CMPT 459 Project Report

Members: Lazar Pajic, Preston Quach, Gen Blaine

Problem Statement:

Credit card payment defaults cause significant financial losses for lenders and long-term harm to customers. To address this, we will preprocess and analyze a dataset containing 30000 data points of clients from Taiwan. This includes 24 features that include demographic and financial qualities. We intend to build and compare predictive models that can accurately forecast whether a client will default next month.

Summary of Methodologies Applied:

Our team implemented a comprehensive data mining pipeline to predict credit card default using the UCI Credit Card dataset. The workflow was divided into four critical parts:

A. Data Preprocessing & Exploratory Data Analysis (EDA):
   a. Data Cleaning: We verified data integrity by checking for null values and duplicates. Inconsistent values in categorical features were filtered and categorical variables were one-hot encoded to create binary indicators.
   b. Class Balancing: Applied Synthetic Minority Over-sampling Technique (SMOTE) to address the significant class imbalance in the target variable default.payment.next.month (was originally ~78% non-default and ~22% default).
   c. Normalization: Numerical features were standardized using StandardScaler to ensure zero mean and unit variance, which was crucial for distance-based algorithms.

B. Clustering Algorithms
   a. We have tested three clustering algorithms. These included GMM, hierarchical clustering, and k-means.

C. Outlier Detection:
   a. We used three outlier detecting algorithms, which are LOF, Isolation Forest, and Elliptic Envelope.

D. Feature Selection
   a. We used three feature selection algorithms. These were lasso regression, mutual information, and RFE. We then analyzed common features that had little impact on the dataset and removed them.

E. Classification & Hyperparameter Tuning
   a. The three classifiers tested are random forest, KNN, and SVM. Each classifier was compared with and without hyperparameter tuning.

Key Results and Findings:

Probabilistic Clustering: Gaussian Mixture Model was employed for probabilistic clustering. The optimal number of components was determined to be 2, guided by the higher Silhouette Scores 0.1206 and the lower Davies-Bouldin Index 3.6517 along with domain alignment of Defaulters vs Non-Defaulters. Cluster 0 (high risk) is characterized by lower credit limits (LIMIT_BAL mean: -0.23) and a history of payment delays (positive PAY_0 mean: 0.29). Cluster 1 (Low Risk) exhibited higher credit limits (LIMIT_BAL mean: 0.48) and consistent payment history (negative PAY_0 mean: -0.8). The clusters showed significant visual overleap in the PCA space, suggesting credit risk exists on a continuum rather than in strictly separated groups.

Hierarchical clustering: The hyperparameters for hierarchical clustering were n, number of clusters, and the distance function. The best hyperparameters by silhouette score were 2 clusters with the euclidean distance. The silhouette score was very high, around 0.95. However, it is clear by looking at the plot that it was not very useful as one of the clusters was much larger than the other. The smaller cluster seemed to only be one point that appears to be an outlier compared to the other points. So, hierarchical clustering did not provide any useful results despite the high silhouette score.

K-means: The main hyperparameter tested is the number of clusters ranging from 2 to 11. After testing, two plots were made which were a plot that displayed the K-Means elbow method and the silhouette scores. Based on the Elbow Method graph, we can see that the elbow point of the data appears at k = 3 due to the largest decrease occurring between 2 and 3. We also see silhouette scores peak at k=3, so k = 3 is chosen. A silhouette score of 0.2317, implies the dataset does not have well-defined cluster boundaries. For future tests, testing algorithms like DBSCAN may capture more complex patterns of the dataset.

Local outlier factor: We used Local Outlier Factor (LOF) with n_neightbors=50 and contamination=0.01 to identify local density deviations. Different n_neighbors and contamination amounts were tested before settling with 50 and 0.01 respectively and only kept this version in our notebook to keep things simple and clear. We chose a neighborhood size of 50 to minimize false positives from local noise by using a robust local context. We selected a contamination rate of 1% to proactively remove a narrower set of erratic financial behaviours, so the data reflects a more stable and representative population of normal credit usage. Although the PCA plot shows outliers scattered throughout the plot rather than solely at the periphery, this can be attributed to how LOF identifies points that are significantly less dense than their specific neighbours along with us visualizing a 24-dimensional dataset compressed into just 2 dimensions, so in the 2D plot an outlier might appear to overlap an inlier, but in the high-dimensional space, it may be clearly isolated.

Isolation forest: We tested six different contamination levels for isolation forest. These were 0.01, 0.05, 0.1, 0.125, 0.15, 0.2. After visualizing all of the different contamination levels, it was determined that 0.01 had the best results. The other contamination levels added points that are not outliers to the list of outliers. To keep the notebook's output clean, only this version of the isolation forest was visualized. The model seemed good at identifying outliers as most of the chosen points were clearly outside of most other points.

Elliptic Envelope: 11 different contamination levels were tested for elliptic envelopes which range from 0.01 to 0.1. After visualizing the different contamination levels for elliptic envelopes, contamination level of 0.01 appears to have the best results as it grasps the outliers found in the scatter plot without including many false positives. Elliptic Envelope does not do a great job at detecting outliers as it already appears to include false outliers. Increasing the contamination level would only increase the number of false positives. As a result, we decided to not use this method to remove outliers.

