The Naïve Associative Classifier With Epsilon Disambiguation

ADOLFO RANGEL-DÍAZ DE LA VEGA¹, YENNY VILLUENDAS-REY¹,², CORNELIO YÁÑEZ-MÁRQUEZ¹, AND OSCAR CAMACHO-NIETO²

¹Centro de Investigación en Computación del Instituto Politécnico Nacional, Mexico City 07738, Mexico
²Centro de Innovación y Desarrollo Tecnológico del Instituto Politécnico Nacional, Mexico City 07738, Mexico

Corresponding author: Yenny Villuendas-Rey (yenny.villuendas@gmail.com)

ABSTRACT
This paper presents the Naïve Associative Classifier with Epsilon disambiguation (NACε), an extension of the Naïve Associative Classifier that, by including a procedure to disambiguate classes in regions where Bayes risk is high, has a positive effect on the performance of the classifiers of the associative approach on several datasets belonging to the financial environment, particularly in terms of credit risk. The experiments conducted to test the NACε were based on 12 datasets composed with financial information and associated with five stages of the credit process: promotion, evaluation, granting, monitoring and recovery. Due to the severe imbalance present in most datasets, the performance of the proposed algorithm was measured using the area under the ROC curve. Likewise, 5 × 2 stratified cross validation was made and finally a couple of statistical tests were applied to compare the results. After applying the NACε to the datasets, a successful disambiguation of classes was observed. In the real world this fact could help financial institutions to evaluate the credit applications more effectively and thus, contribute to the mitigation of monetary losses derived from the poor quality of the information.

INDEX TERMS
Credit-scoring, financial risks, supervised classification.

I. INTRODUCTION
Financial risk includes different approaches: market risk, credit risk, liquidity risk and operational risk, and has been studied since the banking institutions were created. Starting in the 1950’s, analysis and management of financial risks have been turning more relevant in the global economy, although as shown by the financial crises observed since then, these tasks have not fulfilled their mission; as a consequence, the analysis of financial risks is a current topic of research [1]–[23].

Concerns about credit risk have increased, becoming an on-topic issue among various financial institutions. These institutions seek to efficiently maximize potential benefits, which turns out to be a complicated task given the conditions of competition [24].

Credit default events arise when a borrower is unable to partially or fully cover its contractual obligations in terms of the total debt or the due date. Credit risk is measured in terms of the potential loss derived from any event of that kind, and is a value expressed in monetary units.

Considering that potential loss interpretation may be subjective for anyone external to the evaluated transaction, frequently the risk level is reported in terms of rating or credit scoring [25].

Intelligent Computing (IC) algorithms are useful when searching computational solutions to financial risk related problems. Currently, several research papers address this type of problem; specifically: credit risk [2], [3], bankruptcy [1], [4], [5] or marketing [6], [7], among others.

Financial risk forecasting can be interpreted as a pattern classification problem [1]. Considering that the associative approach of intelligent pattern classification has currently acquired great popularity because of its effectiveness and efficiency, this document is based on one of the most recent associative classifier: the Naïve Associative Classifier (NAC) [26].

Recently created, the NAC exhibits characteristics that represent advantages over other state-of-the-art classifiers. It is simple, transparent, transportable and precise, and is capable of handling numerical, categorical and mixed data, in addition to missing values. It has been applied in financial
datasets, including credit allocation, marketing, bankruptcy and banknote authentication [26].

Despite the above, the NAC has a notable disadvantage: under certain conditions, its results exhibit ambiguity in the classes of the objects to be classified. Therefore, in order to face the mentioned disadvantage, we present through this document the Naïve Associative Classifier with Epsilon Disambiguation (NACε), which includes a successful process of disambiguation of classes, thus improving the performance of the Naïve Associative Classifier (NAC). This proposal has important repercussions on the performance of associative approach classifiers, in different datasets belonging to the financial environment.

The paper is structured as follows: Section II reviews some of the most relevant related works. Section III details the proposed model, section IV explains the experimental design, while section V discusses the obtained results. The paper ends with conclusions at section VI.

II. RELATED WORKS

Through IC algorithms, financial risk patterns can be analyzed in the context of four basic tasks: classification, regression, recovery and clustering [27]–[32]. This document emphasizes the task of intelligent classification of patterns in financial data.

