Rotman School of Management
UNIVERSITY OF TORONTO

Scotiabank

Big Data & AI Case Competition

TEAM 6: Alan Liu, Devashree Kumar, Humza Butt & Kevin Zhang

Executive Summary

The purpose of this report is to find the set of characteristics that can best predict whether or not a customer will be classified as a potential low, medium or high risk customer for money laundering. There are 34 attributes (including customer and transaction data sets) and one output variable. The training dataset contains about 3128 records and the test data set contains 1341 records. The objective is to use the dataset to build a model that will predict the rating (whether it can be classified as low, medium or high risk customer). The big data set will be used to identify the characteristics those can be used to classify the customers.The main objective is to use the data to help classify potentially risky customers for money laundering including the following steps:

- Perform data processing on the data set.
- Construct a neural network using all available predictors in the customer dataset as the first model and compare it with other models like random forest, gradient boosting etc using a pipeline.
- Identify the most important variables for predicting income.
- Construct another neural network that maximizes cross validation accuracy hyperparameter tuning and adding the transaction dataset along with customer dataset.
- Compare the results with other suitable models such as a decision tree, SVM, random forest etc.
- Perform K-means clustering on the big dataset and identify characteristics within the clusters to classify as low, medium and high risk customers.

Some of the challenges observed while building the neural network model include deciding on number of epochs to run, number of neurons which would provide the highest test accuracy, number of hidden layers and which activation function to be used for each layer. Another major challenge was working with the big dataset and the technical limitations of our GPUs.

The best model generated a test accuracy of 48.59% for the neural network medium model. The most significant features for classifying a customer as a potentially high risk are: the usage of visa and credit cards, employment, consistency transaction amounts, and account status. Through implementing the segmentation model, the accuracy for labeling the high risk profile increases as it can provide additional insights for end-users.

1.0 Introduction

Money Laundering has become a rising concern across the globe. The action of using layering accounts to clean illegally obtained money damages financial sector institutions, which are critical for economic growth (“Negative Effects of Money Laundering on The Economy”). Money Laundering has become a national crisis in the Canadian economy, accounting for 2.13% of GDP. Over 200 billion dollars are laundered across Canada over five years (Punwasi). In recent years with the advancement of data mining and machine learning approaches, the financial institutions started the Anti-Money-Laundering (AML) initiatives to identify the customer accounts associated with Money Laundering. Scotiabank, as one of the leading financial institutions, has suffered from the money laundering problem. The data science team has worked with the AML experts in identifying and labelling a few customers' accounts with the risk profile. The data science team at Scotiabank has asked us to develop several machine learning models in identifying all the accounts with high risks for Money Laundering.

1.1 Scope and Objectives

Scotiabank is interested to know which individual customers are considered to be at high risk for money laundering with the given dataset. We have decided to approach the question with three different models: "Easy" supervised learning model, "Medium" supervised learning model and "Hard" unsupervised learning. The team is tasked to discover some dominant factors to consider an account as a high risk for money laundering. The team will implement Machine Learning Algorithms to identify customer accounts' risk rating by analyzing different factors. The unsupervised learning for the "Hard" model can draw insights into Money Laundering Account characteristics. The team has narrowed down the questions to focus on only individual customers since the labelled training data only consists of individual customers.

2.0 Data Overview

Scotiabank has provided two distinctive datasets: customer account information and transaction history. In terms of the customer account data, approximately 30 million masked accounts have been provided, and only roughly 4500 customer accounts are labelled with the Money Laundering risk rating. These labelled data can be used as the training dataset for the Easy and Medium model to determine the risk rating for the rest of the customer account. On the other hand, more than 127 million aggregated transactions are provided, which are associated with the 30 million accounts. Through conducting a brief analysis on the dataset, we have discovered that the training dataset is not a full representation of the big test dataset. The training data only consists of individual customers accounts and only limited country code. Appendix A has illustrated that the majority of the data is Canadian accounts and there is only less than 2% of the total data involves foriegn accounts. Therefore, the team has emphasized on the Canadian accounts for the prediction modeling.

2.1 Data Cleaning and Processing

In order to reduce the dimension of the data after encoding, we have decided to clean the data and remove some of the features with similar information.

In terms of "Occupation code" and "Occupation Status" has provided a respective meaning with regarding employment information. The team has decided to use the occupation status with Unemployed, Retired, Self-employed and Employed. The customer account has multiple statuses, including active, inactive, not a customer, and pending. We have narrowed down the status to only "Active" and "Others" to reduce the dimensionality. Similar to the relationship type for the customer account, there are over 22 different categories. The team has limited the categories to the top 10 most common relationship types to simplify the data.

There are 217 different countries being provided in the dataset for the country code, and 98% of the accounts are Canadian accounts (Appendix A). Considering selective bias and omitting variable bias, we have decided to group all the foreign country codes to limit the bias and narrow down the business objective. For similar reasons, we have grouped the jurisdiction code in the datasets. This also reduces the dimensionality after encoding for feature selection. Other missing data like account types, we have assumed to fill in zero to represent that no accounts are available.

For each customer account, there are multiple transaction data entries with different months. In order to have a single data frame for model analysis, the team has grouped the transaction data with a few aggregated statistics as shown in the table below:

Table 1: Aggregate measures for transaction data

Original Columns	Incoming Amount Incoming Count	Outgoing Amount Outgoing Count	Others
Aggregated Columns	Count, Average, Max, Min, Standard Deviation	Count, Average, Max, Min, Standard Deviation	Number of Month without incoming or outgoing transaction, Number of transaction type used Usage Count for each type of transaction

After combining the customer dataset with the aggregated measures for the transaction data, dummy encoding is performed for all the categorical variables. The dataset is then normalized with the Min-Max method to ensure the data has the same scale for modelling.

3.0 Exploratory Data Analysis

After we have cleaned and grouped the data, we have conducted a few exploratory analyses (Appendix B) to verify some of the assumptions that we have made in the previous sections. Crosstab analysis is conducted for all the categorical variables, including occupation status and customer account status. We have observed that most of the high risks for money laundering are

present for inactive accounts shown in Figure 1. On the other hand, retired customers are less likely to commit money laundering, as from the labelled risk profile illustrated in Figure 2.

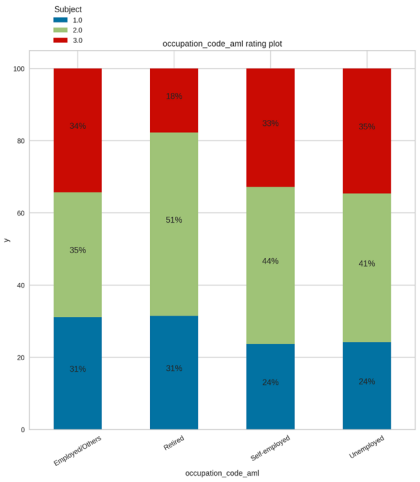


Figure 1: Crosstab Analysis for Occupation Status

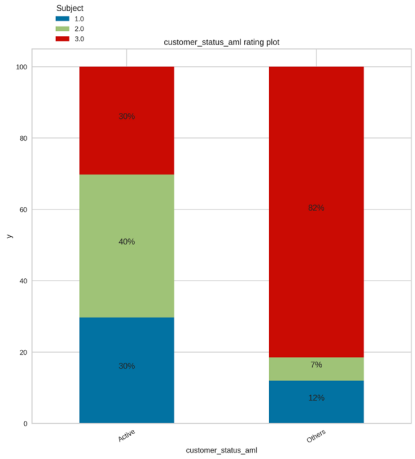


Figure 2: Crosstab Analysis for Customer Status

For Transaction Data, there is a strong seasonality observed where high value and volume during December holiday seasons/ summer holiday and low during the April (Figure 3)

