Kybernetes

Customer credit scoring using a hybrid data mining approach

Mohammadali Abedini Farzaneh Ahmadzadeh Rassoul Noorossana

Article information:

To cite this document:
Mohammadali Abedini Farzaneh Ahmadzadeh Rassoul Noorossana , (2016),"Customer credit scoring using a hybrid data mining approach", Kybernetes, Vol. 45 Iss 10 pp. -
Permanent link to this document:
http://dx.doi.org/10.1108/K-09-2015-0228

Users who downloaded this article also downloaded:

For Authors

About Emerald www.emeraldinsight.com

CUSTOMER CREDIT SCORING USING A HYBRID DATA MINING APPROACH

1 Introduction

In the financial literature, in any loan or credit providing process, the customer's credit risk refers to the probability that the applicant (borrower) will default on repaying his/her debts, causing the bank or credit provider organisation to incur a loss. Customer credit scoring is a two-way classification problem; it seeks to classify good and bad applicants into the appropriate class. Applicants with good credit are more likely to meet their financial obligations; applicants with bad credit have possibility greater likelihood of defaulting (Ghodselahi, 2011). In the past, a customer's credit risk was assessed according to the personal intuition and insight of managers, but financial services are becoming increasingly complex, making the task more difficult. Moreover, there is a huge amount of data available for financial organisations. Therefore, data mining and statistical techniques could be used to access these data and support and corroborate financial managers in their credit risk decisions. Not only will credit scoring tools assist banks, credit card companies and other credit provider organisations, but they also will be beneficial for different companies and people that are interested in assessing themselves for any reason or for the companies that advice their customers on their financial affairs. Furthermore, based on the "Financial Inquiry Commission" report on 2011, the global financial crisis during 2008-2009 is considered as a huge crisis that could have been avoidable and caused by some reasons such as failures in financial supervision, failures of corporate governance and risk management at many systemically important financial institutions, and combination of excessive borrowing. There are also some local cases like (Lin, 2009) that indicate the results of lacking powerful credit scoring aids which can alleviate the associated risks.

There has been a consistent growth of different data mining and statistical techniques over recent decades for classification problems (Abdou, 2009), and (Lessmann, Baesens, Seow, & Thomas, 2015). However, in recent publications on credit scoring, there is a tendency to use ensemble and hybrid approaches for classification problems. Because they increase the advantages of the intelligent techniques while simultaneously reducing their disadvantages (de Andres, Lorca, Sanchez-Lasheras, & Javier De Cos-Juez, 2012), and thus more efficient and effective results are achieved. (de Andres et al., 2012) investigated these models and categorised four types of hybrid systems in credit scoring problems; i.e. (1) hybrid algorithms, (2) ensemble classifiers, (3) feature selectors, (4) clustering and classificatory devices. As a study of ensembles (Wang, Hao, Ma, & Jiang, 2011) used three popular ensemble approaches, i.e. bagging, boosting and stacking, on four base classifiers: LR, DT, ANN and SVM. They reported bagging performs better than boosting across all explored credit datasets, while stacking and bagging DT have the best performance in average accuracy, type I error and type II error. For their part, (Wang, Ma, Huang, & Xu, 2012) proposed two dual ensemble models, RS-Bagging DT and Bagging-RS DT, by combining Random Subspace (RS) and bagging approaches and used DT to overcome the weakness of individual decision tress. Inspired by bagging method, (Yu, Wang, & Lai, 2008) proposed a six stage ensemble model. They trained different neural networks on subsets created by the bagging method and then used a de-correlation maximisation algorithm to select the appropriate ensemble members; after a reliability transformation, they fused the classifiers. In a similar study by (Tsai & Hung, 2014), neural network ensembles and hybrid neural networks were compared and it was concluded that hybrid neural networks and neural network ensembles outperform the single neural network. Another notable study is done by (Ala'raj & Abbod, 2016), they proposed a new combination approach based on classifier consensus to combine multiple classifier systems (MCS) of different classification algorithms. Studies such as (Zhou, Lai, & Yu, 2010) and (Marqués Marzal, García Jiménez, & Sánchez Garreta, 2012) falls in this category as well.

Considering hybrid approaches, (Zhang, Gao, & Shi, 2014) proposed a multi-criteria optimisation classifier based on SVM, and reduced model sensitivity to noisy and anomaly data using fuzzification. Some noteworthy studies such as (Akkoç, 2012) and (Ping & Yongheng, 2011) fall into the hybrid algorithm category. There also some notable studies around the use of feature selectors like (Lin, 2009) and (Wu & Hsu, 2012) and clustering algorithms in the classification approach like (Hsieh & Hung, 2010) and (Xiao, Xiao, & Wang, 2016).

This article surveys the effect of the proposed hybrid approach on the performance of credit scoring in accuracy and errors. It also assesses the effect of the ensemble approaches on single classifiers for a selected credit scoring problem. The suggested model incorporates different ensemble methods throughout a simple partitioning method that inspired by bagging method and this enables the approach to be more accurate and generalised in prediction. It prepares a pool of diverse classifiers under four different situations including base classifiers and three ensembles of those base classifiers. Afterward, it introduces a new two-level voting (under two different schemes) to make the final classification. The main goal of the article is to apply the ensemble learning concept to the modified model of traditional simple majority voting and increase the model's generalisability. Finally, it investigates the effect of the proposed voting approach.

The rest of this paper is constructed as follows. The model architecture is presented in section 2. Section 3 demonstrates the results and shows the model's effectiveness compared to other learning methods. Finally, a brief conclusion is provided in section 4.

2 Research Design and Methodology

An important factor in ensemble and hybrid data mining models is the generation of diverse classifiers to make a generalised model. There are various ways to generate different and diverse classifiers. The architecture of the proposed model takes advantage of different approaches to create diversity and could fall into the first and second reviewed categories. The model consists of four stages as depicted in detail in Figure 1. Inspired by the bagging ensemble method, in the first stage, 21 training subsets are generated from the original dataset to support diversity via different training data. In the second stage, five base classifiers are selected throughout the model to support diversity. These five base classifiers are tuned over 21 training subsets. These 5 tuned classifiers are used as base classifiers in an ensemble learning (Meta-learning) algorithm in the third stage. This means 105=21×5 trained ensemble classifiers will be available. The third stage is repeated for three different ensemble algorithms, AdaBoost, Random Subspace and Rotation Forest. In the fourth stage, two-level majority voting with two different schemes is presented; these two schemes are compared with traditional majority voting (fusing all 105 trained classifiers). The results for the last stage are prepared for the three ensemble algorithms. The voting stage is also performed for the trained base classifiers from the second stage to investigate the effect of the third stage.

