# Explainable Model of Credit Risk Assessment Based on Convolutional Neural Networks

Carlos Cardenas-Ruiz(✉), Andres Mendez-Vazquez, and Luis M. Ramirez-Solis

Department of Computer Science, CINVESTAV - Guadalajara, Zapopan, Mexico
carlos.cardenasruiz@cinvestav.mx

**Abstract.** Credit Risk Assessment estimates the probability of loss due to a borrower's failure to repay a loan or credit. Therefore, one of the principal challenges of financial institutions is to lower the losses generated by leading financial resources to possible default clients. Current models for Credit Risk Assessment used by the industry are based on Logistic regression (LR), thanks to their operational efficiency and interpretability. However, Deep Learning (DL) Algorithms have become more attractive than conventional Machine Learning due to their best general accuracy. However, Models for Credit Risk Assessment based on DL have a problem because their complexity makes them difficult for humans to interpret. Additionally, international regulations for financial institutions require that models be interpretable. In this work, we propose the use of a model based on Convolutional Neural Networks (CNN) and SHapley Additive exPlanations (SHAP) to generate a more accurate and explainable model than LR models. In order to demonstrate its efficacy, we use four datasets commonly used to benchmark classification algorithms for credit scoring. The results show that the method proposed is more accurate than LR for large datasets (more than 5900 samples), with an improvement in accuracy up to 12.3%.

**Keywords:** Credit risk assessment · DeepInsight · Explainability · CNN

## 1 Introduction

Even though Logistic Regression (LR) is one of the most common algorithms used in the financial industry [4], different studies have demonstrated that it is not the most accurate estimation for credit risk classification. A reference to this is two benchmarking studies published by Baesen [2,3] that demonstrate that for the category of individual classifier, Deep Learners are more accurate than LR.

Despite better accuracy of Deep Learners, Financial Institutions use LR for credit scoring due to their Operational efficiency (simplicity), and Interpretability (transparency) in predictions [11]. These two points, and statistical accuracy

O. Pichardo Lagunas et al. (Eds.): MICAI 2022, LNAI 13612, pp. 83–96, 2022.
https://doi.org/10.1007/978-3-031-19493-1_7

form part of the five key characteristics of a successful credit risk mode defined for Baesen [4], shown in Table 1. However, due to the complex nature of Deep Learners, they are considered as Black Box Model(BBM), which refers to complex models that are not straightforwardly interpretable by humans [25], making them unviable for the use of financial institutions due to International regulations [7]. However, the application of interpretability methods permits us to give transparency to DL models.

In this work, we propose an explainable Deep Learning model based on a 2D Convolutional Neural Network (CNN) for credit risk classification. The use of CNN for credit risk is not new. However, Our approach uses DeepInsight [31], a methodology proposed by Alok Sharma et al., to transform tabular data into a 2D representation as input for CNN. The classification accuracy of the DeepInsight combined with CNN showed a better performance than Decision Tree (DT), LR, and RF for large datasets (more than 5900 samples), as shown in the results of our paper. Additionally, the use of SHapley Additive exPlanations (SHAP) to explain the model's prediction gives us an explainable and more accurate model than the LR model for credit risk classification.

**Table 1.** Key characteristics of successful credit risk model [4]

| Characteristic | Description |
|---|---|
| Statistical accuracy | Refers to the power of prediction of a model to generalize well and avoid overfitting to the historical data. |
| Interpretability | A model needs to be interpretable. In other words, the model gives enough information to an expert to understand why the model takes those decisions. |
| Operational efficiency | This point refers to the time needed for the model to evaluate whether a customer is a defaulter. Operational efficiency also includes the following tasks:<br>- Collection and preprocessing of data<br>- Evaluation of the model<br>- monitor and back-test<br>- Reestimate |
| Economical cost | Developing and implementing models have a cost for the organizations. The total cost of all tasks and resources needed is the economical cost of the model. The cost of tasks mentioned in the operational efficiency plus cost of resources like:<br>- Software cost<br>- Human and Computing resources |
| Regulatory compliance | A model construction needs to comply with all applicable regulations and legislation. Basel, for example, specifies what information can or cannot be used. Other regulations like privacy and/or discrimination should be respected |

The rest of the paper is organized as follows. In Sect. 2 we present the motivation for this work, while Sect. 3 discusses the state of the art of Credit risk assessment. In Sect. 4 we introduce the proposed method and in Sect. 5 we provide details about the implementation and experiments made in order to obtain the best model, as well as the comparison with previous models. Finally, in Sect. 6 we summarize our work and discuss some perspectives for future work.

