

MIS 790: SEMINAR In Business Analytics

Customer Segmentation Using K Means Model



May 7, 2023

Bijay KC

Option 3: Credit\_Card

Data Preparation

Data Exploration

The preferred language for this case study was Python. This language comes in handy due to its libraries which support machine learning and artificial intelligence projects apart from the usual statistical analysis work. This study was used to analyze the dataset's structural components to get it ready for the clustering algorithms. Descriptive statistics and data exploration are the preprocessing phases of any analysis. The dataset analyzed contained seventeen variables; excluding Cust\_ID, all of which were numerical. There was a total of 8950 samples, with 314 rows containing missing values. The descriptive statistics of each numerical variable are presented in the table below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable Name** | **Mean** | **Standard Deviation** | **Minimum** | **Maximum** |
| **BALANCE** | 1,564.47 | 2,081.42 | 0.00 | 19,043.14 |
| **BALANCE\_FREQUENCY** | 0.88 | 0.24 | 0.00 | 1.00 |
| **PURCHASES** | 1,003.20 | 2,136.52 | 0.00 | 49,039.57 |
| **ONEOFF\_PURCHASES** | 592.44 | 1,659.80 | 0.00 | 40,761.25 |
| **INSTALLMENTS\_PURCHASES** | 411.07 | 904.29 | 0.00 | 22,500.00 |
| **CASH\_ADVANCE** | 978.87 | 2,097.05 | 0.00 | 47,137.21 |
| **PURCHASES\_FREQUENCY** | 0.49 | 0.40 | 0.00 | 1.00 |
| **ONEOFF\_PURCHASES\_FREQUENCY** | 0.20 | 0.30 | 0.00 | 1.00 |
| **PURCHASES\_INSTALLMENTS\_FREQUENCY** | 0.36 | 0.40 | 0.00 | 1.00 |
| **CASH\_ADVANCE\_FREQUENCY** | 0.14 | 0.20 | 0.00 | 1.50 |
| **CASH\_ADVANCE\_TRX** | 3.25 | 6.82 | 0.00 | 123.00 |
| **PURCHASES\_TRX** | 14.71 | 24.86 | 0.00 | 358.00 |
| **CREDIT\_LIMIT** | 4,494.45 | 3,638.61 | 50.00 | 30,000.00 |
| **PAYMENTS** | 1,733.14 | 2,894.90 | 0.00 | 50,721.48 |
| **MINIMUM\_PAYMENTS** | 864.21 | 2,372.31 | 0.02 | 76,406.21 |
| **PRC\_FULL\_PAYMENT** | 0.15 | 0.29 | 0.00 | 1.00 |
| **TENURE** | 11.52 | 1.34 | 6.00 | 12.00 |

Missing Values and Solution to it

There were two variables with missing values. This might have occurred because the values were not recorded in the dataset initially for some reason or during the data entry phase they were omitted. Also, some of the null values might be due to some of the participants having lost their credit cards hence no data was recorded.

The null values were dropped in this case study. This was done to ensure the quality and accuracy of the analysis. The imputation method could have been used to fill all the rows with the missing values, but it was not used as the number of missing values was high.

Used variables for the K Means model

In creating the vector of features, only ten variables were selected for consideration. These variables, namely 'BALANCE', 'PURCHASES', 'CASH\_ADVANCE', 'PURCHASES\_FREQUENCY', 'CASH\_ADVANCE\_FREQUENCY', 'CASH\_ADVANCE\_TRX', 'PURCHASES\_TRX', 'CREDIT\_LIMIT', 'PAYMENTS', and 'MINIMUM\_PAYMENTS', were chosen because they are the most relevant variables that are likely to have an impact on the credit card usage behavior of the customers.

To evaluate the significance of these features in making accurate predictions, the Random Forest Model was employed. This model generates important scores that represent the relative contribution of each feature in predicting credit card usage behavior. These scores are then used to identify the most significant features that will be used in the K Means model.

Using the top 10 features based on their importance scores, the accuracy and efficiency of the K Means model were improved. Moreover, these variables are essential in the Credit Card dataset because they provide valuable information about customer spending, borrowing, and payment behavior. The inclusion of these columns can help identify patterns in customer behavior and provide insights into their overall financial behavior.

Standardization of the features

Before feeding the features into any machine learning model, one must always make sure that the features being used are on the same scale. In this case, the standardization method was the appropriate one to use. Standardizing features ensures that every feature is equally significant in determining the model's output and that they are on the same scale. This is particularly important when employing distance-based algorithms such as K-means, as the similarity between the samples is calculated using the distance between their features that is the Euclidean distance.

