

## CREDIT RISK RATING USING STATE MACHINES AND MACHINE LEARNING

Behnam Sabeti, Hossein Abedi Firouzjaee,  
Reza Fahmi, S.H.E. Mortazavi Najafabadi

Miras Technologies International  
Number 92, Movahed Danesh St., Tehran  
{behnam, hossein, reza, hani}@miras-tech.com

Saeid Safavi, Wenwu Wang,  
and Mark D. Plumbley

CVSSP, University of Surrey,  
Guildford, Surrey GU2 7XH, UK  
s.safavi@surrey.ac.uk

### ABSTRACT

Credit risk is the possibility of a loss resulting from a borrower’s failure to repay a loan or meet contractual obligations. With the growing number of customers and expansion of businesses, it’s not possible or at least feasible for banks to assess each customer individually in order to minimize this risk. Machine learning can leverage available user data to model a behaviour and automatically estimate a credit score for each customer. In this research, we propose a novel approach based on state machines to model this problem into a classical supervised machine learning task. The proposed state machine is used to convert historical user data to a credit score which generates a data-set for training supervised models. We have explored several classification models in our experiments and illustrated the effectiveness of our modeling approach.

**Index Terms**— State Machine, Machine Learning, Classification, Credit Risk, Financial Regulation

### 1. INTRODUCTION

Credit risk refers to the possibility of loss due to a borrower’s failure to make payments on any type of debt. The goal of credit risk management is to maximize a bank’s risk-adjusted rate of return by maintaining credit risk exposure within acceptable parameters. More specifically, by measuring customers’ credit scores, banks and financial institutes monitor the expected rate of return for any debt and manage their portfolio by rejecting or adjusting high risk applications.

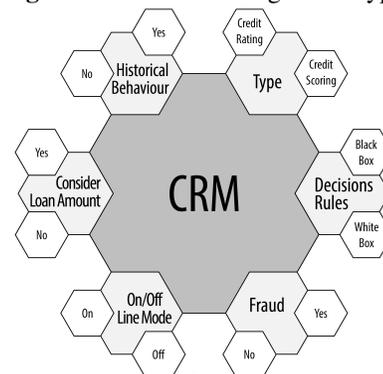
There are several configurations for a credit risk management system based on its features and expected results. These features include:

- **Type:** *Credit scoring* refers to a situation where the Credit Risk Management (CRM) system produces a score for each customer whereas in *Credit rating* a category of credit, e.g. slightly risky, is produced for each customer.

- **Decision Rules:** Specifies whether it’s possible to investigate the reasons behind a produced result or not.
- **Mode:** Specifies whether the system procedure is updated online or not.
- **Loan amount:** This feature states whether the loan amount is considered in producing the credit score or not.
- **History:** This feature specifies if the CRM system uses customer’s history to generate a credit score.
- **Fraud:** This feature indicates whether the CRM system is paired with a *Fraud Detection* module. If so, the output of fraud detectors coupled with customer information can enhance the performance of CRM systems.

Configuration features for a credit risk management system are summarized in Figure 1.

Fig. 1. Credit Risk Management Types



While credit risk management is a critical process for financial institutes, the growing number of customers and expansion of businesses have made it impossible or at least not feasible for bank to maintain their risk within acceptable parameters with traditional regulations. The increase in loan applications and also huge information available for customers,

are indicators that this process is a good candidate for automation. There are also privacy matters, where banks need to make sure that their customers' data is not accessed by any unauthorized entity. This issue can also be addressed by an automation process where customer data is not exposed at any step and is always processed by computers.

Machine learning techniques have been employed for credit risk management in many settings since user information, transactions and historical data are gathered and stored by banks [1, 2]. The information available for users can serve as features for *Regression* and *Classification* models. Several models including *Support Vector Machine (SVM)*, *Decision Tree* and *Neural Network* have been utilized for this task. These previous approaches are briefly described in Section 2.

In this paper we propose a classification model for credit risk rating, where we use available customer information, transactions and historical data to estimate new customers credit risk category. The proposed approach uses a state machine to generate credit scores for previous customers who had been granted a loan. These scores together with customer information generates a data-set for training classification models. *Support Vector Machine (SVM)* and *Decision Tree* have been employed for classification where they showed promising results. The main contribution of this paper is the proposed state machine that can be leveraged to convert previous behaviour in making payments to credit scores. In this way, a unified and consistent credit score can be computed across all previous customers, making it easy to employ machine learning models for credit risk management.

