An Enhanced Comparative Assessment of Ensemble Learning for Credit Scoring

Youssef Tounsi, Larbi Hassouni, and Houda Anoun

***Abstract*—One of the most important aspects of financial risk is credit risk management. Effective credit rating models are crucial for the credit institution in assessing credit applications, they have been widely studied in the field of statistics and machine learning. Given that small improvements in credit rating systems can generate significant profits, any improvement is of high interest to banks and financial institutions. The ensemble methods are a set of algorithms whose individual decisions are combined to perform classification tasks. In this work, we propose an enhanced experimental comparative study of five ensemble methods associated with seven base classifiers using six public credit scoring datasets. Four popular evaluation metrics, including area under the curve (AUC), accuracy, false positive rate (FPR) and Time taken to build the model, are employed to measure the performance of models. The experimental results and statistical tests show that Pegasos model has a better overall performance than the other methods analyzed her for Boosting and Credal Decision Tree (CDT) model has a better overall performance than the other algorithms in the case of Bagging, Random Subspace, DECORATE and Rotation Forest.**

***Index Terms*—Credit scoring, ensemble methods, CART, SVMs, Pegasos.**

I. Introduction

The US subprime crisis that began in 2007 was the worst financial and economic crisis since the great depression of the 1930s, characterized by an increase in subprime mortgage defaults and foreclosures, and the decline in mortgage-backed securities [1]. One of the main causes of the problem comes from the difficulty of classifying the customer as a good or a bad payer depending on his level of risk. In light of this, the Basel Committee issued in 2013 set of principles under the name of BCBS 239, the purpose of which is to enable banks to improve their generation capacity and improve the bank's reporting reliability. BCBS 239 states that banks adhere to a set of basic principles for the effectiveness of aggregation and risk reporting practices (RDARR: Risk Data Aggregation and Risk Reporting) [2]. As a result, credit risk analysis has become more critical than ever. These deficiencies have led to more formal and more precise approaches to credit risk assessment [3]. Moreover, the coming years promise substantial progress in machine learning based on the probabilistic framework. Many studies have shown that the use of these methods in the field of artificial intelligence and data mining shows improvements in the results obtained when compared to those obtained with classical statistical approaches [4].

Manuscript received August 17, 2018; revised September 30, 2018.

Youssef Tounsi is with Lab. RITM/ESTC Hassan II University Casablanca, Morocco (e-mail: tounsi@gmail.com).

Several in-depth studies [3], [4] have been carried out on the use of different overall ensemble methods (AdaBoost, Bagging, Random Subspace, DECORATE and Rotation Forest) with basic classifiers following: 1- nearest neighbor (1-NN), naive Bayesian classifier (NBC), logistic regression (LogR), multilayer perceptron (MLP), radial base function (RBF), support vector machine (SVM), C4.5 decision tree and Credal decision tree (CDT). These studies have yielded some very interesting results, but it can be improved, and that is the purpose of the work we are proposing. In this article, we compare different general procedures studied in previous works by [3], [4]. Here we kept the basic classifiers with the best results, and we added Classification And Regression Tree (CART) and Primal Estimated sub-GrAdient SOlver for SVM (Pegasos) to the set of basic classifiers for many reasons explained below.

In recent years, numerous studies have shown that artificial intelligence techniques, such as the decision tree (DT), artificial neural networks (ANN), and Support Vector Machine (SVM) can be used as alternative methods for credit scoring [5], [6]. On the one hand, CART, C4.5 and CDT and all are classification tree algorithms. CART uses a generalization of the binomial variance called the Gini index. C4.5 uses entropy for its impurity function, whereas the CDT model represents an extension of the classical ID3 and uses imprecise probabilities and uncertainty measures, replacing precise probabilities and entropy with imprecise probabilities and maximum of entropy [2], [7]. CART technique (developed by Breiman, Friedman, Olshen and Stone in 1984), is considered as an innovative, powerful and accurate approach for approximating science and engineering problems [8]. CART is non-parametric in nature and is able to handle data with high skew value, in which the decision tree is constructed by successively splitting the data set into subsets called nodes. A recursive binary partitioning process is applied whereby parent nodes are always divided into two descending nodes (intermediate or terminal), and this process is repeated by considering each intermediate node as a parent node [9]. In this work, we chose CART as a basic learning algorithm because it can deal with both numeric and categorical variables and can easily treat outliers. It is also very easily readable and interpretable into a set of simple rules from datasets [10]. Although CART has many abilities, it has some disadvantages. Such a limitation is the high variance between the samples. This means that the tree structure and the resulting estimates are not necessarily stable in the new samples. Due to their low variance and high predictive accuracy in many areas, the use of CART has been largely improved by resampling methods ("ensemble") that handle the potential instability of CART by averaging the results of many trees [11], [12].

doi: 10.18178/ijmlc.2018.8.5.721

On the other hand, Support vector machine (SVM), is an extremely powerful and widely accepted classifier in the field of risk assessment because of its better generalization ability, it has already surpassed most other classifiers in a large variety of applications [13]. However, conventional SVMs (linear base, polynomial, radial base, exponential radial base, Gaussian radial base, and sigmoid functions) are not suitable for a large scale data set because of its high computational complexity [14], [15]. Nowadays, PEGASOS attracts a lot of attention because it divides the problem into a large scale of sub-problems by stochastic sampling of appropriate size [16], [17]. PEGASOS has been proposed in the study [17] to solve the optimization problem caused by SVM. These authors have proved that the number of iterations needed to obtain a precision solution φ is O (1 / φ), where each iteration operates on a single learning example. They analyzed Pegasos and other SVM training methods and showed that Pegasos is more efficient than other methods in measuring the execution time needed to ensure good predictive performance (test error). So, the use of PEGASOS seemed to be an important choice as a basic classifier in this study.

Through this experimental study, it is shown that the Pegasos model has a better overall performance than the other methods analyzed her for Boosting and CDT model has a better overall performance than the other algorithms in the case of Bagging, Random Subspace, DECORATE and Rotation Forest, for the field of credit scoring using six public datasets in terms of average receiving operator characteristic, accuracy, false positive rate and time used to build the model.

The remainder of this article is organized as follows. We present some related work in Section II. In section III, we present briefly an overview of the classifier ensemble approaches used in this work. In section IV, we describe the set-up of the experiments carried out, section V comments the results obtained from the experiments. We end with a conclusion and discuss possible future working directions in Section VI.

II. Related Work

Ensemble approaches are techniques that create multiple models and then combine them to produce improved results. These methods usually produce more accurate solutions than a single model would [18]-[20]. Ensemble methods in classification tasks have recently been used in many areas (Finance, Computer Security, Marketing, bioinformatics, Environment, Sociology, etc.) to retrieve the useful knowledge from the very large amount of data [21], [22]. In fact, authors of the work [23] suggest a novel ensemble credit model that combines the bagging method with the stacking algorithm. The proposed model performs better in discriminating potential default borrowers. In another field of interest, in [24] authors demonstrate that combining multiple individual classifiers using conventional or custom ensemble learning methods can improve activity recognition accuracy from wrist-worn accelerometer data. Furthermore, in [25], authors have investigated fifteen different machine learning classification algorithms over content based features to classify the spam and non-spam web pages. As a result, ensemble approach is done by using three algorithms which are computed as best on the basis of various parameters, ten-fold Cross-validation approach is also used.