Lasso regression: The variables MARRIAGE_3 (others), MARRIAGE_2 (single), and MARRIAGE_1 (married) had the highest absolute coefficients (ranging from -5.5 to -6.1). The strongly negative signs indicate that having a known marital status is statistically associated with a significantly lower probability to default. On the other hand, PAY_0 had the highest positive coefficient with 0.57. This confirms that a recent delay in payment is the single strongest risk factor for predicting future defaults. BILL_AMOUNT4 was the only redundant feature that was assigned a coefficient of 0 suggesting that it provides no unique information not already captured by other bill amount variables. All other features were retained and ranked by their absolute coefficient. Comparing the top 10 features from Lasso Regression against the top ranked features from RFE reveals that the features match exactly. This strong consensus between the two distinct selection methodologies highlights the robustness of these predictors and confirms their critical role in the model.

Mutual information: 'LIMIT_BAL' was found to be the most important feature with a score over 0.1. The 'PAY_AMT' variables were also important with scores above 0.8. 'AGE', 'MARRIAGE_3', 'EDUCATION', 'MARRIAGE_1' were the variables with the lowest mutual information. Mutual information scores each feature independently which is why similar features like the various 'PAY_AMT' and 'BILL_AMT' variables have similar scores with each other. The other feature selection methods tend to pick one from a group as the best. Everything with a mutual information score less than 0.02 was removed. This removed 'AGE', 'MARRIAGE_3', 'EDUCATION', 'MARRIAGE_1', 'BILL_AMT6', and 'BILL_AMT1'. Since most of the scores were similar, the focus was on removing less significant features rather than only using the best features.

RFE: RFE was performed to find the more important features within the dataset. Based on the results, MARRIAGE_3, MARRIAGE_2, MARRIAGE_1, PAY_AMT2, BILL_AMT3, BILL_AMT1, PAY_AMT1, PAY_2, PAY_0 Education, Sex, and LIMIT_BAL were selected to be the 12 most important features of the dataset (rank 1 represents the most important). The remaining features were ranked lower making them less important. When comparing the top 12 features with Lasso Regression, we see that both methods give similar results, giving a strong confirmation on the important features.

Feature Selection: We determined common features that were least impactful when deciding if a customer defaults. These include BILL_AMT4, BILL_AMT2, AGE, and BILL_AMT5. These features are removed and tested with the classifiers.

Random forest: The parameters tuned for the Random Forest were n-estimators, max_depth, min_samples_split, and min_samples_leaf. The best parameters found through grid search were: 200 for n_estimators, None for max_depth (allows fully grown trees), 1 for min_samples_leaf, and 2 for min_samples_split. The model achieved an accuracy of approximately 0.8551 and an F1-score of 0.8495. The AUC-ROC was notably high at 0.9224, indicating excellent separation between the classes. These results were obtained using the feature set selected by Lasso Regression. When comparing the tuned model to the feature selected model, only slight decreases were found across all metrics with highest being -0.0092 in Recall, suggesting that the Random Forest algorithm is naturally robust and performed near-optimally even without extensive tuning.

K-nearest neighbours: The parameters for KNN were n_neighbours, weights, algorithm, and leaf size. The accuracy, precision, recall, and f1-score were around 0.78 to 0.79 which is good enough on the base model with all features. The AUC-ROC was 0.86. When only using the features selected during feature selection, the results improved. There was a slight increase in all of the scores which means feature selection was effective at improving the model's performance. Comparing the tuned model to the feature selected model shows slight increases in the accuracy, f1-score, and AUC-ROC. The accuracy and f1-score were around 0.814, and the AUC-ROC was around 0.888, implying KNN seems to work well in classifying.

Support vector machine (Linear SVC): The parameters tuned for the SVM were the regularization parameter denoted as C, loss function, the penalty method, and whether the algorithm solves the dual or primal optimization problem. The accuracy, precision, recall, f1-score, and AUC-ROC score of the base model were between 0.68 and 0.84 with Recall having the lowest score of 0.6867 while AUC-ROC having the highest of 0.8451. When using the features selected during feature selection, there were minimal improvements, showing that the algorithm performed near-optimally. When comparing the tuned model to the feature selected model, there were barely any notable changes showing that the default values parameters performed just as well as the tuned parameters. Overall, Linear SVC performed the worst compared to the other two since the dataset is not really separable linearly.

Domain-specific Insights Derived From the Analysis:

The PAY_0 feature emerged as the dominant predicator for credit card defaults. The dominance of PAY_0 over demographic features (Age, Education, Marriage) suggests that a client's current financial liquidity is a far better risk indicator than their static background. This leads us to the conclusion that financial institutes should prioritize dynamic behavioral data over demographic profiling. From this analysis, we can derive that it is difficult to accurately determine whether a person defaults on his/her payments. Although our models were able to get decently high accuracy, f1 and AUC-ROC scores, this was with many important variables. With only a few demographic based variables, it would be nearly impossible to predict the credit card default. So, we cannot make any conclusive statements about how demographics affect the probability of default payments.

Challenges Encountered and how they were Addressed:

The 'no default' class had many more entries compared to the 'default' class. To solve this, we used SMOTE which creates new data points to balance out the data. However, a new problem arose where the dataset was too large. This caused Hierarchical Clustering, kNN, non-linear SVM took substantially longer to run. To deal with this, the large dataset was sampled for Hierarchical Clustering. For the Classifiers, we decided to drop some hyperparameter options to keep the running time reasonable, and adjusted non-linear SVM to linear SVM.

Conclusion:

The process reinforced that data mining is iterative, and insights from clustering and feature selection refined earlier steps. We have also learned to validate metrics visually, as a high clustering score was misleading, and to make practical compromises, like using a linear SVM, to manage computational costs. While the customer data lacks natural clusters, a well-tuned Random Forest can effectively identify default risks. Strategic feature selection improved model efficiency without severely negatively impacting its performance. This confirms that a methodical process of processing, understanding, and verifying data is essential for building reliable models.