## 2. RELATED WORK

In order to approach credit risk management, more accurate and robust systems have been employed to drive expert decisions in recent years, exploring new techniques especially from the field of machine and deep learning. Recently, several approaches have been developed to address the problem of modelling the credit quality of a company or a customer, using both quantitative and qualitative information. In this section we briefly describe these methods.

Several studies employed *Support Vector Machines (SVM)* for credit risk scoring where rather than estimating a credit score, a category of credit is considered as the representative of one's credibility [3, 4]. *SVM* classifiers have been specifically found useful for feature selection and optimization [3]. To overcome the generalization issue of this approach, *Yu* proposed to integrate the concepts of fuzzy set, while adopting least square method to reduce the computational complexity of the model [5]. In order to further improve the computational complexity of *SVM* specially on large data-sets, *Harris* introduced the use of *Clustered Support Vector Machine* [6].

Tree based methods have also been employed for credit risk management [7, 2]. *Addo et al.* focused on credit risk scoring where they examined the impact of the choice of dif-

ferent machine learning and deep learning models [8]. They observed that the tree-based models are more stable than neural network models. Dimensionality reduction has also been leveraged in combination with *Decision Trees* to boost their accuracy [9]. *Random Forest* is another tree based model that improves the performance of other methods for credit risk management with a probabilistic approach [1].

Neural network is another approach to perform discriminant analysis in business research [10]. Using bank's default data, *Tam et al.* compared the neural network approach with linear classifiers, where empirical results show that neural model is a promising method of evaluating customer conditions [11]. Ensemble learning can leverage multiple *Weak* learners to boost their performance. This approach has been employed to further improve the performance of neural networks for credit risk assessment [12].

## 3. CREDIT RISK MANAGEMENT

### 3.1. Problem Definition

As mentioned in Section 1, *Credit Risk Management* aims at minimizing the risk of customer failure to meet contractual obligations. Consider a set of customers for which we have personal, account and historical information. Using this information, we want to build a machine learning model that can automatically infer credit scores for new customers. To this end, we need to find a set of features for each customer and define a credit score associated with it to build a training data-set.

We have gained access to a bank database through our partner<sup>1</sup>, which we are going to use to build our model. After investigating the data, we extracted potentially useful information which are described in Table 1. In addition to this information, the customer behaviour for repaying a loan is also available. We can use customer information to build a feature vector for each customer and leverage behaviour data to generate an associated credit score. In the following section, the procedure for converting user repaying behaviour to a credit score is described.

### 3.2. Credit Score Modeling

According to the bank's data, a customer is assigned to a state based on his adherence to contractual obligations:

- Normal: The customer has repaid all of the payments up until current date, based on the contract.
- Usance: The customer has failed to repay the last payment.
- Deferred: The customer has failed to repay the last  $k$  payments.

---

<sup>1</sup>We cannot disclose the bank information due to an *Non Disclosure Agreement*.

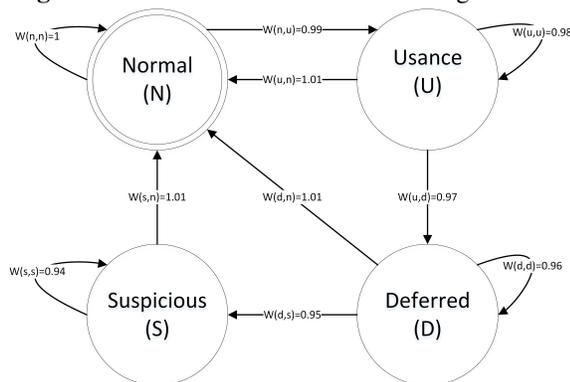
| Type           | Features   |
|----------------|--|
| Personal       | age, gender, education                                     |
| Account        | number of active, inactive and closed accounts             |
| Balance        | min, max, average and variance of account balance          |
| Transaction    | min, max, average and variance of deposits and withdrawals |
| Previous Loans | min, max, average and variance of previous credit scores   |

**Table 1.** Available customer information

- Suspicious: the customer has failed to repay the last  $n$  payments ( $n > k$ ).