## 2 Motivation

Why is credit risk so important? First, it is a matter of common knowledge that any economy, no matter how advanced, cannot develop in the absence of credit [5]. On other hand, a relaxed credit policy can become the core of a global financial crisis like 2007–2009.

The credit cycle begins with credit being easily accessible to customers, and they can borrow and spend more. In the same way, enterprises can borrow and make more significant investments. More consumption and investment create jobs and lead to income and profit growth. However, all economic expansion induced by credit ends when critical economic sectors become incapable of paying off their debts [17]. When the credit cycle is broken, there is a strong possibility of crisis (Fig. 1).

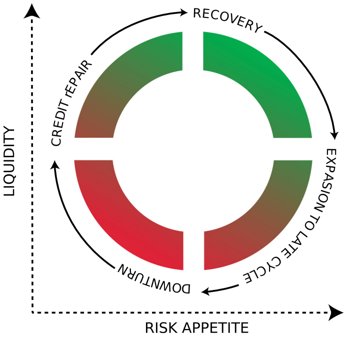

**Fig. 1.** Credit cycle [15]

The production of accurate credit risk tools for financial institutions allows them to make better decisions about granting credits. A reasonable administration of credit is an essential part of the growth of almost all economies. Economic growth is the most powerful instrument for reducing poverty and improving the quality of people's life. Growth can generate virtuous circles of prosperity and opportunity [12]. In conclusion, the research of credit risk topics profoundly impacts the world and people's lives.

## 3 Related Work

Durand [13] gave the bases for statistical credit risk scoring about 80 years ago. Nowadays, thanks to the evolution of statistical classification techniques, computational power, and easy access to sizable and reliable data, financial institutions use the statistical approach for credit risk management [4]. Many different classification models have been developed to address the credit scoring problem during the past few decades. Logistic Regression [6] and Random Forest [4] are the most widely-used model for credit scoring. However, more sophisticated machine learning techniques like Support Vector Machine (SVM) and Artificial Neural Networks (ANN) are also widely applied to credit scoring. Furthermore, ensemble methods that combine the advantage of various single classifiers get good scores like HCES-Bag with the best score in benchmark scoring published by Lessmann and Baesens [3].

Different empirical studies have compared the performance of different classification models for credit scoring. For example, West [33] compares ANN against traditional machine learning techniques. The result showed that ANN has better performance than LR. On the other hand, Cuicui et al. designed a Deep Belief Network (DBN) for credit classification and compared it against SVM, LR, and Multilayer Perceptron on the credit default swaps dataset [23]. The result showed that DBN yields the best performance.

Convolutional Neural Network (CNN) is a representative technique in DL; it first appeared in the work of Yann Lecun et al., designed to handle the variability of data in 2D shape [19]. The impressive achievements of CNN in different areas, including but not limited to Natural Language Processing (NLP) and Computer Vision, attract the attention of industry and academia [21]. Moreover, in the last few years, attracted by the classification ability of CNN, some studies have begun to apply CNN to managing credit risk. Bing Zhu, Wenchuan Yang, and Huaxuan Wang propose a model named "Relief-CCN" [37] that combines CNN and Relief algorithm. The results demonstrate a better performance against LR and RF with the dataset from a Chinese consumer finance company. On the other hand, Xolani Dastale and Turga Gelik [10] propose another CNN model for credit scoring getting better performance than traditional Machine learning methods. However, the last two models change the tabular data into a 2D representation for the CNN input discretizing the data and generating a representation with only ones or zeros in the values with possible data loss.

## 4 Proposed Approach

Credit risk classification is a data mining problem, thinking about this, we propose a process based on CRoss-Industry Standard Process for Data Mining (CRISP-DM) which is a process model for data mining [9]. Our proposal is a modification in the last step, called implementation on CRISP-DM, and changed by Interpretability where the generation of local and global explanations are generated, as shown in Fig. 2.

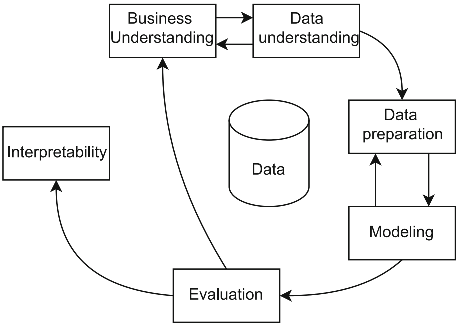

**Fig. 2.** Proposed model phases based on CRISP-DM

However, all the steps of CRISP-DM are essential, we will focus on the steps of Data preparation (especially on the task of Format data), Modeling, and an extra step called Interpretability defined for us after the evaluation step where we make the explanations of the generated mode.

