble learner (for model integration) the techniques of BEM, SR and DS. We compare these ensemble methods to the data sampling techniques of Bagging and AdaBoost.R1. This forms a preliminary investigation of the benefit or otherwise of using RSM to generate base models for regression problems.

Experimental Technique and Results

The techniques were assessed using 10 fold cross validation and the mean absolute error was recorded for each technique. Data sets were pre-processed to remove missing values using a mean or modal technique. The two base learners used were 5-NN and Linear Regression. The loss function for AdaBoost.R1 was chosen to be linear and the number of nearest neighbours in the meta-model for DS was chosen to be 5. We chose the model tree technique M5, which combines instance based learning with regression trees (Quinlan 1993), as the meta-model for SR. We chose this as the technique as it provides greater flexibility in the model it induces than simple linear regression.

There has been much research into the optimal size of ensembles for classification problems in the area of bagging and boosting (Opitz and Maclin 99, Kohavi 99, Dietterich 98). In general it only takes an ensemble of a few base classifiers, to reduce the generalization error, however ensemble sizes up to 200 have been investigated (Dietterich 98). Zenobi and Cunningham (Zenobi and Cunningham 01) state from probabilistic considerations that the diversity and accuracy of the ensemble will plateau between 10 and 50. As such we chose 25 as a suitable size for all the ensemble methods.

The ensemble approaches were compared to the base technique for significant difference in Mean Absolute Error (MAE) using a paired sample t-test ($p = 0.05$). If the result for MAE for an ensemble technique is shown to be significantly better than the base model, the result is shown in bold. If the MAE for an ensemble technique is significantly worse than the base model it is shown italicised and underlined. For data sets where the MAE for the ensemble was significantly less than the MAE for the base-model for a given data set, we measured the average relative percentage reduction in MAE (RRE) over those data sets. For data sets where the MAE was significantly greater than the base model, we measured the average relative percentage increase in MAE to the base model's MAE (RIE). This section shows the results on 15 data sets selected from the WEKA (Witten and Frank 99) repository. These data sets were chosen as they represent real world data. It is divided into two subsections, one where the base regressor was 5-NN and the other where it was Linear Regression.

Nearest Neighbour Base Regressor

Table 1 shows a comparison of the MAE values for the data sampling techniques to the base model 5-NN.

Data set	5-NN	Bagging	AdaBoost.R1
abalone	1.60±0.09	**1.55±0.09**	**1.55±0.09**
autohorse	12.27±3.63	**11.71±3.75**	11.75±4.04
autompg	2.46±0.37	2.39±0.37	**2.37±0.34**
autoorice	1770.91± 738.12	1758.60± 746.23	1750.38± 719.20
auto93	3.90±1.27	3.86±1.33	3.74±1.56
bodyfat	2.39±0.49	2.33±0.37	2.35±0.41
breastTumor	8.59±0.86	**8.43±0.87**	8.47±0.83
cholesterol	41.49±4.52	40.98±4.74	40.36±4.41
cloud	0.55±0.20	0.52±0.2	0.53±0.12
cpu	35.65±15.1	34.67±15.35	34.45±14.30
housing	2.97±0.64	2.92±0.60	2.93±0.68
lowbwt	387.48±93.04	378.04±82.73	**375.41±90.72**
sensory	0.61±0.06	**0.58±0.06**	0.59±0.05
servo	0.56±0.18	0.56±0.18	0.54±0.19
strike	208.64±46.04	**199.02±47.06**	**187.32±46.27**

Table 1 MAE for the base technique and data-sampling ensembles

Either Bagging or Boosting or both proved effective in reducing the error significantly for the data sets abalone, autohorse, autompg, breastTumor, lowbwt, sensory, and strike. Bagging reduced the error significantly for 5 data sets: abalone, autohorse, breastTumor, sensory, strike with a RRE value of 3.61%. Boosting reduced significantly the error of 4 data sets: abalone, autompg, lowbwt and strike with a RRE value of 5.11%.

