Benchmarking Classifiers for Loan Default Prediction using Archetypal Analysis

Anastasios Antoniadis
School of Informatics, Aristotle University, Greece
anastasios@csd.auth.gr

Konstantinos Georgiou
School of Informatics, Aristotle University, Greece
konsgeor@csd.auth.gr

Konstantinos Charmanas
School of Informatics, Aristotle University, Greece
kcharman@csd.auth.gr

Nikolaos Mittas
Department of Chemistry, International Hellenic University, Greece
nmittas@chem.ihu.gr

Lefteris Angelis
School of Informatics, Aristotle University, Greece
lef@csd.auth.gr

ABSTRACT

The prediction of loan default is a critical process for the successful development of financial institutions. To effectively manage credit risk, numerous machine learning models have been employed to distinguish creditworthy from high-risk applicants. However, determining an optimal model remains a challenge. To address this, in the current study, we explore an alternative approach for model benchmarking. The main concept involves the usage of a pipeline that constructs different classifiers for loan prediction and compares their performance across several evaluation metrics. To achieve this goal, we deploy an approach based on a multivariate statistical method, known as Archetypal Analysis (AA). The proposed methodology is applied to four datasets with diverse structural characteristics. The findings demonstrate that advanced classifiers like Random Forests (RF) and Artificial Neural Networks (ANN), with oversampling, simple parameter tuning, and feature selection consistently outperform traditional classifiers across most evaluation criteria. In conclusion, the results showcase the ability of AA to intuitively identify the best and worst models for each unique scenario.

CCS CONCEPTS

• **Mathematics of computing**; • **Probability and statistics**; • **Multivariate statistics**; • **Probabilistic algorithms**; • **Computing methodologies**; • **Machine learning**; • **Social and professional topics**; • **Professional topics**; • **Computing and business**; • **Economic impact**;

KEYWORDS

Loan default prediction, Machine learning, Model benchmarking, Archetypal Analysis

ICACS 2023, October 19–21, 2023, Larissa, Greece

ACM ISBN 979-8-4007-0909-8/23/10.
https://doi.org/10.1145/3631908.3631911

ACM Reference Format:
Anastasios Antoniadis, Konstantinos Georgiou, Konstantinos Charmanas, Nikolaos Mittas, and Lefteris Angelis. 2023. Benchmarking Classifiers for Loan Default Prediction using Archetypal Analysis. In *the 7th International Conference on Algorithms, Computing and Systems (ICACS 2023), October 19–21, 2023, Larissa, Greece.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3631908.3631911

1 INTRODUCTION

Loan default prediction involves the use of statistical methods to distinguish creditworthy from risky borrowers [1]. In the past, loan applicant's creditworthiness was evaluated by employees. However, this process was characterized by subjectivity and inconsistency [2]. Consequently, financial institutions gradually turned into advanced techniques, leading to an abundance of techniques, classifiers, and evaluation criteria [2].

It is widely acknowledged, that no single approach can fully meet the needs of each financial institution [3], since machine learning models have their own advantages and limitations. To uncover the best practices for loan default predictions, numerous studies have conducted model benchmarking [4]. Despite the contribution of these studies, no effective guidelines have yet been proposed.

The limitations of these studies arise from oversights in their experimental design. Firstly, it is common to overlook the structural differences between financial institutions. Moreover, the plethora of the proposed techniques, classifiers and evaluation metrics adds complexity in selecting an optimal model. Lastly, a critical issue lies in the widespread use of aggregation methods for model evaluation.

Based on that, in this study we examine a commonly observed phenomenon, wherein competing models exhibit varying performance. As a result, the research question we pose revolves around the concept of model benchmarking. During this process, it is of primary interest to find a set of models that generally perform well across all criteria, using a trustworthy benchmarking procedure. Yet, valuable insights can also be gained from models that showcase poor performance [5].