Figure 1- architecture of the proposed model

2.1 Generating Training Subsets

In the first stage of the proposed model, training subsets are generated. The bagging (Breiman, 1996) approach generates training sets by making make diverse classifiers. At this point, the bagging concept is adopted and the dataset is divided into two disjoint parts (7/10 and 3/10 portions), i.e. training and testing sets, with 700 and 300 instances respectively. Then 21 training subsets are generated from the training set using the following steps:

1. Training sets are divided in 7 disjoint subparts of equal size.
2. Twenty one subsets of size 5 are generated that could be selected from these 7 subparts, i.e. all possible combinations for 5 of 7 donated by $C(5,7)$, that equals to 21.

Figure 2- Diagram of the generation of training subsets

2.2 Tuning Training Parameters of Base Classifiers

The third stage of the proposed considers more diversity by training various classifiers on 21 generated training subsets. From all subsets in the first stage, the five most popular and successful classification tools, i.e. multilayer perceptron (MLP) networks (Hornik, Stinchcombe, & White, 1989), radial basis function (RBF) networks (Park & Sandberg, 1991), support vector machines (SVM) (Cortes & Vapnik, 1995), C4.5 algorithm for decision tree (C4.5 DT) (Quinlan, 1993) and logistic regression (LR) (Allison, 1999), will be trained as base classifiers. To demonstrate this stage more concisely, the training process of base classifiers takes an ensemble approach. In fact, the model is conducted three times through three different ensemble strategies, AdaBoost (Freund & Schapire, 1997), random subspace (RS) (Ho, 1998), and rotation forest (RF) (Rodriguez, Kuncheva, & Alonso, 2006), as Meta classifiers. Then these ensemble approaches are compared.

Before proceeding to ensemble learning in the third stage, it is important to have base classifiers suitably adjusted to get the most possible accuracy from each one. The algorithms of the classifiers consist of parameters that noticeably affect accuracy and precision. For example, the cost parameter in SVM has a great effect on accuracy of prediction (classification). Therefore, for each training subset, five adjusted base classifiers are trained and used in ensemble learning. Table 1 lists the parameters that could be adjusted to better train algorithms for each base classifier.

Table 1- Tuning parameters for each base classifier

2.3 Ensemble Learning

Since there are 21 training subsets, all five base classifiers are trained for each; thus, there will be five trained clarifiers in each dataset. But as shown in Figure 1, this training process uses a Meta classifier algorithm in the third stage.

2.4 Voting (Fusing Trained Classifiers)

Voting (fusing trained classifiers) is the final stage of the proposed model. In the literature the three most popular voting strategies are majority voting, ranking and weighted averaging (Yu et al., 2008). In this stage of the model, a two-level majority voting strategy is proposed. As the name implies, there are two levels, majority voting through two levels, and two schemes to conduct this voting strategy:

- First voting scheme (I): Performing majority voting among trained base classifiers in each training subset in the first level and performing majority voting among the result of the subsets in the second level.
- Second voting scheme (II): Gathering and segregating same classifiers from training subsets into distinct groups, i.e. segregate 21 trained SVMs from training subsets in a group and name it “SVM group”, and similarly obtain “MLP group”, “RBF group”, “LR group” and “DT group”. Then perform majority voting among members of each group in the first and second levels; finally, perform majority voting among the result of the groups.

The first and second strategies are illustrated in Figure 3 and Figure 4 respectively. The fourth stage is performed for three different ensemble algorithms separately and the results compared. The fourth stage is also performed for the base trained classifiers from the second stage in order to judge the effect of the third stage (effect of selected ensemble learning algorithms).

Figure 3- Two-level voting scheme I

Figure 4 - Diagram for voting scheme II

3 Results and Discussion

3.1 Evaluation Criteria for Classification

In a two-class classification case with good and bad classes, a prediction has four possible outcomes.

- True Positive: Classifies an instance as good when it is actually good (TP)
- True Negative: Classifies an instance as bad when it is actually bad (TN)
- False Positive: Classifies an instance as good when it is actually bad (FP)
- False Negative: Classifies an instance as bad when it is actually good (FN)

The following criteria can be calculated to evaluate a classification model.

$$Type\ I\ error = \frac{FN}{TP + FN} \quad (1)$$

$$Type\ II\ error = \frac{FP}{FP + TN} \quad (2)$$

$$Average\ accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (3)$$

Average accuracy is typically considered the main criterion for comparisons. But to make a judgment about error type I and II, these two criteria should not be considered separately. We want to select those combinations that are simultaneously small and balanced. For example (error type I=35%, error type II=40%) is preferred to (error type I=70%, error type II=2%), because the second combination shows a more biased classifier. In fact, the second classifier tends to make predictions in favour of the "bad" class and is highly biased. In this study a simple criterion, whereby the type I and II error criterion is defined as equation (4) to evaluate the performance of different classifiers for error type I and error type II.

$$Types\ I\ \&\ II\ error\ criterion = \frac{(Type\ I\ - Type\ II\ )^2 + (Type\ I\ + Type\ II\ )^2}{2} \quad (4)$$

3.2 Dataset Description

The study uses the German Credit dataset from the "UCI Machine Learning Repository: Data Sets" to evaluate the proposed model. This dataset contains 1000 instances, with 20 features for each, as well as a class feature, .i.e. good/bad applicant. The dataset has 700 instances of good applicants and 300 instances of bad applicants. Every instance has 13 categorical, 4 set/binary and 3 continuous features.

3.3 Experimental Results

WEKA data mining toolkit Version 3.7.11 is applied to implement data mining algorithms.