### 4.1 Data Preparation

Format data [9] is part of the data preparation phase, which refers to all activities needed to transform the initial raw data into the data used as input for Machine Learning algorithms. The data used for financial institutions are generally in tabular [4] form (data displayed in columns or tables). However, The proposed 2D CCN requires image data representation.

DeepInsight [31] transforms the tabular data with a sequence of steps. First, it generates a feature vector transposing the dataset. Second, it maps each feature into a 2D space using t-SNE. Third, for efficiency, DeepInsight finds the small rectangle area that covers all points to be horizontal framed. Fourth, based on the dimensions of the final image defined for the user DeepInsight make a process of framing and mapping each feature. Finally, each instance is represented using the general feature image generated in the last step modifying the values of each

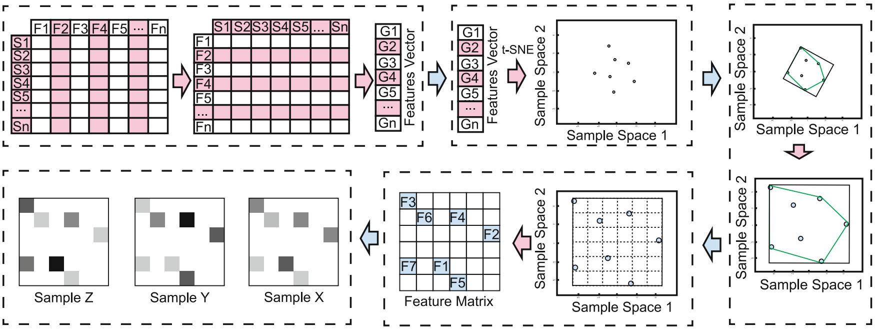

**Fig. 3.** Illustration of the DeepInsight [31] methodology to transform a feature vector to image pixels.

feature for the normalized value of the instance features values, which can be seen as a greyscale image; the Fig. 3 shows the transformation process.

## 4.2 Modeling

Convolutional Neural Network is one of the most used deep learning architectures for image processing [36]. The basic structure of CNN is shown in Fig. 4. There are two particular types of layers in CNN called the convolutional layer and the pooling layer. The convolutional layer is the basic building block of CNN [26]. It contains a set of learnable filters that slide over the image to extract features. The pooling layer reduces the spatial size representation and the number of parameters giving more efficiency and control overfitting.

Convolutional neural networks differ from traditional neural networks by replacing general matrix multiplication with convolutional, reducing the weights in the network, and allowing the import of an image directly. Additionally, The convolution layer has several main features, two of which are local perception and parameter sharing. Local perception refers to the high relevance of image parts that are close compared to the low relevance of the distant parts [35]. On the other hand, parameter sharing learn one set of parameters throughout the whole process instead of learning different parameter sets at each location [37]. These features help to improve the efficiency of the network.

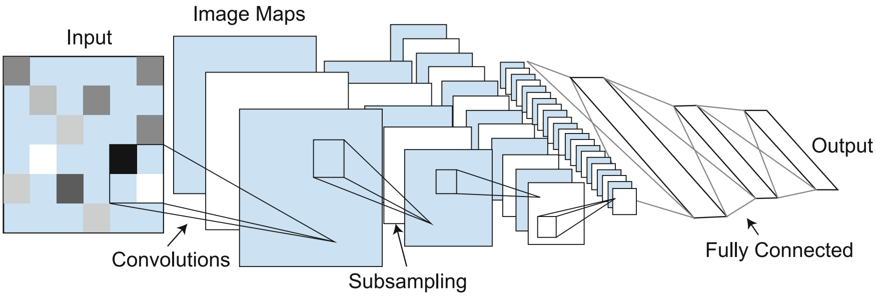

**Fig. 4.** The basic structure of CNN.

## 4.3 Interpretability

Miller defines interpretability as the degree to which a human can understand the cause of a decision [24]. The interpretability of a Machine Learning model is inversely related to its complexity. CCN is considered a Black Box Model (BBM) that refers to complex models that are not straightforwardly interpretable to humans [27]. However, different methods exist to explain BBM, like SHapley Additive exPlanations (SHAP) [22] used in this paper to generate the local and global explanations.

SHAP was proposed by Lundberg and Lee is a unified approach to interpreting model predictions. SHAP is a method to explain individual predictions based on the calculation of Shapley Values [22]. However, SHAP can give us a global explanation of a model based on the average of absolute Shapley values per feature of a random subset of dataset samples.

Shapley Values (SV) proposed by Shapley is a method based on coalitional game theory (or cooperative game theory) [30]. SV explains a prediction assuming that each feature value of a sample is a "player" in a game where the prediction is the goal. In other words, SV is the average marginal contribution of a feature value across all possible coalitions.

