

Unsupervised Machine Learning

IBM Skills Network

Project Report

By:

Fahd Seddik

Table of Contents

[Main Objective 3](#_Toc113483322)

[Brief Description 3](#_Toc113483323)

[Data cleaning & Feature engineering 4](#_Toc113483324)

[Unsupervised Models 6](#_Toc113483325)

[Recommended Model 7](#_Toc113483326)

[Key Findings and Insights 7](#_Toc113483327)

[Suggestions 7](#_Toc113483328)

# Main Objective

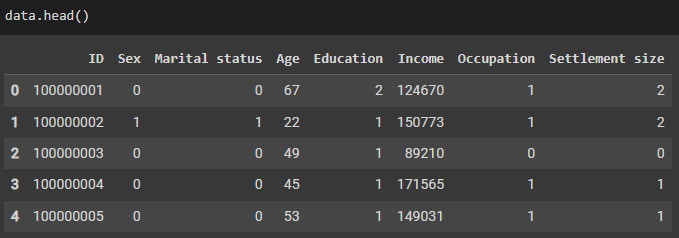
As a company, you would want to know which of your customers would prefer a certain product or maybe you’re making a new campaign and you’re wondering how much percent of your customers would react positively to that campaign. This would be solved by using customer segmentation.

**The target is to do customer segmentation based on customer’s information using several clustering algorithms.**

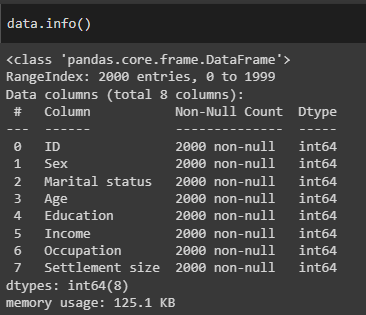
# Brief Description

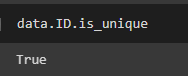
Customer Segmentation is to subdivide customers of a market into discrete customer groups that share similar characteristics. Customer Segmentation can be a powerful means to identify unsatisfied customer needs. Using the above data companies can then outperform the competition by developing uniquely appealing products and services. Our data set consists of **2000** **examples** and we have **7 features** excluding an ID column. The data represents information about customers. The features are as follows:

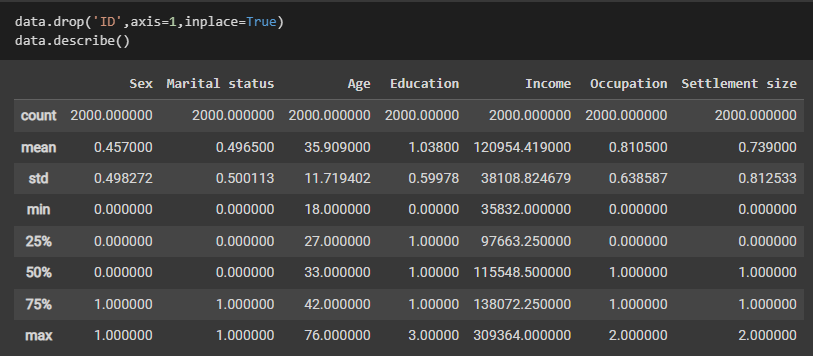
* ID (unique)
* Sex
* Marital status
* Age
* Education
* Income
* Occupation
* Settlement size

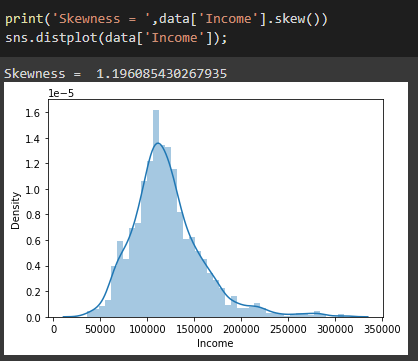
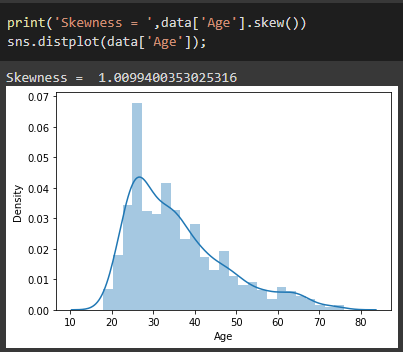


# Data cleaning & Feature engineering

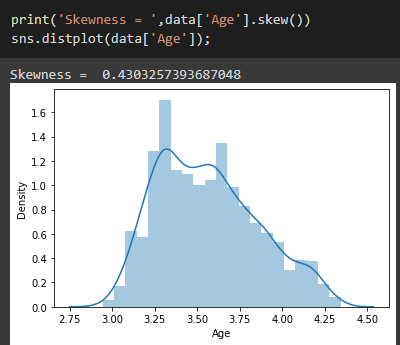
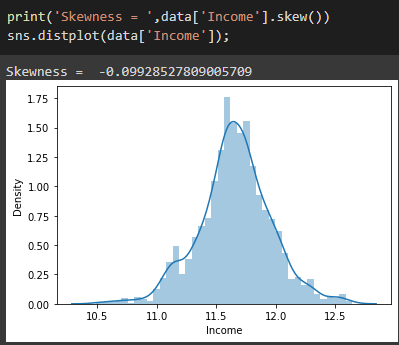
We want to first examine our data set and check for any missing values. After that, we will be checking for duplicates by using the ‘ID’ column to check – for it being unique for each customer. We would use data.info() to check for missing values. As shown below, we can see that there are no missing values as we have 2000 non-null out of 2000 entries and there are no missing values.



