Comparison of Bagging, Boosting and Stacking Ensembles Applied to Real Estate Appraisal

Magdalena Graczyk[1], Tadeusz Lasota[2], Bogdan Trawiński[1], Krzysztof Trawiński[3]

[1] Wrocław University of Technology, Institute of Informatics,
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
[2] Wrocław University of Environmental and Life Sciences, Dept. of Spatial Management
Ul. Norwida 25/27, 50-375 Wroclaw, Poland
[3] European Centre for Soft Computing, Edificio Científico-Tecnológico, 3ª Planta,
C. Gonzalo Gutiérrez Quirós S/N, 33600 Mieres, Asturias, Spain
mag.graczyk@gmail.com, tadeusz.lasota@wp.pl, bogdan.trawinski@pwr.wroc.pl,
krzysztof.trawinski@softcomputing.es

Abstract. The experiments aimed to compare three methods to create ensemble models implemented in a popular data mining system WEKA, were carried out. Six common algorithms comprising two neural network algorithms, two decision trees for regression, and linear regression and support vector machine were used to generate individual committees. All algorithms were employed to actual data sets derived from the cadastral system and the registry of real estate transactions. Nonparametric Wilcoxon signed-rank tests to evaluate the differences between ensembles and original models were conducted. The results obtained show there is no single algorithm which produces the best ensembles and it is worth to seek an optimal hybrid multi-model solution.

Keywords: ensemble models, bagging, stacking, boosting, property valuation

1 Introduction

During the last decade the ensemble learning systems attracted attention of many researchers. This collection of methods combines the output of the machine learning systems, in literature called “weak learners” in due to its performance, from the group of learners in order to get smaller prediction errors (in regression) or lower error rates (in classification). The individual estimator must provide different patterns of generalization, thus the diversity plays a crucial role in the training process. Otherwise, the ensemble, called also committee, would end up having the same predictor and provide as good accuracy as the single one. It was proved [11], [12]; [15], [21] that the ensemble performs better when each individual machine learning system is accurate and makes error on the different examples at the same time.

There are many taxonomies for ensemble techniques [4],[17],[22], [24], and there is a recognized group so called data resampling, which generates different training sets to obtain unique learner. To this group we may include bagging [2],[5], boosting [9],[25], and stacking [3],[27]. In boostrap aggregation (bagging), each machine learning system is independently learned on resampled training set, which is randomly picked from the original samples of the training set. Hence, bagging is devoted to the unstable algorithms, where the small changes in the training set, result in large changes in the output of that system. Training of each predictor could be in parallel, in due to independence of training of each machine.

Boosting provides sequential learning of the predictors. The first one is learned on the whole data set, while the following are learned on the training set based on the performance of the previous one. In other words, the examples which were predicted improperly are noted. Then, such examples are more probable to appear in the training set of the next predictor. It results with different machines being specialized in predicting some areas of the dataset.

Stacking is composed of two phases. Firstly, usually different models are learned in the base of data set. Then, the output of each of the model are collected to create a new dataset. In the new dataset each instance is related to the real value that it is suppose to predict. Secondly, that dataset is used with a learning algorithm, the so-called meta-learning, in order to provide the final output.

However, above mentioned techniques are the most recognized ones, we may obtain diversity through applying randomness to our algorithm, e.g., random subspace [1], [14]. It is a feature selection algorithm, which generates different learners from different subset of attributes taken from the feature space.

Very often it is the case that the whole generated ensemble is not an optimal option, i.e. due to bias in the learners. It has been proven that small ensembles can perform better than large ensembles [14], [28], [29]. Although ensemble selection was mostly developed in the classifier environment like [7], there is some bunch of publications dedicated to ensemble selection for regression. The simplest idea is based on performance ranking of learners and make cut for some best ones [6]. Probably the most common way is to optimize it with a genetic algorithm [7],[29], where algorithm fitness function is a generalization order. Other techniques consist of heuristic approaches, like Kappa pruning [20], greedy ones based on complementaries (biases among regressors) [13], and defined diversity measure [23], or negative correlation learing [19].