A linear model prediction is an explainable model because we can see how a feature affects the prediction.

$$\hat{f}\left(x\right) = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$$

where $x$ is the instance that we want to calculate the contributions. Each $x_j$ is a feature value, with $j = 1, \ldots, p$. $\beta_j$ is the weight of feature j.

The contribution $\phi_j$ of the j-th feature on the prediction $\hat{f}$ is:

$$\phi_j\left(\hat{f}\right) = \beta_j x_j - E\left(\beta_j X_j\right) = \beta_j x_j - \beta_j E\left(X_j\right)$$

The mean effect of feature $j$ is $E\left(\beta_j X_j\right)$, and the contribution of j-th feature is the difference between the feature effect minus the average effect. If we sum all the contributions, we get the following result:

$$\begin{aligned} \sum_{j=1}^{p} \phi_j\left(\hat{f}\right) &= \sum_{j=1}^{p} \beta_j x_j - E\left(\beta_j X_j\right) \\ &= \left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right) - \left(\beta_0 + \sum_{j=1}^{p} E\left(\beta_j X_j\right)\right) \\ &= \hat{f}\left(x\right) - E\left(\hat{f}\left(X\right)\right) \end{aligned}$$

The result is the predicted value for the instance x minus the average predicted value. To do the same to different models other than linear is necessary to compute feature contribution for a single prediction.

To get the Shapley value of a feature value, we need to calculate the contribution made to the result, weighted and summed over all combinations [25]. Then, the Shapley value is defined via a value function for players contained in set S:

$$\phi_j\left(val\right) = \sum_{S \subseteq \{x_1,\dots,x_p\} \backslash \{x_j\}} \frac{|S|!\left(p-|S|-1\right)!}{p!}\left(val\left(S \cup \{x_j\}\right) - val\left(S\right)\right) \tag{1}$$

where $S$ is a subset of model features, $p$ is the number of features and $x$ is the vector of feature values. The result is the contribution of feature $j$ for all feature coalitions.

The exact solution to get the Shapley value requires the evaluation of all coalitions of feature values with and without the $j-th$ feature. However, the exact solution becomes problematic for more than a few features because the number of possible coalitions exponentially increases with each added feature. Therefore, Strumbelj et al. (2014) [32] propose an approximation with the use of Monte-Carlo sampling:

$$\hat{\phi_j} = \frac{1}{M}\sum_{m=1}^{M}\left(\hat{f}\left(x_{+j}^{\mathrm{m}}\right) - \hat{f}\left(x_{-j}^{\mathrm{m}}\right)\right)$$

where $\hat{f}\left(x_{+j}^{m}\right)$ is the prediction for the instance $x$, with a random number of feature values taken from another instance z taken a random, except for the value of feature $j$, the $\hat{f}\left(x_{-j}^{m}\right)$ is equal to $\hat{f}\left(x_{+j}^{m}\right)$, with the difference that the value of j feature is taken from z. The procedure to approximate the Shapley value is explained next:

**Algorithm 1** Shapley value approximation

```
Input: Instance of interest x, data matrix X, feature index j, ML model f, and the
  number of iterations M.
Output: Shapley value of j-th feature
  for 1 to M do
      Get random instance z from matrix X
      Generate a random permutation o of features
      Order the instance x: $x_o = \left(x_{(1)}, ..., x_{(j)}, ..., x_p\right)$
      Order the instance z: $z_o = \left(z_{(1)}, ..., z_{(j)}, ..., z_p\right)$
      Generate two new instances
            $x_{+j} = \left(x_{(1)}, ..., x_{(j-1)}, x_{(j)}, z_{(j+1)}, ..., z_{(p)}\right)$
            $x_{-j} = \left(x_{(1)}, ..., x_{(j-1)}, z_{(j)}, z_{(j+1)}, ..., z_{(p)}\right)$
      Compute MC: $\phi_j^m = \hat{f}\left(x_{+j}\right) - \hat{f}\left(x_{-j}\right)$
  end for
  return the Shapley value as the average: $\phi_j\left(x\right) = \frac{1}{M}\sum_{m=1}^{M}\phi_j^m$
```

SHAP uses Shapley values to make explanations of BBM. However, SHAP proposes different kernel-based estimation approaches for Shapley values inspired by local surrogate models. KernelSHAP [22] is a model-agnostic based on LIME and Shapley values. On the other hand, TreeSHAP and DeepSHAP are model-specific, the first is an efficient estimation approach for tree-based models and the second for Deep learning models.

