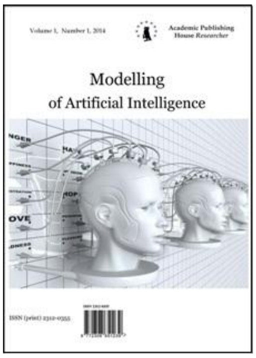
Published in the Russian Federation
Modeling of Artificial Intelligence
Has been issued since 2014.
ISSN: 2312-0355
E-ISSN: 2413-7200
Vol. 11, Is. 3, pp. 140-150, 2016

DOI: 10.13187/mai.2016.11.140
www.ejournal11.com



UDC 004

Comparison of Ensemble Classification Techniques and Single Classifiers Performance for Customer Credit Assessment

Adnan Dželihodžić [a, *] , Dženana Đonko [a]

[a] International Burch University, Bosnia and Herzegovina

Abstract

Assuming that the credit is one of the most important banking products it follows that the quality assessment of customer creditworthiness is an essential factor for reducing the risk. With the intention to make a good assessment of creditworthiness many models and algorithms have been developed. Data mining algorithms for classification are very suitable for determining the validity of the application for credit. This paper presents an analysis of the effectiveness of the algorithms for classification of credit applications when they are used alone (as single classifier) as well as comparison with ensemble techniques usage. The techniques used as single classifiers are Neural Networks, Decision Trees and Support Vector Machines (SVM), and ensemble techniques AdaBoost and Bagging. K-fold cross-validation is used for model validation. Experiment is conducted in the Bosnian commercial bank dataset and results according to classification parameters such as accuracy and AUC are presented.

Keywords: classification, data mining, credit assessment, ensemble techniques.

1. Introduction

Global economic crisis has affected all aspects of economy. Financial risk management and credit risk evaluation has become very important. Besides economic crises, credit risk assessment is also important because of credit industry growing. Accordingly, many credit scoring models are developed in order to facilitate credit admission decision. Every credit scoring model goal is to categorize applicants into at least two groups ('bad' or 'good') with high classification accuracy. Even a fraction of a percent increase in credit classification accuracy is a significant achievement (West, 2000). Classification accuracy is the most important evaluation criteria, but there are another important measure metrics such as accuracy, specificity, sensitivity and AOC (Area under Curve – ROC). In order get better performance and improve credit risk assessment task accuracy, many models have been developed using a variety of techniques for classification. Initially these were statistical techniques, but in recent decade machine learning techniques are increasingly used. These techniques can be categorized into the following groups:

1. Single classifiers – This group represents credit scoring models with single classifiers. It means that only one classification technique is used in model. Also, this group can be divided into this 3 subgroups: a) statistical – discriminant analysis (LDA) (Fisher, 1936), (Chuang, Lin,

* Corresponding author
E-mail addresses: adnandz@gmail.com (A. Dželihodžić), ddonko@etf.unsa.ba (D. Đonko)

2009), (Yu et al., 2011); logistic regression analysis (Wiginton, 1980), (Hosmer, Lemeshow, 1989), (Yap et al., 2011) b) machine learning – decision trees (Lee et al., 2006), (Li, 2006), (Kvesić, 2013); neural networks (Jensen, 1992), (Khashman, 2011), (Bekhet, Eletter, 2014); k-nearest neighbor (Dasarathy, 1991), (Twala, 2010) and support vector machines (Wang et al., 2005), (Harris, 2013), (Tang, Qiu, 2012) c) genetic algorithm (Oreski et al., 2012).

2. Ensemble techniques – Multiple classifiers represents combination of individual classifiers in order to improve classification performance. A classifier ensemble (also referred to as committee of learners, mixture of experts, multiple classifier system) consists of a set of individually trained classifiers (base classifiers) whose decisions are combined in some way, typically by weighted or unweighted voting when classifying new examples

According to (Kuncheva, 2003) there are four basic approaches: (i) using different combination schemes, (ii) using different classifier models, (iii) using different attribute subsets, and (iv) using different training sets.

Examples are combinations of multiyear perceptron, decision tree and support vector machines (Hung, Chen, 2009) or logistic regression and support vector machine (Hua et al., 2007).