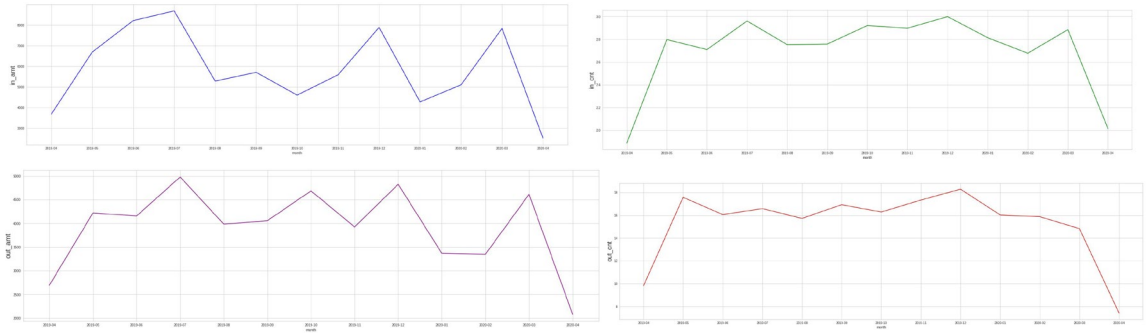


Figure 3: (Top Left) average incoming transaction amount (Top Right) average incoming transaction count (Bottom Left)average outgoing transaction amount (Bottom Right) and average outgoing transaction count

According to the transaction count boxplot in figure 4. It is also interesting to see that most Canadians do not use American Express and have similar transaction statistics on debit and visas.

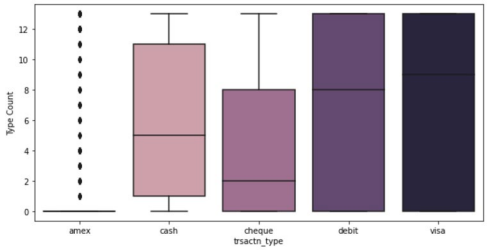


Figure 4: customer level transaction type count

4.0 Easy Model

4.1 Pipeline Model:

To explore other models beyond neural networks, a pipeline was created for Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Machine (SVM), Gradient Boosting Classifier, and KNN. A GridSearchCV was utilized to find the optimal parameters for each model, and the model with the best output was determined in terms of test set accuracy. The F1 score, Recall, Precision, and AUC metrics were utilized as well for comparison. The optimal model was determined to be Gradient Boosting Classifier with the optimal parameters listed in Appendix C and Appendix D. This makes sense, as Gradient Boosting Classifier should perform better than Decision Trees as they learn from the residuals of the last grown trees. Additionally, as this is a multi-label classification problem, logistic regression would not perform as well either. KNN should also perform worse due to the data being high dimensional, requiring even more data due to the curse of dimensionality.

4.2 Neural Network Model:

The model was chosen from multiple model comparisons to fit on the training set (70% of the whole dataset) , based on the accuracy of the validation set that contains 30% of the whole train dataset.

The model was constructed with all predictors. The neural network model is fitted with fixed input layer nodes of 28, activation function ‘relu’ on the input layer, activation function ‘softmax’ and loss=’categorical_crossentropy’ on the output layer. There are three hidden layers with 256 neurons, 368 neurons and 96 neurons.

Keras Tuner was utilized to search and select the optimal number of neurons, learning rate, and activation functions that are used in the hidden layer to maximize validation accuracy and minimize loss. During the tuning process, the Hidden layer’s node number choices and other neural network’s parameter choices are as followed:

- The number of neurons that are used in the hidden layer was chosen between 16 to 512
- Activation function types in the hidden layer were tested using relu and sigmoid
- Model optimizers were tested using rmsprop
- Number of epochs were chosen manually from a range of 50 to 200, with step size of 50
- The batch size was choose from a range of 30 to 50

The best model obtained after careful comparison between different settings of the Neural Network was then passed on to predict the target values in the test dataset.

4.2.1 Neural Network Model Result

The overall test accuracy of the model was 47.58%. The resulting neural network is modelled as follows:

OPERATION		DATA DIMENSIONS	WEIGHTS(N)	WEIGHTS(%)
Input	#####	28		
Dense	XXXXX	----------	7424	5.4%
relu	#####	256		
Dense	XXXXX	----------	94576	68.7%
relu	#####	368		
Dense	XXXXX	----------	35424	25.7%
relu	#####	96		
Dense	XXXXX	----------	291	0.2%
softmax	#####	3		

Figure 5: Model Diagram

5.0 Medium Model

5.1 Feature Selection

As more variables were added to the model, we decided to conduct feature selection through multiple methods. Firstly, we utilized a Decision Tree classifier to determine the feature importance score for each variable. The feature importance score provides the relative importance for predictors when classifying the target variable. The results are seen in the Appendix D Figure D1. Then, we decided to use logistic regression with L1 regularization to conduct both Lasso regression. This will allow us to determine what variables should be removed from the model. The selected variables are shown in Appendix D Figure D2. This led to a model with a final count of 47 predictors.

5.2 Pipeline Model:

The pipeline approach for the easy model was used again, but this time with the transaction dataset using the same algorithms as before. The optimal model for the medium approach was shown to be SVM with the optimal parameters listed in the Appendix D below. Although SVM was shown to be the best model, it performed fairly similarly to the Gradient Boosting Classifier and Random Forest algorithms with 0.001 difference. Although the SVM model performs better, Gradient Boosting Classifier can be thought of as better as the problem is a multi label classification problem. So we would recommend the Gradient Boosting Classifier. Principal Component Analysis was not being chosen due to the lack of interpretability the algorithm provides. Insights are not easily created, and the difference in test accuracy rate does not warrant the decline in interpretability. The results are in Appendix D.

5.3 Neural Network Model:

The neural network model was chosen from multiple model comparisons to fit on the training set (70% of the whole dataset) , based on the accuracy of the validation set that contains 30% of the whole train dataset.

The model was constructed with all predictors. The neural network model is fitted with fixed input layer nodes of 56, activation function ‘sigmoid’ on the input layer, activation function ‘softmax’ and loss=’categorical_crossentropy’ on the output layer. There are two hidden layers with 208 and 288 neurons respectively and the activation function used for the hidden layers is ‘sigmoid’.

The neural network performs feature engineering and hence the feature selected variables were not explicitly passed as an input to the neural network model. The hidden unit activations of the pruned network are the features extracted from the original dataset.

Keras Tuner was utilized to search and select the optimal number of neurons that are used in the hidden layer to maximize validation accuracy and minimize loss. During the tuning process, Hidden layer's node number choices and other neural network's parameter choices are as followed:

- The number of neurons that are used in the hidden layer were chosen between 16 to 512
- Activation function types in the hidden layer were tested using relu and sigmoid
- Model optimizers were tested using rmsprop
- Number of epochs were chosen manually from a range of 100 to 200, with step size of 50
- The batch size was choose from a range of 30 to 50

The best model obtained after careful comparison between different settings of the Neural Network was then passed on to predict the target values in the test dataset.

5.3.1 Medium Model Results

The overall accuracy of the model was 48.59%. The resulting neural network is modeled as follows:

OPERATION		DATA DIMENSIONS	WEIGHTS(N)	WEIGHTS(%)
Input	#####	56		
Dense	XXXXX	----------	11856	16.3%
sigmoid	#####	208		
Dense	XXXXX	----------	60192	82.6%
sigmoid	#####	288		
Dense	XXXXX	----------	867	1.2%
softmax	#####	3		

Figure 6. Model Diagram

6.0 Hard Model

6.0 Overview:

To automate the process of identifying the risk rating of customers, unsupervised learning must be utilized. This is due to the fact that in practice, all customers will not be labeled based on risk levels and therefore a model must be created to help in the labeling.

6.1 Modeling Pipeline

To attempt the hard approach, we attempt to label the data in the big dataset using segmentation modeling and K-Means clustering. Due to the lack of labels in the big dataset, the training set was set up in a way that the input variables in the training set could be found in the big dataset.

6.2 K-Means Approach & Segmentation Modeling

Using the elbow method, the optimal K clusters of 10 was found..The elbow method graph is shown in Appendix E Figure E1.