These state are automatically extracted by the bank and are available in the database for each  $(customer, loan)$  pair (note that multiple loans might be granted to a single customer). In this step we need to define a strategy for converting a state sequence to a credit score. After consulting with domain experts, we defined a state machine which describes the customer transition between defined states. The proposed state machine is illustrated in Figure 2. Note that the state machine is not complete-edged since, for instance, it's not possible by definition for a customer to go straight from the *Normal* state to the *Deferred* state.

**Fig. 2.** The state machine for credit score generation



Each edge in the state machine is associated with a weight, which specifies the penalty/reward for making the transition. At the beginning of repaying period, each customer has a credit score of one and is in the *Normal* state. Each transition in the state machine alters the credit score based on the associated weight. For instance, a transition from *Normal* to *Usance* state reduces the credibility of the customer due to the failure of last payment reimbursement. Thus the initial credit score is multiplied by the transition weights which are specified by domain experts. The final result is considered as the credit score for the given

$(customer, loan)$  pair. Algorithm 1 specifies the procedure for credit score computation.

---

**Algorithm 1:** The procedure for converting a sequence of states to a credit score

---

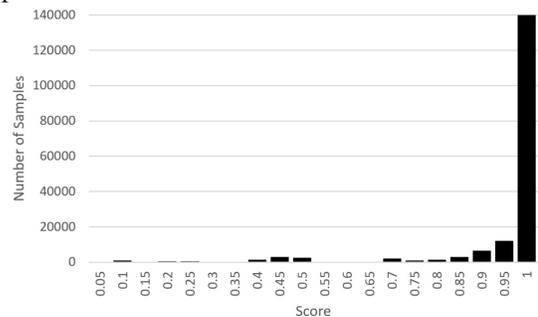
**Input:**  $S$ : State sequence,  $W$ : Transition weights  
**Output:** Credit score  
Score = 1  
 $i = 0$   
**while**  $i < \text{sequence length}$  **do**  
| Score = Score  $\times W[S_i, S_{i+1}]$   
**end**

---

Using the proposed state machine we can extract a credit score for each  $(customer, loan)$  pair. Also, user information discussed in Section 3.1 can serve as feature vectors for a pair of customer and loan. This modeling scheme converts the initial problem to a supervised machine learning task, more specifically a *Regression* task.

Figure 3 illustrates the histogram of computed credit scores for all users. A considerable proportion of users have been assigned with a credit score of one which is feasible since most users obey the obligations enforced by the contracts. This shows the asymmetric nature of the problem which should be considered in the classification phase.

**Fig. 3.** Histogram of the computed scores across all data groups



## 4. MODEL SPECIFICATIONS

So far we have introduced the problem modeling approach. In this section, some extra details are presented to complete the modeling scheme and propose the overall architecture of the proposed framework.

### 4.1. Quantization

Instead of producing a credit score for each customer, we decided to categorize the credit scores and estimate the credit class. This approach turns our problem into a straightforward classification task. The quantization required for this step has

|         | Class Name           | Criteria                 |
|---------|----------------------|--------------------------|
| 2-Class | No Risk              | $Score = 1$              |
|         | Risky                | $Score \neq 1$           |
| 5-Class | No Risk              | $0.99 \leq Score$        |
|         | With Default History | $0.95 \leq Score < 0.99$ |
|         | Slightly Risky       | $0.8 \leq Score < 0.95$  |
|         | Middle Risky         | $0.5 \leq Score < 0.8$   |
|         | Highly Risky         | $Score < 0.5$            |

**Table 2.** Quantization policy for computed credit scores

been done by domain experts. We considered two types of quantization, namely 2-class and 5-class quantization, which are described in Table 2.