3. Hybrid classifiers – combination two or more heterogeneous machine learning techniques. Three different ways to build hybrid classifiers are: cascading different classifiers (neuro-fuzzy inference system (Malhotra, Malhotra, 2002), multilayer perceptron and linear discriminant analysis (Lee et al., 2002), cluster + single classifiers (Self-organizing map + multilayer perceptron (Lee et al., 1996) and integrated-based (genetic algorithm + support vector machines (Huang et al., 2007), fuzzy support vector machines (Tang, Qiu, 2012), etc.)

All mentioned application of machine learning techniques are just small part of current researches and applications. Reasons for such a large number of different credit scoring models is that there is still no universal solution for classification. Usually it is necessary to provide some preprocessing actions to improve machine learning classifier performance

In this paper, re-sampling ensemble techniques Bagging and AdaBoost were conducted and compared against single classifiers used alone. The organization of the remainder of the paper is as follows. First section describes single classifiers: Decision trees, neural networks and SVM. In the second section ensemble techniques briefly overview was given. Experiment setup, dataset description and evaluation criteria are presented in the next sections. Detailed results and findings are described and presented in the penultimate section.

2. Discussion

Single classifiers

Decision trees

Classification And Regression Tree (CART) is introduced by (Breiman, 1996), traditionally involves two phases: growing and pruning. In the growing phase, the input domain is recursively partitioned into cells. Each cell corresponds to a leaf of a large initial tree. The partitioning is often done to fit the data as closely as possible. The initial tree perhaps fit the data perfectly, but the generalization is suboptimal. To improve the generalization and avoid over fitting, the initial tree is pruned. For classification, let *T* denotes the set consisting of the initial tree and all possible pruning of this tree, CART selects the tree in *T* that minimizes

$$C(T) = \widehat{L_n}(T) + \alpha|T| \quad (1)$$

where $\widehat{L_n}(T)$ is the empirical risk using tree T, |T| is the cardinality of the tree and $\alpha > 0$ is a constant that controls the trade-off between fidelity to the training data and the complexity of the tree (Chuang, Lin, 2009).

Measuring of split quality implies calculation of impurity function and impurity is often form of the entropy or GINI Index.

Entropy: $I(N) = -p\log p - (1-p)\log(1-p)$ (2)

Gini index: $I(N) = 2p(1-p)$ (3)

Value of p is a fraction of observations with positive response in the node N (Izenman, 2008).

Neural networks

Neural networks or artificial neural networks (ANN) are mathematical representations inspired by functioning of human brain. ANNs that are used to solve the problem of credit evaluation can be regarded as a statistical method, which transform linear combination variables with a non-linear manner and then recycling the process. There are two stages associated with the back-propagation method: training and classification. The ANN is trained by supplying it with a large number of numerical observations or the patterns to be learned (input data pattern) whose corresponding classifications (target values or desired output) are known. During training, the final sum-of-squares error over the validation data for the network is calculated. The selection of the optimum number of hidden nodes is made on the basis of this error value. The question of how to choose the structure of the network is beyond the scope of this thesis and is a current research issue in neural networks. Once the network is trained, a new object is classified by sending its attribute values to the input nodes of the network, applying the weights to those values, and computing the values of the output units or output unit activations. The assigned class is that with the largest output unit activation (Twala, 2010).

Generally, they have ability to deal with complicated problems and there are many types of ANNs.

Support Vector Machines

The Support Vector Machine (SVM) was first developed by (Cortes, Vapnik, 1995) for binary classification. To achieve this, the algorithm attempts to find the optimal separating hyperplane between classes by maximizing the class margin. If we have a training dataset $\{x_i, y_i\}$ (i=1…N) where x_i are input and y_i are corresponding observed binary variable (output or class). Decision boundary is given by $\boldsymbol{\omega}\,\boldsymbol{x} + b = 0$, and it is necessary to find maximum margin.

The separate hyper plane can be represented as follows (Yu et al., 2007):

$$z(x) = \omega^T \phi(x) + b = 0 \tag{4}$$

where $\boldsymbol{\omega}$ is the normal vector of the hyper plane and b is the bias that is a scalar.

Margin is maximum distance of decision boundary from the data of both classes. Support Vectors are those data points that the margin pushes up against.

If we have nonlinear input space it is impossible to find hyperplane separator.

Optimal solution for the weight vector is given by

$$\omega = \sum_{i=1}^{N_s} \alpha_i y_i \phi(x_i) \tag{5}$$

where Ns is the number of SVs.

