2022

Data Mining Project

YARESH VIJAYASUNDARAM

Post Graduate Program in Data Science and Business Analytics | **THE GREAT LEARNING**

Table of Contents

Problem Statement: 1

Clustering:

Digital Ads Data:

The ads24x7 is a Digital Marketing company which has now got seed funding of $10 Million. They are expanding their wings in Marketing Analytics. They collected data from their Marketing Intelligence team and now wants you (their newly appointed data analyst) to segment type of ads based on the features provided. Use Clustering procedure to segment ads into homogeneous groups.

Data Dictionary

Sl. No	Column Name	Column Description
1	Timestamp	The Timestamp of the particular Advertisement.
2	InventoryType	The Inventory Type of the particular Advertisement. Format 1 to 7. This is a Categorical Variable.
3	Ad - Length	The Length Dimension of the particular Adverstisement.
4	Ad- Width	The Width Dimension of the particular Advertisement.
5	Ad Size	The Overall Size of the particular Advertisement. Length*Width.
6	Ad Type	The type of the particular Advertisement. This is a Categorical Variable.
7	Platform	The platform in which the particular Advertisement is displayed. Web, Video or App. This is a Categorical Variable.
8	Device Type	The type of the device which supports the partciular Advertisement. This is a Categorical Variable.
9	Format	The Format in which the Advertisement is displayed. This is a Categorical Variable.
10	Available_Impressions	How often the particular Advertisement is shown. An impression is counted each time an Advertisement is shown on a search result page or other site on a Network.
11	Matched_Queries	Matched search queries data is pulled from Advertising Platform and consists of the exact searches typed into the search Engine that generated clicks for the particular Advertisement.
12	Impressions	The impression count of the particular Advertisement out of the total available impressions.
13	Clicks	It is a marketing metric that counts the number of times users have clicked on the particular advertisement to reach an online property.
14	Spend	It is the amount of money spent on specific ad variations within a specific campaign or ad set. This metric helps regulate ad performance.
15	Fee	The percentage of the Advertising Fees payable by Franchise Entities.
16	Revenue	It is the income that has been earned from the particular advertisement.
17	CTR	CTR stands for "Click through rate". CTR is the number of clicks that your ad receives divided by the number of times your ad is shown. Formula used here is CTR = Total Measured Clicks / Total Measured Ad Impressions x 100. Note that the Total Measured Clicks refers to the 'Clicks' Column and the Total Measured Ad Impressions refers to the 'Impressions' Column.
18	CPM	CPM stands for "cost per 1000 impressions." Formula used here is CPM = (Total Campaign Spend / Number of Impressions) * 1,000. Note that the Total Campaign Spend refers to the 'Spend' Column and the Number of Impressions refers to the 'Impressions' Column.
19	CPC	CPC stands for "Cost-per-click". Cost-per-click (CPC) bidding means that you pay for each click on your ads. The Formula used here is CPC = Total Cost (spend) / Number of Clicks. Note that the Total Cost (spend) refers to the 'Spend' Column and the Number of Clicks refers to the 'Clicks' Column.

1.1 Clustering: Read the data and perform basic analysis such as printing a few rows (head and tail), info, data summary, null values duplicate values, etc

The below figure 1 shows the first 5 Rows of the dataset.

	Timestamp	InventoryType	Ad - Length	Ad-Width	Ad Size	Ad Type	Platform	Device Type	Format	Available_Impressions	Matched_Queries	Impressions	Clicks	Spenc
0	2020-9-2-17	Format1	300	250	75000	Inter222	Video	Desktop	Display	1806	325	323	1	0.(
1	2020-9-2-10	Format1	300	250	75000	Inter227	App	Mobile	Video	1780	285	285	1	0.(
2	2020-9-1-22	Format1	300	250	75000	Inter222	Video	Desktop	Display	2727	356	355	1	0.(
3	2020-9-3-20	Format1	300	250	75000	Inter228	Video	Mobile	Video	2430	497	495	1	0.(
4	2020-9-4-15	Format1	300	250	75000	Inter217	Web	Desktop	Video	1218	242	242	1	0.(

Figure 1 Head of the dataset

The below figure 2 shows the last 5 rows of the dataset.

	Timestamp	InventoryType	Ad - Length	Ad-Width	Ad Size	Ad Type	Platform	Device Type	Format	Available_Impressions	Matched_Queries	Impressions	Clicks
23061	2020-9-13-7	Format5	720	300	216000	Inter220	Web	Mobile	Video	1	1	1	1
23062	2020-11-2-7	Format5	720	300	216000	Inter224	Web	Desktop	Video	3	2	2	1
23063	2020-9-14-22	Format5	720	300	216000	Inter218	App	Mobile	Video	2	1	1	1
23064	2020-11-18-2	Format4	120	600	72000	inter230	Video	Mobile	Video	7	1	1	1
23065	2020-9-14-0	Format5	720	300	216000	Inter221	App	Mobile	Video	2	2	2	1

Figure 2: Tail of the dataset

Data information:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 23066 entries, 0 to 23065
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Timestamp             23066 non-null  object
 1   InventoryType         23066 non-null  object
 2   Ad - Length           23066 non-null  int64
 3   Ad- Width             23066 non-null  int64
 4   Ad Size               23066 non-null  int64
 5   Ad Type               23066 non-null  object
 6   Platform              23066 non-null  object
 7   Device Type           23066 non-null  object
 8   Format                23066 non-null  object
 9   Available_Impressions 23066 non-null  int64
 10  Matched_Queries       23066 non-null  int64
 11  Impressions           23066 non-null  int64
 12  Clicks                23066 non-null  int64
 13  Spend                 23066 non-null  float64
 14  Fee                   23066 non-null  float64
 15  Revenue               23066 non-null  float64
 16  CTR                   18330 non-null  float64
 17  CPM                   18330 non-null  float64
 18  CPC                   18330 non-null  float64
dtypes: float64(7), int64(7), object(6)
memory usage: 3.3+ MB
```

```
Timestamp                0
InventoryType            0
Ad - Length              0
Ad- Width                0
Ad Size                  0
Ad Type                  0
Platform                 0
Device Type              0
Format                   0
Available_Impressions    0
Matched_Queries          0
Impressions              0
Clicks                   0
Spend                    0
Fee                      0
Revenue                  0
CTR                   4736
CPM                   4736
CPC                   4736
dtype: int64
```

Figure 3: Data information

From the above figure 3, we can see there are 23066 rows and 19 columns. The dataset collected by the 24x7 digital marketing company. By analysing this the data, the company can target right ad for the right segment. There are 19 Variable out of which 6 are float data type, 7 are Integer data type and 6 categorical (object) datatype. Furthermore, there are 3 variable (CTR, CPM & CPC) having 4736 missing values in the dataset.

Data Description:

	Ad - Length	Ad- Width	Ad Size	Available_Impressions	Matched_Queries	Impressions	Clicks	Spend	Fee	Re
count	23066.000000	23066.000000	23066.000000	2.306600e+04	2.306600e+04	2.306600e+04	23066.000000	23066.000000	23066.000000	23066.(
mean	385.163097	337.896037	96674.468048	2.432044e+06	1.295099e+06	1.241520e+06	10678.518816	2706.625689	0.335123	1924.2
std	233.651434	203.092885	61538.329557	4.742888e+06	2.512970e+06	2.429400e+06	17353.409363	4067.927273	0.031963	3105.2
min	120.000000	70.000000	33600.000000	1.000000e+00	1.000000e+00	1.000000e+00	1.000000	0.000000	0.210000	0.(
25%	120.000000	250.000000	72000.000000	3.367225e+04	1.828250e+04	7.990500e+03	710.000000	85.180000	0.330000	55.3
50%	300.000000	300.000000	72000.000000	4.837710e+05	2.580875e+05	2.252900e+05	4425.000000	1425.125000	0.350000	926.3
75%	720.000000	600.000000	84000.000000	2.527712e+06	1.180700e+06	1.112428e+06	12793.750000	3121.400000	0.350000	2091.3
max	728.000000	600.000000	216000.000000	2.759286e+07	1.470202e+07	1.419477e+07	143049.000000	26931.870000	0.350000	21276.7

Figure 4: Data Description

The above data description depicts the mean, medium, min and max values of the dataset. The dataset looks skewed.

```
df.duplicated().sum()
```

```
0
```

It is also evident from the above image that there are no duplicate value present in the dataset.

1.2 Clustering: Treat missing values in CPC, CTR and CPM using the formula given.

Formulas:

CPM = (Total Campaign Spend / Number of Impressions) * 1,000. Note that the Total Campaign Spend refers to the 'Spend' Column in the dataset and the Number of Impressions refers to the 'Impressions' Column in the dataset.

CPC = Total Cost (spend) / Number of Clicks. Note that the Total Cost (spend) refers to the 'Spend' Column in the dataset and the Number of Clicks refers to the 'Clicks' Column in the dataset.

CTR = Total Measured Clicks / Total Measured Ad Impressions x 100. Note that the Total Measured Clicks refers to the 'Clicks' Column in the dataset and the Total Measured Ad Impressions refers to the 'Impressions' Column in the dataset

As there are 4736 missing value in CPM, CPC, CTR variables of the dataset. User-defined function along with the given formula has been used to treat the missing value in the data. The below figure 5. shows after imputing, there are no missing values present in the data.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 23066 entries, 0 to 23065
Data columns (total 19 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Timestamp              23066 non-null  object
 1   InventoryType          23066 non-null  object
 2   Ad - Length            23066 non-null  int64
 3   Ad- Width              23066 non-null  int64
 4   Ad Size                23066 non-null  int64
 5   Ad Type                23066 non-null  object
 6   Platform               23066 non-null  object
 7   Device Type            23066 non-null  object
 8   Format                 23066 non-null  object
 9   Available_Impressions  23066 non-null  int64
 10  Matched_Queries        23066 non-null  int64
 11  Impressions            23066 non-null  int64
 12  Clicks                 23066 non-null  int64
 13  Spend                  23066 non-null  float64
 14  Fee                    23066 non-null  float64
 15  Revenue                23066 non-null  float64
 16  CTR                    23066 non-null  float64
 17  CPM                    23066 non-null  float64
 18  CPC                    23066 non-null  float64
dtypes: float64(6), int64(7), object(6)
memory usage: 3.3+ MB
```

Figure 5: Treating missing Values

1.3 Clustering: Check if there are any outliers. Do you think treating outliers is necessary for K-Means clustering? Based on your judgement decide whether to treat outliers and if yes, which method to employ. (As an analyst your judgement may be different from another analyst).