Table 2 shows a comparison of the MAE values for the feature sampling techniques to the base model 5-NN. BEM proved the least reliable ensemble approach to error reduction. It improved the accuracy for abalone, autohorse, cloud and cpu with RRE of 9.53%, however it dramatically increased the error for the data sets lowbwt, sensory and servo with an RIE of 80.05%. DS showed a degradation in accuracy for 2 data sets, abalone and lowbwt with a RIE value of 10.7%.

Data set	5-NN	BEM	SR	DS
Abalone	1.60±0.09	**1.55±0.09**	**1.51±0.7**	*1.68±0.1*
autohorse	12.27±3.63	**10.18±3.96**	**9.35±3.34**	**8.86±3.49**
autompg	2.46±0.37	2.40±0.34	**2.05±0.45**	2.32±0.44
autoprice	1770.91± 738.12	1658.59± 625.27	1395.89± 426.73	1561.30± 542.55
auto93	3.90±1.27	3.75±1.4	3.32±1.19	4.16±1.07
bodyfat	2.39±0.49	2.48±0.38	**0.41±0.27**	**1.24±0.26**
breast-Tumor	8.59±0.86	8.28±0.64	8.01±1.02	9.09±0.71
cholesterol	41.49±4.52	39.64±4.34	40.46±5.79	42.41±4.73
cloud	0.55±0.20	**0.47±0.18**	**0.29±0.09**	**0.37±0.2**
cpu	35.65± 15.1	**31.81± 15.26**	**21.52± 7.02**	31.99± 13.77
housing	2.97±0.64	2.92±0.59	**2.19±0.28**	**2.54±0.4**
lowbwt	387.48± 93.04	*455.42± 60.16*	377.90± 81.59	*439.34± 84.97*
sensory	0.61±0.06	*1.24±0.08*	0.60±0.06	0.68±0.09
servo	0.56±0.18	*0.87±0.18*	**0.35±0.24**	**0.42±0.23**
strike	208.64± 46.04	205..37± 42.66	226.75± 34.37	**181.80± 41.15**

Table 2 MAE for 5-NN and feature-sampling ensembles

Bagging	Adaboost.R1	BEM	SR	DS
5/10/0	4/11/0	4/8/3	8/7/0	5/8/2

Table 3 Summary of significance comparison

However it improved the performance of 6 data sets autohorse, bodyfat, cloud, housing, servo and strike with a RRE value of 26.76%. SR proved the most effective technique. It never significantly reduced the accuracy of any of the data- sets and in the case of abalone, autohorse, autompg, bodyfat, cloud, cpu, housing, servo the error was reduced significantly with a RRE value of 35.01%. Comparing, the RRE values for DS and SR to the values for Bagging and Boosting, shows that when the techniques significantly reduce the error there are more substantive gains with DS and SR.

Table 3 summarises the results of the significance comparison for both the data-sampling and feature sampling techniques. The format of this table is in the *win/tie/loss* format where a *win* indicates the number of data sets for which the ensemble technique significantly reduced the MAE in comparison to the base learner, *tie* indicates the number of data sets for which there was no significant difference and *loss* indicates the number of data sets for which the MAE was significantly increased.

Linear Regression Base Regressor

Table 4 shows a comparison of the MAE values for Bagging and Boosting to the base model LR. It is clear that neither Bagging nor Boosting provided any benefit over the data sets. In fact if anything Boosting was detrimental to the performance for 5 data sets (abalone, bodyfat, housing, sensory and strike) with an RIE value of 95.99%.