Besides, CART and bagging with CART were evaluated on two UCI credit data sets in the work [10]. In these experiments, ensemble learning yields more favorable results than single CART.

III. Ensemble of Classifiers

The concepts of ensemble learning are usually used to improve overall accuracy, by addressing the weaknesses inherent in each learning algorithm used individually. To that end, ensemble learning involves two stages, creating a set of base models and combining their predictions using some pooling mechanism [18]. Assume we have a library of T base models $M=(M_1, M_2, ..., M_T)$ Then, the ensemble prediction for an example x_i,$P(x_i,M)$, is a composite forecast of the form:

$$P(x_i, M) = \frac{1}{T}\sum_{t=1}^{T} \beta_t \; M_t(x) \quad (1)$$

where $M_t(x)$ denotes the individual prediction of base model M_t and β_t its weight within the ensemble.

A. Bagging

Bootstrapping on predictions has been known since the work of Leo Breiman (1996) under the name of bagging, or Bootstrap aggregating. The Bagging method generates multiple classifiers by manipulating the training set. Each time a different training set is presented to the learning machine. The new training set is created by drawing samples from the original training set randomly with replacement. The final results are obtained by a majority vote for classification [10]. The procedure of Bagging is illustrated in Fig. 1.

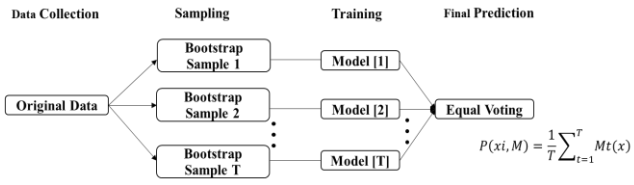

Fig. 1. Bagging approach.

B. Boosting

The boosting algorithm (invented in 1996 by Yoav Freund and Robert E), is based on the observation that finding many rough rules of thumb can be a lot easier than finding a single, highly accurate prediction rule [26]. To apply the boosting approach, we follow these steps:

a) Applying a weak classifier Mi to the learning data set Di, where each observation is assigned an initially equal weight (for i=1);
b) Applying weights to the observations in the learning sample that are inversely proportional to the accuracy of the classification of the computed predicted classifications;
c) Going to the step (i) T-times (T previously fixed);
d) Combining predictions from individual models (weighted by accuracy of the models).

The resampling the original dataset (D1, D2, ..., DT) should provide the most informative learning data for each consecutive classifier: for misclassified samples, the weights are increased, while for correctly classified samples, the weights are decreased. The main idea is to use a series of successive models (M1, M2, ..., MT) where each depends on its predecessors, and this model Mi takes into account the error of the previous model Mi-1 to decide on what to focus on next iteration of data. AdaBoost, Gradient tree boosting, XGBoost, all are boosting algorithms [27], [28]. The AdaBoost approach used in this work is illustrated below:

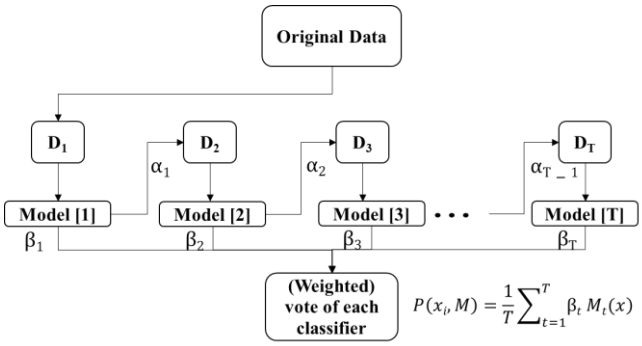

Fig. 2. AdaBoost approach.

C. Random Subspace

Tin Kam Ho uses several classifiers built on randomly selected subspaces of the original input space and combines them into a final decision rule via a simple majority voting procedure in 1998. Each unique classifier uses only a subset of all the features available in the dataset for training and testing. These characteristics are chosen uniformly randomly among the set of characteristics. Thus, Random Subspace sets offer an elegant answer to the problem of the very large dimension [6]. This method is showed below:

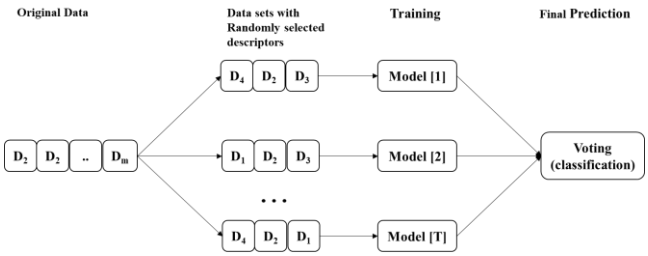

Fig. 3. Random subspace approach.

D. DECORATE

In 2003, Melville proposes a framework named DECORATE (Diverse Ensemble Creation by Oppositional Relabeling of Artificial Training Examples). This technique builds diverse ensembles of classifiers by using specially constructed artificial training examples. It differs from the other ensemble methods described above. Indeed, this meta-algorithm of learning is one of the few approaches proposing an evaluation and an explicit use of the diversity at each iteration of creation of the set of classifiers. The overall architecture of DECORATE is close to that of Adaboost, where the set of classifiers is built incrementally by modifying the learning data set at each iteration. However, DECORATE differs from Adaboost in the changes made to the training data of each classifier. At each iteration, a classifier is trained on the original learning data set, to which are added artificial data, called diversified data. They consist of examples generated from the distribution of the problem but labeled so that the differences with the predictions of the current data set are maximal. The number of artificial examples to be generated at each iteration is a parameter of the algorithm. A new classifier is trained on the original data and these diverse data, with the assumption that this classifier increases the overall diversity of the current set of classifiers. To maintain a reasonable level of precision, only classifiers that increase the predictive capabilities of the current set of classifiers are added. This process is repeated until the set has the desired number of elements or when a maximum number of iterations is reached. The risk is that there are too many rejects, which leads to the construction of a set of smaller size than initially desired. DECORATE reduces the correlation between ensemble algorithms by training classifiers on oppositely labeled artificial samples. Furthermore, the method ensures that the training error of the ensemble is always less than or equal to the error of the base algorithm, this generally leads to a reduction in the generalization error [29], [30].

E. Rotation Forest

More recently, the Rotation Forest algorithm (Developed by Rodriguez, Kuncheva and Alonso in 2006) proposes to increase the diversity between decision trees by applying a feature extraction method (principal component analysis is used in this work) on random subsets of attributes, and to drive the decision trees on the transformed data [31]. To apply the Rotation Forest approach, we follow these steps:

a) The feature set is randomly split into K subsets;
b) PCA is applied to each subset;
c) All principal components are taken;
d) Arrange the PCA coefficients in a matrix (rotation matrix);
e) Apply the rotation matrix to the data features;
f) Build each decision tree on the rotated training data.