IC algorithms, mainly intelligent classifiers of patterns [33] are theoretically supported by several conceptual bases: Bayesian classifiers, based mainly on the Bayes Theorem, such as the proposals of Berrar [34] and Ala‘raj and Abbod [35]; classifiers based on the nearest k neighbors (k-NN), and one of whose emblematic algorithms, 1-NN, proposed by Cover and Hart [31], was used in the experimental part of this document; decision trees, such as C4.5, widely used for their simplicity and effectiveness; additionally, models that imitate nature, implemented to classify patterns or optimize pattern-classifying algorithms through metaheuristics, among which neuronal classifiers stand out; there are also algorithms that are based on logical functions or on the optimization of analytic functions, such as vector support machines by Chen at al. [36]. In 1968, one of the first attempts to apply IC in financial issues was the development of the Z-Score, proposed by Altman [37].

A recurring topic in financial risk management is bankruptcy, because investors (financial institutions, firms or even common people) need to minimize the possibility of not receiving the expected dividends or even not recovering their capital. Estimation of a firm’s bankruptcy may be addressed as a problem of pattern classification, as described by Chen et al. [1], where a bankruptcy prediction model using diffuse k-NN is proposed.

Credit risk has been the subject of several scientific publications. A computational model to improve corporate credit rating systems oriented to help financial experts in decision making is presented by Petropoulos et al. [2], where two important aspects of financial data are considered: the distribution of heavy tails and their time series nature, for which they propose a system based on Markov models; a new scheme for the modeling of financial time series is also proposed.

Another approach, based on social media, and proposed by Yang and Zhou, is shown in [3], which analyzes whether the different opinions shared by users about different financial topics are relevant or not for the prediction of credit risk. Opinions were analyzed from two social networks for financial investors in China, in addition to various articles published by financial analysts. The authors showed that opinions published online can strongly predict the credit risk of companies, while the analysis of the opinions of financial analysts obtained lower performance.

Regarding the classification of clients, a model to help in the prediction of desirable clients (those who will fulfill their payment obligations, labeled as good) and undesirable clients (those who are likely not to fulfill their payment obligations because they are bankrupted, labeled as bad) is proposed by Gahlaut et al. [5]. Data mining techniques are used to select the best attributes that can support the credit rating, in order to identify good clients.

Credit granting is the core business of retail banking and other financial institutions, and therefore they should take advantage of any resource that gives them competitive advantages and focus on creating long-term relationships with their customers instead of just promoting products individually. Ladyzynski et al. [38] proposed a time-series-based approach to predict the willingness of individuals to hire personal credit. For this purpose, historical information of banking transactions of clients is used to detect significant patterns.

Referring to marketing, data mining techniques are proposed by Moro et al. [6] to predict the outcome of a telephone call in which a banking service is offered. We used a dataset that contains information from a telemarketing campaign carried out by a Portuguese bank from 2008 to 2013. Another similar proposal is presented in [7] by the same authors, but applying artificial neural networks.

A conceptual foundation of IC that is relevant for this document is that in which associative models are based. The first associative model registered in the annals of literature is the Lernmatrix, created in 1961 by Karl Steinbuch; since then, several associative models with applications in various fields has been generated, as the one by Yáñez-Márquez et al. [39]. In the Computational Research Center of the National Polytechnic Institute (México), the Alpha-Beta branch of associative models was created in 2002 [39]; one of the most recent models, the NAC classifier by Villuendas-Rey et al. [26], is the foundation for the proposed model.

Despite the numerous techniques for credit risk analysis, there is still a need for transportable and transparent models, able to deal with both mixed and incomplete datasets [26]. In addition, the NAC model (which fulfills the above requirements) has some difficulties in classification where the overall class similarities are too close. To overcome these limitations, we introduce the Naïve Associative Classifier with Epsilon disambiguation (NACε).
III. PROPOSED MODEL

The Naïve Associative Classifier with Epsilon disambiguation (NAC£) is based on the Naïve Associative Classifier (NAC) [26]. This proposal aims to solve NAC’s shortcomings by including a class disambiguation process in zones where the Bayes risk is high.

A. TRAINING PHASE

The proposed model allows to handle mixed and incomplete data, and during its training phase it is assumed that:

- There is a training set $T$, whose elements are instances described by a set of attributes $A = \{A_1, \ldots, A_m\}$.
- Each instance in $T$ belongs to one and only one class in the set of classes $K = \{K_1, \ldots, K_p\}$.
- Set $A$ must be associated to a set of weights $w = \{w_1, \ldots, w_m\}$.
- No attribute has only missing values and no instance has its entire attribute values missing.

The training phase of the NAC£ is done through two steps:

1) Storage of the training set $T$.
2) Computation and storage of the standard deviation $\sigma_{ij}$ for each numeric attribute $A_i$ in each class $K_j$.