To address the research question, we consider the use of a multivariate statistical method, known as Archetypal Analysis (AA) [6], that has the potential to provide the answers to the problem of benchmarking models, that may present varying degrees of efficiency across different evaluation metrics. To the best of our

knowledge, AA has not yet been applied in the field of loan default prediction for benchmarking classification models. For the effective application of AA, we follow a streamlined process where, initially, multiple datasets about loan prediction with different characteristics are selected. Then, we employ well-known machine learning techniques and classifiers while a wide range of evaluation metrics is used for deducing their performance. By completing these steps and subsequently applying AA, our objective is to illustrate an alternative method of selecting the optimal model on each specific case, based on the dataset characteristics.

Hence, the primary contribution of our paper, in relation to recent literature on loan prediction is that we offer a novel way of benchmarking classification models on this domain, using a multivariate statistical methodology. The advantage of the proposed methodology is that it relies on multiple evaluation criteria and can find the optimal (or worse) classifier on one or more metrics, thus providing a more comprehensive way of model comparison. In that sense, our paper differs from other known methodologies on loan prediction and data mining such as KDD, CRISP-DM or SEMMA as we do not create a classification model and analyze its evaluation but rather, an efficient way of selecting targeted models based on pure statistical outputs.

The remainder of this study is organized as follows. The next chapter offers an analytical view on the datasets, classifiers, and benchmarking work on loan default prediction. In Chapter 3, the applied methodology is described in detail. The results obtained from the experiments are presented in Chapter 4 while Chapter 5 provides conclusions and suggestions for potential future directions.

2 RELATED WORK

2.1 Datasets for Loan Prediction

Availability is a common challenge regarding the datasets employed for loan default prediction. This issue stems from data privacy laws that impose restrictions on data sharing [7]. As a result, public datasets had lower usage rates, with the majority of research conducted on the so called Australian and German datasets [8]. Additionally, most studies also rely on a single dataset for their analysis while the respective average was approximately around two datasets [9]. However, single dataset usage hampers the ability to generalize the findings.

2.2 Classifiers for Loan Prediction

A plethora of different classification models has been used for loan default prediction, with one of the most widely known being Linear Discriminant Analysis (LDA). LDA still remains a reliable technique used for predicting loan defaults [10]. Logistic Regression (LR) has also emerged as a popular alternative [11]. In particular. LR has been used as a multivariate model for credit risk assessment [12] and continues to be considered as one of the fundamental models used by financial institutions [13], due to its ease of implementation, and good performance [14]. Artificial Neural Networks (ANN) classifiers are also considered as one of the most popular classifiers in loan default prediction [10]. Over the years, several ANN models have been proposed [15]. Among them, feed-forward networks are the most commonly used due to their comprehensibility and intuitiveness. Finally, the various categories of ensembles, with their numerous variations, are utilized in loan prediction, especially Random Forests (RF), which are particularly effective [16].

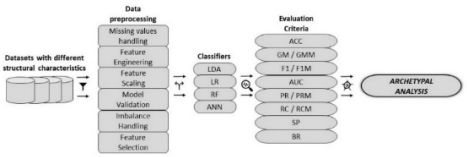

Figure 1: Methodology Schema

2.3 Benchmarking

Over the years, there has been a decrease in model benchmarking, which indicates that researchers seek to propose new classifiers [8]. However, given the abundance of the available methods, financial institutions are unable to choose the optimal model based on their needs. Although this difficulty has been thoroughly examined [2, 5, 25, 26], identifying an optimal model is hard to achieve and it depends on multiple factors [17].

To address these gaps, researchers in various fields had used the Archetypal Analysis (AA) algorithm [6]. For instance, AA had been proposed in economics [18] and software development [5]. Moreover, recent studies that estimate the effort required for software development, highlight the benefits of AA as an effective benchmarking method [19].

3 METHODOLOGY

The proposed methodology has been divided into separate steps, as shown in Figure 1. Our approach is divided into (i) data collection, (ii) data preprocessing, (iii) classification, (iv) evaluation and (v) benchmarking with AA.

