UNSUPERVISED CUSTOMER SEGMENTATION AND PSYCHOLOGICAL TRAITS OF CREDIT CARD USERS

Department: Statistics

Subhrajit Dasgupta

(Reg no. 100438 of 2020-21

Roll - 96/STA No. 200028)

Ankan Ghosh

(Reg no. 100360 of 2017-2018

Roll - 96/STS No. 170002)

Pritam Saha

(Reg no. 100421 of 2020 - 2021

Roll - 96/STA No. 200014)

Under the supervision of Dr. Sushovan Jana (Maulana Abul Kalam Azad University of Technology, West Bengal)

Contents

Acknowledgement

We would like to express our deepest gratitude to our project supervisor Dr. Sushovan Jana (Maulana Abul Kalam Azad University of Technology, West Bengal) for his continuous support and guidance. Without him, it would not have been possible for us to shape and frame this project well enough in this particular field of study.

We would also wish to express our sincere thanks to our respected HOD Dr. Chandranath Pal (Department of Statistics, University of Kalyani) for his constant cooperation in completing our project. Finally, we are thankful to University of Kalyani for accepting us into the graduate program and nurturing us with the knowledge.

Abstract

Customer segmentation is important for any company for determining the marketing strategy. The available dataset tells us about the behaviors of about 9000 credit card users. With the newly emerging clustering techniques Spectral Clustering and Kernel K means we aim to divide the customers in different subpopulation such that customers with similar kind of behaviors can be separated for further study of common factors among the subpopulations by the method of factor analysis.

Keywords: Spectral Clustering, Kernel K means, Factor Analysis

Introduction

A credit card is a payment card issued to users (cardholders) to enable the cardholder to pay a merchant for goods and services based on the cardholder's accrued debt. The card issuer (usually a bank or credit union) creates a revolving account and grants a line of credit to the cardholder, from which the cardholder can borrow money for payment to a merchant or as a cash advance. There are two credit card groups: consumer credit cards and business credit cards.

Credit card providers usually target users using their behavior and demographic information. The behavior of the customers are described by the reports of purchases and pays made by customers. Customer segmentation is a technique of separating a customer base into groups which allows consumers to efficiently themselves and provides issuers to decide on marketing plans and strategies.

Clustering is a famous machine learning tool used for various data analysis. Among all the clustering algorithms K means clustering is widely used. One drawback of K means is that it can not separate the non-linearly separable data. Recent approaches have showed to work better to separate clusters that are non-linearly separable in input space. One of them is Kernel K means approach and other one is Spectral Clustering. Kernel K means maps the points to a higher dimensional space using some non-linear functions (e.g Gaussian Kernel, Polynomial Kernel, Sigmoid Kernel etc.) and then separates the points by linear separators in the new space. Spectral Clustering algorithm considers the datapoints as nodes of an undirected graph and constructs an affinity matrix based on some similarity measure and hence uses the eigenvectors of the affinity matrix along with k means algorithm to partition the points in different clusters. The new approach uses grouping by splitting vertices of the graph into disjoint sets and keeping the similarity high within a set and low between sets. The partitions are then evaluated based on loss known as normalized cut. Recent research has established theoretical connection between Weighted Kernel K means and Spectral Clustering (choosing the weights of Kernel K means in a certain way makes the objective function of kernel k means identical to normalized cuts) which implies eigen based algorithms which are computationally complex are not essential to minimize normalized cuts and local search& acceleration schemes may be used to improve the quality as well as speed of Kernel K means [1]. Moreover research work has been published to improve the quality of clustering when there is presence of multiple scales in the data and when the clusters are placed within a cluttered background. [2] shows how the standard spectral clustering using NG-JORDAN-WEISS algorithm fails for the data that incorporates multiple scales. The NJW algorithm suggested running the algorithm for different values of scaling parameter used in calculation of affinity matrix and then choose the value of the parameter for which clusters are least destroyed. This method needs extensive search and huge amount of time. Lihi Zelnik-Manor and Pietro Perona suggested using local scaling parameter for each data point in calculation of elements of affinity matrix. The selection of scaling parameter for each point can be done by studying the local statistics of neighborhood of a point.

Another challenging task of any clustering algorithm lies in choosing the number of clusters. One approach to discover the number of clusters is to analyze the eigenvalues of affinity matrix. The analysis given in [3] shows that the first eigenvalue of highest magnitude of the affinity matrix will be repeated eigenvalue of magnitude 1 with multiplicity being equal to the number of clusters. But this approach does not work if the clusters are overlapped and noise is introduced. [4] suggests the method of ‘eigen gap heuristic’ which can be used for both normalized and unnormalized graph Laplacian. Here the goal is to choose the number of groups K such that first k smallest eigenvalues of the Laplacian are very small (close to zero and difference among those k eigenvalues are less than some threshold say 0.00001) but the k+1 th eigenvalue is relatively large. Then according to the eigen gap heuristic K is our desired number of clusters. Again in this case also if we don’t have our clusters pretty much pronounced (presence of overlapping and noise) the eigenvalues will be equally spaced when plotted in a graph. As a result we are indecisive to choose the number of clusters. There is no hard and fast rule that all these above mentioned methods of choosing the number of clusters will work on our dataset. Our dataset unfortunately shows overlapping and presence of noise. As a result we have used trial and error method by varying the range of hyperparameters (number of clusters and number of neighbors in K-neighbor graph for constructing affinity matrix) and calculating a performance metric called Silhouette index for each of the combinations and hence the combination of hyperparameters showing best result according to performance metric has been chosen. Among all the existing algorithms of Spectral Clustering the one we have used makes use of normalized Graph Laplacian.

After dividing the customers in desired number of clusters our task was to run factor analysis on the subpopulations to summarize the characteristics of each subpopulation and then finding the common factors among them. In case of Factor analysis we need to determine the number of factor for each subpopulation after clustering. [5] suggests a procedure called Parallel Analysis in which the eigenvalues of the data prior to rotation are compared to that of random values of same dimensionality and then the Factor Analysis eigenvalues greater than Parallel Analysis eigenvalues from the corresponding random data are retained.

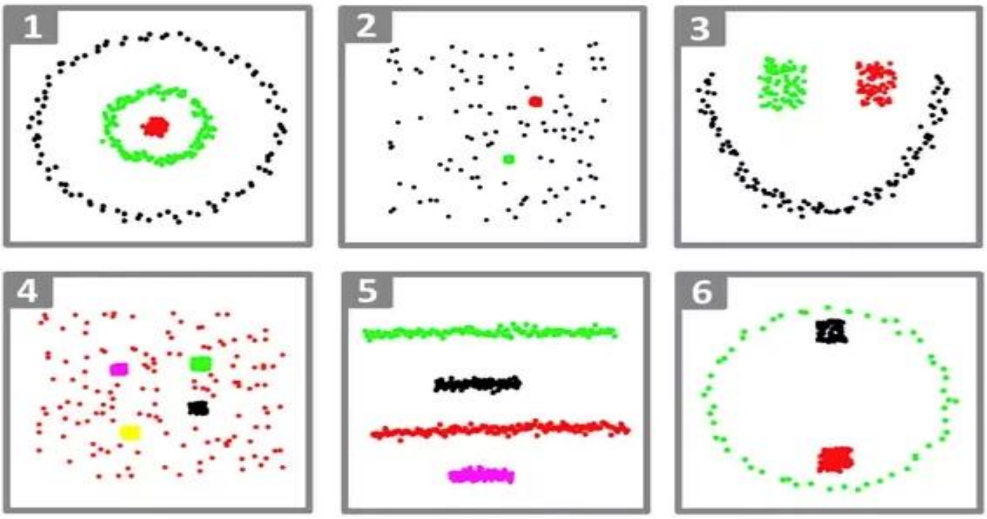
Objective of the study

1. Customer segmentation, which gives insight to card issuing company about the natures of the customers. Card issuers can not only make high-priced proposals, but can discover groups that have poorly serviced by present offers using improved segmentation.
2. Finding the common factors among the subpopulation of customers as their behavioral style can help study these groups and provide better alternatives and strategies to meet their demands.

Methodology Used

Spectral Clustering:

In spectral clustering data points are treated as nodes of a graph which makes it a graph partitioning problem. The nodes are mapped to low-dimensional space that can be easily segregated to form clusters. Unlike K means Spectral Clustering does not make any assumption on the shape of clusters and clusters the data based on connectivity while K-means is an algorithm based on compactness criteria.



(This figure is taken from Lihi Zelnik-Manor, Pietro Perona, "Self-Tuning Spectral Clustering"[2])

In all of these six instances the clustering based on compactness criteria i.e K means fails to model the data even if we select the correct k value by scree plot or any other method.

If we have P data points each with N features input matrix of Spectral Clustering would be PxP matrix. Spectral Clustering is indifferent to the number of features we use.

Algorithm:

- Project data into $\mathfrak{R}^n$ matrix
- Define an Affinity matrix A , using a Gaussian Kernel K or an Adjacency matrix
- Construct the Graph Laplacian from A (i.e. decide on a normalization)
- Solve the Eigenvalue problem
- Select k eigenvectors corresponding to the k lowest (or highest) eigenvalues to define a k-dimensional subspace
- Form clusters in this subspace using k-means.

Affinity Matrix:

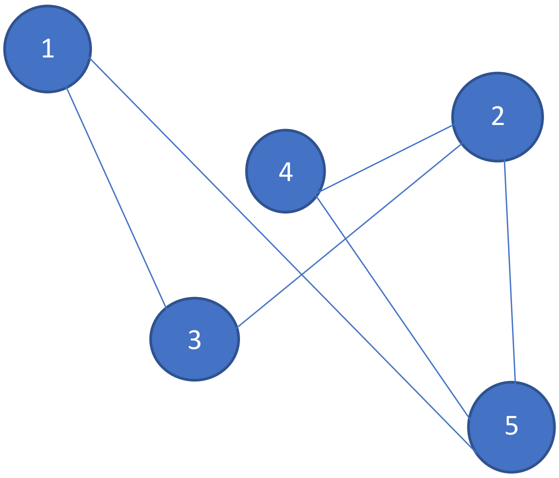
We first create an undirected graph G = (V, E) with vertex set V = $\{V_1, V_2, \ldots, V_n\}$ = 1, 2, …, n observations in the data.

- **Epsilon-neighbourhood Graph:** A parameter epsilon is fixed beforehand. Then each point is connected to all the points which lie in it's epsilon-radius. If all the distances between any two points are similar in scale then typically the weights of the edges i.e the distance between the two points are not stored since they do not provide any additional information. Thus, in this case, the graph built is an undirected and unweighted graph.

- **K-Nearest Neighbours:** A parameter k is fixed beforehand. Then, for two vertices u and v, an edge is directed from u to v only if v is among the k-nearest neighbours of u. Note that this leads to the formation of a weighted and directed graph because it is not always the case that for each u having v as one of the k-nearest neighbours, it will be the same case for v having u among its k-nearest neighbours. To make this graph undirected, one of the following approaches are followed
  - Direct an edge from u to v and from v to u if either v is among the k-nearest neighbours of u or u is among the k-nearest neighbours of v.
  - Direct an edge from u to v and from v to u if v is among the k-nearest neighbours of u and u is among the k-nearest neighbours of v.
- **Fully-Connected Graph:** To build this graph, each point is connected with an undirected edge-weighted by the distance between the two points to every other point. Since this approach is used to model the local neighbourhood relationships thus typically the Gaussian similarity metric is used to calculate the distance.

S(x_i,x_j)= exp (-$\frac{\|x_i-x_j\|^2}{2\sigma^2}$)

Thus, when we create an adjacency matrix for any of these graphs, A_{ij} ~ 1 when the points are close and A_{ij}→ 0 if the points are far apart.



	1	2	3	4	5
1	0	0	w_{13}	0	w_{15}
2	0	0	w_{23}	w_{24}	w_{25}
3	w_{31}	w_{32}	0	0	0
4	0	w_{42}	0	0	w_{45}
5	w_{51}	w_{52}	0	w_{54}	0

The graph with nodes 1 to 5 weights (or similarity) w_{ij} and its adjacency matrix.

Affinity metric determines how close, or similar, two points our in our space. We will use a Gaussian Kernel and not the standard Euclidean metric.

Given 2 data points x_i,x_j(projected in $\mathbb{R}^n$), we define an Affinity A_{ij} that is positive, symmetric, and depends on the Euclidian distance $\|x_i - x_j\|^2$between the data points

$$A_{ij}= \exp(-\alpha\|x_i - x_j\|^2)$$

We might provide a hard cut off R , so that

A_{ij}= 0 if $\|x_i - x_j\|^2 \geq$ R
$A_{ij}\simeq1$, $A_{ij}\simeq1$ when the points are close in $\mathbb{R}^n$, and $A_{ij}\rightarrow0$ if the points x_i,x_j are far apart. Close data points are in the same cluster. Data points in different clusters are far away. But data points in the same cluster may also be far away even farther away than points in different clusters. Our goal then is to transform the space so that when 2 points x_i,x_j are close, they are always in same cluster, and when they are far apart, they are in different clusters.

Graph Laplacian is just another matrix representation of a graph. It can be computed as:

- Simple Laplacian L= D−A
- Normalized Laplacian L_N = $D^{-1/2}$L $D^{-1/2}$
- Generalized Laplacian L_G= D^{-1}L

Ng, Jordan, & Weiss Laplacian L_{NJW}=$D^{-1/2}$A$D^{-1/2}$ where A_{ii}=0

L=D−A where A is the Adjacency matrix and D is the Degree Matrix.

$$D_i= \sum_{j|(i,j)\in E} w_{ij}$$
$$L_{ij}= d_{ii} \text{ if i=j}$$
$$L_{ij} = -w_{ij} \text{ if i,j} \in \text{E}$$
$$L_{ij}\text{=0 if i,j not in E}$$

	1	2	3	4	5
1	$w_{13}+$ w_{15}	0	$-w_{13}$	0	$-w_{15}$
2	0	$w_{23}+$ $w_{24}+w_{25}$	$-w_{23}$	$-w_{24}$	$-w_{25}$
3	$-w_{31}$	$-w_{32}$	$w_{31}+w_{32}$	0	0
4	0	$-w_{42}$	0	$w_{42}+$ w_{45}	$-w_{45}$
5	$-w_{51}$	$-w_{52}$	0	$-w_{54}$	$w_{51}+$ $w_{52}+w_{54}$

This is the calculation for Graph Laplacian for the above diagram of undirected graph. The whole purpose of computing the Graph Laplacian L was to find eigenvalues and eigenvectors for it, in order to embed the data points into a low-dimensional space. To identify the number of clusters we can use the Graph Laplacian which should look like a block diagonal matrix where each block corresponds to a cluster.A graph G has k connected components iff

the algebraic multiplicity of eigenvalue 0 of the graph Laplacian matrix is k. The aforementioned methods (see 'Introduction') can also be used to determine number of clusters.
For K clusters, we have to compute the first K eigenvectors v_1, v_2,.....,v_k. Stack the vectors vertically to form the matrix with eigen vectors as columns. Represent every node as the corresponding row of this new matrix, these rows form the feature vector of the nodes. Use K means to cluster these points into k clusters C_1,C_2,.........,C_k.

Kernel k-means:

K-Means is one of the most widely used and fundamental unsupervised algorithms. K-Means aims to partition N observations into K clusters in which each observation belongs to the cluster with the nearest mean (cluster centroid) i.e. k-means tries to minimize within cluster variation or simply the sum of squared error within each cluster. In other words, it minimizes within-cluster dissimilarity using the following objective function:

$$\min_{C_1,C_2,C_3,\ldots,C_k} \sum_{i=1}^{k} \sum_{x \in C_i} ||x - u_i||^2$$

where

$$u_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$$

The objective function is minimized using an iterative approach that converges to a locally optimal solution. We alternate between (a) treating the cluster centroids (u_i's) as fixed and computing cluster assignments (C_i's) and (b) treating the cluster assignments as fixed and computing cluster centroids.

Now a major drawback of K-Means clustering method is that it fails when the clusters are non-linearly separable in input space. Now Kernel K-means is an extension of the standard k-means clustering algorithm that identifies nonlinearly separable clusters. This algorithm applies the same trick as k-means but with one difference that here in the calculation of distance, kernel method is used instead of the Euclidean distance. In kernel k-means before clustering, points are mapped to a higher-dimensional feature space using a nonlinear transformation Φ, and then kernel k-means partitions the points by linear separators in the new space.

TABLE I

EXAMPLES OF KERNEL FUNCTIONS

Polynomial Kernel	$\kappa(a,b)=(a\cdot b+c)^{d}$
Gaussian Kernel	$\kappa(a,b)=\exp\ (-||a-b||^{2}/2\sigma^{2})$
Sigmoid Kernel	$\kappa(a,b)=\tanh(c(a\cdot b)+\theta)$

Now to show how kernel method is applied to K-means algorithm, we have to vectorize the objective function of K-means. This re-formulation is rigorously shown by Ali Caner Türkmen in his A Review of Nonnegative Matrix Factorization Methods for Clustering.

Variables and Matrix definitions:

Let us define variables and matrices to be utilized to vectorize the objective function of K-Means.

Variables:

- x is the input data point, a vector of dimension M
- N be the number of data points
- K is the number of clusters

Matrix Definitions:

- Matrix X is the input data points arranged as the columns, dimension M x N
- Matrix B is the cluster assignments of each data point, dimension N x K
- Matrix D is the number of data points assigned to each cluster, inverted and placed along the diagonal, dimension K x K

Matrix B: B is a matrix where the rows are one-hot encoded cluster assignments for each data point of the X matrix. In hard clustering, each row can only have one column with a 1, indicating the data point is assigned to that cluster.

$$\mathrm{B} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Matrix D: D is the number of data points assigned to each cluster, inverted and placed along the diagonal. Notice that the diagonal elements inverted and summed are equal to N, number of data points and the dimension of the matrix is K x K, number of clusters. Also, notice that D can be derived completely from B, where D = inverse of B^TB.

$$\mathrm{D} = \mathrm{diag}\big(1/|C_1|, \quad 1/|C_2|, \quad 1/|C_3|, \quad \cdots, \quad 1/C_k|\big)$$

Matrix X: Matrix X is the input data points arranged as columns. Each data point (x_i) is a vector of dimension M. X = [x_1, x_2, x_3, …., x_N]

The matrix product XBD then represents the matrix of cluster centroids, having dimension M (vector dimension) x K (number of clusters).