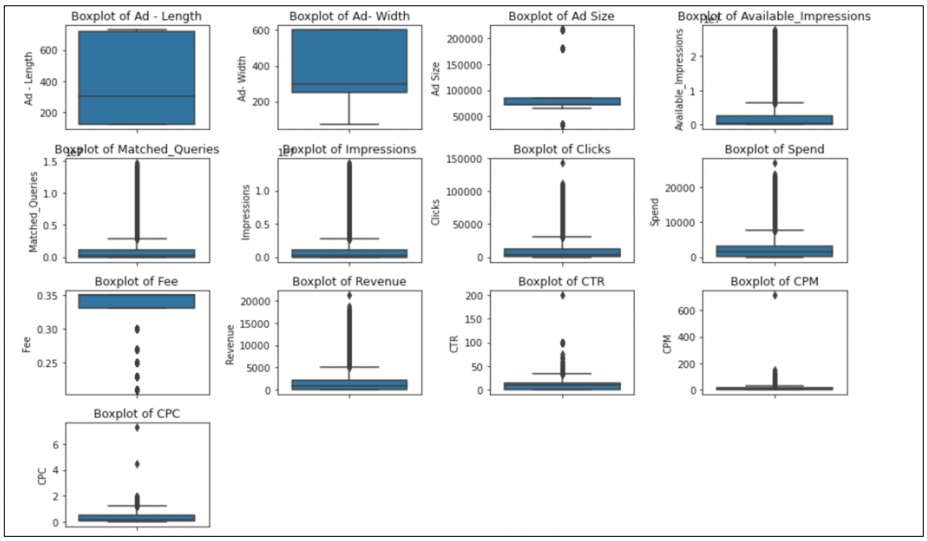

Figure 6: Boxplot before treating outliers

As we can see in the above boxplot figure 6, there are outliers in the dataset. It is essential to treat the outliers since K-means clustering is sensitive to the outliers, and the mean can influence the data due to the extreme value present in the dataset. And might cause an issue with the statistical analysis. IQR (Interquartile Range) and boxplot methods identify the outliers. The categorical variables have been dropped before analysing and treating the outliers.

The below is Fig 7. Boxplot shows after treating the outliers. (By using IQR Method – a barrier has been created by taking the 1.5 x IQR and subtracting with the Q1, and adding with the Q3)

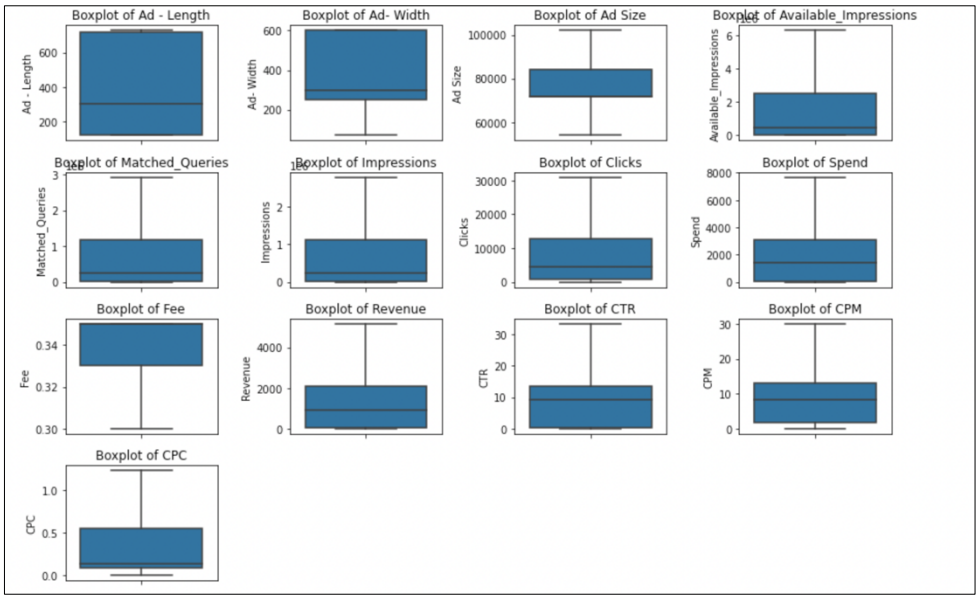

Figure 7: Boxplot after treating outliers

1.4 Clustering: Perform z-score scaling and discuss how it affects the speed of the algorithm.

```
df1_scaled.head()
```

	Ad - Length	Ad-Width	Ad Size	Available_Impressions	Matched_Queries	Impressions	Clicks	Spend	Fee	Revenue	CTR	CPM	C
0	-0.364496	-0.432797	-0.102518	-0.755333	-0.778949	-0.768478	-0.867488	-0.89317	0.535724	-0.880093	-0.958795	-0.938655	-1.04
1	-0.364496	-0.432797	-0.102518	-0.755345	-0.778988	-0.768516	-0.867488	-0.89317	0.535724	-0.880093	-0.953948	-0.938655	-1.04
2	-0.364496	-0.432797	-0.102518	-0.754900	-0.778919	-0.768445	-0.867488	-0.89317	0.535724	-0.880093	-0.962430	-0.938655	-1.04
3	-0.364496	-0.432797	-0.102518	-0.755040	-0.778781	-0.768302	-0.867488	-0.89317	0.535724	-0.880093	-0.972123	-0.938655	-1.04
4	-0.364496	-0.432797	-0.102518	-0.755610	-0.779030	-0.768560	-0.867488	-0.89317	0.535724	-0.880093	-0.946679	-0.938655	-1.04

Figure 8: Scaled Dataset head

The above fig 8 shows the dataset after z-score Scaling. Z-score Scaling is used lower the distance between the data. Z-score rescales the feature from a range of -1 to 1, and it converts the skewed data to a normal distribution.

Unscaled data can negatively affect the algorithm. Therefore, Scaling makes all the variables contribute equally. It speeds up the function and provides us with reliable and effective results in order to make better business decisions.

1.5 Clustering: Perform Hierarchical by constructing a Dendrogram using WARD and Euclidean distance.

```
from scipy.cluster.hierarchy import dendrogram, linkage
wardlink = linkage(df1_scaled, method='ward', metric='euclidean')
dend=dendrogram(wardlink)
```

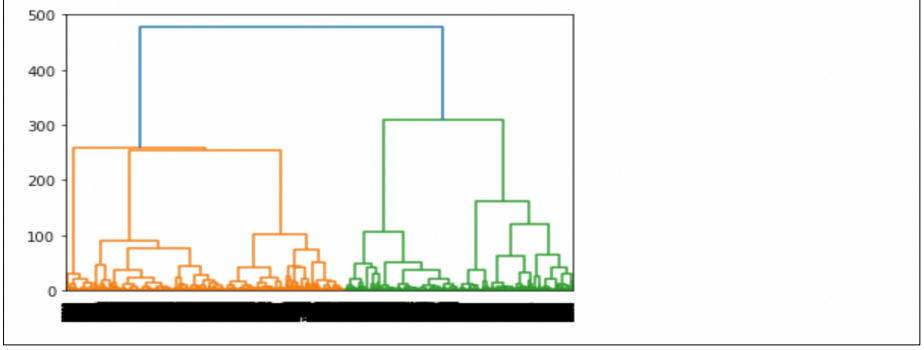

Figure 9: Dendrogram using WARD and Euclidean

The above fig 9 shows the Dendrogram using WARD and Euclidean distance. The Dendrogram performs Hierarchical clustering of data, and the algorithm creates clusters of data by measuring similarities between them. Dendrogram helps in visualisation the result of the Hierarchical clustering of the data. In the above Dendrogram, ward Linkage and Euclidean distance have been used where the WARD Linkage averages the similarities among the groups, and its centroid distance and the Euclidean shortens the distance between two points irrespective of the dimension.
As we can see from the above figure 8, at the bottom of dendrogram there are several cluster since the points are closer to each other. As we go higher several cluster have combined to form a new cluster, eventually all cluster merged into single cluster.

1.6 Clustering: Make Elbow plot (up to n=10) and identify optimum number of clusters for k-means algorithm.

```
a=[1,2,3,4,5,6,7,8,9,10]
```

```
sns.pointplot(a, wss);
```

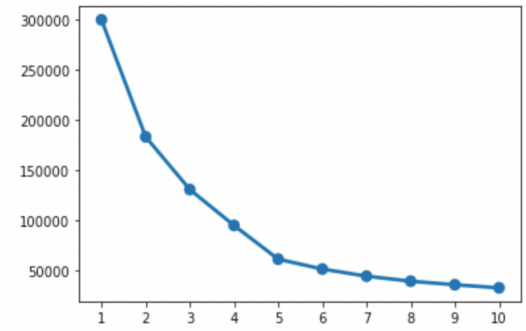

Figure 10: Elbow Plot

From the above Figure 10 elbow plot, it is evident that there is a significant drop from K=1 to K=5. Therefore, the Within sum of the square is not dropping significantly beyond 5. After 5, the decline has become very gradual.

Hence, from the plot, we can conclude that the optimum number of the cluster for the K - means algorithm is 5.

1.7 Clustering: Print silhouette scores for up to 10 clusters and identify optimum number of clusters.

```
for i in range(2,11):
    k_means=KMeans(n_clusters=i,random_state=1)
    k_means.fit(df1_scaled)
    labels=k_means.labels_
    print('silhouette_score for',i,'Clusters:',silhouette_score(df1_scaled,labels))
```

```
silhouette_score for 2 Clusters: 0.3857276961910116
silhouette_score for 3 Clusters: 0.38254860365700916
silhouette_score for 4 Clusters: 0.4532427055259853
silhouette_score for 5 Clusters: 0.5240956940501869
silhouette_score for 6 Clusters: 0.5221533662938672
silhouette_score for 7 Clusters: 0.5165635029478554
silhouette_score for 8 Clusters: 0.4797224989383805
silhouette_score for 9 Clusters: 0.43206365640251304
silhouette_score for 10 Clusters: 0.4312485458108503
```

Figure 11: Silhouette Score

The top cluster silhouette score is printed from a range of 2 to 10 in the above fig 11.; it is clear that the silhouette score is maximum at the 5th cluster. Therefore, we conclude by saying the optimum number of clusters is 5.

1.8 Clustering: Profile the ads based on optimum number of clusters using silhouette score and your domain understanding [Hint: Group the data by clusters and take sum or mean to identify trends in Clicks, spend, revenue, CPM, CTR, & CPC based on Device Type. Make bar plots].

The data is grouped/profiled using the optimum number of clusters using silhouette score, which is five, and the mean has been taken to identify the trend in clicks, spends, revenue, CPM, CTR and CPC based on device type.

Mean Clicks based on Device Type:

Device Type	Desktop	Mobile
C_kmeans		
0	14541.243713	14331.704923
1	3267.725314	3260.603163
2	1950.679487	1894.185946
3	11312.535836	11207.966396
4	65285.184919	65332.222449

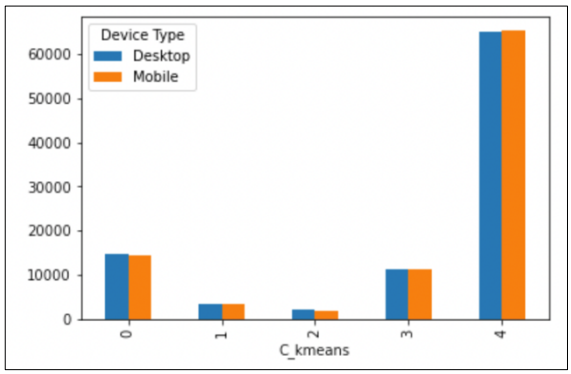

Figure 12:Mean Clicks based on Device Type

The above bar chart, fig 12, clearly depicts the mean of cluster 4 at the peak for both the desktop and mobile devices at above 650000 clicks. Followed by Clusters 0 and 3 are at the highest, with means of above 14,000 and 11,000 clicks, respectively. And cluster 1 and cluster 2 registered with the lowest click.