SHAP Feature Importance (FI) is one of the global interpretations based on aggregations of Shapley values. SHAP FI order features importance based on the absolute value of Shapley values per feature across the data [25]:

$$I_j = \frac{1}{n}\sum_{i=1}^{n}\left|\phi_j^{(i)}\right|$$

After, SHAP order features by decreasing importance. For example, Fig. 5 shows the SHAP FI for a pre-trained CNN for the lending club dataset.

After training the CNN models and judging their performances, we use SHAP to generate local and global explanations of the model. For example, SHAP can generate local explanations of tabular data and images but not the same for the global explanation of images. Additionally, explanations of images are given for the SHAP values of pixels based on the predictions of the trained model, which is not easy to understand for humans an example is shown in Fig. 6. Nevertheless, thanks to DeepInsight, we have the mapping between each pixel and feature. It allows us to return SHAP values to tabular form that generates more interpretable local and global explanations (Fig. 7).

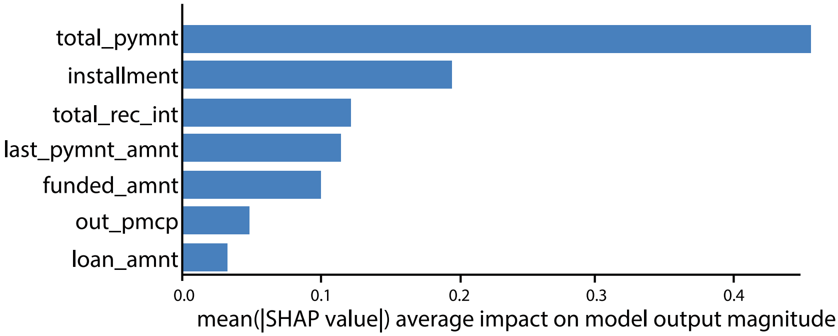

**Fig. 5.** SHAP Global explanation of first eight important features measured as the mean absolute Shapley values for Lending club dataset

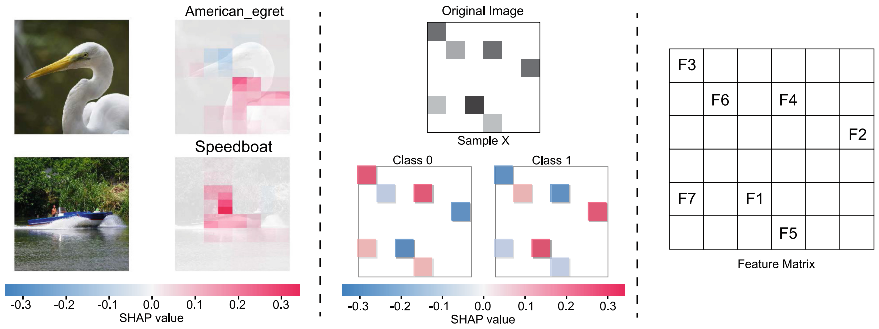

**Fig. 6. (left)** Example of SHAP local explanation for a Multi-class ResNet50 on ImageNet [29]. **(middle)** Example of SHAP local explanation of credit image generated from tabular data transformed using DeepInsight. **(right)** Example of feature matrix generated using DeepInsight.

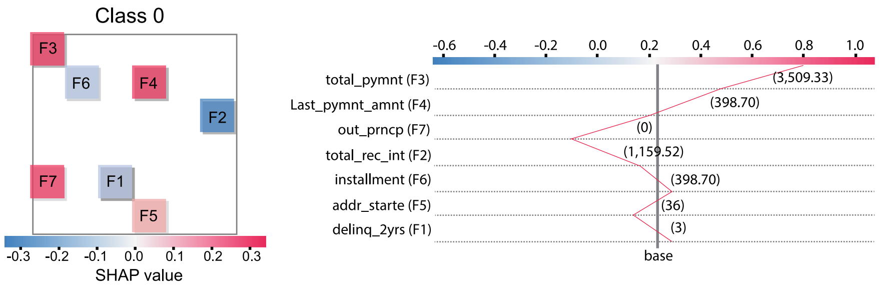

**Fig. 7. (left)** Example of SHAP local explanation for DeepInsight credit image. **(right)** Example of SHAP local explanation for DeepInsight credit image after conversion to tabular form.