3.3.1 Stage 1: Generating Training Subsets

The described method of generating training subsets from the German dataset requires a test set of size 300 and 21 training subsets of size 500. These 21 training subsets are used to develop the model; the test set is left out for testing and comparative purposes.

3.3.2 Stage 2: Tuning Training Parameters of Base Classifiers

This stage requires a lot of work, because parameter tuning is done for 21 generated training subsets and five base classifiers. To tune the C-Support Vector classifier, data are normalised and a grid search is conducted for (C, Gamma) for all 21 subsets using 5-fold cross validation, with average accuracy used as the evaluation criterion. For all training subsets, the search areas for C and Gamma are {1 to 20} and $\{10^{-10}$ to $10^{10}\}$ respectively. A grid search is also used to tune the ridge value in the LR algorithm. The search area for the ridge value is $\{10^{-10}$ to $10^{10}\}$ for all training subsets. For the MLP neural net, 0.15, 0.2, 0.25, 0.3 values are examined to determine the validation set percentage and the coordinated accuracy of MLP. The portion with the highest accuracy is selected for each training subset. To tune the RBF neural network, it is desirable to find a suitable value for a number of clusters; therefore, for each training subset, RBF accuracy is compared for 1, 2, 3 and 4 clusters and the best option selected. Finally, in this stage, a grid search is used to find a suitable option for the confidence factor and the minimum number of instances in each leaf to create a tuned DT. {0.05 to 0.5} values are selected as the search area for the confidence factor and {2 to 50} as the search area for the minimum number of instances. All parameter tuning is done via 5-fold or 10-fold cross validation; the results for selected values of all described parameters and coordinated accuracies are presented in Table 2. Based on the results from Table 2, it is obvious that different configurations are proposed for each base classifier in the different training subsets. This happened since the training subsets have different members and these tuned classifiers will contribute to the model generalisability.

Table 2- Selected values for parameters of base classifier algorithms in tuning stage

3.3.3 Stage 3: Ensemble learning

In the second stage, 105=5×21 tuned base classifiers are derived and used in an ensemble approach. For example, for the RS algorithm, 21 RS-SVM models, 21 RS-MLP models, 21 RS-RBF models, 21 RS-logistic regression and 21 RS-decision tree models are trained. As mentioned in the previous section this is performed for AdaBoost and RF algorithms for purposes of comparison. For the three ensemble approaches, ensemble learning is performed using 10-fold cross validation. Each single classifier and ensemble model is re-evaluated on the test dataset.

The effects of ensemble approaches on them are examined in Table 3 for average accuracy and the error type I and II criterion.

Table 3 - Ranking and comparison of base and ensemble classifiers for accuracy and error type I and II criterion

As Table 3 indicates, LR and the RS-DT show the best and worst accuracies among all trained classifiers respectively (75.81% and 70.10%). Furthermore, among all ensemble approaches, the RS approach always decreases the accuracy of base classifiers for the dataset. Comparing base classifiers shows ensemble approaches only make improvements for DTs and MLPNNs on this dataset. Moreover, RS approach obviously can't deal with this unbalanced dataset and tends to make

predictions in favour of the larger class (good applicant). Therefore, it has a high type II error and low type I error compared to other ensemble approaches. According to this criterion (equation (4)), single LR shows the best performance for the error type I and II criterion. Adaboost-LR shows results similar to LR but makes no improvement over single LR.
In order to evaluate the significance of the results from Table 3, one-way ANOVA - Dunnett's method - is used and the effect of each ensemble method on the base classifier as the control group is examined in term of accuracy in Table 4.

Table 4- Effect of ensemble approaches on each base classifier in the stage3

As shown in Table 4, ensemble methods significantly decrease the performance of the base classifiers, except RF for DT where we see a significant positive improvement. Observed improvements for MLP_Adaboost, MLP_RF, LR_Adaboost and DT_Adaboost are not significant.

Comparative information about base classifier performances and the effect of ensemble approaches are in contrast to the expectations, in this stage. Ensemble methods mostly decreased the performance of the base classifiers, except for RF which makes an improvement in the DT for this dataset.

3.3.4 Stage 4: Voting (Fusing Trained Classifiers)

In this stage, the two-level voting strategy is performed in two different schemes, as mentioned in section 3. Traditional majority voting is also conducted among all 105 trained classifiers. This process is repeated four times for four investigated approaches; i.e. single classifier, AdaBoost, RS and RF approaches. Finally, to evaluate the model - given the original training and test subsets - some powerful classifiers, including bagging MLP, bagging LR, RF-LR, SVM and MLP, are trained, and the results are compared in accuracy, error type I and II.

Table 5- Final results of the proposed model for different ensemble approaches

According to the final results of the proposed model shown in Table 5, the two-level voting strategy scheme II leads to the best accuracy. For single classifiers, it shows the best performance in terms of accuracy (78.33%). Traditional majority voting for a single classifier and RF (76.67%) show the next best performances in accuracy. Comparing error type I and type II in Table 5, we see the two-level voting scheme for the AdaBoost approach (12.08%, 49.46% respectively), traditional majority voting for the AdaBoost approach (12.56%, 50.54% respectively) and two-level voting scheme II for a single classifier approach (8.21%, 51.61% respectively) show the best combinations of type I and II error. This result is validated by the error type I and II criterion (25.92%, 27.12%, 27.31%, scheme I, traditional majority voting, scheme II, respectively). Generally speaking, the results show the proposed voting model leads to more accuracy and fewer errors when voting among base classifiers.

Comparing the two schemes for two-level voting strategy, the two-level voting scheme II for single learning and RF ensemble learning are more accurate and show best values in terms of accuracy. Scheme II leads to a better combination of error type I and II for single learning and RF ensemble learning. These results suggest that two-level voting leads to more accurate results in traditional majority voting. Furthermore, the best results of two-level majority voting are obtained when voting among single classifiers – that was predictable from the results of third stage- for the German credit dataset. Performing the proposed two-level voting with the ensemble method makes no improvements in accuracy but the Adaboost ensemble method leads to less error type I and II.