Mean Spend based on Device type:

Device Type	Desktop	Mobile
C_kmeans		
0	1251.027671	1252.984401
1	1495.319825	1502.717003
2	209.291786	209.090363
3	8637.451242	8651.852032
4	6972.607522	7000.451337

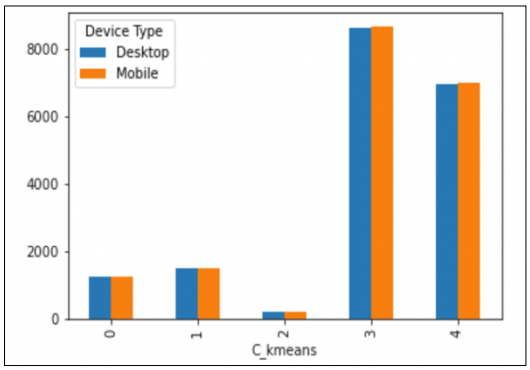

Figure 13: Mean Spend based on Device type

The above figure 13 bar chart represents clustered data of mean spending based on device type. As we can spend mean, cluster 3 is at the top for both devices and followed the cluster 4 at the second spot. Whereases cluster 2 is at the lowest spend for both the Desktop and mobile at approx.209.

Mean revenue based on Device type:

Device Type	Desktop	Mobile
C_kmeans		
0	814.675848	816.022933
1	974.211859	979.192636
2	136.091814	135.938327
3	6366.972127	6377.444079
4	5002.676306	5025.985350

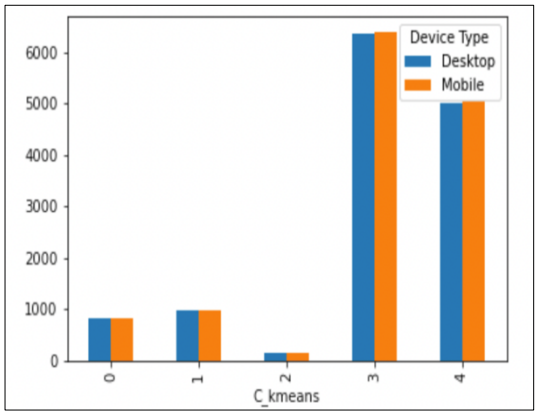

Figure 14:Mean revenue based on Device type

The above fig 14 chart shows the clustered data of mean revenue based on device type. It is evident that the cluster 3 mean revenue is the highest with approx. 6300 and followed by cluster 4 with approx. 5000. Cluster 2 holds the last position with the lowest mean revenue.

Mean CPM (Cost per 1000 impressions)based on Device type:

Device Type	Desktop	Mobile
C_kmeans		
0	12.069679	12.114045
1	1.789589	1.788258
2	14.514239	14.793727
3	1.561860	1.579743
4	15.431593	15.359699

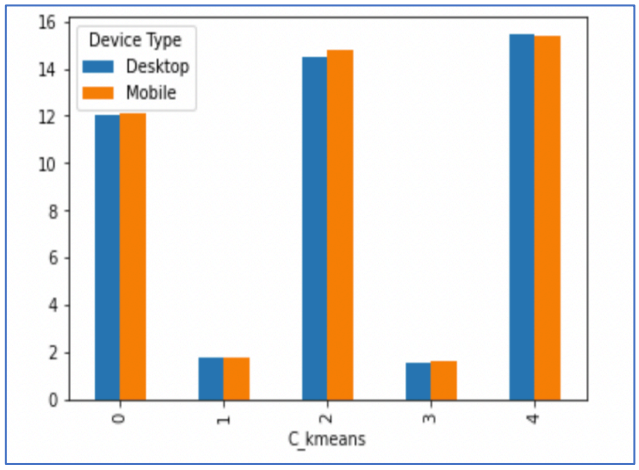

Figure 15: Mean CPM (Cost per 1000 impressions)based on Device type

The above chart, fig 15, shows clustered data of the mean of Cost per 1000 Impressions (CPM) based on the device type. As we can see, cluster 4 has the highest mean CPM with close to 16, followed by cluster 2 with approx. 15 for both devices. Clusters 1 and 2 are at the lowest, with approx. 2 mean CPM.

Mean CTR (Click through Rate) based on Device type:

Device Type	Desktop	Mobile
C_kmeans		
0	13.823352	13.876035
1	0.408147	0.402325
2	16.053553	16.029142
3	0.217558	0.217063
4	13.793226	13.729611

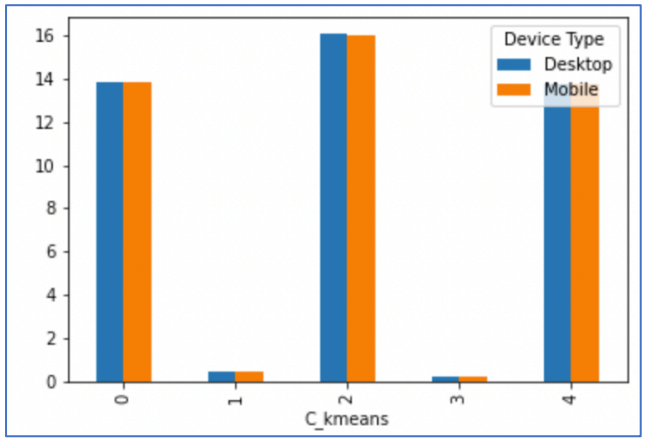

Figure 16: Mean CTR (Click through Rate) based on Device type

The above fig 16 chart shows clustered data of mean of Click Through Rate(CTR) based on the device type. The cluster 2 is at the highest mean CTR for the both mobile and desktop device at 16. Followed by the cluster 0 and cluster 4 at mean CTR of 14 and Cluster 3 at the lowest at 1.

Mean CPC (Cost per Click) based on Device type:

Device Type	Desktop	Mobile
C_kmeans		
0	0.089394	0.090356
1	0.540612	0.546817
2	0.102897	0.102737
3	0.754708	0.764450
4	0.111911	0.111922

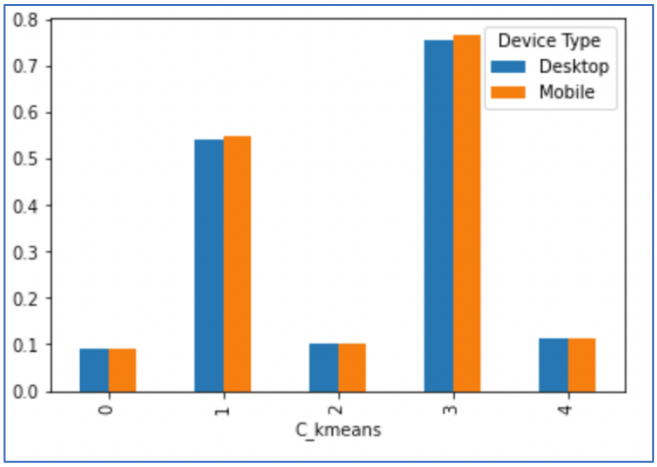

Figure 17:Mean CPC (Cost per Click) based on Device type:

The above fig 17 chart shows clustered data of mean Spend per Click (CPC) based on the device type. As we can see, cluster 3 is at the peak for both devices at closer to 0.7 and followed by cluster 1 with a mean of 0.55 CPC. Clusters 0, 2 and 3 follow a similar trend with 0.1.

1.9 Clustering: Conclude the project by providing summary of your learnings.

Summary of the dataset:

The ad 24x7 marketing company have collected the data from the marketing intelligence to analyse and segmentalize the ads to target the right set of groups. The following details were found during the assessment of the dataset.

- The dataset has 23066 rows and 19 features. Out of 19 features – 6 are float type, 7 are integer and six are categorical.
- During the project, we also found 4736 missing values; those values were imputed and treated with the given formula using a user-defined function.
- The dataset also had outliers; those outliers were treated using the IQR method since the K-mean clustering is sensitive to outliers and could negatively influence the dataset.
- The dataset also has been scaled. Since the unscaled data could negatively impact the speed of the algorithm and scaling data can make the variable contribute equally to the analysis to take better business decisions.
- Hierarchal clustering has been performed with the data to find the optimum number of clusters, which is 5.
- The below table show which cluster has the highest, moderate or lowest of the following: clicks, revenue, spend, CPM, CTR and CPC.

	Clicks	**Spend**	**Revenue**	**CPM**	**CTR**	**CPC**
Cluster 0	Moderate	Moderate	Low	High	High	Low
Cluster 1	Low	Moderate	Moderate	Low	Low	High
Cluster 2	Low	Low	Low	High	Highest	Low
Cluster 3	Low	Highest	Highest	Low	Low	Highest
Cluster 4	Highest	High	High	Highest	High	Low

Problem 2:

PCA:

PCA FH (FT): Primary census abstract for female headed households excluding institutional households (India & States/UTs - District Level), Scheduled tribes - 2011 PCA for Female Headed Household Excluding Institutional Household. The Indian Census has the reputation of being one of the best in the world. The first Census in India was conducted in the year 1872. This was conducted at different points of time in different parts of the country. In 1881 a Census was taken for the entire country simultaneously. Since then, Census has been conducted every ten years, without a break. Thus, the Census of India 2011 was the fifteenth in this unbroken series since 1872, the seventh after independence and the second census of the third millennium and twenty first century. The census has been uninterruptedly continued despite of several adversities like wars, epidemics, natural calamities, political unrest, etc. The Census of India is conducted under the provisions of the Census Act 1948 and the Census Rules, 1990. The Primary Census Abstract which is important publication of 2011 Census gives basic information on Area, Total Number of Households, Total Population, Scheduled Castes, Scheduled Tribes Population, Population in the age group 0-6, Literates, Main Workers and Marginal Workers classified by the four broad industrial categories, namely, (i) Cultivators, (ii) Agricultural Laborers, (iii) Household Industry Workers, and (iv) Other Workers and also Non-Workers. The characteristics of the Total Population include Scheduled Castes, Scheduled Tribes, Institutional and Houseless Population and are presented by sex and rural-urban residence. Census 2011 covered 35 States/Union Territories, 640 districts, 5,924 sub-districts, 7,935 Towns and 6,40,867 Villages.
The data collected has so many variables thus making it difficult to find useful details without using Data Science Techniques. You are tasked to perform detailed EDA and identify Optimum Principal Components that explains the most variance in data. Use Sklearn only.