Once the optimal pair (w, b) is determined, the decision function of SVM is obtained as

$$Z(x) = sign\left(\sum_{i=1}^{N_s} \alpha_i y_i K(x_i, x_j) + b\right) \tag{10}$$

where $K(x_i, x_j)$ is the kernel function in the input space that computes the inner product of two data points in the feature space.

Kernel functions are:

- Linear: $k(x, y) = x^T y + c$
- Polynomial: $k(x, y) = (x^T y + 1)^d$
- Sigmoid (MLP): $k(x, y) = \tanh(\alpha x^T y + c)$
- Radial Basis Function: $k(x, y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$
- Cauchy: $k(x, y) = \left(1 + \frac{\|x-y\|^2}{\sigma^2}\right)^{-1}$
- Log: $k(x, y) = -\log(\|x - y\|^d + c)$

Ensemble techniques

Bagging

In its standard form, the bagging (Bootstrap Aggregating) algorithm (Breiman, 1996) creates M bootstrap samples $T_1, T_2, \ldots, T_M$ randomly drawn (with replacement) from the original training set T of size n. Each bootstrap sample T_i of size n is then used to train a base classifier C_i.

Predictions on new observations are made by taking the majority vote of the ensemble C built from $C_1, C_2, \ldots, C_M$.

As bagging resamples the training set with replacement, some instances may be represented multiple times while others may be left out. It reduces complexity of a model class prone to overfit by resample the data many times and generate a predictor on that resampling.

AdaBoost

Similar to bagging, boosting also creates an ensemble of classifiers by resampling the original data set, which are then combined by majority voting. However, in boosting, resampling is directed to provide the most informative training data for each consecutive classifier. Thus, instances misclassified by model C_{i-1} are more likely to appear in the next bootstrap sample T_i. The final decision is then obtained through a weighted vote of the base classifiers (the weight wi of each classifier C_i is computed according to its performance on the weighted sample T_i it was trained on)

Experiment setup

Three different experiment setup is provided in this paper. For each setup single classifier, Bagging and AdaBoost technique is applied. Single classifiers: Decision tree, neural network and Support Vector Machine (SVM) were used as single and in ensemble technique. In the ensemble techniques (Bagging and AdaBoost) same parameter setup for base classifier is used. K-fold cross-validation with 10 folds is used to split data on test and training and avoid overfitting. Experiments were conducted in the Bosnian dataset and results according to main classification parameters such as accuracy and AUC are presented. Below tables presents experiment setup which is actually algorithm's parameters setup.

Table 1. Experiment setup 1 (ES1)

DECISION TREE		NEURAL NETWORKS		SVM	
Criterion	gain ratio	Training cycles	1000	Kernel type	radial
Maximal depth	20	Learning rate	0.01	Gamma	1
Confidence	0.25	Momentum	0.4	C	0
Apply pruning	TRUE			Convergence epsilon	0.001
Minimal gain	0.1				
Minimal leaf size	2				

Table 2. Experiment setup 2 (ES2)

DECISION TREE		NEURAL NETWORKS		SVM	
Criterion	information gain	Training cycles	500	Kernel type	dot
Maximal depth	20	Learning rate	0.3	Gamma	1
Confidence	0.25	Momentum	0.2	C	0
Apply pruning	TRUE			Convergence epsilon	0.001
Minimal gain	0.1				
Minimal leaf size	2				

Table 3. Experiment setup 3 (ES3)

DECISION TREE		NEURAL NETWORKS		SVM	
Criterion	Gini index	Training cycles	500	Kernel type	polynomial
Maximal depth	20	Learning rate	0.6	Kernel degree	2
Confidence	0.25	Momentum	0.01	C	0
Apply pruning	TRUE			Convergence epsilon	0.001
Minimal gain	0.1				
Minimal leaf size	2				

The previous table shows the settings of algorithms that are used in experiments.

All experiments are conducted in RapidMiner software (https://rapidminer.com/ products/studio/). In Decision tree, we select the criterion on which attributes will be selected for splitting. It can have one of the following values (Oreski et al., 2012):

- information_gain: The entropy of all the attributes is calculated. The attribute with minimum entropy is selected for split. This method has a bias towards selecting attributes with a large number of values.
- gain_ratio: It is a variant of information gain. It adjusts the information gain for each attribute to allow the breadth and uniformity of the attribute values.
- gini_index: This is a measure of impurity of an ExampleSet. Splitting on a chosen attribute gives a reduction in the average gini index of the resulting subsets.