For the execution of the experiments, the Python programming language was used in the Jupyter Notebooks environment. Additionally, for the implementation of the AA algorithm, the R programming language was used in the RStudio environment, with the primary library utilized being Archetypes [20]. The experiments were conducted on a laptop with an AMD Ryzen i7 processor operating at 2.30 GHz and 8 GB of memory.

3.1 Data Collection

In the first step, we collected four datasets with different structural characteristics, as described in Table 1. The datasets have been collected from Kaggle [24], which is a popular platform for sharing publicly available datasets. Therefore, all selected datasets are public, as access to private financial institution datasets was not feasible. A brief description of the selected datasets is provided below.

The Loan Prediction dataset pertains to a company's attempt to automate the loan approval process for applicants through online applications. The company provides a subset of the data for use. On the other hand, the Credit Risk Classification dataset focuses on banking products and consists of two datasets containing both customer transactions and demographic information. Additionally, the Credit Card Approval Prediction dataset involves the approval

Table 1: Dataset structure

Datasets	Observations	Default	Qualitative	Quantitative	Missing
Loan Prediction	614	31	8	5	YES
Credit Risk1	8250	16.81	2	21	YES
Credit Card Approval	25128	0.48	6	15	NO
Credit Risk2	32581	22	4	8	YES

or rejection of credit card applicants and lastly, the Credit Risk Dataset contains simulating credit bureau data.

3.2 Data Preprocessing

Data preprocessing is particularly important for improving the performance of the models. The following techniques were applied: 1) median imputation for quantitative features, 2) mode imputation for categorical features and 3) deletion of features with missing values exceeding 30%. During feature engineering, variables that contained categories or classes were merged or split when deemed useful. For the transformation of the variable values, label encoding, and one-hot encoding techniques were used to handle categorical variables. Next, the datasets were split into 70% for the training set and 30% for the test set. After the split, Standard Scaling was performed.

The primary objective in loan default prediction is to accurately predict the minority class, as misclassifying the minority class often leads to greater negative consequences [21]. In this study, for class imbalance handling, four different techniques are being employed based on the examined model. The default option was to ignore class imbalance or to manage it with oversampling (SMOTE or ADASYN) undersampling (Near Miss) and hybrid (SMOTE-ENN). Lastly, in the feature selection stage, the models could have no feature selection, or feature selection using the feature importance method of the RF classifier, with a setting of a 5% significance threshold [22].

3.3 Classifiers and Evaluation Criteria

The constructed models were based on four classifiers, namely LDA, LR, RF and ANN, with all selected classifiers used either with or without parameter tuning. Due to the extensive number of experiments, we apply Random Search CV algorithm to all classifiers. In the case of ANN, a wider set of parameters was constructed, and the Random Search algorithm from the keras_tuner library was used.

Regarding evaluation, an inspection of recent literature reveals that accuracy is among the most common evaluation metrics [2, 9]. However, accuracy can lead to misleading conclusions because classifiers tend to prioritize predicting the majority class. The Area Under the Curve (AUC) metric is equally important, particularly for imbalanced datasets [18], and it is one of the most popular metrics [8]. Additional effective metrics are Precision, F-measure and G-mean [8], while less commonly used metrics are the Brier Score, Kolmogorov-Smirnov Statistic, H-measure, and Gini index [8]. Hence, after careful consideration, we based the model evaluation on 8 different metrics.

We also make use of the macro scores (F1M, RCM, PRM, GMM) which represent the average of each class for each metric (loan – no loan), having a total of 12 evaluation metrics. As shown in Table 2, multiple combinations are created, resulting in a total of 80 models per dataset, for a total of 320 classification models that are subsequently benchmarked with the AA algorithm.

3.4 Archetypal Analysis (AA)

We base the application of AA on the representation of each constructed classifier. A detailed description of the algorithm can be found in the Supplementary Material of the results repository [23].