#### 4.2. Parameters

The parameters of the proposed state machine are critical to the performance of the overall modeling scheme, since they specify the credit score for previous customers and construct the training data. We have asked our partner’s domain experts to set these parameters in a way that the produced scores would be interpret-able for the bank. Table 3 shows the specified parameters.

| W      | Value | W      | Value |
|--------|-------|--------|-------|
| W(n,n) | 1     | W(d,d) | 0.96  |
| W(n,u) | 0.99  | W(d,n) | 1.01  |
| W(u,u) | 0.98  | W(d,s) | 0.95  |
| W(u,n) | 1.01  | W(s,s) | 0.94  |
| W(u,d) | 0.97  | W(s,n) | 1.01  |

**Table 3.** State machine parameters assigned by domain experts

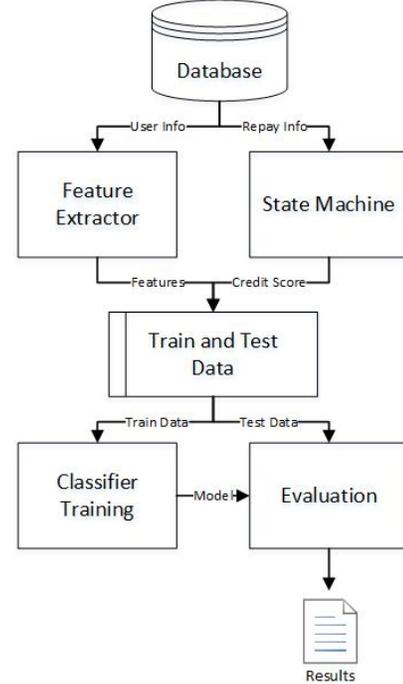
#### 4.3. Classifiers

We have used *Support Vector Machine (SVM)* [13, 14] and *Decision Tree* [15] as baselines for classification and evaluating the proposed problem modeling. In order to train the *Decision Tree* we leveraged *Gradient Boosting* [16] which will enhance the performance of classification through an ensemble of models. More Specifically, we used *CatBoost* which is a library for gradient boosting on decision trees [17]. Figure 4 illustrates the overall process for *Credit Risk Rating*.

### 5. EXPERIMENTS AND RESULTS

In this section, the proposed approach for credit risk score computation is employed to construct the training data-set. After presenting some insights about the generated data-set, *SVM* and *Decision Tree* classifiers are trained and evaluated.

**Fig. 4.** The overall process for data-set generation, modeling and evaluation.



The available information is not consistent for all customers. For instance, some customers are new to the bank and do not have any transaction data, while others might not have been granted a loan before. Thus it’s not possible to extract all features for all customers. In order to extract the features in a coherent way, we have split the users:

- Group I: Customers who had never been granted a loan before and had less than two months of transaction history in time of their application.
- Group II: Customers who had never been granted a loan before and had more than two months of transaction history in time of their application.
- Group III: Customers who have been granted a loan before and had more than two months of transaction history in time of their current application.

Based on these categories, we can extract certain features for each group. For instance, users in *group I* do not have enough transaction history to extract related features, thus we ignore transactions for this group entirely. Table 4 specifies the available features for each group of users.

So far we have introduced the procedure for credit score computation which generates the targets. Furthermore, User Splitting produces a consistent approach for feature extraction which combined with extracted targets generates the required data-set for training classification models. The obtained data-sets are described in Table 5.

| Feature Type   | Group I | Group II | Group III |
|----------------|---------|----------|-----------|
| Personal       | ✓       | ✓        | ✓         |
| Account        | ✓       | ✓        | ✓         |
| Balance        | ✗       | ✓        | ✓         |
| Transactions   | ✗       | ✓        | ✓         |
| Previous loans | ✗       | ✗        | ✓         |

**Table 4.** Available features for each customer group

|         |                 | Group I | Group II | Group III |
|---------|-----------------|---------|----------|-----------|
| 2-Class | No Risk         | 33580   | 99109    | 6917      |
|         | Risky           | 9844    | 23952    | 776       |
| 5-Class | No Risk         | 28440   | 83241    | 6125      |
|         | Default History | 8744    | 25220    | 1168      |
|         | Slightly Risky  | 2412    | 6148     | 208       |
|         | Middle Risky    | 2348    | 4696     | 88        |
|         | Highly Risky    | 1480    | 3756     | 104       |
| Total   |                 | 43424   | 123061   | 7693      |

**Table 5.** Data-set statistics

In order to evaluate the classifiers, we split 20% of the data as test set. *Accuracy (Acc)*, *Precision (P)*, *Recall (R)* and *F-Score (F1)* are the metrics used for evaluation. Table 6 specifies these evaluation metrics for *SVM* and *Decision Tree* classifiers trained in two configurations, namely 2-class and 5-class classification.