With all the above formulations, you can see matrix multiplying X and B gives a new matrix, where each column is the sum of the vectors in each cluster. Then, multiplying with D gives each cluster centroid on the columns, where the cluster centroid is defined as sum of data points in each cluster /number of data points in the cluster. In other words, XBD yields a matrix of dimension M x K and can be defined as,

XBD = [u_1, u_2, u_3, ……, u_k]

Multiplying the XBD matrix by B^T gives a matrix with the same dimension as X, where each column is the respective centroid of the cluster the data point is assigned to.

$XBDB^T$ = [u_2, u_1, u_3, ……, u_{k-2}]

The following matrix has the difference between X and the respective cluster centroid that X belongs to on each column. Hence minimizing the Frobenius norm is equivalent to minimizing the K-Means objective function written previously.

$$\min_B ||X - XBDB^T||_F^2$$

Notice that matrix D is fully derived from matrix B. In other words, the only variable in the vectorized objective is the assignment matrix B. X is the input data. D = inverse of B^TB.

We can use trace operators to show that minimizing the objective function in written above is equivalent to maximizing the following function. Ali Caner Türkmen shows the derivations for the equivalency of this.

$$\max_B tr(X^T XBDB^T)$$

Kernel methods for K-Means

Kernel methods can be applied when the objective function can be written as a function of dot products. We can see this is the case after re-formulating the K-Means objective function to have the term X^TX. This allows us to map our current feature vectors into higher dimensional spaces in a more computationally efficient manner using the kernel trick.

Why do we need higher dimension feature vectors in K-Means?

As mentioned, K-Means performs best when clusters are spherical, dense, and linearly separable, like in the image on the right in Figure 1. For non-linearly separable clusters, like in the image on the left in Figure 1, we often would like to project our data into a higher dimensional space to make the resulting clusters linearly separable. Projecting to a higher dimension directly and calculating the objective for K-Means can be computationally expensive, as this requires calculating X^TX. Kernel trick allows us to project our data into a higher dimensional space to achieve linear separability and solve the K-Means problem in a more efficient way.

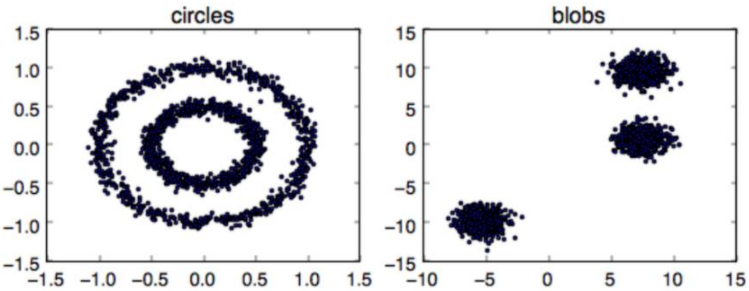


Fig: 1 Example data points for clustering. Where the left is non-linearly separable, and the right is linearly separable

The mechanics of kernel methods

A kernel function corresponds to an inner product of vectors in a certain space. It can be thought of as a similarity function over pairs of data points in this space. It is important to keep in mind that not all vector spaces have a corresponding kernel function.

In other words, kernel methods allow us to get the result of X^TX where X may be in a higher dimension without actually calculating X in that higher dimension. Hence anytime X exists in an objective function only in the form of X^TX, we can apply kernel tricks to easily expand the feature set to higher dimensions.

The objective function that Kernel k-means tries to minimize is the equivalent of the clustering error in the feature space shown below

E (u_1, u_2, u_3, ……, u_k) = $\sum_{i=1}^{N}\sum_{j=1}^{K} I\ (x_i\ \in\ C_j)\left|\left|\phi(x_i)\ -\ u_j\right|\right|\quad where\ u_j\ = \frac{\sum_{i=1}^{N} I(x_i \in C_j)\phi(x_i)}{\sum_{i=1}^{N} I(x_i \in C_j)}$

…… (i)

We can define a kernel matrix where $K\ \in\ \mathbb{R}^{N\times N} where K_{ir} = \phi(x_i)^T\phi(x_r)$ and by taking advantage of the kernel trick, we can compute the squared Euclidian distances in (i) without explicit knowledge of the transformation using

$$\left|\left|\phi(x_i)\ -\ u_j\right|\right|^2\ =\ K_{ii}\ -\ \frac{2\sum_{r=1}^{N} I(x_r \in C_j)K_{ir}}{\sum_{r=1}^{N} I(x_r \in C_j)}\ +\ \frac{\sum_{r=1}^{N}\sum_{l=1}^{N} I(x_r \in C_j)I(x_l \in C_j)K_{rl}}{\sum_{r=1}^{N}\sum_{l=1}^{N} I(x_r \in C_j)I(x_l \in C_j)} \dots \dots (ii)$$

Any positive semi-definite matrix can be used as a kernel matrix. Notice that in this case cluster centers $\mathbf{u_j}$ in the feature space cannot be calculated directly. Usually a kernel function **K (x$_i$, x$_r$)** is used to directly provide the inner products in the feature space without explicitly defining transformation Φ (for certain kernel functions the corresponding transformation is intractable). Some kernel function examples are given in Table I;

K (x$_i$, x$_r$) = K$_{ir}$

It must be noted that, by associating a weight with each data point, the weighted kernel k-means algorithm is derived and it is proven that its objective function is equivalent to that of many graph partitioning problems such as ratio association, normalized cut etc. if the weights and kernel are set appropriately.

Algorithmic steps for Kernel K-means clustering:

Input: Kernel matrix K, Number of clusters k, Initial clusters C_1, C_2, C_3, ……, C_k.

Output: Final clusters C_1, C_2, C_3, ……, C_k, Clustering error E.

1. For each point x_n and every cluster C_j compute $\left|\left|\phi(x_n\ -\ u_j)\right|\right|^2$ using (ii)

2. Find c*(x $_n$) = $\arg\min_j \left|\left|\phi(x_n\ -\ u_j)\right|\right|^2$

3. Update clusters as C_i = {x $_n$ | c*(x $_n$) = i}

4. If not converged go to step 1 otherwise stop and return final clusters C_1, C_2, C_3, ……, C_k and E calculated using (2).

Advantages

1) Algorithm is able to identify the non-linear structures.

2) Algorithm is best suited for real life data set.

Disadvantages

1) Number of cluster centers need to be predefined.

2) Algorithm is complex in nature and time complexity is large.

Evaluation Methods

Silhouette Score:-

Silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1, where +1 indicates clusters are well apart from each other and are clearly distinguished,0 indicates clusters are indifferent, or we can say that the distance between clusters is not significant and -1 indicates clusters are assigned in a poor manner.

Silhouette Score = (b-a)/max(a,b)

where a= average intra-cluster distance i.e., the average distance between each point within a cluster, and b= average inter-cluster distance i.e., the average distance between all clusters.

Dunn Index:-

Dunn Index is a metric used to calculate the goodness of a clustering technique. Its value ranges from 0 to infinity. A high value of Dunn Index indicates better clustering since observations in each cluster are closer together, while clusters themselves are further away from each other. It is defined as the ratio between the smallest inter-cluster distance to largest intra-cluster distance.

Connectivity Score:-

Connectivity Score indicates the degree of connectedness of the clusters, as determined by the k-nearest neighbours. Connectivity corresponds to what extent items are placed in the same cluster as their nearest neighbours in the data space. Its value ranges from 0 to infinity. A low Connectivity Score indicates better clustering.

Factor Analysis

Factor Adequacy Check :

The Kaiser-Meyer-Olkin (KMO) Test is a measure of how suited your data is for Factor Analysis. The test measures sampling adequacy for each variable in the model and for the complete model. The statistic is a measure of the proportion of variance among variables that might be common variance. The lower the proportion, the more suited your data is to Factor Analysis.

$$KMO = \frac{\sum\sum_{j\neq k} r_{jk}^2}{\sum\sum_{j\neq k} r_{jk}^2 + \sum\sum_{j\neq k} p_{jk}^2}$$

Here r_{jk} is the correlation between the variable in question and another, and p_{jk} is the partial correlation.

KMO returns values between 0 and 1. A rule of thumb for interpreting the statistic:

KMO values between 0.8 and 1 indicate the sampling is adequate. KMO values less than 0.6 indicate the sampling is not adequate and that remedial action should be taken. Some authors put this value at 0.5, so use your own judgment for values between 0.5 and 0.6.

KMO Values close to zero means that there are large partial correlations compared to the sum of correlations. In other words, there are widespread correlations which are a large problem for factor analysis. For reference, Kaiser put the following values on the results:

0.00 to 0.49 unacceptable. 0.50 to 0.59 miserable. 0.60 to 0.69 mediocre. 0.70 to 0.79 middling. 0.80 to 0.89 meritorious. 0.90 to 1.00 marvellous.

Bartlett's Test of Sphericity compares an observed correlation matrix to the identity matrix. Essentially it checks to see if there is a certain redundancy between the variables that we can summarize with a few numbers of factors.

The null hypothesis of the test is that the variables are orthogonal, i.e., not correlated. The alternative hypothesis is that the variables are not orthogonal, i.e., they are correlated enough to where the correlation matrix diverges significantly from the identity matrix.

Factor Analysis:-

Factor Analysis is a method for modeling observed variables, and their covariance structure, in terms of a smaller number of underlying unobservable (latent) "factors". The factors are broad concepts or ideas that may describe an observed phenomenon.

Model:-

Our factor model can be thought of as a series of multiple regressions, predicting each of the observable variables X_i from the values of the unobservable common factors f_i as shown below :

$$\begin{aligned}
X_1 &= \mu_1 + l_{11}f_1 + l_{12}f_2 + \dots + l_{1m}f_m + \epsilon_1 \\
X_2 &= \mu_2 + l_{21}f_1 + l_{22}f_2 + \dots + l_{2m}f_m + \epsilon_2 \\
&\vdots \\
X_p &= \mu_p + l_{p1}f_1 + l_{p2}f_2 + \dots + l_{pm}f_m + \epsilon_p
\end{aligned}$$

The regression coefficients l_{ij} (the partial slopes) for all of these multiple regressions are called factor loadings. Here, l_{ij}= loading of the i^{th} variable on the j^{th}factor, m<<p. Here,

$$\mathbf{L} = \begin{pmatrix} l_{11} & l_{12} & \dots & l_{1m} \\ l_{21} & l_{22} & \dots & l_{2m} \\ \vdots & \vdots & & \vdots \\ l_{p1} & l_{p2} & \dots & l_{pm} \end{pmatrix} = \text{matrix of factor loadings}$$

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_p \end{pmatrix} = \text{vector of specific factors} \qquad \mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{pmatrix} = \text{vector of traits}$$

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{pmatrix} = \text{population mean vector} \qquad \mathbf{f} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_m \end{pmatrix} = \text{vector of common factors}$$

Thus, our model reduces to

$$\mathbf{X} = \boldsymbol{\mu} + \mathbf{L}\mathbf{f} + \boldsymbol{\epsilon}$$

Model assumptions:-

$E(\epsilon_i) = 0$; *i* = 1, 2, … , *p*

$E(f_i) = 0$; *i* = 1, 2, … , *m*

$\mathrm{var}(f_i) = 1$; *i* = 1, 2, … , *m*

$\mathrm{var}(\epsilon_i) = \psi_i$; *i* = 1, 2, … , *p* Here, ψ_i is called the *specific variance*.

$\mathrm{cov}(f_i, f_j) = 0$ for *i* ≠ *j*

$\mathrm{cov}(\epsilon_i, \epsilon_j) = 0$ for *i* ≠ *j*

$\mathrm{cov}(\epsilon_i, f_j) = 0$; *i* = 1, 2, … , *p*; j=1,…,m

Estimation of parameters:-

Estimation of parameters can be done either by using Principal Component Method or by using Maximum Likelihood Estimation (MLE) Method. Since the principal component method does not provide a test for lack-of-fit, so the MLE method is to be used for estimating the parameters of the factor model.

- ### MLE method:-

Using the Maximum Likelihood Estimation Method, we **assume** that **the data are independently sampled from a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and variance-covariance matrix of the form $\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}' + \boldsymbol{\Psi}$** where $\mathbf{L}$ is the matrix of factor loadings and $\boldsymbol{\Psi}$ is the diagonal matrix of specific variances. The data vectors for *n* subjects are $\mathbf{X_1}, \mathbf{X_2}, \dots, \mathbf{X_n}$.

Maximum likelihood estimation involves estimating the mean, the matrix of factor loadings, and the specific variance. The maximum likelihood estimator for the mean vector μ, the factor loadings L, and the specific variances Ψ are obtained by finding $\hat{\mu}$, $\hat{\mathbf{L}}$, and $\hat{\boldsymbol{\Psi}}$ that maximize the log likelihood given by the following expression:

$$l(\mu, \mathbf{L}, \boldsymbol{\Psi}) = -\frac{np}{2}\log\ 2\pi - \frac{n}{2}\log|\boldsymbol{LL}' + \boldsymbol{\Psi}| - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{X_i} - \mu)'(\mathbf{LL}' + \boldsymbol{\Psi})^{-1}(\mathbf{X_i} - \mu)$$

The log of the joint probability distribution of the data is maximized. In this way estimation of parameters of factor model is done.

- ### Regression Method:-

  There are *m* unobserved factors in our model and we would like to estimate those factors. Therefore, given the factor model:

$$\mathbf{Y_i} = \boldsymbol{\mu} + \mathbf{L}\mathbf{f_i} + \boldsymbol{\epsilon}_\mathbf{i}; i = 1,2,\dots,n$$

we may wish to estimate the vectors of factor scores $\mathbf{f_1}, \mathbf{f_2}, \dots, \mathbf{f_n}$ for each observation. This method is used for **maximum likelihood estimates** of **factor loadings**. The joint distribution of the data $\mathbf{Y_i}$ and the factor $\boldsymbol{f}_i$ is $\binom{\mathbf{Y_i}}{\mathbf{f_i}} \sim N[\binom{\boldsymbol{\mu}}{0}, \begin{pmatrix} \mathbf{LL'} + \boldsymbol{\Psi} & \mathbf{L} \\ \mathbf{L'} & \mathbf{I} \end{pmatrix}]$.So, $E(\mathbf{f_i}|\mathbf{Y_i}) = \mathbf{L'}(\mathbf{LL'} + \boldsymbol{\Psi})^{-1}(\mathbf{Y_i} - \boldsymbol{\mu})$. Therefore, $\hat{\mathbf{f}}_\mathbf{i} = \hat{\mathbf{L}}'(\hat{\mathbf{L}}\hat{\mathbf{L}}' + \hat{\boldsymbol{\Psi}})^{-1}(\mathbf{Y_i} - \overline{\mathbf{y}})$.

There is a little bit of a fix that often takes place to reduce the effects of incorrect determination of the number of factors. This tends to give you results that are a bit more stable.So, $\tilde{\mathbf{f}}_\mathbf{i} = \hat{\mathbf{L}}'\mathbf{S}^{-1}(\mathbf{Y_i} - \overline{\mathbf{y}})$.

Factor Rotations:-

Factor rotation is motivated by the fact that factor models are not unique. The factor model $\mathbf{X} = \boldsymbol{\mu} + \mathbf{LF} + \boldsymbol{\epsilon}$ is equivalent to a rotated factor model, $\mathbf{X} = \boldsymbol{\mu} + \mathbf{L^*F^*} + \boldsymbol{\epsilon}$, where we have set L*=LT and f *=T'f for some orthogonal matrix T where T'T=TT'=I. Note that there are an infinite number of possible orthogonal matrices, each corresponding to a particular factor rotation.We plan to find an appropriate rotation, defined through an orthogonal matrix T**,** that yields the most easily cleaner interpretable factors.

Varimax Rotation:-

It involves scaling the loadings by dividing them by the corresponding communality : $\tilde{\boldsymbol{l}}^*_{ij} = \hat{\boldsymbol{l}}^*_{ij}/\hat{\boldsymbol{h}}_i$. Varimax rotation finds the rotation that maximizes this quantity. The Varimax procedure selects the rotation in order to maximize

$$V = \frac{1}{p}\sum_{j=1}^{m}\left\{\sum_{i=1}^{p}(\tilde{l}^*_{ij})^4 - \frac{1}{p}\left(\sum_{i=1}^{p}(\tilde{l}^*_{ij})^2\right)^2\right\}$$

The interpretation after varimax rotation is much cleaner than that of the original analysis.

