Predicting Customer Payment Methods Using Multinomial Logistic Regression and Artificial Neural Networks

Saisrijith Reddy Maramreddy
Seth Kaufman
Edvin Leon
Alexander Mendez

December 2025

1 Background and Motivation

Predicting customer payment methods is an important task in retail analytics, helping businesses optimize checkout experiences, personalize promotions, and forecast transaction processing costs. The challenge lies in the fact that payment choices are often driven by subtle behavioral patterns rather than strong numerical signals. Transaction-level features such as price, units sold, and total revenue do not always show clear separation across payment methods, making the classification problem inherently difficult.

Multinomial logistic regression provides an interpretable baseline for understanding how these features relate to payment choice. Artificial neural networks (ANNs) offer a more flexible, non-linear modeling framework, though they typically require substantial sample sizes and stronger predictive signal to outperform simpler models. Comparing these approaches helps assess whether payment behavior in this dataset is governed primarily by linear patterns or more complex structure.

2 Data Description and Objective

The dataset used in this project comes from a Kaggle repository of online retail sales and contains 240 individual transaction records. Each entry includes information such as transaction date, product name, product category, units sold, unit price, total revenue, payment method, and geographic region. The target variable, payment method, includes three categories: Credit Card, Debit Card, and PayPal.

2.1 Objective

The objective of this project is to evaluate whether customer payment method can be predicted using a leak-free set of retail transaction features. The analysis tests how well multinomial logistic regression and a simple artificial neural network classify payment method using numerical variables (unit price, units sold, total revenue, month) and safely engineered features such as HighRevenue and PriceTier. By restricting the model to non-leaking predictors, the project aims to measure the true predictive signal in the dataset and establish a realistic baseline for payment method classification.

3 Leakage-Aware Feature Engineering

Few categorical variables in the dataset exhibit deterministic relationships with the payment method, creating severe data leakage if used as predictors. Crosstab analysis shows that product category, product name, and region nearly encode the payment method one-to-one. For example, all Electronics and Sports transactions were paid using credit cards, all Clothing transactions with debit cards, and all Home Appliances transactions with PayPal. Regions follow similarly strict patterns. Including these variables would allow a model to memorize these mappings rather than learn generalizable behavioral patterns.

To avoid this, all leaking categorical features were removed, and feature engineering focused strictly on non-leaking numerical and derived variables. Two additional predictors were created: *HighRevenue*, indicating whether a transaction's total revenue exceeds the dataset median, and *PriceTier*, a three-level discretization of unit price using quantile-based binning. The final feature set therefore includes total revenue, units sold, unit price, month of purchase, HighRevenue, and one-hot–encoded PriceTier indicators.

For modeling, the payment method was encoded as an integer response variable (`Payment_Code`), with each class (Credit Card, Debit Card, PayPal) mapped to a distinct category. Numerical features were standardized using a Z-score transformation to place them on a comparable scale prior to model fitting. This leakage-free feature design provides a fair basis for evaluating how multinomial logistic regression and artificial neural networks perform when restricted to predictors that do not encode the target variable. Because some retained predictors are mechanically related, the final feature set exhibits multicollinearity. In particular, total revenue is strongly correlated with unit price and units sold by construction, and the engineered features HighRevenue and PriceTier are derived directly from these same numerical quantities. While multicollinearity can inflate coefficient variance in regression models, it does not invalidate classification performance, especially when the objective is predictive accuracy rather than causal inference. These correlated and derived predictors were therefore retained to preserve potential predictive signal, with coefficient stability later assessed using bootstrap resampling rather than relying on individual coefficient significance.

Table 1: Examples of Deterministic Mappings Creating Leakage

Feature	**Observed Mapping to Payment Method**
Product Category	Electronics → Credit Card; Clothing → Debit; Home Appliances → PayPal
Region	North America → Credit Card; Asia → Debit; Europe → PayPal
Product Name	Many items purchased exclusively with a single payment method

4 Exploratory Data Analysis

Exploratory analysis was conducted to understand the structure of the leakage-free numerical features and their relationship with the target variable. The distribution of payment methods (Figure 1) shows a moderate class imbalance, with Credit Card being the most common method, followed by PayPal and Debit Card. Stratified sampling was therefore used to preserve these proportions in the train–test split.