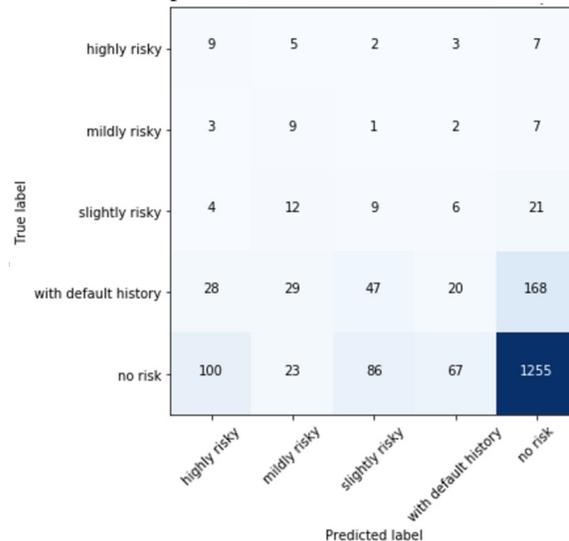
The results described in Table 6 show the effect of using more features on the classification performance. As described in Table 4 *Group III* has all the features, while *Group II* lacks previous loans features. The same condition holds for *Group I* where no balance, transaction and previous loans data is available. Both classifiers reach their maximum performance on the third data group that show the effectiveness of previous loans information. The same holds for the second data group compared to the first data group, where adding transaction data improves the performance of both classifiers.

Comparing the 2-class and 5-class configurations, we can see that in the fine-grained case the classifiers are producing smaller *F-scores*, which makes sense since increasing the number of classes makes it more complicated for the classifier to model the data. On another note, *SVM* classifier outperforms *CatBoost* in *Group I* and *Group II*, while *CatBoost* shows better performance than *SVM* on the third data group. Overall the best performance belongs to the *CatBoost* classifier in the 2-class configuration and trained and evaluated on the third data group.

In order to further investigate the performance of classifiers in the fine-grained situation, the confusion matrix for

*CatBoost* classifier trained and evaluated on the third data group, a setup which reaches the best performance in 5-class configuration, is illustrated in Figure 5. The confusion matrix shows the inability of the model to differentiate between *Risky* classes.

**Fig. 5.** The confusion matrix for *CatBoost* model trained and evaluated on Group III data



## 6. CONCLUSION AND FUTURE WORK

Credit risk refers to the possibility of loss due to a borrower’s failure to make payments on any type of debt. The goal of credit risk management is to maximize a bank’s risk-adjusted rate of return by maintaining credit risk exposure within acceptable parameters. The growing number of customers and increasing amount of stored information for each customer makes it impossible to assess loan applications using traditional methods, while making the problem specifically fit for machine learning methods. In this paper, we proposed a state machine for converting previous repayments of each user to a credit score. The generated credit scores alongside some engineered features construct a supervised classification task. We then employed *SVM* and *Decision Tree* models for classification and evaluated the models with *F-Score*. Our experiments showed that *SVM* outperforms *Decision Tree* in some cases, while *Decision Tree* classifier reaches the best performance overall. We plan to further improve the proposed approach using feature selection and domain knowledge for feature engineering. Also after gathering more data samples, *Deep Learning* models will be evaluated for the given task.

## Acknowledgment

This work was partially supported by the EPSRC Project entitled “Making Sense of Sounds” with grant reference number

|         |     | SVM   |      |      |      | CatBoost    |      |      |      |             |
|---------|-----|-------|------|------|------|-------------|------|------|------|-------------|
|         |     | Group | Acc  | P    | R    | F1          | Acc  | P    | R    | F1          |
| 2-class | I   |       | 0.53 | 0.68 | 0.54 | <b>0.53</b> | 0.41 | 0.7  | 0.41 | 0.42        |
|         | II  |       | 0.79 | 0.71 | 0.79 | <b>0.73</b> | 0.59 | 0.76 | 0.59 | 0.63        |
|         | III |       | 0.88 | 0.79 | 0.89 | 0.84        | 0.83 | 0.9  | 0.83 | <b>0.85</b> |
| 5-class | I   |       | 0.26 | 0.38 | 0.27 | <b>0.32</b> | 0.21 | 0.39 | 0.21 | 0.26        |
|         | II  |       | 0.44 | 0.55 | 0.45 | <b>0.5</b>  | 0.37 | 0.57 | 0.37 | 0.44        |
|         | III |       | 0.79 | 0.61 | 0.79 | 0.68        | 0.68 | 0.72 | 0.68 | <b>0.69</b> |

