TÜBİTAK

Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2013) 21: 586 – 602
© TÜBİTAK
doi:10.3906/elk-1102-1033

A comparative review of regression ensembles on drug design datasets

Mehmet Fatih AMASYALI[1,*], **Okan ERSOY**[2]
[1]Computer Engineering Department, Yıldız Technical University, 34349 İstanbul, Turkey
[2]School of Electrical and Computer Engineering, Purdue University, West Lafayette,
Indiana 47907, USA

Received: 09.02.2011 • **Accepted:** 29.01.2012 • **Published Online:** 22.03.2013 • **Printed:** 22.04.2013

Abstract: Drug design datasets are usually known as hard-modeled, having a large number of features and a small number of samples. Regression types of problems are common in the drug design area. Committee machines (ensembles) have become popular in machine learning because of their good performance. In this study, the dynamics of ensembles used in regression-related drug design problems are investigated with a drug design dataset collection. The study tries to determine the most successful ensemble algorithm, the base algorithm–ensemble pair having the best/worst results, the best successful single algorithm, and the similarities of algorithms according to their performances. We also discuss whether ensembles always generate better results than single algorithms.

Key words: Drug design datasets, ensemble algorithms, regression, regression ensembles

1. Introduction

Drug datasets are often known as hard-modeled datasets because of a small number of samples and a large number of dimensions. Getting good prediction results with such datasets in the process of drug design can provide large financial and time savings in pharmaceutical research and development.

In machine learning, it is popular to use algorithm ensembles by using several algorithms and combining their results. In ensembles, the base algorithms generate partially dependent or independent results on the same or a different part of a dataset, and then the results are combined in several ways. The success of an ensemble depends on 2 main properties: the first is the individual success of the base algorithms of the ensemble and the second is the independence of the base algorithms' results from each other (low error, high diversity) [1].

This study aims at overcoming the difficulties of modeling drug datasets using ensembles. Our experiments focus on regression ensembles because most drug design problems are of the regression type. The performance of ensemble algorithms over drug datasets is investigated both with respect to the ensemble algorithms themselves and to the base algorithms used within the ensemble algorithms. In the literature, several ensemble algorithms are proposed. However, the application of these algorithms to drug design datasets has been limited. To provide more comprehensive results to the drug design community, the performances of 4 different ensemble algorithms, 1 feature selection algorithm, and 7 base algorithms for each ensemble are comparatively evaluated on 15 drug design datasets in this paper. The same experiments are repeated with the dimensionally reduced drug design datasets.

The paper consists of 6 sections. Section 2 discusses the algorithms used in the study. Section 3 presents previous works in this area in the form of a table. Section 4 introduces the dataset collection in the form of 3

[*]Correspondence: mfatih@ce.yildiz.edu.tr

tables. The experimental results are presented in Section 5. Section 6 contains the conclusions.

2. Algorithms used in the study

In this section, the base and ensemble algorithms used in our study are briefly described. For the evaluation of the algorithms, the WEKA software was used [2]. Each ensemble algorithm was used with each of the base algorithms. The base algorithms were also used alone. With this configuration (4 ensemble + 1 single) × (7 base) = 35 different algorithms were obtained and used for the prediction of the drug design datasets.

2.1. Ensemble algorithms

Bagging/bootstrapping: Bagging generates N new equal-sized datasets from the original dataset by selecting samples with a replacement [3]. The base algorithms are trained with the datasets. The independence of the individual results is confirmed in the experiments to some degree. N was chosen as 10 in our experiments. The results of the base algorithms are simply averaged to produce the ensemble result.

Additive regression: This is the adaptation of the AdaBoost algorithm to regression types of problems [4]. At each iteration, the samples having big errors at the previous iteration are considered. The iteration number was chosen as 10 in our study. The ensemble result is the weighted mean of the base algorithms. The weights are inversely proportional to the errors of the base algorithms.

Random subspace: In this ensemble algorithm, all of the samples are used, but all of the features are not used. Each algorithm in the committee is trained by a randomly selected subset of all of the features [5]. With this approach, the diversity of the results of the algorithms is increased. In our study, the number of features in each subspace is chosen as half of the original number of features. The results of 10 algorithms trained in different subspaces are combined. The results of the base algorithms are simply averaged to produce the ensemble result.

Rotation forest: This is an ensemble method that trains N decision trees independently, using a different set of extracted features for each tree [6]. Bootstrap samples are taken as the training set for the individual classifiers, as in bagging. The main heuristic is to apply the feature extraction and to subsequently reconstruct a full feature set for each classifier in the ensemble. To do this, the feature set is split randomly into K subsets, principal component analysis is run separately on each subset, and a new set of linear extracted features is constructed by pooling all of the principal components. The data is transformed linearly into the new feature space. The base learner is trained with this data set. Different splits of the feature set will lead to different extracted features. N was chosen as 10 in our experiments. The results of the base algorithms are simply averaged to produce the ensemble result.

The ensemble algorithms and their abbreviations used in this study are shown in Table 1.

Table 1. Ensemble algorithms and their abbreviations.

Ensemble algorithm	Abbreviation
Bagging	BG
Additive regression (boosting)	AR
Random subspace	RS
Rotation forest	RF

2.2. Base regression algorithms

In our study, 7 regression algorithms were used as base learners in the ensembles. They are as follows:

M5 model trees: The regression tree algorithm proposed by Quinlan [7]. The dataset is divided into subspaces within the leaves. A linear model is utilized in each subspace. The subspace boundaries are defined by the "feature-threshold value" pairs, which mostly decrease the standard deviations of the output values.

REP: A fast regression tree algorithm [2]. Its leaves contain constant output values. At each node, a "feature-threshold value" pair is selected based on the most reduction in the variance of the output. The tree is then pruned by a bottom-up reduced-error pruning.

Partial least squares: Principal component analysis identifies directions with the greatest variation, but does not use the output information. Partial least squares also takes into account the direction of the output values when transforming the dataset into a lower dimensional space [8].

Simple linear regression: A linear regression model is constructed for each single feature. The model having the lowest squared error is selected as the final model [2].

K nearest neighbor: A sample-based algorithm. The prediction of a test sample is the averaged output of its K nearest training samples.

Decision stump: This algorithm constructs a decision tree with only one decision node. The decision node is selected according to the lowest root mean squared error (RMSE).

Support vector regression: This algorithm implements the support vector machine for regression [9].

The base algorithms and their abbreviations are shown in Table 2.

Table 2. Base regression algorithms used and their abbreviations.