Under the assumption that each model is trained to predict observations of a dataset and is evaluated with a set of evaluation metrics, we represent the models as shown in Table 3. Hence, each model can be expressed as a multidimensional vector $M = \{Score1, Score2, \ldots, Scorem\}$ with the scores accounting for the model's performance across the defined evaluation metrics. In a multidimensional space, these vectors represent points and multiple points in a multidimensional space form a geometrical shape.

The AA algorithm is then employed in the shape of points in the multidimensional space to find the convex hull that surrounds the multidimensional vectors. The points of the convex hull are called archetypes and are located in the boundary of the multidimensional space. In the case of our study, the archetypes are constructed models with specific performance across the evaluation metrics and they may have efficient or poor performance in one or more evaluation metrics. Finally, the output of the AA algorithm is a matrix where each of the remaining models that are not archetypes is assigned weights, that portray how similar (or close) a model is to each of the detected archetypes. In Table 4, we illustrate the output of AA.

The weights represent the a-coefficients of the AA algorithm and, as they sum up to 1 for each model, they act as indicators of the resemblance of each model with each archetype. For example, in a solution with three archetypes and model coefficients 0.9, 0.08 and 0.02, we can deduce that the model is 90% similar to archetype 1, 8% similar to archetype 2 and 2% similar to archetype 3, and thus is closer to archetype 1 in terms of performance, whether its performance is poor or efficient.

The benchmarking process of this study is based on this type of interpretation, providing a multidimensional type of model evaluation which not only pinpoints models that excel in their predictions but also models that perform poorly, enabling the financial institutions to monitor the prediction process and choose the best models.

4 RESULTS

In this section, we present the results of applying the AA algorithm to the constructed classifiers. The presented models are named

Table 2: Combination of techniques and classifiers per dataset

Imbalance handling	Feature Selection	Classifiers	Tuning
SMOTE (SM), ADASYN (AD), NEAR MISS (NM), SMOTE-ENN (SME), No Handling (NIH)	Random forest feature selection (RFI) or No feature selection (NFS)	LDA, LR, RF, ANN	With tuning (TN) or No Tuning (DF)

Table 3: Model Representation (AA Input)

Model	Metric 1	Metric 2	...	Metric m
Model 1	Score11	Score12	...	Score1n
Model 2	Score21	Score22	...	Score2n
...	...	...	...	...
Model m	Score$_{\mathrm{m}}$1	Score$_m$2	...	Score$_{mn}$

Table 4: AA Output

Model	Archetype 1	Archetype 2	...	Archetype n
Model 1	*Weight*11	*Weight*12	...	*Weight*1n
Model 2	*Weight*21	*Weight*22	...	*Weight*2$_n$
...	...	...	...	...
Model m	*Weight*$_m$1	*Weight*$_m$2	...	*Weight*$_{\mathrm{mn}}$

following a classifier tuning imbalance features structure. For example, RF- df_nih_nfs represents a Random Forest classifier without tuning, without imbalance handling and without feature selection while ANN-tn_sm_rfi represents an Artificial Neural Network classifier, with parameter tuning, with SMOTE technique for imbalance handling, using feature selection.

To simplify the results of the proposed methodology, the analysis for the following two sections will solely focus on the Loan Prediction dataset. The entirety of the results for all datasets can be found in this Repository [23], along with Supplementary material, plots and insights. The presentation of results is based on evaluating the produced archetypes, finding relations between models and archetypes, and briefly analyzing the results of the archetypes in terms of model performance. We should emphasize that the archetypes detected by the AA algorithm do not necessarily represent models with optimal performance, but rather, models that present interesting characteristics and act as "extreme" or "noteworthy" cases. Of course, the AA algorithm is also capable of detecting models with optimal performance on multiple evaluation metrics.