Boxplots of unit price, units sold, and total revenue (Figure 2) reveal substantial overlap across payment methods. Although Credit Card transactions tend to involve higher-priced items and greater revenue variability, none of the numeric predictors show strong visual separation across classes.

To further assess distributional differences, kernel density estimates (Figure 3) were plotted for each numerical feature by payment method. These smoothed densities confirm the substantial overlap observed in the boxplots: all three classes share similar distributional shapes across unit price, total revenue, units sold, and month. The KDE plots therefore reinforce that the numerical predictors contain limited discriminative structure.

A correlation heatmap of the numerical features (Figure 4) confirms expected relationships—most notably, a strong positive correlation between unit price and total revenue. Other predictors, including month of purchase, exhibit weak associations, indicating limited linear structure within the dataset. Engineered features such as HighRevenue and PriceTier were excluded from correlation diagnostics, as their dependencies are structurally defined rather than empirically discovered.

Finally, the distribution of payment methods across months (Figure 5) shows no meaningful seasonal or temporal pattern. Overall, the available numerical features provide only weak signal for distinguishing payment methods, motivating the evaluation of both linear and non-linear models under constrained predictive conditions.

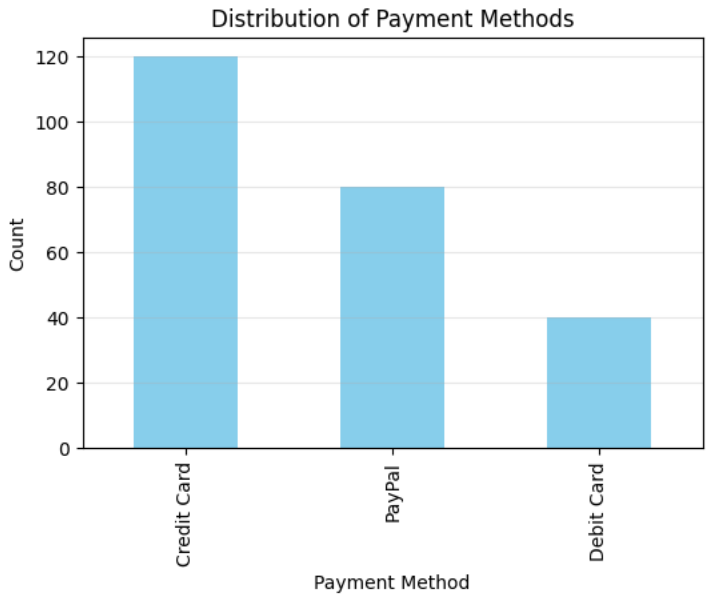

Figure 1: Distribution of payment methods across the dataset. Credit Card is the most frequent method, followed by PayPal and Debit Card.