B. CLASSIFICATION PHASE

Let $o$ be an instance with unknown class. The classification of such instance (Fig. 2) takes two steps.

Step 1 – Obtaining the average similarities

First, the average similarity of $o$ against each available class is computed:

Let $T_j$ be the set of instances belonging to the class $K_j$. The average similarity between an instance $o$ and $K_j$ is given by:

$$s_j(o) = \frac{1}{|T_j|} \sum_{y \in T_j} s'(o, y)$$

(1)

where $s'$ is a similarity function between two patterns $o$ and $y$.

Let $w_i$ be the weight of attribute $A_i$ and let $m$ be the total number of attributes. The total similarity function is given by:

$$s'(o, y) = \sum_{i=1}^{m} w_i \star \text{MMIDSO}(o, y, A_i)$$

(2)

MMIDSO (Modified Mixed and Incomplete Data Similarity Operator) is an extension of NAC’s operator (MIDSO), and as MIDSO, it can handle missing values as well as categorical and numerical attributes. However, MMIDSO considers the standard deviation of each attribute for each class.

MMIDSO separates categorical and numerical attributes:

$$\text{MMIDSO}(o, y, A_i) = \begin{cases} 
  s_c(o, y, A_i) & \text{if } A_i \text{ is categorical} \\
  s_n(o, y, A_i) & \text{if } A_i \text{ is numerical} 
\end{cases}$$

(3)

$$s_c(o, y, A_i) = \begin{cases} 
  0 & \text{if } (o_i \neq y_i) \vee (o_i = '?') \vee (y_i = '?') \\
  1 & \text{otherwise}
\end{cases}$$

(4)

$$s_n(o, y, A_i) = \begin{cases} 
  0 & \text{if } [(|o_i - y_i| > \sigma_{ij}) \vee (o_i = '?') \vee (y_i = '?')] \\
  1 & \text{otherwise}
\end{cases}$$

(5)

It should be noted that standard deviation $\sigma_{ij}$ of a numerical attribute $A_i$ is computed without considering instances containing a missing value in attribute $i$.

Step 2 – Disambiguation

Once $s_j(o)$ has been obtained for each $j$, the second step consists in defining in which class does the test instance $o$ suits better. To do so, the average similarities must be sorted in a descending manner in order to select the first and the second one.

Let $st_1$ and $st_2$ be the first and the second sorted similarities and let $\varepsilon \in [0,1]$ be a user-defined disambiguation value. On determining the existence of a significant difference between $st_1$ and $st_2$, the ratio between them will be considered. If such ratio is higher than $\varepsilon$, then no ambiguity in class assignment is considered. Otherwise, ambiguity existence is considered.
Formally, the disambiguation function, denoted by $des$ and with parameters $st_1$, $st_2$ and $\varepsilon$, is defined as follows:

$$des(st_2, st_1, \varepsilon) = \begin{cases} 
0 & st_2 > \varepsilon \text{ (no ambiguity)} \\
1 & st_1 \leq \varepsilon \text{ (ambiguity)}
\end{cases}$$

(6)

If there is no ambiguity, the class of greatest similarity is returned. Otherwise, the ambiguity is eliminated through one out of two variants proposed here.

Variant 1. Based on the total similarity, determine which of all the instances of the training set is the most similar to $o$ and return its class.

Variant 2. Apply a user-defined dissimilarity function and determine which of all objects in the training set is closest to $o$ and return its class.

C. COMPLETE PSEUDOCODE OF THE ALGORITHM

To better understand the proposed algorithm and its variants, we will use their pseudocode (Fig. 3 and Fig. 4).

D. COMPLEXITY OF THE ALGORITHM

The efficiency of NAC$\varepsilon$ can be measured in terms of space complexity or time complexity. Space complexity is measured from the storage of the training set as well as the storage of the standard deviation of each class in each attribute. Hence, the space complexity is bounded by $O(n \times m) + O(k \times m)$, where $n$ is the number of instances, $m$ is the number of attributes and $k$ is the number of classes.

On the other hand, time complexity includes training and classification phases. Training phase complexity derives from the computation of the standard deviation of each attribute at a class level; then, for $m$ attributes, the complexity is bounded by $O(m \times k)$.

In turn, classification phase comprises two steps: obtaining average similarities and disambiguation. The complexity of classification is measured from these steps. The step in which the average similarities are obtained compares the similarity of the test instance with the instances of the training set, and its complexity is given by $O(n \times s)$, where $s$ is the complexity of computing the similarity between attributes. However, the computational complexity of the similarity is constant and defined as $O(1)$ because it is a simple operation, and therefore the complexity of obtaining the average similarities is bounded by $O(n)$.

