Clustering Customer Demographics

PhillipCapital Assignment for Data Scientist Role

Madhav Mittal

1st April 2022

Contents

1. Introduction

Clustering is an unsupervised machine learning task that divides the data points into several groups such that data points in the same groups are more like other data points in the same group and dissimilar to the data points in other groups.

Clustering has found its way into the mainstream today. Clustering algorithms are used in various applications such as market research and customer segmentation, biological data and medical imaging, search result clustering etc.

In this paper, the author attempts to use different clustering algorithms on a customer demographic dataset. These algorithms are compared, and a final model is chosen among them.

2. Background

There are many available clustering algorithms that can be chosen for any given machine learning. Some of them are K-Means, DBSCAN, MeanShift, Hierarchical Clustering etc. **However, since the given dataset has a majority of categorical features, thus we choose K-Means based methods.**

2.1 K-Means

K-means clustering is one of the simplest and most popular unsupervised machine learning algorithms. It works by minimizing within cluster variances and typically converges within a few iterations. It is widely used because of its simplicity and ease of use, coupled with its speed.

The K-means algorithm starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster. After this initialization, the K-means algorithm starts an iterative process with two steps -

1. Expectation - Assigning each point of the dataset to its closest centroid
2. Maximization - Computing the new centroid of each cluster

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \left\| x_n - \mu_k \right\|^2$$

Figure 1: K-means Cost Function

This is done until either the centroid positions don't change, or a stopping condition like number of in-step iterations is reached.

2.2 K-Modes

K-modes is used to cluster categorical datasets. K-means uses mathematical measures (distance) to cluster continuous data. The lesser the distance, the more similar our data points are.

$$d_1(X, Y) = \sum_{j=1}^{m} \delta(x_j, y_j)$$

where

$$\delta(x_j, y_j) = \begin{cases} 0 & (x_j = y_j) \\ 1 & (x_j \neq y_j) \end{cases}$$

Figure 2: K-Modes Cost Function

But for categorical data points, we cannot calculate the distance. Thus, KModes is used. It uses the dissimilarities(total mismatches) between the data points. The lesser the dissimilarities the more similar the data points are. It uses modes instead of means.

2.3 K-Prototypes

K-Prototypes is a lesser-known sibling of K-Means but offers an advantage of working with mixed data types. It measures distance between numerical features using Euclidean distance (like K-means) but also measure the distance between categorical features using the number of matching categories.

Thus, the dis-similarity equation is written as a sum of the previously mentioned cost functions as shown in Figure 3.

$$d_2(X, Y) = \sum_{j=1}^{p} (x_j - y_j)^2 + \gamma \sum_{j=p+1}^{m} \delta(x_j, y_j)$$

Figure 3: K-Prototypes Cost Function

Thus, K-Prototypes is thus suitable for datasets consisting of both – continuous as well as categorical variables.

3. Pre-Processing

Data Pre-processing includes the steps we need to follow to transform or encode data so that it may be easily parsed by the machine. The main agenda for a model to be accurate and precise in predictions is that the algorithm should be able to easily interpret the data's features.

We employ several pre-processing steps to clean our data.

3.1 Dropping Null Values

We first find the columns that have null values.

```
#Missing Data
df_main.isnull().sum()
```

```
AGE                    0
OccupationCategory    17
AnnualIncome         177
Category               0
dtype: int64
```

Figure 4: Finding Null Values

We find out that there are 17 null values in *OccupationCategory* and 177 in *AnnualIncome*, thus we drop those rows from our data frame.

3.2 Dropping Age Value Equal to 0

We also find out that several rows had *AGE* = 0 which did not make sense, and it was probably an error. Thus, we dropped the rows containing *AGE* = 0.

4. Exploratory Data Analysis

It is difficult to visualize a plot with categorical features using normal plots, and thus we use Seaborn's *catplot* and *barplot* to visualize the dataset.