So far the authors have investigated several methods to construct regression models to assist with real estate appraisal: evolutionary fuzzy systems, neural networks, decision trees, and statistical algorithms using MATLAB, KEEL, RapidMiner, and WEKA data mining systems [10], [16], [18]. In this paper we present results of several experiments with actual data of residential premises sales/purchase transactions aimed at the comparison of bagging, boosting and stacking ensemble techniques using WEKA [26]. Six machine learning algorithms implemented in WEKA were employed including neural networks, regression trees, support vector machine, and statistical linear regression.

2 Techniques and Algorithms Used and Plan of Experiments

The main goal of our study was to compare three approaches to ensemble learning i.e. bagging, stacking and additive regression to examine how they improve the performance of models to assist with real estate appraisal. All experiments were conducted using *WEKA (Waikato Environment for Knowledge Analysis),* a non-commercial and open-source data mining system [8],[26]. WEKA contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

Following WEKA algorithms for building, learning and optimizing models were employed to carry out the experiments.

MLP – MultiLayerPerceptron. Algorithm is performed on networks consisting of multiple layers, usually interconnected in a feed-forward way, where each neuron on layer has directed connections to the neurons of the subsequent layer.

RBF – Radial Basis Function Neural Network for Regression Problems. The algorithm is based on feed-forward neural networks with radial activation function on every hidden layer. The output layer represents a weighted sum of hidden neurons signals.

M5P – Pruned Model Tree. The algorithm is based on decision trees, however, instead of having values at tree's nodes, it contains a multivariate linear regression model at each node. The input space is divided into cells using training data and their outcomes, then a regression model is built in each cell as a leaf of the tree.

M5R – M5Rules. The algorithm divides the parameter space into areas (subspaces) and builds in each of them a linear regression model. It is based on M5 algorithm. In each iteration a M5 Tree is generated and its best rule is extracted according to a given heuristic. The algorithm terminates when all the examples are covered.

LRM - Linear Regression Model. Algorithm is a standard statistical approach to build a linear model predicting a value of the variable while knowing the values of the other variables. It uses the least mean square method in order to adjust the parameters of the linear model/function.

SVM – NU-Support Vector Machine. Algorithm constructs support vectors in high-dimensional feature space. Then, hyperplane with the maximal margin is constructed. Kernel function is used to transform the data, which augments the dimensionality of the data. This augmentation provokes that the data can be separated with an hyperplane with much higher probability, and establish a minimal prediction probability error measure.

Three metalearning methods were employed to create ensembles:

Additive regression – is an example of boosting. The algorithm starts with an empty ensemble and incorporates new members sequentially. At each stage the model that maximizes the performance of the ensemble as a whole is added, without altering those already in the ensemble.

Bagging – consists in aggregating results of n models which were created on the basis of n bootstrap sets. The bootstrap sets are created out of the original dataset through feature selection or random drawing with replacement. The final result is calculated by averaging the outputs of individual models built over each bootstrap set.

Stacking – in stacking, the result of a set of different base learners at the level-0 is combined by a metalearner at the level-1. The role of the metalearner is to discover how best to combine the output of the base learners. In each run LRM, M5P, M5R, MLP, RBF, and SVM were the base learners and one of them was the metalearner.

Actual data used to generate and learn appraisal models came from the cadastral system and the registry of real estate transactions referring to residential premises sold in one of big Polish cities at market prices within two years 2001 and 2002. They constituted original data set of 1098 cases of sales/purchase transactions. Four attributes were pointed out by an expert as price drivers: 1 – usable area of premises, 2 – number of storeys in the building where premises were located, 3 – floor on which premises were located, 4 – year of building construction, in turn, price of premises was the output variable. In order to assure diversity five data sets were used in the experiments to build appraisal models: first one denoted by 1234 contained all four attributes of premises, and the other included all combinations of three attributes and were denoted as 123, 124, 134, and 234. However in this paper we present only the results obtained with 1234 and 124 data sets which revealed the best performance.