IV. Experimentation

A. Data and Variables

TABLE I: Data Set Description

Data set	N	Features	Good	Bad
Australian	690	15	307	383
German	1000	17	700	300
Japanese	653	16	357	296
Iranian	1000	27	950	50
Polish	240	30	128	112
UCSD	2435	38	1836	599

Six sets of real-world credit data have been used to compare the performance of different basic classifiers in several overall systems. A brief description of these datasets can be found in Table I. The widely used Australian, German, and Japanese datasets are from the UCI machine learning database repository (http://archive. cs.uci.edu/ml/). The UCSD dataset is a small version of a database used in the Data Mining 2007 competition organized by the University of California at San Diego and Fair Isaac Corporation. The Iranian dataset comes from a change in a corporate client database of a small private bank in Iran. The

Polish dataset contains information on the bankruptcy of 120 companies registered over a two-year period [4].

B. Research Design

In this section we will describe the experiments carried out and show in the Section V the results obtained. The base classifiers considered in this experimental study are: LogR, MLP, C4.5, CDT, CART, SVM and Pegasos. We mention once more, that only the best base classifiers in the studies [3], [4] are used, together with the CART and Pegasos methods. The ensemble schemes analyzed are the ones described in Section III, i.e. AdaBoost, Bagging, Random Subspace, DECORATE and Rotation Forest. In total, 35 different classifiers have been taken into account for the mentioned six scoring data sets. All experiments were executed in Weka 3.8 software, on a desktop PC with 2.6 GHz (4 CPUs), Intel i7 CPU, 16GB RAM, and Microsoft Windows 10 operating system (64 bits). All used classifiers, are provided by Weka, they were used with their default configurations. We repeated 50 times a 5-fold cross validation procedure for each data set as in the previous works.

C. Measurement of Model Performance

Generally, the evaluation metrics in classification problems are defined from a matrix with the numbers of examples correctly and incorrectly classified for each class, named confusion matrix. The confusion matrix for a binary classification problem (which has only two classes – positive and negative), is shown in the table below:

TABLE II: CONFUSION MATRIX FOR CREDIT SCORING

	Predicted class	
Actual class	Good loans	Bad loans
Good loans	TP	FN (Type II error)
Bad loans	FP (Type I error)	TN

A TP stands for good applicant correctly classified as good, TN stands for bad applicant correctly classified as bad, FN (Type II) stands for good applicant incorrectly classified as bad customer and FP (Type I) stands for Bad customer incorrectly classified as Good customer (high risk).

The standard classifier metrics used for analysis are shown in Table III. The metrics in bold plus time taken to build model (TBM), are the four metrics used to evaluate the ensemble techniques experimented in this work. Indeed, ROC-AUC and accuracy, represent an important reference about the performance when several methods are compared. We have added FPR for the reason that predicting a bad payer as a good payer presents significant risks in case of credit scoring. The total time required to build the model is also a crucial parameter in comparing the classification algorithms especially in the age of massive data [32], [33].

In our work, the Friedman test, which is a non-parametric rank-based test, is used to compare the different models according to Demšar's recommendation [34]. The Friedman test classifies the algorithms separately for each data set, with the best performing algorithm obtaining rank 1, second best rank 2, and so on. This test is based on a Friedman statistic that is distributed according to a chi square with n-1 degree of freedom, where n is the number of algorithms used. This value is based on the individual average rank of each algorithm on each set of data r, j, i, where j = 1, ..., n and i = 1, ..., m, with m the number of data sets. The Friedman test is calculated as follows:

$$\chi_F^2 = \frac{12m}{n(n+1)}\left[\sum_j \left(\sum_i r_i^j\right)^2 - \frac{n(n+1)^2}{4}\right] \quad (2)$$

All the algorithms are equivalent in the case of the null hypothesis. In the opposite case, the null hypothesis of the Friedman test is rejected, we can then compare all the algorithms with each other with a post hoc test in order to find the particular comparisons in pairs that produce significant differences. The Bonferroni-Dunn test is used in this case.

TABLE III: STANDARD BINARY CLASSIFIER METRICS

Metrics	Formula
True Positive Rate (TPR)/ Sensitivity/ Recall	$\frac{TP}{TP+FN}$
True Negative Rat / Specificity (TNR)	$\frac{TN}{FP+TN}$
False Positive Rate (**FPR**)/ Type I error	$\frac{FP}{FP+TN}$
False Negative Rate (FNR) /Type II error	$\frac{FN}{FN+TP}$
Precision/ Positive Prediction Rate (PPR)	$\frac{TP}{TP+FP}$
Negative Prediction Rate (NPR)	$\frac{TN}{FN+TN}$
F-Measure	$\frac{2*precision*recall}{precision+recall}$
Correct Classification % / **Accuracy**	$\frac{(TP+TN)*100}{TP+FP+TN+FN}$
Incorrect Classification %	$\frac{(FP+FN)*100}{TP+FP+TN+FN}$
Area Under Curve (**AUC**)	$\frac{1+TPR-FPR}{2}$
Mathews Correlation Coefficient (MCC)	$\frac{(TP*TN)-(FP*FN)}{\sqrt{(TP+TN)*(TP+FN)*(FP+TN)*(TN+FN)}}$

The performance of two algorithms is clearly distinguished if the corresponding average ranks differ by at least the critical difference. The test statistics for measuring the differences between the i-th and th-classifiers using these approaches are as follows:

$$z = \left(\sum_i r_i^j - \sum_i r_i^t\right) \Big/ \sqrt{\frac{n(n+1)}{m}} \quad (3)$$

The z value is used to define the corresponding probability in the normal distribution table, which is then compared with an appropriate α. In our experiments, on all the tests carried out, the level of significance has been of α= 0.1.

V. Result Discussion

In this paper, we considered Friedman's rank values for each measure: ROC-AUC, accuracy, false-positive rate, and time taken to build the model. Although the objective of this work is not to evaluate the performance of basic classifiers, Table IV reports the mean rank (Friedman score) of each model as a reference for further comparisons. The technique obtaining the lowest average rank ROC (highlighted in bold) corresponds to LogR, whereas the SVM appears as the individual classifier with the worst overall performance. Thus, the lowest average accuracy rank corresponds to CDT, where the MLP appears as the individual classifier with the lowest overall accuracy rate.

Table V reports the results of the ROC curves and the average rank of the different sets. For each set method, the basic classifier with the best average ranking among the six sets of credit score data is highlighted in bold and the overall best for each set of data is also noted with the italic and bold fonts. As can be seen, the C4.5 seems to be the best basic classifier with DECORATE and Rotation Forest, while Pegasos has the highest ranking when used with AdaBoost. CART ranks lowest with Bagging, while CDT ranks lowest with the Random Subspace algorithm. It should be noted that LogR, which is the individual model with the highest (lowest average ranking), performs less well than most classifiers when used in an overall approach.