Neural networks parameters are selected by using grid search algorithm for parameter optimization. Experiment setup 2 neural networks parameters default parameters values selected. For Support Vector Machine (SVM) different kernels selected for each setup and default values of other parameters such as complexity constant C that means a boundary sensitivity.

Each of this single classifiers settings remain the same in Bagging and AdaBoost algorithms.

Dataset

Bosnian commercial bank customer credit application dataset is used in experiments. It consist of 1147 examples (742 Good and 405 Bad), and 22 features/attributes. Features are shown in the following table.

Table 4. Dataset attributes

Attribute	Description
Existing client	True/false – Does applicant is client already or not?
Age	Numeric – Applicant's age
Client year	Numeric – Number of years in the bank as a client
Client salary	
Client savings	True/false – Does applicant have savings in the bank?
Sex	Numeric – Applicant's sex (male/female)
Marital status	Applicant's marital status with possible values: 1- Married or cohabiting; 2- Single; 3- Separated;4-Widowed
Education	Applicant's education with possible values: 1 – High school; 2- University degree; 3- Unknown
Occupational group	Applicant's occupational group with possible values: 1- Self-employed; 2- State; 3- Salaried employee; 4-Bank's group employee
Employed since	Numeric - number of years of experience

Industry	Economic sector where applicant is employed. Possible values: Construction industry/trade/mining/scaffolding; Public service; Retail trade; Electronics/precision engineering/optics; Bank, service, insurance; Gastronomy; Other
Type of Residence	Status of applicant's residence with possible values: Own home; Own apartment; With parents; Sub-tenancy
Monthly Income	Total monthly income of the applicant
Number of Dependants	Number of applicant's dependents
Monthly Expenses	Total monthly expenses
Housing Costs	Applicant's housing costs
Living Costs	Applicant's living costs
Existing credit loan installments	Applicant's existing credit loan installments
Fictitious credit loan installments	Applicant's fictitious credit loan installments
Loan Amount	Requested loan amount
Long Term	Number of months for repayment of loans
Interest rate	Interest amount

Evaluation criteria

Standard performance evaluation criteria in the fields of credit scoring include accuracy, error rate, Gini coefficient, Kolmogorov–Smirnov statistic, mean squared error, area under the ROC curve, and type-I and type-II errors. Accuracy, AUC (area under ROC curve), Specificity (precision) and Sensitivity (recall) were used as the main evaluation criteria.

Most credit scoring applications often employ the accuracy as the criterion for performance evaluation. It represents the proportion of the correctly predicted cases (good and bad) on a particular data set. Because credit data are commonly imbalanced, the area under the ROC curve (AUC) has been suggested as an appropriate performance evaluator without regard to class distribution or misclassification costs (Jensen, 1992).

K-fold cross-validation is used to reduce the bias related with random sampling of the training and test sets. The cross-validation accuracy (CVA) is the average of the k individual accuracy measures

$$CVA = \frac{1}{k}\sum_{j=1}^{k} A_i \quad (6)$$

where k is the number of folds used, and A_i is the accuracy measure of each fold, $i = 1, \ldots, k$ (Zhang et al., 2006).

The number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) are used in evaluating the performance of a classifier. Different terms are used in different domains. The sensitivity and specificity are widely used in credit application classifications. Sensitivity refers to the proportion of applicants that got a credit and have creditworthiness to get it.

$$Sensitivity = \frac{TP}{TP+FN} x100\% \quad (7)$$

Specificity refers to the proportion of applicants that didn't have a credit and don't have creditworthiness to get it.

$$Specificity = \frac{TN}{TN+FP} x100\% \quad (8)$$

Accuracy is used as an overall measure and it is defined as:

$$Accuracy = \frac{Sensitivity\ +Specificity}{2} x100\% \quad (9)$$

Actually, accuracy is number of correct classified applications/number of total applications. Receiver operating characteristic (ROC) analysis was used to evaluate the discrimination ability of the classifiers. ROC curves denote the performance of a classifier without regard to class distribution or error costs. The classification performance was then measured by the mean area under the ROC curve (AUC). The mean AUC, as an average performance, gives an indication of a typical AUC obtained using the given input data, and indicates how reliably result is estimated (Hanley, McNeil, 1982).