All runs of experiments were repeated for from 1 to 30 steps of each ensemble method, 10-fold cross validation and root mean square error as fitness function were applied. Schema of the experiments with WEKA was depicted in Fig. 1. In the case of bagging *i-th* step means that the ensemble model was built with *i* bootstrap sets. As an output of each run WEKA provided two vectors of actual and predicted prices of premises for all input instances. Having such output we were able to calculate the final predictive accuracy of each committee in terms of MAPE (mean absolute percentage error). We also conducted nonparametric Wilcoxon signed-rank test to evaluate the differences between ensembles and original models.

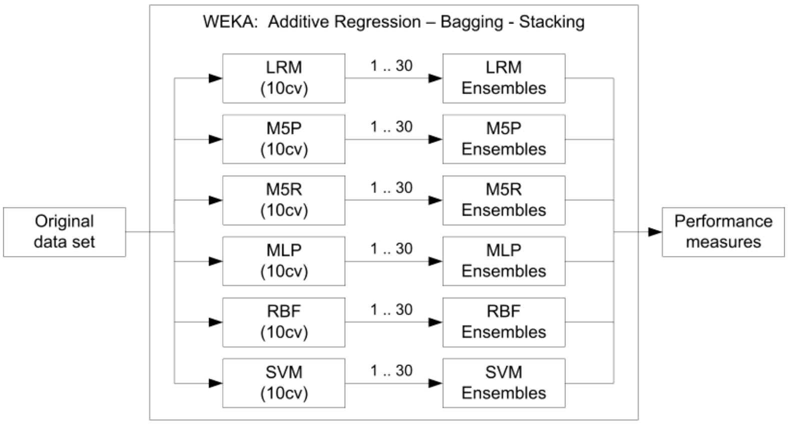

Fig. 1. Schema of the experiments with WEKA

3 Results of Experiments

The performance of the ensemble models built by additive regression, bagging and stacking for 2-30 steps was presented in Fig. 2,3 and 4 respectively. The outcome of 1234 models was placed in left columns whereas the right ones contain the results provided by 124 models. The charts allow you to observe how the accuracy of sequential ensembles changed when the number of steps was increased, so the scales of the axes were retained the same in each figure. The horizontal lines reflect the MAPE of original models. If the error reduction achieved by ensembles compared to an original model takes place, then the horizontal line is located above tops of bars representing the prediction accuracy of individual committees. In the case of *Additive Regression* the predictive accuracy of ensembles remained at the same level as the number of was increased except for the M5P models. For *Bagging* it could be observed that the values of MAPE tended to decrease when the number of bags was getting bigger. *Stacking* was characterized by varying performance of models. In general there were no substantial differences between 1234 and 124 models.

In Table 1 the results of nonparametric Wilcoxon signed-rank test to evaluate the outcome of individual ensemble models were presented. The zero hypothesis H0 stated there were not significant differences in accuracy, in terms of MAPE, between a given ensemble model and the model built using an original (base) data set. N denotes that there was no evidence against the H0 hypothesis, whereas Y means that there was evidence against the H0 hypothesis. In most cases the differences were significant except for LRM and M5R models for *Additive Regression*, LRM, M5P, and SVM models for *Bagging*, and SVM ones for *Stacking*.

In order to consider the possibility to apply a hybrid multi-model model approach in Tables 2 and 3 the ensembles with the lowest values of MAPE for individual algorithms were placed for 1234 and 124 models respectively. The columns denoted by No contain the numbers of steps at which the best result was achieved, by Gain – the benefit of using committees i.e. the percentage reduction of MAPE values of respective ensembles compared to MAPE provided by the original models , and H0 the results of Wilcoxon tests of the same meaning as in Table 1.

Table 1. Results of Wilcoxon tests for 1234 models

No	Additive Regression						Bagging						Stacking					
	LRM	M5P	M5R	MLP	RBF	SVM	LRM	M5P	M5R	MLP	RBF	SVM	LRM	M5P	M5R	MLP	RBF	SVM
2	N	N	N	Y	Y	Y	N	Y	N	N	Y	Y	Y	Y	Y	Y	N	N
3	N	Y	N	Y	Y	Y	N	N	N	N	Y	N	Y	Y	Y	Y	Y	N
4	N	Y	N	Y	Y	Y	N	N	N	N	Y	N	Y	Y	Y	Y	Y	N
5	N	Y	N	Y	Y	Y	N	N	N	Y	Y	N	Y	Y	Y	Y	Y	N
6	N	Y	N	Y	Y	Y	N	N	N	Y	Y	N	Y	Y	Y	Y	Y	N
7	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
8	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
9	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
10	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
11	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
12	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
13	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
14	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
15	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
16	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
17	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
18	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
19	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
20	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
21	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
22	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
23	N	N	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
24	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
25	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
26	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
27	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
28	N	Y	N	Y	Y	Y	N	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
29	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
30	N	Y	N	Y	Y	Y	Y	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N