Data Dictionary:	
Name	Description
State	State Code
District	District Code
Name	Name
TRU1	Area Name
No_HH	No of Household
TOT_M	Total population Male
TOT_F	Total population Female
M_06	Population in the age group 0-6 Male
F_06	Population in the age group 0-6 Female
M_SC	Scheduled Castes population Male
F_SC	Scheduled Castes population Female
M_ST	Scheduled Tribes population Male
F_ST	Scheduled Tribes population Female
M_LIT	Literates population Male
F_LIT	Literates population Female
M_ILL	Illiterate Male
F_ILL	Illiterate Female
TOT_WORK_M	Total Worker Population Male
TOT_WORK_F	Total Worker Population Female

MAINWORK_M	Main Working Population Male
MAINWORK_F	Main Working Population Female
MAIN_CL_M	Main Cultivator Population Male
MAIN_CL_F	Main Cultivator Population Female
MAIN_AL_M	Main Agricultural Labourers Population Male
MAIN_AL_F	Main Agricultural Labourers Population Female
MAIN_HH_M	Main Household Industries Population Male
MAIN_HH_F	Main Household Industries Population Female
MAIN_OT_M	Main Other Workers Population Male
MAIN_OT_F	Main Other Workers Population Female
MARGWORK_M	Marginal Worker Population Male
MARGWORK_F	Marginal Worker Population Female
MARG_CL_M	Marginal Cultivator Population Male
MARG_CL_F	Marginal Cultivator Population Female
MARG_AL_M	Marginal Agriculture Labourers Population Male
MARG_AL_F	Marginal Agriculture Labourers Population Female
MARG_HH_M	Marginal Household Industries Population Male
MARG_HH_F	Marginal Household Industries Population Female
MARG_OT_M	Marginal Other Workers Population Male
MARG_OT_F	Marginal Other Workers Population Female
MARGWORK_3_6_M	Marginal Worker Population 3-6 Male
MARGWORK_3_6_F	Marginal Worker Population 3-6 Female
MARG_CL_3_6_M	Marginal Cultivator Population 3-6 Male
MARG_CL_3_6_F	Marginal Cultivator Population 3-6 Female
MARG_AL_3_6_M	Marginal Agriculture Labourers Population 3-6 Male
MARG_AL_3_6_F	Marginal Agriculture Labourers Population 3-6 Female
MARG_HH_3_6_M	Marginal Household Industries Population 3-6 Male
MARG_HH_3_6_F	Marginal Household Industries Population 3-6 Female
MARG_OT_3_6_M	Marginal Other Workers Population Person 3-6 Male
MARG_OT_3_6_F	Marginal Other Workers Population Person 3-6 Female
MARGWORK_0_3_M	Marginal Worker Population 0-3 Male
MARGWORK_0_3_F	Marginal Worker Population 0-3 Female
MARG_CL_0_3_M	Marginal Cultivator Population 0-3 Male
MARG_CL_0_3_F	Marginal Cultivator Population 0-3 Female
MARG_AL_0_3_M	Marginal Agriculture Labourers Population 0-3 Male
MARG_AL_0_3_F	Marginal Agriculture Labourers Population 0-3 Female
MARG_HH_0_3_M	Marginal Household Industries Population 0-3 Male
MARG_HH_0_3_F	Marginal Household Industries Population 0-3 Female
MARG_OT_0_3_M	Marginal Other Workers Population 0-3 Male
MARG_OT_0_3_F	Marginal Other Workers Population 0-3 Female
NON_WORK_M	Non-Working Population Male
NON_WORK_F	Non-Working Population Female

2.1 PCA: Read the data and perform basic checks like checking head, info, summary, nulls, and duplicates, etc.

The below figure shows the first 5 Rows of the dataset.

	State Code	Dist.Code	State	Area Name	No_HH	TOT_M	TOT_F	M_06	F_06	M_SC	...	MARG_CL_0_3_M	MARG_CL_0_3_F	MARG_AL_0_3_M	MARG_AL_0
0	1	1	Jammu & Kashmir	Kupwara	7707	23388	29796	5862	6196	3	...	1150	749	180	
1	1	2	Jammu & Kashmir	Badgam	6218	19585	23102	4482	3733	7	...	525	715	123	
2	1	3	Jammu & Kashmir	Leh(Ladakh)	4452	6546	10964	1082	1018	3	...	114	188	44	
3	1	4	Jammu & Kashmir	Kargil	1320	2784	4206	563	677	0	...	194	247	61	
4	1	5	Jammu & Kashmir	Punch	11654	20591	29981	5157	4587	20	...	874	1928	465	

Figure 18: Head of the Dataset

The below figure shows the last 5 rows of the dataset:

	State Code	Dist.Code	State	Area Name	No_HH	TOT_M	TOT_F	M_06	F_06	M_SC	...	MARG_CL_0_3_M	MARG_CL_0_3_F	MARG_AL_0_3_M	MARG_Al
635	34	636	Puducherry	Mahe	3333	8154	11781	1146	1203	21	...	32	47	0	
636	34	637	Puducherry	Karaikal	10612	12346	21691	1544	1533	2234	...	155	337	3	
637	35	638	Andaman & Nicobar Island	Nicobars	1275	1549	2630	227	225	0	...	104	134	9	
638	35	639	Andaman & Nicobar Island	North & Middle Andaman	3762	5200	8012	723	664	0	...	136	172	24	
639	35	640	Andaman & Nicobar Island	South Andaman	7975	11977	18049	1470	1358	0	...	173	122	6	

Figure 19: Tail of the dataset

Data Info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 640 entries, 0 to 639
Data columns (total 61 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   State Code      640 non-null    int64
 1   Dist.Code       640 non-null    int64
 2   State           640 non-null    object
 3   Area Name       640 non-null    object
 4   No_HH           640 non-null    int64
 5   TOT_M           640 non-null    int64
 6   TOT_F           640 non-null    int64
 7   M_06            640 non-null    int64
```

```
 8   F_06            640 non-null    int64
 9   M_SC            640 non-null    int64
 10  F_SC            640 non-null    int64
 11  M_ST            640 non-null    int64
 12  F_ST            640 non-null    int64
 13  M_LIT           640 non-null    int64
 14  F_LIT           640 non-null    int64
 15  M_ILL           640 non-null    int64
 16  F_ILL           640 non-null    int64
 17  TOT_WORK_M      640 non-null    int64
 18  TOT_WORK_F      640 non-null    int64
 19  MAINWORK_M      640 non-null    int64
 20  MAINWORK_F      640 non-null    int64
 21  MAIN_CL_M       640 non-null    int64
 22  MAIN_CL_F       640 non-null    int64
 23  MAIN_AL_M       640 non-null    int64
 24  MAIN_AL_F       640 non-null    int64
 25  MAIN_HH_M       640 non-null    int64
 26  MAIN_HH_F       640 non-null    int64
 27  MAIN_OT_M       640 non-null    int64
 28  MAIN_OT_F       640 non-null    int64
 29  MARGWORK_M      640 non-null    int64
 30  MARGWORK_F      640 non-null    int64
 31  MARG_CL_M       640 non-null    int64
 32  MARG_CL_F       640 non-null    int64
 33  MARG_AL_M       640 non-null    int64
 34  MARG_AL_F       640 non-null    int64
 35  MARG_HH_M       640 non-null    int64
 36  MARG_HH_F       640 non-null    int64
 37  MARG_OT_M       640 non-null    int64
 38  MARG_OT_F       640 non-null    int64
 39  MARGWORK_3_6_M  640 non-null    int64
 40  MARGWORK_3_6_F  640 non-null    int64
 41  MARG_CL_3_6_M   640 non-null    int64
 42  MARG_CL_3_6_F   640 non-null    int64
 43  MARG_AL_3_6_M   640 non-null    int64
 44  MARG_AL_3_6_F   640 non-null    int64
 45  MARG_HH_3_6_M   640 non-null    int64
 46  MARG_HH_3_6_F   640 non-null    int64
 47  MARG_OT_3_6_M   640 non-null    int64
 48  MARG_OT_3_6_F   640 non-null    int64
 49  MARGWORK_0_3_M  640 non-null    int64
 50  MARGWORK_0_3_F  640 non-null    int64
 51  MARG_CL_0_3_M   640 non-null    int64
 52  MARG_CL_0_3_F   640 non-null    int64
 53  MARG_AL_0_3_M   640 non-null    int64
 54  MARG_AL_0_3_F   640 non-null    int64
 55  MARG_HH_0_3_M   640 non-null    int64
 56  MARG_HH_0_3_F   640 non-null    int64
 57  MARG_OT_0_3_M   640 non-null    int64
 58  MARG_OT_0_3_F   640 non-null    int64
 59  NON_WORK_M      640 non-null    int64
 60  NON_WORK_F      640 non-null    int64
dtypes: int64(59), object(2)
memory usage: 305.1+ KB
```

From the above data, we can see 640 rows with 61 columns. Out of 61 features – 59 columns belong to the integer data type, and 2 are object (Categorical data type).

The below fig 20 shows the data that depicts the mean, median, min and max values of the dataset. The dataset looks skewed.

	State Code	Dist.Code	No_HH	TOT_M	TOT_F	M_06	F_06	M_SC	F_SC	M_ST	...	M
count	640.000000	640.000000	640.000000	640.000000	640.000000	640.000000	640.000000	640.000000	640.000000	640.000000	...	
mean	17.114062	320.500000	51222.871875	79940.576563	122372.084375	12309.098438	11942.300000	13820.946875	20778.392188	6191.807813	...	
std	9.426486	184.896367	48135.405475	73384.511114	113600.717282	11500.906881	11326.294567	14426.373130	21727.887713	9912.668948	...	
min	1.000000	1.000000	350.000000	391.000000	698.000000	56.000000	56.000000	0.000000	0.000000	0.000000	...	
25%	9.000000	160.750000	19484.000000	30228.000000	46517.750000	4733.750000	4672.250000	3466.250000	5603.250000	293.750000	...	
50%	18.000000	320.500000	35837.000000	58339.000000	87724.500000	9159.000000	8663.000000	9591.500000	13709.000000	2333.500000	...	
75%	24.000000	480.250000	68892.000000	107918.500000	164251.750000	16520.250000	15902.250000	19429.750000	29180.000000	7658.000000	...	
max	35.000000	640.000000	310450.000000	485417.000000	750392.000000	96223.000000	95129.000000	103307.000000	156429.000000	96785.000000	...	

Figure 20: Data description

The below fig 21 proves that there are no duplicates and null values present in the dataset.

```
df_p.duplicated().sum()
0

df_p.isnull().sum()
State Code          0
Dist.Code           0
State               0
Area Name           0
No_HH               0
                   ..
MARG_HH_0_3_F       0
MARG_OT_0_3_M       0
MARG_OT_0_3_F       0
NON_WORK_M          0
NON_WORK_F          0
Length: 61, dtype: int64
```

Figure 21: Checking null values and duplicate values

The below fig shows the 5 districts' names repeated twice in the data, and the same hasn't been treated since 2 states might have the same district names.