3. Results and findings

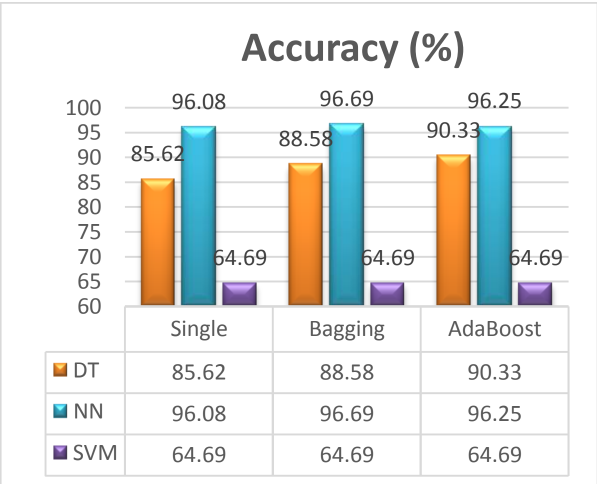
Following sections presents results of application three different classification techniques used alone as single classifiers and in ensemble techniques Bagging and AdaBoost. In Tables 5.-7. we used the abbreviated designation for Single(Si), Bagging(Ba), AdaBoost (AB) and area under ROC (AUC).

Experiment setup 1

Table 5. Experiment setup 1 results

	Accuracy			Specificity			Sensitivity			AUC		
Technique	Si	Ba	AB	Si	Ba	AB	Si	Ba	AB	Si	Ba	AB
Decision tree	85.62	88.58	90.33	80.57	85.51	87.16	80.57	81.45	85.19	0.813	0.934	0.960
Neural network	96.08	96.69	96.25	93.81	95.14	97.15	95.3	95.55	92.12	0.994	0.995	0.284
SVM	64.69	64.69	64.69							0.792	0.795	0.208

Comparing the value of accuracy in Table 5, we can conclude that for this parameter setup, AdaBoost and Bagging have better results than single classifiers. Improvement was greatest for the decision tree, although improvement with neural networks of 0.5 percent is not insignificant taking into account the importance of assessing the large number of cases. Also, AUC is almost 1 for AdaBoost so we can conclude that this result is reliable. Improvement in SVM is not noticeable because Bagging and AdaBoost are not very suitable ensemble technique for SVM. Besides that, SVM accuracy is very poor for selected radial kernel.



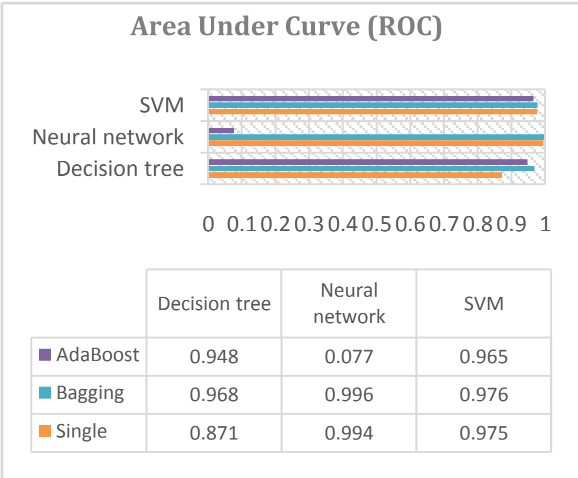


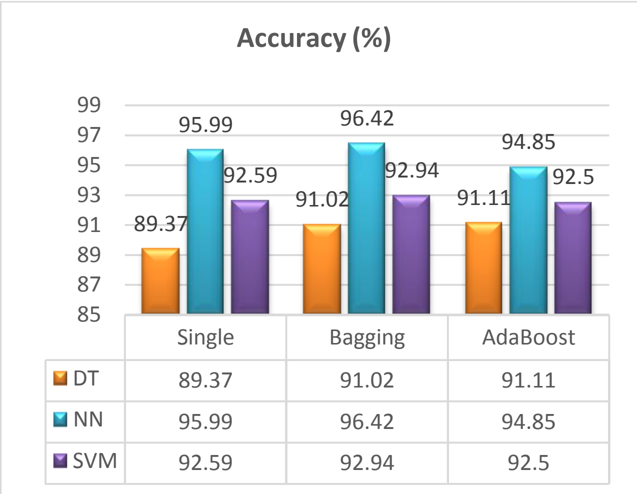
Fig. 1. Accuracy and AUC for experiment setup1

Experiment setup 2

The importance of setting input parameter is reflected in the second experiment. If we compare the results of the neural network, we can see that the accuracy decreased and with single or with AdaBoost and bagging. This reduction is not much, but it shows that the correct tuning parameters can improve accuracy. Results for the decision tree confirm the conclusions of the first experiment and that the accuracy increases when we use ensemble techniques. Also modified SVM kernel radically improves the accuracy and SVM leads in rang of with neural networks and decision tree.