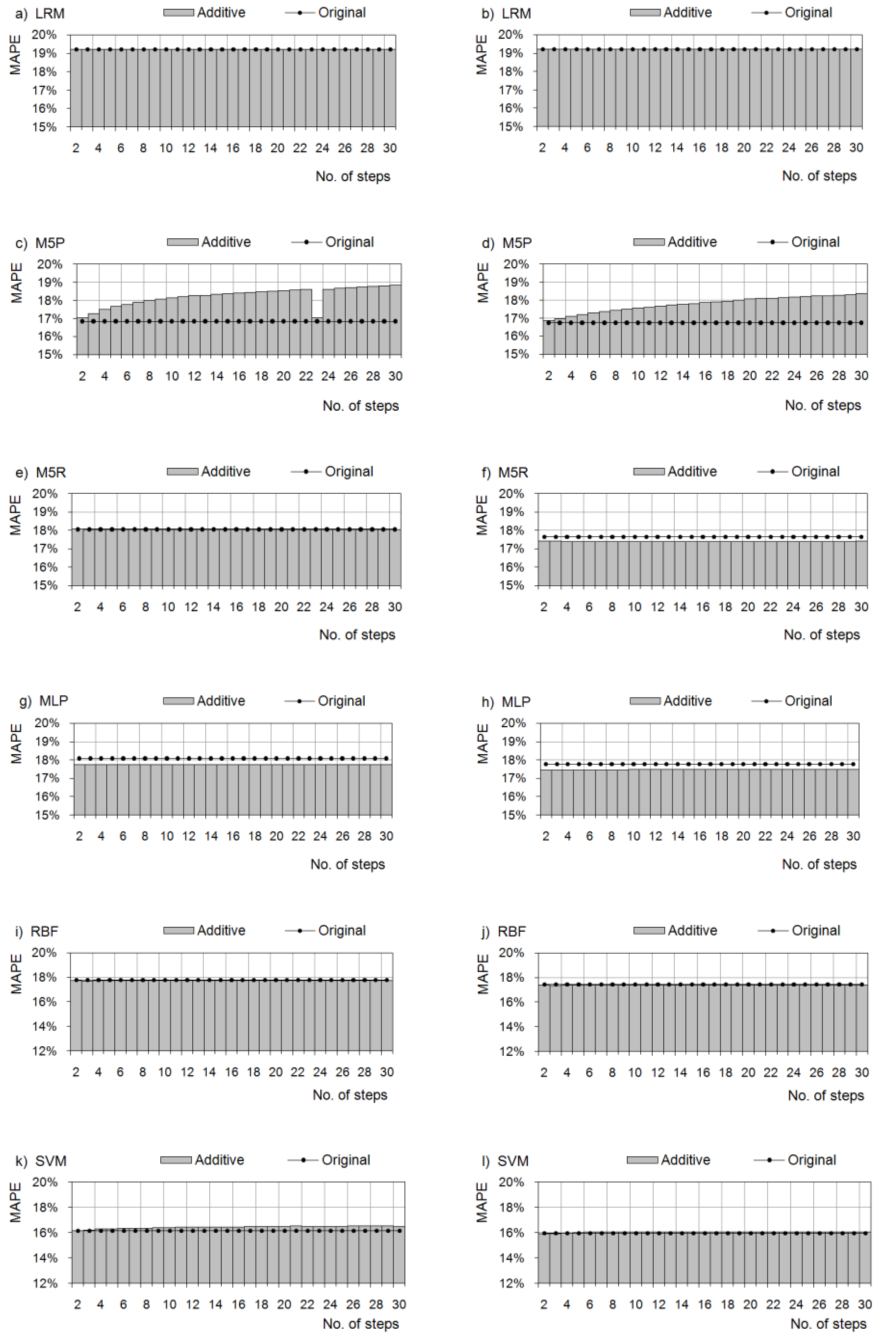

Fig. 2 a)-l). Performance of *Additive Regression* ensembles, in terms of MAPE, compared with original models for individual algorithms (1234 – left column, 124 – right column)