TABLE IV: Result of **ROC**, **ACCURACY**, **TBM** and **FPR** for Each Base Classifier and Data Set

ROC	Australian	German	Japanese	Iranian	Polish	UCSD	Rank average
LogR	**0,9020**	**0,7880**	**0,9310**	**0,7120**	**0,8150**	**0,8820**	**1,0**
MLP	0,8830	0,7350	0,8940	0,6990	0,8000	0,8350	2,5
C4.5	0,8790	0,6850	0,8470	0,5680	0,7110	0,7650	5,2
CDT	0,8930	0,7120	0,9000	0,5000	0,7810	0,8580	3,0
CART	0,8680	0,7160	0,8820	0,5000	0,7660	0,8250	4,2
SVM	0,8680	0,6860	0,8700	0,5000	0,7100	0,7420	5,8
Pegasos	0,8530	0,6710	0,8720	0,5020	0,7230	0,7620	5,7

Accuracy	Australian	German	Japanese	Iranian	Polish	UCSD	Rank average
LogR	84,93	75,70	**87,29**	94,00	**74,58**	**84,02**	3,0
MLP	82,61	72,00	82,08	93,30	74,17	80,62	6,5
C4.5	**85,51**	73,30	84,69	94,10	68,75	82,14	4,8
CDT	85,22	72,10	86,06	**95,00**	75,83	83,66	**2,8**
CART	85,07	73,70	86,06	**95,00**	75,00	82,26	3,2
SVM	85,07	76,00	86,37	**95,00**	70,83	83,20	3,2
Pegasos	84,93	**76,20**	86,52	93,60	71,67	83,70	3,5

Time	Australian	German	Japanese	Iranian	Polish	UCSD	Rank average
LogR	0,05	0,23	0,04	0,07	**0,02**	0,18	2,8
MLP	7,81	18,68	1,35	3,75	1,29	21,39	7,0
C4.5	0,05	0,10	**0,01**	0,04	0,02	0,12	1,7
CDT	**0,03**	**0,08**	**0,01**	**0,01**	**0,02**	**0,09**	**1,0**
CART	0,67	1,11	0,07	0,07	0,05	0,40	5,3
SVM	0,20	0,61	0,15	0,28	**0,02**	0,18	4,2
Pegasos	0,14	0,18	0,03	0,50	**0,02**	0,20	3,7

FPR	ROC	German	Japanese	Iranian	Polish	UCSD	Rank average
LogR	0,1400	0,1400	0,1110	0,0120	0,2420	0,0940	4,0
MLP	0,1990	0,1970	0,1930	0,0280	**0,1560**	0,1300	6,0
C4.5	0,1630	0,1490	0,1550	0,0150	0,3280	0,1140	5,3
CDT	0,1430	0,1290	0,0980	0,0000	0,2420	0,0930	2,8
CART	**0,1070**	0,1360	0,0950	0,0000	0,2970	0,1060	3,0
SVM	**0,1070**	0,1290	**0,0610**	0,0000	0,3200	**0,0810**	**1,8**
Pegasos	0,1140	**0,1010**	**0,0610**	0,0160	0,3670	0,0900	3,3

TABLE V: Average Result of the **ROC Curves** for Each Base Classifier and Data Set Grouped by Ensemble

Ensemble	Base	Australian	German	Japanese	Iranian	Polish	UCSD	Average rank
Adaboost	**LogR**	0,8990	0,6980	0,9090	0,6370	0,8180	0,8530	6,2
	MLP	0,9070	0,7020	0,8900	0,6550	0,8390	0,8780	5,2
	C4.5	0,9170	0,7320	0,9060	0,6800	0,8340	**0,9040**	4,2
	CDT	0,9200	**0,7700**	0,9130	0,6710	0,7930	0,9020	3,7
	CART	0,9230	0,7430	0,9140	0,6810	**0,8410**	0,8990	2,7
	SVM	0,9200	0,7690	0,9160	**0,7600**	0,7940	0,8510	3,5
	Pegasos	**0,9250**	0,7640	**0,9180**	0,7150	0,8380	0,8670	**2,5**
Bagging	**LogR**	0,9290	0,7900	**0,9310**	0,7220	0,8250	0,8840	3,3
	MLP	0,9220	0,7820	0,9160	**0,7950**	**0,8590**	0,8770	3,8
	C4.5	***0,9300***	0,7680	0,9280	0,7880	0,8490	**0,9110**	2,5
	CDT	0,9230	**0,8010**	0,9190	0,7220	0,8290	0,9100	3,3
	CART	***0,9300***	0,7890	0,9280	0,7410	0,8360	0,9100	**2,3**
	SVM	0,8970	0,7800	0,9030	0,4990	0,8150	0,8060	6,7
	Pegasos	0,9230	0,7800	0,9170	0,5600	0,8260	0,8350	5,2
Random Subspace	**LogR**	**0,9280**	0,7970	***0,9310***	0,7460	0,8300	0,8820	**2,5**
	MLP	0,9240	0,7860	0,9230	**0,7740**	0,8390	0,8940	3,0
	C4.5	0,9270	0,7750	0,9210	0,6860	0,8220	**0,9120**	3,7
	CDT	0,9250	***0,8060***	0,9220	0,7170	0,8390	0,9040	**2,5**
	CART	0,9250	0,7670	0,9280	0,6420	**0,8470**	0,8970	3,2
	SVM	0,9190	0,7490	0,9210	0,5000	0,7880	0,8050	6,5
	Pegasos	0,9150	0,7630	0,9220	0,5180	0,7590	0,8400	6,0
Decorate	**LogR**	**0,9180**	**0,7910**	**0,9270**	0,6750	0,8050	0,8640	2,3
	MLP	0,9110	0,7080	0,8850	0,6740	0,8470	0,8430	4,5
	C4.5	0,9140	0,7670	0,9240	**0,7310**	**0,8550**	0,8870	**1,8**
	CDT	0,9130	0,7750	0,8970	0,6890	0,8350	**0,8900**	2,7
	CART	0,9130	0,7540	0,9180	0,5780	0,8510	0,8450	3,8
	SVM	0,8620	0,6950	0,8700	0,6400	0,7190	0,8360	5,7
	Pegasos	0,8890	0,7050	0,8800	0,5850	0,7730	0,8080	6,2
Rotation Forest	**LogR**	**0,9280**	**0,7910**	***0,9310***	0,7120	0,8180	0,8830	2,8
	MLP	0,9200	0,7650	0,9220	**0,7630**	0,8340	0,8910	3,5
	C4.5	0,9240	0,7730	0,9260	0,7490	**0,8570**	***0,9220***	**1,8**
	CDT	0,9150	0,7880	0,9260	0,7320	0,8420	0,9070	2,7
	CART	0,9080	0,7580	0,9250	0,5430	0,8460	0,9050	4,2
	SVM	0,8990	0,7030	0,9120	0,5000	0,7980	0,8050	6,3
	Pegasos	0,8680	0,6650	0,8970	0,5530	0,7230	0,8200	6,5

TABLE VI: Average Result of **Accuracy** for Each Base Classifier and Data Set Grouped by Ensemble