```
: df_p['Area Name'].value_counts()
: Raigarh           2
  Bijapur           2
  Aurangabad        2
  Hamirpur          2
  Bilaspur          2
                   ..
  Darbhanga         1
  Muzaffarpur       1
  Gopalganj         1
  Siwan             1
  South Andaman     1
  Name: Area Name, Length: 635, dtype: int64
```

Figure 22: District names

2.2 PCA: Perform detailed Exploratory analysis by creating certain questions like (i) Which state has highest gender ratio and which has the lowest? (ii) Which district has the highest & lowest gender ratio? (Example Questions). Pick 5 variables out of the given 24 variables below for EDA: No_HH, TOT_M, TOT_F, M_06, F_06, M_SC, F_SC, M_ST, F_ST, M_LIT, F_LIT, M_ILL, F_ILL, TOT_WORK_M, TOT_WORK_F, MAINWORK_M, MAINWORK_F, MAIN_CL_M, MAIN_CL_F, MAIN_AL_M, MAIN_AL_F, MAIN_HH_M, MAIN_HH_F, MAIN_OT_M, MAIN_OT_F

1. **Which State has the highest number of literate men and the lowest number of literate men?**

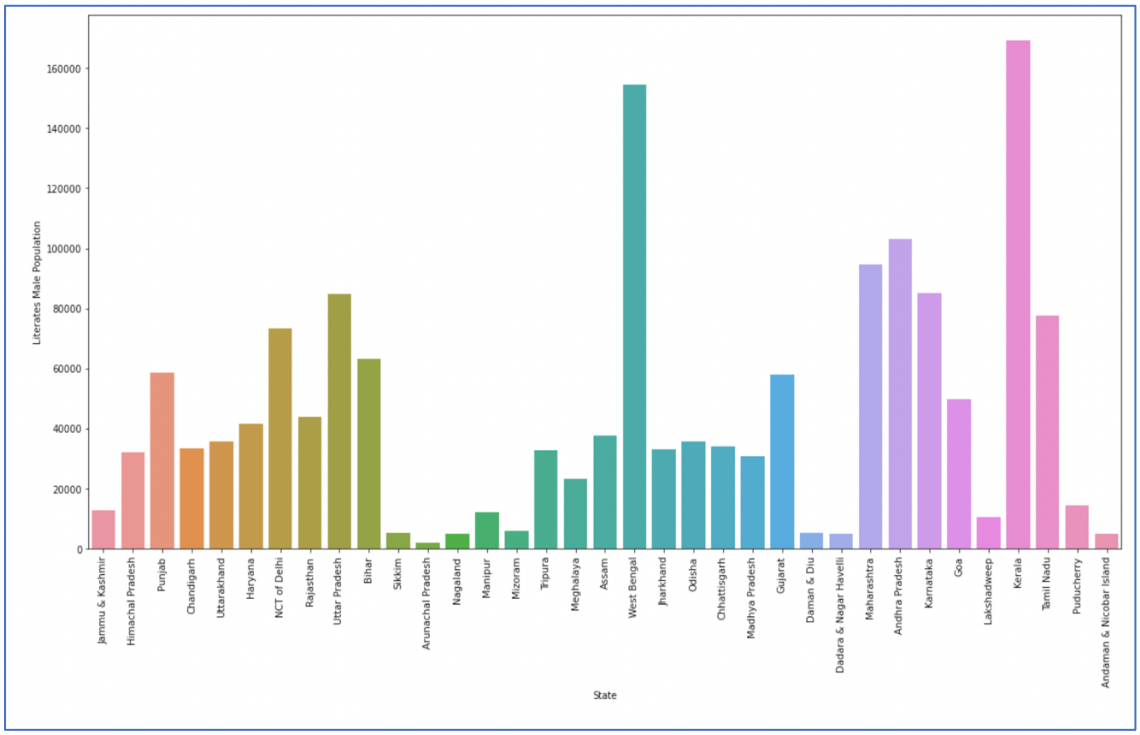

Figure 23: Literate population of men

The above bar chart fig. 23 depicts that literate men in all the states. And it is evident that Kerala has the highest literacy men population, and Arunachal Pradesh has the lowest number of literate men among all the states in India.

2. **Which state has the highest gender ratio and which has the lowest?**

The below bar graph fig 24 represents the gender ratio among the Indian states. We can see that Lakshadweep has the highest gender ratio with over 0.9, and Chhattisgarh and Andra Pradesh have the lowest gender ratio with 0.5.

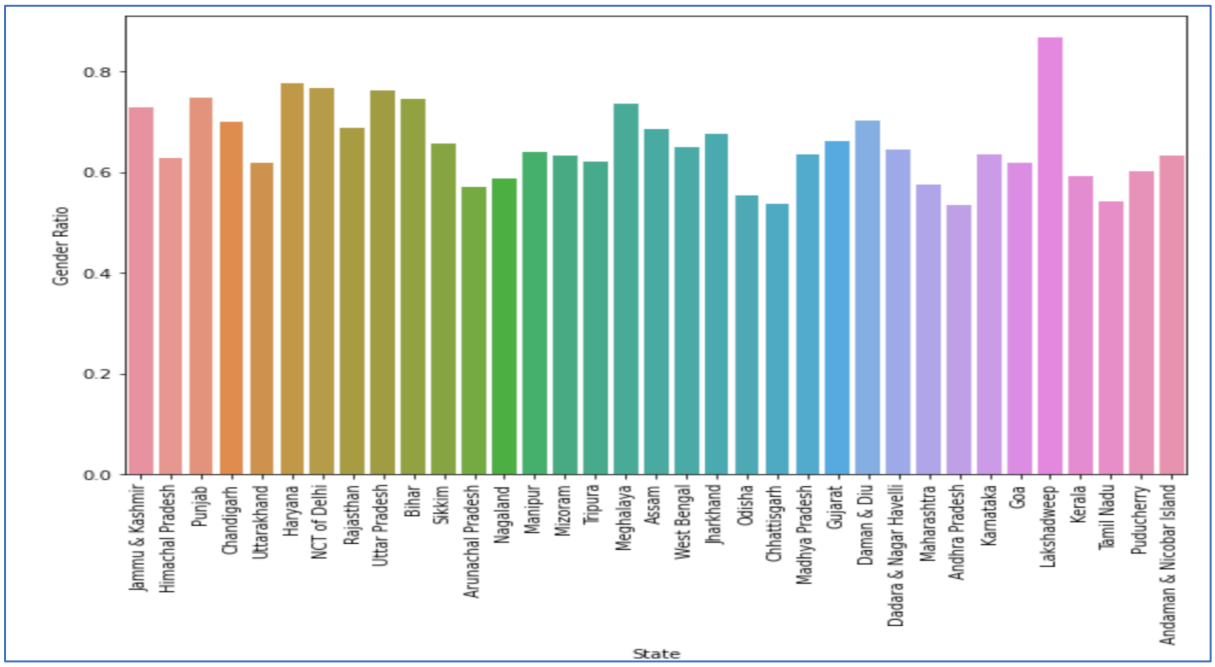

Figure 24: Gender Ratio

3. Which state has the male's highest and lowest main cultivator population?

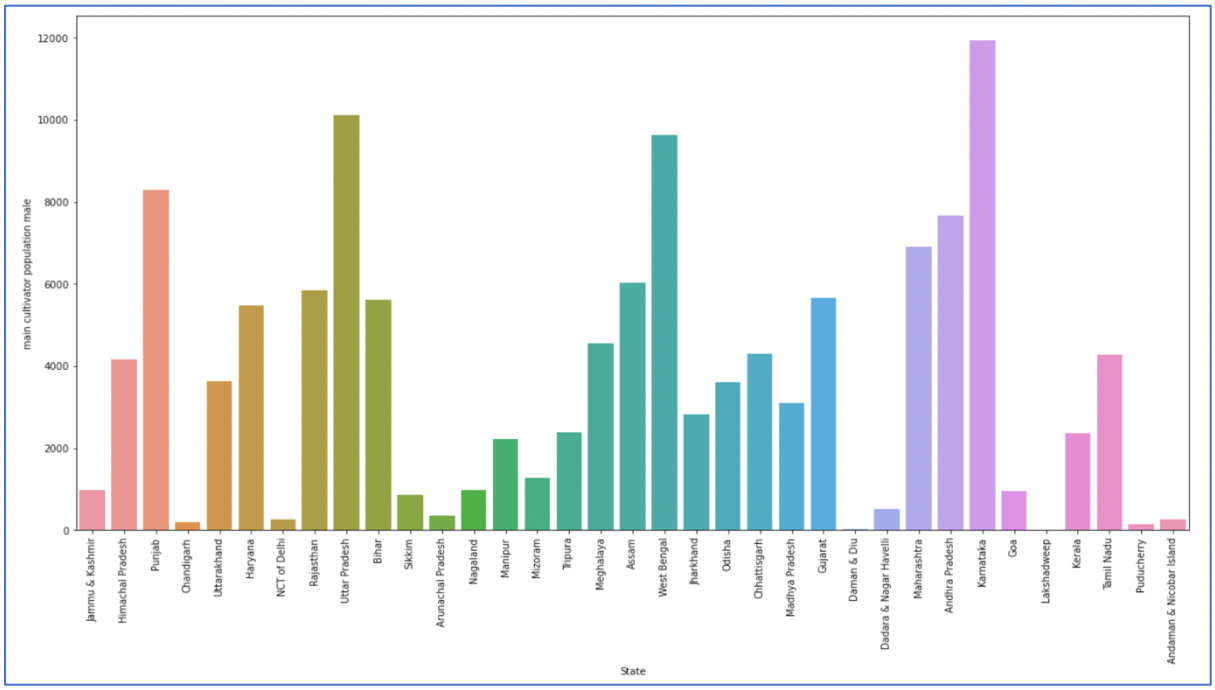

Figure 25: Male Cultivator population

The above bar chart fig 25 shows the main cultivator male population among all the states in India. As we can see, the Karnataka state has the highest male main cultivator, whereas Lakshadweep as the lowest main male cultivator.

4. **Which state has the highest and lowest Main Agricultural Labourers Population Female?**

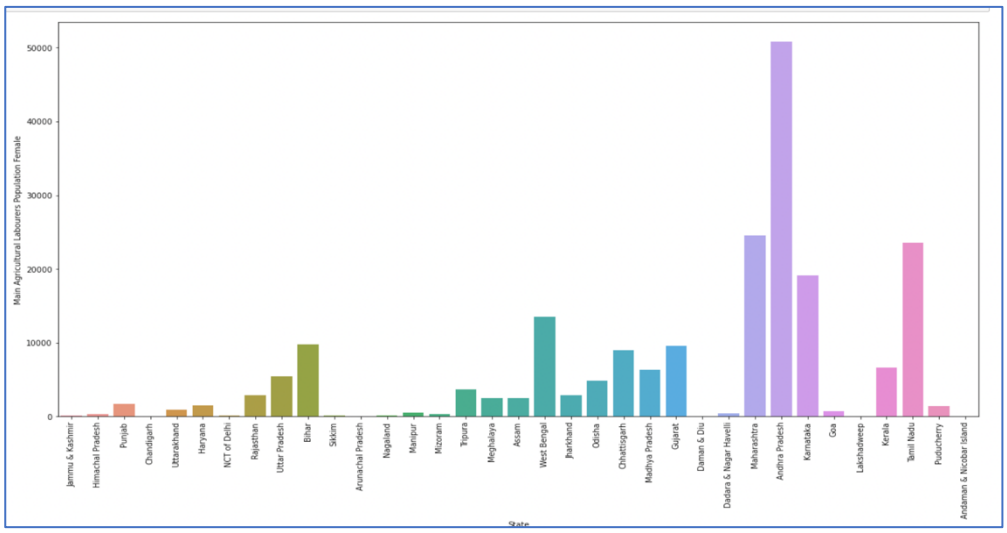

Figure 26: Main agricultural labourers population female

The above fig. 26 bar chart represents the main agricultural labourer female population among all the states. As we can see, Andra Pradesh has the highest female agricultural labourers population, and Arunachal Pradesh and Lakshadweep have the lowest agricultural labourers female population.