Table 6 – Comparing algorithms with proposed model

Comparisons of the proposed model and other powerful models in Table 6 indicate the suggested model outperforms traditional single classifiers such as SVM or MLP and also popular ensembles methods such as bagging LR, bagging MLPs, and rotation forest LR in terms of accuracy and error type I as well.. Considering error type I and error type II as evaluation criteria, the proposed model displays better performance in comparison to single classifiers such as MLP. However, compared to typical ensemble approaches, the proposed model fails to outperform some in term of error type II, for example, bagging-MLP. Although the proposed approach fails to outperform some of base classifiers and ensembles methods in term of error type II criterion, but the proposed approach leads to more reliable results, especially in terms of accuracy, and error type I. Therefore single approaches could be replaced by the proposed model, especially for the circumstances that more accuracy is needed and there are more costs associated to the error type I.

4 Conclusion

There is a wide range of methods and tools to assist the fiscal managers in credit scoring decisions. Among these tools there is a growing trend in favour of ensemble and hybrid methods in the literature that try to decrease weakness of single classifiers and boost their strength. Therefore, this study proposes a four stage hybrid data mining approach that takes advantage of the bagging ensemble concept in the first stage. It makes a pool of diverse classifiers – through the first and second stages- and takes advantage of the novel two-level majority voting to produce the final classification. The results indicate that the proposed approach outperforms traditional single classifiers and popular ensemble classifiers in term of accuracy. In spite of the fact that the approach fails to outperform some base classifiers in term error type II, it is a reliable approach since it makes a decision based on the two-level voting.

Throughout the third stage some comparison is made among different ensemble methods for the selected credit scoring dataset and effect of the ensemble methods over the single base classifiers were analysed. Although the considered ensemble methods do not improve the performances of the base classifiers in the third stage, but the ensemble concept used by the approach yields more accurate overall results than single and ensemble classifiers throughout the all stages. This is because of fusing diverse and different tuned base classifiers and due to the two-level majority voting as well. Moreover, the second suggested voting scheme - two-level majority voting II- leads to more accurate results and it is recommended for this case. Exploring different types and different numbers of base classifiers is a suggestion for future work. Moreover, the proposed model can be performed using other ensemble approaches like DECORATE.

References

Abdou, H. A. (2009). An evaluation of alternative scoring models in private banking. *Journal of Risk Finance, The, 10*(1), 38-53.

Akkoç, S. (2012). An empirical comparison of conventional techniques, neural networks and the three stage hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) model for credit scoring analysis: The case of Turkish credit card data. *European Journal of Operational Research, 222*(1), 168-178.

Ala'raj, M., & Abbod, M. F. (2016). Classifiers consensus system approach for credit scoring. *Knowledge-Based Systems, 104*, 89-105.

Allison, P. (1999). *Logistic regression using SAS®: theory and application*: SAS Publishing.

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

Commission, F. I. (2011). Final report of the National Commission on the Causes of the Financial and Economic Crisis in the United States. *27th January, Washington DC.*

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

de Andres, J., Lorca, P., Sanchez-Lasheras, F., & Javier De Cos-Juez, F. (2012). Bankruptcy prediction and credit scoring: a review of recent developments based on hybrid systems and some related patents. *Recent Patents on Computer Science, 5*(1), 11-20.

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences, 55*(1), 119-139.

Ghodselahi, A. (2011). A Hybrid Support Vector Machine Ensemble Model for Credit Scoring. *International Journal of Computer Applications, 17*.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 20*(8), 832-844.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks, 2*(5), 359-366.

Hsieh, N.-C., & Hung, L.-P. (2010). A data driven ensemble classifier for credit scoring analysis. *Expert Systems with Applications, 37*(1), 534-545.

Lessmann, S., Baesens, B., Seow, H., & Thomas, L. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: A ten-year update. *European Journal of Operational Research*.

Lin, S. L. (2009). A new two-stage hybrid approach of credit risk in banking industry. *Expert Systems with Applications, 36*(4), 8333-8341.

Marqués Marzal, A. I., García Jiménez, V., & Sánchez Garreta, J. S. (2012). Exploring the behaviour of base classifiers in credit scoring ensembles.

Park, J., & Sandberg, I. W. (1991). Universal approximation using radial-basis-function networks. *Neural computation, 3*(2), 246-257.

Ping, Y., & Yongheng, L. (2011). Neighborhood rough set and SVM based hybrid credit scoring classifier. *Expert Systems with Applications, 38*(9), 11300-11304.

Quinlan, J. R. (1993). *C4. 5: programs for machine learning* (Vol. 1): Morgan kaufmann.

Rodriguez, J. J., Kuncheva, L. I., & Alonso, C. J. (2006). Rotation forest: A new classifier ensemble method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28*(10), 1619-1630.

Tsai, C.-F., & Hung, C. (2014). Modeling credit scoring using neural network ensembles. *Kybernetes, 43*(7), 1114-1123.

Wang, G., Hao, J., Ma, J., & Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring. *Expert Systems with Applications, 38*(1), 223-230.

Wang, G., Ma, J., Huang, L., & Xu, K. (2012). Two credit scoring models based on dual strategy ensemble trees. *Knowledge-Based Systems, 26*, 61-68.