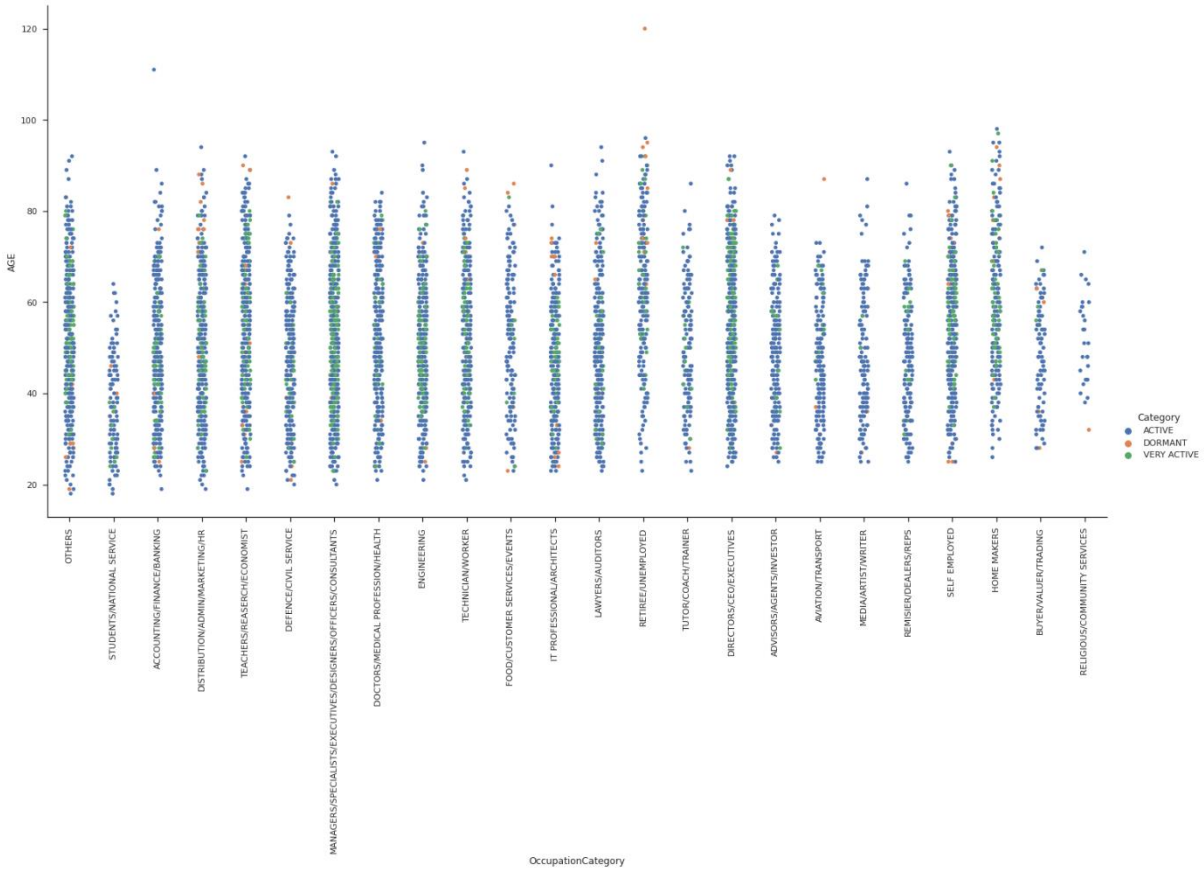

Figure 5: OccupationCategory vs Age Catplot

Figures 5 and 6 shows the relationship between OccupationCategory, Age and Category features. It is very clear that as the age is increasing, dormant people are also increasing. It can also be seen that some professions are more active than other professions, regardless of age. For example, Engineers, CEO's, and Self-Employed people are much more active than Students and Religious.

This information can be used by the model for clustering.

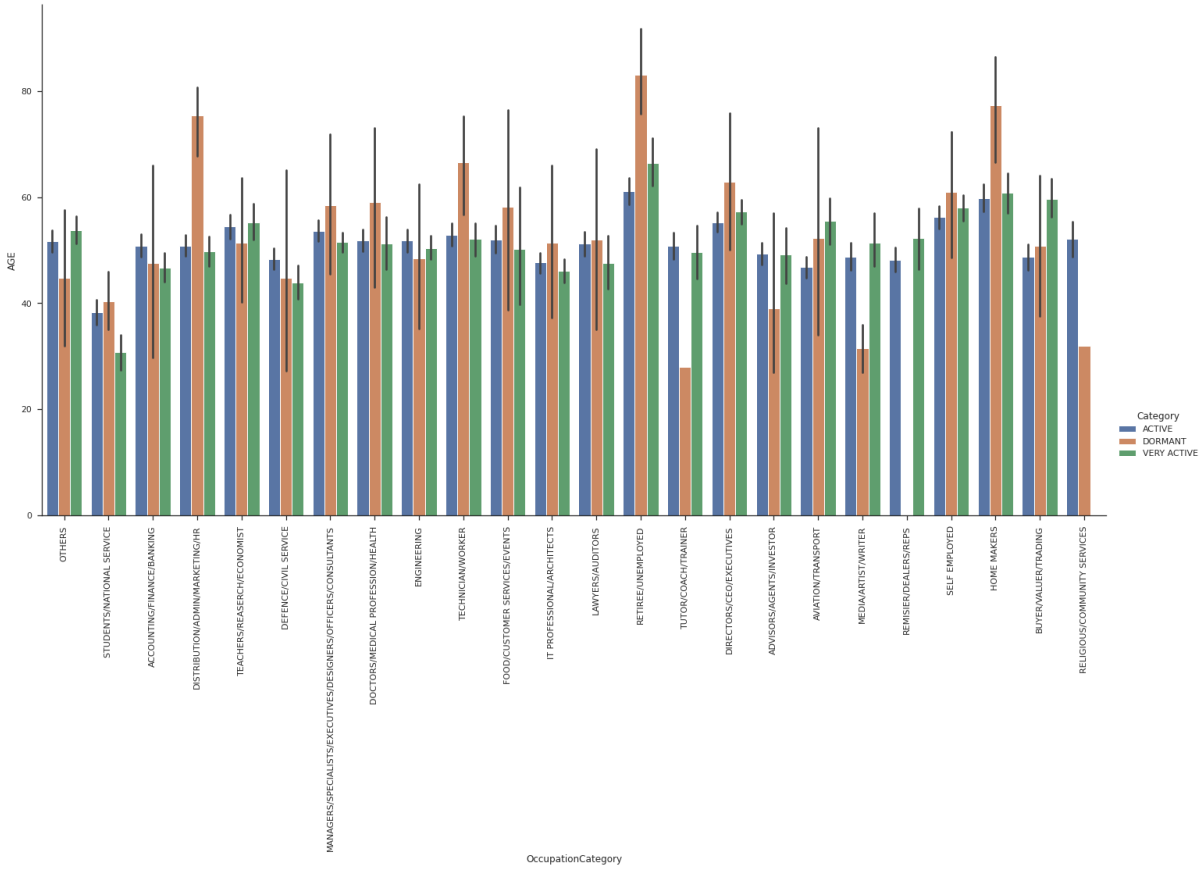

Figure 6: OccupationCategory vs Age Catplot

Figure 7 tells us that the most dormant people lie within the 30-50k AnnualIncome category. Also, the age remains almost constant for different AnnualIncomes, except for dormant. For dormant people, high aged people seem to have below 30k salary.