Dataset	LR	Bagging	AdaBoost.R1
abalone	1.58±0.08	1.58±0.08	*2.49±0.18*
autohorse	7.30±1.79	7.80±4.17	7.15±3.44
autompg	2.27±0.34	2.25±0.21	2.47±0.24
autoprice	1919.90± 363.34	1936.25± 343.42	2065.06± 435.41
auto93	3.79±1.3	3.84±1.41	3.63±1.35
bodyfat	0.5±0.18	0.49±0.25	*1.93±0.38*
breastTumor	7.89±0.71	8.02±0.9	8.64±0.87
cholesterol	39.23±6.94	39.27±5.89	41.48±6.33
cloud	0.268±0.11	0.269±0.09	0.273±0.1
cpu	36.35±9.81	32.33±5	28.97±6.35
housing	3.40±0.43	3.42±0.31	*4.78±0.46*
lowbwt	370.48± 78.48	369.24± 56.37	380.26± 67.66
sensory	0.61±0.04	0.61±0.05	*0.63±0.04*
servo	0.62±0.12	0.64±0.27	0.75±0.23
strike	224.79 ±45.05	220.41 ±36.87	*426.37 ±81.16*

Table 4 MAE for Linear Regression and data-sampling ensembles

Dataset	LR	BEM	SR	DS
abalone	1.58±0.08	1.66±0.11	1.51±0.08	1.57±0.06
autohorse	7.30±1.79	8.15±3.82	6.86±3.4	6.87±4.05
autompg	2.27±0.34	2.38±0.16	2.19±0.68	**1.98±0.21**
autoprice	1919.90± 363.34	1792.03± 390.64	1596.31± 491.39	**1554.48± 321.11**
auto93	3.79±1.3	3.23±1.16	3.6±1.33	4.21±1
bodyfat	0.5±0.18	*1.65±0.28*	0.43±0.228	0.43±0.26
breast-tumor	7.89±0.71	7.76±0.94	8.51±1.02	7.98±1.03
cholesterol	39.23±6.94	38.21±4.65	40.62±6.56	38.57±5.02
cloud	0.268±0.11	0.264±0.09	0.29±0.12	0.3±0.08
cpu	36.35±9.81	28.53±7.76	**15.69±5.99**	**25.87±7.37**
housing	3.40±0.43	3.76±0.62	**2.53±0.37**	**2.66±0.54**
lowbwt	370.48± 78.48	389.55± 80.08	369.7± 62.26	384.71± 71.68
sensory	0.61±0.04	0.62±0.06	**0.58±0.05**	**0.58±0.05**
servo	0.62±0.12	*0.8±0.2*	**0.37±0.28**	0.45±0.28
strike	224.79 ±45.05	225.66± 35.5	219.24± 45.67	**177.57± 46.58**

Table 5 MAE for the base technique and feature sampling ensembles

Bagging	Adaboost.R1	BEM	SR	DS
0/15/0	0/10/5	0/13/2	4/11/0	6/9/0

Table 6 Summary of significance comparison

Table 5 shows a comparison of the feature sampling techniques to LR. BEM again proved an in-effective technique. It never significantly improved the MAE for any data sets and in the case of bodyfat and servo it significantly increased the error with an RIE value of 129.56%. SR and DS never significantly decreased the accuracy of any of the data sets. In the case of SR, the error for 4 data sets (cpu, housing, sensory, servo) was significantly reduced. The RRE value for these data sets was 32.13%. DS reduced the error significantly also for 6 data sets (autompg, autoprice, cpu, housing, sensory, strike) with a RRE value of 18.99%. Table 6 summarises the results of the significance comparison for both the data-sampling and feature sampling techniques in a similar fashion to Table 3. It shows clearly that SR and DS were the most effective techniques at significantly reducing the error.

In summary, the regression ensemble techniques of SR and DS in combination with Random Subspace method proved a more effective mechanism of improving the generalization performance of simple regressors than the popular ensemble methods of Bagging and AdaBoost. SR was most effective with nearest neighbours where the RRE value was much larger than that of Bagging and AdaBoost, whereas DS was more effective with Linear Regression. Clearly the technique of BEM did not perform well because RSM, although creating diversity, does not guarantee the level of accuracy of the base models and simple averaging is often not able to compensate for that. The problem of inaccurate models is clearly better addressed by the more sophisticated techniques of SR and DS.

