Bank Customers Credit Cards - Clustering

Homework 5 Report - Ujwal Joshi

April 2024

Introduction and Dataset

The dataset being used for cluster analysis contains information about the usage behavior of about 9000 active credit card holders over a period lasting 6 months. The data contains exactly 8950 rows of information, and has 18 features including an ID to identify each row. The other 17 features relate to:

- The customers balance and how frequently that balance is updated
- Installment purchases, purchases, one off purchases and their frequencies and number of purchase transactions
- Cash advance, their frequencies and the number of cash advance transactions
- Payments, minimum payments made by the user and the percent of full payment paid by the user
- The credit limit and tenure of service for that credit card

Using these features the aim is to train clustering models and compare their performance.

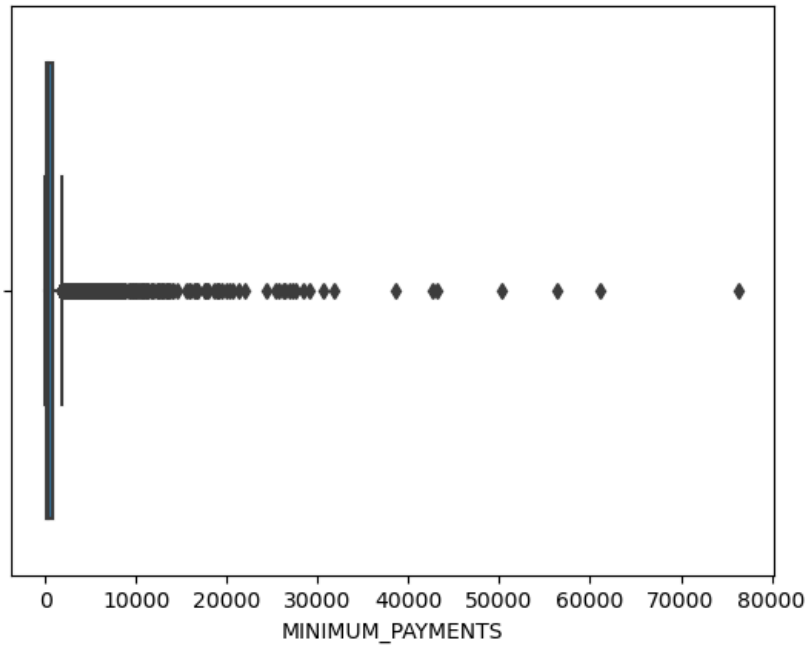
Data Processing and Investigation

Null Values

Investigation of the data reveals that two features, CREDIT_LIMIT and MINIMUM_PAYMENTS, contain empty or null values. However, the credit limit

feature has only one value, and since that one feature may not significantly impact the overall cluster analysis of the other 8949 rows of data, it can be safely excluded from the dataset.