Base regression algorithm	Abbreviation
M5 model trees	M5P
REP	REP
Partial least squares	PLS
Simple linear regression	SLR
Decision stump	DS
K nearest neighbor	NN
Support vector regression	SVR

2.3. Dimension reduction process

Drug design datasets generally have a very large number of features. In our study, the original datasets and their dimensionally reduced versions are used. By doing so, the effects of the feature selection process on the accuracies of the algorithms are investigated. The accuracies over the original and dimensionally reduced datasets are compared. The CfsSubsetEval method is used for feature selection [10]. This method chooses the subsets of the features that are highly correlated with the output while having low intercorrelation. The method is a wrapper type of feature selection strategy. It starts with the empty set of attributes and searches forward by considering all of the possible single attribute additions at a given point. It iteratively adds attributes with the highest correlation with the output as long as there is not already an attribute in the subset that has a higher correlation with the attribute in question. It stops adding attributes when there are no attributes having these conditions.

2.4. Dataset collection

Our drug data collection consists of 15 drug datasets obtained from several studies. The datasets are shown in Table 3. The datasets with 1142 features were formed using the Adriana.Code software [11]. The molecules and outputs were obtained from the original studies. The other datasets were obtained exactly from the original studies. The datasets in ARFF file format are available in [12].

Table 3. The 15 drug design datasets.

Dataset ID	Dataset name	Number of samples	Original number of features	Number of selected features	Reference
1	benzo	195	32	32	[13]
2	carbolenes	37	1142	15	[14]
3	chang	34	1142	7	[15]
4	cristalli	32	1142	14	[15]
5	depreux	26	1142	12	[15]
6	mtp	274	1142	24	[13]
7	pah	80	112	10	[16]
8	pdgfr	79	320	11	[17]
9	phen	22	110	6	[18]
10	phenetyl	22	628	7	[19]
11	qsbr_y	15	9	3	[20]
12	qsfsr	19	9	3	[21]
13	selwood	31	53	5	[22]
14	strupcz	34	1142	15	[15]
15	yokohoma	12	1142	11	[15]

3. Experimental results

Seven base regressors were used together with each ensemble algorithm on 15 regression-type drug design problems. The experiments were done to answer the following questions in drug design problems:

- Do the algorithm ensembles generate more successful results than a single algorithm?
- What is the most successful ensemble algorithm?
- What is the base algorithm–ensemble pair with the best results?
- Which algorithm performs well with the ensembles?
- What is the most successful single algorithm?
- How are the algorithms and datasets grouped according to their performances?
- How does the dimension reduction process affect the results?

To answer these questions, 36 algorithms ((4 ensemble + 1 single) × (7 base algorithms) + Zero Rule algorithm = 36) were employed on the 15 drug design datasets described in Table 3 and their dimensionally reduced versions. A 5 × 2 cross validation was used and the RMSE results were averaged.

The RMSE is defined as:

$$RMSE_{a\lg.name} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y^{i}_{a\lg.name} - y^{i}_{actual}\right)^{2}}, \quad (1)$$

where $y^{i}_{a\lg.name}$ is the prediction of alg.name for the i*th* test sample, y^{i}_{actual} is the actual output value of the i*th* test sample, and N is the number of test samples.

The Zero Rule algorithm measures the default error of a dataset. The RMSE value of the Zero Rule is calculated as follows:

$$RMSE_{ZeroRule} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y^m - y^i_{actual}\right)^2}, \quad y^m = \frac{1}{T}\sum_{j=1}^{T} y^j_{actual}, \tag{2}$$

where y^j_{actual} is the actual output value of the jth training sample, $y^i_{a\lg .name}$ is the prediction of alg.name for the ith test sample, T is the number of training samples, and N is the number of test samples.

Our base learners and ensemble algorithms have some hyperparameters to optimize. We used 2-fold cross-validation to optimize these parameters. In bagging, we optimized the bagging size by trying values of 50%, 75%, and 100%. In additive regression, we optimized the shrinkage by trying values of 0.1, 0.5, and 1. In random subspace, we optimized the subspace size by trying values of 25%, 50%, and 75%. In rotation forest, we optimized the remove percentage by trying values of 25%, 50%, and 75%. In the M5P and REP trees, we optimized the minimum number of instances by trying values of 1, 2, 3, 4, and 5. In K nearest neighbor, we optimized K by trying values of 1, 3, and 5. In support vector regression, we optimized C by trying values of 0.01, 0.1, 1, 10, and 100.

In the 5 × 2 cross validation methodology, the dataset is randomly divided after shuffling into 2 halves. One half is used in the training and the other is used in the testing, and vice versa. This validation is repeated 5 times. In the results of this validation, 10 estimates of testing the RMSE were obtained for each algorithm and each dataset. In some experiments, very high RMSE results were obtained, especially with the simple linear regression algorithm disturbing the overall averages. Because of this, the performance comparisons of the algorithms were done with the algorithms' success ranking instead of the averaged RMSEs. In each experiment, the averaged 5 × 2 cross-validation RMSEs were sorted in ascending order. The algorithm with the lowest RMSE got the 1st ranking. The worst got the 36th ranking. These success rankings are given in Tables 4 and 5. In Table 4, the results with the original datasets are shown. In Table 5, the results with the dimensionally reduced datasets are shown. The 15 datasets are ordered along the columns of the tables. The algorithms are ordered along the rows of the tables. The average success rate and standard deviation of each algorithm are shown in the last 2 columns.

In Tables 6 and 7, the summaries of Tables 4 and 5 are given, respectively. Each cell is the averaged success ranking of the experiments with the base algorithm in the cell's row and the ensemble algorithm in the cell's column. The average success rankings of the single algorithms used are given in the 'Single' column. In the Avg. column, the averaged success rankings of the experiments with respect to the base algorithms are given. In the 'Avg.' row, the averaged success rankings of the experiments with respect to the ensemble algorithms are given.

The Nemenyi test [23] was also applied to determine whether there was a statistically significant difference between the algorithms' average ranks. According to the Nemenyi test for 15 datasets, 36 algorithms, and a significance level of 5%, 2 algorithms are different if the distance between their average ranks is at least 14.76. In Figures 1 and 2, the graphical representation of the Nemenyi test results is shown.

When Tables 4, 5, 6, and 7 and Figures 1 and 2 are investigated, the following conclusions are reached.

For the experiments with the original datasets (Tables 4 and 6):

- The best ranking performance (6.00) is obtained with the additive regression-partial least squares (AR-PLS) algorithm.

- The best performed ensemble algorithms are additive regression (AR) and bagging (BG).