Communalities:-

The total communality value is the sum of all communality values where the communalities for the i^{th} variable are the proportion of variation in that variable explained by the p factors. It is given by $\sum_{i=1}^{p} \hat{h}_i^2 = \sum_{i=1}^{p} \sum_{j=1}^{m} \hat{l}_{ij}^2 = \sum_{i=1}^{m} \hat{\lambda}_i$

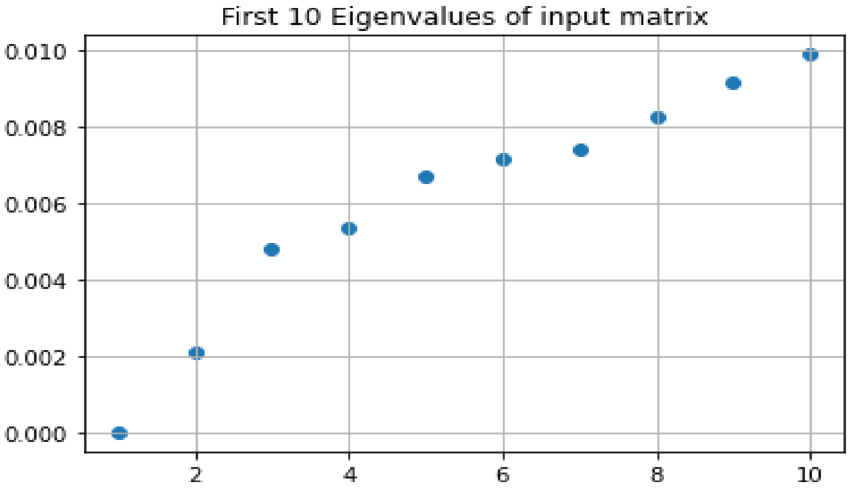
The percentage of variation explained by our model is $(\sum_{i=1}^{p} \hat{h}_i^2)/\text{p}$, which also depicts the performance of the model.

Checking The Model Fit: It is customary to check whether the hypothesized model fit the data well. Two model fit indices are widely used namely RMSEA Index and Tucker Lewis Index. RMSEA index is an absolute fit index which assess how far a hypothesized model is from a perfect model. Tucker Lewis Index is an incremental fit index which compares the fit of a hypothesized model with that of a Baseline model i.e. a model with the worst fit. RMSEA value of < 0.05 indicates close fit and that < 0.08 suggests a reasonable model data fit. Tucker Lewis Index > 0.9 indicates good fit.

Data Analysis

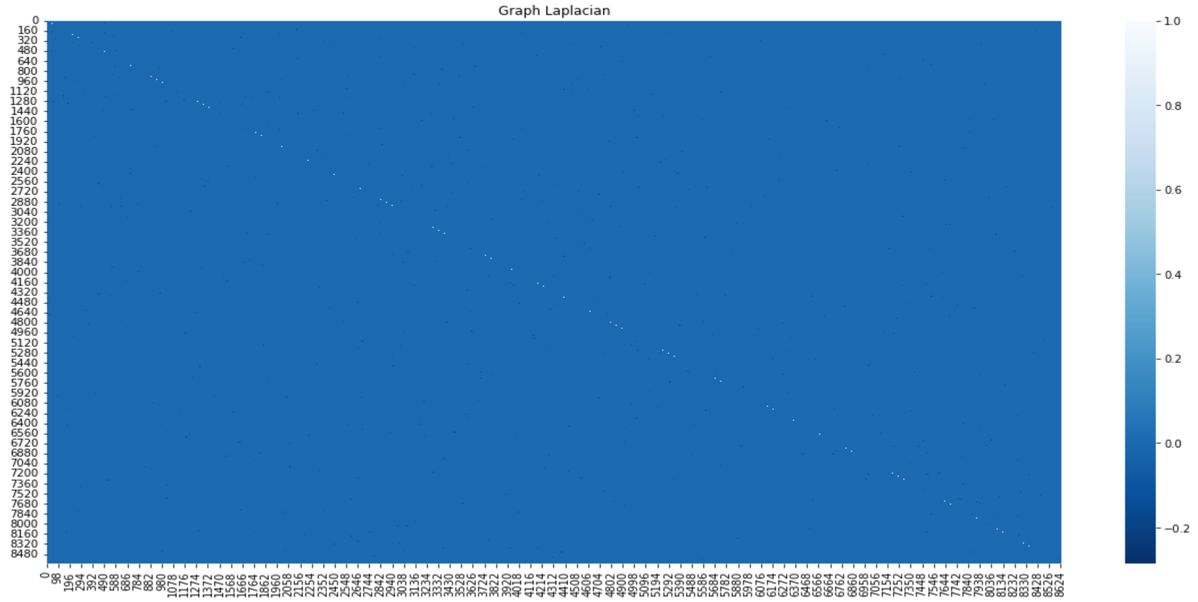
Based on Spectral Clustering:-

After data cleaning procedures are done our data becomes ready for further analysis. The first step of implementing the spectral clustering algorithm is to determine the number of clusters. To select the appropriate number of clusters one can use the ‘eigen gap heuristic’ which plots the first ten smallest eigenvalues of the Graph Laplacian matrix and looks for the sudden jump in the plot. Suppose we have first three eigenvalues very close to each other(all of them near zero) and a jump at the fourth one is noticed then the appropriate cluster number would be 3. This is based on theorem that a Laplacian of regular graph has k connected components iff the Laplacian has eigenvalue zero with multiplicity k.

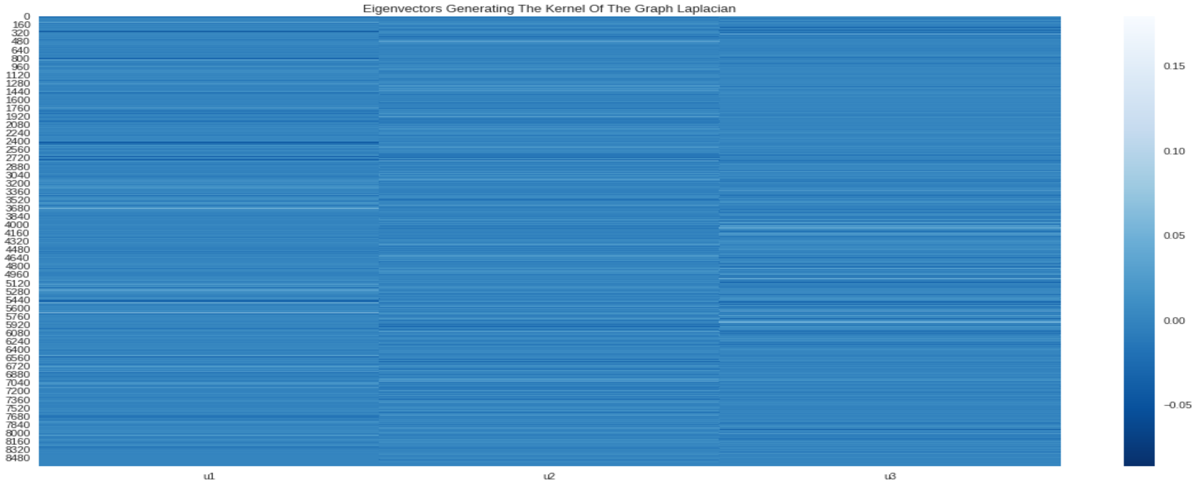


In this plot we see that the eigen gap heuristic fails to determine the appropriate number of clusters because there is presence of overlapping and noise in the data. It is proven that a regular graph is connected iff there is an eigenvalue 0 with algebraic multiplicity 1. So our graph is connected.

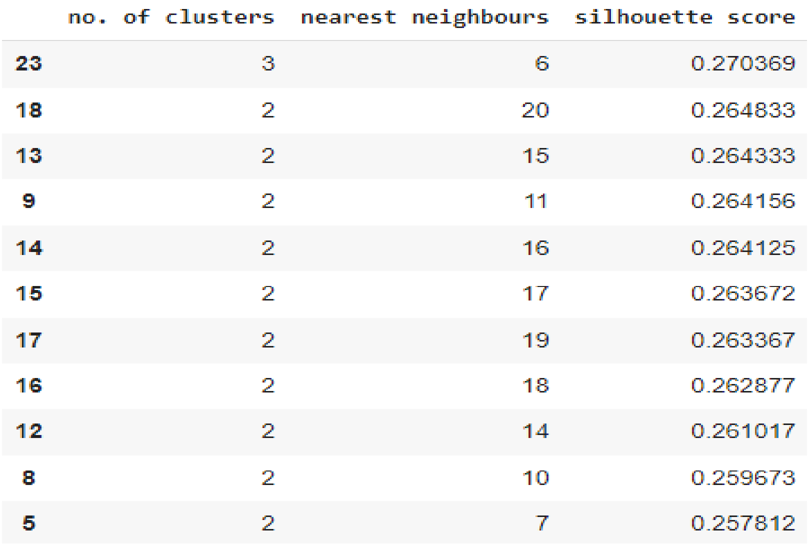
The heatmap of Graph Laplacian generated with six nearest neighbours:



The dark blue color represents values close to zero which shows the Laplacian is block diagonal. The eigenvectors of Laplacian matrix is often used to determine the number of clusters. It is wise to construct the heatmap of eigenvectors of the Laplacian in which we should have an eigenvector at principle diagonal corresponding to each block of Laplacian and off diagonals are null vector. If the heatmap of the eigenvectors of Laplacian shows block diagonal structure the number of blocks corresponds to the number of clusters. Below is the heatmap of eigenvectors of Laplacian.

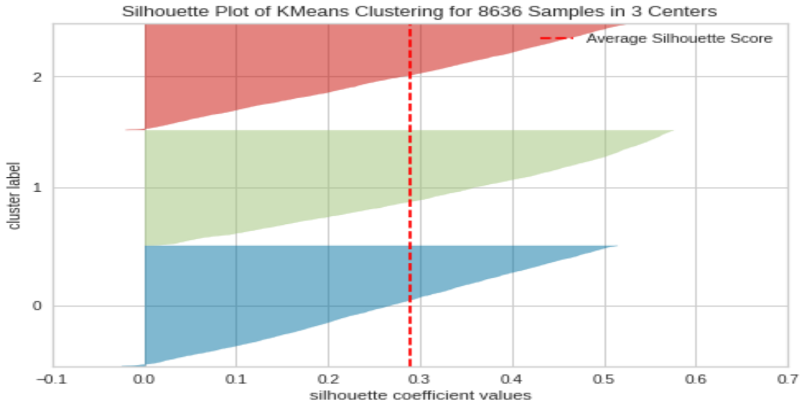


This heatmap doesn't provide any clear idea about the number of clusters. Hence different combinations of hyperparameters were implemented to find manually the number of clusters and the hyperparameter of the nearest neighbours. The graph Laplacian is created with 6 nearest neighbours because after running different combinations of k and number of clusters with normalized Laplacian the Silhouette score is maximum for k=6 and number of clusters=3.

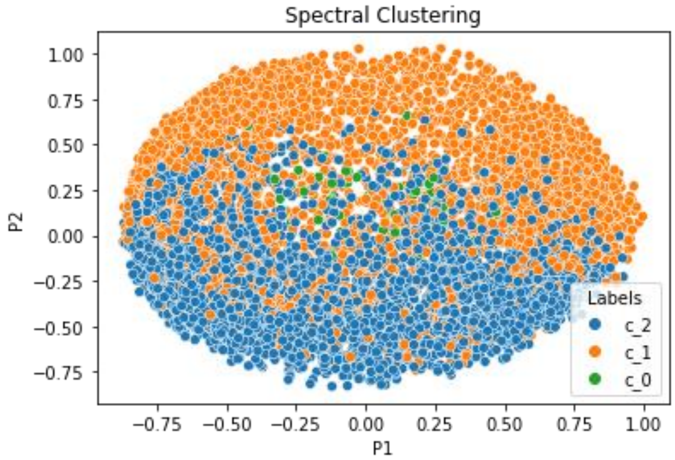


	no. of clusters	nearest neighbours	silhouette score
23	3	6	0.270369
18	2	20	0.264833
13	2	15	0.264333
9	2	11	0.264156
14	2	16	0.264125
15	2	17	0.263672
17	2	19	0.263367
16	2	18	0.262877
12	2	14	0.261017
8	2	10	0.259673
5	2	7	0.257812

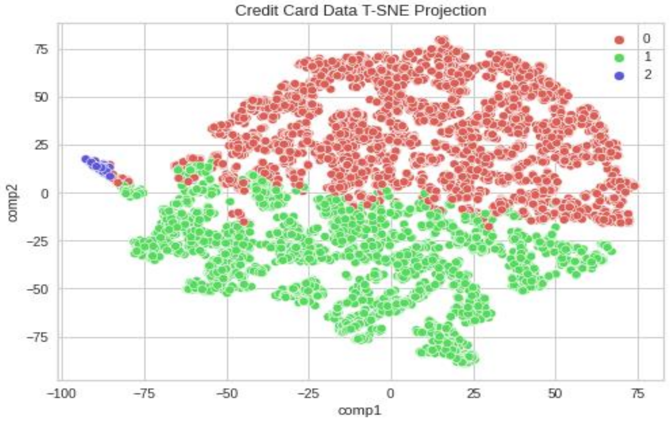
The model was executed with Python's Sklearn Package with the above mentioned hyperparameters. After fitting the model a Silhouette Analysis was done to evaluate the density and separation between clusters. The following plot displays the Silhouette coefficient(lies between -1 to 1 where score near 1 implies high separation, score near 0 indicates overlapping and score near -1 indicates wrong cluster assignment) for each sample on a per-cluster basis allowing us to visualize the density and separation of the clusters. The vertical red-dotted line indicates the average Silhouette score for all observations.



The plot shows that our average Silhouette Score is close to 0.3 which is 0.270369 indeed. To get a visualization of the clustering labels one can use PCA(Principal Component Analysis) to project the data in lower dimension and visualize how the data is clustered. Below is the two dimensional such plot done with first two principal components.

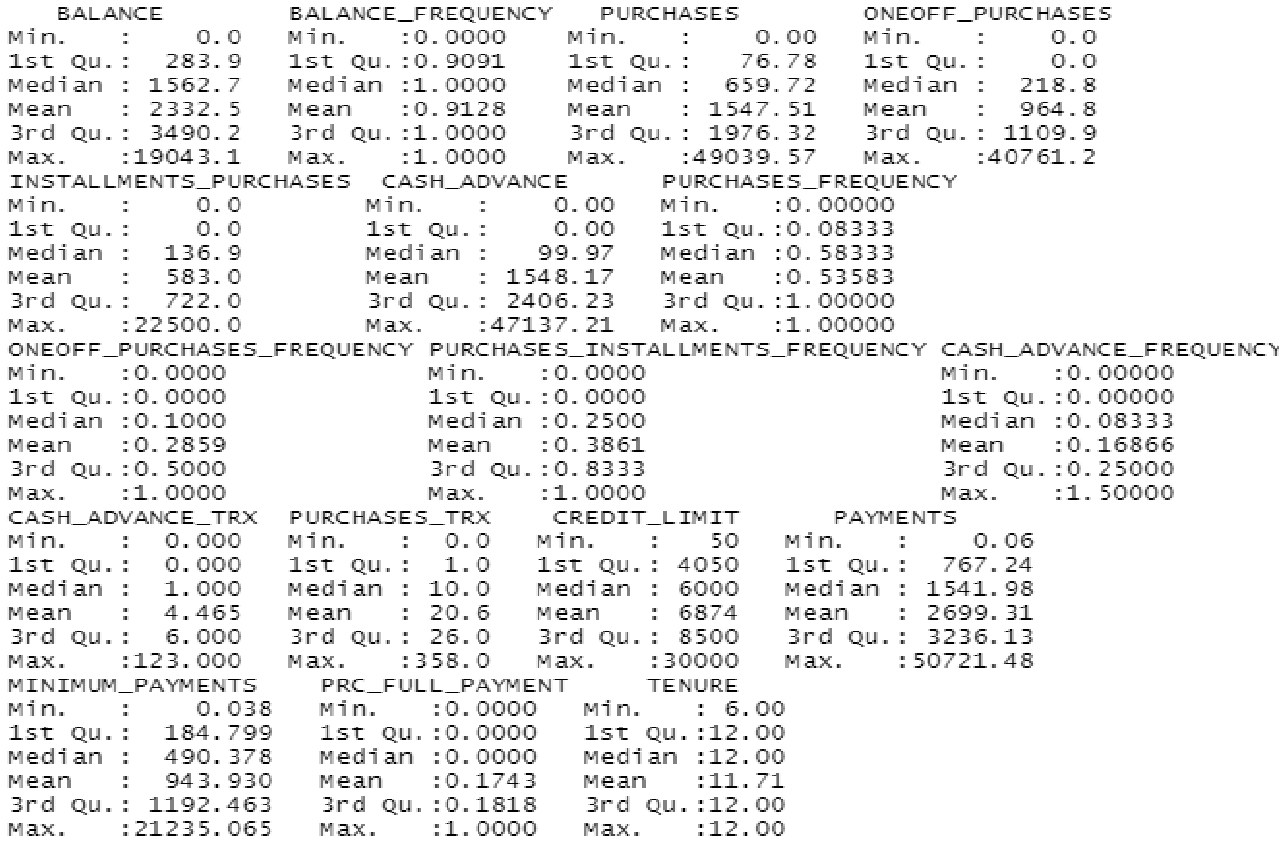


PCA is a linear dimensionality reduction technique. One drawback with PCA is when the data is projected to lower dimension they are overlapped and hence the visualization of the clustering labels become difficult. One alternative to this problem is non-linear dimensionality reduction like t distributed stochastic neighbourhood embedding(t-SNE). Below is the t-SNE plot of our data with clustering labels labelled by the Spectral Clustering algorithm.