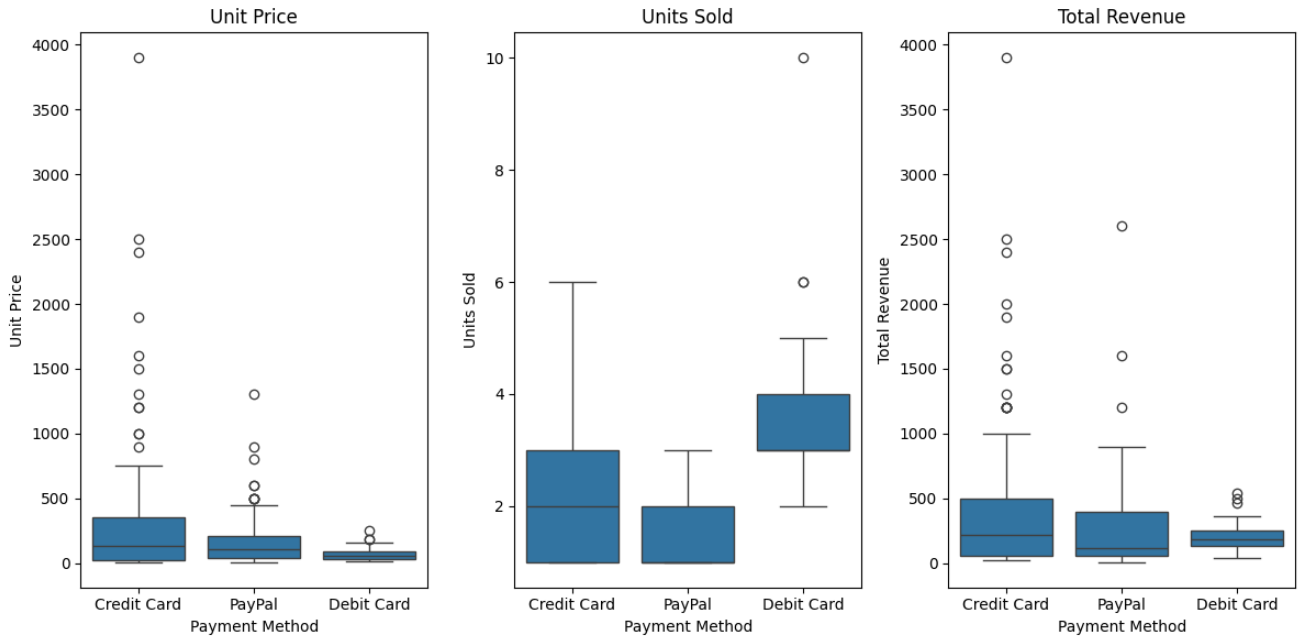

Figure 2: Boxplots of unit price, units sold, and total revenue by payment method. Numeric features show substantial overlap across classes.

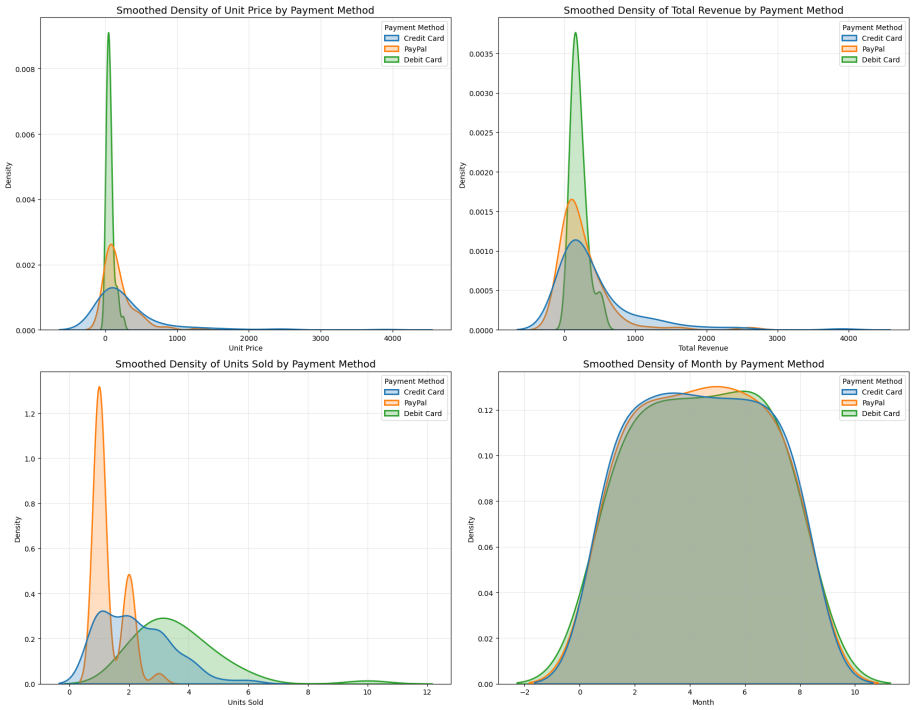

Figure 3: Kernel density estimates of numerical features by payment method.