4.1 Evaluating Archetypes

In Table 5, the models identified by AA as archetypes for the Loan Prediction dataset are presented along with their performance across the selected evaluation metrics. We should point out that the AA algorithm automatically detects the optimal number of archetypes to be extracted, so as to minimize error and optimize the performance of the algorithm.

As observed, the six detected archetypes showcase different characteristics, performance in each evaluation metric and overall performance. More specifically, AR5 (ANN-tn_ad_nfs) and AR6 (ANN- tn_nih_nfs) are particularly ineffective in most evaluation metrics, as they represent ANN classifiers without feature selection and without imbalance handling, in the case of AR6. Hence, these models can be considered as baseline classifiers that do not adapt to the characteristics of this specific dataset and perform poorly, based on the proposed benchmarking process. AR3 (ANN-df_nm_nfs) is moderately effective in most evaluation metrics, representing a model that performs undersampling with NM but does not use feature selection. Finally, AR1 (ANN-df_ad_rfi), AR2 (RF-tn_nm_nfs) and AR4 (LDA-df_nih_rfi) exhibit high effectiveness in most criteria, with AR4 and AR1 being the optimal models. This indicates that selecting the most appropriate features, with the RF feature importance method, is crucial for the performance of a model while RF retains its position as a classification model that has consistently acceptable performance. AR1 appears to be the most effective in most of the evaluation criteria compared to the other archetypes. However, each evaluation metric is assessed differently based on the needs of each financial institution. Thus, a model can be considered optimal, if it consists of a desirable mixture of archetypes.

4.2 Relations between Remaining Models and Archetypes

Another part of the study that can be extracted are the a-coefficients generated by the AA algorithm, to explore the similarity of the constructed models with the archetypes. To explore the similarity of models based on the a-coefficients, the common practice is to set a threshold. Based on that, when the a-coefficient of a specific model exceeds this threshold, it is considered as a neighbor to the archetype, meaning that its overall performance closely resembles that of the archetype.

The selection of such a threshold can be arbitrary, but we define that if the a-coefficient is above 70%, it indicates a strong relationship between the model and the archetype [5]. Models that do not exceed this threshold do not appear in the results as they cannot be associated with any archetype. Additionally, models that exhibit identical performance across evaluation metrics are merged, thus reducing the number of models presented. In Table 6, the indicative performance of some models that passed the 70% threshold is showcased, while Table 7 displays the a-coefficients of the models with respect to the archetypes, pinpointing the archetype that is closer to each respective model. The full tables can be found in the Supplementary Material [23]

Based on Table 6, the benchmarking process indicates a well-rounded matching between models and archetypes, as in the majority of the models, the archetype to which they are closest to can be easily identified. This proves that AA can indeed be a potent

Table 5: Archetypes Performance

#	ACC	PR	PRM	RC	RCM	F1	F1M	AUC	SP	GM	GMM	BR
AR1	0.81	0.72	0.78	0.63	0.76	0.67	0.77	0.76	0.89	0.75	0.76	0.19
AR2	0.64	0.45	0.66	0.79	0.68	0.58	0.63	0.68	0.57	0.67	0.68	0.36
AR3	0.6	0.38	0.56	0.49	0.57	0.43	0.56	0.57	0.65	0.56	0.57	0.4
AR4	0.82	0.9	0.85	0.46	0.72	0.6	0.74	0.72	0.98	0.67	0.72	0.18
AR5	0.31	0.31	0.15	1	0.5	0.47	0.24	0.5	0	0	0.5	0.69
AR6	0.69	0	0.35	0	0.5	0	0.41	0.5	1	0	0.5	0.31