The second step, disambiguation, includes finding the two higher similarities, bounded by $O(2n)$, the computation of the disambiguation function, bounded by $O(n)$ and, eventually, the application of some disambiguation variant, also bounded by $O(n)$, that is, the total time complexity of classification phase is bounded by $O(2n)$.

IV. EXPERIMENTAL DESIGN

Along this section, the validation processes are described, and the obtained results are discussed.
**TABLE 1. Description of the used datasets.**

| Dataset                  | Instances | Attributes | Missing values | Imbalance ratio |
|--------------------------|-----------|------------|----------------|-----------------|
| Australian               | 690       | 14         | 0              | 1.248           |
| Bank                     | 4,521     | 16         | 5,254          | 7.678           |
| Bank add                 | 4,119     | 20         | 1,230          | 8.133           |
| Bank add full            | 41,188    | 10         | 12,718         | 7.877           |
| Bank full                | 45,211    | 16         | 52,124         | 7.548           |
| Japanese                 | 690       | 15         | 67             | 1.248           |
| Banknote                 | 1,372     | 4          | 0              | 1.249           |
| Default                  | 30,000    | 23         | 0              | 3.521           |
| German                   | 1,000     | 20         | 0              | 2.333           |
| Iranian                  | 1,000     | 28         | 101            | 19              |
| Polish I                 | 7,027     | 64         | 5,835          | 24.930          |
| Qualitative              | 250       | 6          | 0              | 1.336           |

**FIGURE 5. Confusion matrix for a 2-class problem.**

### A. DATASETS

Several datasets containing financial data were used; almost all of them are available in the machine learning repository of the University of California at Irvine. The information contained in the datasets is related to three phases of the credit process: granting, promotion (specifically fund raising) and recovery.

Accordingly, Australian-credit, Japanese-credit, Default-of-credit-card, German-credit, Iranian-credit [40] include information that is useful when authorizing or rejecting the granting of credit. The first two are variations of Credit-approval, which is a dataset consisting of information collected to evaluate credit card applications.

The Bank datasets, donated by Moro [6], contain information related to a telephone campaign from a Portuguese bank seeking to fundraise financial resources via long-term deposits; there are two classes that describe if the product offered was hired or not.

Even that recovery stage is strongly related to payment collection, it includes prospecting of clients through analyzing the financial health of every credit applicant or current borrowers. Given the importance of the foregoing, loss prediction has been studied from various approaches, such as financial ratio analysis. This approach is exemplified through the Polish-companies-bankruptcy-data. Another default forecasting approach considers a qualitative point of view and is based on the comparison of categorical parameters assigned by “experts” to hard-to-quantify variables. The Qualitative-bankruptcy dataset, containing 250 instances described by six attributes, is an example of it.

Finally, we explored the Banknote-authentication dataset. This is not a financial risk problem, however, the characteristics of this dataset are suitable for testing the model proposed in this document.

Table 1 describes the characteristics of each dataset and highlights the imbalance ratio in most of them; an imbalance ratio greater than 9 means that most algorithms will not perform well when classifying patterns belonging to the minority class.

### B. PERFORMANCE MEASURES

In order to deal with the problem of imbalanced classes, frequently observed in datasets related to the financial environment, we use the area under the ROC curve (AUC) to evaluate the performance of NACε, since AUC is independent of the class distribution of the dataset [41], [42]. AUC is directly based on the confusion matrix, which is described in Figure 5. In addition, AUC is based on averaging sensitivity (TPR) and specificity (TNR).

Confusion matrix describes the possible cases when classifying a problem with two classes. True Positives (TP) are instances labeled by the classifier as positive when they are positive indeed. False Positives (FP) occur when the classifier labels an instance as negative, but it is, in fact, positive.

A False Negative (FN) is a negative instance classified as positive and True Negatives (TN) happen when an instance is classified as negative and it is negative indeed. It has been proved [42] that AUC can be calculated as the average of the True Positive Rate (TPR) and the True Negative Rate (TNR).

Hence, the AUC is given by:

\[
AUC = \frac{TPR + TNR}{2} 
\]

\[
TPR = \frac{TP}{TP + FN} 
\]

\[
TNR = \frac{TN}{FP + TN} 
\]

Regarding data partition, the stratified cross validation 5×2 (5 × 2 scv) was chosen because it has been recommended to deal with imbalanced data [43], [44]. When using 5 × 2 scv, the training set is stratified and divided into two subsets of the same size: one for training and the other one for testing. This process is repeated five times, resulting in ten partitions (five for training and five for testing). Finally, to measure the performance of the classifiers, the average of the performance in each of the five partitions is considered.