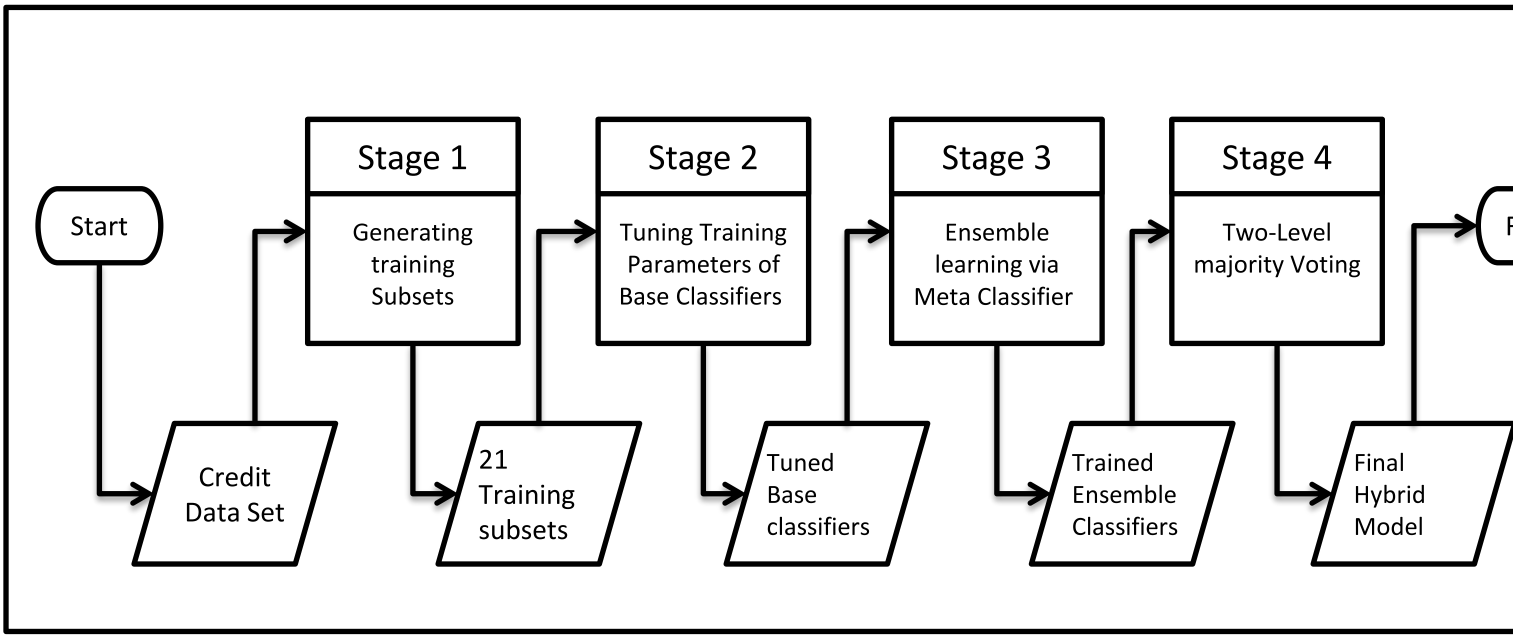
Wu, T.-C., & Hsu, M.-F. (2012). Credit risk assessment and decision making by a fusion approach. *Knowledge-Based Systems, 35*, 102-110.

Xiao, H., Xiao, Z., & Wang, Y. (2016). Ensemble classification based on supervised clustering for credit scoring. *Applied Soft Computing, 43*, 73-86.

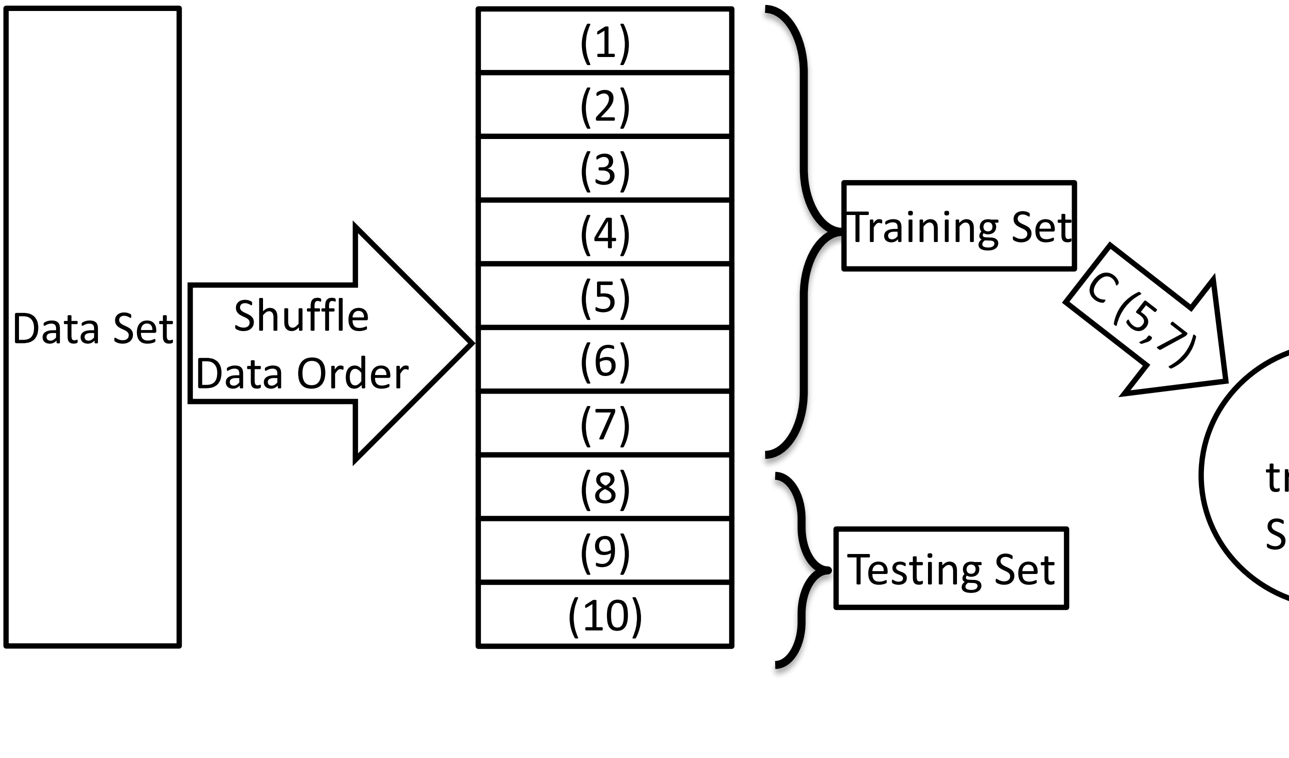
Yu, L., Wang, S., & Lai, K. K. (2008). Credit risk assessment with a multistage neural network ensemble learning approach. *Expert Systems with Applications, 34*(2), 1434-1444.

Zhang, Z., Gao, G., & Shi, Y. (2014). Credit risk evaluation using multi-criteria optimization classifier with kernel, fuzzification and penalty factors. *European Journal of Operational Research, 237*(1), 335-348.