Table 4. The success ranking of 36 algorithms on 15 original drug datasets (best to worst, 1 to 36).

Dataset ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Avg.
RF-M5P	1	20	29	28	24	35	32	33	2	5	34	30	29	28	10	22.67
RF-REP	15	8	5	8	1	6	17	1	16	25	5	11	11	12	7	9.867
RF-PLS	2	29	17	25	30	26	18	34	4	29	32	27	17	21	24	22.33
RF-SLR	16	23	28	35	31	7	4	24	35	2	30	32	25	35	27	23.6
RF-DS	17	1	18	9	10	8	24	2	11	13	17	15	1	13	1	10.67
RF-NN	8	2	10	4	7	9	13	14	12	11	6	12	4	1	25	9.2
RF-SVR	29	36	36	36	36	32	36	36	36	36	11	34	19	36	36	32.33
AR-M5P	3	25	23	10	11	27	5	3	13	7	23	1	18	2	11	12.13
AR-REP	30	9	6	20	2	28	27	25	32	28	7	16	20	14	4	17.87
AR-PLS	4	3	11	5	12	1	1	4	1	3	8	13	5	3	16	**6**
AR-SLR	9	24	24	26	27	10	2	5	5	17	35	17	32	27	34	19.6
AR-DS	18	10	19	14	15	11	19	6	17	18	18	7	6	22	2	13.47
AR-NN	10	11	20	15	21	2	20	15	26	19	3	14	12	4	19	14.07
AR-SVR	31	35	35	33	35	36	25	26	9	33	24	35	34	34	30	30.33
BG-M5P	5	4	2	16	16	34	3	35	33	26	29	21	21	5	12	17.47
BG-REP	19	12	3	17	3	12	21	7	21	27	12	8	7	20	3	12.8
BG-PLS	6	13	12	6	8	3	6	8	6	6	13	2	2	6	13	**7.333**
BG-SLR	20	26	34	34	26	13	7	9	18	12	25	31	33	29	32	23.27
BG-DS	21	14	13	7	6	14	26	10	25	22	14	9	8	15	14	14.53
BG-NN	11	5	15	3	17	15	14	16	19	14	1	5	3	7	5	10
BG-SVR	32	34	33	30	34	29	35	27	3	32	33	33	22	33	31	29.4
RS-M5P	12	28	7	22	29	33	34	28	14	9	19	29	23	26	8	21.4
RS-REP	22	6	4	18	4	16	15	17	22	24	9	10	13	16	15	14.07
RS-PLS	13	15	21	11	18	4	8	11	7	4	20	18	9	17	20	13.07
RS-SLR	23	21	32	29	25	20	9	18	23	10	31	22	30	31	33	23.8
RS-DS	24	22	22	21	13	17	29	19	27	16	21	23	14	8	9	19
RS-NN	14	7	8	1	14	18	16	12	20	15	2	6	10	9	21	11.53
RS-SVR	33	32	30	31	33	21	31	29	10	35	22	24	31	30	28	28
M5P	25	31	14	12	22	31	10	21	28	20	26	3	26	25	22	21.07
REP	34	16	9	24	9	30	28	30	29	31	15	28	24	18	17	22.8
PLS	7	17	16	13	19	5	11	13	8	1	16	4	15	10	23	11.87
SLR	27	30	26	27	28	22	12	22	30	8	36	25	36	23	35	25.8
DS	28	27	27	23	23	23	30	31	31	23	27	19	27	24	18	25.4
NN	26	18	25	2	20	19	22	20	24	21	4	20	16	11	26	18.27
SVR	35	33	31	32	32	24	23	23	15	34	28	36	35	32	29	29.47
Zero0	36	19	1	19	5	25	33	32	34	30	10	26	28	19	6	21.53

Table 5. The success ranking of 36 algorithms on 15 dimensionally reduced drug datasets (best to worst, 1 to 36).

Dataset ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Avg.
RF-M5P	1	4	36	35	32	8	9	11	2	4	27	15	16	29	22	16.73
RF-REP	15	16	5	19	7	9	17	12	17	27	12	27	8	10	17	14.53
RF-PLS	2	5	16	32	33	10	1	1	3	1	32	16	20	30	1	13.53
RF-SLR	16	7	17	23	35	11	2	3	8	6	31	17	11	35	4	15.07
RF-DS	17	8	10	5	8	12	31	13	21	28	23	28	2	1	5	14.13
RF-NN	9	11	2	7	9	1	18	4	13	14	6	9	1	7	6	7.8
RF-SVR	29	26	22	33	36	24	26	36	35	11	13	20	23	36	15	25.67
AR-M5P	3	21	23	8	10	13	10	20	14	15	18	1	34	15	23	15.2
AR-REP	30	31	28	25	14	32	32	33	33	34	7	31	24	20	32	27.07
AR-PLS	4	6	24	2	22	2	11	5	4	7	25	4	21	2	7	9.733
AR-SLR	10	22	31	26	30	14	12	14	9	8	30	5	33	28	24	19.73
AR-DS	18	12	29	12	2	25	27	21	22	26	19	25	17	11	25	19.4
AR-NN	11	23	3	13	15	15	28	15	23	19	2	21	12	16	8	14.93
AR-SVR	31	35	32	29	28	16	19	29	34	20	36	32	26	32	28	28.47
BG-M5P	5	9	11	6	3	17	3	30	11	16	28	18	31	17	33	15.87
BG-REP	19	17	12	20	16	18	20	16	27	32	8	29	13	21	34	20.13
BG-PLS	6	1	6	4	11	3	4	6	5	2	34	10	4	8	3	**7.133**
BG-SLR	20	24	25	34	31	26	13	22	18	21	20	6	32	9	9	20.67
BG-DS	21	13	13	14	4	27	30	17	24	29	14	22	9	3	29	17.93
BG-NN	12	14	1	15	12	4	21	7	12	17	3	7	3	18	10	10.4
BG-SVR	32	34	26	36	34	28	22	27	29	9	15	11	30	33	18	25.6
RS-M5P	13	10	7	9	27	19	5	8	10	10	9	12	27	22	26	14.27
RS-REP	22	18	20	16	17	20	23	18	26	30	1	26	18	12	30	19.8
RS-PLS	7	2	18	1	5	5	6	2	6	5	26	13	5	4	11	**7.733**
RS-SLR	23	25	27	21	13	29	14	23	19	18	10	8	19	13	19	18.73
RS-DS	24	19	33	10	1	30	33	24	25	31	21	30	14	5	16	21.07
RS-NN	14	15	8	11	18	6	24	9	15	12	4	23	6	14	12	12.73
RS-SVR	33	27	19	24	24	21	7	25	7	13	11	24	25	34	13	20.47
M5P	25	28	21	17	25	22	15	26	16	22	29	2	35	24	20	21.8
REP	34	29	14	31	26	33	34	28	30	33	16	34	15	26	27	27.33
PLS	8	3	15	3	19	7	8	10	1	3	33	14	10	6	2	9.467
SLR	27	32	35	27	23	31	16	31	28	25	24	3	36	25	21	25.6
DS	28	33	30	22	6	34	35	32	31	35	22	35	22	27	36	28.53
NN	26	20	4	18	20	23	25	19	20	23	17	19	7	19	14	18.27
SVR	35	36	34	30	29	35	29	34	32	24	35	33	28	31	31	31.73
Zero0	36	30	9	28	21	36	36	35	36	36	5	36	29	23	35	28.73

Table 6. The averaged success rankings of the algorithms on the original datasets (best to worst, 1 to 36).