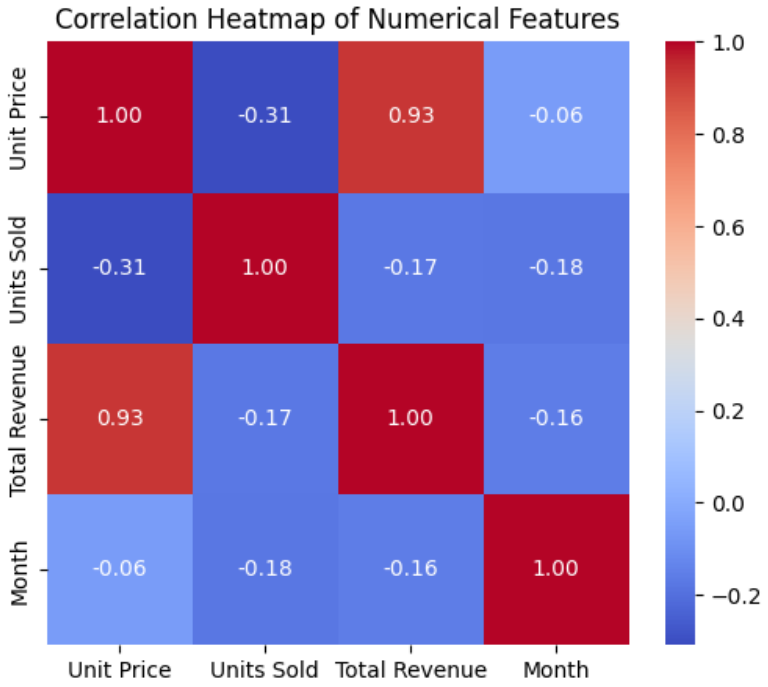

Figure 4: Correlation heatmap of numerical features. Unit price and total revenue are strongly correlated, while other variables show weak relationships.

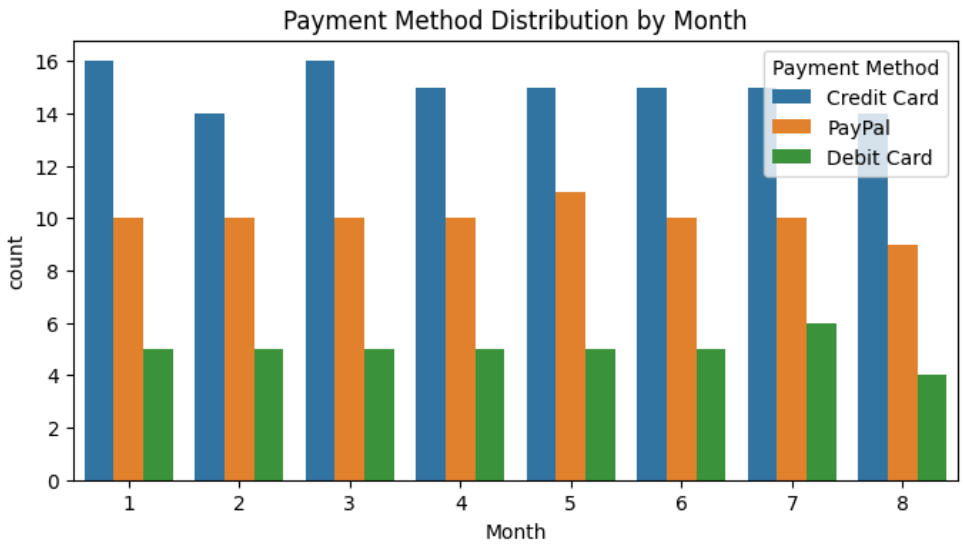

Figure 5: Distribution of payment methods by month. No meaningful seasonal pattern is observed.