Table 6. Experiment setup 2 results

	Accuracy			Specificity			Sensitivity			AUC		
Technique	Si	Ba	AB	Si	Ba	AB	Si	Ba	AB	Si	Ba	AB
Decision tree	89.37	91.02	91.11	84.84	87.55	89.15	85.39	87.13	85.19	0.871	0.968	0.948
Neural network	95.99	96.42	94.85	95.31	94.99	97.90	93.33	95.05	87.45	0.994	0.996	0.077
SVM	92.59	92.94	92.50	92.06	92.66	92.05	86.63	87.13	86.38	0.975	0.976	0.965



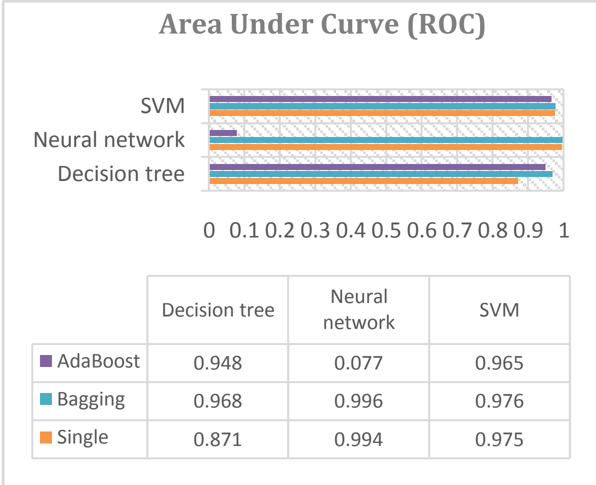


Fig. 2. Accuracy and AUC for experiment setup 2

Experiment setup 3

The third experiment confirms that Bagging and AdaBoost very suitable technique for improving algorithm performance (primarily accuracy). Especially notable improvement with decision tree where improvements are achieved from 86.92 to 91.98 percent. Also, it is noticeable that Bagging always improves neural networks around one percentage. Besides accuracy, specificity and sensitivity as essential factors in the evaluation of assessment in all experiments show improvement when using ensemble technique.

Table 7. Experiment setup 3 results

	Accuracy			Specificity			Sensitivity			AUC		
Technique	Si	Ba	AB	Si	Ba	AB	Si	Ba	AB	Si	Ba	AB
Decision tree	86.92	89.10	91.98	82.76	84.90	89.40	79.99	84.45	87.91	0.851	0.948	0.970
Neural network	96.08	96.51	94.94	93.95	94.95	97.06	95.06	95.31	88.62	0.994	0.995	0.287
SVM	69.92	69.57	70.61	97.22	96.75	96.33	15.31	14.32	18.24	0.670	0.849	0.670