5. **Which state has the highest and lowest non-working population Female?**

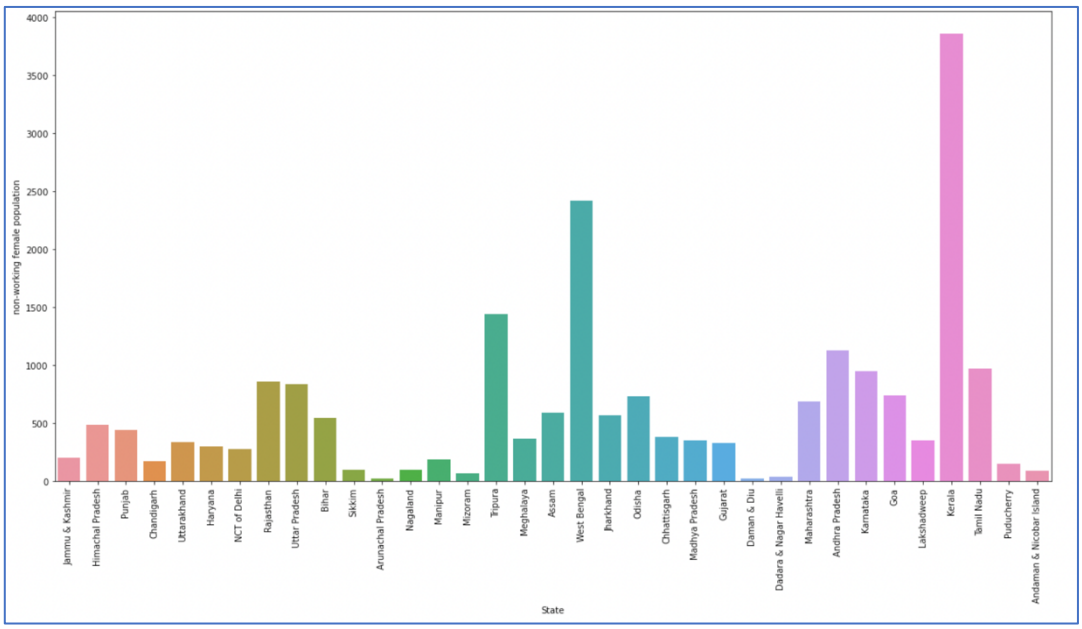

Figure 27: Non-working population female

The above fig 27 bar chart explains the non-working female population in the Indian state. Kerala state has the highest non-working population, whereas Arunachal Pradesh and Daman & Di have the lowest non-working female population.

6. **Which state has the highest and lowest ST population Male?**

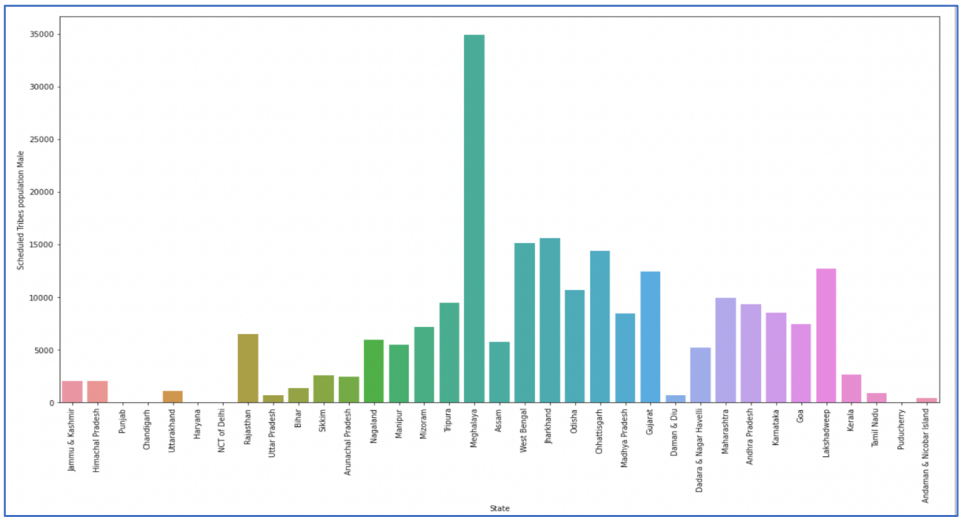

Figure 28: ST Population Male

The above fig 28 bar chart depicts the scheduled tribe's male population of all the states in India. It is clear that the West Bengal state has the highest scheduled tribe male population. It is a surprise to see that Punjab, Chandigarh, Haryana, NCT of Delhi and Puducherry have absolutely no tribal male population.

2.3 PCA: We choose not to treat outliers for this case. Do you think that treating outliers for this case is necessary?

Principal Component Analysis (PCA) is a highly flexible multivariate data dimension reduction method. In the presence of outliers, classical PCA is highly sensitive to them and may draw false conclusions.

2.4 PCA: Scale the Data using z-score method. Does scaling have any impact on outliers? Compare boxplots before and after scaling and comment.

Before scaling and checking the outliers, the Categorical variable has been dropped from the dataset. The below image represents the boxplot of the dataset before scaling.

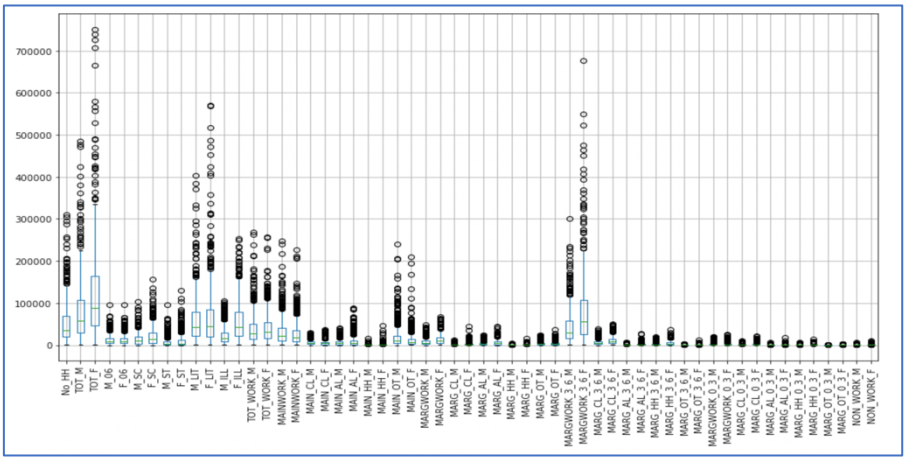

Figure 29: Boxplot before scaling

The below fig 30 shows the boxplot after z-score scaling method:

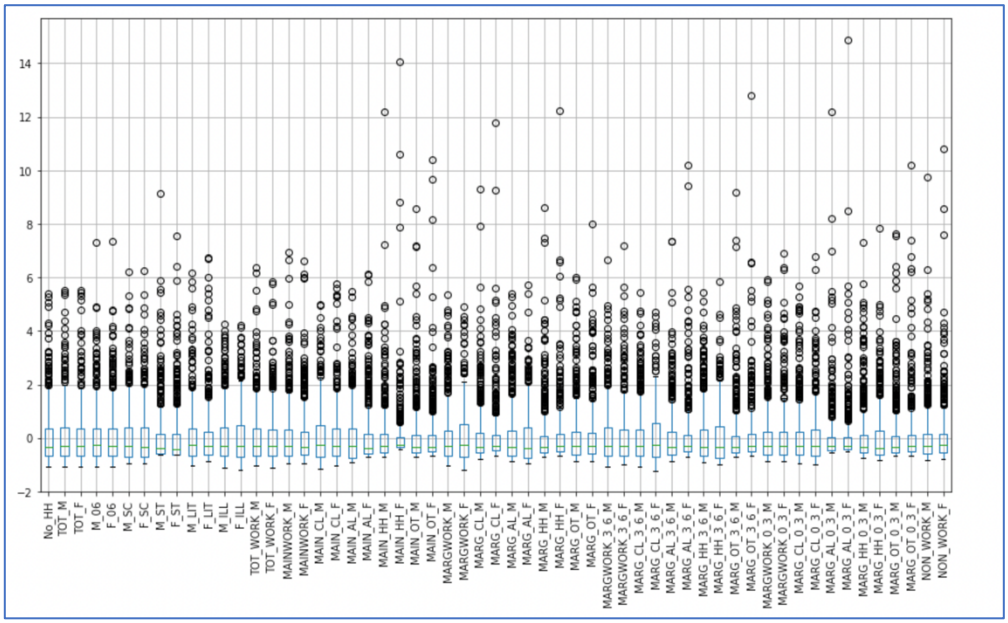

Figure 30: Boxplot after scaling

Apart from the scaling adjustment, there are absolutely no changes when we compare the box plot before and after scaling.

2.5. PCA: Perform all the required steps for PCA (use sklearn only) Create the covariance Matrix Get eigen values and eigen vector.

Statistical test is done before performing PCA. Though we have seen few corelation in the dataset. The ***Bartletts test of Sphericity*** is performed to understand corelation significance in the population.

The Null hypothesis an alternative hypothesis will be defined.

H0: All variables in the data are uncorrelated
Ha: At least one pair of variables in the dataset are correlated.

We Decide the significance level
Here we select α = 0.05.

We will reject the Null hypothesis (H0) if the p value is less than the significance level and if we cannot reject the null hypothesis the PCA will not be conducted.