After determining the optimal clusters, a combination of pipeline and GridSearchCV was used to determine the best model and parameters for each cluster. Every cluster was split into training and test sets with a 70/30 split. The output for the model is seen in the AppendixE Figure E3. The test accuracy, recall, AUC and F1 score were used as measures of performance to determine the optimal model.

Table E1 in Appendix E lists out the best model for each cluster in terms of test accuracy. Logistic Regression, SVM, Decision Tree, Random Forest, Gradient Boosting Classifier, KNN, and Neural Networks were utilized as potential models. The parameters selection is seen in the Appendix E Figure E3.

From the results, we notice that in general, the segmentation does lead to an increase in overall accuracy to 47.77% and an over 43.13% increase in accuracy over the naive model. This is higher than both the pipeline easy and medium models and provides interpretability as compared to the neural network models. Additionally, when it comes to the high risk clusters 0,7,9, the accuracy is 49.23%, which is higher than the medium neural network model. Finally, in terms of recall the model also performs better in terms of not failing to predict a rating of 3 than a global model. This is useful as the risk of not classifying someone as risk 3 is more costly than falsely classifying someone as risk level 3.

6.2.1 Cluster Findings

We found different clusters have some obvious differences (Figure F1-6). A summary table (Table 1) of all the characteristics of the 10 clusters is shown in the Appendix F. After comparing clusters' different characteristics and their different rating compositions in rating diagram (Figure F7 in Appendix F), we summarized features that are potentially related to high risk of money laundering. We list them by rank with representative clusters shown in the brackets.

- High percentage usage of visa (cluster 0,9)
- High percentage of credit cards (cluster 7,9)
- Self-employed (cluster 9)
- Joint accounts (cluster 7)
- High positive average net incoming amount (incoming amount minus outgoing amount) (cluster 7)
- High but inconsistent outcoming amount (cluster 9)
- High but inconsistent incoming amount (cluster 7)

We also list the features that are not related to high risk of money laundering, in other words, more related to low risk of money laundering.

- Active accounts (cluster 4)
- Retiree (cluster 4)
- Owner relationship type (cluster 6)

- High percentage of term deposits (cluster 6)

For other characteristics that have no association with high risk or low risk, they fall into the middle category. Also, the lists above are not absolutely conclusive as there are some violations (highlighted in red in Table 1) such as some clusters show a high percentage of credit cards, but they do not have the highest percentage of high-risk rating. These features essentially show possible risk trends but do not determine the classification of the ratings. But importantly, these characteristics can help us with the cluster labelling in the big dataset.

6.3 Validation Using Big Dataset

To validate that our segmentation models will to an extent lead to similar results in terms of label accuracy, we decided to determine if the clusters fitted in the training set would have the same characteristics in the big dataset. To accomplish this, we first fit the K-Means model in the training set and then predicted using the big dataset. The results are shown below.

Figure 7: Comparison of Labeled Dataset with Full Big Dataset

From the graphs above, we can see that within the big dataset, similar patterns can be found. For example cluster 1 and 10, demonstrate near similar distribution patterns for across both datasets.

As you can see, in general, the clusters created in the big dataset will lead to similar looking cluster characteristics as seen in the training set. We cannot validate that the labels made from our optimal models will be accurate, however we can validate that the clusters created are. This means that the optimal models created using the segmentation modeling approach would be valid to use for the big data. The underlying importance and distribution of data between the clusters of each set are similar. Due to technical constraints, more analysis on the big dataset could not be accomplished due to the size of the files.

7.0 Conclusion and Recommendations

After we have explored all three types of models, we have gained a significant number of insights and findings with regard to the Money Laundering individual characteristics. From the Easy Model, without including the transaction data, we are able to achieve a-lower test accuracy of 47.58%. One of the main reasons is omitted variable bias, where none of the transaction data is factored in the analysis. On the other hand, the Medium Model incorporated the transaction data with the aggregated statistics. This has provided a lot more information to the customer. However, at the same time, it has also introduced a lot of noise and outliers. The most significant one can be the maximum transaction value, which can be significantly separated from the rest of the data. As a result, the Neural network method is not able to achieve high accuracy for the prediction. The complex Neural Network lacks interpretability which fails to characterize the individual with Money Laundering.

In terms of the Hard Model, the segmentation is able to provide remarkable insights with 10 clusters as labelled in Section 6.2.1. The clustering performed on the training dataset shows a significant separation between different risk levels along with the essential characteristics. The application to big data also illustrates similar features, which can be used for classification. As indicated from Figure C7, clusters 0,7,9 have a potential high risk of Money Laundering activities, according to the provided labels. On the other hand, clusters 4,5,6 have a potential lower risk of Money Laundering, which can be classified as safe clusters. With the segmentation approach, the model accuracy increases for all the clusters. In particular, high-risk clusters 0,7,9 on average have above 50% accuracy in identifying the risk rating. It was observed that customers with high percentage usage of visa and credit cards, self-employed, high inconsistent transaction amounts, and the joint account could be classified as potentially risky customers who may be involved in money laundering. The segmentation model approach can help monitor customers that are deemed to be high risk. By dividing the big dataset into smaller clusters, potentially high risk customers can be flagged more accurately.

7.1 Next Steps & Limitations

Although the accuracy did improve by using our model, it could be beneficial to convert the target variables to binary instead. This would lead to accuracies in the range of 70-75%+. In this case, 1 would be Rating 3 and 0 would be not Rating 3. Therefore you can predict for high risk versus not high risk customers rather than three targets. Another way to improve the results would also be to validate our labels from our models, and potentially provide more computing power to explore the big dataset more. This would allow for even further insights and lead to better results.

8.0 Reference

"Negative Effects of Money Laundering on The Economy." *Sanction Scanner*, 2020, https://sanctionscanner.com/blog/negative-effects-of-money-laundering-on-the-economy-132#:~:text=Money%20laundering%20damages%20financial%20sector,real%20sector%20of%20the%20economy.&text=The%20effect%20of%20successfully%20clearing,%2C%20more%20crime%2C%20. Accessed 14 Mar 2021.

Punwasi, Stephen. "Canada Would Be In A Recession Without Money Laundering." *Better Dwelling*, 13 May 2019, https://betterdwelling.com/canada-would-be-in-a-recession-without-money-laundering/#:~:text=Over%20%24200%20Billion%20Money%20Laundered%20Across%20Canada%20Over%205%20Years&text=This%20represents%202.13%25%20of%20GDP,over%20the%20five%20year%20period. Accessed 13 Mar 2021.

For Coding:

Mayo, Matthew. "Managing Machine Learning Workflows with Scikit-learn Pipelines Part 3: Multiple Models, Pipelines, and Grid Searches." *KD Nuggets*, 2018, https://www.kdnuggets.com/2018/01/managing-machine-learning-workflows-scikit-learn-pipelines-part-3.html.

Scikit-Learn Open Source Package: https://scikit-learn.org/stable/

TensorFlow Keras Open Source Package: https://www.tensorflow.org/

9.0 Appendix:

Appendix A: Country Code Distribution

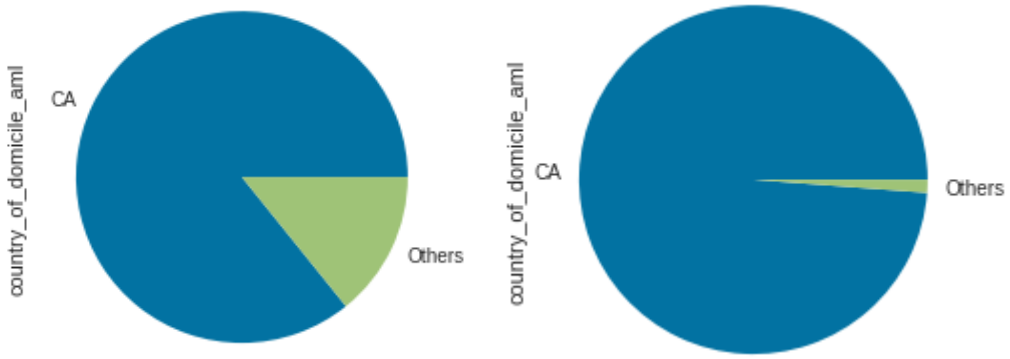


Figure A1: (Left) Country Distribution for Training (Right) Country Distribution for Testing