**Table 6.** SVM and CatBoost performance on test set

EP/N014111/1. We would also like to appreciate everyone at *Miras Technologies International* for their technical and moral support in this project.

## 7. REFERENCES

- [1] Jochen Kruppa, Alexandra Schwarz, Gerhard Arminger, and Andreas Ziegler, “Consumer credit risk: Individual probability estimates using machine learning,” *Expert Systems with Applications*, vol. 40, no. 13, pp. 5125–5131, 2013.
- [2] Amir E Khandani, Adlar J Kim, and Andrew W Lo, “Consumer credit-risk models via machine-learning algorithms,” *Journal of Banking & Finance*, vol. 34, no. 11, pp. 2767–2787, 2010.
- [3] Tony Bellotti and Jonathan Crook, “Support vector machines for credit scoring and discovery of significant features,” *Expert Systems with Applications*, vol. 36, no. 2, pp. 3302–3308, 2009.
- [4] Paulius Danenas, Gintautas Garsva, and Saulius Gudas, “Credit risk evaluation model development using support vector based classifiers,” *Procedia Computer Science*, vol. 4, pp. 1699–1707, 2011.
- [5] Lean Yu, “Credit risk evaluation with a least squares fuzzy support vector machines classifier,” *Discrete Dynamics in Nature and Society*, vol. 2014, 2014.
- [6] Terry Harris, “Credit scoring using the clustered support vector machine,” *Expert Systems with Applications*, vol. 42, no. 2, pp. 741–750, 2015.
- [7] Jorge Galindo and Pablo Tamayo, “Credit risk assessment using statistical and machine learning: basic methodology and risk modeling applications,” *Computational Economics*, vol. 15, no. 1-2, pp. 107–143, 2000.
- [8] Peter Addo, Dominique Guegan, and Bertrand Hassani, “Credit risk analysis using machine and deep learning models,” *Risks*, vol. 6, no. 2, pp. 38, 2018.
- [9] Anastasios Petropoulos, Vasilis Siakoulis, Evaggelos Stavroulakis, Aristotelis Klamargias, et al., “A robust machine learning approach for credit risk analysis of large loan level datasets using deep learning and extreme gradient boosting,” .
- [10] Vincenzo Pacelli and Michele Azzollini, “An artificial neural network approach for credit risk management,” *Journal of Intelligent Learning Systems and Applications*, vol. 3, no. 02, pp. 103, 2011.
- [11] Kar Yan Tam and Melody Y Kiang, “Managerial applications of neural networks: the case of bank failure predictions,” *Management Science*, vol. 38, no. 7, pp. 926–947, 1992.
- [12] Lean Yu, Shouyang Wang, and Kin Keung Lai, “Credit risk assessment with a multistage neural network ensemble learning approach,” *Expert Systems with Applications*, vol. 34, no. 2, pp. 1434–1444, 2008.
- [13] Saeid Safavi, Martin Russell, and Peter Jančovič, “Automatic speaker, age-group and gender identification from children’s speech,” *Computer Speech Language*, vol. 50, pp. 141 – 156, 2018.
- [14] Craig Saunders, Mark O Stitson, Jason Weston, Leon Bottou, A Smola, et al., “Support vector machine-reference manual,” 1998.
- [15] S. Safavi, A. Pearce, W. Wang, and M. Plumbley, “Predicting the perceived level of reverberation using machine learning,” in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 27–30.
- [16] Jerome H Friedman, “Greedy function approximation: a gradient boosting machine,” *Annals of Statistics*, pp. 1189–1232, 2001.
- [17] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin, “Catboost: unbiased boosting with categorical features,” in *Advances in Neural Information Processing Systems*, 2018, pp. 6638–6648.