```
#Confirm the statistical significance of correlations
#H0: Correlations are not significant, H1: There are significant correlations
#Reject H0 if p-value < 0.05
from factor_analyzer.factor_analyzer import calculate_bartlett_sphericity
chi_square_value,p_value=calculate_bartlett_sphericity(df_scaled)
p_value
```

```
0.0
```

Figure 31:Bartletts test of Sphericity

The p-value is 0, which is less than the significant level. Therefore, the null hypothesis will be rejected. Hence, it proves that at least one pair of variables in the dataset is correlated, and PCA will be performed.

And the next step would KMO test: (Kaiser-Meyer-Olkin)

The KMO test will be conducted to measure the sample adequacy (MSA) of the dataset.

If MSA is less than 0.5, PCA will not be suggested. Alternatively, if it is greater than 0.7, it gives a substantial reduction in dimension and extracts significant components.

```
from factor_analyzer.factor_analyzer import calculate_kmo
kmo_all,kmo_model=calculate_kmo(df_scaled)
kmo_model
```

```
0.8039889932781299
```

Figure 32:KMO Test

KMO is 0.80, Since it is greater 0.7 – the PCA is recommend.

Covariance Matrix:

```
]: cov_matrix=np.cov(df_scaled.T)
   print('covarience matrix: \n',cov_matrix)
```

```
covarience matrix:
 [[1.00156495 0.91760364 0.97210871 ... 0.53769433 0.76357722 0.73684378]
 [0.91760364 1.00156495 0.98417823 ... 0.5891007  0.84621844 0.71718181]
 [0.97210871 0.98417823 1.00156495 ... 0.572748   0.82894851 0.74775097]
 ...
 [0.53769433 0.5891007  0.572748   ... 1.00156495 0.61052325 0.52191235]
 [0.76357722 0.84621844 0.82894851 ... 0.61052325 1.00156495 0.88228018]
 [0.73684378 0.71718181 0.74775097 ... 0.52191235 0.88228018 1.00156495]]
```

```
#Variance covariance matrix
np.round(df_scaled.cov(),2)
```

	No_HH	TOT_M	TOT_F	M_06	F_06	M_SC	F_SC	M_ST	F_ST	M_LIT	...	MARG_CL_0_3_M	MARG_CL_0_3_F	MARG_AL_0_3_M	MARG
No_HH	1.00	0.92	0.97	0.80	0.80	0.78	0.83	0.15	0.17	0.93	...	0.56	0.56	0.07	
TOT_M	0.92	1.00	0.98	0.95	0.95	0.84	0.83	0.09	0.09	0.99	...	0.70	0.60	0.17	
TOT_F	0.97	0.98	1.00	0.91	0.91	0.82	0.83	0.12	0.13	0.99	...	0.66	0.60	0.14	
M_06	0.80	0.95	0.91	1.00	1.00	0.78	0.75	0.06	0.04	0.91	...	0.76	0.65	0.27	
F_06	0.80	0.95	0.91	1.00	1.00	0.77	0.74	0.07	0.05	0.91	...	0.76	0.65	0.26	
M_SC	0.78	0.84	0.82	0.78	0.77	1.00	0.99	-0.05	-0.05	0.82	...	0.67	0.57	0.18	
F_SC	0.83	0.83	0.83	0.75	0.74	0.99	1.00	-0.01	-0.01	0.82	...	0.65	0.59	0.16	
M_ST	0.15	0.09	0.12	0.06	0.07	-0.05	-0.01	1.00	0.99	0.09	...	0.12	0.20	0.03	
F_ST	0.17	0.09	0.13	0.04	0.05	-0.05	-0.01	0.99	1.00	0.09	...	0.12	0.22	0.02	
M_LIT	0.93	0.99	0.99	0.91	0.91	0.82	0.82	0.09	0.09	1.00	...	0.65	0.56	0.14	
F_LIT	0.93	0.93	0.96	0.83	0.83	0.72	0.73	0.10	0.10	0.97	...	0.55	0.49	0.09	
M_ILL	0.76	0.91	0.86	0.95	0.95	0.80	0.76	0.08	0.07	0.84	...	0.75	0.63	0.21	
F_ILL	0.86	0.89	0.89	0.86	0.87	0.83	0.85	0.14	0.15	0.84	...	0.71	0.67	0.20	
TOT_WORK_M	0.94	0.97	0.97	0.86	0.85	0.83	0.82	0.12	0.12	0.98	...	0.60	0.51	0.07	
TOT_WORK_F	0.93	0.81	0.88	0.68	0.69	0.71	0.78	0.27	0.29	0.82	...	0.49	0.55	0.12	
MAINWORK_M	0.93	0.93	0.94	0.79	0.79	0.78	0.78	0.11	0.11	0.95	...	0.47	0.39	-0.01	
MAINWORK_F	0.89	0.75	0.82	0.59	0.59	0.65	0.71	0.23	0.25	0.77	...	0.30	0.34	-0.03	
MAIN_CL_M	0.43	0.53	0.49	0.56	0.56	0.61	0.58	0.10	0.08	0.47	...	0.47	0.39	0.24	
MAIN_CL_F	0.38	0.36	0.39	0.38	0.38	0.36	0.39	0.19	0.20	0.33	...	0.31	0.37	0.37	
MAIN_AL_M	0.67	0.59	0.62	0.55	0.56	0.63	0.67	0.14	0.15	0.54	...	0.38	0.39	-0.04	
MAIN_AL_F	0.59	0.38	0.47	0.30	0.30	0.41	0.51	0.20	0.23	0.37	...	0.12	0.23	-0.11	
MAIN_HH_M	0.64	0.74	0.70	0.66	0.66	0.71	0.68	-0.03	-0.03	0.73	...	0.53	0.41	0.08	

Figure 33:Covariance Matrix

Eigen Values:

The below fig 34 represents the Eigen value of all 57 principal components.

```
var_exp=pca.explained_variance_
var_exp#eigen value
```

```
array([3.18135647e+01, 7.86942415e+00, 4.15340812e+00, 3.66879058e+00,
       2.20652588e+00, 1.93827502e+00, 1.17617374e+00, 7.51159086e-01,
       6.17053743e-01, 5.28300887e-01, 4.29831189e-01, 3.53440201e-01,
       2.96163013e-01, 2.81275560e-01, 1.92158325e-01, 1.36267920e-01,
       1.13389199e-01, 1.06303946e-01, 9.72885376e-02, 8.01062194e-02,
       5.76089954e-02, 4.43955966e-02, 3.78910846e-02, 2.96360194e-02,
       2.70797618e-02, 2.34458139e-02, 1.45111511e-02, 1.09852268e-02,
       9.31507853e-03, 8.13540203e-03, 7.89250253e-03, 5.02601514e-03,
       2.59771182e-03, 1.06789820e-03, 7.13559124e-04, 1.12676254e-30,
       1.06191652e-30, 2.47799812e-31, 2.47799812e-31, 2.47799812e-31,
       2.47799812e-31, 2.47799812e-31, 2.47799812e-31, 2.47799812e-31,
       2.47799812e-31, 2.47799812e-31, 2.47799812e-31, 2.47799812e-31,
       2.47799812e-31, 2.47799812e-31, 2.47799812e-31, 2.47799812e-31,
       2.47799812e-31, 2.47799812e-31, 2.47799812e-31, 1.41787114e-31,
       8.77457297e-32])
```

Figure 34: Eigen Value

Eigen Vector:

The below figure are the Eigen Vector of all the principal components. It derived by using .Components_

```
pca.components_#eigen vector
```

```
array([[ 0.15602058,  0.16711763,  0.16555318, ...,  0.13219224,
         0.15037558,  0.1310662 ],
       [-0.12634653, -0.08967655, -0.10491237, ...,  0.05081332,
        -0.06536455, -0.07384742],
       [-0.00269025,  0.05669762,  0.03874947, ..., -0.07871987,
         0.11182732,  0.1025525 ],
       ...,
       [ 0.        ,  0.35707146,  0.19665047, ...,  0.03591739,
        -0.0098253 , -0.02768831],
       [ 0.        ,  0.00998631, -0.00250899, ...,  0.02017974,
        -0.07939787,  0.04104616],
       [ 0.        , -0.21465651,  0.30242653, ...,  0.00471471,
        -0.1184565 ,  0.01737791]])
```

Figure 35: Eigen Vector

2.6 PCA: Identify the optimum number of PCs (for this project, take at least 90% explained variance). Show Scree plot.

The **Explained variance ratio** explain the proposition of variance of the principal components.

```
pca.explained_variance_ratio_
```

```
array([5.57260632e-01, 1.37844354e-01, 7.27529548e-02, 6.42641771e-02,
       3.86504944e-02, 3.39516923e-02, 2.06023855e-02, 1.31576386e-02,
       1.08085894e-02, 9.25395468e-03, 7.52911540e-03, 6.19101667e-03,
       5.18772384e-03, 4.92694855e-03, 3.36593119e-03, 2.38692984e-03,
       1.98617593e-03, 1.86206747e-03, 1.70414955e-03, 1.40317638e-03,
       1.00910494e-03, 7.77653131e-04, 6.63717190e-04, 5.19117774e-04,
       4.74341222e-04, 4.10687364e-04, 2.54183814e-04, 1.92422147e-04,
       1.63167083e-04, 1.42503342e-04, 1.38248605e-04, 8.80379297e-05,
       4.55026824e-05, 1.87057826e-05, 1.24990208e-05, 1.97368768e-32,
       1.86010049e-32, 4.34057237e-33, 4.34057237e-33, 4.34057237e-33,
       4.34057237e-33, 4.34057237e-33, 4.34057237e-33, 4.34057237e-33,
       4.34057237e-33, 4.34057237e-33, 4.34057237e-33, 4.34057237e-33,
       4.34057237e-33, 4.34057237e-33, 4.34057237e-33, 4.34057237e-33,
       4.34057237e-33, 4.34057237e-33, 4.34057237e-33, 2.48360653e-33,
       1.53699346e-33])
```

Figure 36: Explained Variance Ratio

Cumulative explained variance ratio:

The cumulative explained variance ratio to find a cut off for selecting the number of Principal components. (PCs)