	RF	AR	BG	RS	Single	Avg.
M5P	22.67	12.13	17.47	21.40	21.07	18.95
REP	9.87	17.87	12.80	14.07	22.80	15.48
PLS	22.33	6.00	7.33	13.07	11.87	12.12
SLR	23.60	19.60	23.27	23.80	25.80	23.21
DS	10.67	13.47	14.53	19.00	25.40	16.61
NN	9.20	14.07	10.00	11.53	18.27	12.61
SVR	32.33	30.33	29.40	28.00	29.47	29.91
Avg.	18.67	16.21	16.4	18.70	22.10	

Table 7. The averaged success rankings of the algorithms on the dimensionally reduced datasets (best to worst, 1 to 36).

	RF	AR	BG	RS	Single	Avg.
M5P	16.73	15.20	15.87	14.27	21.80	16.77
REP	14.53	27.07	20.13	19.80	27.33	21.77
PLS	13.53	9.73	7.13	7.73	9.47	9.52
SLR	15.07	19.73	20.67	18.73	25.60	19.96
DS	14.13	19.40	17.93	21.07	28.53	20.21
NN	7.80	14.93	10.40	12.73	18.27	12.83
SVR	25.67	28.47	25.60	20.47	31.73	26.39
Avg.	15.35	19.22	16.82	16.40	23.25	

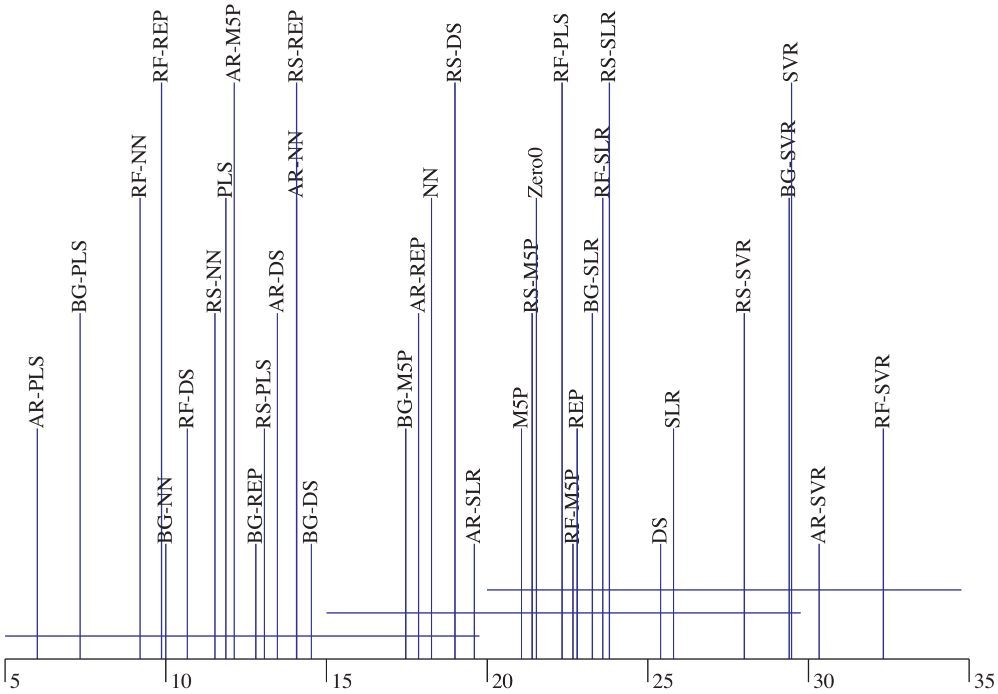

Figure 1. Graphical representation of the Nemenyi test results of the compared methods with the ranks given in Table 6 (on original datasets). The numbers on the line represent the average ranks. Bold lines connect the algorithms that have no significant difference.

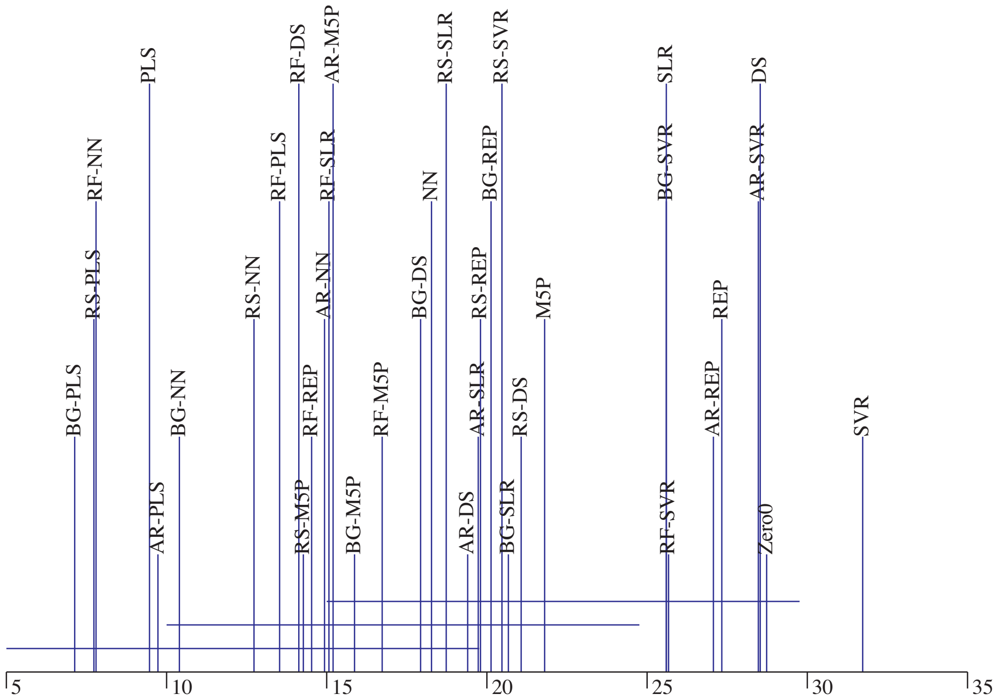

Figure 2. Graphical representation of the Nemenyi test results of the compared methods with the ranks given in Table 7 (on dimensionally reduced datasets).