It is perhaps not surprising that Bagging and Boosting would not be effective techniques for base learners such as nearest neighbours and linear regression. Primarily Bagging and Adaboost are variance reducing techniques which have shown to be effective in reducing the error of unstable learners. Typical examples of such learners are neural networks and regression trees. What is principally of interest is that the random sub-space method seems to ensure diversity, allowing bias-reducing techniques such as SR and DS to improve the performance of simple regression models. Tysmbal et al. (Tsymbal et al. 03) make the conjecture that for classification, RSM provides good coverage, i.e. at least, one base model is likely to classify the instance correctly. We intend to investigate further this issue in the context of regression.

In addition, we assessed the effect of ensemble size for the techniques of SR and DS on the MAE. We chose the two data sets, housing and cpu, for which both the ensemble techniques had been effective in significantly reducing the error, and chose LR as our base model. The MAE was recorded for a given random sub-sample and ensemble size using cross-validation. This process was repeated 9 times using a different collection of random sub-samples and the same ensemble size. The average MAE over the 10 runs was then calculated. This process was carried out for ensemble size 3, 5, 10, 15 and 20. The results of these experiments are displayed in Figure 1 and 2.

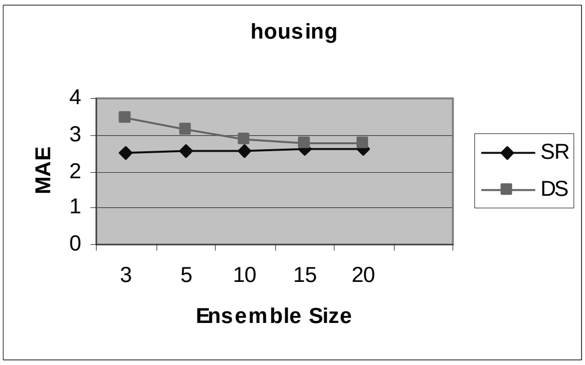

Figure 1 The effect on MAE for the housing data set with increasing ensemble size

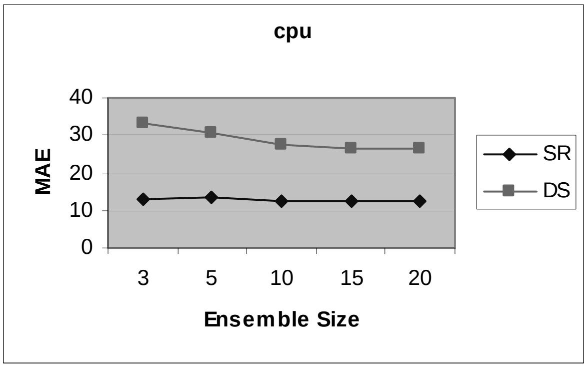

Figure 2 The effect on MAE for the cpu data set with increasing ensemble size

It is clear from Figure 1 and 2 that there are no real benefits of forming ensembles greater than size 3 for the SR technique. In the case of DS, the MAE reaches its minimum by size 10, beyond which there is no significant improvement in error. Clearly in both cases, the ensemble size is much less than the usual ensemble size of 25 chosen for Bagging and Boosting (Opitz and Maclin 99), indicating that RSM/DS and RSM/SR could be potentially a more efficient ensemble technique.

Conclusions

In this paper, we have shown that the technique of random subspace method can be used to create different subspaces of the data upon which the same regression algorithm can be used to build diverse models. These models can be combined using the integration methods of Stacking or Dynamic Selection in order to produce an ensemble that was demonstrated to be more effective for simple regression models than the popular ensemble learning techniques of Bagging and Boosting. We have presented preliminary results which indicate that the ensemble size need not be large for this approach to work. We intend to focus on improving RSM to ensure small ensembles of sufficiently diverse and accurate base models. The reason for this is two-fold: it makes the learning approach more efficient, and it allows for the problem of comprehensibility to be addressed.