## 5 Experimental Results

The datasets used in this thesis is about four datasets provided for financial and academic institutions widely used in research of credit scoring. All the datasets are different in almost all their characteristics like the number of samples and features. A resume of the characteristics of the dataset are shown in Table 2:

**Table 2.** Datasets

| | Datasets | | | |
|---|---|---|---|---|
| | Australian [28] | German [18] | HMEQ [4] | LC 2007–2014 [20] |
| # Samples | 690 | 1,000 | 5,960 | 466,285 |
| # non default | 307 | 700 | 4,771 | 415,317 |
| # default | 383 | 300 | 1,189 | 50,968 |
| # features | 14 | 20 | 13 | 75 |
| Categorical | 8 | 13 | 2 | 22 |
| Continuous | 6 | 7 | 11 | 53 |

Since the data contained in the four datasets may contain redundant features that can increase computation and affect the performance, for numerical and categorical features, we use ANOVA and Chi-Squared, respectively [8]. Additionally, when features with high correlation exist, SHAP generates redundant local and global explanations with features with the same SHAP values. Then eliminating high correlated features is need it.

For each dataset, we use cross-validation with ten stratified folds. The training set was used to find the optimal parameter of the CNN model. The metrics of the test set were used to assess the performance of the CNN model. Table 3 shows the optimal architecture of our CNN model for Australian, German, HMEQ, and Lending club datasets, respectively.

To compare the performance of our model, we use different studies that use the same datasets. Additionally, we train base models with LR and RF to compare the results. For each dataset, we calculate Accuracy and Area Under the Receiver Operating Characteristics (AUROC). Accuracy is not the best metric for the evaluation of models of credit risk classification. However, many studies only use Accuracy for evaluation. Therefore, Table 4 compares the Accuracy against each model and dataset. On the other hand, a better metric than Accuracy for credit risk classification is AUROC; in Table 5, we compare the AUROC for the different studies containing this information and our results.

**Table 3. CNN Architecture**: Parameters and architectures of the CNN model in **(a)** Australian dataset epochs = 500, **(b)** German dataset epochs = 50, **(c)** HMEQ dataset epochs = 1000, and **(d)** Lending club dataset epochs = 50. Legend: Conv (Convolutional Layer), PL (Pooling Layer), FC (Fully Connected Layer).

| Layer | Parameters and Architecture |
|---|---|
| Input | Input shape: k=8, d=8, c=1 |
| Conv1 | Fil: 64, Kernel: 3x3, Str=1,Pad=1 |
| PL1 | type: max, size 2x2 |
| Conv2 | Fil:128, Kernel: 3x3, Str=1, Pad=1 |
| PL2 | type: max, size 2x2 |
| FC1 | neurons:128 |
| FC2 | neurons: 2, activation: softmax |
| Act. F. | ReLU |

a

| Layer | Parameters and Architecture |
|---|---|
| Input | Input shape: k=20, d=20, c=1 |
| Conv1 | Fil:128, Kernel: 3x3, Str=1,Pad=1 |
| PL1 | type: max, size 2x2 |
| Conv2 | Fil:256, Kernel: 3x3, Str=1,Pad=1 |
| PL2 | type: max, size 2x2 |
| FC1 | neurons:128 |
| FC2 | neurons:64 |
| FC3 | neurons: 2, activation: softmax |
| Act. F. | ReLU |

b

| Layer | Parameters and Architecture |
|---|---|
| Input | Input shape: k=8, d=8, c=1 |
| Conv1 | Fil: 64, Kernel: 3x3, Str=1,Pad=1 |
| PL1 | type: max, size 2x2 |
| Conv2 | Fil:128, Kernel: 3x3, Str=1, Pad=1 |
| PL2 | type: max, size 2x2 |
| FC1 | neurons:128 |
| FC2 | neurons: 2, activation: softmax |
| Act. F. | ReLU |

c

| Layer | Parameters and Architecture |
|---|---|
| Input | Input shape: k=20, d=20, c=1 |
| Conv1 | Fil:128, Kernel: 3x3, Str=1,Pad=1 |
| PL1 | type: max, size 2x2 |
| Conv2 | Fil:256, Kernel: 3x3, Str=1,Pad=1 |
| PL2 | type: max, size 2x2 |
| FC1 | neurons:128 |
| FC2 | neurons: 2, activation: softmax |
| Act. F. | ReLU |

d

**Table 4.** Accuracy comparison against datasets and models

| Source | Year | Algorithm | Australian | German | HMEQ | LC 2007–2014 |
|---|---|---|---|---|---|---|
| [16] | 2016 | MLP | 0.860 | 0.750 | – | – |
| [1] | 2016 | ENSEMBLE | 0.880 | 0.780 | – | – |
| [34] | 2017 | ENSEMBLE | 0.880 | 0.770 | – | – |
| [14] | 2018 | ENSEMBLE | **0.930** | 0.860 | – | – |
| [10] | 2021 | CNN | 0.880 | **0.950** | 0.820 | – |
| Base | 2022 | LR | 0.871 | 0.761 | 0.803 | 0.945 |
| Base | 2022 | RF | 0.869 | 0.771 | 0.810 | 0.874 |
| Our | 2022 | CNN-DeepInsight | 0.844 | 0.734 | **0.926** | **0.991** |