### C. STATISTICAL TESTS

To compare the results of classification algorithms on several data sets, the use of nonparametric statistical Friedman test for more than two related samples is considered appropriate [45]–[48]. If the number of samples used in the experiments is small, statistical tests lack of power to determine the existence of statistically significant differences between the performance of the algorithms. The required number of samples to obtain satisfactory results is given by the expression samples ≥ 2 * algorithms to compare [48].

In this research only 12 data sets were used, which means that at most six algorithms can be compared through the Friedman test, without losing statistical power. The application of Friedman test implies the creation of a block for each one of the analyzed subjects in such a way that each...
block contains an observation coming from the application of each one of the different contrasts or treatments. In terms of matrices, the blocks correspond to rows and treatments to columns. The null hypothesis establishes that the performances obtained by different treatments are equivalent, while the alternative hypothesis proposes that there is a difference between these performances, which would imply differences in the central tendency.

If the Friedman test determines the existence of significant differences between the performance of the algorithms, a post hoc test is highly recommended to find in which of the compared algorithms those differences exist. Among several post hoc tests suggested [48], we chose the Holm Test [49]. The Holm method is designed to reduce the Type I errors when analyzing phenomena that include several hypotheses and consists of adjusting the rejection criteria for each one of them.

V. RESULTS AND DISCUSSION

The results obtained by applying the proposed model to the different datasets are shown below. First, the performance of the algorithm is illustrated from different values of $\epsilon$ for each of the two variants of the model. In the second variant, the dissimilarity function HEOM, proposed by Wilson and Martinez [50], was used. Subsequently, the results obtained are compared with NAC [26] and Nearest Neighbor (NN) [31] classifiers, because the proposed model constitutes a generalization of them, as well as with respect to other classifiers. Finally, the results obtained are compared with other state-of-the-art classifiers. In each case, the corresponding statistical tests were applied.

A. ANALYSIS OF THE IMPACT OF $\epsilon$ VALUE

We tested the performance of first variant of the proposed model, in terms of several values of the disambiguation parameter. AUC was considered as a performance measure. After applying Friedman Test over the results obtained from the compared data sets with different $\epsilon$ values, the resulting $p$-value was $p = 0.0998057$, which means that the null hypothesis is not rejected and therefore, in terms of the performance of the first variant, there are no significative differences when choosing the $\epsilon$ parameter.

The performance of second variant of the proposed model was also tested, in terms of different values of $\epsilon$. AUC was again considered as a performance measure. Consequently, the Friedman Test was applied to establish the existence, if any, of significant differences.

After applying the Friedman Test with the results obtained from different $\epsilon$ values in the compared data sets. The resulting $p$-value was $p = 0.05484$, which is very close to the defined significance level ($\alpha = 0.05$); because of that, even that the null hypothesis is not rejected, the results cannot be considered conclusive. This is why we apply the Holm test, comparing the best ranked algorithm ($\epsilon = 0.2$) with the rest. Table 2 shows these results. Holm Test rejects hypotheses which non adjusted Holm probability value $\leq 0.007143$.

Thus, the mentioned test rejects the null hypotheses of means equality ($H_0$) when comparing the second variant of the proposed model with $\epsilon = 0.2$ against the algorithm with $\epsilon = 0.8$, and the algorithm with $\epsilon = 0.9$, respectively. The conducted experiments suggest that it is preferable to use $\epsilon$ values less than 0.8 in the second variant, however it may be convenient to perform more experiments to establish if there are significant differences between the first and last values in the obtained ranking.

B. COMPARISON OF THE PROPOSED MODEL WITH OTHER CLASSIFIERS

On the other hand, this research analyzes the performance of the proposed model with respect to the NAC and Nearest Neighbor (NN) models. We also compared the proposed model with respect to other well-known classifiers, such as Naive Bayes (NB), Multilayer Perceptron (MLP), and Support Vector Machines (SVM). In addition, we include a Deep Neural Network (DL) architecture, as a reviewer suggestion. These results are shown in Table 3. In each case, the best disambiguation values for each data set were chosen for the proposed model. The classifiers of the state of art were executed using the Weka software [51]. We use the default parameters for all of them.

