ADScAI
Applied Data Science & Artificial Intelligence Symposium

Internal Risk Rating Model for Finance Institutes Based on Customer Payment Behaviour

1st Thenuwan Jayasinghe
Boffin Institute of Data Science
Colombo, Sri Lanka
thenuwanj@boffin.lk

2nd Thisara Watawana
Boffin Institute of Data Science
Colombo, Sri Lanka
thisaraw@boffin.lk

Keywords—credit risk, payment behavior, non-performing loans, time series feature extraction, clustering

I. Introduction

With the implementation of International Financial Reporting Standards (IFRS) 9, Licensed Finance Companies (LFCs) are required to adopt a risk-averse approach to conservatively predict customer risk. As per the Central Bank of Sri Lanka's (CBSL) directives on credit risk management for LFCs [1], credit facilities must be classified as either performing loans (PLs) or non-performing loans (NPLs) based on two criteria: (1) days past due (DPD) and (2) potential risk. Currently, LFCs predominantly rely on the DPD-based method. While this approach is simple to compute and interpret, it overlooks a customer's payment history and behavioral trends.

The second approach, classification based on potential risk—also known as Internal Risk Rating (IRR)—is not widely practiced. Organizations that have attempted this method primarily rely on demographic and geographical attributes. A significant drawback of this approach is that the data remains static, as it is recorded only at the time the credit facility is granted and does not capture subsequent changes.

Additionally, LFCs collect vast volumes of business transaction data daily, including detailed payment histories. However, this rich dataset is primarily utilized for business monitoring through reports and dashboards rather than for predictive modeling. This study aims to develop an IRR model leveraging customer payment history data to improve risk assessment.

II. Materials and Methods

A. Data Sources

For this study, we utilized a dataset from a vehicle leasing company. A summary of the dataset is provided in Table 1.

TABLE I. Data Sources

Data Source	Description	No. of observations
1.Contract Master	Details about the individual contract such as customer id, monthly rental, contract start and end dates	~ 668,000
2. Customer Master	Contains details on individual customer such as customer type (individual / business), birth date, gender, marital status, city	~538,000
3. Vehicle Master	Contains details about vehicle type, manufactured year, make, and model for the contracts which vehicle leasing was done	~ 661,000
4. Portfolio	Contains detail about for a given contract during its contract activation period the DPD, total arrears amount. for contracts active at any point from 2016 to 2024	~ 11,000,000
5. Payment History	At each time a payment is made by the customer for a contract, the date and the payment amount is recorded for contracts active at any point from 2016 to 2024.	~ 17,000,000

B. Key Features Extracted

Based on transactional data available in the payment history, we extracted key features to characterize customer payment patterns:

1) Number of payments made (n_payments): Customers may make single or multiple payments per month. This feature captures the number of payments made for a given contract during a specific reporting month.

2) Payments made as a percentage of EMI (emi_perc): The Equated Monthly Installment (EMI) represents the amount a customer is required to pay each month. Customers may make full payments (100% EMI), partial payments (<100% EMI), or balloon payments (>100% EMI). This feature quantifies the percentage of the EMI paid in a given reporting month.

3) Days between payments made: To analyze the regularity of payments, we extracted two features: a) The number of days between the last payment of the previous month and the first payment of the current month *(diff_prev_curr)*, and b) The number of days between the first and last payment made within the current month (*diff_curr_first_last*).

TABLE II. Time Series Features Extracted

tsfresh method	Description [2]
agg_autocorrelation	Descriptive statistics on the autocorrelation of the time series
agg_linear_trend	Calculates a linear least-squares regression for values of the time series that were aggregated over chunk
autocorrelation	Calculates the autocorrelation of the specified lag

https://doi.org/10.31705/ADScAI.2025.05

tsfresh method	Description [2]
fft_coefficient	Calculates the fourier coefficients of the one-dimensional discrete Fourier Transform for real input by fast fourier transformation algorithm
fft_aggregated	Returns the spectral centroid (mean), variance, skew, and kurtosis of the absolute fourier transform spectrum

C. *Time Series Feature Extraction*

Since the extracted key features (mentioned in section B) are computed for each contract across multiple reporting months, we obtained time series data for each contract comprising four time-series data points per contract. We then extracted time series features using the tsfresh [2] Python package. Tsfresh can automatically calculate a large number of time-series characteristics (features) from a given time-series. These features can capture complex patterns in the data such as trends, seasonality, and non-linear relationships. Table 2 summarizes some of the key time series features derived from the dataset.

A total of 332 time series features were extracted per contract. To account for variations across different product types (e.g., two-wheelers, three-wheelers, four-wheelers, mini-trucks), we segregated contracts accordingly. We then applied the k-means clustering algorithm [3] implemented in scikit-learn [4] to identify similar contract groups within each product category.