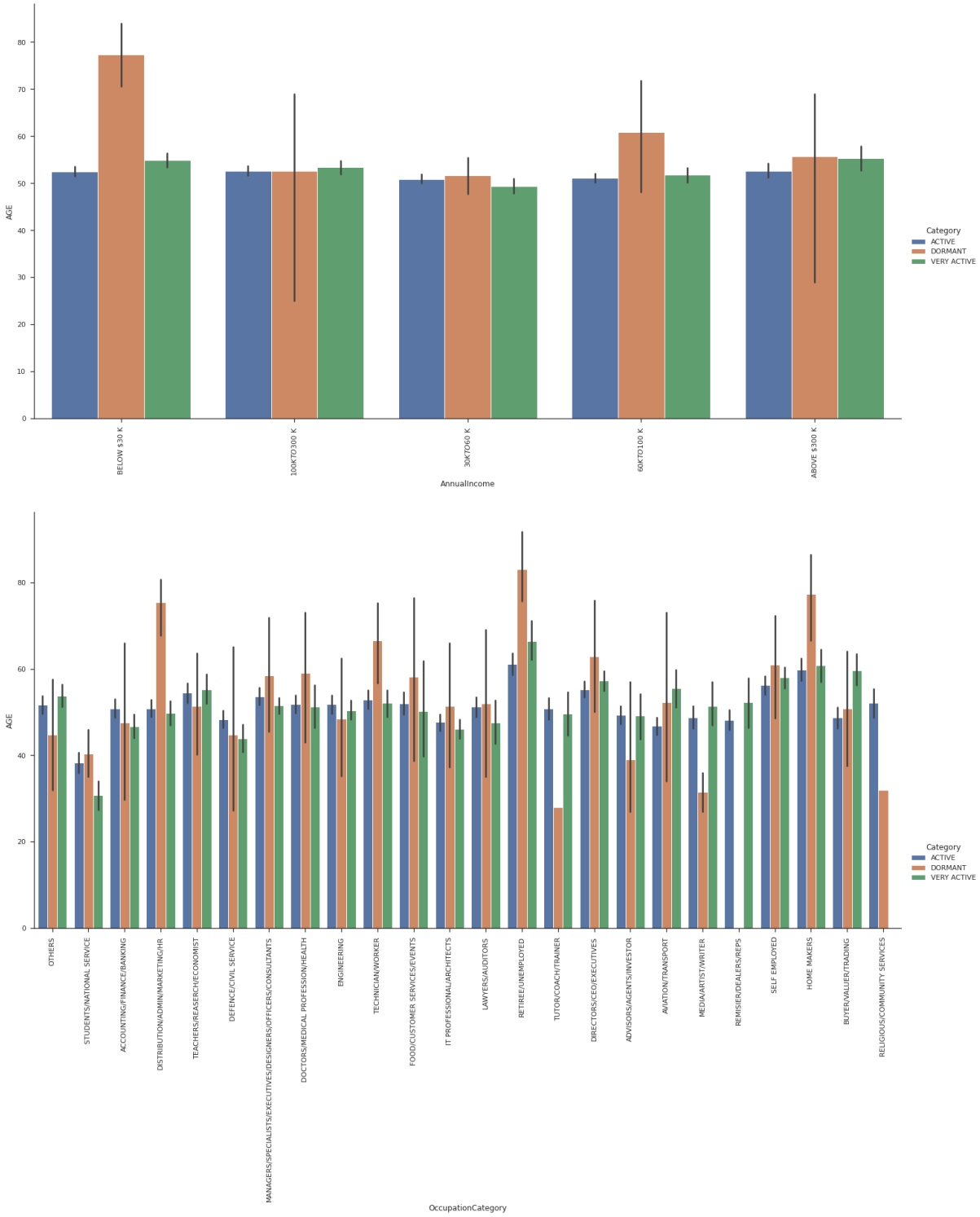

Figure 7: Catplot and Barplot of *OccupationCategory* vs *Age*

5. Methodology

For this project, we have selected 3 models -

1. K-Modes
2. K-Prototype
3. K-Means

5.1 K-Modes

5.1.1 Creating *Age Group* Feature

Since K-Modes only accepts categorical values, and the column *AGE* is a continuous integer, thus the algorithm interprets the individual integer values as categories. This is not ideal, and thus we create a new feature called *Age Group* from the *AGE* feature.

The *Age Group* is defined by binning the *AGE* feature. Bins are made for the following ages–

1. 18 – 36
2. 36 – 54
3. 54 – 72
4. 72 – 88
5. 88 – 120

The *AGE* feature is subsequently dropped.

5.1.2 Plotting the Elbow Graph

Since we need to specify the number of clusters in K-Modes beforehand, the performance of the model is greatly dependent on the value of *K*. Thus, we run K-Modes for values of *K* from 1 – 30.

We then plot the cost for each model vs K and find the elbow. We choose this value to be the optimal *K* value for K-Modes. The plot can be seen in Figure 8.