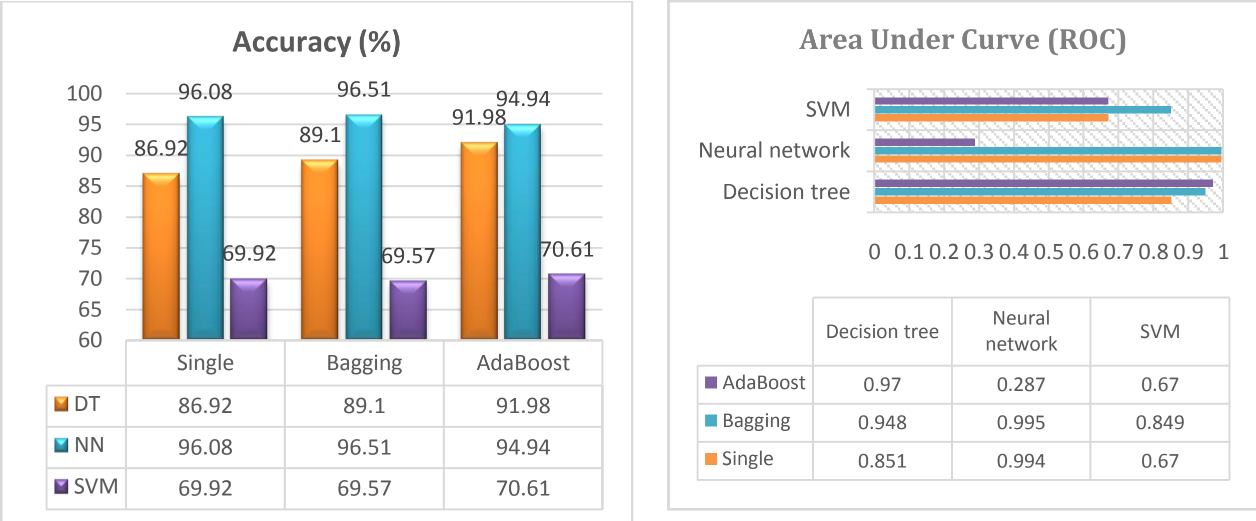


Fig. 3. Accuracy and AUC for experiment setup 3

4. Conclusion

Based on the evaluation criteria: accuracy, sensitivity, specificity and AUC, and three conducted experiments, we can conclude that the use of ensemble techniques leads to improving the assessment of loan applications. Results indicates that both ensemble techniques Bagging and AdaBoost have better classification accuracy then single classifiers. Also, AUC is almost 1 that leads to conclude that accuracy is reliable. Difference between "Single", "AdaBoost" and "Bagging" results is reflected in a few percent, but in credit risk assessment every portion of percentage is important. Even if different parameter setup of base classifiers is used, common is that ensemble techniques have better accuracy and other performance indicators than base classifiers itself. Decision tree have improvement with Bagging and AdaBoost in all experiments. Neural networks accuracy is improved with bagging ensemble in all experiments. Also, neural networks have the best results generally with approximately 96 % accuracy and 0.99 AUC. The advantage of SVM in relation to the other two algorithms is execution time.

References

Baesens et el., 2003 – *Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J.* (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society, 54(6)*, 627–634.

Bekhet, Eletter, 2014 – *Bekhet, H. A., & Eletter, S. F.* (2014). Credit risk assessment model for Jordanian commercial banks: Neural scoring approach. *Review of Development Finance*.

Breiman, 1996 – *Breiman, L.* (1996). Bagging predictors. *Machine Learning, 24(2)*, 123-140.

Breiman et al., 1984 – *Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J.* (1984). *Classification and regression trees. Pacific Grove, CA: Wadsworth.* Pacific Grove, CA: Wadsworth.

Chen, Ma, 2009 – *Chen, W., Ma, C., & Ma, L.* (2009). Mining the customer credit using hybrid support vector machine technique. *Expert Systems with Applications 36*, 7611–7616.

Chuang, Lin, 2009 – *Chuang, C.-L., & Lin, R.-H.* (2009). Constructing a reassigning credit scoring model. *Expert Systems with Applications 36*, 1685-1694.

Cortes, Vapnik, 1995 – *Cortes, C., & Vapnik, V.* (1995). Support vector networks. *Machine Learning, 20(3)*, 273–297.

Dasarathy, 1991 – *Dasarathy, B.* (1991). *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press.

Fisher, 1936 – *Fisher, R.* (1936). The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics 7*, 179-188.

Hanley, McNeil, 1982 – *Hanley, J., & McNeil, B.* (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology 143*, 29–36.

Harris, 2013 – *Harris, T.* (2013). Quantitative credit risk assessment using support vector machines: Broad versus Narrow default definitions. *Expert Systems with Applications 40*, 4404–4413.

Hosmer, Lemeshow, 1989 – *Hosmer, D. W., & Lemeshow, S.* (1989). *Applied Logistic Regression*. New York: John Wiley & Sons, Inc.

Hua et al., 2007 – *Hua, Z., Wang, Y., X, X., Zhang, B., & Liang, L.* (2007). Predicting corporate financial distress based on integration of support vector machine and logistic regression. *Expert Syst. Appl., vol. 33*, 434-440.

Huang et al., 2007 – *Huang, C.-L., Chen, M.-C., & Wang, C.-J.* (2007). Credit scoring with a data mining approach based on supportvector machines. *Expert Systems with Applications 33*, 847–856.

Hung, Chen, 2009 – *Hung, C., & Chen, J.-H.* (2009). A selective ensemble based on expected probabilities for bankruptcy prediction. *Expert Syst. Appl. vol 36*, 5297-5303.

Izenman, 2008 – *Izenman, A. J.* (2008). *Modern multivariate statistical techniques: Regression, classification, and manifold learning*. New York: Springer.