Once again, the Friedman Test was applied to find if there were significant differences in the performance of the

### Table 2. Post hoc comparison between the best ranked $\epsilon$ (0.2) in the second variant and the rest of $\epsilon$ values.

| $\epsilon$ | $z$   | Holm | Decision |
|------------|-------|------|----------|
| 0.9        | 2.775638 | 0.005509 | 0.006250 | Reject HO |
| 0.8        | 1.646018 | 0.099760 | 0.007145 | Reject HO |
| 0.7        | 0.387298 | 0.698535 | 0.008333 | Do not reject HO |
| 0.6        | 0.225924 | 0.821261 | 0.010900 | Do not reject HO |
| 0.5        | 0.193649 | 0.846451 | 0.012500 | Do not reject HO |
| 0.4        | 0.000000 | 1.000000 | 0.016667 | Do not reject HO |
| 0.3        | 0.000000 | 1.000000 | 0.025000 | Do not reject HO |
| 0.2        | 0.000000 | 1.000000 | 0.050000 | Do not reject HO |

### Table 3. AUC results of the compared algorithms.

| Dataset       | NB       | NN       | MLP       | SVM       | NAC       | DL       | NAC $\epsilon$ |
|---------------|----------|----------|-----------|-----------|-----------|----------|----------------|
| Australian    | 0.77     | 0.74     | 0.82      | 0.56      | 0.84      | 0.85     | 0.84           |
| Bank          | 0.77     | 0.57     | 0.75      | 0.60      | 0.73      | 0.65     | 0.77           |
| Bank add      | 0.77     | 0.62     | 0.68      | 0.77      | 0.72      | 0.71     | 0.72           |
| Bank add full | 0.73     | 0.65     | 0.70      | 0.60      | 0.73      | 0.70     | 0.73           |
| Bank full     | 0.76     | 0.65     | 0.69      | 0.64      | 0.72      | 0.42     | 0.72           |
| Japanese      | 0.76     | 0.81     | 0.84      | 0.55      | 0.84      | 0.85     | 0.84           |
| Banknote      | 0.84     | 0.72     | 0.99      | 0.98      | 0.85      | 0.98     | 0.93           |
| Default       | 0.64     | 0.56     | 0.59      | 0.64      | 0.66      | 0.60     | 0.66           |
| German        | 0.72     | 0.59     | 0.72      | 0.63      | 0.69      | 0.69     | 0.69           |
| Iranian       | 0.51     | 0.53     | 0.88      | 0.62      | 0.58      | 0.50     | 0.95           |
| Polish        | 0.41     | 0.50     | 0.50      | 0.50      | 0.50      | 0.50     | 0.60           |
| Qualitative   | 0.98     | 0.99     | 0.99      | 1.00      | 0.99      | 1.00     | 0.99           |
| Times Best    | 5        | 0        | 2         | 2         | 3         | 5        |                |

1 A Type I error occurs when the null hypothesis is rejected although it is true.
TABLE 4. Friedman ranking of the compared algorithms.

| Algorithm | Ranking |
|-----------|---------|
| NAC ε     | 2.58    |
| NAC       | 3.38    |
| MLP       | 3.63    |
| DL        | 3.96    |
| NB        | 4.00    |
| SVM       | 4.58    |
| NN        | 5.88    |

TABLE 5. Post hoc comparison using HOLM’s test.

| i | Algorithm | Z   | p       | Holm | Decision                  |
|---|-----------|-----|---------|------|----------------------------|
| 6 | NN        | 3.7643 | 0.000167 | 0.008333 | Reject $H_0$              |
| 5 | SVM       | 2.4004 | 0.016377 | 0.010000 | Reject $H_0$              |
| 4 | NB        | 1.8003 | 0.071814 | 0.012500 | Reject $H_0$              |
| 3 | DL        | 1.5591 | 0.118972 | 0.016667 | Do not reject $H_0$       |
| 2 | MLP       | 1.2002 | 0.230062 | 0.025000 | Do not reject $H_0$       |
| 1 | NAC       | 0.9820 | 0.326109 | 0.050000 | Do not reject $H_0$       |

The resulting $p$-value was $p = 0.011548$, which is lower than the defined significance level ($\alpha = 0.05$), which means that the means equality null hypothesis is rejected. Thus, the Holm Test was applied to compare the best ranked algorithm (NACε) against the rest. It can be observed in Table 5 that hypotheses with unadjusted Holm probability value $\leq 0.0125$ are rejected. Therefore, null hypotheses of mean equality when comparing the proposed model against Nearest Neighbor, Support Vector Machines and Naïve Bayes classifiers, are rejected.

According to the AUC, the conducted experiments show that the proposed model has a better performance than NAC. It outperformed the NAC classifier in four datasets, and obtained a similar result in the remaining eight datasets. However, the Holm test did not found significant differences.