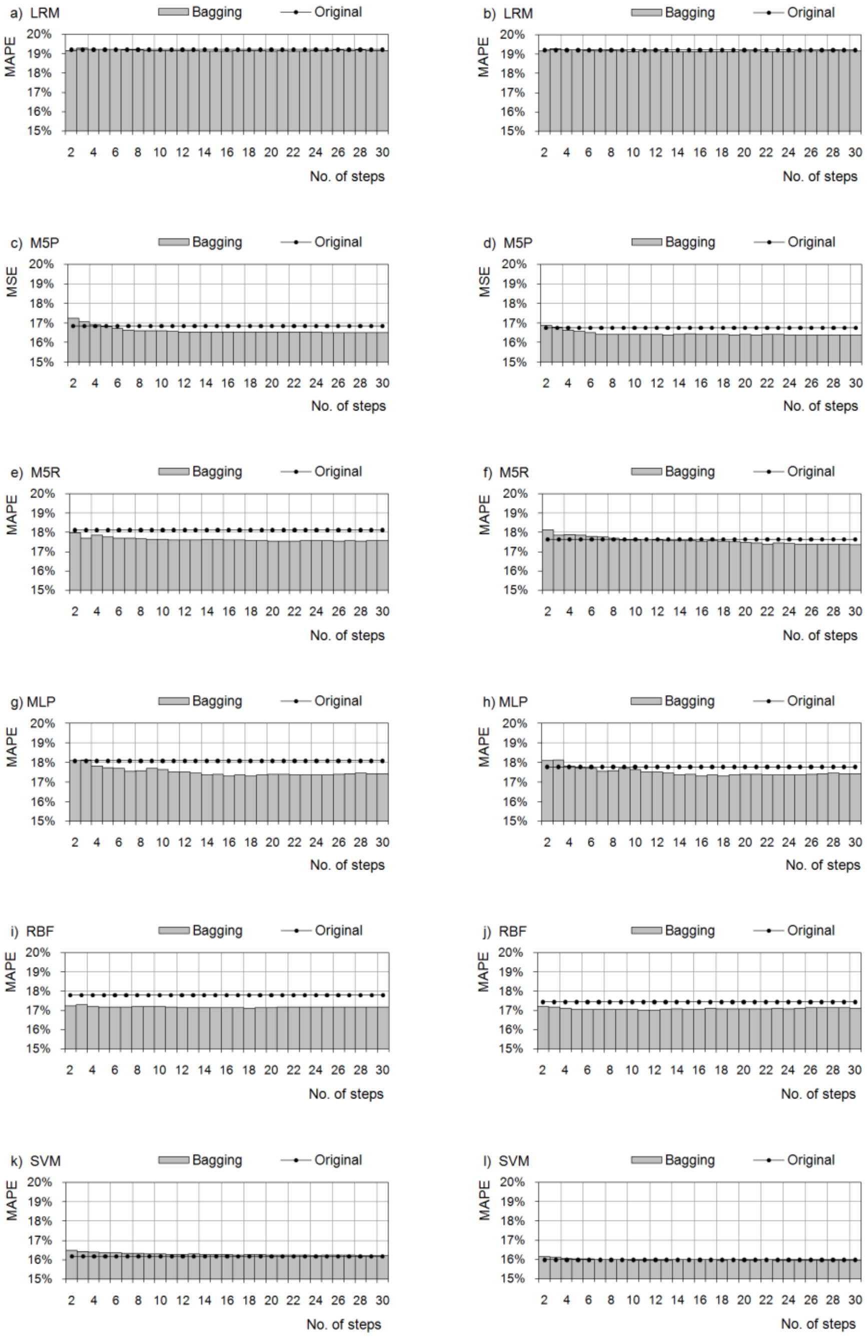

Fig. 3 a)-l). Performance of *Bagging* ensembles, in terms of MAPE, compared with original models for individual algorithms (1234 – left column, 124 – right column)