5 Modeling Approach

To evaluate whether payment method can be predicted from leakage-free transaction features, two supervised learning models were implemented: multinomial logistic regression and a simple artificial neural network (ANN). All models were trained using standardized numerical predictors and the encoded target variable `Payment_Code`, where Credit Card = 0, Debit Card = 1, and PayPal = 2. An 80/20 stratified train–test split was used to preserve class proportions.

5.1 Predictor Selection Rationale

Predictors were chosen based on modeling objectives rather than automated significance-based procedures. Given the small sample size ($n = 240$) and the goal of establishing a leakage-free, interpretable baseline, we did not use stepwise selection or AIC-driven optimization. Variables with deterministic mappings to payment method were removed to prevent leakage, and a small set of behaviorally meaningful numerical and engineered features (unit price, units sold, total revenue, month, *HighRevenue*, and *PriceTier*) was pre-specified based on domain reasoning. Predictor relevance was evaluated using odds-ratio plots and bootstrap coefficient stability rather than post-hoc significance testing.

5.2 Multinomial Logistic Regression

Model performance was first evaluated using 10-fold stratified cross-validation. Accuracy values ranged from 0.50 to 0.83, with a mean of 0.68 and a standard deviation of 0.12. This variability suggests that although the numerical predictors contain limited signal, the model remains moderately stable across splits.

On the held-out test set, the multinomial logistic regression classifier achieved an accuracy of 0.67. Precision and recall varied across classes, with Credit Card performing best and Debit Card performing worst, consistent with its smaller sample size. The confusion matrix in Figure 6 shows that most misclassifications occur between Credit Card and PayPal.

To interpret the model, Figure 7 presents an odds ratio plot for all predictors. Predictors such as *Units Sold* and *PriceTier_Mid* strongly increase the odds of a Debit Card transaction, while *Unit Price* and *PriceTier_High* raise the odds of selecting PayPal. Because logistic regression is linear in the log-odds space, these effects are directly interpretable. In addition, the model was fit with L2 regularization, which helps mitigate the effects of multicollinearity by shrinking correlated coefficients toward each other, improving numerical stability without materially affecting predictive performance.

Table 2: 10-fold stratified cross-validation accuracy for multinomial logistic regression.

Fold	Accuracy
1	0.75
2	0.71
3	0.83
4	0.63
5	0.54
6	0.54
7	0.75
8	0.79
9	0.79
10	0.50
Mean	0.68
Std. Dev.	0.12

On the held-out test set, multinomial logistic regression achieved an overall accuracy of 0.67. Precision and recall varied across classes, with Credit Card performing best and Debit Card performing worst, consistent with its smaller support size. The confusion matrix (Figure 6) shows that misclassifications occur primarily between Credit Card and PayPal.

Table 3: Classification report for multinomial logistic regression.

Class	Precision	Recall	F1-score
Credit Card (0)	0.66	0.79	0.72
Debit Card (1)	0.62	0.62	0.62
PayPal (2)	0.73	0.50	0.59
Overall Accuracy		0.67	

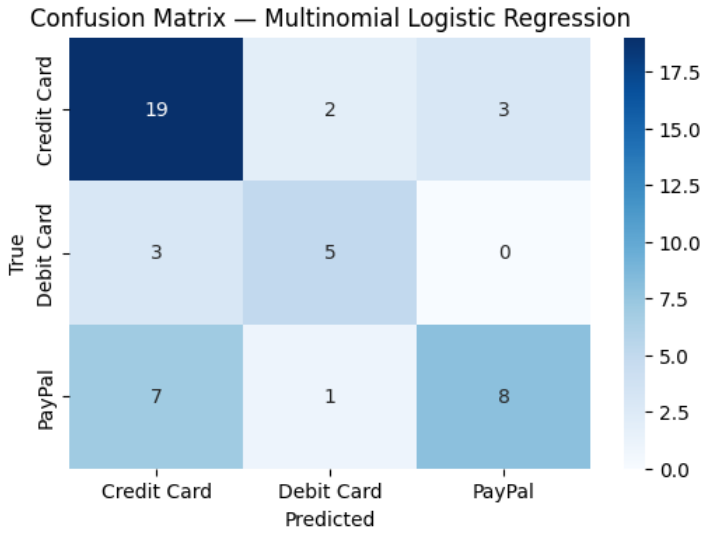

Figure 6: Confusion matrix for the multinomial logistic regression classifier.