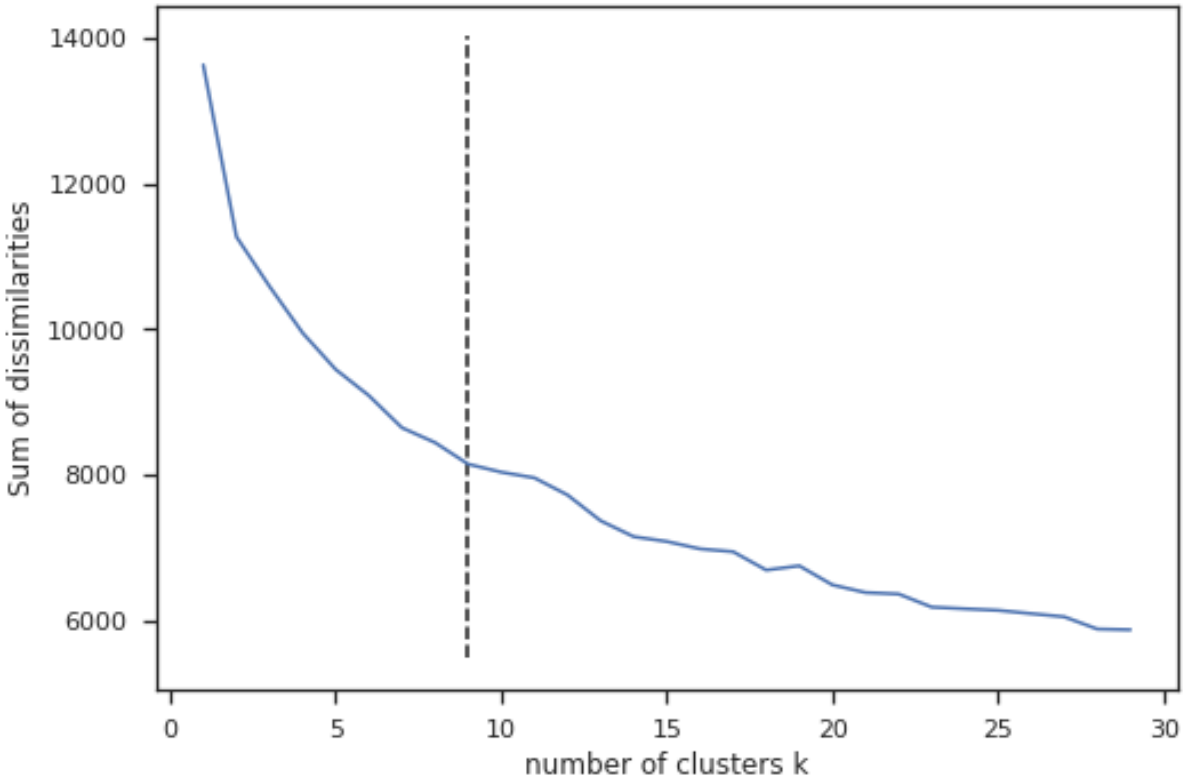

Figure 8: Elbow Plot for K-Modes

The ideal elbow is found to be at *K*=8, with the cost 8442.0.

5.2 K-Prototypes

5.2.1 Normalizing the *Age Group* Feature

Since K-Prototypes can process both continuous as well as categorical variables, thus we keep the *AGE* feature and drop the *Age Group* feature. Additionally, we scale *AGE* to have a zero mean and unit variance using the *StandardScaler* library.

5.2.2 Plotting the Elbow Graph

We plot the elbow plot again by training the model for 1 to 30 number of clusters and recording the cost function, as seen in Figure 9.

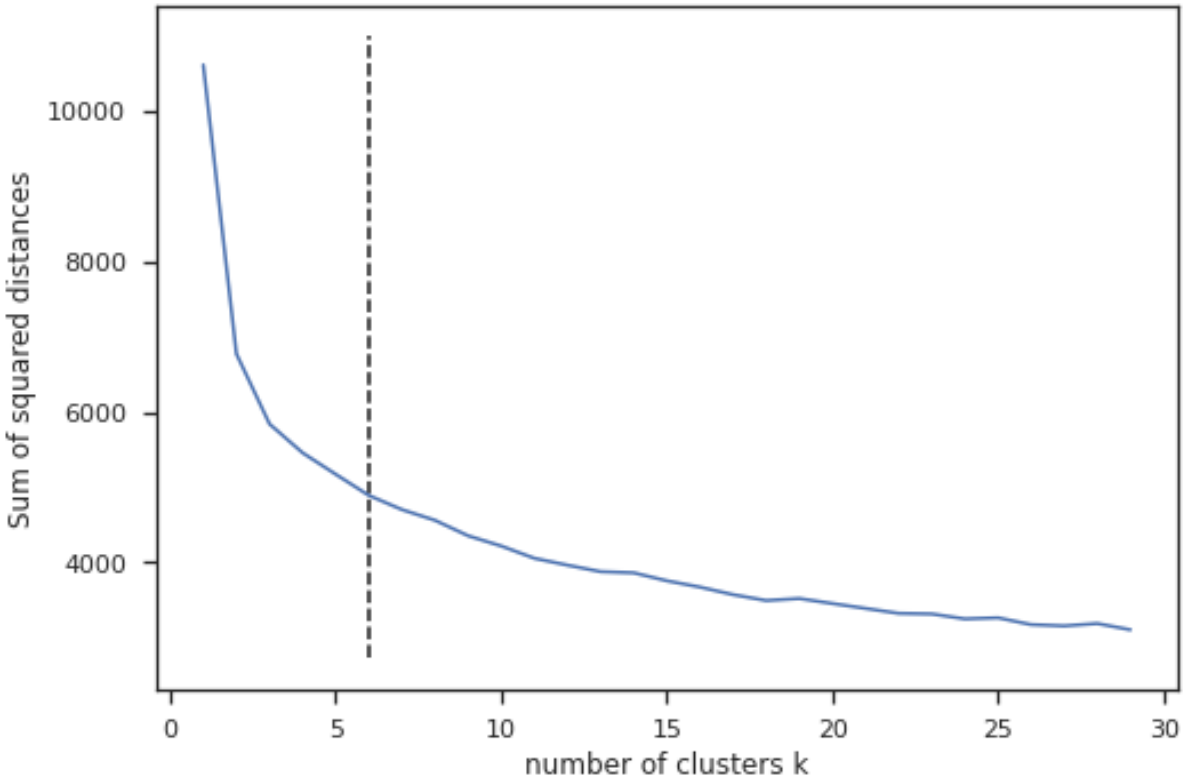

Figure 9: K-Prototypes Elbow Plot

The ideal elbow is found to be at *K*=6, with the cost 4891.35.

5.3 K-Means

5.3.1 One-Hot Encoding the Dataset

Since K-Means only works on continuous data, thus we one-hot encode the entire dataset, excluding the *AGE* feature. We leave the *AGE* feature as it is since it is already a continuous feature. However, we normalize it using *StandardScaler*.