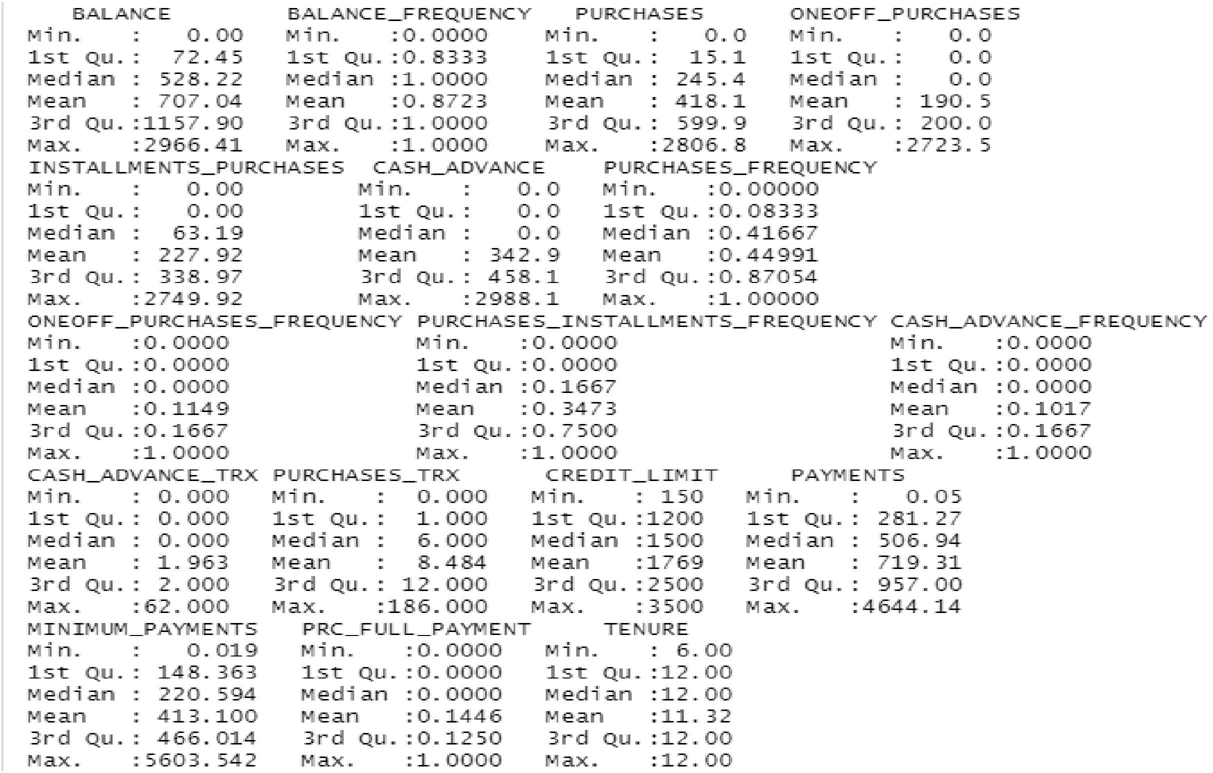
After we are done with Clustering we have three subpopulations in hand. It is customary to summarize different characteristics of the subpopulations.

First Subpopulation:



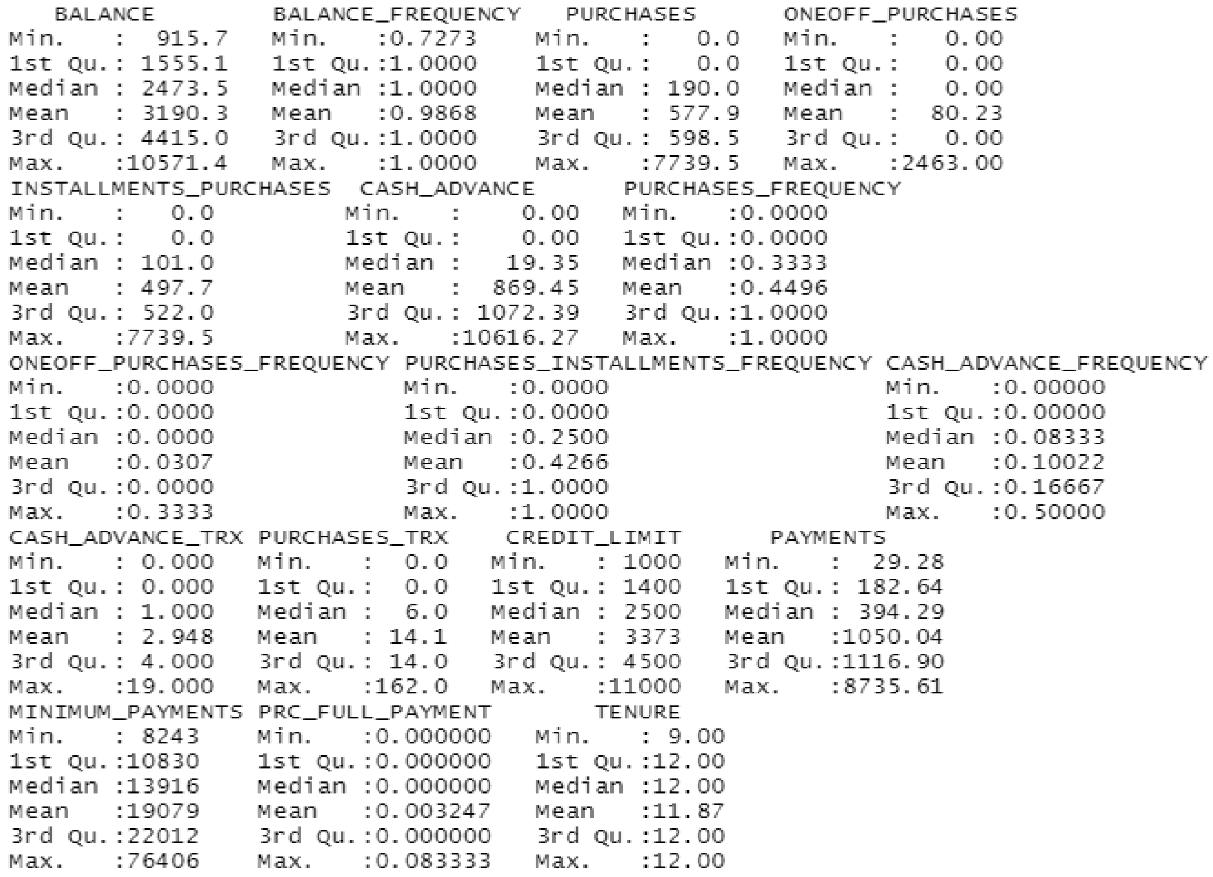
```
    BALANCE          BALANCE_FREQUENCY    PURCHASES         ONEOFF_PURCHASES
 Min.   :    0.0   Min.   :0.0000    Min.   :    0.00   Min.   :    0.0
 1st Qu.:  283.9   1st Qu.:0.9091    1st Qu.:   76.78   1st Qu.:    0.0
 Median : 1562.7   Median :1.0000    Median :  659.72   Median :  218.8
 Mean   : 2332.5   Mean   :0.9128    Mean   : 1547.51   Mean   :  964.8
 3rd Qu.: 3490.2   3rd Qu.:1.0000    3rd Qu.: 1976.32   3rd Qu.: 1109.9
 Max.   :19043.1   Max.   :1.0000    Max.   :49039.57   Max.   :40761.2
 INSTALLMENTS_PURCHASES  CASH_ADVANCE      PURCHASES_FREQUENCY
 Min.   :    0.0        Min.   :    0.00   Min.   :0.00000
 1st Qu.:    0.0        1st Qu.:    0.00   1st Qu.:0.08333
 Median :  136.9        Median :   99.97   Median :0.58333
 Mean   :  583.0        Mean   : 1548.17   Mean   :0.53583
 3rd Qu.:  722.0        3rd Qu.: 2406.23   3rd Qu.:1.00000
 Max.   :22500.0        Max.   :47137.21   Max.   :1.00000
 ONEOFF_PURCHASES_FREQUENCY PURCHASES_INSTALLMENTS_FREQUENCY CASH_ADVANCE_FREQUENCY
 Min.   :0.0000             Min.   :0.0000                   Min.   :0.00000
 1st Qu.:0.0000             1st Qu.:0.0000                   1st Qu.:0.00000
 Median :0.1000             Median :0.2500                   Median :0.08333
 Mean   :0.2859             Mean   :0.3861                   Mean   :0.16866
 3rd Qu.:0.5000             3rd Qu.:0.8333                   3rd Qu.:0.25000
 Max.   :1.0000             Max.   :1.0000                   Max.   :1.50000
 CASH_ADVANCE_TRX  PURCHASES_TRX    CREDIT_LIMIT      PAYMENTS
 Min.   :  0.000   Min.   :  0.0   Min.   :   50   Min.   :    0.06
 1st Qu.:  0.000   1st Qu.:  1.0   1st Qu.: 4050   1st Qu.:  767.24
 Median :  1.000   Median : 10.0   Median : 6000   Median : 1541.98
 Mean   :  4.465   Mean   : 20.6   Mean   : 6874   Mean   : 2699.31
 3rd Qu.:  6.000   3rd Qu.: 26.0   3rd Qu.: 8500   3rd Qu.: 3236.13
 Max.   :123.000   Max.   :358.0   Max.   :30000   Max.   :50721.48
 MINIMUM_PAYMENTS     PRC_FULL_PAYMENT      TENURE
 Min.   :    0.038   Min.   :0.0000   Min.   : 6.00
 1st Qu.:  184.799   1st Qu.:0.0000   1st Qu.:12.00
 Median :  490.378   Median :0.0000   Median :12.00
 Mean   :  943.930   Mean   :0.1743   Mean   :11.71
 3rd Qu.: 1192.463   3rd Qu.:0.1818   3rd Qu.:12.00
 Max.   :21235.065   Max.   :1.0000   Max.   :12.00
```

Second Subpopulation:



```
    BALANCE         BALANCE_FREQUENCY   PURCHASES       ONEOFF_PURCHASES
 Min.   :   0.00   Min.   :0.0000    Min.   :   0.0   Min.   :   0.0
 1st Qu.:  72.45   1st Qu.:0.8333    1st Qu.:  15.1   1st Qu.:   0.0
 Median : 528.22   Median :1.0000    Median : 245.4   Median :   0.0
 Mean   : 707.04   Mean   :0.8723    Mean   : 418.1   Mean   : 190.5
 3rd Qu.:1157.90   3rd Qu.:1.0000    3rd Qu.: 599.9   3rd Qu.: 200.0
 Max.   :2966.41   Max.   :1.0000    Max.   :2806.8   Max.   :2723.5
 INSTALLMENTS_PURCHASES  CASH_ADVANCE     PURCHASES_FREQUENCY
 Min.   :   0.00        Min.   :   0.0   Min.   :0.00000
 1st Qu.:   0.00        1st Qu.:   0.0   1st Qu.:0.08333
 Median :  63.19        Median :   0.0   Median :0.41667
 Mean   : 227.92        Mean   : 342.9   Mean   :0.44991
 3rd Qu.: 338.97        3rd Qu.: 458.1   3rd Qu.:0.87054
 Max.   :2749.92        Max.   :2988.1   Max.   :1.00000
 ONEOFF_PURCHASES_FREQUENCY PURCHASES_INSTALLMENTS_FREQUENCY CASH_ADVANCE_FREQUENCY
 Min.   :0.0000             Min.   :0.0000                   Min.   :0.0000
 1st Qu.:0.0000             1st Qu.:0.0000                   1st Qu.:0.0000
 Median :0.0000             Median :0.1667                   Median :0.0000
 Mean   :0.1149             Mean   :0.3473                   Mean   :0.1017
 3rd Qu.:0.1667             3rd Qu.:0.7500                   3rd Qu.:0.1667
 Max.   :1.0000             Max.   :1.0000                   Max.   :1.0000
 CASH_ADVANCE_TRX PURCHASES_TRX      CREDIT_LIMIT      PAYMENTS
 Min.   : 0.000   Min.   :  0.000   Min.   : 150   Min.   :   0.05
 1st Qu.: 0.000   1st Qu.:  1.000   1st Qu.:1200   1st Qu.: 281.27
 Median : 0.000   Median :  6.000   Median :1500   Median : 506.94
 Mean   : 1.963   Mean   :  8.484   Mean   :1769   Mean   : 719.31
 3rd Qu.: 2.000   3rd Qu.: 12.000   3rd Qu.:2500   3rd Qu.: 957.00
 Max.   :62.000   Max.   :186.000   Max.   :3500   Max.   :4644.14
 MINIMUM_PAYMENTS    PRC_FULL_PAYMENT      TENURE
 Min.   :   0.019   Min.   :0.0000   Min.   : 6.00
 1st Qu.: 148.363   1st Qu.:0.0000   1st Qu.:12.00
 Median : 220.594   Median :0.0000   Median :12.00
 Mean   : 413.100   Mean   :0.1446   Mean   :11.32
 3rd Qu.: 466.014   3rd Qu.:0.1250   3rd Qu.:12.00
 Max.   :5603.542   Max.   :1.0000   Max.   :12.00
```

Third Subpopulation:

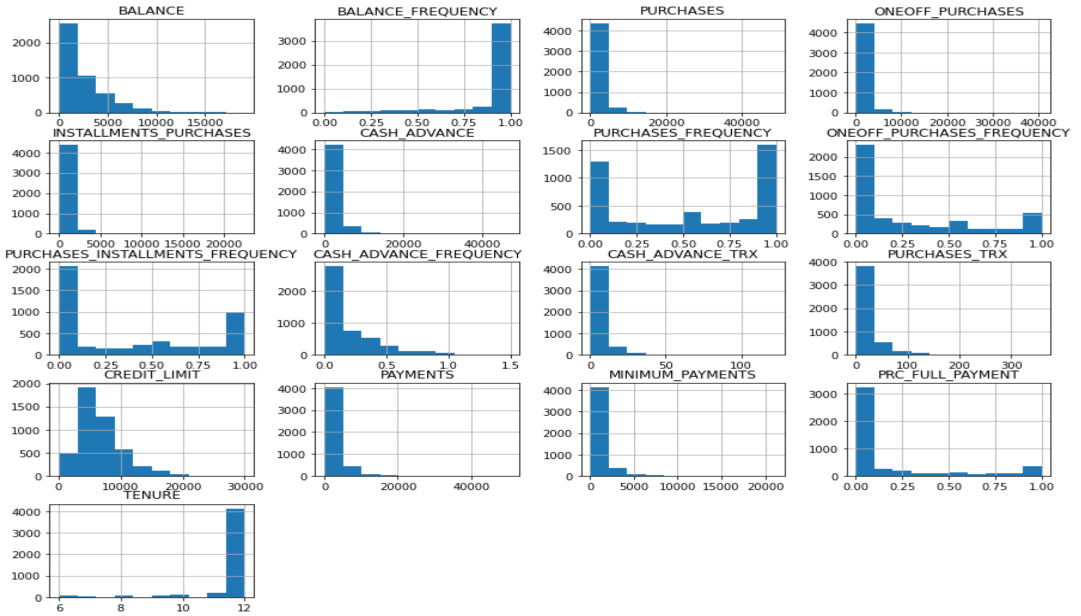


```
    BALANCE        BALANCE_FREQUENCY   PURCHASES      ONEOFF_PURCHASES
 Min.   :  915.7   Min.   :0.7273    Min.   :   0.0   Min.   :   0.00
 1st Qu.: 1555.1   1st Qu.:1.0000    1st Qu.:   0.0   1st Qu.:   0.00
 Median : 2473.5   Median :1.0000    Median : 190.0   Median :   0.00
 Mean   : 3190.3   Mean   :0.9868    Mean   : 577.9   Mean   :  80.23
 3rd Qu.: 4415.0   3rd Qu.:1.0000    3rd Qu.: 598.5   3rd Qu.:   0.00
 Max.   :10571.4   Max.   :1.0000    Max.   :7739.5   Max.   :2463.00
 INSTALLMENTS_PURCHASES  CASH_ADVANCE      PURCHASES_FREQUENCY
 Min.   :   0.0        Min.   :    0.00   Min.   :0.0000
 1st Qu.:   0.0        1st Qu.:    0.00   1st Qu.:0.0000
 Median : 101.0        Median :   19.35   Median :0.3333
 Mean   : 497.7        Mean   :  869.45   Mean   :0.4496
 3rd Qu.: 522.0        3rd Qu.: 1072.39   3rd Qu.:1.0000
 Max.   :7739.5        Max.   :10616.27   Max.   :1.0000
 ONEOFF_PURCHASES_FREQUENCY PURCHASES_INSTALLMENTS_FREQUENCY CASH_ADVANCE_FREQUENCY
 Min.   :0.0000             Min.   :0.0000                   Min.   :0.00000
 1st Qu.:0.0000             1st Qu.:0.0000                   1st Qu.:0.00000
 Median :0.0000             Median :0.2500                   Median :0.08333
 Mean   :0.0307             Mean   :0.4266                   Mean   :0.10022
 3rd Qu.:0.0000             3rd Qu.:1.0000                   3rd Qu.:0.16667
 Max.   :0.3333             Max.   :1.0000                   Max.   :0.50000
 CASH_ADVANCE_TRX PURCHASES_TRX    CREDIT_LIMIT      PAYMENTS
 Min.   : 0.000   Min.   :  0.0   Min.   : 1000   Min.   :  29.28
 1st Qu.: 0.000   1st Qu.:  0.0   1st Qu.: 1400   1st Qu.: 182.64
 Median : 1.000   Median :  6.0   Median : 2500   Median : 394.29
 Mean   : 2.948   Mean   : 14.1   Mean   : 3373   Mean   :1050.04
 3rd Qu.: 4.000   3rd Qu.: 14.0   3rd Qu.: 4500   3rd Qu.:1116.90
 Max.   :19.000   Max.   :162.0   Max.   :11000   Max.   :8735.61
 MINIMUM_PAYMENTS PRC_FULL_PAYMENT      TENURE
 Min.   : 8243    Min.   :0.000000   Min.   : 9.00
 1st Qu.:10830    1st Qu.:0.000000   1st Qu.:12.00
 Median :13916    Median :0.000000   Median :12.00
 Mean   :19079    Mean   :0.003247   Mean   :11.87
 3rd Qu.:22012    3rd Qu.:0.000000   3rd Qu.:12.00
 Max.   :76406    Max.   :0.083333   Max.   :12.00
```