In PySpark, one can standardize features by utilizing the StandardScaler function. This function scales the features to have a mean of zero and a variance of one, thus standardizing them. Through standardizing the features, they become normally distributed hence outputting statistically significant results. It accepts the input column with the features and generates a new column with the standardized features. This approach ensures that the features are comparable and equally weighted in the model's predictions.

Model Evaluation and Improvement

When it comes to determining the optimum number of clusters, we need to consider both the WSSE and silhouette coefficient scores. The WSSE score measures the sum of squared errors within each cluster, while the silhouette coefficient score measures the similarity between clusters relative to their points. An ideal number of clusters will have a low WSSE score and a high silhouette coefficient score.

A picture containing text, diagram, line, plot

Description automatically generatedA picture containing text, line, diagram, plot

Description automatically generatedAfter analyzing the WSSE scores, a steady decrease was observed as the number of clusters increased. However, the rate of decrease slows down after k=3, suggesting that further increasing the number of clusters might not significantly improve the clustering performance.

On the other hand, the silhouette coefficient scores indicate that the highest score was obtained for k=2, followed by k=3. However, negative scores for k=5, k=6, k=7, k=8, k=9, and k=10 indicate that clustering is not appropriate for these values of k as shown above.

Therefore, based on the WSSE and silhouette coefficient scores, we can conclude that the optimum number of clusters is k=3

Improvement:

In this study, I evaluated the performance of the KMeans clustering algorithm using two different models. The first model removed all the missing values rows, while the second model replaced the missing values with the mean values. The results of the study showed that the model with three clusters and imputed missing values with mean values performed better than the model that removed the missing values.

This finding suggests that missing data imputation using the mean values did not negatively impact the performance of the KMeans algorithm. In fact, it improved the clustering results by providing a more complete dataset for analysis. This approach can be particularly useful when dealing with large datasets where the presence of missing values is common.

A picture containing screenshot, text, diagram, plot

Description automatically generatedSummary of the finding

The above graph helps us to describe each of the clusters. A summary is provided below.

### Cluster 1:

In the realm of credit card clusters, this particular assemblage stands out with the lowest of low balances, purchase amounts, and cash advances. Additionally, the inhabitants of this cluster exhibit a markedly deficient purchase frequency and cash advance frequency. With lower credit limits, these customers naturally conduct fewer transactions overall. However, an intriguing facet of their behavior is the conspicuous relative uptick in the frequency of purchasing in installments.

### Cluster 2:

Cluster 2 boasts the most substantial balance amounts among the different clusters, coupled with a relatively high incidence of cash advances. In addition, customers in this cluster tend to make a greater number of purchases when compared to those in Cluster 1. These purchases tend to be more one-off in nature, but their frequency of installment purchases remains relatively low. Overall, this cluster exhibits a unique spending pattern that combines both significant spending power and a preference for short-term purchasing.

### Cluster 3:

Among the analyzed clusters, this particular group boasts the most impressive purchase amounts and transaction frequencies, alongside a notable inclination towards purchasing in installments. Although their credit limits and balance amounts fall somewhere between those of the first and second clusters, they exhibit a lower tendency towards cash advances. Overall, this cluster's spending habits suggest a financially responsible yet active consumer group.

Top of Form

Conclusion

Issues and challenges

In this project, managing missing values posed a significant challenge. Although deleting rows with missing values could have been a solution, it would have resulted in significant data loss. Another alternative, MICE imputation, could have been appropriate, but with roughly 314 missing values, it would have been computationally expensive and time-consuming. Replacing the missing values with the means was also considered.

Selecting the ideal number of clusters in this project was challenging, as the elbow and silhouette curves provided different results. The elbow curve showed a significant drop in the within-cluster sum of squares error (WSSSE) from K=2 to K=3 and then leveled off, suggesting that K=3 was optimal. On the other hand, the Silhouette curve revealed a high score at K=2, followed by K=3, with a significant decrease from K=3 to K =4, implying that K =2 or K=3 could be used. As a result, determining the appropriate number of clusters was difficult since both methods gave different outcomes.

Future Improvements

1. Removing the outliers: In K-means clustering, outliers can significantly impact the results by skewing the clusters around them. Removing outliers can improve the accuracy of the model by ensuring that the clusters are representative of the majority of the data points and reducing the influence of extreme values. Outliers can skew the distribution of data and create clusters that are not representative of the majority of the data points. Removing outliers allows the K means algorithm to focus on creating clusters that are more representative and improve the overall accuracy of the model. However, it is important to be careful when removing outliers as it can lead to loss of information and should be done thoughtfully. This is especially important if the outliers are meaningful or represent a specific pattern or behavior in the data.