5.3.2 Controlling Curse of Dimensionality by Explained Variance Plot

One-hot encoding all features can lead to the creation of a dataset with a huge number of columns. This in turn could lead the course of dimensionality, which can harm our model. Thus, we need to make sure that there are not too many features in our dataset.

This is done by plotting an explained variance graph. Evaluating the cumulative explained variance ratio is a reliable method to choose the number of components on PCA, which performs the dimensionality reduction while maximizing the variance between the original data and the projected data.

We keep a variance threshold of 99%, so as to root out the useless features who contribute less than 1%. The plot can be found in Figure 10.

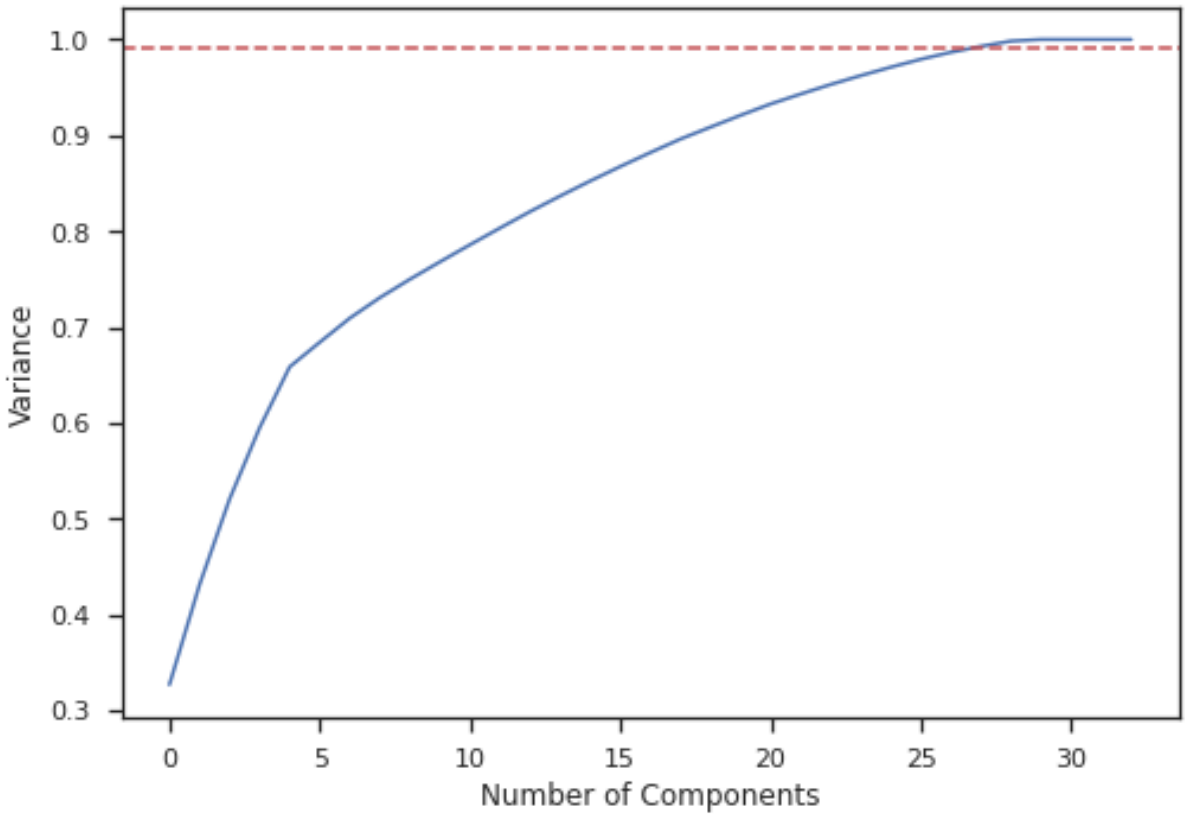

Figure 10: Number of components vs explained variance plot

It can be seen that features till feature number 29 provide a covariance ratio of 99.4%, after which the features do not contribute much. Thus, using PCA, we reduce the features from 34 to 29.

The dataset is now ready for K-Means

5.3.3 Plotting the Elbow Graph

We plot the dataset acquired from PCA for values of K = 1 to K = 30, as seen in Figure 11.