Table 6: Indicative Model Performance

	ACC	*PR*	*PRM*	*RC*	*RCM*	*F1*	*F1M*	*AUC*	*SP*	*GM*	*GMM*	*BR*
RF-tn_nih_rfi	0.79	0.68	0.75	0.6	0.74	0.64	0.74	0.74	0.87	0.72	0.74	0.21
ANN-df_ad_rfi	0.81	0.72	0.78	0.63	0.76	0.67	0.77	0.76	0.89	0.75	0.76	0.19
ANN-df_nm_nfs	0.6	0.38	0.56	0.49	0.57	0.43	0.56	0.57	0.65	0.56	0.57	0.4
LDA-df_nih_nfs	0.81	0.84	0.82	0.47	0.72	0.61	0.74	0.72	0.96	0.67	0.72	0.19
LR-tn_nih_rfi	0.8	0.84	0.82	0.46	0.71	0.59	0.73	0.71	0.96	0.66	0.71	0.2
LDA-tn_nih_rfi	0.81	0.87	0.83	0.46	0.71	0.6	0.74	0.71	0.97	0.66	0.71	0.19
ANN-tn_ad_nfs	0.31	0.31	0.15	1	0.5	0.47	0.24	0.5	0	0	0.5	0.69
ANN-tn_nih_nfs	0.69	0	0.35	0	0.5	0	0.41	0.5	1	0	0.5	0.31

Table 7: Indicative A-coefficients of models to archetypes

	AR1	*AR2*	*AR3*	*AR4*	*AR5*	*AR6*
RF-tn_nih_rfi	0.9	0	0.07	0.007	0.005	0.017
ANN-df_ad_rfi	1	0	0	0	0	0
ANN-df_nm_nfs	0	0.002	0.995	0	0.002	0
LDA-df_nih_nfs	0.173	0	0.006	0.807	0	0.014
LR-tn_nih_rfi	0.063	0	0.05	0.876	0	0.011
LDA-tn_nih_rfi	0	0	0.031	0.968	0	0.001
ANN-tn_ad_nfs	0	0	0	0	1.00	0
ANN-tn_nih_nfs	0	0	0	0	0	1

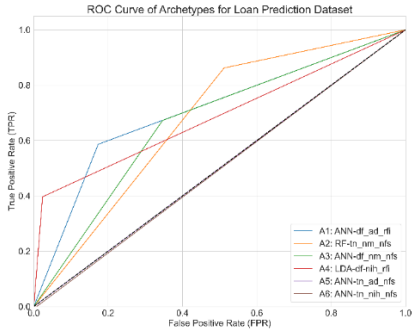

Figure 2: ROC-AUC curves of archetypes

indicator of model performance, extracting different models across different evaluation criteria.

For example, models ANN-tn_ad_nfs and ANN_tn_nih_nfs are identical to AR5 and AR6, respectively, which were the worst performing archetypes overall. On the other hand, the ANN-df_nm_nfs model closely resembles AR3 and can be considered a moderate model, with moderate ACC, PR and SP. The models RF- tn_nih_rfi and ANN-df_ad_rfi are 90% and 100% similar to AR1 and are highly effective in all metrics. Furthermore, models like LDA-tn_nih_rfi and LR-tn_nih_rfi closely resemble AR 4, which is highly effective in all metrics except RC. Lastly, it can be observed that model LDA-df_nih_nfs is 80.7% similar to AR4 and 17.3% similar to AR1, making it a combination of archetypes that are particularly effective, which is an encouraging sign for a model that is adaptable and conforms to the characteristics of the specific dataset.

In a broader context, with the help of a-coefficients, we can easily discern the best and worst models, based on their evaluation in one or more metrics. However, it is important to note that most

models are not optimal in all evaluation metrics. For instance, models similar to AR4, have a very low RC but perform well in the remaining metrics. This is made abundantly clear when examining the ROC-AUC curves of the archetypes, presented in Figure 2 where different performances per archetype are detected. Hence, the financial institution of this dataset should carefully consider if this model suits their purposes or if it would be better to select a model belonging to a different archetype that may present better RC performance. (e.g. AR1).