- The best performed base algorithm is partial least squares (PLS).

- Additive regression and bagging increase the performance of each base algorithm. Rotation forest increases the performances of REP, decision stump (DS), and nearest neighbor (NN). It decreases the performance of partial least squares. Random subspace (RS) generally increases performance.

- The M5P, PLS, and SLR base algorithms had their best performances with additive regression. REP and the DS algorithm with rotation forest, and the SVR algorithm with random subspace, had their best performances.

- Rotation forest and random subspace had their best performances with NN. Additive regression and bagging with PLS had their best performances.

- According to the Nemenyi test, there is no statistical difference between the best algorithm (AR-PLS) and the algorithms having average ranks below 20.76 (= 6.00 + 14.76).

For the experiments with the dimensionally reduced datasets (Tables 5 and 7):

- The best performance (7.13) is obtained with the BG-PLS algorithm.

- The best performing ensemble algorithm is rotation forest.

- The best performing base algorithm is partial least squares.

All of the ensemble algorithms generally increased the performance of each base algorithm. The exceptions are AR-PLS and RF-PLS.

- The M5P and SVR base algorithms had their best performances with random subspace. The REP, SLR, DS, and NN algorithms with rotation forest, and PLS with bagging, achieved their best performances.

- Rotation forest had its best performances with NN. Additive regression, random subspace, and bagging with PLS had their best performances.

- According to the Nemenyi test, there is no statistical difference between the best algorithm (BG-PLS) and the algorithms having average ranks below 21.89 (= 7.13 + 14.76).

The average successes of the algorithms were investigated above. Next, the best performing algorithm will be investigated over each individual dataset. In Table 8, the dataset name, the error of the Zero Rule algorithm, and the error and the name of the best performing algorithm are shown for the original and dimensionally reduced datasets.

The Zero Rule predicts a single value for all of the test samples. This value is the mean value of all of the training samples' outputs. It only considers the outputs of the samples. It can be thought of as the default error of a dataset. Thus, the Zero Rule errors are the same for the original and dimensionally reduced datasets.

Comparing the Zero Rule error and other algorithms errors shows whether the algorithms can decrease the default error.

Table 8. The best performing algorithms on the original and dimensionally reduced datasets.

Dataset name	Zero Rule error	With all of the features		With the selected features	
		Best performing algorithm	RMSE	Best performing algorithm	RMSE
benzo	0.25	RF-M5P	0.21	RF-M5P	0.21
carbolenes	0.23	RF-DS	0.22	BG-PLS	0.15
chang	0.20	Zero0	0.20	BG-NN	0.18
cristalli	0.28	RS-NN	0.24	RS-PLS	0.18
depreux	0.20	RF-REP	0.20	RS-DS	0.16
mtp	0.18	AR-PLS	0.16	RF-NN	0.15
pah	0.20	AR-PLS	0.10	RF-PLS	0.10
pdgfr	0.23	RF-REP	0.20	RF-PLS	0.17
phen	0.27	AR-PLS	0.13	PLS	0.14
phenetyl	0.27	PLS	0.10	RF-PLS	0.06
qsbr_y	0.27	BG-NN	0.25	RS-REP	0.26
qsfsr	0.27	AR-M5P	0.19	AR-M5P	0.17
selwood	0.30	RF-DS	0.25	RF-NN	0.21
strupcz	0.22	RF-NN	0.21	RF-DS	0.16
yokohoma	0.28	RF-DS	0.27	RF-PLS	0.20

When Table 8 is investigated, the following conclusions are reached:

- The best performing algorithms are generally ensemble algorithms. This is in agreement with the average success of the algorithms.

- The experiments with dimensionally reduced datasets have equal or better results than the original datasets, except for 2 datasets (phen, qsbr_y).

- The dimension reduction process changes the best performing algorithm, except for 2 datasets (benzo, qsfrs).

The experiments with dimensionally reduced datasets were further investigated in detail. The results of the best 10 algorithms and the Zero Rule are compared using the paired t-test [24]. In Table 9, the wins and significant wins are shown between each pair of these 11 algorithms. The results are given in X(Y) form, which means that the algorithm in the corresponding row has better results at X datasets out of 15 than the algorithm in the corresponding column. The number in brackets (Y) represents the number of significant wins for the row with regard to the column. A 0 means that the scheme in the corresponding column did not score a single (significant) win with regard to the scheme in the row. For example, the RF-PLS algorithm has a better result than the Zero Rule for 10 datasets, and the differences for 5 out of 10 datasets are significant.

Table 9. The significant differences of the algorithms' performances.

	RF-PLS	RF-DS	RF-NN	AR-PLS	BG-PLS	BG-NN	RS-M5P	RS-PLS	RS-NN	PLS	ZeroR
RF-PLS	-	9(2)	7(0)	7(0)	4(0)	7(0)	9(0)	5(0)	8(0)	2(0)	10(5)
RF-DS	6(0)	-	4(0)	5(0)	3(0)	5(0)	7(0)	4(0)	5(0)	4(0)	13(6)
RF-NN	8(0)	11(0)	-	5(0)	3(0)	7(0)	8(0)	3(0)	9(0)	4(0)	14(5)
AR-PLS	8(0)	10(4)	10(0)	-	5(0)	9(0)	12(0)	3(0)	10(1)	4(0)	12(6)
BG-PLS	11(0)	12(3)	12(0)	10(0)	-	11(0)	12(0)	4(0)	12(0)	7(0)	14(7)
BG-NN	8(0)	10(1)	8(0)	6(0)	4(0)	-	8(0)	4(0)	8(0)	5(0)	15(7)
RS-M5P	6(0)	8(2)	7(0)	3(0)	3(0)	7(0)	-	2(0)	7(0)	2(0)	11(5)
RS-PLS	10(0)	11(3)	12(0)	12(0)	11(0)	11(0)	13(0)	-	13(0)	9(0)	13(7)
RS-NN	7(0)	10(0)	6(0)	5(0)	3(0)	7(0)	8(0)	2(0)	-	4(0)	15(6)
PLS	13(0)	11(2)	11(1)	11(0)	8(0)	10(0)	13(0)	6(0)	11(0)	-	13(7)
ZeroR	5(0)	2(0)	1(0)	3(0)	1(0)	0(0)	4(0)	2(0)	0(0)	2(0)	-

When Table 9 is investigated, the following conclusions are reached:

- The BG-PLS, BG-NN, RS-PLS, and PLS algorithms are the most significantly winning algorithms over the Zero Rule (at 7 datasets).