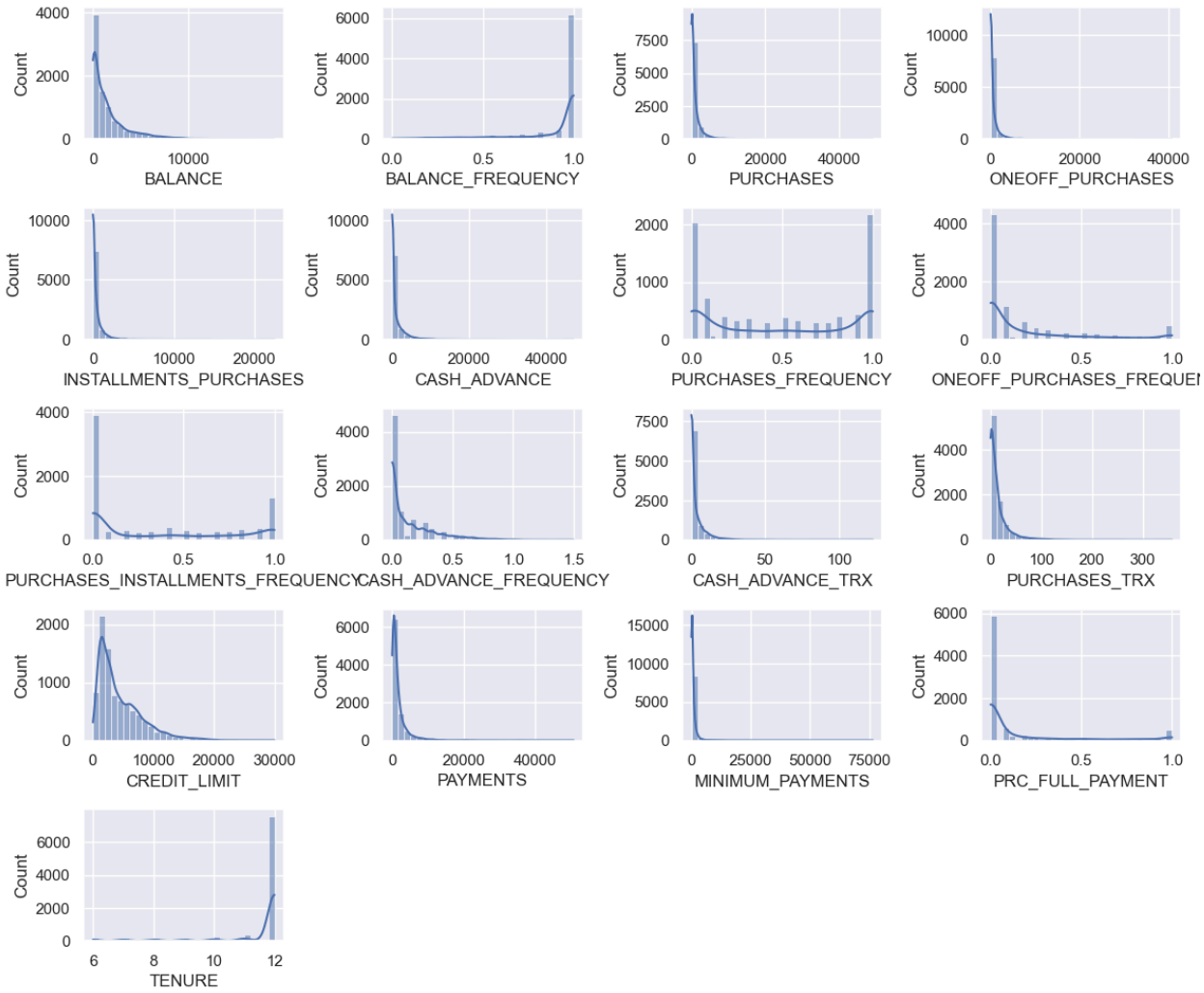
The minimum payments feature has 313 missing values and therefore cannot be completely ignored. Analysis of this feature reveals that it has a rather large standard deviation of 2372.57 with an extremely small minimum and a very large maximum value (0.019, 76406.21). Plotting this feature on a box plot confirms that it has lots of outliers.



To keep the feature robust to these outliers and preserve the overall distribution of the data the missing values will be imputed with the median.

Investigation: Distribution of Data

Insights into this data and an intuition of how the models may perform can be gained by examining the distribution of all the features:

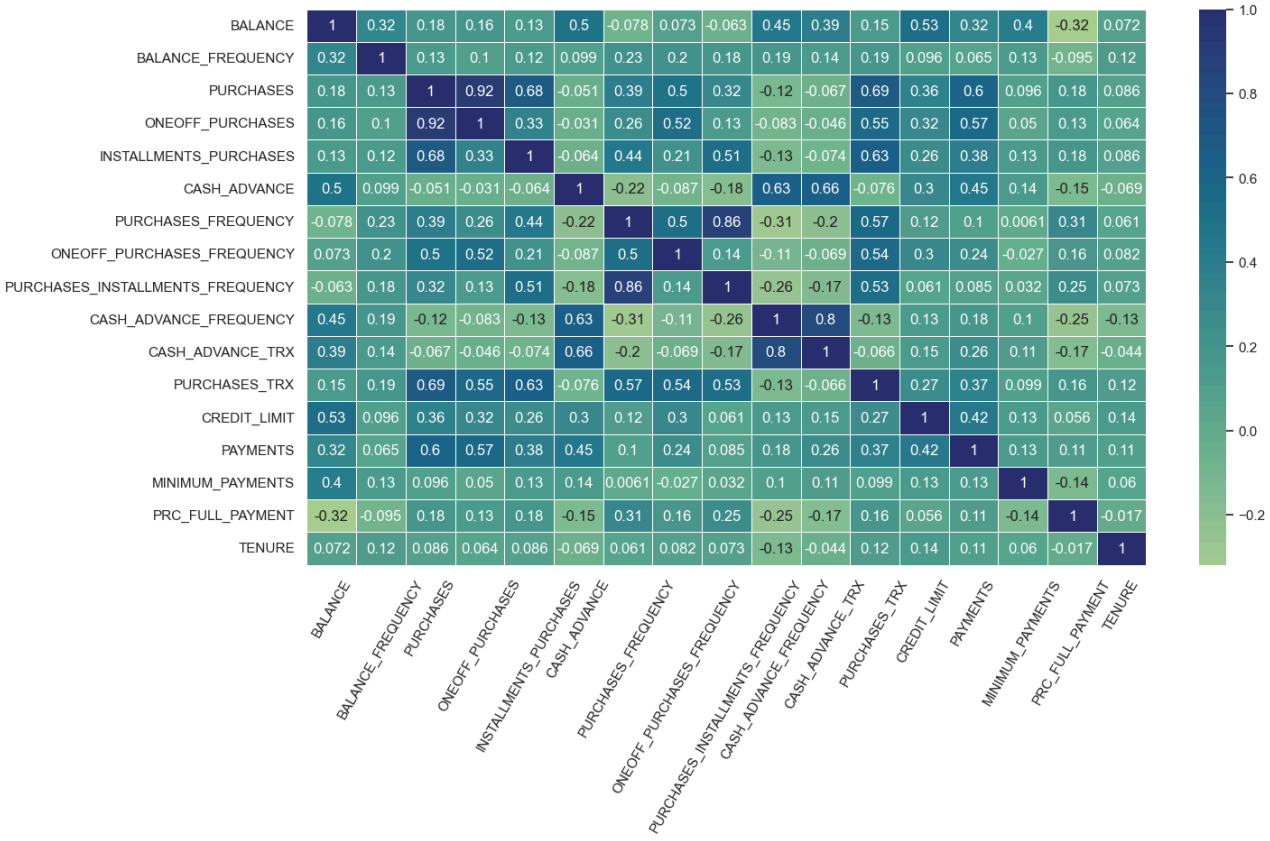


The distribution plots indicate that nearly all of the features exhibit a right skew which indicates that for these features most customers have lower values. This means that a large concentration of customers in this dataset have lower balances, purchase amounts, transactions and attributes related to those.

The data for the purchase frequency however is an exception and has a more even spread of the data, while the balance frequency and tenure features have a more left skewed distribution. This indicates that a relatively large concentration of customers update their balance more frequently and have a high tenure of credit card service.

Investigation: Correlation

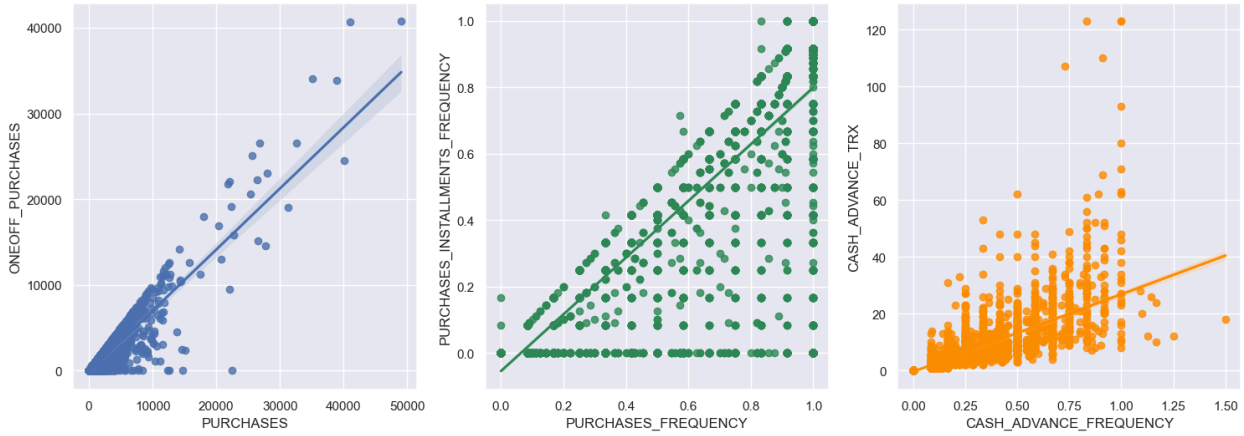
Correlations among features of the dataset can be important and allow us to remove highly correlated features to ensure stability of feature importance. The correlation matrix among features can be viewed with a correlation heatmap of the data:



While several features have some moderate correlation the following features exhibit a meaningful strong correlation:

- Purchases and One Off Purchases
- Purchases Frequencies and Purchases Installments Frequencies
- Cash Advance Frequencies and Cash Advance Transactions

The correlation can be further seen on scatter plots of these features:



Insights gained from these correlations indicate that customers who make purchases tend to also make significant one-time transactions. The meaningful correlation between purchase frequency and purchase installment frequency indicates that customers generally seem to have a preference regarding the timing of their purchases. Finally the correlation between cash advance frequency and cash advance transactions also provides insight into customer preference regarding the nature of their transactions.

Feature Selection

Not all features are needed for the model, firstly the Customer ID feature is removed from consideration as it only gives a unique identifier to each row and has no further inherent meaningful value or insight about that customer's data. Additionally, the highly correlated features: One Off Purchases, Purchase Installments Frequency, and Cash Advance Frequency are also excluded to prevent redundant information being used in the clustering models.

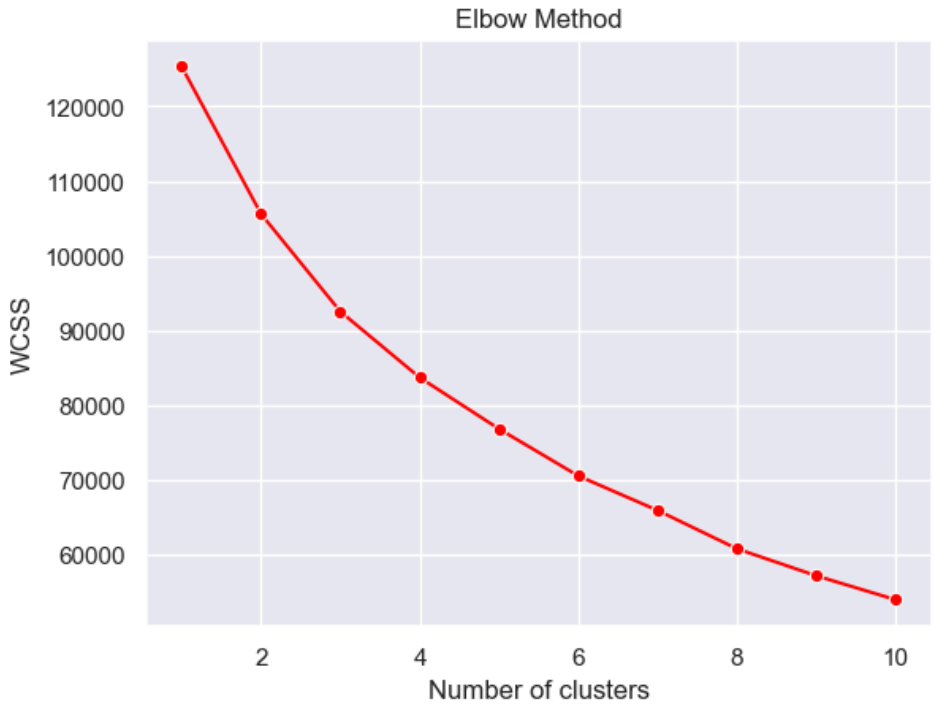
Data Scaling

To ensure that features with different scales don't disproportionately influence the model's fitting process the data was scaled with the standard scaler.

K-Means Clustering

K-Means clustering is an unsupervised algorithm that partitions a dataset into K clusters based on the similarity of data points. It iteratively assigns data points to the nearest cluster centroid and updates the centroids until convergence.

The parameter needed for this model is the K, the number of clusters, to find an optimal value for this parameter the elbow method can be used:



Based on the plot, a K value of 3 seems optimal to try and fit the dataset with this model.