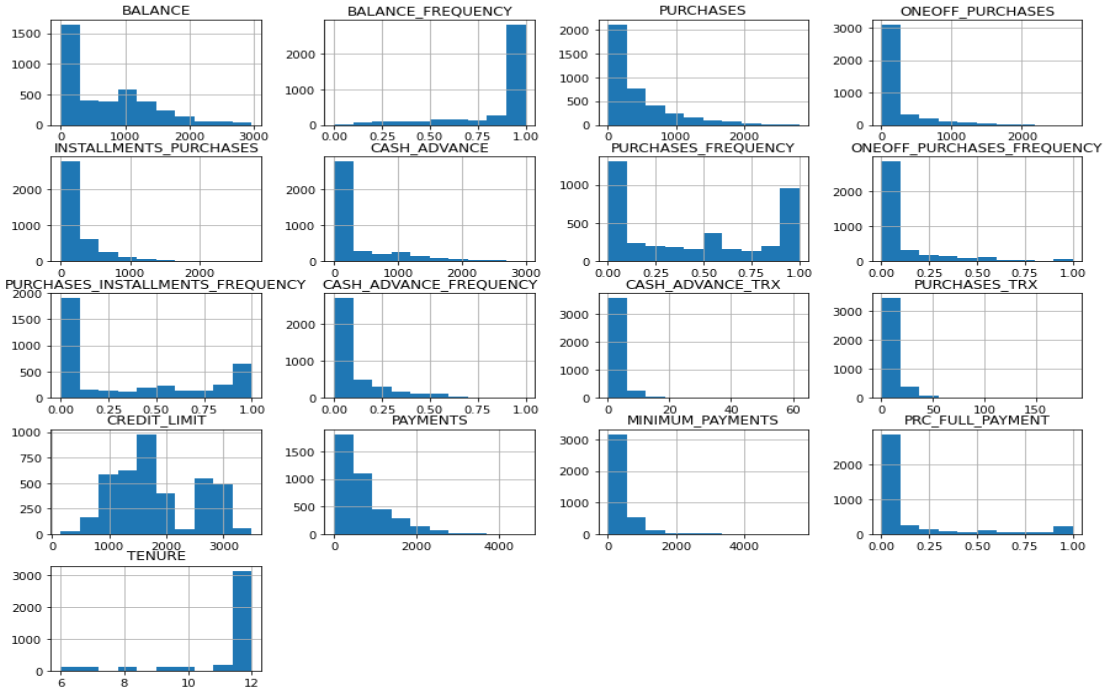
	Cluster 1	**Cluster 2**	**Cluster 3**
Purchases_TRX	At least 50% people have done 10 transactions	At least 50% people have done 6 transactions	At least 50% people have done 6 transactions
Credit Limit	Mean credit limit of card 6874	Mean credit limit of card 1769	Mean credit limit 3373
PRC Full Payment	Mean percent of full payment is 0.1743	Mean percent of full payment is 0.1446	Mean percent of full payment is 0.003
Cash Advance	Mean amount withdrawn from credit limit 1548.17	Mean amount withdrawn from credit limit 342.9	Mean amount withdrawn from credit limit 869.45
Balance	At least 50% of people have 1562.7 amount of money left for use	At least 50% of people have 528.22 amount of money left for use	At least 50% of people have 2473.5 amount of money left for use
Payments	Mean amount of payment done by card is 2699.31	Mean amount of payment done by card is 719.31	Mean amount of payment done by card is 1050.04

It is customary to see how the variables are distributed in each of the clusters which will help us have an insight of the behaviors of the customers.

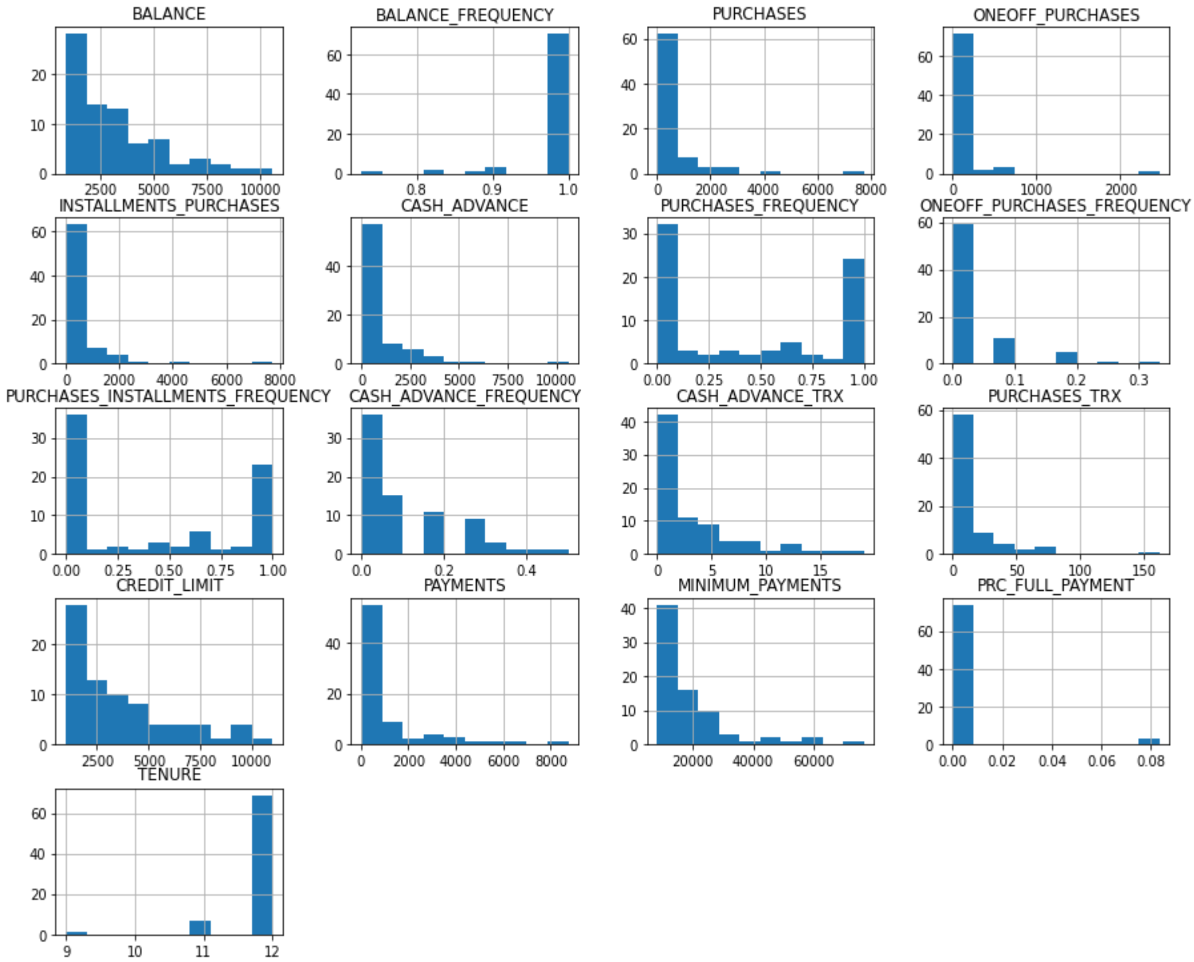
Cluster 1:



Cluster 2:



Cluster 3:



KMO Test Results:

KMO test is conducted to check whether our subpopulations are appropriate for factor analysis. Overall MSA below 0.6 is unacceptable. For cluster 1 and cluster 2 our overall MSA are 0.65 and 0.61 respectively. But for third cluster the overall MSA was 0.57 so we had to eliminate the feature ONEOFF_PURCHASES corresponding to lowest MSA for that cluster and further calculation of KMO test showed overall MSA to be 0.61. which means all of the three clusters can be passes to treatment of Factor Analysis. The variables with individual MSA <0.55 were dropped off before choosing the number of factors by Parallel Analysis.

Cluster	Eliminated Variables
Cluster 1	ONEOFF_PURCHASES, INSTALLMENT_PURCHASES
Cluster 2	TENURE,CREDIT_LIMIT,INSTALLMENTS_PURCHASES,ONEOFF_PURCHASES, PURCHASES,ONEOFF_PURCHASES_FREQUENCY
Cluster 3	BALANCE,ONEOFF_PURCHASES_FREQUENCY,BALANCE_FREQUENCY,CREDIT_LIMIT,ONEOFF_PURCHASES

Bartlett Sphericity Test results:

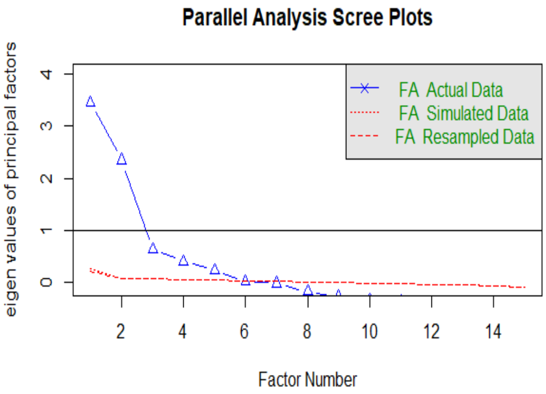
	Statistic	DF	P value
Cluster 1	91656.952	136	<2.22e-16
Cluster 2	69487.29	136	<2.22e-16
Cluster 3	1260.437	120	<2.22e-16

For all the clusters the Bartletts test rejects the null hypothesis which means all of them are ready to be passed to factor analysis.

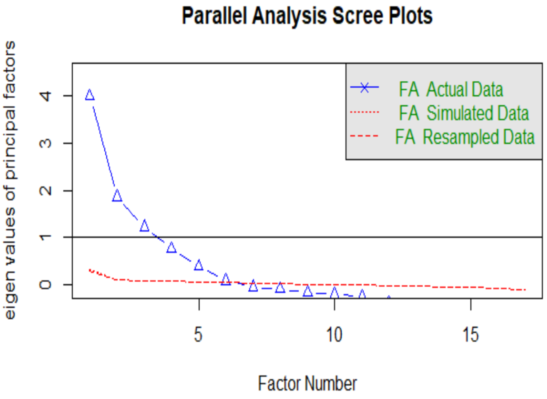
Results of Parallel Analysis:

Subpopulation	Number of Factors
Cluster 1	5
Cluster 2	4
Cluster 3	3

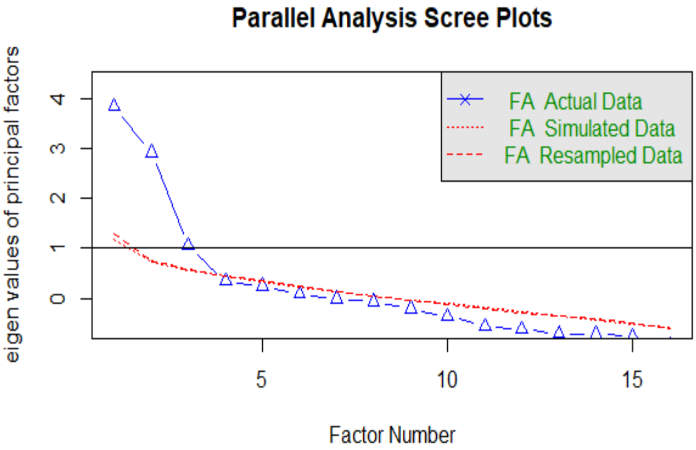
Cluster 1:



Cluster 2



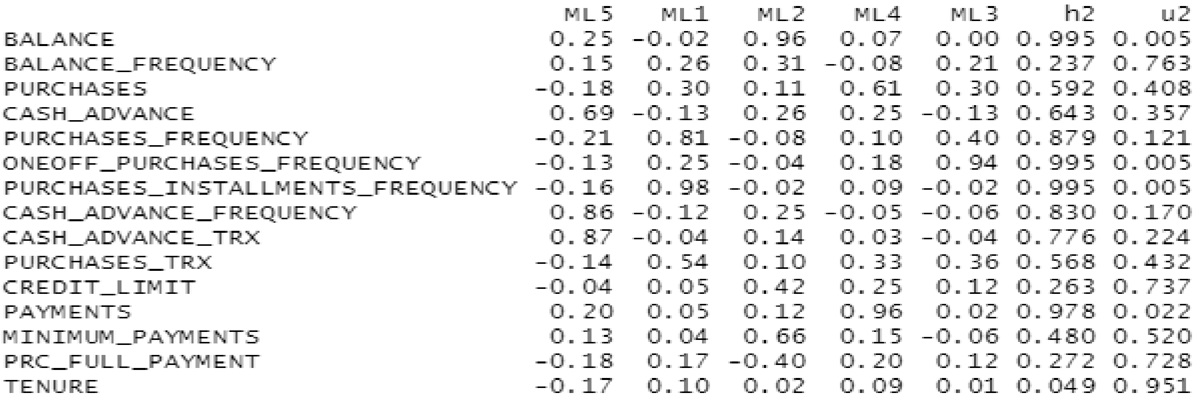
Cluster 3:



Factor Analysis Results:

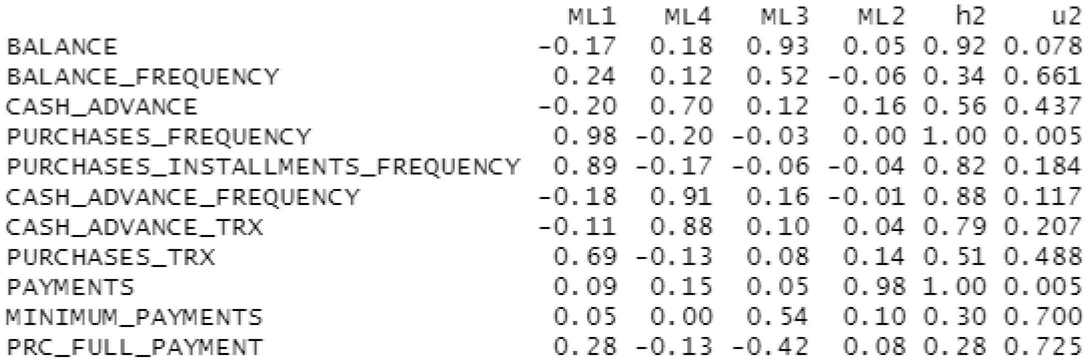
The analysis was executed by R’s psych library by selecting the parameter estimation method ‘maximum likelihood’ and the factor scores are calculated by regression method.

Loading Matrix for Cluster 1:



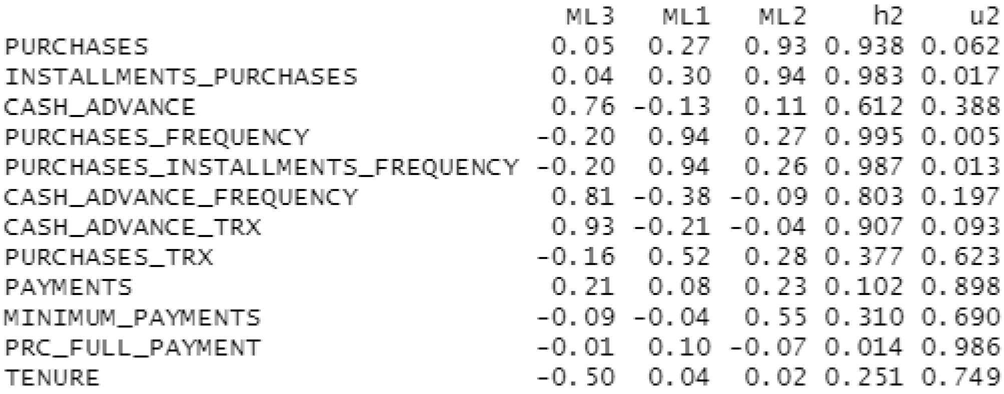
```
                                    ML5   ML1   ML2   ML4   ML3    h2    u2
BALANCE                            0.25 -0.02  0.96  0.07  0.00 0.995 0.005
BALANCE_FREQUENCY                  0.15  0.26  0.31 -0.08  0.21 0.237 0.763
PURCHASES                         -0.18  0.30  0.11  0.61  0.30 0.592 0.408
CASH_ADVANCE                       0.69 -0.13  0.26  0.25 -0.13 0.643 0.357
PURCHASES_FREQUENCY               -0.21  0.81 -0.08  0.10  0.40 0.879 0.121
ONEOFF_PURCHASES_FREQUENCY        -0.13  0.25 -0.04  0.18  0.94 0.995 0.005
PURCHASES_INSTALLMENTS_FREQUENCY -0.16  0.98 -0.02  0.09 -0.02 0.995 0.005
CASH_ADVANCE_FREQUENCY             0.86 -0.12  0.25 -0.05 -0.06 0.830 0.170
CASH_ADVANCE_TRX                   0.87 -0.04  0.14  0.03 -0.04 0.776 0.224
PURCHASES_TRX                     -0.14  0.54  0.10  0.33  0.36 0.568 0.432
CREDIT_LIMIT                      -0.04  0.05  0.42  0.25  0.12 0.263 0.737
PAYMENTS                           0.20  0.05  0.12  0.96  0.02 0.978 0.022
MINIMUM_PAYMENTS                   0.13  0.04  0.66  0.15 -0.06 0.480 0.520
PRC_FULL_PAYMENT                  -0.18  0.17 -0.40  0.20  0.12 0.272 0.728
TENURE                            -0.17  0.10  0.02  0.09  0.01 0.049 0.951
```

Loading Matrix for Cluster 2:



```
                                   ML1   ML4   ML3   ML2   h2    u2
BALANCE                          -0.17  0.18  0.93  0.05 0.92 0.078
BALANCE_FREQUENCY                 0.24  0.12  0.52 -0.06 0.34 0.661
CASH_ADVANCE                     -0.20  0.70  0.12  0.16 0.56 0.437
PURCHASES_FREQUENCY               0.98 -0.20 -0.03  0.00 1.00 0.005
PURCHASES_INSTALLMENTS_FREQUENCY  0.89 -0.17 -0.06 -0.04 0.82 0.184
CASH_ADVANCE_FREQUENCY           -0.18  0.91  0.16 -0.01 0.88 0.117
CASH_ADVANCE_TRX                 -0.11  0.88  0.10  0.04 0.79 0.207
PURCHASES_TRX                     0.69 -0.13  0.08  0.14 0.51 0.488
PAYMENTS                          0.09  0.15  0.05  0.98 1.00 0.005
MINIMUM_PAYMENTS                  0.05  0.00  0.54  0.10 0.30 0.700
PRC_FULL_PAYMENT                  0.28 -0.13 -0.42  0.08 0.28 0.725
```

Loading Matrix for Cluster 3:



```
                                   ML3   ML1   ML2    h2    u2
PURCHASES                         0.05  0.27  0.93 0.938 0.062
INSTALLMENTS_PURCHASES            0.04  0.30  0.94 0.983 0.017
CASH_ADVANCE                      0.76 -0.13  0.11 0.612 0.388
PURCHASES_FREQUENCY              -0.20  0.94  0.27 0.995 0.005
PURCHASES_INSTALLMENTS_FREQUENCY -0.20  0.94  0.26 0.987 0.013
CASH_ADVANCE_FREQUENCY            0.81 -0.38 -0.09 0.803 0.197
CASH_ADVANCE_TRX                  0.93 -0.21 -0.04 0.907 0.093
PURCHASES_TRX                    -0.16  0.52  0.28 0.377 0.623
PAYMENTS                          0.21  0.08  0.23 0.102 0.898
MINIMUM_PAYMENTS                 -0.09 -0.04  0.55 0.310 0.690
PRC_FULL_PAYMENT                 -0.01  0.10 -0.07 0.014 0.986
TENURE                           -0.50  0.04  0.02 0.251 0.749
```

For each of the three clusters h2, u2 and com represents the communality, uniqueness and Hoffman’s complexity. We can think of the communalities as value of R-square for regression models predicting the variables of interest from chosen number of factors for each cluster. In a nutshell, communalities represent the proportion of variance in a variable explained by the factors Communalities help us to assess how well the model performs. Total of communalities divided by the number of variables for each cluster shows proportion of total variance explained by the factors.