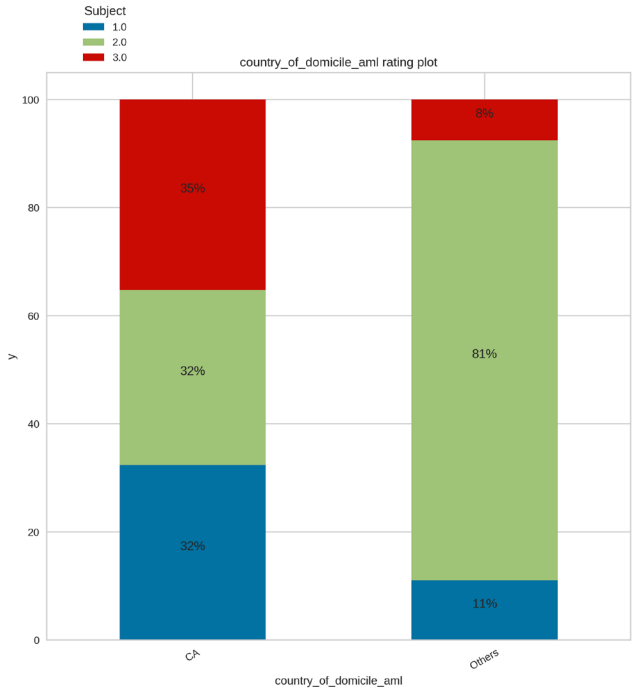
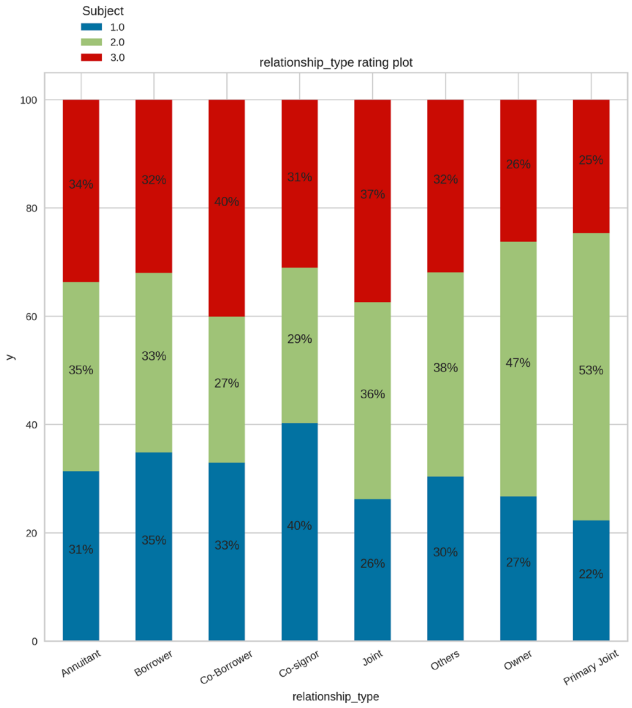


Figure A2: Crosstab Analysis for Country Code

Appendix B: Exploratory Analysis



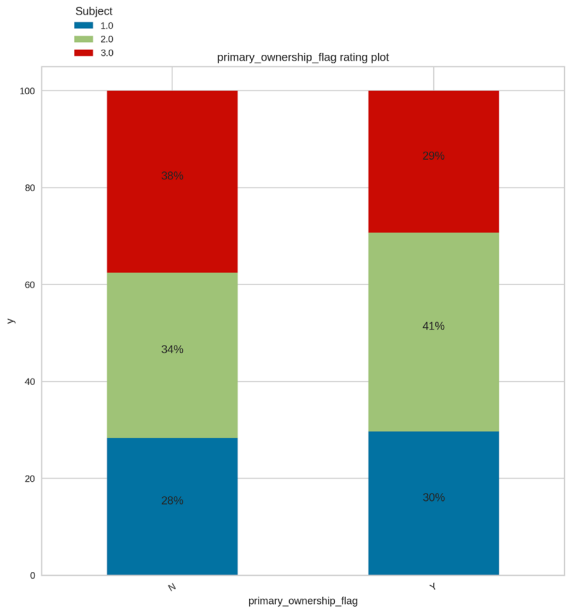


Figure B1: Crosstab Analysis for Account Type **Figure B2:** Crosstab Analysis for Primary Account

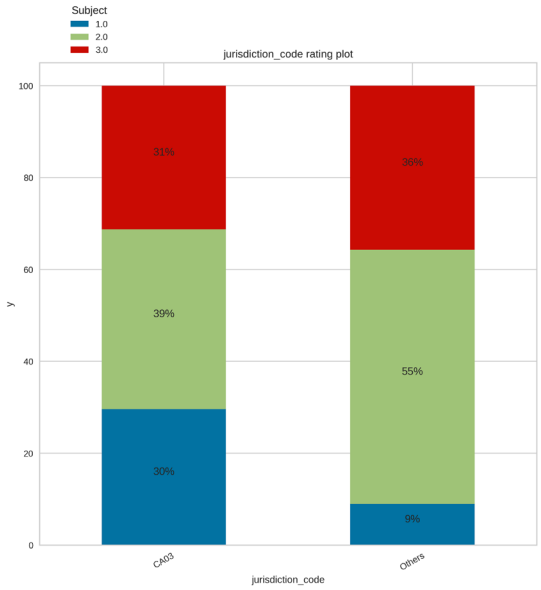


Figure B3: Crosstab Analysis for Jurisdiction Code

Appendix C: Result for Easy before feature selection:

```
Estimator: Gradient Boosting Classifier
Best params: {'clf__learning_rate': 0.1, 'clf__loss': 'deviance', 'clf__n_estimators': 100}
Best training accuracy: 0.451
Test set accuracy score for best params: 0.460
              precision    recall  f1-score   support

         1.0       0.36      0.34      0.35       371
         2.0       0.53      0.60      0.56       548
         3.0       0.44      0.38      0.41       422

    accuracy                           0.46      1341
   macro avg       0.44      0.44      0.44      1341
weighted avg       0.45      0.46      0.46      1341
```

	precision	recall	f1-score	support	pred	AUC
1.0	0.371831	0.355795	0.363636	371.0	355.0	0.609941
2.0	0.549669	0.605839	0.576389	548.0	604.0	0.674856
3.0	0.429319	0.388626	0.407960	422.0	382.0	0.639221
avg / total	0.462596	0.468307	0.464526	1341.0	1341.0	0.651864

Appendix D: Result for Medium before feature selection:

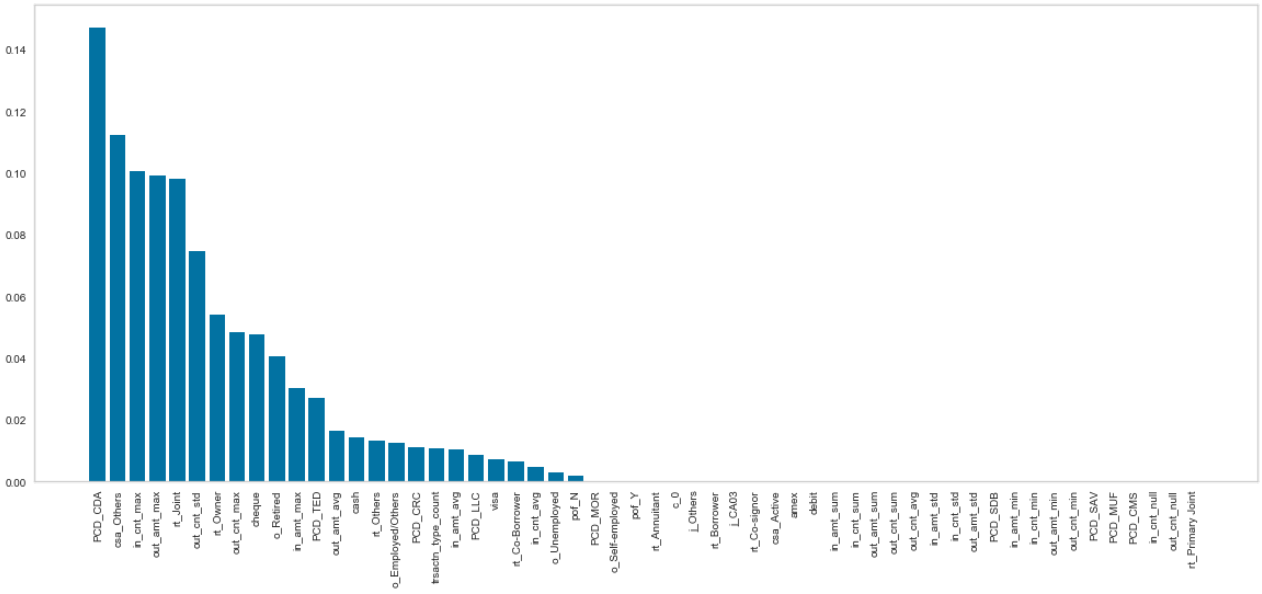


Figure D1: Decision Tree Feature Importance

	Predictor	Coefficients
38	j_Others	-1.211907e+00
45	csa_Others	-1.155181e+00
54	rt_Owner	-4.900006e-01
0	PCD_CDA	-3.799143e-01
43	o_Unemployed	-3.701852e-01
51	rt_Co-signor	3.541128e-01
39	c_0	-3.215311e-01
31	trsactn_type_count	-3.091671e-01
42	o_Self-employed	-2.925306e-01
48	rt_Annuitant	-1.927481e-01
55	rt_Primary Joint	-1.891245e-01
2	PCD_CRC	1.407186e-01
22	in_cnt_min	1.066918e-01
8	PCD_TED	9.156492e-02
41	o_Retired	7.737355e-02
49	rt_Borrower	-7.075788e-02
4	PCD_MOR	-7.024898e-02
46	pof_N	-6.558559e-02
24	out_cnt_min	-5.871862e-02
36	visa	5.303453e-02
6	PCD_SAV	4.952730e-02
35	debit	4.075089e-02
34	cheque	3.517524e-02
3	PCD_LLC	3.247072e-02
33	cash	3.208445e-02
14	in_cnt_avg	-2.107545e-02
30	out_cnt_null	2.015289e-02
16	out_cnt_avg	1.706504e-02
18	in_cnt_std	1.630816e-02
52	rt_Joint	1.501535e-02
20	out_cnt_std	-1.498699e-02
10	in_cnt_sum	-4.615618e-03
29	in_cnt_null	-4.344357e-03
32	amex	-4.142899e-03
50	rt_Co-Borrower	3.408981e-03
28	out_cnt_max	-1.307428e-03
12	out_cnt_sum	-2.487149e-04
15	out_amt_avg	2.995869e-05
23	out_amt_min	-2.704371e-05
19	out_amt_std	-1.835016e-05
17	in_amt_std	-1.262050e-05
13	in_amt_avg	9.480494e-06
25	in_amt_max	3.706474e-06
21	in_amt_min	-2.574449e-06
27	out_amt_max	1.726694e-06
9	in_amt_sum	-8.511916e-07
11	out_amt_sum	-4.903434e-07

Figure D2: Feature Selection with Lasso Regression

```
Estimator: Gradient Boosting Classifier
Best params: {'clf__learning_rate': 0.1, 'clf__loss': 'deviance', 'clf__n_estimators': 100}
Best training accuracy: 0.461
Test set accuracy score for best params: 0.456
              precision    recall  f1-score   support

         1.0       0.40      0.32      0.36       545
         2.0       0.50      0.61      0.55       684
         3.0       0.44      0.39      0.41       559

    accuracy                           0.46      1788
   macro avg       0.44      0.44      0.44      1788
weighted avg       0.45      0.46      0.45      1788
```

	precision	recall	f1-score	support	pred	AUC
1.0	0.392936	0.326606	0.356713	545.0	453.0	0.610482
2.0	0.504294	0.600877	0.548366	684.0	815.0	0.671934
3.0	0.432692	0.402504	0.417053	559.0	520.0	0.640295
avg / total	0.447966	0.455257	0.448895	1788.0	1788.0	0.646521

Appendix E: Hard Model Results

Table E1. Test accuracy results for each model for each cluster

Clusters	Logistic Regression	Decision Tree	KNN	SVC	Gradient Boosting Classifier	Random Forest	ANN	N
Cluster 0	0.373	0.396	0.354	0.377	0.420	0.448	0.260	212
Cluster 1	0.409	0.337	0.346	0.418	0.447	0.418	0.327	208
Cluster 2	0.385	0.385	0.367	0.402	0.385	0.408	0.367	169
Cluster 3	0.510	0.385	0.385	0.471	0.500	0.529	0.5	104
Cluster 4	0.494	0.481	0.449	0.442	0.455	0.481	0.461	156
Cluster 5	0.485	0.466	0.450	0.504	0.454	0.466	0.328	262
Cluster 6	0.386	0.400	0.386	0.386	0.377	0.400	0.386	215
Cluster 7	0.511	0.412	0.473	0.489	0.451	0.533	0.407	182
Cluster 8	0.510	0.524	0.538	0.531	0.531	0.545	0.4	145
Cluster 9	0.449	0.413	0.406	0.507	0.406	0.478	0.442	138