5 CONCLUSIONS AND FUTURE DIRECTIONS

The aim of this study is to present a benchmarking method that detects classifiers with varying performance. Generally, results may vary significantly, with no golden standard, as the AA solution can differ based on the utilized dataset with effective and ineffective models used in drawing conclusions. Nevertheless, the proposed solution is a valid way of determining the performance of multiple models. Despite the challenges, the application of AA provides rich results. For example, results indicate that the ANN classifier with complex parameter tuning often emerges as a poor model. This leads us to the conclusion that the ANN algorithm is quite challenging to adequately tune, requiring considerable time and expertise.

Additionally, RF and ANN, showcase better performance compared to the traditional LDA and LR classifiers though LDA and LR have a solid performance even without parameter tuning and class imbalance treatment. In class imbalance, the SMOTE and ADASYN, are particularly effective while Near Miss and SMOTE-ENN performed poorly in most model combinations. Additionally, the ANN classifier, appears to perform ineffectively in combinations where oversampling techniques are not used. Moreover, feature selection generally leads to more effective results than selecting all variables.

Regarding metrics, ANN models with complex parameter tuning, or models with the NM technique usually have a high Brier Score. Moreover, models as RF with tuning, oversampling and feature selection, and ANN with simple parameter tuning and oversampling excel in the majority of the evaluation metrics.

One limitation of our work is the selection of only public datasets, as the use of private datasets was not possible due to restricted access. However, the AA benchmarking process can potentially be used in any dataset, as long as the proper preprocessing is employed. In addition, no outlier handling was performed, which can influence AA results. Finally, AA needs to be complemented by other methods for a better interpretation of the results. As this work serves as an introductory study in loan default prediction using AA, several future work directions are the employment of private datasets and the use of additional methods and visualizations that support AA. In addition, we plan to further investigate outlier treatment and its influence on the AA benchmarking as well as experimenting with different techniques and classifiers.

REFERENCES

[1] M. K. Lim and S. Y. Sohn, "Cluster-based dynamic scoring model," Expert Syst Appl, vol. 32, no. 2, pp. 427–431, Feb. 2007, doi: 10.1016/j.eswa.2005.12.006.

[2] H. A. Abdou and J. Pointon, "CREDIT SCORING, STATISTICAL TECHNIQUES AND EVALUATION CRITERIA: A REVIEW OF THE LITERATURE," Intelligent Systems in Accounting, Finance and Management, vol. 18, no. 2–3, pp. 59–88, Apr. 2011, doi: 10.1002/isaf.325.

[3] J. Breeden, "A survey of machine learning in credit risk," The Journal of Credit Risk, 2021, doi: 10.21314/JCR.2021.008.

[4] D. J. Hand and W. E. Henley, "Statistical classification methods in consumer credit scoring: A review," J R Stat Soc Ser A Stat Soc, vol. 160, no. 3, pp. 523–541, 1997, doi: 10.1111/j.1467-985X.1997.00078.x.

[5] N. Mittas, V. Karpenisi, and L. Angelis, "Benchmarking effort estimation models using Archetypal Analysis," in ACM International Conference Proceeding Series, Association for Computing Machinery, 2014, pp. 62–71. doi: 10.1145/2639490.2639502.

[6] A. Cutler and L. Breiman, "Archetypal analysis," 1994. doi: 10.1080/00401706.1994.10485840.

[7] G. Shingi, "A federated learning based approach for loan defaults prediction," in IEEE International Conference on Data Mining Workshops, ICDMW, IEEE Computer Society, Nov. 2020, pp. 362–368. doi: 10.1109/ICDMW51313.2020.00057.

[8] F. Louzada, A. Ara, and G. B. Fernandes, "Classification methods applied to credit scoring: Systematic review and overall comparison," Surveys in Operations Research and Management Science, vol. 21, no. 2. Elsevier Science B.V., pp. 117–134, Dec. 01, 2016. doi: 10.1016/j.sorms.2016.10.001.

[9] S. Lessmann, B. Baesens, H. V. Seow, and L. C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," Eur J Oper Res, vol. 247, no. 1, pp. 124–136, Nov. 2015, doi: 10.1016/j.ejor.2015.05.030.