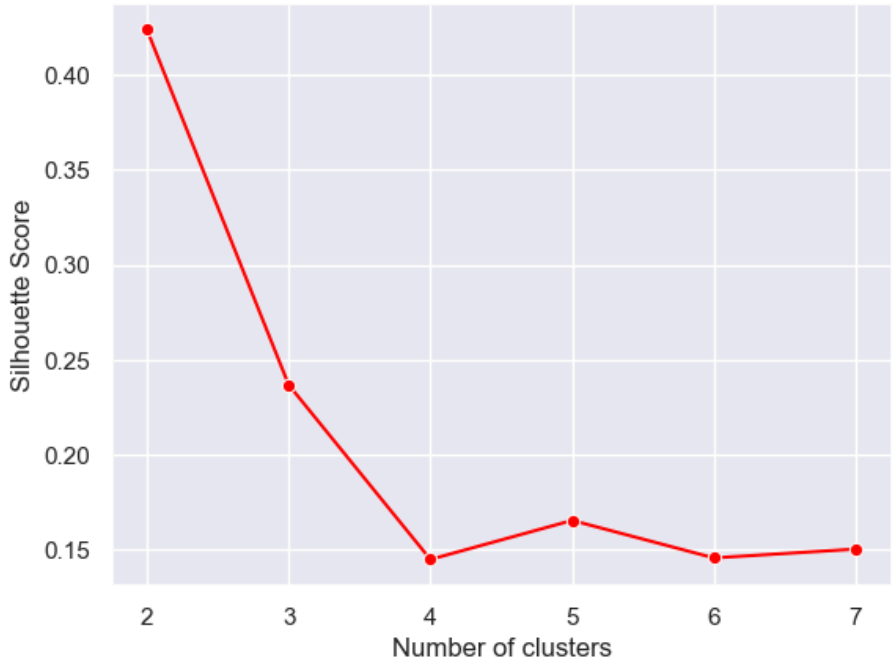
The silhouette score can be used to judge the performance of the model. This score is a metric used to evaluate the quality of clusters in unsupervised learning. It measures how similar an object is to its own cluster compared to other clusters.

The silhouette score for this K Means clustering model was roughly 0.2798 indicates that the clusters are poorly defined and data points may not necessarily be defined to the right cluster.

K-Means clustering is computationally efficient and easy to implement, making it suitable for large datasets and exploratory data analysis. However, K-Means is sensitive to the initial selection of cluster centroids and is not suitable for clusters of irregular shapes or varying densities. Further choosing a predetermined number of clusters which is not necessarily simple in determining accurately.

Hierarchical Clustering

Hierarchical clustering organizes data into a hierarchy of clusters, represented as a dendrogram. The optimal number of clusters can be determined by comparing the silhouette scores of models using differing cluster numbers:



Two clusters have the highest silhouette scores, the score output for this hierarchical clustering model which uses 2 clusters is about 0.4237.
While this is an improvement over the K-Means clustering model the score below 0.5 still indicates that not all the clusters are well defined and data points may still be assigned to clusters which may not be appropriate for them.

Hierarchical clustering does not require a predefined number of clusters and can handle various shapes and sizes of clusters, offering flexibility in cluster exploration. This model can also provide an informative visual representation of the clusters via a dendrogram. However, hierarchical clustering can be computationally expensive for large datasets and is sensitive to noise and outliers.

DBSCAN

DBSCAN is a density-based clustering algorithm that groups together points that are closely packed, separating regions of high density from regions of low density. This model requires the use of an epsilon and min_samples parameter to determine the dense regions needed for clustering. Optimal values for this parameter can be found using a form of grid search to try out different options and combinations for these parameters and choosing the combination with the best silhouette score.

After the search tried several combinations the tuned parameters for the DBSCAN model were 4.5 for epsilon and 5 for min_samples. The silhouette score for the model was about 0.7314. This indicates that the model performed relatively well in separating data points into distinct clusters.

DBSCAN is effective in identifying arbitrarily shaped clusters and is robust to noise and outliers due to its density-based approach. However, the model may struggle

with clusters of varying densities, and its performance can not be very high in high-dimensional spaces.

Conclusion

The silhouette scores for each model are as follows:

Model	**Silhouette Score**
K-Means Clustering	0.2798
Hierarchical Clustering	0.4237
DBSCAN	0.7314

Based on the silhouette scores the best performing model is DBSCAN. The K-means clustering algorithm performed the worst which can perhaps be explained by the large number of outliers in this dataset which K-means is very vulnerable to. Hierarchical clustering performed moderately better but still could not achieve a preferable score. This may be because this clustering model may be better suited for identifying clusters in data that do not have arbitrary shapes or sizes.

DBSCAN performed the best as it was able to separate out outliers and leave them out of the clusters it made, allowing more related and dense data points to populate each cluster.