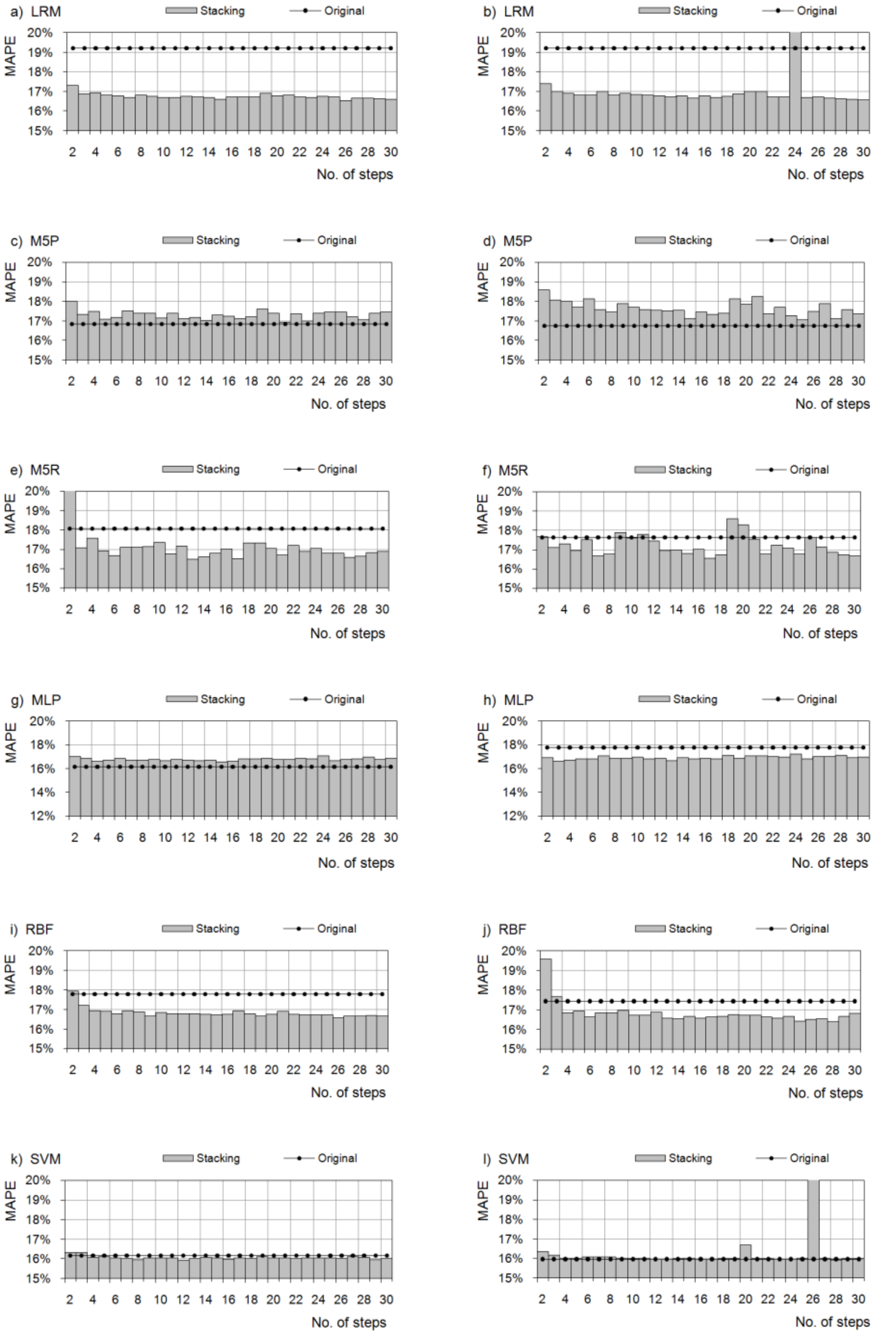

Fig. 4 a)-l). Performance of *Stacking* ensembles, in terms of MAPE, compared with original models for individual algorithms (1234 – left column, 124 – right column)