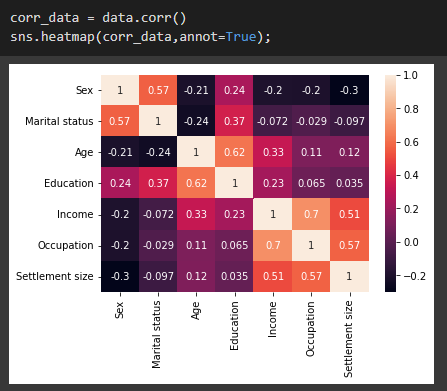
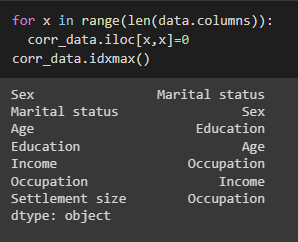
As we can see the features Sex,Marital status,Education, Occupation, and Settlement size are all ordinal encoded. We want to check for skewness values for Age, and Income.

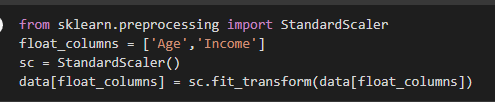


As we can see from the distplots above, the columns Age, and Income are both skewed with positive skewness. We will correct this skewness with np.log1p. Shown below are the results of the log transformation performed on each of the two columns.

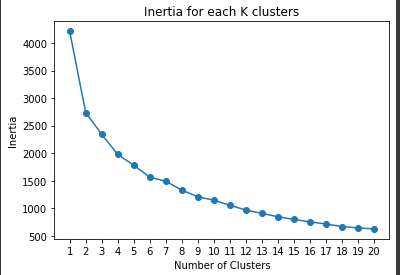


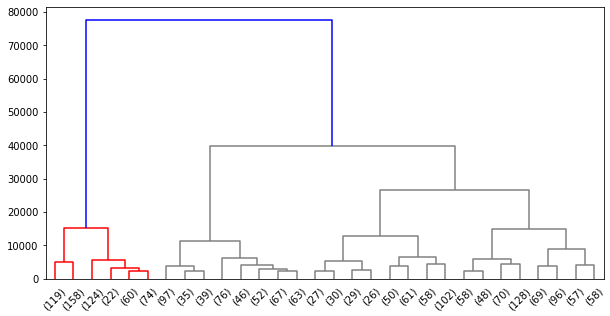
We will visualize correlation between our features in order to have a basic idea of which of them correlate with which of the other features. As you can see below, there are the most correlation features printed together on each row using the idxmax() function. We will then apply standard scaling.

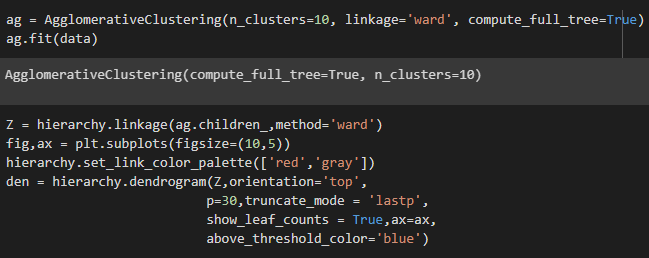


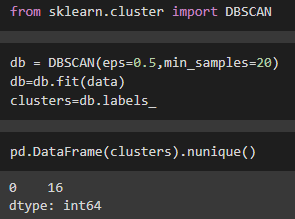


# Unsupervised Models

We will be using **K-Means**, **AgglomerativeClustering**, and **DBSCAN** as our clustering models. We will first examine the K-Means model as a base-line. We will be using the MiniBatch version as it is fast. We will be using the elbow method to determine which k is best using an inertia plot. However, the plot did not give us any idea of the best value for k.

We will then train our AgglomerativeClustering model and try to visualize the dendrogram of the model using **ward** **linkage**.



For our last model we will be using DBSCAN to try to not specify number of clusters and see if we can better cluster our data. However, DBSCAN requires two parameters and finding appropriate values of epsilon and n\_clu can be difficult. We have used a eps=0.5 and min\_samples=20 for our DBSCAN model which lead to group our data into 16 clusters.

# Recommended Model

Without a doubt, each of our models did its best to try to cluster our data given the specific hyperparameters of each of them. However, we can see that a DBSCAN can be hard to train and find the right hyperparameters. Therefore, we recommend the usage of **AgglomerativeClustering** as the recommended model in order to help visualize our customer segmentation more. This would help us negotiate with decision makers and lead to moderate results.

# Key Findings and Insights

We found that using elbow method in an inertia plot for a K-Means algorithm does not always yield the results we want. As seen in previous sections, the curve for inertia did not have any “elbow” inflection point. On the contrary, a model like Agglomerative Clustering can be easily visualized using dendrograms which would help us in many data sets specially in customer segmentation to visualize groups of our customers and better understand their needs.

# Suggestions

It is certain that improvements can be made for all our models. Some of the suggestions would be to use grid search to choose hyperparameters of our model. We can also try and to use other evaluation metrics. On top of that, the Mean-shift model might be able to perform better on such a problem since it eliminates having to choose a value for “k”. Lastly, if the data set contained more features we can use PCA or LDA to try to figure out underlying correlations.