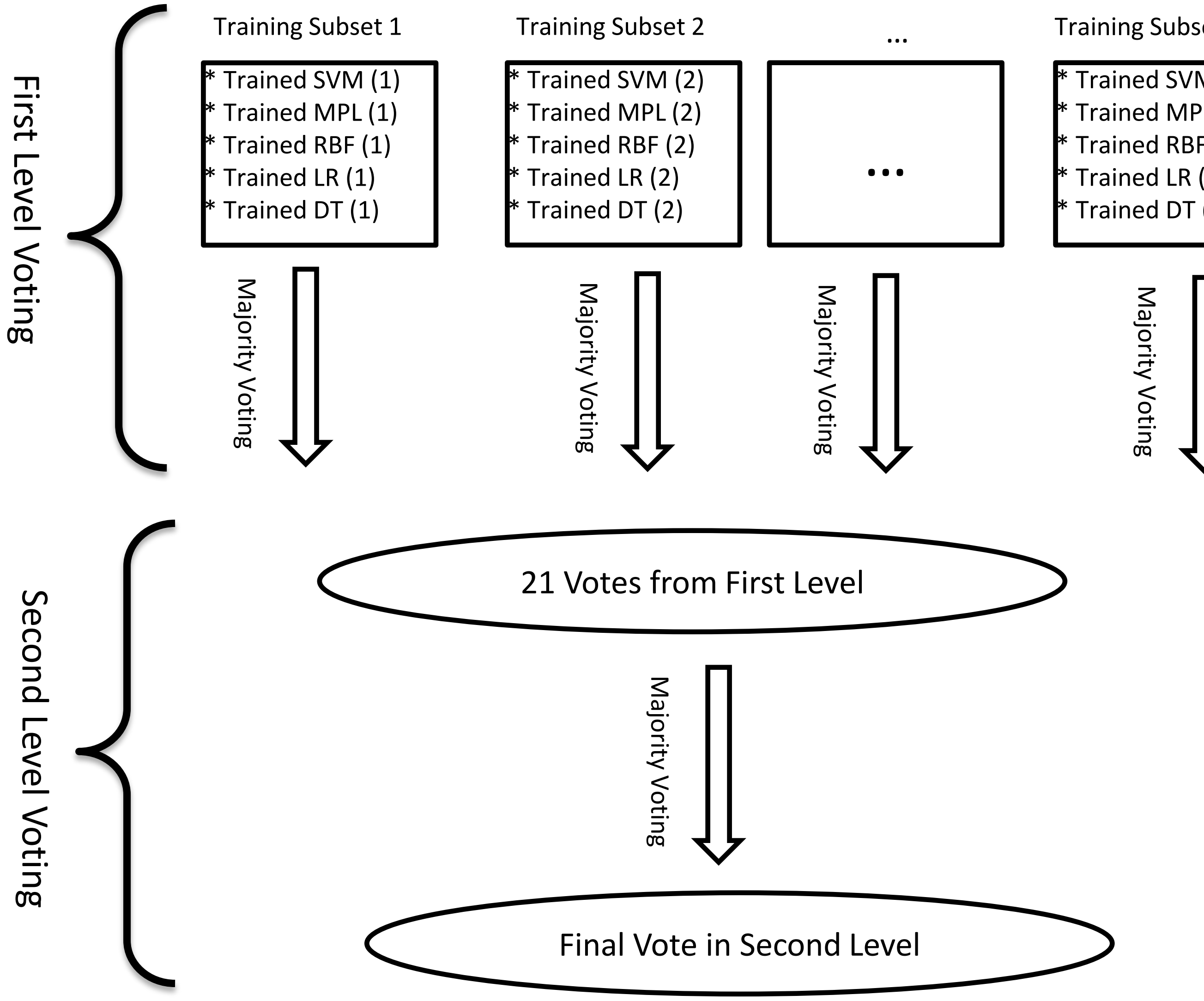
Zhou, L., Lai, K. K., & Yu, L. (2010). Least squares support vector machines ensemble models for credit scoring. *Expert Systems with Applications, 37*(1), 127-133.


Start
Stage 1
Generating training Subsets
Stage 2
Tuning Training Parameters of Base Classifiers
Stage 3
Ensemble learning via Meta Classifier
Stage 4
Two-Level majority Voting
F
Credit Data Set
21 Training subsets
Tuned Base classifiers
Trained Ensemble Classifiers
Final Hybrid Model