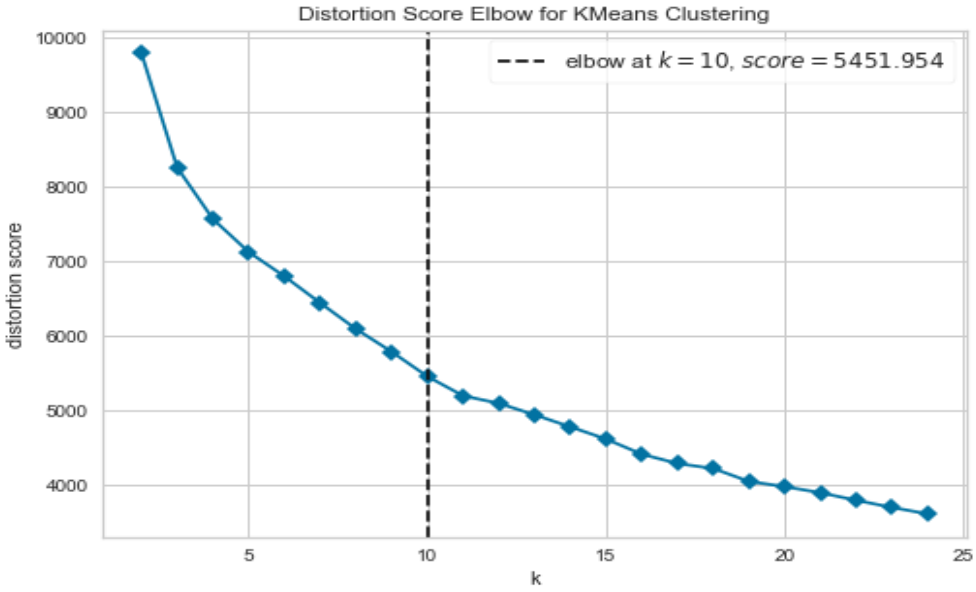


Figure E1. Elbow Method to find optimal K

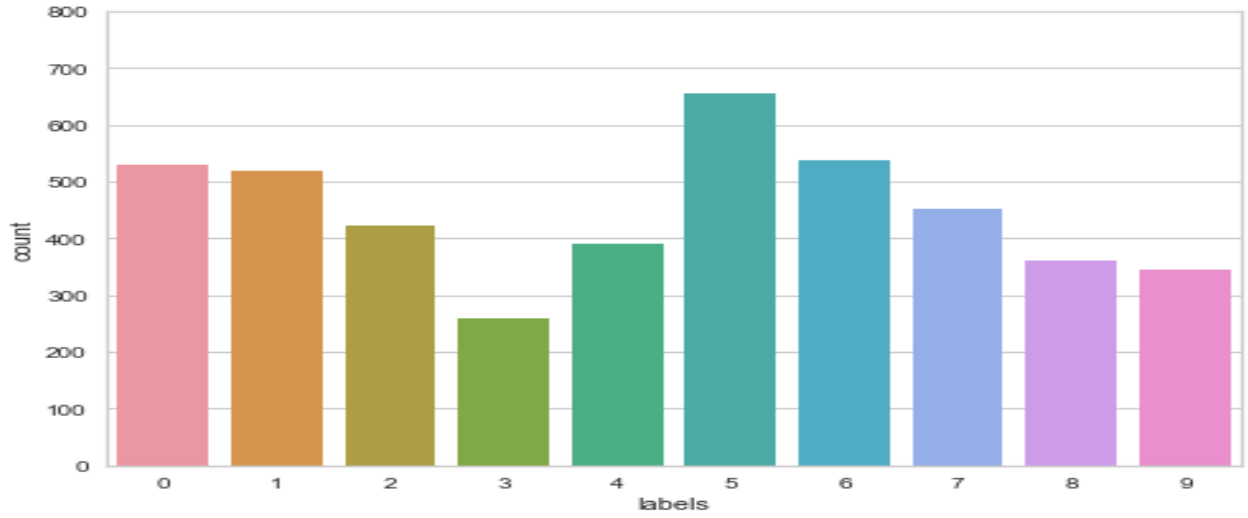


Figure E2: Parameter Tuning and Results for Hard Model

Cluster 1

```
Estimator: Random Forest
Best params: {'clf__criterion': 'gini', 'clf__max_depth': 10, 'clf__min_impurity_decrease': 0.01, 'clf__min_samples_split': 40}
Best training accuracy: 0.365
Test set accuracy score for best params: 0.448
              precision    recall  f1-score   support

         1.0       0.29      0.03      0.06        62
         2.0       0.50      0.05      0.10        55
         3.0       0.45      0.95      0.61        95

    accuracy                           0.45       212
   macro avg       0.41      0.34      0.26       212
weighted avg       0.42      0.45      0.32       212


Classifier with best test set accuracy: Random Forest
```

Cluster 2

```
Estimator: Gradient Boosting Classifier
Best params: {'clf__learning_rate': 1.0, 'clf__loss': 'deviance', 'clf__n_estimators': 300}
Best training accuracy: 0.410
Test set accuracy score for best params: 0.447
              precision    recall  f1-score   support

         1.0       0.39      0.46      0.42        61
         2.0       0.46      0.44      0.45        68
         3.0       0.49      0.44      0.46        79

    accuracy                           0.45       208
   macro avg       0.45      0.45      0.45       208
weighted avg       0.45      0.45      0.45       208
```

Cluster 3

```
Estimator: Random Forest
Best params: {'clf__criterion': 'gini', 'clf__max_depth': 20, 'clf__min_impurity_decrease': 0.01, 'clf__min_samples_split': 20}
Best training accuracy: 0.413
Test set accuracy score for best params: 0.408
              precision    recall  f1-score   support

         1.0       0.28      0.17      0.21        41
         2.0       0.47      0.63      0.54        62
         3.0       0.38      0.35      0.36        66

    accuracy                           0.41       169
   macro avg       0.38      0.38      0.37       169
weighted avg       0.39      0.41      0.39       169


Classifier with best test set accuracy: Random Forest
```

Cluster 4

```
Estimator: Random Forest
Best params: {'clf__criterion': 'gini', 'clf__max_depth': 10, 'clf__min_impurity_decrease': 0.01, 'clf__min_samples_split': 20}
Best training accuracy: 0.488
Test set accuracy score for best params: 0.529
              precision    recall  f1-score   support

         1.0       0.29      0.16      0.21        25
         2.0       0.58      0.81      0.68        52
         3.0       0.50      0.33      0.40        27

    accuracy                           0.53       104
   macro avg       0.46      0.43      0.43       104
weighted avg       0.49      0.53      0.49       104


Classifier with best test set accuracy: Random Forest
```

Cluster 5

```
Estimator: Logistic Regression
Best params: {'clf__C': 1.0, 'clf__penalty': 'l2', 'clf__solver': 'liblinear'}
Best training accuracy: 0.538
Test set accuracy score for best params: 0.494
              precision    recall  f1-score   support

         1.0       0.45      0.24      0.31        54
         2.0       0.51      0.86      0.64        72
         3.0       0.40      0.07      0.11        30

    accuracy                           0.49       156
   macro avg       0.45      0.39      0.36       156
weighted avg       0.47      0.49      0.43       156
```

Cluster 6

```
Estimator: SVC
Best params: {'clf__C': 1, 'clf__kernel': 'linear'}
Best training accuracy: 0.473
Test set accuracy score for best params: 0.504
              precision    recall  f1-score   support

         1.0       0.45      0.87      0.59        90
         2.0       0.61      0.53      0.57        86
         3.0       0.57      0.09      0.16        86

    accuracy                           0.50       262
   macro avg       0.55      0.50      0.44       262
weighted avg       0.54      0.50      0.44       262
```

Cluster 7

```
Estimator: Decision Tree
C:\Users\buttb\Anaconda3\lib\site-packages\sklearn\metrics\_classification.py:1221: UndefinedMetricWarning: Precision and F-sco
re are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this beha
vior.
  _warn_prf(average, modifier, msg_start, len(result))
Best params: {'clf__criterion': 'gini', 'clf__max_depth': 10, 'clf__min_impurity_decrease': 0.01, 'clf__min_samples_split': 20}
Best training accuracy: 0.410
Test set accuracy score for best params: 0.400
              precision    recall  f1-score   support

         1.0       0.36      0.14      0.20        64
         2.0       0.44      0.58      0.50        83
         3.0       0.36      0.43      0.39        68

    accuracy                           0.40       215
   macro avg       0.39      0.38      0.36       215
weighted avg       0.39      0.40      0.38       215
```

Cluster 8

```
Estimator: Random Forest
Best params: {'clf__criterion': 'gini', 'clf__max_depth': 20, 'clf__min_impurity_decrease': 0.001, 'clf__min_samples_split': 6
0}
Best training accuracy: 0.509
Test set accuracy score for best params: 0.533
              precision    recall  f1-score   support

         1.0       0.00      0.00      0.00        40
         2.0       0.50      0.74      0.60        74
         3.0       0.58      0.62      0.60        68

    accuracy                           0.53       182
   macro avg       0.36      0.45      0.40       182
weighted avg       0.42      0.53      0.47       182
```

Cluster 9

```
Estimator: Random Forest
Best params: {'clf__criterion': 'gini', 'clf__max_depth': 10, 'clf__min_impurity_decrease': 0.005, 'clf__min_samples_split': 6
0}
Best training accuracy: 0.539
Test set accuracy score for best params: 0.545
              precision    recall  f1-score   support

         1.0       0.46      0.55      0.50        38
         2.0       0.56      0.86      0.68        58
         3.0       0.89      0.16      0.28        49

    accuracy                           0.54       145
   macro avg       0.63      0.53      0.48       145
weighted avg       0.64      0.54      0.49       145
```

Cluster 10

```
Estimator: SVC
Best params: {'clf__C': 10, 'clf__kernel': 'rbf'}
Best training accuracy: 0.513
Test set accuracy score for best params: 0.507
              precision    recall  f1-score   support

         1.0       0.40      0.25      0.31        32
         2.0       0.55      0.67      0.61        61
         3.0       0.48      0.47      0.47        45

    accuracy                           0.51       138
   macro avg       0.48      0.46      0.46       138
weighted avg       0.49      0.51      0.49       138
```

Appendix F: Cluster Characteristics:

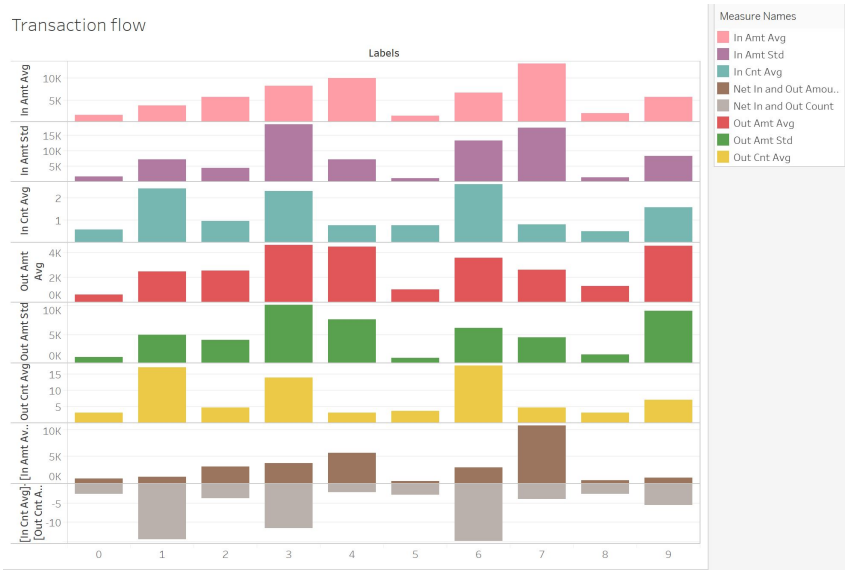


Figure F1: Transaction Flow Diagram

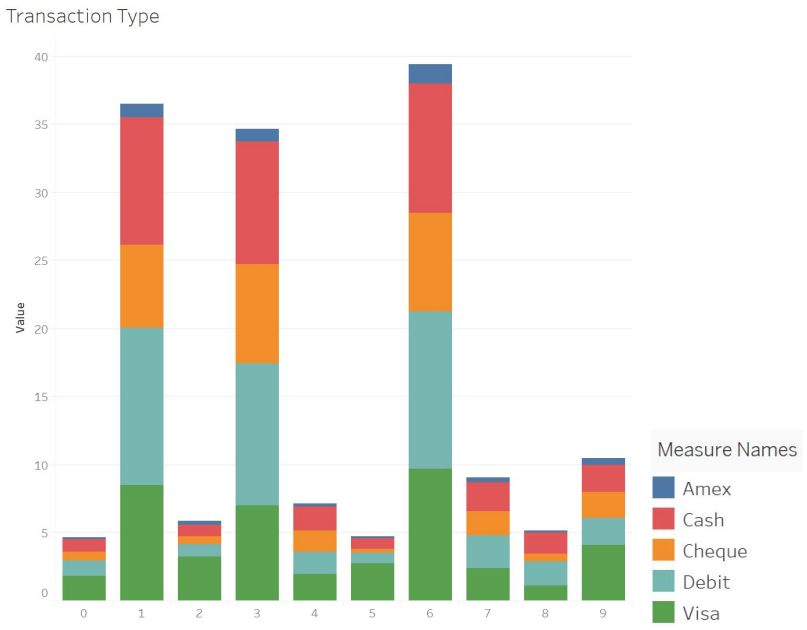


Figure F2: Transaction Type Diagram

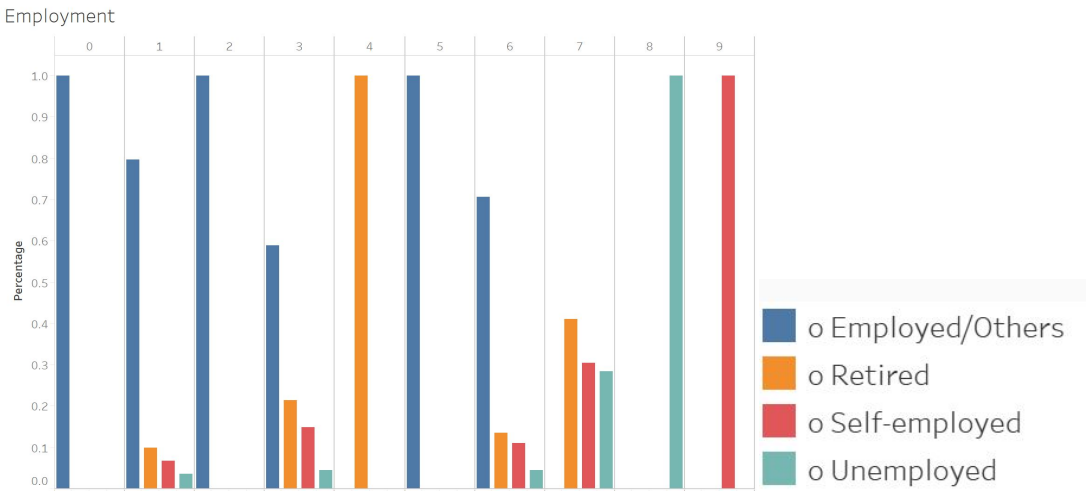


Figure F3: Employment Diagram

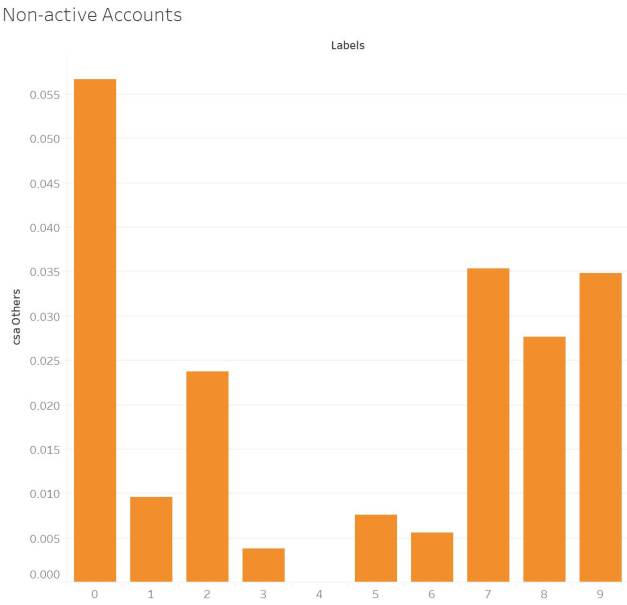


Figure F4: Non-active Accounts Diagram

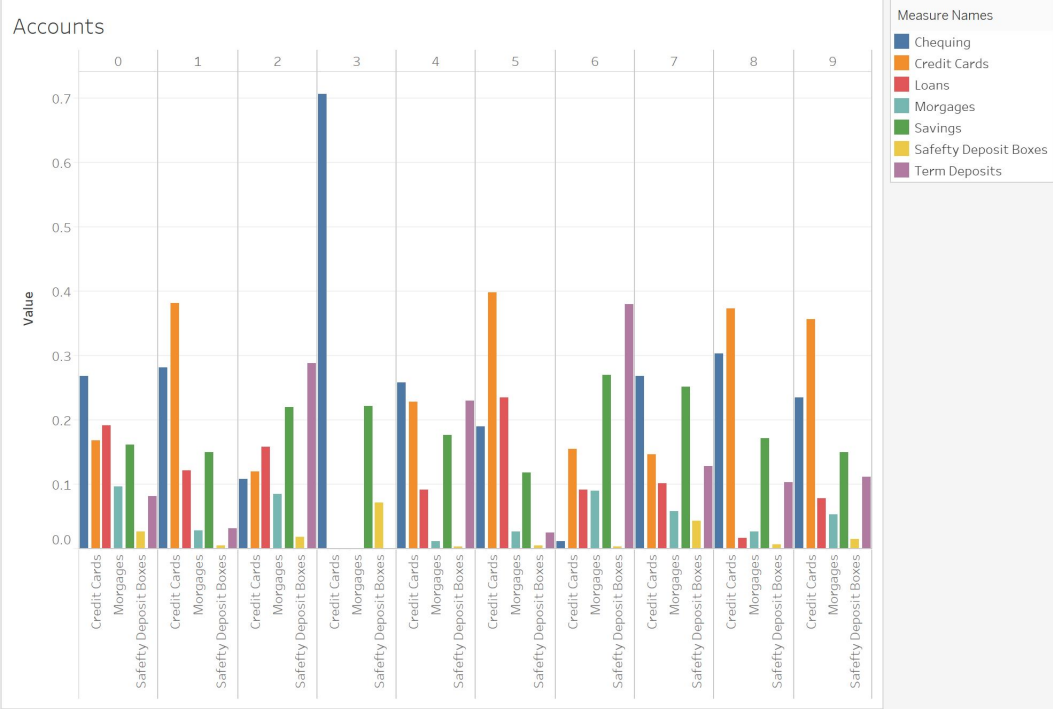


Figure F5: Account Type Diagram

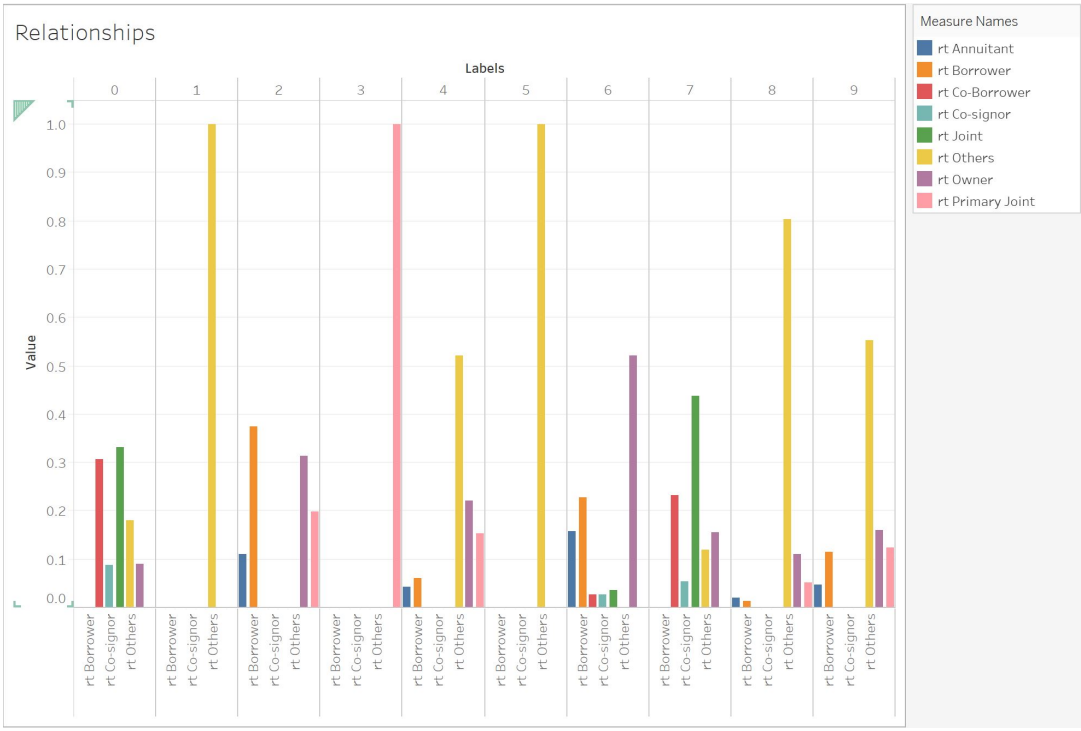


Figure F6: Relationship Type Diagram

Table 1: Cluster Characteristics Summary

Cluster	0	1	2	3	4	5	6	7	8	9
Transaction flow	Low average and std of in and out amount	High in and out counts	Positive in amount	High in and out counts	Low in and out counts	Low average and std of in and out amount	High in and out counts	High but inconsistent in amount	Low average and std of in and out amount	High and inconsistent out amount
	Low in and out counts	Low in and out amount		High average and std in and out amount	High in and out amount	Low in and out counts	Positive in amount	Low in and out counts	Low in and out counts	Neutral in and out amount