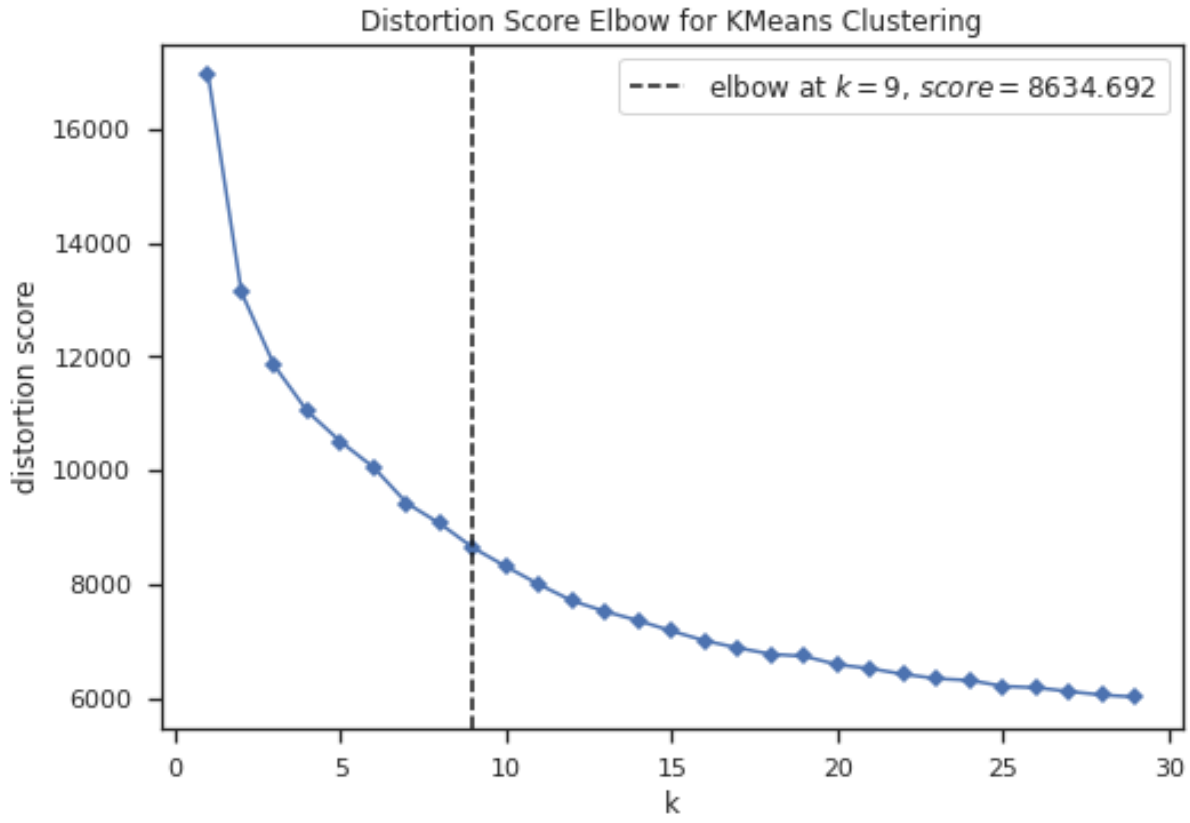

Figure 11: K-Means Elbow Plot

The ideal K-Means elbow is found to be at k=9 with cost=8634.7

6. Analysing Results for K-Prototypes

The ideal number of K-Prototypes was found to be at K=6, with the cost 4891.35. The cluster centres can be found in Figure 12.

	NumData	AGE	Occupation	CategoryAnnual	IncomeCategory
Cluster 0	831	-0.3778234710675814	ACCOUNTING/FINANCE/BANKING	$100 K TO $300 K	ACTIVE
Cluster 1	899	0.028614654062942318	ENGINEERING	BELOW $30 K	ACTIVE
Cluster 2	1049	1.4492153937577696	DIRECTORS/CEO/EXECUTIVES	BELOW $30 K	ACTIVE
Cluster 3	1122	0.5237107906952523	MANAGERS/SPECIALISTS/EXECUTIVES/DESIGNERS/OFFI...	$30 K TO $60 K	ACTIVE
Cluster 4	816	-0.9307141111939577	DEFENCE/CIVIL SERVICE	$30 K TO $60 K	ACTIVE
Cluster 5	801	-1.3234968914146008	MANAGERS/SPECIALISTS/EXECUTIVES/DESIGNERS/OFFI...	BELOW $30 K	ACTIVE

Figure 12: Cluster Centres using K-Prototypes

The final 3-D graph can be shown as follows.