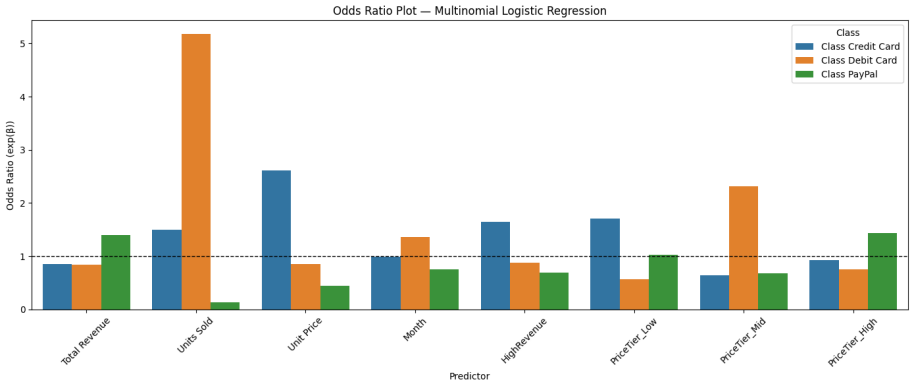

Figure 7: Odds Ratio Plot

5.3 Artificial Neural Network

An artificial neural network (ANN) classifier was trained using the same standardized predictors, with hyperparameters tuned via 10-fold cross-validation. The final model consisted of a single hidden layer with 32 units, ReLU activation, and L2 regularization ($\alpha = 0.0001$), optimized using the Adam solver. This configuration achieved a mean cross-validated accuracy of approximately 0.70.

On the held-out test set, the ANN achieved an accuracy of 0.63, which is lower than but comparable to the multinomial logistic regression model. As shown in Figure 8, the ANN performed reasonably well for Credit Card transactions but struggled to reliably identify the Debit Card class, which had the smallest support. PayPal predictions exhibited high precision but lower recall, indicating conservative classification behavior.

To provide an interpretability analogue to odds ratios, Figure 9 presents a feature influence heatmap constructed from a linearized approximation of the network's weight matrices. While these values are not directly interpretable as log-odds, they reflect the relative directional influence of each feature on the class scores after propagation through the hidden layer. The resulting patterns broadly align with the logistic regression results—most notably, *Units Sold* strongly influences the Debit Card class and negatively influences PayPal— but exhibit greater variability due to nonlinear transformations within the network.

Overall, despite cross-validation tuning, the ANN does not outperform the multinomial logistic regression model. This outcome is consistent with the small sample size and low-signal setting, suggesting that the available predictors contain limited nonlinear structure for the neural network to exploit.

Table 4: Classification report for the artificial neural network.

Class	Precision	Recall	F1-score
Credit Card (0)	0.64	0.75	0.69
Debit Card (1)	0.40	0.50	0.44
PayPal (2)	0.80	0.50	0.62
Overall Accuracy		0.63	

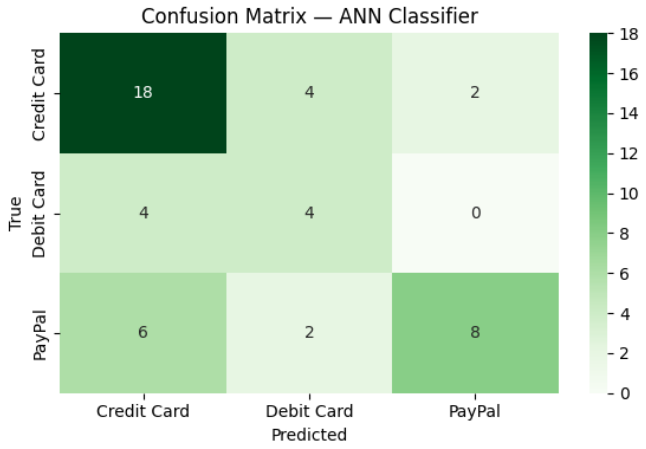

Figure 8: Confusion matrix for the ANN classifier.