Ensemble	Base	Australian	German	Japanese	Iranian	Polish	UCSD	Average rank
Adaboost	**LogR**	86,81	75,70	**87,29**	94,00	74,58	83,86	3,7
	MLP	83,48	71,20	81,78	93,20	76,25	83,94	5,8
	C4.5	85,94	74,30	84,99	94,10	**77,50**	**87,72**	3,5
	CDT	86,38	74,00	86,37	94,30	71,67	86,20	4,0
	CART	**87,39**	73,10	86,52	94,70	77,08	86,61	**2,5**
	SVM	84,49	76,00	85,60	**95,00**	70,42	83,04	4,5
	Pegasos	86,23	**76,50**	83,31	94,60	75,42	83,16	4,0
Bagging	**LogR**	86,67	76,50	87,29	94,40	73,75	83,94	4,3
	MLP	86,09	76,00	84,69	94,90	***79,58***	83,86	4,8
	C4.5	87,54	75,10	87,14	**95,20**	76,67	**86,98**	2,8
	CDT	86,96	75,30	***88,36***	95,00	73,33	85,95	3,5
	CART	***87,83***	***77,10***	87,60	95,10	77,92	86,32	***1,7***
	SVM	85,51	76,70	86,37	95,00	72,92	83,08	5,2
	Pegasos	85,94	76,70	86,68	94,90	72,50	83,45	5,2
Random Subspace	**LogR**	85,51	74,90	86,22	94,80	74,17	83,82	4,5
	MLP	86,09	**76,10**	86,22	**95,00**	72,50	84,76	3,2
	C4.5	86,23	73,90	86,98	94,90	**77,08**	**86,16**	2,5
	CDT	**86,96**	74,20	**86,98**	**95,00**	74,58	85,46	**1,8**
	CART	86,23	72,80	86,83	**95,00**	76,25	84,97	3,0
	SVM	85,51	72,40	85,45	**95,00**	72,08	76,35	5,5
	Pegasos	85,22	73,10	85,60	94,70	71,25	80,90	6,3
Decorate	**LogR**	86,67	76,20	**86,52**	94,30	73,75	84,11	3,5
	MLP	86,09	70,80	82,24	94,40	75,83	82,14	4,7
	C4.5	85,65	74,70	85,45	**95,20**	**78,75**	**85,87**	**2,7**
	CDT	85,80	74,60	86,52	94,90	77,50	**85,71**	3,5
	CART	**87,10**	74,60	86,22	95,10	77,50	83,98	3,2
	SVM	85,51	75,40	86,37	94,80	67,50	83,04	5,5
	Pegasos	85,94	**76,20**	86,37	93,70	70,83	83,45	5,2
Rotation Forest	**LogR**	**86,81**	**76,10**	**87,14**	94,00	73,75	84,11	3,3
	MLP	85,36	74,80	84,99	94,70	74,17	84,68	5,2
	C4.5	**86,81**	74,50	85,76	94,70	**79,17**	***87,19***	3,2
	CDT	**86,81**	73,40	86,37	**95,00**	74,58	85,54	**2,8**
	CART	85,94	74,40	85,91	**95,00**	77,08	85,42	3,5
	SVM	85,51	75,40	86,37	**95,00**	72,92	83,04	4,2
	Pegasos	85,65	75,00	86,52	94,90	71,67	82,30	4,7

The Friedman's test finds significant differences for the AUC.

The Friedman's test finds also significant differences for the accuracy. His results are described as in Table VI, the CART appears to be the best base classifier with Adaboost and Bagging, while the CDT presents the best rank when used with the Random Subspace and Rotation Forest, C4.5 as the lowest rank with the DECORATE algorithm.

TABLE VII: RESULT OF THE **FPR** FOR EACH BASE CLASSIFIER AND DATA SET GROUPED BY ENSEMBLE

Ensemble	Base	Australian	German	Japanese	Iranian	Polish	UCSD	Average Rank
Adaboost	LogR	0,1240	0,1400	**0,1110**	0,0120	0,2420	0,0970	3,3
	MLP	0,1760	0,2170	0,1790	0,0290	0,1950	0,0830	5,5
	C4.5	0,1500	0,1400	0,1590	0,0190	**0,1560**	**0,0650**	**3,2**
	CDT	0,1530	0,1460	0,1260	0,0130	0,3040	0,0750	4,0
	CART	0,1530	0,1460	0,1450	0,0150	0,1950	0,0700	3,7
	SVM	0,1600	**0,1290**	0,1420	***0,0000***	0,3280	0,0870	3,8
	Pegasos	**0,1140**	0,1410	0,2090	0,0050	0,2270	0,0960	4,0
Bagging	LogR	0,1170	0,1310	0,1080	0,0080	0,3040	0,0950	5,8
	MLP	0,1500	0,1210	0,1590	0,0070	**0,1960**	0,0930	5,0
	C4.5	0,1240	**0,1170**	0,1280	0,0020	0,2500	**0,0710**	3,7
	CDT	0,1110	0,1130	0,1460	0,0000	0,2580	0,0770	3,2
	CART	0,1140	0,1090	0,1150	**0,0010**	0,2410	0,0720	**2,5**
	SVM	***0,0750***	0,1260	***0,0610***	0,0507	0,2680	0,0800	4,0
	Pegasos	0,0850	0,1210	0,0680	0,0010	0,2950	0,0880	3,5
Random Subspace	LogR	0,1890	0,0600	0,1760	0,0020	0,2950	0,0740	5,3
	MLP	0,1860	0,0570	0,1890	***0,0000***	0,2950	0,0740	4,5
	C4.5	0,1890	0,0390	0,1620	0,0020	0,2410	0,0840	4,8
	CDT	0,1790	0,0230	0,1090	0,0000	0,2580	0,0780	2,8
	CART	0,1730	***0,0210***	0,1590	***0,0000***	0,2410	0,0820	2,8
	SVM	**0,1240**	0,0240	0,1320	**0,0000**	0,2950	***0,0070***	**2,3**
	Pegasos	0,1470	0,0410	**0,1280**	0,0030	**0,1790**	0,0470	3,2
Decorate	LogR	0,1270	0,1530	0,1350	0,0070	0,2860	0,1000	5,0
	MLP	0,1530	0,1800	0,1930	0,0130	0,3040	0,1210	6,7
	C4.5	0,1470	0,1300	0,1450	0,0050	0,2140	0,0910	4,2
	CDT	0,1300	0,1090	0,1510	0,0010	**0,2110**	**0,0830**	***2,2***
	CART	0,1240	**0,0970**	0,1250	**0,0010**	0,2320	0,0940	***2,2***
	SVM	***0,0750***	0,1310	***0,0610***	0,0020	0,3390	0,0860	3,2
	Pegasos	0,1110	0,1160	***0,0610***	0,0150	0,2410	0,0940	3,5
Rotation Forest	LogR	0,1240	0,1340	0,1150	0,0120	0,2680	0,0940	6,0
	MLP	0,1560	0,1630	0,1590	0,0030	0,3130	0,0820	6,2
	C4.5	0,1210	0,1290	0,1110	0,0070	0,2410	0,0780	4,3
	CDT	0,1010	0,0930	0,0810	***0,0000***	0,2500	0,0770	**2,3**
	CART	0,1140	**0,0740**	0,0810	***0,0000***	0,2500	0,0800	2,7
	SVM	***0,0750***	0,1130	***0,0610***	***0,0000***	0,3130	**0,0690**	**2,3**
	Pegasos	0,1010	0,0910	***0,0610***	0,0010	**0,1880**	0,1110	2,8