```
np.cumsum(pca.explained_variance_ratio_)
```

```
array([0.55726063, 0.69510499, 0.76785794, 0.83212212, 0.87077261,
       0.9047243 , 0.92532669, 0.93848433, 0.94929292, 0.95854687,
       0.96607599, 0.97226701, 0.97745473, 0.98238168, 0.98574761,
       0.98813454, 0.99012071, 0.99198278, 0.99368693, 0.99509011,
       0.99609921, 0.99687687, 0.99754058, 0.9980597 , 0.99853404,
       0.99894473, 0.99919891, 0.99939134, 0.9995545 , 0.99969701,
       0.99983525, 0.99992329, 0.9999688 , 0.9999875 , 1.        ,
       1.        , 1.        , 1.        , 1.        , 1.        ,
       1.        , 1.        , 1.        , 1.        , 1.        ,
       1.        , 1.        , 1.        , 1.        , 1.        ,
       1.        , 1.        , 1.        , 1.        , 1.        ,
       1.        , 1.        ])
```

Figure 37:Cumulative explained variance ratio

We can see from the above fig.37 that Cumulative explained variance ratio of 6 pcs is more than 90%. Therefore, we can conclude by saying the optimum number of PCs is 6 and the below fig 38 Scree plot supports the same.

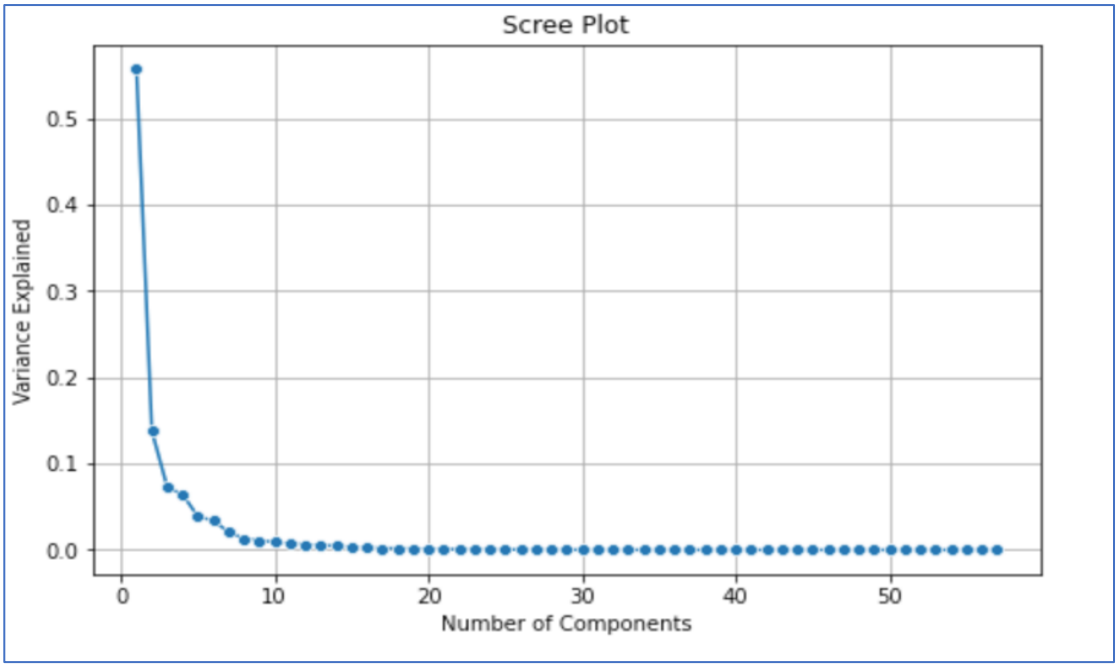

Figure 38: Scree Plot

The dots on the Scree plots are the 57 principal components. We can see a formation of the elbow at 6 PCs. It is evident that post 6 PCs, the drop is not that significant. Therefore the optimum number of PCs is 6.

2.7 PCA: Compare PCs with Actual Columns and identify which is explaining most variance. Write inferences about all the Principal components in terms of actual variables.

To categorise the pattern, the components are loaded against each feature in a new data frame. The below image shows that we have 6 principal components and one co-efficient each for all 57 variables.

```
df1_pca_loading = pd.DataFrame(pca.components_, columns = list(df_scaled))
df1_pca_loading.shape
```

```
(6, 57)
```

To analysis the variable that has the highest loading among the principle components. The component has to be loaded on a heatmap. For each variable with maximum loading, the heatmap shows a red rectangle box that is marked across the components. Ref Fig. 39

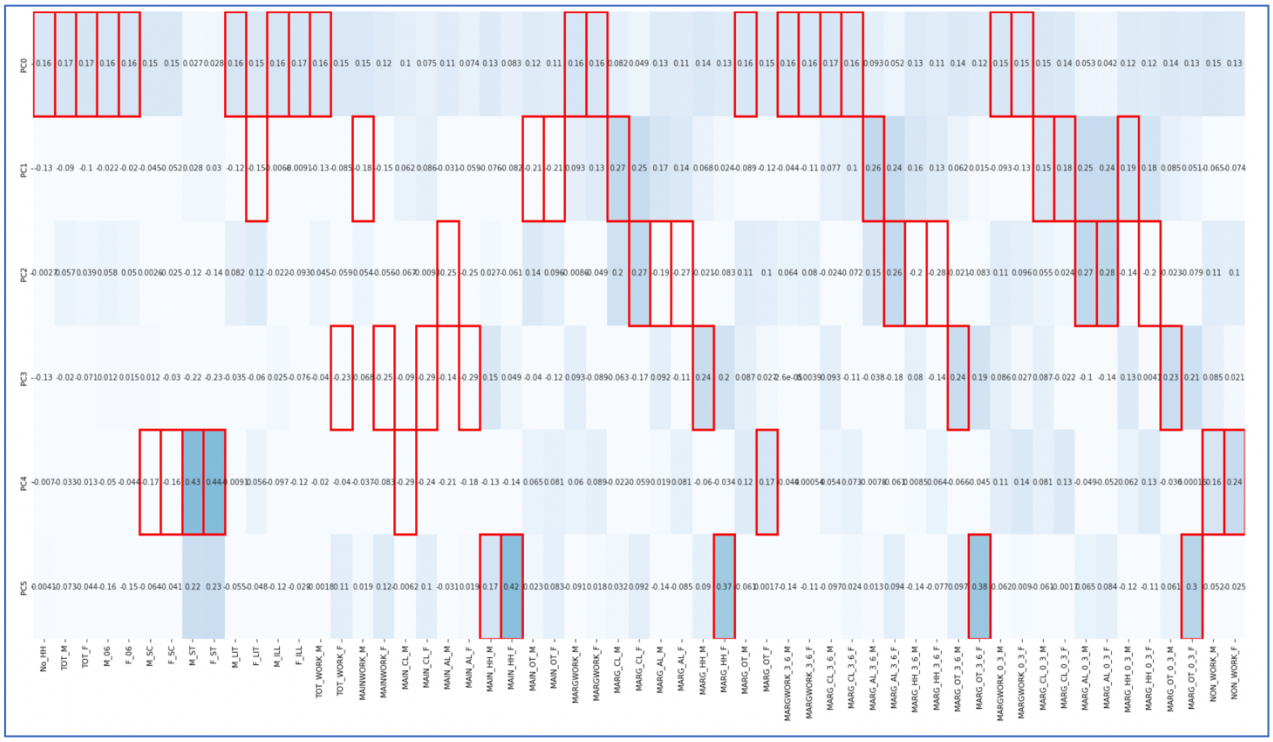

Figure 39: Heatmap

The below table shows the variables that contributes maximum towards the respective PCs:

Principle Components	PC0	PC1
Highest Contributing Variables	No of Household	Literates population Female
	Total population Male	Main Working Population Male
	Total population Female	Main Other Workers Population Male
	Population in the age group 0-6 Male	Main Other Workers Population Female
	Population in the age group 0-6 Female	Marginal Cultivator Population Male
	Literates population Male	Marginal Agriculture Labourers Population 3-6 Male
	Illiterate Male	Marginal Cultivator Population 0-3 Male
	Illiterate Female	Marginal Cultivator Population 0-3 Female
	Total Worker Population Male	Marginal Household Industries Population 0-3 Male
	Marginal Worker Population Male	
	Marginal Worker Population Female	
	Marginal Other Workers Population Male	
	Marginal Worker Population 3-6 Male	
	Marginal Worker Population 3-6 Female	
	Marginal Cultivator Population 3-6 Male	
	Marginal Cultivator Population 3-6 Female	
	Marginal Worker Population 0-3 Male	
	Marginal Worker Population 0-3 Female	
Principle Components	**PC2**	**PC3**
Highest Contributing Variables	Main Agricultural Labourers Population Male	Total Worker Population Female
	Marginal Cultivator Population Female	Main Working Population Female
	Marginal Agriculture Labourers Population Male	Main Cultivator Population Female
	Marginal Agriculture Labourers Population Female	Main Agricultural Labourers Population Female
	Marginal Agriculture Labourers Population 3-6 Female	Marginal Household Industries Population Male
	Marginal Household Industries Population 3-6 Male	Marginal Other Workers Population Person 3-6 Male
	Marginal Household Industries Population 3-6 Female	Marginal Other Workers Population 0-3 Male
	Marginal Agriculture Labourers Population 0-3 Male	
	Marginal Agriculture Labourers Population 0-3 Female	
	Marginal Household Industries Population 0-3 Female	
Principle Components	**PC4**	**PC5**
Highest Contributing Variables	Scheduled Castes population Male	Main Household Industries Population Male
	Scheduled Castes population Female	Main Household Industries Population Female
	Scheduled Tribes population Male	Marginal Household Industries Population Female
	Scheduled Tribes population Female	Marginal Other Workers Population Person 3-6 Fema
	Main Cultivator Population Male	Marginal Other Workers Population 0-3 Female
	Marginal Other Workers Population Female	
	Non Working Population Male	
	Non Working Population Female	

2.8 PCA: Write linear equation for first PC.

The below fig 40 depicts the linear equation for the first PC (PC0):

The value in the Parentheses are the coefficient and those multiplied by variables.

```
02]: for i in range(0,57):
         print("(",np.round(pca.components_[0][i],2),")",'*',df_scaled.columns[i], end=' + ')

( 0.16 ) * No_HH + ( 0.17 ) * TOT_M + ( 0.17 ) * TOT_F + ( 0.16 ) * M_06 + ( 0.16 ) * F_06 + ( 0.15 ) * M_SC + ( 0.
15 ) * F_SC + ( 0.03 ) * M_ST + ( 0.03 ) * F_ST + ( 0.16 ) * M_LIT + ( 0.15 ) * F_LIT + ( 0.16 ) * M_ILL + ( 0.17 )
* F_ILL + ( 0.16 ) * TOT_WORK_M + ( 0.15 ) * TOT_WORK_F + ( 0.15 ) * MAINWORK_M + ( 0.12 ) * MAINWORK_F + ( 0.1 ) *
MAIN_CL_M + ( 0.07 ) * MAIN_CL_F + ( 0.11 ) * MAIN_AL_M + ( 0.07 ) * MAIN_AL_F + ( 0.13 ) * MAIN_HH_M + ( 0.08 ) *
MAIN_HH_F + ( 0.12 ) * MAIN_OT_M + ( 0.11 ) * MAIN_OT_F + ( 0.16 ) * MARGWORK_M + ( 0.16 ) * MARGWORK_F + ( 0.08 )
* MARG_CL_M + ( 0.05 ) * MARG_CL_F + ( 0.13 ) * MARG_AL_M + ( 0.11 ) * MARG_AL_F + ( 0.14 ) * MARG_HH_M + ( 0.13 )
* MARG_HH_F + ( 0.16 ) * MARG_OT_M + ( 0.15 ) * MARG_OT_F + ( 0.16 ) * MARGWORK_3_6_M + ( 0.16 ) * MARGWORK_3_6_F +
( 0.17 ) * MARG_CL_3_6_M + ( 0.16 ) * MARG_CL_3_6_F + ( 0.09 ) * MARG_AL_3_6_M + ( 0.05 ) * MARG_AL_3_6_F + ( 0.13
) * MARG_HH_3_6_M + ( 0.11 ) * MARG_HH_3_6_F + ( 0.14 ) * MARG_OT_3_6_M + ( 0.12 ) * MARG_OT_3_6_F + ( 0.15 ) * MAR
GWORK_0_3_M + ( 0.15 ) * MARGWORK_0_3_F + ( 0.15 ) * MARG_CL_0_3_M + ( 0.14 ) * MARG_CL_0_3_F + ( 0.05 ) * MARG_AL_
0_3_M + ( 0.04 ) * MARG_AL_0_3_F + ( 0.12 ) * MARG_HH_0_3_M + ( 0.12 ) * MARG_HH_0_3_F + ( 0.14 ) * MARG_OT_0_3_M +
( 0.13 ) * MARG_OT_0_3_F + ( 0.15 ) * NON_WORK_M + ( 0.13 ) * NON_WORK_F +
```

Figure 40:Linear Equation