The model does not show significant differences in terms of performance using the AUC measure with respect to the Multilayer Perceptron and Deep Learning. However, the proposed model has a lower computational cost than MLP and DL, and maintain the desired characteristics of NAC of being transparent and transportable, extremely valuable for financial environments.

C. CONTRIBUTIONS TO THE CREDIT PROCESS

Throughout the second decade of the 21st century, the process of analyzing credit applications has evolved. The traditional way to authorize or reject the granting of a loan has been influenced by the irruption of technology and by the plethora of data, causing that expert judgment, that which credit analysts acquired based on repeating the same task hundreds or perhaps thousands of times, fall into obsolescence to make way for automated credit scoring systems. Ideally, based on the information of each applicant and several parameters, an automated credit scoring system can decide whether the granting of a loan implies a risk level greater than that one the lender is willing to take, and consequently, suggest a rejection.

A system with these characteristics oversees, implicitly, compliance with the regulatory policies for financial risk management, but also optimizes the evaluation stage by allowing the simultaneous management of several applications and increasing business opportunities. However, there is still a lot to do.

Frequently, the information used to train the scoring models lacks precision and certainty, since it is part of datasets whose content is considered sensitive because they describe the equity of firms (or the wealth of persons) and their consuming patterns. Also, the collection of information that applicants provide when asking for a loan remains nowadays as a task exposed to human error, which can arise for many reasons: regular tiredness of persons in charge of data capturing, inadequate design of a questionnaire that may contain questions whose subjective response can be misunderstood or misinterpreted by any analyst. These imperfections can be solved through the application of the NACε, because it has been designed to handle incomplete information and therefore reduce the chances in which analysts can misinterpret or decide in a wrong way.

VI. CONCLUSION

Despite of its simplicity, the proposed model shows ability to solve problems of financial data classification and obtains competitive results. The computational cost is low and the interpretability and transportability requirements that are needed in financial scenarios are fully covered, since the model has the capacity to directly handle mixed or incomplete data.

The proposed model successfully disambiguated classes, which makes easier the handling of data sets with a high degree of imbalance, missing values or heterogeneity in terms of the features that describe them. In addition, the statistical tests carried out support the claim of the proposed model being more accurate than Nearest Neighbor, Support Vector Machines and Naïve Bayes classifiers, for financial data.

Although it is mandatory the suggestion to apply NACε in classifying tasks related to disciplines unrelated to financial risk management, the extent of the latter offers, on its own, a lot of research alternatives.

Financial risk is inherent to economic activity. Unlike other disciplines, economic science is directly influenced by human conduct. This fact adds a chaotic factor to the studied phenomena and complicates classification and regression tasks. The random behavior of economic and financial variables is highly unpredictable, which makes imperative to always consider the factors that modify trends and historical behavior. Incorporating those factors and their dynamics to the intelligent computing algorithms poses great challenges and new phenomena to study.

As future work, we want to explore the impact of feature weighting in the proposed algorithm.
ACKNOWLEDGMENT

The authors would like to thank the Instituto Politécnico Nacional (Secretaría Académica, Comisión de Operación y Fomento de Actividades Académicas, Secretaría de Investigación y Posgrado, Centro de Investigación en Computación, y Centro de Innovación y Desarrollo Tecnológico en Cómputo), the Consejo Nacional de Ciencia y Tecnología, y el Sistema Nacional de Investigadores for their economic support to develop this work.

REFERENCES

[1] H.-L. Chen, B. Yang, G. Wang, J. Liu, X. Xu, S.-J. Wang, and D.-Y. Liu, “A novel bankruptcy prediction model based on an adaptive fuzzy k-nearest neighbor method,” Knowl.-Based Syst., vol. 24, no. 8, pp. 1348–1359, Dec. 2011.

[2] A. Petropoulos, S. P. Chatzis, and S. Xanthopoulos, “A novel corporate credit rating system based on Student’s t hidden Markov models,” Expert Syst. Appl., vol. 53, pp. 87–105, Jul. 2016.

[3] W. Fei, J. Gu, Y. Yang, and Z. Zhou, “Credit risk evaluation based on social media,” Procedia Comput. Sci., vol. 55, pp. 725–731, 2015.

[4] V. García, A. I. Marqués, and J. S. Sánchez, “Exploring the synergetic effects of sample types on the performance of ensembles for credit risk and corporate bankruptcy prediction,” Inf. Fusion, vol. 47, pp. 88–101, May 2019.

[5] A. Gahlaut, Tushar, and P. K. Singh, “Prediction analysis of risky credit using data mining classification models,” in Proc. ICCCNIT, New Delhi, India, Jul. 2017, pp. 1–7. [Online]. Available: https://ieeexplore.ieee.org/document/8203982.