TABLE VIII: RESULT OF THE TBM (MEASURED BY SECOND) FOR EACH BASE CLASSIFIER AND DATA SET GROUPED BY ENSEMBLE

Ensemble	Base	Australian	German	Japanese	Iranian	Polish	UCSD	Average rank
Adaboost	LogR	***0,14***	**0,31**	**0,21**	**0,47**	0,27	***1,47***	***1,2***
	MLP	10,06	78,56	7,02	16,78	7,44	857,57	7,0
	C4.5	0,75	1,36	0,73	1,02	0,66	7,27	4,0
	CDT	0.77	1,22	0,61	0,54	0,27	8,21	3,3
	CART	2,00	13,03	1,59	1,69	1,80	15,84	6,0
	SVM	0,23	5,72	0,45	0,66	***0,13***	7,50	3,0
	Pegasos	0,66	1,80	0,39	1,14	0,31	6,11	3,3
Bagging	LogR	1	3,72	0,84	3,49	0,94	8,18	4,0
	MLP	47,41	735,13	44,05	194,74	56,60	702,3	7,0
	C4.5	0,41	0,55	0,3	0,97	0,56	5,16	2,3
	CDT	**0,22**	**0,39**	**0,21**	**0,67**	0,27	**3,06**	**1,3**
	CART	1,37	10,00	1,42	3,03	1,49	18,03	5,0
	SVM	2,4	14,28	2,84	4,49	**0,20**	8,95	4,3
	Pegasos	1,61	5,10	1,29	2,17	0,78	10,56	4,0
Random Subspace	LogR	0,63	1,74	0,6	1,22	0,51	5,11	3,7
	MLP	25,17	227,20	26,22	62,83	17,95	284,2	7,0
	C4.5	0,28	*0,45*	0,24	0,53	0,22	3,1	2,2
	CDT	***0,14***	***0,25***	***0,17***	***0,36***	0,19	**1,87**	***1,2***
	CART	1,19	5,51	1,21	1,91	0,71	9,49	5,3
	SVM	1,45	5,59	1,84	1,96	**0,15**	3,74	4,7
	Pegasos	1,06	3,15	0,69	1,26	0,40	4,45	4,0
Decorate	LogR	2,28	6,94	2,15	3,51	1,16	16,35	3,7
	MLP	105,21	1466,52	111,68	300,99	83,91	1421,24	7,0
	C4.5	1	1,64	0,71	2,92	0,87	20,73	2,8
	CDT	**0,95**	**1,45**	**0,64**	1,7	0,79	10,85	**1,3**
	CART	3,81	64,00	7,9	3,78	3,45	67,79	5,7
	SVM	1,2	85,69	1,41	**1,11**	**0,39**	**14,17**	2,7
	Pegasos	3,35	30,16	2,19	5,28	1,36	16,68	4,7
Rotation Forest	LogR	0,25	1,69	1,51	0,86	1,34	10,13	3,7
	MLP	10,93	297,84	58,21	30,11	42,87	796,03	7,0
	C4.5	**0,19**	1,88	0,91	**0,38**	0,63	13,41	**2,7**
	CDT	1,08	**1,48**	**0,88**	**1,83**	0,56	8,88	3,0
	CART	0,59	4,86	2,68	0,84	**0,37**	30,46	4,5
	SVM	0,44	1,91	2,37	0,42	0,38	**5,41**	2,8
	Pegasos	1,69	2,66	1,46	0,52	1,13	11,69	4,3

Taking into account Table VII about the false positive rate, the Friedman's test finds significant differences. The technique achieving the lowest FRP average rank (highlighted in bold face) corresponds to:

- SVM for Random Subspace and Rotation Forest.
- CART for Bagging and DECORATE.
- CDT for DECRATE and Rotation Forest.
- C4.5 for Adaboost.

For the time taken to build model (measured by second), presented in Table VIII, the CDT has the lowest rank for most ensemble methods, closely followed by the SVM, C4.5, Pegasos and LogR and SVM. While, MLP correspond to the worst in terms of time taken to constitute the model.

Taking into account the Table IX about the summary of results, the technique achieving the lowest average rank (highlighted in bold face) corresponds to:

- CDT for Bagging, Random subspace, DECORATE and Rotation Forest
- Pegasos for Adaboost scheme.
- C4.5 for Bagging scheme.

TABLE IX: RESULT OF (ROC, ACCURACY, COST AND FRP) RANK'S FOR EACH BASE CLASSIFIER AND DATA SET GROUPED BY ENSEMBLE

Ensemble	Base	ROC	Accuracy	Time	FPR	Rank average
Adaboost	LogR	6,2	3,7	***1,2***	3,3	3,6
	MLP	5,2	5,8	7,0	5,5	5,9
	C4.5	4,2	3,5	4,0	**3,2**	3,7
	CDT	3,7	4,0	3,3	4,0	3,8
	CART	2,7	**2,5**	6,0	3,7	3,7
	SVM	3,5	4,5	3,0	3,8	3,7
	Pegasos	**2,5**	4,0	3,3	4,0	**3,5**
Bagging	LogR	3,3	4,3	4,0	5,8	4,4
	MLP	3,8	4,8	7,0	5,0	5,2
	C4.5	2,5	2,8	2,3	3,7	**2,8**
	CDT	3,3	3,5	**1,3**	3,2	**2,8**
	CART	**2,3**	***1,7***	5,0	2,5	2,9
	SVM	6,7	5,2	4,3	4,0	5,0
	Pegasos	5,2	5,2	4,0	3,5	4,5
Random Subspace	LogR	**2,5**	4,5	3,7	5,3	4,0
	MLP	3,0	3,2	7,0	4,5	4,4
	C4.5	3,7	2,5	2,2	4,8	3,3
	CDT	**2,5**	**1,8**	***1,2***	2,8	***2,1***
	CART	3,2	3,0	5,3	2,8	3,6
	SVM	6,5	5,5	4,7	**2,3**	4,8
	Pegasos	6,0	6,3	4,0	3,2	4,9
Decorate	LogR	2,3	3,5	3,7	5,0	3,6
	MLP	4,5	4,7	7,0	6,7	5,7
	C4.5	***1,8***	**2,7**	2,8	4,2	2,9
	CDT	2,7	3,5	**1,3**	***2,2***	**2,4**
	CART	3,8	3,2	5,7	***2,2***	3,7
	SVM	5,7	5,5	2,7	3,2	4,3
	Pegasos	6,2	5,2	4,7	3,5	4,9
Rotation Forest	LogR	2,8	3,3	3,7	6,0	4,0
	MLP	3,5	5,2	7,0	6,2	5,5
	C4.5	***1,8***	3,2	**2,7**	4,3	3,0
	CDT	2,7	**2,8**	3,0	**2,3**	**2,7**
	CART	4,2	3,5	4,5	2,7	3,7
	SVM	6,3	4,2	2,8	**2,3**	3,9
	Pegasos	6,5	4,7	4,3	2,8	4,6

Fig. 4-Fig. 8 illustrate the average ranking of each basic algorithm by meta-algorithm.

The summary of the findings are described as in Table X and Fig. 9, we compare method by method in each ensemble scheme:

LogR: This method obtains second place in the Adaboost scheme and achieve an average performance in the other ensemble methods.