	Number of Factors	Proportion of Total Variation Explained
Cluster 1	6	0.636
Cluster 2	6	0.672
Cluster 3	3	0.606

The Hoffman's index of complexity tells us how much an item reflects a single construct. It equals one if a variable loads only on one factor and two if a variable loads evenly on two factors, so on. It will be lower for relatively lower loadings.

Next three tables represent the proportion of variance explained by the factors for each three clusters. Proportion of variance explained is calculated by subtracting Proportion variance(how much overall variance the factors accounts) from the Proportion variance and dividing it by sum of proportion variances of factors.

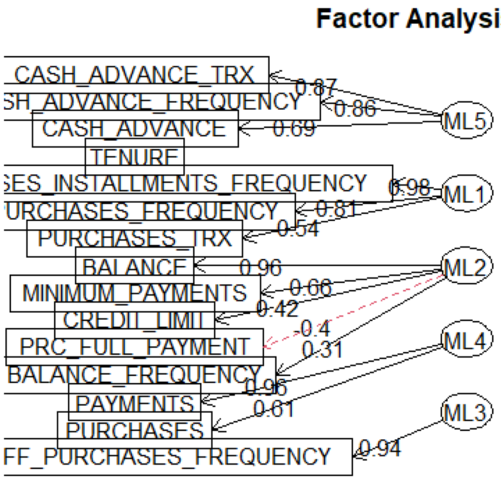
	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5
Proportion of Variance	0.16	0.15	0.13	0.11	0.09
Proportion of Variance Explained	0.24	0.23	0.21	0.17	0.14

	Factor 1	Factor 2	Factor 3	Factor 4
Proportion of Variance Explained	0.33	0.30	0.22	0.14
Proportion of Variance	0.22	0.20	0.15	0.09

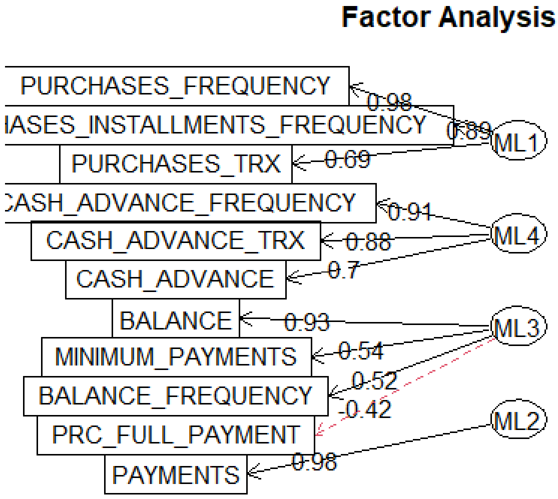
	Factor 1	Factor 2	Factor 3
Proportion of Variance Explained	0.34	0.33	0.32
Prop of Variance	0.21	0.20	0.20

Following diagrams show that how the different factors for each cluster summarizes the features

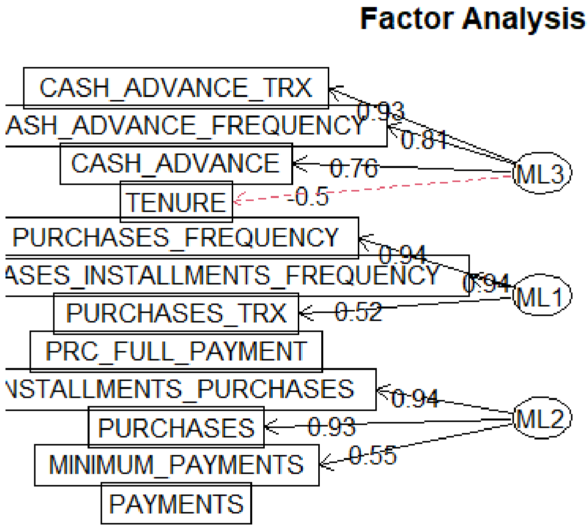
For cluster 1:



For Cluster 2:



For Cluster 3:



Measures of Lack Of Fit:

	Tucker-Lewis Index	RMSEA Index	Root Mean Square Residuals	BIC
Cluster 1	0.822	0.119	0.03	2325.53
Cluster 2	0.978	0.048	0.01	29.1
Cluster 3	0.954	0.079	0.13	-94.07

Here our chosen method is maximum likelihood among other alternatives the maximum likelihood was free from Heywood problem which makes the loadings of the correlation matrix of features and factors more than 1 plus this method gives us reduction of BIC than any other method. Moreover it is better than principal component method because it gives us measures of model's lack of fit which is essential to see how well our model performs. The root mean square of residuals value below 0.08 is considered good. The Tucker Lewis Index > 0.9 suggests very good fit of the model, the value near 0.5 indicates moderate fit of the model. And the suggested value of RMSEA Index for good fit is <0.08.

Based on Kernel k-Means:-

Data Preparation:

Data Preparation is done in two steps.

Firstly, Customer ID is an unique id for each customer and hence won't play any role in determining the clusters. So, we have dropped it.

Secondly, we drop the rows corresponding to the missing value in any column.

Determining no. of clusters:-

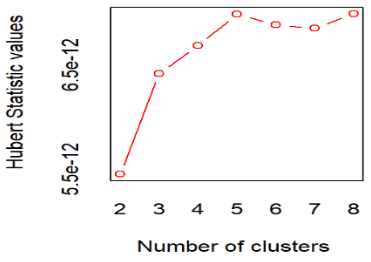
i)Analytical Method:-

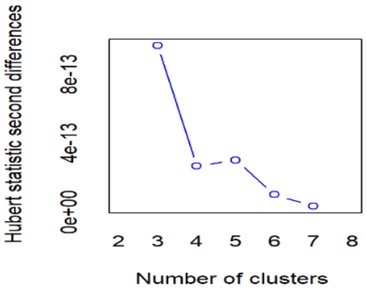
For determining the best no. of clusters, we have used NbClust Package of R Programming language. Here, we have found the best no. of clusters to be 3 based on the majority rule.

ii)Graphical Method:-

a)Hubert Index Method:-

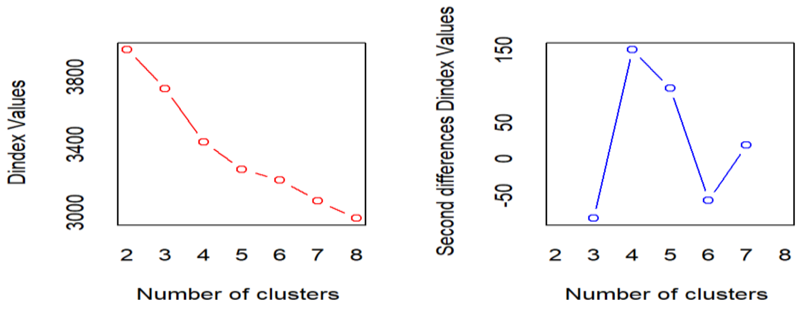
The Hubert Index is a graphical method of determining the number of clusters. In the plot of Hubert index, we seek a significant knee that corresponds to a significant increase of the value of the measure i.e., the significant peak in Hubert index second differences plot. Here, we have found the no. of clusters to be 3.





b) D Index Method:-

The D Index is a graphical method of determining the number of clusters.In the plot of D index, we seek a significant knee (the significant peak in D index second differences plot) that corresponds to a significant increase of the value of the measure. Here, we have found the no. of clusters to be 3.



One drawback of Kernel K-means is that number of cluster centers need to be predefined. Hence by using both the analytical method and the graphical methods, we come to a conclusion that the optimal number of clusters is 3. In the next step, setting the number of clusters as 3, we applied three most popular kernel transformations to the dataset. For evaluating the density and separation between the clusters we calculated the corresponding Silhouette Scores (lies between -1 to 1 where score near 1 implies high separation, score near 0 indicates overlapping and score near -1 indicates wrong cluster assignment) for these 3 non-linear splitting of the dataset. The scores are given below

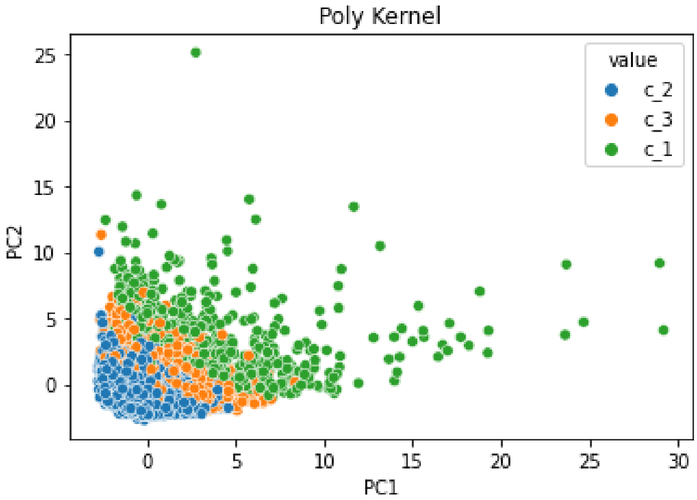
Kernel	**Silhouette Score**
Polynomial	0.33422278610924916
Radial Basis Function (Gaussian)	0.035308683417481106
Hyperbolic	-0.012615015542211127

The silhouette coefficients suggests that the Gaussian Kernel and Hyperbolic kernel methods will be having high overlapping among the clusters whereas the Polynomial Kernel performs better separation than the previous Kernel functions. From these results we select the Polynomial Kernel as the best kernel transformation for this dataset and proceed with our further analysis.

The cluster sizes we get for this splitting,

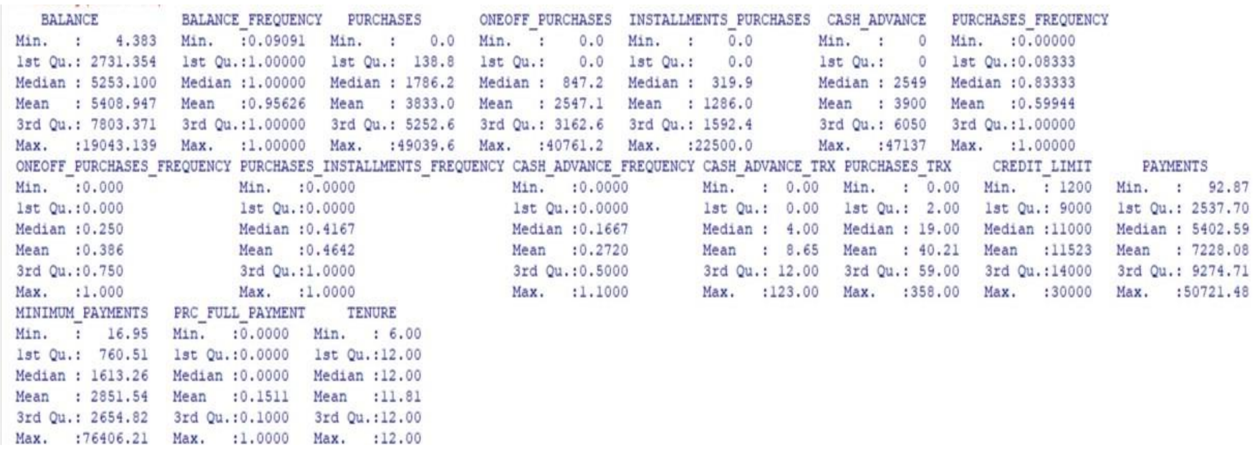
Cluster 1	Cluster 2	Cluster 3
758	4794	3084

To get a visualization of the clustering labels one can use PCA (Principal Component Analysis) to project the data in lower dimension and visualize how the data is clustered. Below is the two dimensional such plot done with first two principal components.



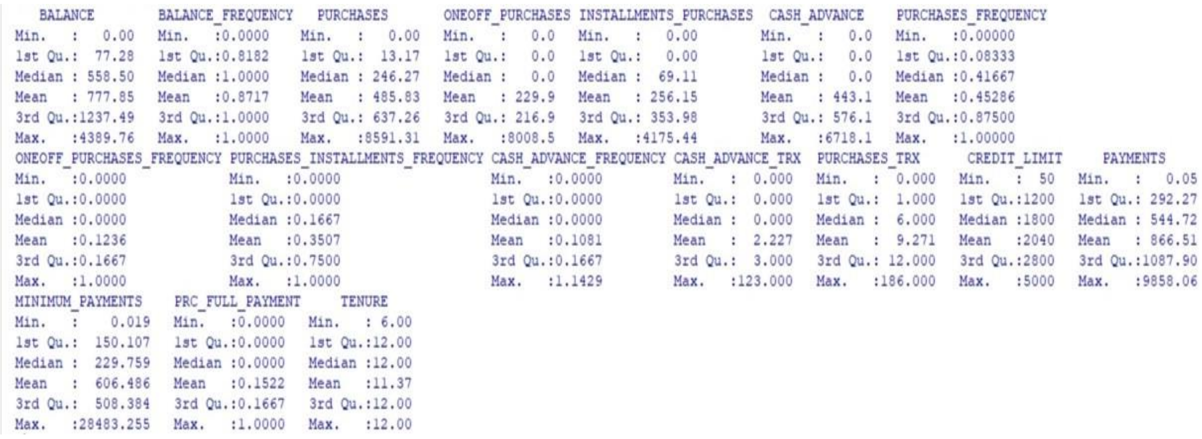
First Subpopulation:

Summary



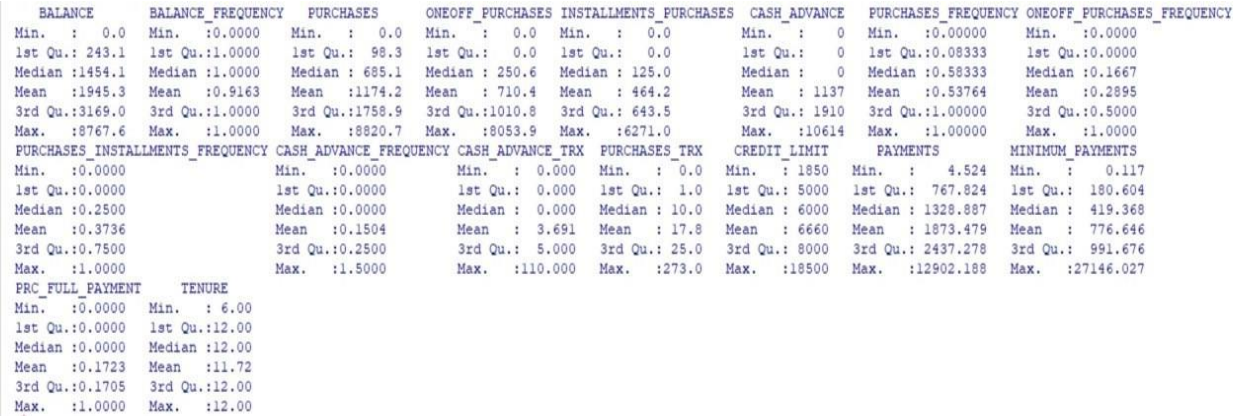
```
   BALANCE          BALANCE_FREQUENCY   PURCHASES        ONEOFF_PURCHASES  INSTALLMENTS_PURCHASES  CASH_ADVANCE    PURCHASES_FREQUENCY
 Min.   :    4.383  Min.   :0.09091   Min.   :    0.0   Min.   :    0.0   Min.   :    0.0        Min.   :    0   Min.   :0.00000
 1st Qu.: 2731.354  1st Qu.:1.00000   1st Qu.:  138.8   1st Qu.:    0.0   1st Qu.:    0.0        1st Qu.:    0   1st Qu.:0.08333
 Median : 5253.100  Median :1.00000   Median : 1786.2   Median :  847.2   Median :  319.9        Median : 2549   Median :0.83333
 Mean   : 5408.947  Mean   :0.95626   Mean   : 3833.0   Mean   : 2547.1   Mean   : 1286.0        Mean   : 3900   Mean   :0.59944
 3rd Qu.: 7803.371  3rd Qu.:1.00000   3rd Qu.: 5252.6   3rd Qu.: 3162.6   3rd Qu.: 1592.4        3rd Qu.: 6050   3rd Qu.:1.00000
 Max.   :19043.139  Max.   :1.00000   Max.   :49039.6   Max.   :40761.2   Max.   :22500.0        Max.   :47137   Max.   :1.00000
 ONEOFF_PURCHASES_FREQUENCY PURCHASES_INSTALLMENTS_FREQUENCY CASH_ADVANCE_FREQUENCY CASH_ADVANCE_TRX PURCHASES_TRX    CREDIT_LIMIT      PAYMENTS
 Min.   :0.000              Min.   :0.0000                   Min.   :0.0000         Min.   :  0.00   Min.   :  0.00   Min.   : 1200   Min.   :   92.87
 1st Qu.:0.000              1st Qu.:0.0000                   1st Qu.:0.0000         1st Qu.:  0.00   1st Qu.:  2.00   1st Qu.: 9000   1st Qu.: 2537.70
 Median :0.250              Median :0.4167                   Median :0.1667         Median :  4.00   Median : 19.00   Median :11000   Median : 5402.59
 Mean   :0.386              Mean   :0.4642                   Mean   :0.2720         Mean   :  8.65   Mean   : 40.21   Mean   :11523   Mean   : 7228.08
 3rd Qu.:0.750              3rd Qu.:1.0000                   3rd Qu.:0.5000         3rd Qu.: 12.00   3rd Qu.: 59.00   3rd Qu.:14000   3rd Qu.: 9274.71
 Max.   :1.000              Max.   :1.0000                   Max.   :1.1000         Max.   :123.00   Max.   :358.00   Max.   :30000   Max.   :50721.48
 MINIMUM_PAYMENTS   PRC_FULL_PAYMENT     TENURE
 Min.   :   16.95   Min.   :0.0000   Min.   : 6.00
 1st Qu.:  760.51   1st Qu.:0.0000   1st Qu.:12.00
 Median : 1613.26   Median :0.0000   Median :12.00
 Mean   : 2851.54   Mean   :0.1511   Mean   :11.81
 3rd Qu.: 2654.82   3rd Qu.:0.1000   3rd Qu.:12.00
 Max.   :76406.21   Max.   :1.0000   Max.   :12.00
```

Second Subpopulation:



```
   BALANCE        BALANCE_FREQUENCY   PURCHASES         ONEOFF_PURCHASES INSTALLMENTS_PURCHASES  CASH_ADVANCE     PURCHASES_FREQUENCY
 Min.   :   0.00  Min.   :0.0000    Min.   :   0.00   Min.   :   0.0   Min.   :   0.00        Min.   :   0.0   Min.   :0.00000
 1st Qu.:  77.28  1st Qu.:0.8182    1st Qu.:  13.17   1st Qu.:   0.0   1st Qu.:   0.00        1st Qu.:   0.0   1st Qu.:0.08333
 Median : 558.50  Median :1.0000    Median : 246.27   Median :   0.0   Median :  69.11        Median :   0.0   Median :0.41667
 Mean   : 777.85  Mean   :0.8717    Mean   : 485.83   Mean   : 229.9   Mean   : 256.15        Mean   : 443.1   Mean   :0.45286
 3rd Qu.:1237.49  3rd Qu.:1.0000    3rd Qu.: 637.26   3rd Qu.: 216.9   3rd Qu.: 353.98        3rd Qu.: 576.1   3rd Qu.:0.87500
 Max.   :4389.76  Max.   :1.0000    Max.   :8591.31   Max.   :8008.5   Max.   :4175.44        Max.   :6718.1   Max.   :1.00000
 ONEOFF_PURCHASES_FREQUENCY PURCHASES_INSTALLMENTS_FREQUENCY CASH_ADVANCE_FREQUENCY CASH_ADVANCE_TRX  PURCHASES_TRX     CREDIT_LIMIT    PAYMENTS
 Min.   :0.0000             Min.   :0.0000                   Min.   :0.0000         Min.   :  0.000   Min.   :  0.000   Min.   :  50   Min.   :   0.05
 1st Qu.:0.0000             1st Qu.:0.0000                   1st Qu.:0.0000         1st Qu.:  0.000   1st Qu.:  1.000   1st Qu.:1200   1st Qu.: 292.27
 Median :0.0000             Median :0.1667                   Median :0.0000         Median :  0.000   Median :  6.000   Median :1800   Median : 544.72
 Mean   :0.1236             Mean   :0.3507                   Mean   :0.1081         Mean   :  2.227   Mean   :  9.271   Mean   :2040   Mean   : 866.51
 3rd Qu.:0.1667             3rd Qu.:0.7500                   3rd Qu.:0.1667         3rd Qu.:  3.000   3rd Qu.: 12.000   3rd Qu.:2800   3rd Qu.:1087.90
 Max.   :1.0000             Max.   :1.0000                   Max.   :1.1429         Max.   :123.000   Max.   :186.000   Max.   :5000   Max.   :9858.06
 MINIMUM_PAYMENTS    PRC_FULL_PAYMENT     TENURE
 Min.   :    0.019   Min.   :0.0000   Min.   : 6.00
 1st Qu.:  150.107   1st Qu.:0.0000   1st Qu.:12.00
 Median :  229.759   Median :0.0000   Median :12.00
 Mean   :  606.486   Mean   :0.1522   Mean   :11.37
 3rd Qu.:  508.384   3rd Qu.:0.1667   3rd Qu.:12.00
 Max.   :28483.255   Max.   :1.0000   Max.   :12.00
```

Third Subpopulation:



```
    BALANCE       BALANCE_FREQUENCY   PURCHASES       ONEOFF_PURCHASES INSTALLMENTS_PURCHASES  CASH_ADVANCE   PURCHASES_FREQUENCY ONEOFF_PURCHASES_FREQUENCY
 Min.   :   0.0   Min.   :0.0000    Min.   :   0.0   Min.   :   0.0   Min.   :   0.0        Min.   :    0   Min.   :0.00000     Min.   :0.0000   
 1st Qu.: 243.1   1st Qu.:1.0000    1st Qu.:  98.3   1st Qu.:   0.0   1st Qu.:   0.0        1st Qu.:    0   1st Qu.:0.08333     1st Qu.:0.0000   
 Median :1454.1   Median :1.0000    Median : 685.1   Median : 250.6   Median : 125.0        Median :    0   Median :0.58333     Median :0.1667   
 Mean   :1945.3   Mean   :0.9163    Mean   :1174.2   Mean   : 710.4   Mean   : 464.2        Mean   : 1137   Mean   :0.53764     Mean   :0.2895   
 3rd Qu.:3169.0   3rd Qu.:1.0000    3rd Qu.:1758.9   3rd Qu.:1010.8   3rd Qu.: 643.5        3rd Qu.: 1910   3rd Qu.:1.00000     3rd Qu.:0.5000   
 Max.   :8767.6   Max.   :1.0000    Max.   :8820.7   Max.   :8053.9   Max.   :6271.0        Max.   :10614   Max.   :1.00000     Max.   :1.0000   
 PURCHASES_INSTALLMENTS_FREQUENCY CASH_ADVANCE_FREQUENCY CASH_ADVANCE_TRX  PURCHASES_TRX    CREDIT_LIMIT      PAYMENTS          MINIMUM_PAYMENTS  
 Min.   :0.0000                   Min.   :0.0000         Min.   :  0.000   Min.   :  0.0   Min.   : 1850   Min.   :    4.524   Min.   :    0.117  
 1st Qu.:0.0000                   1st Qu.:0.0000         1st Qu.:  0.000   1st Qu.:  1.0   1st Qu.: 5000   1st Qu.:  767.824   1st Qu.:  180.604  
 Median :0.2500                   Median :0.0000         Median :  0.000   Median : 10.0   Median : 6000   Median : 1328.887   Median :  419.368  
 Mean   :0.3736                   Mean   :0.1504         Mean   :  3.691   Mean   : 17.8   Mean   : 6660   Mean   : 1873.479   Mean   :  776.646  
 3rd Qu.:0.7500                   3rd Qu.:0.2500         3rd Qu.:  5.000   3rd Qu.: 25.0   3rd Qu.: 8000   3rd Qu.: 2437.278   3rd Qu.:  991.676  
 Max.   :1.0000                   Max.   :1.5000         Max.   :110.000   Max.   :273.0   Max.   :18500   Max.   :12902.188   Max.   :27146.027  
 PRC_FULL_PAYMENT     TENURE     
 Min.   :0.0000   Min.   : 6.00  
 1st Qu.:0.0000   1st Qu.:12.00  
 Median :0.0000   Median :12.00  
 Mean   :0.1723   Mean   :11.72  
 3rd Qu.:0.1705   3rd Qu.:12.00  
 Max.   :1.0000   Max.   :12.00  
```

Variables	**Cluster 1**	**Cluster 2**	**Cluster 3**
Balance	5408.9 RS. is the Mean balance amount left for use.	777.8 RS. is the Mean balance left for use.	1945.2 RS. is the Mean balance left for use.
Purchases	3832.9 Rs. is the mean amount spent in purchases.	485.8 Rs. is the mean amount spent in purchases.	1174.1 Rs. is the mean amount spent in purchases.
ONEOFF_PURCHASES	2547.1 Rs. Is the mean Amount of purchase done in one-go.	229.9 Rs. Is the mean Amount of purchase done in one-go.	710.4 Rs. Is the mean Amount of purchase done in one-go.
Frequency	0.599 is the mean purchase frequency, 0.386 is the mean One-off purchase frequency and 0.464 is the installment purchase frequency.	0.453 is the mean purchase frequency, 0.124 is the mean One-off purchase frequency and 0.350 is the installment purchase frequency.	0.537 is the mean purchase frequency, 0.289 is the mean One-off purchase frequency and 0.373 is the installment purchase frequency.
PURCHASES_TRX	40.2 is the mean number of purchase transactions made.	9.3 is the mean number of purchase transactions made.	17.8 is the mean number of purchase transactions made.
CREDIT_LIMIT	11000 Rs is the median credit limit.	1800 Rs is the median credit limit.	6000 Rs is the median credit limit.
PAYMENTS	7228.08 Rs. is the mean amount of payment done.	866.5 Rs. is the mean amount of payment done.	1873.4 Rs. is the mean amount of payment done.
PRC_FULL_PAYMENT	15.1 % is the mean percentage of full payment.	15.2 % is the mean percentage of full payment.	17.2 % is the mean percentage of full payment.

Factor Adequacy Check:

Kaiser-Meyer-Olkin (KMO) Test results

Cluster 1	Cluster 2	Cluster 3
Overall MSA = 0.67	Overall MSA = 0.62	Overall MSA = 0.68

Overall MSA below 0.6 is unacceptable. Here for all the clusters overall MSA is higher than 0.6. Hence the clusters can be passed to the further treatment of Factor Analysis. The variables with individual MSA ≤ 0.50 were dropped off before choosing the number of factors by Parallel Analysis.

Cluster	Eliminated Variables
Cluster 1	INSTALLMENT_PURCHASES, CREDIT_LIMIT, MINIMUM_PAYMENTS
Cluster 2	TENURE, CREDIT_LIMIT, ONEOFF_PURCHASES, INSTALLMENTS_PURCHASES, ONEOFF_PURCHASES_FREQUENCY
Cluster 3	CREDIT_LIMIT, ONEOFF_PURCHASES, INSTALLMENTS_PURCHASES

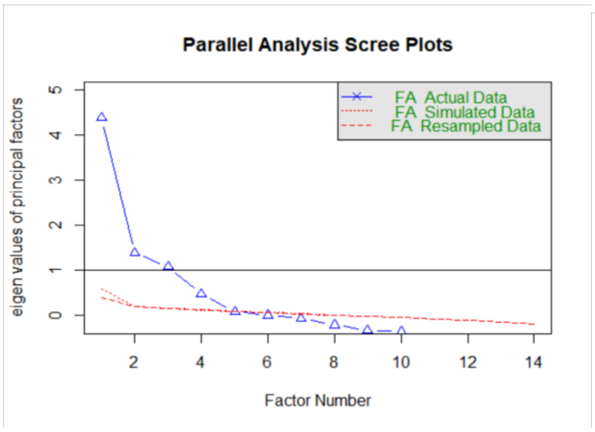
Bartlett's Test of Sphericity Results:

Cluster	Chi – square statistic	DF	P - value
Cluster 1	7329.476	91	2.22e-16
Cluster 2	30184.393	66	2.22e-16
Cluster 3	23976.578	91	2.22e-16

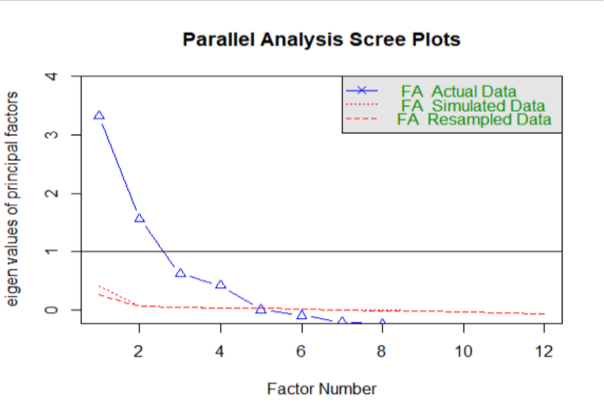
For all the clusters the p value is smaller than our level of significance (say 0.05). So, we reject the null hypothesis and conclude that all the subpopulations are suitable for Factor Analysis.

Results of Parallel Analysis:

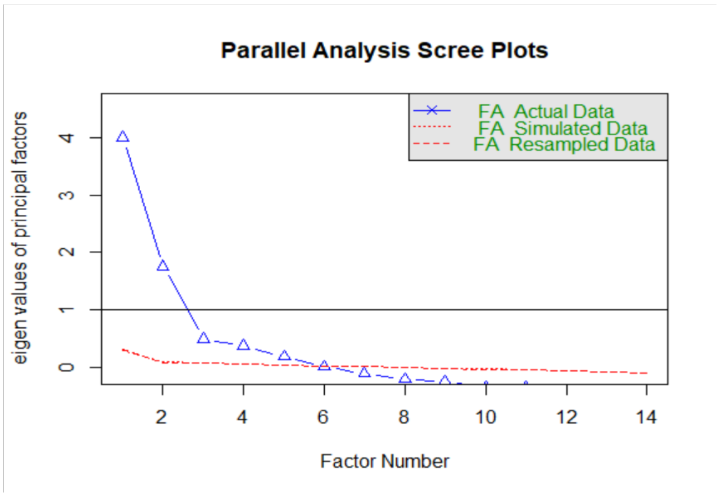
Cluster 1:



Cluster 2:



Cluster 3:



Cluster 1: Parallel analysis suggests that the number of factors = 4

Cluster 2: Parallel analysis suggests that the number of factors = 4

Cluster 3: Parallel analysis suggests that the number of factors = 5

Factor Analysis results:

Loading matrix for cluster 1:

	ML1	ML2	ML3	ML4	h2	u2	com
BALANCE	-0.08	0.09	0.07	0.81	0.663	0.337	1.1
BALANCE_FREQUENCY	0.32	0.07	0.13	0.43	0.278	0.722	2.1
PURCHASES	0.07	0.95	-0.09	0.05	0.995	0.005	1.0
ONEOFF_PURCHASES	-0.01	0.92	-0.06	0.01	0.863	0.137	1.0
CASH_ADVANCE	-0.12	-0.04	0.73	-0.04	0.622	0.378	1.1
PURCHASES_FREQUENCY	1.00	-0.02	-0.01	-0.05	0.995	0.005	1.0
ONEOFF_PURCHASES_FREQUENCY	0.60	0.22	-0.04	-0.17	0.599	0.401	1.4
PURCHASES_INSTALLMENTS_FREQUENCY	0.88	-0.02	0.00	0.05	0.755	0.245	1.0
CASH_ADVANCE_FREQUENCY	-0.02	-0.07	0.83	0.13	0.795	0.205	1.1
CASH_ADVANCE_TRX	0.07	0.00	0.89	0.00	0.750	0.250	1.0
PURCHASES_TRX	0.43	0.44	-0.05	0.06	0.540	0.460	2.1
PAYMENTS	-0.12	0.63	0.33	-0.25	0.501	0.499	2.0
PRC_FULL_PAYMENT	0.06	0.20	-0.05	-0.61	0.513	0.487	1.3
TENURE	0.10	0.01	-0.11	0.02	0.028	0.972	2.1

Loading matrix for cluster 2:

```
                                      ML2   ML4   ML1   ML3   h2    u2 com
BALANCE                             -0.05  0.02  0.00  0.96 0.95 0.054 1.0
BALANCE_FREQUENCY                    0.32  0.07  0.01  0.49 0.31 0.695 1.8
PURCHASES                            0.02 -0.06  0.98  0.00 1.00 0.005 1.0
CASH_ADVANCE                        -0.05  0.70 -0.02  0.07 0.56 0.441 1.0
PURCHASES_FREQUENCY                  0.93 -0.03  0.08 -0.03 0.96 0.041 1.0
PURCHASES_INSTALLMENTS_FREQUENCY     0.95  0.00 -0.07 -0.02 0.86 0.140 1.0
CASH_ADVANCE_FREQUENCY              -0.03  0.87 -0.02  0.04 0.82 0.182 1.0
CASH_ADVANCE_TRX                     0.05  0.91  0.00 -0.04 0.78 0.221 1.0
PURCHASES_TRX                        0.47 -0.02  0.45  0.07 0.61 0.387 2.0
PAYMENTS                            -0.12  0.28  0.56 -0.03 0.32 0.681 1.6
MINIMUM_PAYMENTS                     0.07 -0.10 -0.01  0.40 0.14 0.858 1.2
PRC_FULL_PAYMENT                     0.20  0.02  0.08 -0.41 0.25 0.745 1.5
```