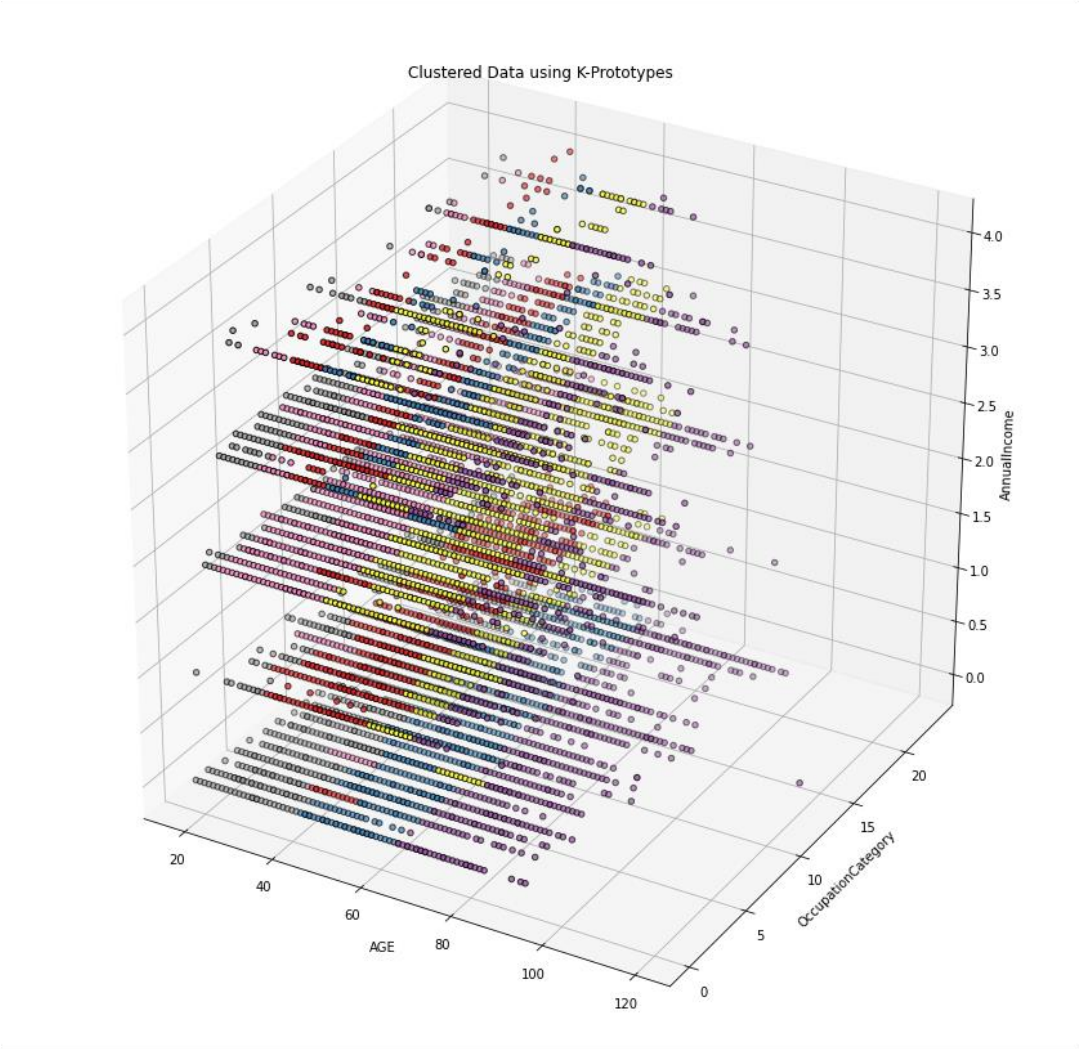

Figure 13: 3-D Graph After Clustering using K-Prototypes

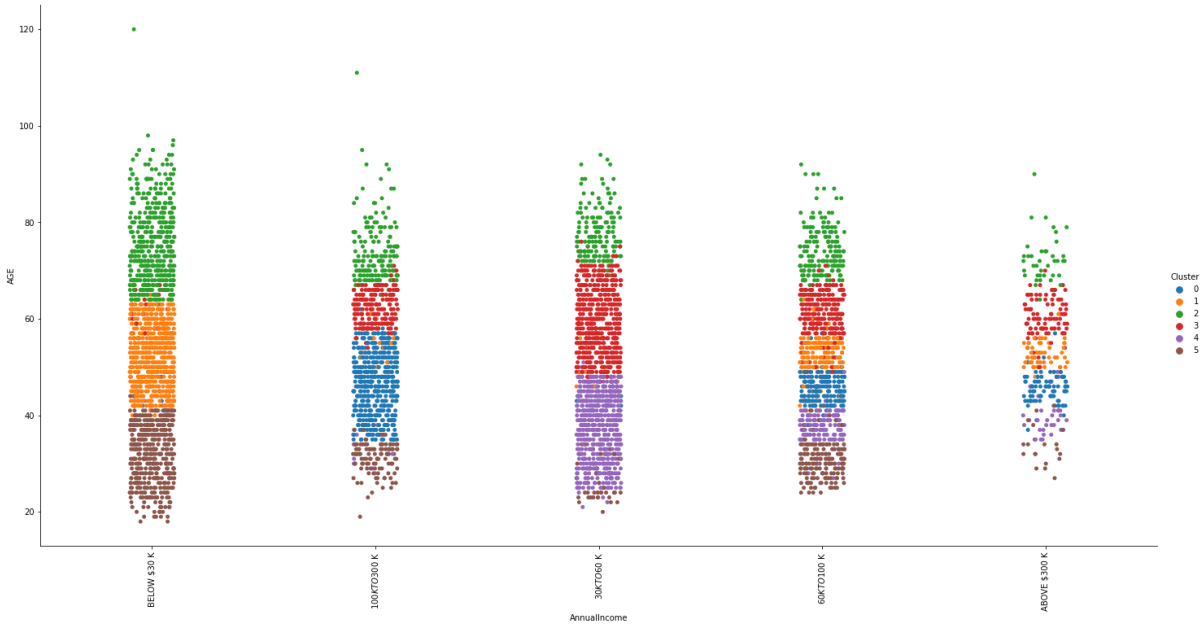

Figure 14: AnnualIncome vs Age vs cluster

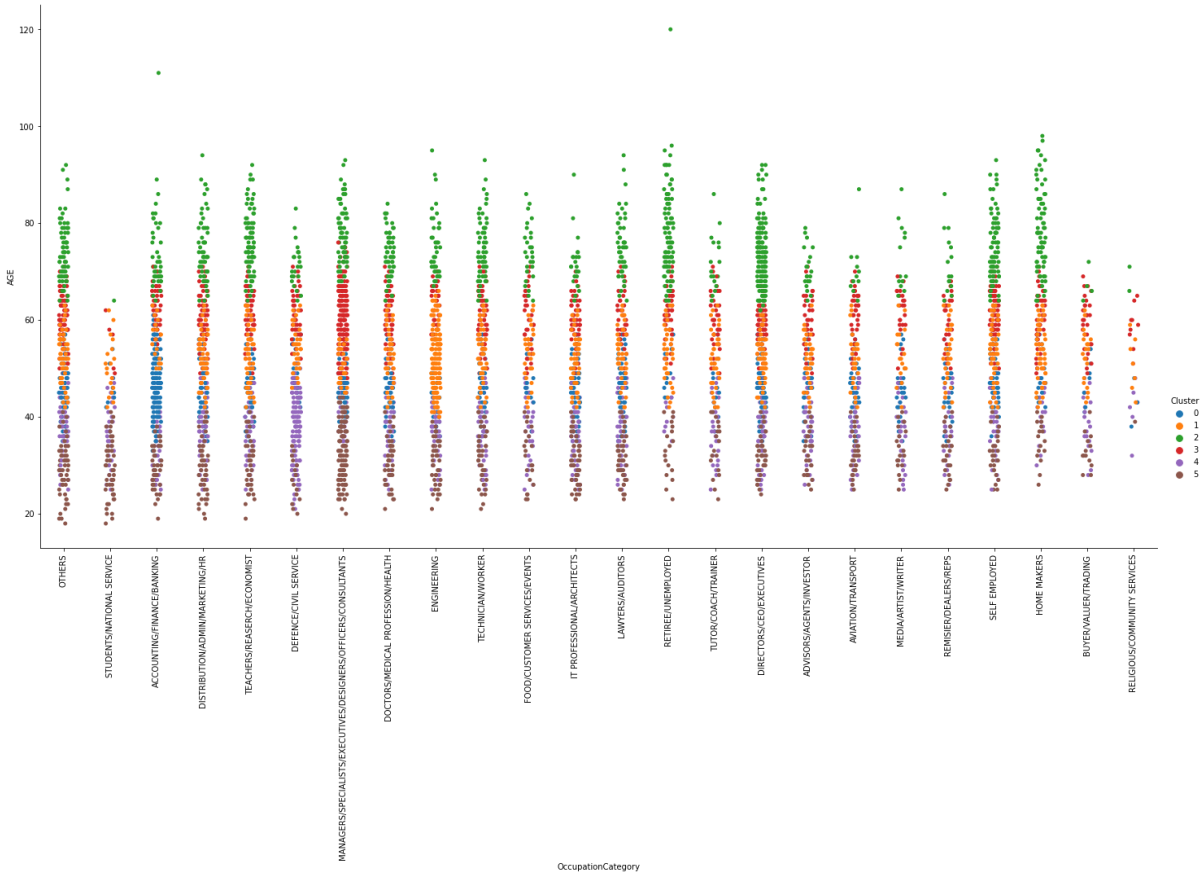

Figure 15: OccupationCategory vs Age vs cluster

From the above figures, it can clearly be seen that a lot of importance has been given to the Age continuous variable the cluster seem to have been formed according to increase in Age.

The next most important feature seems to be the Annual Income, since most of the individual annual income features categories have datapoints belonging to either 3 or 5 clusters, never 6. This means that Annual Income is a good differentiator and thus an important feature.

The OccupationCategory feature, similar to the AnnualIncome feature is an important feature, since each category only has 4, or 5 cluster datapoints. However, the divide due to this feature is more homogenous, and thus, this feature is not as strong as the previous one.

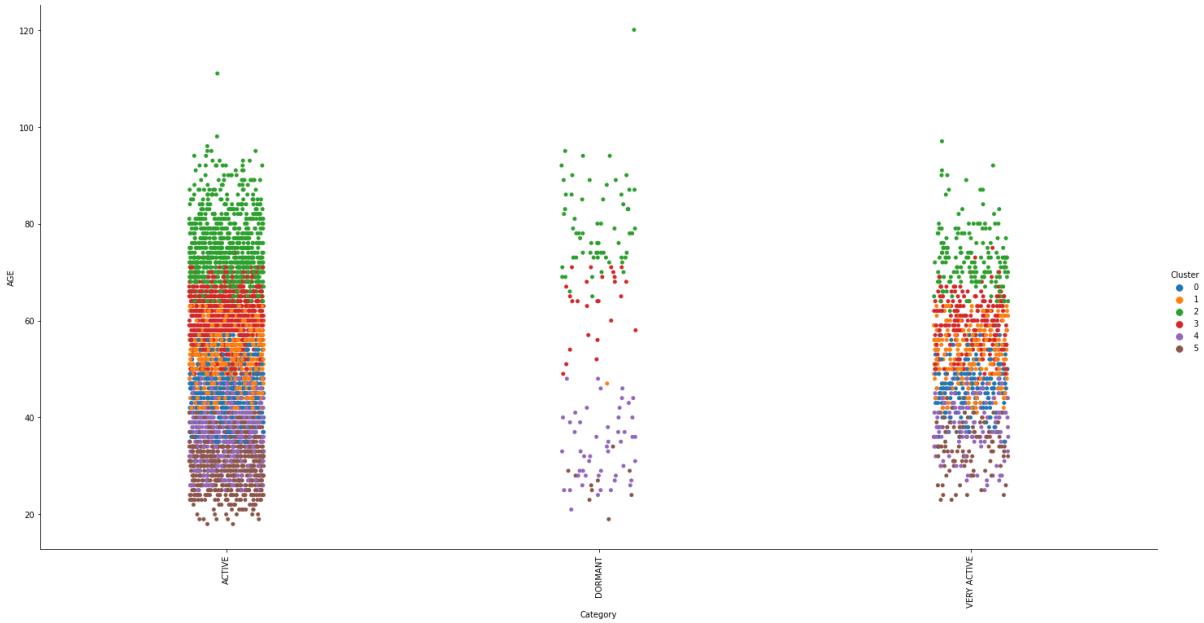

Figure 16: Category vs Age vs cluster

From Figure 16, it is noticed that the dormant category does not have any features from cluster 1, but other than that, all categories have every cluster datapoints, with no clear divide.

Thus, age was the most important feature, followed by Income, Occupation and then Category.

7. Conclusion

Ideally, K-Means, K-Modes and K-Prototypes use different cost functions and thus their performances cannot be compared using their cost functions. However, since we one-hot encode the categorical variables, thus, the sum of squared distances cost function of K-Means devolves into the dissimilarity-based cost function.