MLP: It is the worse method in this experimental study. It is especially a very bad method when Adaboost, Decorate or Rotation Forest ensembles are applied. It can be concluded also that this method obtains the highest rank regarding the results of time to build the model.

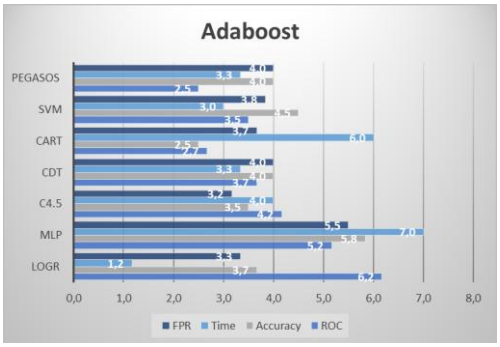

Fig. 4. Average rank Adaboost.

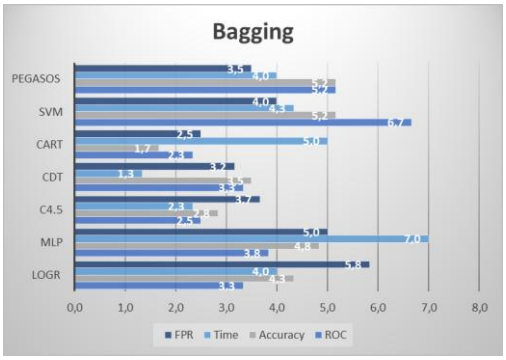

Fig. 5. Average rank bagging.

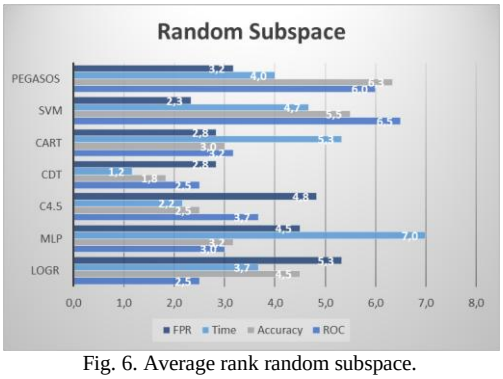

Fig. 6. Average rank random subspace.

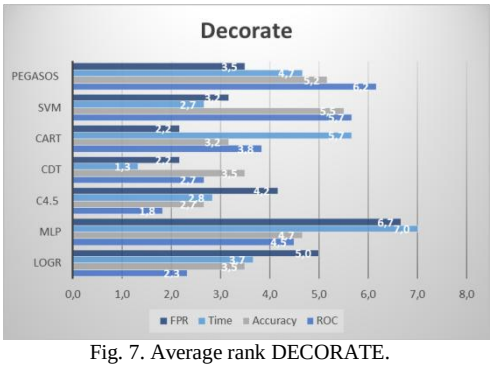

Fig. 7. Average rank DECORATE.

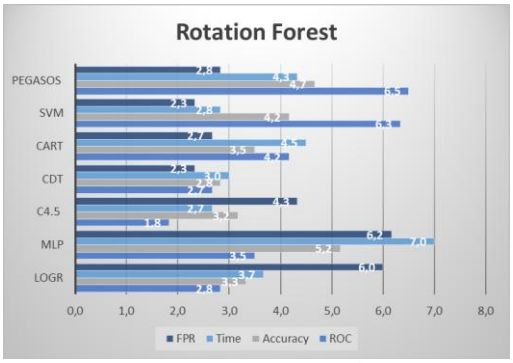

Fig. 8. Average rank rotation forest.

CDT: It is the winner method in 4 of the 5 ensemble schemes. The greatest differences in favor of the CDT with respect the rest are obtained when the Random Subspace ensemble is applied. When Bagging ensemble is used, this method is the winner but with similar result than the C.5 method.

C4.5: It is the winner method only in the Bagging ensemble, and the second one in the Adaboost, Random Subspace, Decorate and Rotation Forest. If we look at the results of the base classifiers Table IV, we can say that ensemble schemes allow to improve this method.

CART: This method obtains second place in the Bagging scheme and the third place in general.

SVM: This method obtains second place in the Adaboost scheme, but it is a bad method when Bagging or Random Subspace are used.

Pegasos: It is the winner method only in the Adaboost ensemble. If we look at the results of the base classifiers Table IV, we can say that ensemble schemes allow to improve this method.

TABLE X: SUMMARY OF THE FINDINGS: RANK FOR EACH BASE CLASSIFIER AND DATA SET GROUPED BY ENSEMBLE

Base	Adaboost	Bagging	Random Subspace	Decorate	Rotation Forest	Avg
LogR	3,6	4,4	4,0	3,6	4,0	3,9
MLP	5,9	5,2	4,4	5,7	5,5	5,3
C4.5	3,7	**2,8**	3,3	2,9	3,0	3,1
CDT	3,8	**2,8**	**2,1**	**2,4**	**2,7**	**2,8**
CART	3,7	2,9	3,6	3,7	3,7	3,5
SVM	3,7	5,0	4,8	4,3	3,9	4,3
Pegasos	**3,5**	4,5	4,9	4,9	4,6	4,5

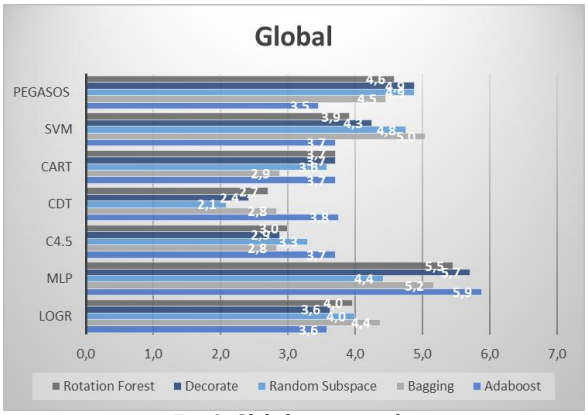

Fig. 9. Global average rank.

VI. CONCLUDING REMARKS

Risk credit has been transformed over the past decade, mainly in response to regulations that occurred from the global financial crisis. Technological innovations continuously arise, enabling new risk management techniques and helping the risk function make better risk decisions at lower cost: Big data, machine learning, and artificial intelligence illustrate the potential impact. This work completes a previous one where other base classifiers are used, also we have presented a survey of credit scoring using ensemble learning. With this objective, seven classification methods and five ensemble approaches have been applied to six credit scoring problems.

In this experimental study, four evaluation metrics, including area under the curve (AUC), accuracy, false positive rate (FPR) and Time taken to build the model, are used to measure the performance of models. As results,

Pegasos algorithm displays the general better performance than the other methods examined in this study for Adaboost, whereas C4.5, CDT and CART models present a general better performance than the other methods analyzed here for Bagging, Random Subspace, Decorate and Rotation Forest. These results confirm that C4.5, CART and CDT algorithms perform the best as base classifier of meta-learning methods studied here. On the other hand, some interesting directions for further research have emerged from this study, such as: (i) to extend the present analysis to other individual classifiers and other ensemble approaches; (ii) to study ensemble methods for credit scoring using a Big Data platform, (iii) to compare between Deep Learning, and ensemble methods in the field of credit rating.