Table 2. The best ensembles for individual algorithms with minimal values of MAPE for 1234

	Additive regression				Bagging				Stacking			
Alg.	**MAPE**	**No**	**Gain**	**H0**	**MAPE**	**No**	**Gain**	**H0**	**MAPE**	**No**	**Gain**	**H0**
LRM	0.19202	2	0.00%	N	0.19133	14	0.36%	Y	0.16533	26	13.90%	Y
M5P	0.17025	2	-1.08%	N	0.16500	28	2.04%	N	0.16994	21	-0.60%	Y
M5R	0.18038	7	0.12%	N	0.17537	22	3.18%	Y	0.16498	13	8.65%	Y
MLP	0.17735	29	1.89%	Y	0.17306	18	4.27%	Y	0.16567	15	-2.48%	Y
RBF	0.17707	3	0.42%	Y	0.17126	18	3.69%	Y	0.16590	26	6.60%	Y
SVM	0.16166	2	0.00%	Y	0.16224	24	-0.36%	N	0.15935	12	1.43%	Y

Table 3. The best ensembles for individual algorithms with minimal values of MAPE for 124

	Additive regression				Bagging				Stacking			
Alg.	**MAPE**	**No**	**Gain**	**H0**	**MAPE**	**No**	**Gain**	**H0**	**MAPE**	**No**	**Gain**	**H0**
LRM	0.19202	2	0.00%	N	0.19121	16	0.42%	Y	0.16572	30	13.70%	Y
M5P	0.16859	2	-0.65%	N	0.16379	24	2.24%	Y	0.17048	25	-1.78%	N
M5R	0.17392	4	1.41%	Y	0.17370	30	1.54%	N	0.16545	17	6.21%	Y
MLP	0.17462	2	1.71%	Y	0.17306	18	2.59%	Y	0.16607	3	4.71%	Y
RBF	0.17374	3	0.34%	N	0.17020	11	2.38%	Y	0.16404	28	5.91%	Y
SVM	0.15899	2	0.46%	N	0.15962	26	0.07%	N	0.15931	28	0.26%	N

When analysing the data presented in tables 2 and 3 it can be noticed that the most substantial benefit provided LRM and M5P used as the metalearners in *Stacking*. The lowest values of MAPE achieved SVM models but with minor, if any, gain and statistically insignificant. All algorithms but SVM ensured gain in *Bagging*. These results are encouraging to undertake further research on hybrid ensemble methods.

4 Conclusions and Future Work

Our study on the application of six different machine learning algorithms to three metalearning methods in WEKA, i.e. *Additive Regression*, *Bagging*, and *Stacking* has provided some interesting observations. The results in terms of MAPE as the predictive accuracy measure have shown there are substantial differences between individual algorithms and methods. In general models obtained with *Stacking* were characterized by the lowest prediction error but the outcome tended to vary giving once better results and the other times much worse. *Bagging* approach, in turn, seemed to be more stable but gave worse results and *Additive Regression* provided similar results for all steps. With *Additive Regression* all MLP and RBF ensembles did yield significant error reduction compared to original models. With *Bagging* most M5R, MLP and RBF ensembles produced significant error reduction compared to original models. With *Stacking* most LRM, M5R and RBF multiple-models achieved significant improvement in accuracy. The results obtained show there is no single algorithm which produces the best ensembles and it is worth to seek an optimal hybrid multi-model solution using greater number of different data sets.

References

1. Banfield, R. E., et al.: A Comparison of Decision Tree Ensemble Creation Techniques. IEEE Trans. on Pattern Analysis and Machine Intelligence 29:1, pp. 173--180 (2007)