The table of results can be seen in Table 1.

Model	Number of Clusters	Cost
K-Modes	8	8442
K-Prototypes	6	4891
K-Means	9	8647

Table 1: Comparison of Performances of All Models

We conclude that K-Prototypes outperforms all the other models on this dataset.

Firstly, K-prototypes seemed to evenly consider categorical and continuous features. Meanwhile, K-means seemed to weigh categorical features much more heavily, which would likely be undesirable.

Secondly, K-Modes loses important information when we convert the feature *AGE* from continuous to discrete. Also,

K-Means does not perform well because one-hot encoding every categorical variable is not the ideal way to feature engineer a dataset. It leads to sparse matrices, that are difficult for many clustering algorithms to deal with.

Finally, K-Prototypes outperforms the other models because it does not involve any information loss due to feature engineering, and it does not need one-hot encoding the categorical variables. Since it can work with both continuous and categorical variables, thus it takes both sum of squared differences as well sum of dissimilarities into account, making it the ideal clustering algorithm.

Additionally, we notice that the number of clusters chosen by K-Means and K-Modes are a few too many. From a business point-of-view, the number of demographics we should cluster

should be reasonably less, so as to derive meaningful conclusions from them. Since K-Prototypes clusters the customers into 6 demographics, thus, it is the chosen algorithm.