[10] V. S. Desai, J. N. Crook, and G. A. Overstreet, "A comparison of neural networks and linear scoring models in the credit union environment," Eur J Oper Res, vol. 95, no. 1, pp. 24–37, 1996, doi: 10.1016/0377-2217(95)00246-4.

[11] J. Berkson, "Application of the Logistic Function to Bio-Assay," J Am Stat Assoc, vol. 39, no. 227, pp. 357–365, Sep. 1944, doi: 10.1080/01621459.1944.10500699.

[12] J. A. Ohlson, "Financial Ratios and the Probabilistic Prediction of Bankruptcy," 1980.

[13] B. Baesens, T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, and J. Vanthienen, "Benchmarking state-of-the-art classification algorithms for credit scoring," Journal of the Operational Research Society, vol. 54, no. 6, pp. 627–635, 2003, doi: 10.1057/palgrave.jors.2601545.

[14] T. Van Gestel, B. Baesens, P. Van Dijcke, J. Suykens, and J. Garcia, "Linear and non-linear credit scoring by combining logistic regression and support vector machines," The Journal of Credit Risk, vol. 1, no. 4, pp. 31–60, 2005, doi: 10.21314/jcr.2005.025.

[15] L. Munkhdalai, T. Munkhdalai, O. E. Namsrai, J. Y. Lee, and K. H. Ryu, "An empirical comparison of machine-learning methods on bank client credit assessments," Sustainability (Switzerland), vol. 11, no. 3, Jan. 2019, doi: 10.3390/su11030699.

[16] F. Barboza, H. Kimura, and E. Altman, "Machine learning models and bankruptcy prediction," Expert Syst Appl, vol. 83, pp. 405–417, Oct. 2017, doi: 10.1016/j.eswa.2017.04.006.

[17] D. J. Hand, "Assessing the Performance of Classification Methods," International Statistical Review, vol. 80, no. 3, pp. 400–414, Dec. 2012, doi: 10.1111/j.1751-5823.2012.00183.x.

[18] G. C. Porzio, G. Ragozini, and D. Vistocco, "On the use of archetypes as benchmarks," Appl Stoch Models Bus Ind, vol. 24, no. 5, pp. 419–437, Sep. 2008, doi: 10.1002/asmb.727.

[19] N. Mittas and L. Angelis, "Data-driven benchmarking in software development effort estimation: The few define the bulk," Journal of Software: Evolution and Process, vol. 32, no. 9, Sep. 2020, doi: 10.1002/smr.2258.

[20] M. J. A. Eugster and F. Leisch, "From Spider-Man to Hero – Archetypal Analysis in R." [Online]. Available: http://CRAN.R- project.org/package=archetypes.

[21] Phua C, Alahakoon D, and Lee V, "Minority Report in Fraud Detection: Classification of Skewed Data," SIGKDD Explorations Newsletter 6(1):50–59, 2004.

[22] R. Genuer, J. M. Poggi, and C. Tuleau-Malot, "Variable selection using random forests," Pattern Recognit Lett, vol. 31, no. 14, pp. 2225–2236, Oct. 2010, doi: 10.1016/j.patrec.2010.03.014.

[23] https://drive.google.com/drive/u/0/folders/1qWNToMTCWcw45vnZo8dXnR8O-s2TM4oC (accessed Aug. 15, 2023).

[24] https://www.kaggle.com/ (accessed Aug. 15, 2023).

[25] Tsungnan Chou and Mingmin Lo, "Predicting Credit Card Defaults with Deep Learning and Other Machine Learning Models," International Journal of Computer Theory and Engineering vol. 10, no. 4, pp. 105-110, 2018.

[26] Wen Tian, Ying Zhang, Yinfeng Li, and Huili Zhang, "Probabilistic Demand Prediction Model for En-Route Sector," International Journal of Computer Theory and Engineering vol. 8, no. 6, pp. 495-499, 2016.