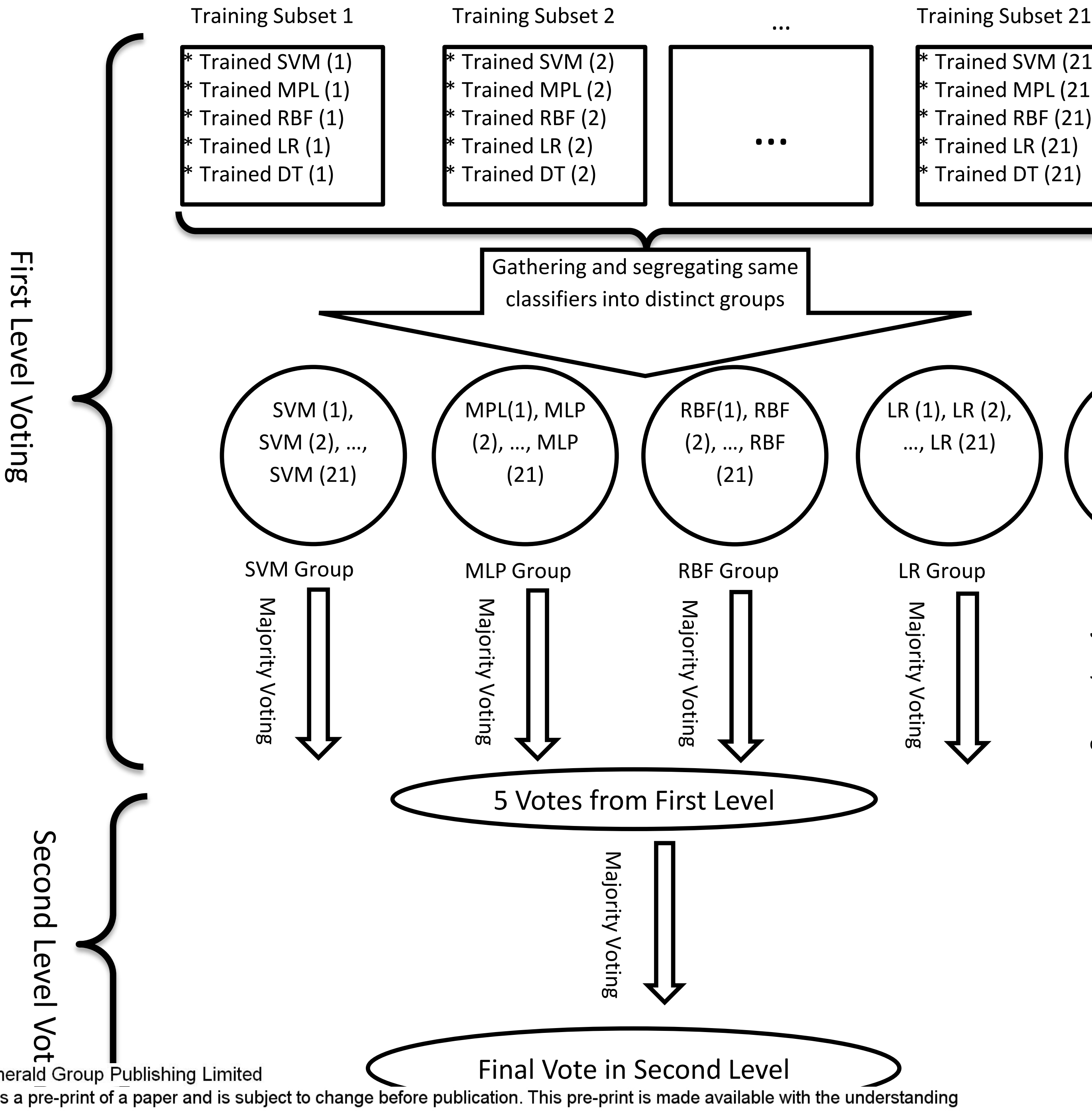
F1


Data Set
Shuffle
Data Order
(1)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
(10)
Training Set
Testing Set
C (5,7)
tr
S

F2


First Level Voting
Training Subset 1
* Trained SVM (1)
* Trained MPL (1)
* Trained RBF (1)
* Trained LR (1)
* Trained DT (1)
Training Subset 2
* Trained SVM (2)
* Trained MPL (2)
* Trained RBF (2)
* Trained LR (2)
* Trained DT (2)
...
...
Training Subs
* Trained SVN
* Trained MP
* Trained RBF
* Trained LR (
* Trained DT (
Majority Voting
Majority Voting
Majority Voting
Majority Voting
Second Level Voting
21 Votes from First Level
Majority Voting
Final Vote in Second Level

F3


Training Subset 1
Training Subset 2
...
Training Subset 21
* Trained SVM (1)
* Trained MPL (1)
* Trained RBF (1)
* Trained LR (1)
* Trained DT (1)
* Trained SVM (2)
* Trained MPL (2)
* Trained RBF (2)
* Trained LR (2)
* Trained DT (2)
...
* Trained SVM (21
* Trained MPL (21
* Trained RBF (21)
* Trained LR (21)
* Trained DT (21)
First Level Voting
Gathering and segregating same classifiers into distinct groups
SVM (1), SVM (2), ..., SVM (21)
MPL(1), MLP (2), ..., MLP (21)
RBF(1), RBF (2), ..., RBF (21)
LR (1), LR (2), ..., LR (21)
SVM Group
MLP Group
RBF Group
LR Group
Majority Voting
Majority Voting
Majority Voting
Majority Voting
5 Votes from First Level
Second Level Vot
Majority Voting
Final Vote in Second Level

Base classifier	**Parameters to be tuned**
C-Support Vector Classification formulation for Support Vector Machines	1. Gamma Parameter in radial base function 2. Regularisation parameter
Multilayer Perceptron Neural Network (MLP)	1. Validation Set Percentage
Radial Basis Function (RBF) Neural Networks	1. Number of clusters through k-means clustering
Logistic Regression (LR)	1. Ridge parameter
C4.5 Decision Tree (DT)	1. Minimum number of instances per leaf 2. Confidence interval

Table 1- Tuning parameters for each base classifier

Training Subset No.	SVM		MLP NN	RBF NN	LR	DT - C4.5	
	Cost (C)	Gamma	Validation Set Percentage	Number of Clusters	Ridge Value	Confidence Factor	Minimum Number of instances
No. 1	3	0.1	20%	2	0.1	0.30	22
accuracy	76.6%		74.2%	74.6%	75.6%	70.0%	
No. 2	13	0.01	35%	1	10	0.50	34
accuracy	75.6%		73.8%	74.2%	75.4%	73.0%	
No. 3	1	0.1	20%	1	10	0.25	13
accuracy	71.0%		74.0%	75.0%	76.8%	73.8%	
No. 4	3	0.1	15%	1	10	0.50	11
accuracy	73.4%		72.8%	75.8%	74.6%	72.6%	
No. 5	9	0.01	15%	1	10	0.40	42
accuracy	73.0%		72.4%	71.2%	73.4%	70.0%	
No. 6	18	0.01	15%	4	10	0.25	14
accuracy	71.2%		70.2%	72.0%	70.6%	71.0%	
No. 7	15	0.01	15%	2	10	0.25	6
accuracy	76.8%		74.4%	73.8%	76.2%	70.4%	
No. 8	13	0.01	20%	4	10	0.35	45
accuracy	74.4%		72.8%	72.8%	73.0%	72.2%	
No. 9	7	0.01	20%	3	0.1	0.10	11
accuracy	73.8%		69.8%	72.0%	73.8%	70.4%	
No. 10	20	0.01	20%	1	10	0.20	34
accuracy	74.0%		71.8%	71.0%	73.6%	70.4%	
No. 11	1	0.1	20%	1	10	0.30	15
accuracy	76.2%		72.6%	76.8%	77.4%	74.2%	
No. 12	18	0.1	25%	1	10	0.15	19
accuracy	75.2%		73.2%	72.6%	76.4%	74.2%	
No. 13	10	0.01	15%	4	10	0.25	33
accuracy	75.4%		74.0%	71.6%	76.8%	71.6%	
No. 14	14	0.01	25%	1	1	0.15	26
accuracy	75.2%		73.2%	73.0%	74.8%	70.0%	
No. 15	2	0.1	25%	1	1	0.25	33
accuracy	71.4%		73.0%	73.8%	73.4%	71.6%	
No. 16	6	0.1	25%	1	1	0.15	26
accuracy	71.4%		71.2%	73.2%	72.0%	70.0%	
No. 17	10	0.1	15%	4	10	0.30	22
accuracy	71.2%		71.6%	70.6%	72.0%	70.4%	
No. 18	11	0.01	15%	1	10	0.15	21
accuracy	71.6%		71.0%	72.4%	69.8%	70.6%	
No. 19	2	0.1	30%	3	10	0.40	25
accuracy	70.8%		71.2%	70.8%	72.0%	69.8%	
No. 20	3	0.1	20%	1	1E-10	0.15	26
accuracy	72.0%		70.0%	68.4%	71.0%	67.4%	
No. 21	20	0.01	25%	1	10	0.30	22
accuracy	73.2%		70.4%	70.2%	73.8%	71.2%	