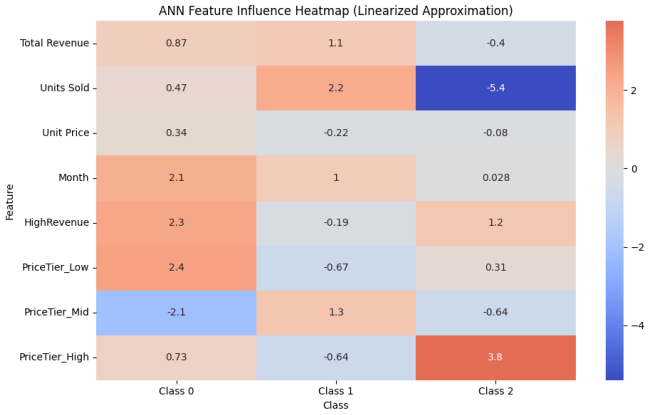

Figure 9: ANN Feature Influence Heatmap

5.4 Interpretation

The modeling results consistently show that payment method is only weakly predictable from leakage-free numerical features. Multinomial logistic regression performs the best among the tested models, achieving an accuracy of 0.67 and demonstrating stable behavior relative to the ANN. Its superior performance suggests that whatever signal exists in the data is largely linear and low-dimensional. In contrast, the ANN underperforms despite having greater representational capacity, indicating that the dataset does not contain meaningful nonlinear structure for the network to exploit. This reinforces the conclusion from the exploratory analysis: numerical features such as unit price, units sold, and total revenue exhibit substantial overlap across payment types, offering limited class separation. Consequently, model performance plateaus quickly, and additional complexity does not yield gains. The collective evidence supports the interpretation that, when categorical leakage is removed, customer payment choice in this dataset behaves more like noise than a function of observable transaction characteristics.

6 Bootstrap Stability Analysis

To assess how stable the multinomial logistic regression model is, a bootstrap procedure with 1000 resamples was conducted. In each iteration, a new training sample was drawn with replacement, the model was refit, and its accuracy was evaluated on the same test set. This approach provides an empirical measure of how much the model's performance varies across different samples of the data.

6.1 Bootstrap Accuracy Distribution

Table 5 summarizes the bootstrap accuracy results. The mean accuracy was 0.657 with a standard deviation of 0.032, and the 95% percentile interval ranged from 0.583 to 0.708. These results closely match the original model

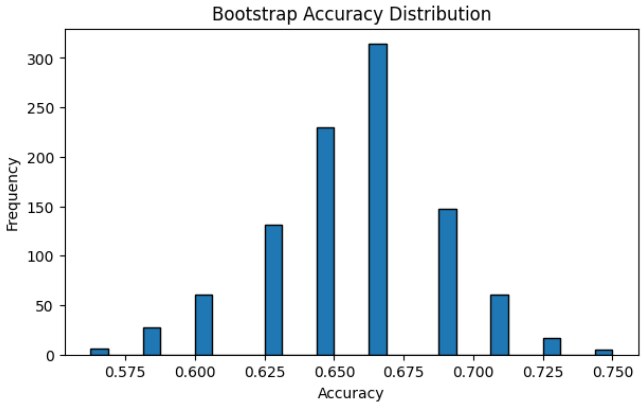

Figure 10: Distribution of test accuracy across 1000 bootstrap replications

accuracy of 0.67, indicating that the classifier is moderately stable across different samples of the training data.