**Table 5.** AUROC comparison against datasets and models

| Source | Year | Algorithm | Australian | German | HMEQ | LC 2007-2014 |
|---|---|---|---|---|---|---|
| [1] | 2016 | ENSEMBLE | **0.940** | 0.802 | – | – |
| [10] | 2021 | CNN | 0.800 | **0.960** | 0.830 | – |
| Base | 2022 | LR | 0.931 | 0.790 | 0.631 | 0.952 |
| Base | 2022 | RF | 0.870 | 0.803 | 0.872 | 0.937 |
| Our | 2022 | CNN-DeepInsight | 0.897 | 0.738 | **0.907** | **0.995** |

## 6 Conclusions

In this paper, tabular datasets were converted into images using DeepInsight. The images were used to train 2D CNN. The performance of the trained CNN was compared with literature results and base models trained by us using LR and RF for reference. We found that the trained CNN performed better than the literature results, and our base models based on LR and RF when the dataset size was greater than 5,900 samples, getting results that surpassed the Accuracy and AUROC of the second-best model with up to 0.106 and 0.046, respectively.

Additionally, thanks to the mapping generated for DeepInsight when the images were created, we can return SHAP values based on predictions of trained models to the tabular form, allowing us to generate local and global explanations.

**Acknowledgments.** This work was carried out with the support of CONACYT and the Centro de Investigacion y de Estudios Avanzados del Instituto Politecnico Nacional.

## References

1. Ala'Raj, M., Abbod, M.F.: Classifiers consensus system approach for credit scoring. Knowledge-Based Systems **104**, 89–105 (2016). https://doi.org/10.1016/j.knosys.2016.04.013

2. Baesens, B., Gestel, T.V., Viaene, S., Stepanova, M., Suykens, J., Vanthienen, J.: Benchmarking state-of-the-art classification algorithms for credit scoring. J. Oper. Res. Soc. **54**(6), 627–635 (2003)
3. Baesens, B., Lessmann, S., Seow, H.V., Thomas, L.C.: Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. Eur. J. Oper. Res. **247**(1), 124–136 (2015). https://doi.org/10.1016/j.ejor.2015.05.030
4. Baesens, B., Rosch, D.: Credit Risk Analytics, vol. 1 (2016)
5. Banu, I.M.: The impact of credit on economic growth in the global crisis context. Procedia Econ. Fin. **6**(February 2007), 25–30 (2013). https://doi.org/10.1016/s2212-5671(13)00109-3
6. Bensic, M., Sarlija, N., Zekic-Susac, M.: Modelling small-business credit scoring by using logistic regression, neural networks and decision trees. Intell. Syst. Account. Finan. Manage. **13**(3), 133–150 (2005). https://doi.org/10.1002/ISAF.261,https://onlinelibrary.wiley.com/doi/full/10.1002/isaf.261, https://onlinelibrary.wiley.com/doi/abs/10.1002/isaf.261https://onlinelibrary.wiley.com/doi/10.1002/isaf.261
7. BIS: Basel iii: International regulatory framework for banks (12 2017). https://www.bis.org/bcbs/basel3.htm
8. Brownlee, J.: Data Preparation for Machine Learning (2020). https://doi.org/10.1109/ICIMCIS51567.2020.9354273
9. Chapman, P., et al.: Crisp-dm 1.0 (2000)
10. Dastile, X., Celik, T.: Making deep learning-based predictions for credit scoring explainable. IEEE Access **9**, 50426–50440 (2021). https://doi.org/10.1109/ACCESS.2021.3068854
11. Dastile, X., Celik, T., Potsane, M.: Statistical and machine learning models in credit scoring: a systematic literature survey. Appl. Soft Comput. J. **91**, 106263 (2020). https://doi.org/10.1016/j.asoc.2020.106263, https://doi.org/10.1016/j.asoc.2020.106263
12. DFID: Growth Building Jobs and Prosperity in Developing Counttries. Departement for International Development pp. 1–25 (2007). https://www.oecd.org/derec/unitedkingdom/40700982.pdf
13. Durand, D.: Risk elements in consumer instalment financing 8 (1941)
14. Edla, D.R., Tripathi, D., Cheruku, R., Kuppili, V.: An efficient multi-layer ensemble framework with BPSOGSA-based feature selection for credit scoring data analysis. Arabian J. Sci. Eng. **43**(12), 6909–6928 (2017). https://doi.org/10.1007/S13369-017-2905-4, https://link.springer.com/article/10.1007/s13369-017-2905-4
15. Fahey, T.: Unlocking the credit cycle (2019). https://info.loomissayles.com/unlocking-the-credit-cycle
16. Ha, V.S., Nguyen, H.N.: Credit scoring with a feature selection approach based deep learning. MATEC Web Conf. **54** (2016). https://doi.org/10.1051/matecconf/20165405004
17. Hayes, A.: Credit Cycle (2021). https://www.investopedia.com/terms/c/credit-cycle.asp
18. Hofmann, H.: UCI Machine Learning Repository: Statlog (German Credit Data) Data Set. https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
19. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2323 (1998). https://doi.org/10.1109/5.726791
20. Lending-club: loan data 2007 2014 Kaggle. https://www.kaggle.com/datasets/devanshi23/loan-data-2007-2014

21. Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J.: A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans. Neural Networks Learn. Syst. 1–21 (2021). https://doi.org/10.1109/TNNLS.2021.3084827
22. Lundberg, S.M., Lee, S.I.: A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems 2017-Decem, 4766–4775 (2017). https://doi.org/10.48550/arxiv.1705.07874, https://arxiv.org/abs/1705.07874v2
23. Luo, C., Wu, D., Wu, D.: A deep learning approach for credit scoring using credit default swaps. Eng. Appl. Artif. Intell. **65**, 465–470 (2017). https://doi.org/10.1016/J.ENGAPPAI.2016.12.002
24. Miller, T.: Explanation in artificial intelligence: insights from the social sciences. Artif. Intell. **267**, 1–38 (2019). https://doi.org/10.1016/J.ARTINT.2018.07.007
25. Molnar, C.: Interpretable Machine Learning (2021)
26. Montagnon, E., et al.: Deep learning workflow in radiology: a primer. Insights into Imaging **11**(1), 1–15 (2020). https://doi.org/10.1186/S13244-019-0832-5/TABLES/2, https://insightsimaging.springeropen.com/articles/10.1186/s13244-019-0832-5
27. Petch, J., Di, S., Nelson, W.: Opening the black box: the promise and limitations of explainable machine learning in cardiology. Can. J. Cardiol. **38**(2), 204–213 (2022). https://doi.org/10.1016/J.CJCA.2021.09.004
28. Quinlan: UCI Machine Learning Repository: Statlog (Australian Credit Approval) Data Set. https://archive.ics.uci.edu/ml/datasets/statlog+(australian+credit+approval)
29. Shap: Explain ResNet50 ImageNet classification using Partition explainer - SHAP latest documentation. https://shap.readthedocs.io
30. Shapley, L.S.: Contributions to the Theory of Games (1953)
31. Sharma, A., Vans, E., Shigemizu, D., Boroevich, K.A., Tsunoda, T.: DeepInsight: a methodology to transform a non-image data to an image for convolution neural network architecture. Sci. Rep. **9**(1), 1–7 (2019). https://doi.org/10.1038/s41598-019-47765-6
32. Štrumbelj, E., Kononenko, I.: Explaining prediction models and individual predictions with feature contributions. Knowl. Inf. Syst. **41**(3), 647–665 (2014). https://doi.org/10.1007/S10115-013-0679-X/TABLES/4, https://link.springer.com/article/10.1007/s10115-013-0679-x
33. West, D.: Neural network credit scoring models. Comput. Oper. Res. **27**(11–12), 1131–1152 (2000). https://doi.org/10.1016/S0305-0548(99)00149-5
34. Xia, Y., Liu, C., Li, Y.Y., Liu, N.: A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. Expert Syst. Appl. **78**, 225–241 (2017). https://doi.org/10.1016/J.ESWA.2017.02.017
35. Yang, J., Li, J.: Application of deep convolution neural network. In: 2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2017 2018-February, 229–232, October 2017. https://doi.org/10.1109/ICCWAMTIP.2017.8301485
36. Zhang, S., Wu, Y., Men, C., He, H., Liang, K.: Research on OpenCL optimization for FPGA deep learning application. PLoS ONE **14**(10), e0222984 (2019). https://doi.org/10.1371/JOURNAL.PONE.0222984,https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0222984
37. Zhu, B., Yang, W., Wang, H., Yuan, Y.: A hybrid deep learning model for consumer credit scoring. In: 2018 International Conference on Artificial Intelligence and Big Data, ICAIBD 2018 pp. 205–208, June 2018. https://doi.org/10.1109/ICAIBD.2018.8396195