III. Results and Discussion

The clustering results revealed distinct customer segments representing varying risk levels, validated by the organization's DPD-based approach. We assessed risk levels by calculating the proportion of contracts in each cluster ever classified as NPLs (DPD > 90 days). For example, Fig. 1 illustrates clustering results for three-wheeler contracts using payment behavior. For comparison, we also applied k-prototypes clustering [5], implemented in the k-modes [6] Python library, using customer demographic attributes (Fig. 2). Payment behavior clustering provided a clearer distinction between high and low-risk segments. For example, Cluster 0 (Fig. 1) exhibited the lowest risk, while Cluster 6 had the highest. In contrast, demographic-based clustering (Fig. 2) yielded more homogeneous risk distributions. Thus, clustering based on customer payment history provides a more effective risk segmentation approach than demographic-based clustering. For business communication, clusters were grouped into four risk categories (A-D) as shown in Figure 3, where A is the lowest and D is the highest risk. A predictive model can estimate default probability by classifying contracts into these risk categories.

Limitations include potential biases in data from external events like the economic crisis, which impacted payment patterns. Retraining the model with updated data will reduce the impact of such biases. Validation with business users is also challenging. LFCs primarily use the DPD method, making the IRR model's interpretability difficult for them. Though user feedback is important for model improvement, it is difficult to obtain due to the difference in understanding between the DPD and the IRR model. Future work should focus on enhancing business stakeholder awareness of the IRR model to facilitate effective collaboration.

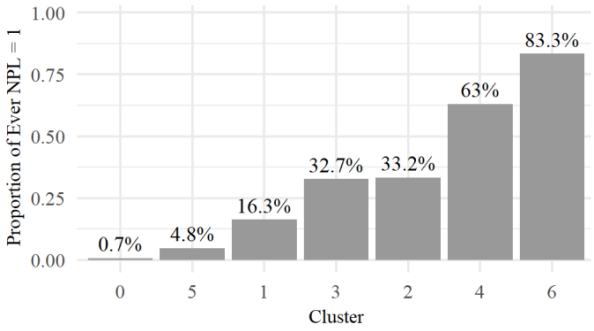

Fig 1. Clustering with time series features (product Three-Wheeler)

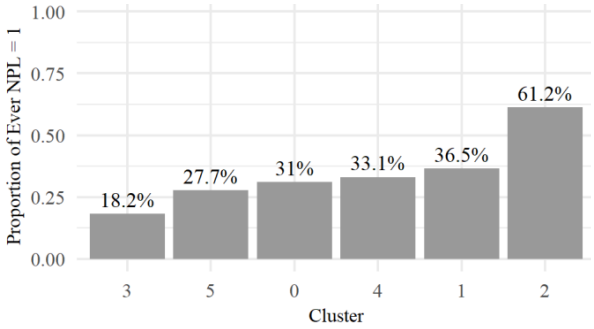

Fig 2. Clustering using customer attributes (product Three-Wheeler)

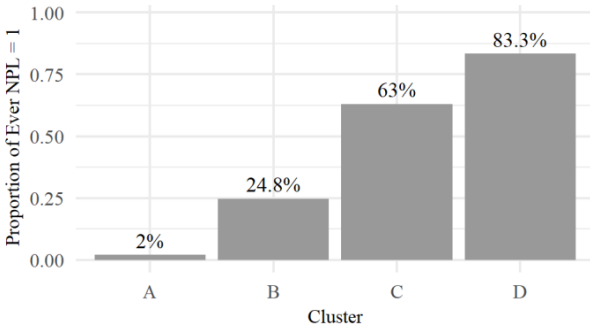

Fig 3. Reduced clusters (product Three-Wheeler)

References

[1] "Credit Risk Management for Licensed Finance Companies (LFCs)" Central Bank of Sri Lanka, 24/01/015/0004/014, Oct. 2019. [Online]. Available: https://www.cbsl.gov.lk/sites/default/files/cbslweb_documents/laws/consultation_paper_20191016_credit_risk_management_for_LFCs_e.pdf

[2] M. Christ, N. Braun, J. Neuffer, and A. W. Kempa-Liehr, "Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package)," *Neurocomputing*, vol. 307, pp. 72–77, Sep. 2018, doi: 10.1016/j.neucom.2018.03.067.

[3] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-Means Clustering Algorithm," *Appl. Stat.*, vol. 28, no. 1, p. 100, 1979, doi: 10.2307/2346830.

[4] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[5] Z. Huang, "Clustering Large Data Sets with Mixed Numeric and Categorical Values," 1997. [Online]. Available: https://api.semanticscholar.org/CorpusID:3007488

[6] N. J. de Vos, "kmodes categorical clustering library." 2015. [Online]. Available: https://github.com/nicodv/kmo