- The RF-PLS, AR-PLS, BG-PLS, BG-NN, RS-M5P, RS-PLS, and PLS algorithms have no significant losses.

- The AR-PLS algorithm has the biggest significant winning number (11).

In Figures 3 and 4, the hierarchical clusters of the algorithms and datasets are given, respectively. The closeness of the connection point of the clusters to the left side directly represents the similarity of the algorithms/datasets.

When the algorithms are clustered, the algorithms are represented by points having 15 (the number of datasets) features (dimensions). When the datasets are clustered, the datasets are represented by points having 36 (the number of algorithms) features (dimensions).

According to Figure 3, the following conclusions are reached:

- In both figures, the ensemble–algorithm pairs are generally clustered with their base single algorithms.

- The feature selection process does not affect the similarities of the algorithms dramatically.

According to Figure 4, the following conclusions are reached:

- On the left side of Figure 4, the datasets having 1142 features are generally clustered together.

- On the right side of Figure 4, there is no obvious pattern between the clusters and the number of features/samples.

4. Previous works

The selected previous studies in this area for both classification and regression are shown comparatively in Table 10. It is observed that a larger number of datasets was used in the classification problems. However, the number of chemical/drug design datasets used is not sufficient to reach general conclusions.

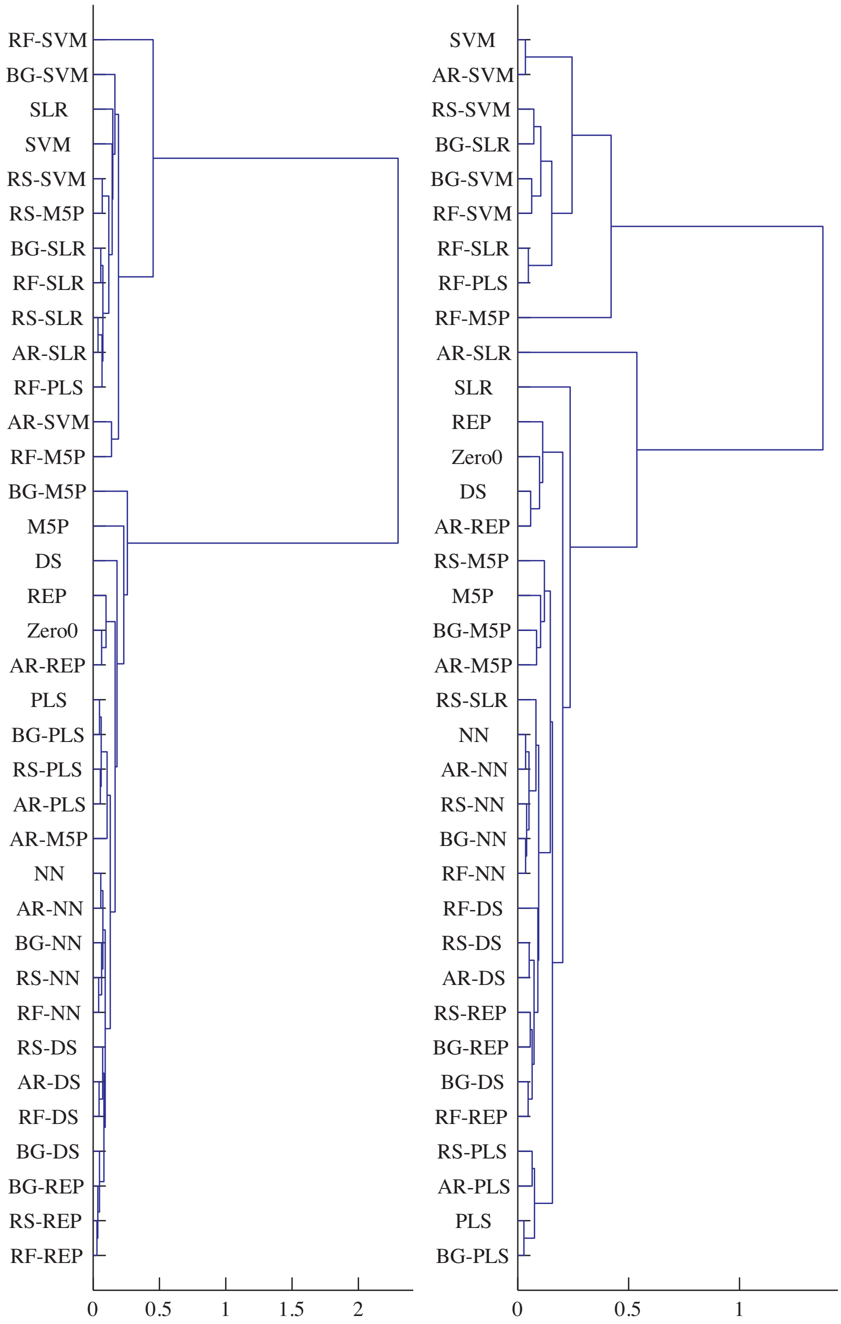

Figure 3. The hierarchical clusters of the algorithms according to their RMSE values on the original (left) and dimensionally reduced (right) 15 datasets.