Table 2- Selected values for parameters of base classifier algorithms in tuning stage

Base Classifier	**SVM**	**MLP NN**	**RBF NN**	**LR**	**DT C4.5**	**SVM**	**MLP NN**	**RBF NN**	**LR**	**DT C4.5**
Criterion	Accuracy					Error type I & II criterion				
Single Classifier	75.08%	72.95%	74.24%	75.81%	71.59%	29.20%	48.97%	32.56%	25.23%	39.89%
RANK	**4**	**14**	**7**	**1**	**17**	**6**	**15**	**8**	**1**	**9**
AdaBoost	73.73%	73.10%	73.76%	75.81%	72.37%	26.27%	39.92%	26.46%	25.23%	31.11%
RANK	**9**	**13**	**8**	**1**	**15**	**4**	**10**	**5**	**1**	**7**
Random Subspace	71.35%	71.08%	73.44%	73.73%	70.10%	72.84%	80.81%	53.08%	52.42%	87.46%
RANK	**18**	**19**	**11**	**9**	**20**	**18**	**19**	**17**	**16**	**20**
Rotation Forest	74.67%	74.40%	72.37%	75.78%	73.40%	43.58%	42.75%	46.42%	26.17%	43.94%
RANK	**5**	**6**	**15**	**3**	**12**	**12**	**11**	**14**	**3**	**13**

Table 3 - Ranking and comparison of base and ensemble classifiers for accuracy and error type I and II criterion

Base Classifier	Difference of Levels	Difference of Means	P-Value	Interpretation
SVM	(SVM_Adaboost)-(SVM_Single)	-0.01349	0.058	*
	(SVM_RS)-(SVM_Single)	-0.03730	0.000	sig. difference
	(SVM_RF)-(SVM_Single)	-0.00413	0.816	*
MLP	(MLP_Adaboost)-(MLP_Single)	0.00143	0.993	*
	(MLP_RS)-(MLP_Single)	-0.01873	0.019	sig. difference
	(MLP_RF)-(MLP_Single)	0.01444	0.090	*
RBF	(RBF_Adaboost)-(RBF_Single)	-0.00476	0.743	*
	(RBF_RS)-(RBF_Single)	-0.00794	0.375	*
	(RBF_RF)-(RBF_Single)	-0.01873	0.005	sig. difference
LR	(LR_Adaboost)-(LR_Single)	0.00000	1	*
	(LR_RS)-(LR_Single)	-0.02079	0.001	sig. difference
	(LR_RF)-(LR_Single)	-0.00032	1	*
DT	(DT_Adaboost)-(DT_Single)	0.00778	0.382	*
	(DT_RS)-(DT_Single)	-0.01492	0.027	sig. difference
	(DT_RF)-(DT_Single)	0.01810	0.006	sig. difference

Table 4 – Effect of ensemble approaches on each base classifier

Evaluati of criteria	**Traditional Majority Voting**				**Two-Level Voting scheme I**				**Two-Level Voting scheme II**			
	Single	**AdaBoost**	**RS**	**RF**	**Single**	**AdaBoost**	**RS**	**RF**	**Single**	**AdaBoost**	**RS**	**RF**
Accuracy	76.67%	75.67%	71.00%	76.67%	76.33%	76.33%	72.00%	76.00%	78.33%	75.67%	71.67%	76.33%
error type I	9.18%	12.56%	2.42%	6.76%	9.66%	12.08%	1.93%	7.73%	8.21%	12.08%	2.42%	7.25%
error type II	54.84%	50.54%	88.17%	60.22%	54.84%	49.46%	86.02%	60.22%	51.61%	51.61%	86.02%	60.22%
error type I & II c.	30.92%	27.12%	77.80%	36.72%	31.01%	25.92%	74.03%	36.86%	27.31%	28.10%	74.05%	36.79%

Table 5- Final results of the proposed model for different ensemble approaches

Algorithm	Accuracy	error type I	error type II	error type I & II c.
Bagging_LR	75.33%	14%	48%	25%
MLP	76.00%	6%	63%	40%
Rotation Forest_LR	75.67%	16%	35%	15%
Bagging_MLP	77.33%	14%	34%	14%
SVM	76.00%	14%	36%	15%
Traditional Majority Voting-Single	76.67%	9%	55%	31%
Traditional Majority Voting-AdaBoost	75.67%	13%	51%	28%
Traditional Majority Voting-RF	76.67%	7%	60%	36%
Two-Level Voting scheme I-AdaBoost	76.33%	12%	49%	25%
Two-Level Voting scheme II-Single	78.33%	8%	52%	28%

Table 6 – Comparing algorithms with proposed model