[6] S. Moro, P. Cortez, and P. Rita, “A data-driven approach to predict the success of bank telemarketing,” Decis. Support Syst., vol. 62, pp. 22–31, Jun. 2014.

[7] S. Moro, P. Cortez, and P. Rita, “Using customer lifetime value and neural networks to improve the prediction of bank deposit subscription in telemarketing campaigns,” Neural Comput. Appl., vol. 26, no. 1, pp. 131–139, Jan. 2015.

[8] J. Luo, H. Chen, Q. Zhang, Y. Xu, H. Huang, and X. Zhao, “An improved grasshopper optimization algorithm with application to financial stress prediction,” Appl. Math. Model., vol. 64, pp. 654–668, Dec. 2018.

[9] H. Zaheer and M. Pant, “Solving portfolio optimization problem through support vector machine with genetic algorithms to optimize investments,” Decis. Support Syst., vol. 68, pp. 223–234, Feb. 2016.

[10] Y. Zelenkov, E. Fedorova, and D. Chekrizov, “Two-step classification method based on genetic algorithm for bankruptcy forecasting,” Expert Syst. Appl., vol. 88, pp. 393–401, Dec. 2017.

[11] Y. Zelenkov, E. Fedorova, and D. Chekrizov, “Two-step classification method based on genetic algorithm for bankruptcy forecasting,” Expert Syst. Appl., vol. 88, pp. 393–401, Dec. 2017.

[12] N. Metawa, M. K. Hassan, and M. Ehloussen, “Genetic algorithm based model for optimizing bank lending decisions,” Expert Syst. Appl., vol. 80, pp. 75–82, Sep. 2017.
YENNY VILLUENDAS-REY received the B.S. and M.S. degrees in computer science from the University of Ciego de Ávila, Cuba, in 2005 and 2007, respectively, and the Ph.D. degree in technical sciences from the Universidad Central “Marta Abreu” de Las Villas, in 2014. Since 2015, she has been working as a Researcher and Professor at the CIDETEC, Instituto Politécnico Nacional, Mexico. She is the author of two books and more than 20 articles. Her research interests include data analysis, hybrid and incomplete data management, classification, bio-inspired algorithms, and soft computing.

Dr. Villuendas-Rey is a member of the Cuban Association for Pattern Recognition (ACRP). She received the national prize to the Scientific Research, the National Mention to the Best Young Research in Technical Sciences, and the National Prize to the Best Student in Technical Sciences, all from the Cuban Academy of Sciences, in 2015, 2014, and 2005, respectively. In addition, she received the Annual Rector Award to the Best Researcher from the Rector of the University of Ciego de Ávila, in 2012, 2013, and 2014, respectively. She is also a recipient of the National Researcher System (SNI) level C from CONACYT, Mexico.

CORNELIO YÁÑEZ-MÁRQUEZ received the bachelor’s degree in physics and mathematics from the Physics and Mathematics Superior School, Instituto Politécnico Nacional (IPN), Mexico, in 1989, and the M.Sc. and Ph.D. degrees from the IPN Center for Computing Research (CIC), in 1995 and 2002, respectively. He is currently a Researcher Professor, Titular C, and the Chief of the Intelligent Computing Laboratory at CIC, Instituto Politécnico Nacional, Mexico. He established the Alpha-Beta research group, and is the leader of the Alpha-Beta models development. He is the author of 30 books and more than 60 scientific articles. His areas of interests are associative memories, neural networks, machine learning, and software engineering.

Dr. Yáñez-Márquez was granted the “Lázaro Cárdenas Award” by the President of the Mexican Republic. He is also a recipient of the National Researcher System (SNI) level II from CONACYT, Mexico.

OSCAR CAMACHO-NIETO received the bachelor’s degree in communications and electronics engineering from the Superior School of Mechanical and Electrical Engineering (ESIME), Instituto Politécnico Nacional (IPN), in 1989, and the M.Sc. degree in computer engineering and the Ph.D. degree in computer sciences from the IPN Center for Computing Research (CIC), in 1995 and 2003, respectively. He is currently a Researcher Professor, Titular C, at IPN, and also the Head of the IPN-DEC. He is the author of three books and more than 30 articles. His areas of interests include computer architectures, associative memories, microprocessors, digital systems, and neural networks.

Dr. Camacho-Nieto was granted with several recognitions from the IPN, Mexico. He is also a recipient of the National Researcher System (SNI) level I from CONACYT, Mexico.