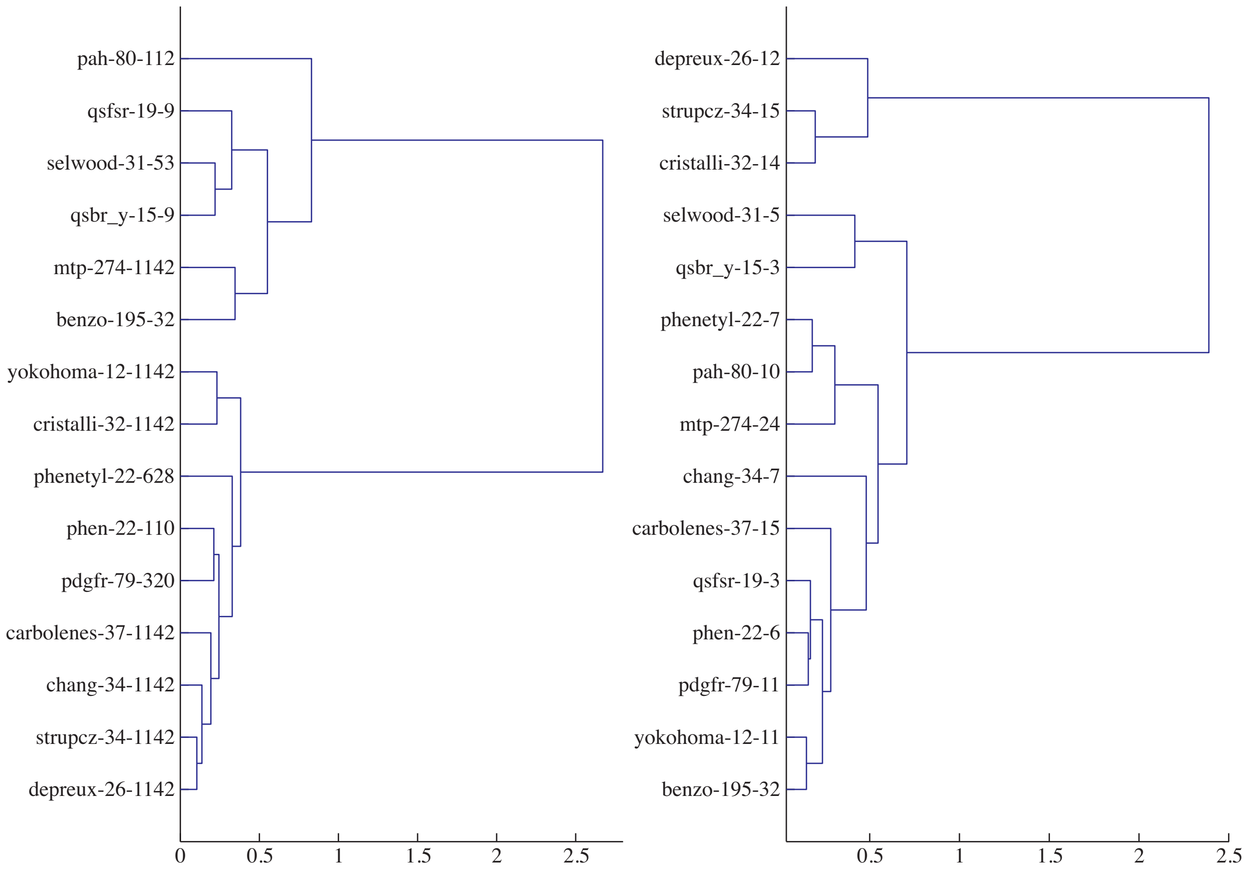

Figure 4. The hierarchical clusters of the original (left) and dimensionally reduced (right) 15 datasets according to their RMSE values obtained with 36 algorithms. In the figures, the dataset names, the number of features, and the samples are given.

According to Table 10, together with our experiments, the following conclusions are reached:

- The number of used drug design/chemical datasets in our experiments is larger than those in previous works.

- The success of PLS in our experiments verifies the high usage of PLS in previous studies.

- The superior success of ensemble algorithms over single algorithms is confirmed.

Table 10. Previous works.

Reference	Compared methods in the study	Datasets	Results
[8]	PLS, BG with PLS, PLS ensemble with and without noise	The datasets are generated from one regression-type near-infrared (NIR) datum with several types of additive noise.	Noise ensemble PLS is better than regular PLS. BG does not seem to give any improvement over PLS.
[25]	Kernel PLS (KPLS), PLS, PLS BG, PLS boosting, KPLS BG, KPLS boosting,	2 regression-type NIR datasets.	KPLS is better than PLS. BG and boosting have no significant effect on KPLS and PLS.
[26]	Boosting, random forest, decision tree, PLS, KNN, SVR	4 regression, 6 classification datasets (chemical data)	Boosting and random forest are better than other algorithms.
[27]	SVR, SVR ensembles, RS KNN, ridge regression	2 chemical classification-type datasets	Single SVR and SVR ensembles are better than others.
[28]	One base learner (multilayer perception). BG, ensemble with full and partial samples.	4 chemical regression-type datasets	Ensembles with full samples are better than having BG sample ones.
[29]	Decision tree, BG, boosting, random forest, SVR	8 chemical classification-type datasets	SVR and random forest are better than the other algorithms.
[30]	One base learner (C4.5). boosting, RS, random trees, BG, random forest	34 University of California - Irvine (UCI) classification datasets	All of the ensembles are better than a single C4.5, but no algorithm is significantly better than BG. The best performing algorithm is RF.
[3]	One base learner (C4.5). BG, boosting, randomization	32 UCI classification datasets	On original datasets: boosting > BG = randomization. On datasets with class noise, BG is the best.
[31]	BG, boosting, randomized C4.5	57 UCI classification datasets	Boosting, random forest, and randomized trees are better performers than BG.

5. Conclusions

In machine learning, committee algorithms (ensembles), especially those with classification applications, are highly popular because they have better performances than single algorithms.

In this study, the comparative performances of algorithm ensembles with drug design datasets in regression applications were investigated. A drug design dataset collection with 15 regression-type datasets was used for this purpose. We obtained the performances of the single algorithms and the algorithm ensembles on those datasets. The combinations of 7 base algorithms and 4 ensemble algorithms were investigated.

In Table 11, conclusions are given in the form of questions that we tried to answer and the answers obtained from our experiments.

Table 11. The questions and their answers obtained with the experimental studies on drug datasets.

Question	Answer (based on our drug design experiments)
Do the ensemble algorithms generate more successful results than a single algorithm?	Generally, yes.
How are the most successful ensemble algorithms ranked?	Success ranking in original datasets: AR > BG > RF > RSs > single. In dimensionally reduced datasets: RF > RSs > BG > AR > single.
What is the base algorithm–ensemble pair having the best results?	In original datasets: AR with PLS. In dimensionally reduced datasets: BG with PLS.
Which ensemble algorithm works well with which base algorithms?	In original datasets: RF and RS work well with NN. AR and BG with PLS had their best performances. The best single algorithm is PLS. In dimensionally reduced datasets: AR, RS, and BG with PLS had their best performances. RF works well with NN. The best single algorithm is PLS.
Which base algorithm works well with which ensemble algorithms?	In original datasets: M5P, PLS, and SLR work well with AR. REP and DS algorithm with RF, and SVR algorithm with RS, had their best performances. In dimensionally reduced datasets: M5P and SVR work well with RS. REP with BG; SLR and DS with RF; REP, SLR, DS, and NN algorithms with RF; and PLS with BG had their best performances.
What are the similarities of the algorithms according to their performances?	The ensemble–algorithm pairs are mainly grouped with the base algorithm. This shows that the performance of an experiment is determined by the base algorithms, not the ensemble algorithm.