2. Breiman, L.: Bagging Predictors. Machine Learning 24:2, pp. 123--140 (1996)
3. Breiman, L.: Stacked Regressions, Machine Learning 24: 1, pp. 49--64 (1996)
4. Brown, G., Wyatt, J., Harris, R., Yao, X.: Diversity Creation Methods: A Survey and Categorisation. Journal of Information Fusion 6:1, pp. 5--20 (2005)
5. Büchlmann, P., Yu, B.: Analyzing bagging, Annals of Statistics 30, pp. 927--961 (2002)
6. Chawla, N.V., Hall, L.O., Bowyer, K. W., Kegelmeyer, W. P.: Learning Ensembles From Bites: A Scalable and Accurate Approach. J. of Mach. Learn. Res. 5, pp. 421--451 (2004)
7. Cordón, O., Quirin, A.: Comparing Two Genetic Overproduce-and-choose Strategies for Fuzzy Rule-based Multiclassification Systems Generated by Bagging and Mutual Information-based Feature Selection. Int. J. Hybrid Intelligent Systems, in press (2009)
8. Cunningham S.J., Frank E., Hall M., Holmes G., Trigg L., Witten I.H., WEKA: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, New Zealand (2005)
9. Freund, Y., Schapire, R.E.: Decision-theoretic generalization of on-line learning and an application to boosting, J. Computer and System Sciences 55:1, pp. 119--139 (1997)
10. Graczyk, M., Lasota, T., and Trawiński, B.: Comparative Analysis of Premises Valuation Models Using KEEL, RapidMiner, and WEKA. In Nguyen N.T. et al. (Eds.): ICCCI 2009, LNCS (LNAI) 5796, pp. 800-812, Springer, Heidelberg (2009)
11. Hansen, L., Salamon, P.: Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence 12:10, pp. 993--1001 (1990)
12. Hashem, S.: Optimal linear combinations of neural networks. Neural Networks, 10:4, pp. 599--614 (1997)
13. Hernandez-Lobato, D., Martinez-Munoz, G., Suarez, A.: Pruning in ordered regression bagging ensembles, in: G.G. Yen (Ed.), Proceedings of the IEEE World Congress on Computational Intelligence, pp 1266--1273 (2006)
14. Ho, K.T.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20:8, pp. 832--844 (1998)
15. Krogh, A. and Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: Advances in Neural Inf. Proc. Systems, MIT Press, pp. 231--238 (1995)
16. Król, D., Lasota, T., Trawiński, B., Trawiński, K.: Investigation of Evolutionary Optimization Methods of TSK Fuzzy Model for Real Estate Appraisal. International Journal of Hybrid Intelligent Systems 5:3, pp. 111--128 (2008)
17. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley (2004)
18. Lasota, T., Mazurkiewicz, J., Trawiński, B., Trawiński, K.: Comparison of Data Driven Models for the Validation of Residential Premises Using KEEL. International Journal of Hybrid Intelligent Systems, in press (2009)
19. Liu, Y., Yao, X.: Ensemble learning via negative correlation, Neural Networks 12:10, pp. 1399--1404 (1999)
20. Margineantu, D.D., Dietterich, T.G.: Pruning Adaptive Boosting, Proc. 14th Int. Conf. Machine Learning, pp. 211--218 (1997)
21. Opitz, D. and Shavlik, J.W.: Actively searching for an effective neural network ensemble, Connection Science 8:3-4, pp. 337--353 (1996)
22. Polikar, R.: Ensemble Learning. Scholarpedia 4:1, pp. 2776 (2009)
23. Prodromidis, A.L., Chan, P.K., Stolfo, S.J.: Meta-Learning in a Distributed Data Mining System: Issues and Approaches. In Kargupta, H., Chan, P.K. (Eds.) Advances of Distributed Data Mining, AAAI Press (2000)
24. Rokach L.: Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. Comp. Stat. and Data Anal. 53, pp. 4046-4072 (2009)
25. Schapire, R. E.: The Strength of Weak Learnability. Mach. Learning 5:2, 197--227 (1990)
26. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco (2005)
27. Wolpert, D.H.: Stacked Generalization. Neural Networks 5:2, pp. 241--259 (1992)
28. Yao, X., Liu, Y.: Making Use of Population Information in Evolutionary Artificial Neural Networks. IEEE Trans. Systems, Man, and Cybernetics, Part B 28:3, pp. 417--425 (1998)
29. Zhou Z.H., Wu, J., Tang, W.: Ensembling Neural Networks: Many Could Be Better Than All. Artificial Intelligence 137, pp. 239--263 (2002)