Jensen, 1992 – *Jensen, H. L.* (1992). Using neural networks for credit scoring. *Managerial Finance, 18(1)*, 15-26.

Khashman, 2011 – *Khashman, A.* (2011). Credit risk evaluation using neural networks: Emotional versus conventional models. *Applied Soft Computing 11*, 5477–5484.

Kuncheva, 2003 – *Kuncheva, L. I.* (2003). Combining classifiers: Soft computing solutions. *n S. K. Pal &*, 427–449.

Kvesić, 2013 – *Kvesić, L.* (2013). Application of decision trees in credit scoring. *Ekonomski vjesnik*, 382-390.

Lee et al., 1996 – *Lee, K. C., Han, I., & Kwon, Y.* (1996). Hybrid neural network models for bankruptcy predictions. *Decision Support Syst., vol. 18*, 63-72.

Lee et al., 2006 – *Lee, T.-S. C.-C., Chou, Y.-C., & Lu, C.-J.* (2006). Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. *Computational Statistics & Data Analysis 50*, 1113 – 1130.

Lee et al., 2002 – *Lee, T.-S., Chiu, C.-C., Lu, C.-J., & Chen, I.-F.* (2002). Credit scoring using the hybrid neural discriminant technique. *Expert Syst. Appl., vol. 23*, 245-254.

Li, 2006 – Li, Y. (2006). Predicting materials properties and behavior using classification and regression trees. *Materials Science and Engineering, 433*, 261–268.

Malhotra, Malhotra, 2002 – *Malhotra, R., & Malhotra, D.* (2002). Differentiating between good credits and bad credits using neuro-fuzzy systems. *European Journal of Operation Research, vol. 136*, 190-2011.

Oreski et al., 2012 – *Oreski, S., Oreski, D., & Oreski, G.* (2012). Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. *Expert Systems with Applications 39*, 12605–12617.

Provost, Fawcett, 1997 – *Provost, F., & Fawcett, T.* (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. *Proc. of the third international conference on knowledge discovery and data mining*, pp. 43-48. Newport Beach, CA.

RapidMiner Documentation, 2016 – RapidMiner Documentation. (2016, 5 14). Retrieved from RapidMiner Documentation: http://docs.rapidminer.com/

Tang, Qiu, 2012 – *Tang, B., & Qiu, S.* (2012). A new Credit Scoring Method Based on Improved Fuzzy Support Vector Machine. *Computer Scinecne and Automation Engineering(CSAE), 2012 IEEE International conference*, (pp. 73-75). Zhangjiajie.

Tang, Qiu, 2014 – *Tang, B., & Qiu, S. B.* (2014). An Improved Support Vector Machine for Credit Scoring. *Applied Mechanics and Materials*, 4407-4410.

Twala, 2010 – *Twala, B.* (2010). Multiple classifier application to credit risk assessment. *Expert Systems with Applications 37*, 3326-3336.

Wang et al., 2005 – Wang, Y., Wang, S., & Lai, S. (2005). A New Fuzzy Support Vector Machine to Evaluate Credit Risk. *IEEE Trans. Fuzzy Systems*, 820-831.

West, 2000 – *West, D.* (2000). Neural network credit scoring models. *Computer and Operations Research 27*, 1131-1152.

Wiginton, 1980 – *Wiginton, J. C.* (1980). A note on the comparison of logit and discriminant models of consumer credit behavior. *The Journal of Finance and Quantitative Analysis*, 757-770.

Yap et al., 2011 – *Yap, B. W., Ong, S. H., & Husain, N. H.* (2011). Using data mining to improve assessment of credit worthiness via credit scoring models. *Expert Systems with Applications 38*, 13274–13283.

Yu et al., 2007 – *Yu, L., Lai, K. K., Wang, S., & Zhou, L.* (2007). A Least Squares Fuzzy SVM Approach to Credit Risk Assessment. *Advances in Soft Computing Volume 40*, 865-874.

Yu et al., 2011 – Yu, L., Yao, X., Wang, S., & Lai, K. (2011). Credit risk evaluation using a weighted least squares SVM classifier with designof experiment for parameter selection. *Expert Systems with Applications 38*, 15392–15399.

Zhang et al., 2006 – Zhang, R., McAllister, G., Scotney, B., McClean, S., & Houston, G. (2006). Combining wavelet analysis and bayesian networks for the classification of auditory brainstem response. *IEEE Trans. Info. Tech. Biomed.*, 458–467.