References

[1] G. Brown, J.L. Wyatt, P. Tino, "Managing diversity in regression ensembles", Journal of Machine Learning Research, Vol. 6, pp. 1621–1650, 2005.

[2] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, 2nd Edition, San Francisco, Morgan Kaufmann, 2005.

[3] T.G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization", Machine Learning, Vol. 40, pp. 139–157, 1998.

[4] J.H. Friedman, "Greedy function approximation: a gradient boosting machine", Technical Report, Department of Statistics, Stanford University, 1999.

[5] T.K. Ho, "The random subspace method for constructing decision forests", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, pp. 832–844, 1998.

[6] J.J. Rodríguez, L.I. Kuncheva, C.J. Alonso, "Rotation forest: a new classifier ensemble method", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, pp. 1619–1630, 2006.

[7] E. Frank, Y. Wang, S. Inglis, G. Holmes, I.H. Witten, "Using model trees for classification", Machine Learning, Vol. 32, pp. 63–76, 1998.

[8] B.H. Mevik, V.H. Segtnan, T. Næs, "Ensemble methods and partial least squares regression", Journal of Chemometrics, Vol. 18, pp. 498–507, 2004.

[9] S.K. Shevade, S.S. Keerthi, C. Bhattacharyya, K.R.K. Murthy, "Improvements to the SMO algorithm for SVM regression", IEEE Transactions on Neural Networks, Vol. 11, pp. 1188–1193, 2000.

[10] M.A. Hall, "Correlation-based feature selection for machine learning", PhD, Department of Computer Science, University of Waikato, 1998.

[11] ADRIANA.Code, Molecular Networks, Germany; www.mol-net.de.

[12] WEKA Collections of Datasets; www.cs.waikato.ac.nz/ml/weka/index_datasets.html.

[13] M. Karthikeyan, R.C. Glen, A. Bender, "General melting point prediction based on a diverse compound dataset and artificial neural networks", Journal of Chemical Information and Modeling, Vol. 45, pp. 581–590, 2005.

[14] B.D. Silverman, E. Daniel., J. Platt, "Comparative molecular moment analysis (CoMMA): 3D-QSAR without molecular superposition", Journal of Medicinal Chemistry, Vol. 39, pp. 2129–2140, 1996.

[15] D.E. Patterson, R.D. Cramer, A.M. Ferguson, R.D. Clark, L.W. Weinberger, "Neighborhood behavior: a useful concept for validation of molecular diversity descriptors", Journal of Medicinal Chemistry, Vol. 39, pp. 3049–3059, 1996.

[16] R. Todeschini, P. Gramatica, E. Marengo, R. Provenzani, "Weighted holistic invariant molecular descriptors", Chemometrics and Intelligent Laboratory Systems, Vol. 27, pp. 221–229, 1995.

[17] R. Guha, P. Jurs, "The development of linear, ensemble and non-linear models for the prediction and interpretation of the biological activity of a set of PDGFR inhibitors", Journal of Chemical Information and Computer Sciences, Vol. 44, pp. 2179–2189, 2004.

[18] A. Cammarata, "Interrelationship of the regression models used for structure-activity analyses", Journal of Medicinal Chemistry, Vol. 15, pp. 573–577, 1972.

[19] H. Kubinyi, QSAR: Hansch Analysis and Related Approaches, New York, VCH Publishers/Weinheim, VCH Verlagsgesellschaft, pp. 57–68, 1993.

[20] J. Damborský, K. Manova, M. Kuty, Biodegradability Prediction, Dordrecht, Kluwer Academic Publishers, pp. 75–92, 1996.

[21] J. Damborský, "Quantitative structure-function and structure-stability relationships of purposely modified proteins", Protein Engineering, Vol. 11, pp. 21–30, 1998.

[22] D.L. Selwood, D.J. Livingstone, J.C. Comley, A.B. O'Dowd, A.T. Hudson, P. Jackson, K.S. Jandu, V.S. Rose, J.N. Stables, "Structure-activity relationships of antifilarial antimycin analogues: a multivariate pattern recognition study", Journal of Medicinal Chemistry, Vol. 33, pp. 136–142, 1990.

[23] J. Demsar, "Statistical comparisons of classifiers over multiple data sets", Journal of Machine Learning Research, Vol. 7, pp. 1–30, 2006.

[24] D.W. Zimmerman, "A note on interpretation of the paired-samples t test", Journal of Educational and Behavioral Statistics, Vol. 22, pp. 349–360, 1997.

[25] H. Shinzawa, J.H. Jiang, P. Ritthiruangdej, Y. Ozaki, "Investigations of bagged kernel partial least squares (KPLS) and boosting KPLS with applications to near-infrared (NIR) spectra", Journal of Chemometrics, Vol. 20, pp. 436–444, 2006.

[26] V. Svetnik, T. Wang, C. Tong, A. Liaw, R.P. Sheridan, Q. Song, "Boosting: an ensemble learning tool for compound classification and QSAR modeling", Journal of Chemical Information and Modeling, Vol. 45, pp. 786–799, 2005.

[27] C. Merkwirth, H. Mauser, T. Schulz-Gasch, O. Roche, M. Stahl, T. Lengauer, "Ensemble methods for classification in cheminformatics", Journal of Chemical Information and Computer Sciences, Vol. 44, pp. 1971–1978, 2004.

[28] D.K. Agrafiotis, W. Cedeño, V.S. Lobanov, “On the use of neural network ensembles in QSAR and QSPR”, Journal of Chemical Information and Computer Sciences, Vol. 42, pp. 903–911, 2002.

[29] C.L. Bruce, J.L. Melville, S.D. Pickett, J.D. Hirst, “Contemporary QSAR classifiers compared”, Journal of Chemical Information and Modeling, Vol. 47, pp. 219–227, 2007.

[30] R.E. Banfield, L.O. Hall, K.W. Bowyer, D. Bhadoria, W.P. Kegelmeyer, S. Eschrich, “A comparison of ensemble creation techniques”, The 5th International Conference on Multiple Classifier Systems, pp. 223–232, 2004.

[31] R.E. Banfield, L.O. Hall, K.W. Bowyer, W.P. Kegelmeyer, “A comparison of decision tree ensemble creation techniques”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, pp. 173–180, 2007.