Acknowledgements

We gratefully acknowledge the Weka (Witten and Frank 99) framework within which the experimental work was carried out. This material is based upon works supported by the Science Foundation Ireland under Grant No. S.F.I.-02IN.1I111.

References

Breiman, L. 1996a. Stacked Regression. *Machine Learning*, 24:49-64.

Breiman, L. 1996b. Bagging Predictors. *Machine Learning*, 24:123-140.

Christensen, S. 2003. Ensemble Construction via Designed Output Distortion, In *4th International Workshop on Multiple Classifier Systems*, LNAI, Vol. 2709, pp. 286-295, Springer-Verlag.

Dietterich, T. 1998. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine Learning*, 40:139-157.

Dietterich, T. 2000. Ensemble Methods in Machine Learning, In *1st International Workshop on Multiple Classifier Systems* LNAI, Vol 1857, pp. 1-10, Springer-Verlag.

Drucker, H. 1997. Improving Regressors Using Boosting Techniques. In *Proceedings of the 14th International Conference on Machine Learning*, 107-215, Eds. Douglas H. Fisher, Jr., Morgan-Kaufmann.

Freund, Y. and Schapire, R.E. 1996. Experiments with a new Boosting Algorithm, In *Proceedings of 13th International Conference in Machine Learning*, pp. 148-156. Morgan Kaufmann**.**

Ho, T. K. 1998a. The random subspace method for constructing decision forests. *IEEE PAMI*, 20(8):832-844.

Ho, T.K. 1998b. Nearest Neighbors in Random Subspaces, In *Proceedings of the 2nd Int'l Workshop on Statistical Techniques in Pattern Recognition*, Lecture Notes in Computer Science: Advances in Pattern Recognition, pp. 640-648.

LeBlanc, M. and Tibshirani, R. 1992. Combining estimates in Regression and Classification, Technical Report, Dept. of Statistics, University of Toronto.

Merz, C.J., 1996. Dynamical selection of learning algorithms. In Learning from data, artificial intelligence and statistics. .Fisher and H.-J.Lenz (Eds.) New York: Springer.

Merz, C. and Pazzani, M., 1999. A principal components approach to combining regression estimates, *Machine Learning*, 36:9-32.

Opitz, D.W. and Maclin, R. 1999. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169-198.

Perrone, M. and Cooper, L. 1993. When networks disagree: ensemble methods for hybrid neural networks. In *Artificial Neural Networks for Speech and Vision*, pp. 126-142. Chapman & Hall, London.

Puuronen S., Terziyan V., Tsymbal A. 1999. A Dynamic Integration Algorithm for an Ensemble of Classifiers. *Foundations of Intelligent Systems, 11th International Symposium ISMIS'99,* Lecture Notes in Artificial Intelligence, Vol. 1609: 592-600, Springer-Verlag.

Quinlan, R. 1993. Learning with continuous classes, In *Proceedings of the 5th Australian Joint Conference on Artificial Intelligence*, pp. 343-348, World Scientific.

Schaffer, C. 1993. Overfitting avoidance as bias. *Machine Learning* 10:153-178.

Skurichina, M. and Duin, R.P.W. 2002 Bagging, Boosting and the Random Subspace Method for Linear Classifiers, Pattern Analysis and Applications, Vol. 5, pp. 121-135.

Tsymbal, A., Puuronen, S., Patterson, D. 2003. Ensemble feature selection with the simple Bayesian classification, *Information Fusion* 4:87-100, Elsevier.

Witten, I. and Frank, E. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann.

Zenobi, G. and Cunningham, P. 2001. Using Diversity in Preparing Ensemble of Classifiers Based on Different Subsets to Minimize Generalization Error, In *Proceedings of the 12 European Conference on Machine Learning*, pp. 576-587, Springer-Verlag.