Loading matrix for cluster 3:

```
                                      ML4   ML1   ML2   ML3   ML5    h2    u2 com
BALANCE                              0.02 -0.02  0.98  0.00  0.00 0.995 0.005 1.0
BALANCE_FREQUENCY                    0.09  0.27  0.39  0.23 -0.04 0.265 0.735 2.7
PURCHASES                           -0.14  0.12  0.02  0.19  0.69 0.789 0.211 1.3
CASH_ADVANCE                         0.67 -0.08  0.10 -0.08  0.03 0.604 0.396 1.1
PURCHASES_FREQUENCY                 -0.04  0.72 -0.06  0.41 -0.03 0.903 0.097 1.6
ONEOFF_PURCHASES_FREQUENCY          -0.02 -0.05 -0.03  0.92  0.10 0.931 0.069 1.0
PURCHASES_INSTALLMENTS_FREQUENCY    -0.01  1.00 -0.02 -0.15  0.07 0.995 0.005 1.1
CASH_ADVANCE_FREQUENCY               0.91  0.00  0.05  0.02 -0.02 0.881 0.119 1.0
CASH_ADVANCE_TRX                     0.94  0.04 -0.05  0.03  0.01 0.796 0.204 1.0
PURCHASES_TRX                       -0.05  0.37  0.05  0.18  0.47 0.687 0.313 2.3
CREDIT_LIMIT                        -0.14  0.03  0.09  0.13 -0.20 0.051 0.949 3.1
PAYMENTS                             0.18 -0.11 -0.06 -0.09  0.63 0.323 0.677 1.3
MINIMUM_PAYMENTS                    -0.04  0.02  0.54 -0.08  0.05 0.284 0.716 1.1
PRC_FULL_PAYMENT                    -0.03  0.08 -0.44  0.10  0.07 0.279 0.721 1.2
TENURE                              -0.22  0.06  0.05 -0.04  0.07 0.053 0.947 1.5
```

The analysis was executed by R's psych library by selecting the parameter estimation method 'maximum likelihood' and the factor scores are calculated by regression method. For each of the three clusters h2, u2 and com represents the communality, uniqueness and Hoffman's complexity. Communality is the proportion of each variable's variance that can be explained by the factors. It is also noted as h2 and can be defined as the sum of squared factor loadings for the variables. Uniqueness is the variance that is 'unique' to the variable and not shared with other variables. It is equal to 1 – communality (variance that is shared with other variables). Hoffman's index of Complexity represents the number of latent components needed to account for the observed variables. Whereas a perfect simple structure solution has a complexity of 1 in that each item would only load on one factor, a solution with evenly distributed items has a complexity greater than 1.

Next three tables represent the proportion of variance explained by the factors for each three clusters. Proportion of variance explained is calculated by subtracting Proportion variance (how much overall variance the factors accounts) from the Proportion variance and dividing it by sum of proportion variances of factors.

Cluster 1:

	Factor 1	Factor 2	Factor 3	Factor 4
Proportion of Variance Explained	0.30	0.29	0.25	0.16
Proportion of Variance	0.19	0.19	0.16	0.10

Cluster 2:

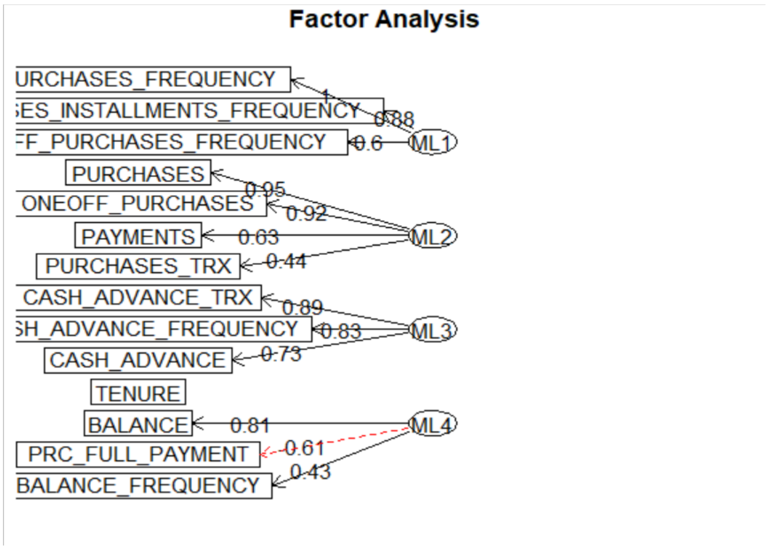
	Factor 1	Factor 2	Factor 3	Factor 4
Proportion of Variance Explained	0.21	0.30	0.20	0.29
Proportion of Variance	0.13	0.19	0.13	0.18

Cluster 3:

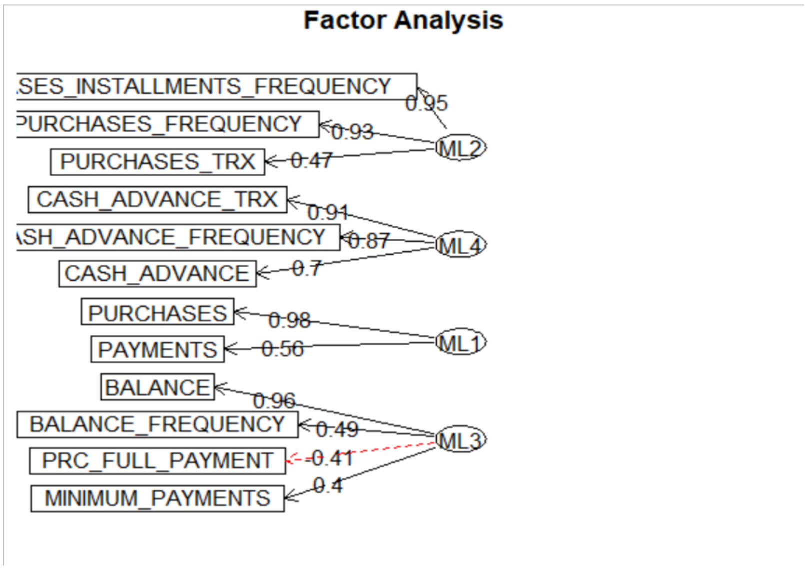
	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5
Proportion of Variance Explained	0.23	0.19	0.16	0.27	0.15
Proportion of Variance	0.13	0.11	0.09	0.16	0.09

Following diagrams show that how the different factors for each cluster summarizes the features:

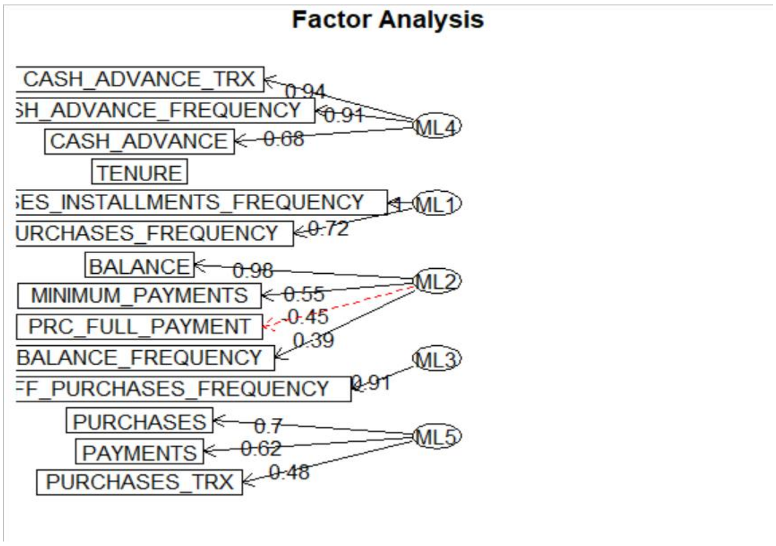
Cluster 1:



Cluster 2:



Cluster 3:



Measures of Lack of Fit:

Cluster	Tucker – Lewis Index	RMSEA index	Root Mean Square Residuals	BIC
Cluster 1	0.746	0.163	0.04	593.2
Cluster 2	0.913	0.091	0.03	775.38
Cluster 3	0.906	0.084	0.03	582.67

Here we have chosen the method of maximum likelihood among other alternatives to avoid the Ultra - Heywood problem which makes the loadings of the correlation matrix of features and factors more than 1 plus this method gives us reduction of BIC than any other method. Moreover, it is better than principal component method because it gives us measures of model's lack of fit which is essential to see how well our model performs.

Conclusion

Conclusion From Spectral Clustering: The persons belonging to first cluster uses credit card more often and they use this medium for transaction more than other two clusters, moreover they have higher credit limit that two clusters because they purchase through credit card more than other two clusters which is clear by their mean amounts of payments done by credit card or they are the customers who have high salary. Cluster 2 has lowest mean credit card limit and mean payments made by the card is also lowest. People belonging to first cluster tend to withdraw money from the credit limit more than other two clusters which may mean that they use that money for some business purpose and in a requirement of loan. As often the credit card issuing companies have a motive to convince the customers to take loan these customers can be target of the company as part of their business strategy. So the summary of three clusters provide us a good insight about the persons belonging to the different clusters.

Above histograms show that the persons in first cluster have higher credit limit than other two which means that they either have high salary structure, less burden of loan on them proportionate to the salary or they have high CIBIL score than other two clusters. Persons belonging to first cluster frequently purchase through credit card than other two clusters. Around 21.5% people very frequently purchases through installments, the figure is 17.8% and 28.57% for cluster 2 and 3 respectively. Persons of cluster 1 make payments of high magnitude through the card than other two clusters. Moreover total number of transactions are high among persons of cluster 1. Around 69.06% of persons of cluster 1 make 0 to 0.1% of full payments, while for cluster 2 around 70% of people make 0 to 0.1% of full payment. Surprisingly around 90.90% of the people in cluster 3 only makes 0.0 to 0.01% of full payment.

For cluster 1 the third factor 4 can be named **'Saving Habits'** because it represents good correlation with payments and purchases. Many customer make online payments with credit cards instead of debit cards because of discounts. Second factor can be named **'Parsimony'** because customers who pay minimum payments every month to avoid penalties and who don't prepay percent of full payment to avoid additional charges and have a good amount of balance left in card can show this trait due to their financial discipline. Factor 3 represents how frequently customers spends in one go, the factor can be named **‘Cost Effectiveness’** because they may do online shopping in bulk for availing discount. Factor 1 can be named **‘Lack of Affordability’** because it represents installment purchases frequency and purchases. Factor 5 can be named **‘Financial Burden’** they represent cash advance, cash advance frequency and cash advance transaction, if all these three variables are high for some customers it may mean that they use the credit card to pay home loans, car loans etc because through credit cards they can convert large amount of loan money into EMIs.

For cluster 2 the first Factor is named **‘Lack of Affordability’**, second factor is named **‘Saving Habits’**, third factor is named **‘Parsimony’**, fourth factor is named **‘Financial Burden’**,

For cluster 3 first factor can be named **‘Lack Of Affordability’**. Second factor can be named **‘Financial Discipline’** because it represents that the customers purchase through installments and timely pay minimum amount every month to escape penalty charges and third factor can be named **‘Financial Burden’.**

Cluster 1 and cluster 2 have all the factors common except **‘Cost Effectiveness’**. Cluster 3 has only Factor 1 and Factor 3 common with other two clusters. The factor **‘Financial Burden’** is intense in persons of cluster 2 because the loading of cash advance frequency is highest for this cluster. The factor **‘Lack of Affordability’** may have deeper effect on cluster 1 as the loading of installment purchase frequency is high for this cluster. The **‘Saving Habits’** of cluster 1 and 2 are nearly same as the loadings i.e correlation of the factors with payments through credit cards are close. Persons of Cluster 1 looks more parsimonious as the loadings of Balance and Minimum Payments are higher than cluster 2 though percent of full payment is close for both cases, moreover loading corresponding to balance update frequency is very low for cluster 1 which means their balance is updated few times because they tend to spend less in comparison to cluster 2.

Persons in first clusters have high balances and cash advances with low purchase frequency and high credit limit, which are evident from summary and histogram. These customers use credit cards as loans it will be wise to offer them **‘Business Credit Cards’.** Persons in second cluster are spenders with low purchases . Our strategy will be to offer them **‘Rewards Credit Cards’** or **‘Cashback Credit Cards’** to influence their spending habits. Persons in cluster 3 have medium balances and moderate purchase frequency with moderately high credit limit. The credit card issuing company may instigate their spending habits by increasing their credit limit. **‘Low Interest Credit Cards’** often offer high credit limits with low interest rate on what a person spend that’s why card holders may make transactions of high purchase. This may increase the purchases of the persons of this cluster from ‘moderate’ to ‘high’.

Conclusion from Kernel k-Means:

Cluster 1 has the smallest size among all the clusters. The persons belonging to cluster 1 uses the credit card for purchases more often than the other two clusters with higher credit balance and using cards with higher credit limits. We can assume that this Customer segment uses credit card for business purpose or for loan. Cluster 2 has the minimum purchase record among these subpopulations with credit limit much lower than the other two. Persons of cluster 2 is quite inclined towards installment purchase compared to one-off purchases which indicates that they might have a low salary compared to other customer segments. Persons of cluster 3 use credit cards moderately. The credit limit and the number of transactions is lower than cluster 1 but higher than cluster 2. The summary of three clusters provides us a great insight of the customer segmentation from the point of view of the companies.

Cluster 1: Factor 1 of cluster 1 includes the frequency measures of the purchases, so we can name it as **'Purchase regularity '**. The second factor of cluster 1 includes the factors related to purchase amount and one-off purchase amount. So, this factor mainly indicates how much one customer is likely to spend in one go or for any purchase. We can name this factor as **'Impulsive spending'**. Factor 3 includes all the term related to Cash Advance. This might happen due to regular requirement of physical money which is an indication of offline spending. So, this factor can be named as **'Cash requirement '**. Factor 4 deals is related to the features of maintaining the credit card account. This factor can be named as **'Account maintenance'**.

Cluster 2: The first factor includes Purchases and Payments. This indicates the amount of money getting used with the help of Credit Card. So this factor can be named as '**Amount of Expenditure**'. The second factor is named as **'Purchase regularity'**. The third factor is named as **'Account maintenance'**. Factor 4 can be named as **'Cash requirement'.**

Cluster 3: The first factor includes the terms Purchases Installments frequency and purchases frequency. This indicates that the persons are regularly purchasing things in installments which indicated their financial constraints. So this factor can be named as **'Lack of Affordability'**. The second factor is named as **'Account maintenance'.** The third factor includes the frequency of One-off purchases**.** In this case the customers are probably are using the credit card for online shopping to avail some offers or discounts. This factor can be named as **'Credit benefits'**. Factor 4 can be named as **'Cash requirement'.** Factor 5 can be named as '**Amount of Expenditure'**.

In terms of '**Purchase regularity**', persons of cluster 1 can be considered as the most regular spender as they might be using the card for some business purposes and the persons of cluster 2 are the least regular spenders as they might be suffering from low salary or some financial constraints and they are mostly inclined towards installment purchases. This also suggests that persons of Cluster 2 are most influenced cluster by the factor **'Lack of Affordibility'**. Persons of Cluster 1 has the tendency of **'Impulsive spending'.** Persons of cluster 1 uses the credit card for cash withdrawal more than the persons of other clusters. As the persons of first cluster are the most frequent users of the Credit card, they are usually most concerned about **Account maintainance**. Persons of cluster 3 are mostly balanced spenders. They make one-off purchases for availing discounts and also they also make payments in Installments for '**Lack of Affordability'**.

Persons in first clusters have high balances and cash advances with high purchase frequency and high credit limit. These customers use credit cards for business purposes. it will be wise to offer them **'Business Credit Cards'.** Persons in second cluster are spenders with low purchases and usually make payments in installments. Our strategy will be to offer them '**Low Interest Credit Cards'** which might help them to make more purchases without having fear of high interest and debts. Persons in cluster 3 have medium balances and moderate purchase frequency with moderately high credit limit. The credit card issuing company may instigate their spending habits by offering them **'Rewards Credit Cards'** or **'Cashback Credit Cards '**or **'Lifestyle credit cards'**.

Limitations: The dataset we have is quite huge in volume and posses highly variable features. The behaviours of the credit card customers are very diverse and for that reason overlapping is present quite heavily in the clusters. Also, the number of features are quite high and for that reason projecting into higher dimension and using ‘Euclidean distance’ as distance measure is not sufficient.

References and Bibliography

We have taken the help of some books for completing our project. They are:

1) Inderjit and Dhiller, Yuqiang Guan, Brian Kulis,
   Kernel K-means, Spectral Clustering and Normalized Cuts
2) Lihi Zelnik-Manor, Pietro Perona,
   Secf-Tuning Spectral Clustering
3) A.Ng, M.Jordan and Y.Weiss
   “On Spectral Clustering : Analysis and an Algorithm” In Advances in Neural Information Processing System 14,2001
4) Ulrike von Luxborg,
   ”A Tutorial on Spectral Clustering”,2007
5) Franklin, Scott B, Gibson, David J., Robertson, Philip A., Pohlmann, John T. and Fralish James S,
   ” Parallel Analysis: a method for determining Significant Principal Components”.
6) Ali Caner Turkmen “A review of Non-negative Matrix Factorization Methods for Clustering”