REFERENCES

[1] C. Luo, D. Wu, and D. Wu, "A deep learning approach for credit scoring using credit default swaps," *Engineering Applications of Artificial Intelligence*, 2016.

[2] L. Leonida and E. Muzzupappa, "Do Basel Accords influence competition in the banking industry? A comparative analysis of Germany and the UK," *Journal of Banking Regulation*, 2017.

[3] A. Marqués, V. García, and J. Sánchez, "Exploring the behaviour of base classifiers in credit scoring ensembles," *Expert Systems with Applications*, 2012.

[4] J. Abellán and J. Castellano, "A comparative study on base classifiers in ensemble methods for credit scoring," *Expert Systems with Applications*, 2017.

[5] L. Vanneschi, D. Horna, M. Castelli, and A. Popovic, "An artificial intelligence system for predicting customer default in e-commerce," *Expert Systems with Applications*, vol. 104, pp. 1–21, 2018.

[6] G. Wang and J. Ma, "Study of corporate credit risk prediction based on integrating boosting and random subspace," *Expert Systems with Applications*, vol. 38, issue 11, October 2011.

[7] W. Y. Loh. "Classification and regression trees," *Wiley Interdiscip. Rev.: DataMin. Knowl. Discov.*, vol. 1, no. 1, pp. 14–23, 2011.

[8] L. Yanga, S. Liua, S. Tsokac, and L. Papageorgioua, "A regression tree approach using mathematical programming," *Expert Systems with Applications*, vol. 78, no. 347–357, 2017.

[9] M. Khandelwal, D. Armaghani, R. Faradonbeh, M. Yellishetty, M. Majid, and M. Monjezi, "Classification and regression tree technique in estimating peak particle velocity caused by blasting," *Engineering with Computers*, vol. 33, issue 1, pp. 45–53, 2017.

[10] P. Yao, "Credit scoring using ensemble machine learning," in *Proc. Ninth International Conference on IEEE Hybrid Intelligent Systems*, 2009.

[11] R. Yana, Z. Maa, Y. Zhaob, and G. Kokogiannakisa, "A decision tree based data-driven diagnostic strategy for airhandling units," *Energy and Buildings*, vol. 133, pp. 37–45, 2016.

[12] T. Hayes, S. Usami, R. Jacobucci, and J. McArdle, "Using Classification and Regression Trees (CART) and random forests to analyze attrition: Results from two simulations," *Psychology and Aging*, vol. 30, no. 4, pp. 911-929, 2015.

[13] C. Zhang, Y. Tian, and N. Deng, "The new interpretation of support vector machines on statistical learning theory," *Science China*, vol. 53, no. 1, pp. 151–164, 2010.

[14] A. Priyadarshini and S. Agarwa, "A map reduce based support vector machine for big data classification," *International Journal of Database Theory and Application*, vol. 8, no. 5, pp. 77-98, 2015.

[15] P. Danenasa, G. Garsvab, and S. Gudasc, "Credit risk evaluation model development using support vector based classifiers," *Procedia Computer Science*, vol. 4, pp. 1699–1707, 2011.

[16] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," in *Proc. the Twenty-First International Conference on Machine Learning*, 2004, p. 116.

[17] S. Shai, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for svm. Mathematical programming," vol. 127, no. 1, pp. 3–30, 2011.

[18] L. Nanni and A. Lumini, "An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring," *Expert Systems with Applications*, vol. 36, pp. 3028–3033, 2009.

[19] L. Zhou, K. Lai, and L. Yu, "Least squares support vector machines ensemble models for credit scoring," *Expert Systems with Applications*, vol. 37, pp. 127–133, 2010.

[20] G. Wang, J. Ma, L. Huang, and K. Xu, "Two credit scoring models based on dual strategy ensemble trees," *Knowledge-Based Systems*, vol. 26, pp. 61–68, 2012.

[21] L. Yu, S. Wang, and K. Lai, "Credit risk assessment with a multistage neural network ensemble learning approach," *Expert Systems with Applications*, vol. 34, pp. 1434–1444, 2008.

[22] A. Ghodselahi, "A hybrid support vector machine ensemble model for credit scoring," *International Journal of Computer Applications*, vol. 17, no. 5, 2011.

[23] Y. Xia, C. Liu, B. Da and F. Xie, "A novel heterogeneous ensemble credit scoring model based on bstacking approach," *Expert Systems with Applications*, vol. 93, no. 1, pp. 182-199, 2017.

[2] A. Chowdhury, D. Tjondronegoro, V. Chandran, S. Trost, "Ensemble methods for classification of physical activities from wrist accelerometry," *Med Sci Sports Exerc*, 2017.

[24] A. Makkar and S. Goel, "Spammer classification using ensemble methods over content-based features," in *Proc. Sixth International Conference on Soft Computing for Problem Solving*, Springer, Singapore, 2017, vol. 547.

[25] R. Schapire. "The boosting approach to machine learning: An overview, nonlinear estimation and classification," *Lecture Notes in Statistics*, vol. 171, Springer, New York, 2003.

[26] S. Sadatrasoul, M. Gholamian, M. Siami, and Z. Hajimohammadi, "Credit scoring in banks and financial institutions via data mining techniques," *Journal of AI and Data Mining*, vol. 1, issue 2, pp. 119-129, 2013.

[27] T. Chen, "XGBoost: A scalable tree boosting system," in *Proc. the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785-794, 2016.

[28] M. Patel, M. Panchal, and H. Bhavsar, "Decorate ensemble of artificial neural networks with high diversity for classification," *International Journal of Computer Science and Mobile Computing*, vol. 2, issue 5, pp. 134-138, 2013.

[29] P. Melville and R. Mooney, "Creating diversity in ensembles using artificial data," *Journal of Information Fusion: Special Issue on Diversity in Multi Classifier Systems*, vol. 6, pp. 99-111, 2004.

[30] J. Rodriguez, L. Kuncheva, and C. Alonso, "Rotation forest: A new classifier ensemble method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, 2006.

[31] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, March 2015.

[32] A. L'Heureux, K. Grolinger, H. F. Elyamany, and M. A. M. Capretz, ''Machine learning with big data: Challenges and approaches,'' *IEEE Access*, vol. 5, pp. 7776–7797, 2017.

[33] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

Youssef Tounsi is currently a PhD student at RITM Laboratory, CED Engineering Sciences, Ecole Supérieure de Technologie, Hassan II University of Casablanca, Morocco. He received his master degree in computer science from ENIM Rabat in 2004. His current research interests include credit scoring using machine learning algorithms. He is also a banking consultant with experience in ERP, project management, artificial intelligence and analytics.

Houda Anoun is an assistant professor in computer science at ESTC (Ecole Supérieure de Technologie, Hassan II University, Casablanca) since 2009. She received her PhD degree in computer science from the University of Bordeaux I, France in 2007. Her research interests focus on data mining, big data and computational linguistics.

Larbi Hassouni got the Ph.D in computer science from AIX MARSEILLE University-France and currently he is a teacher-researcher at the RITM laboratory of the Center for Doctoral Studies in Engineering Sciences at Hassan II University-Morocco.