	Neutral in and out amount	Neutral in and out amount		Positive in amount	Positive in amount	Neutral in and out amount	More out counts	High Positive in amount	Neutral in and out amount	
		More out counts		More out counts						
Employment	Only have employed		Only have employed		Only have Retired	Only have employed			Only have unemployed	Only have self-employed
Transaction Type	High Percentage of Visa		High Percentage of Visa		Averaged	High Percentage of Visa			Relatively high in Cash and Debit	High Percentage of Visa
			Low in cheques			Low in cheques				
Accounts		High credit cards		High chequing		High credit cards	High term deposits		High credit cards	High credit cards
Relationships		Only Others	Borrowers/Owners	Only Primary Joint		Only Others	Owner	Joints	Others	Others
Customer Status	Mostly Inactive				All active					

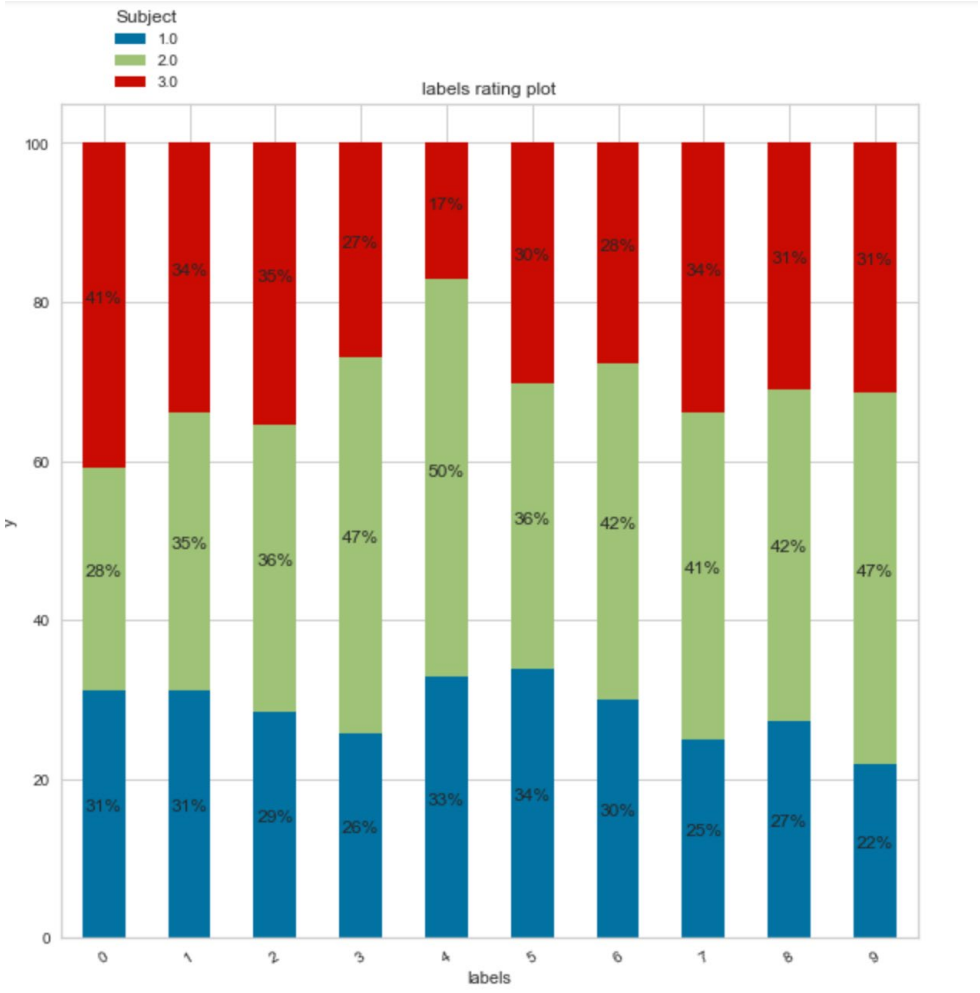


Figure E7: Cluster Rating Composition