Table 5: Bootstrap accuracy summary based on 1000 resamples.

Statistic	Value
Mean Accuracy	0.6567
Standard Deviation	0.0316
95% Percentile CI	[0.5833, 0.7083]

Figure 10 shows the empirical distribution of accuracies. The resulting accuracy histogram is unimodal and tightly clustered, indicating that the model's performance is fairly stable across bootstrap samples despite the limited predictive signal in the features.

6.2 Bootstrap Coefficient Intervals

Bootstrap resampling also provides empirical confidence intervals for the multinomial regression coefficients. Table 6 reports a subset of the most informative class–feature combinations, showing the mean coefficient and corresponding 95% percentile interval.

Table 6: Bootstrap coefficient intervals for selected predictors in the multinomial logistic regression model.

Class	Feature	Mean Coef	2.5%	97.5%
Credit Card	Units Sold	0.395	0.045	0.717
Credit Card	Unit Price	0.976	0.477	1.489
Credit Card	PriceTier_Low	0.563	0.214	0.937
Credit Card	PriceTier_Mid	-0.483	-0.802	-0.139
Debit Card	Units Sold	1.711	1.330	2.196
Debit Card	PriceTier_Low	-0.577	-1.010	-0.157
Debit Card	PriceTier_Mid	0.879	0.430	1.283
PayPal	Units Sold	-2.105	-2.526	-1.723
PayPal	Unit Price	-0.798	-1.295	-0.287
PayPal	PriceTier_Mid	-0.396	-0.768	-0.013

These selected coefficients show the clearest and most stable effects. For example, *Units Sold* has a positive association with Debit Card and a strong negative association with PayPal, while higher unit prices and low price tiers are more strongly linked to Credit Card payments. In the full 24-row coefficient table (not shown here), many other predictors have intervals that include zero, indicating weaker and less stable relationships with payment method.

6.3 Interpretation

The bootstrap results reinforce the main modeling conclusions. The accuracy distribution is tightly concentrated around 0.60–0.70, indicating that model performance is stable but modest. The full coefficient table shows substantial uncertainty, with many intervals containing zero, suggesting that most predictors exert weak or inconsistent effects across resamples. Only a few features, such as *Units Sold* and *PriceTier*, exhibit reliably nonzero coefficients, and even these effects remain moderate in magnitude. Overall, the bootstrap analysis confirms that the model's limitations arise from the low predictive capability of the leakage-free numerical features rather than instability in model estimation.

7 Discussion and Future Work

The results of this project illustrate both the potential and the limitations of predicting payment method using leakage-free numerical features. After removing categorical variables with deterministic mappings, the remaining predictors offered only modest discriminatory power. This pattern was consistent across analyzes: exploratory plots showed substantial overlap between classes, logistic regression achieved moderate precision, and the ANN did not uncover additional nonlinear structure. The bootstrap results confirmed that these results are stable rather than artifacts of sampling variability, pointing to the inherent low predictive capability of the available features.

Future improvements would likely require richer and more diverse predictors. Customer-level attributes (e.g., purchase history or loyalty status), product metadata (e.g., brand or quality indicators), and finer temporal granularity (e.g., time of day or promotional context) could introduce behavioral patterns absent from this dataset. More sophisticated feature engineering—such as interaction terms, engineered price–volume ratios, or learned latent representations—may also reveal structure not captured by the models used here. With larger datasets, deeper neural networks or alternative approaches such as gradient boosting could be evaluated without significant risk of overfitting.

Overall, this study highlights the importance of controlling for data leakage and demonstrates how doing so provides a more realistic understanding of what numerical features can—and cannot—predict. The findings emphasize that meaningful progress in payment method prediction will depend primarily on the availability of richer, behaviorally informative features.

Appendix: Code Repository

All code and analysis for this project, including the full Jupyter notebook, is available at:
`github.com/srijith-reddy/Payment-Method-Classification-ML-ANN`