Machine Learning Engineer Nanodegree Capstone Project Report

Arvato Financial Solutions Customer Segmentation

Ahmed Khaled Metwally

8nd October 2022

Project Overview

Project Domain

Arvato is a services company that provides financial services, Information Technology (IT) services and Supply Chain Management (SCM) solutions for business customers on a global scale. It develops and implements innovative solutions with a focus on automation and data analytics. Arvato's customers come from a wide range of industries such as insurance companies, e-commerce, energy providers, IT and Internet providers[1].
Arvato is wholly owned by Bertelsmann, which is a media, services and education company[2].
Arvato is helping its customers get valuable insights from data in order to make business decisions. Customer centric marketing is one of the growing fields. Identifying hidden patterns and customer behaviour from the data is providing valuable insights for the companies operating in customer centric marketing.
In the project, Arvato is helping a Mail-order company—which sells organic products in Germany—in understanding its customer segments in order to identify the next probable customers. The existing customer data and the demographic data of the population in Germany are to be studied to understand different customer segments, and then build a system based on a machine learning model to make predictions on whether a person will be a customer or not based on demographic data.

Dataset Description

There are four data files associated with this project:

- **Udacity_AZDIAS_052018.csv**: Demographics data for the general population of Germany; 891 211 persons (rows) x 366 features (columns).
- **Udacity_CUSTOMERS_052018.csv**: Demographics data for customers of a mail-order company; 191 652 persons (rows) x 369 features (columns).
- **Udacity_MAILOUT_052018_TRAIN.csv**: Demographics data for individuals who were targets of a marketing campaign; 42 982 persons (rows) x 367 (columns).

[1] https://www.bertelsmann.com/divisions/arvato/#st-1

[2] https://www.bertelsmann.com/company/

• **Udacity_MAILOUT_052018_TEST.csv**: Demographics data for individuals who were targets of a marketing campaign; 42 833 persons (rows) x 366 (columns).

Additionally, 2 metadata files have been provided to give attribute information:

• **DIAS Information Levels -** Attributes 2017.xlsx: top-level list of attributes and descriptions, organised by informational category

• **DIAS Attributes -** Values 2017.xlsx: detailed mapping of data values for each feature in alphabetical order.

All the files associated with the project have been provided by Arvato in the context of the Machine Learning Nanodegree Program for analysis and customer segmentation purposes. The four csv files are the demographic data files, in which each row represents demographics of a single person. Each row also includes additional information about their household, building and neighbourhood in addition to their demographics. Customer data has three additional columns indicating their specifics with regard to the mail order company. The Train and Test data have been provided to evaluate supervised learning algorithms.

Problem Statement

"Considering the demographics of the customers, how can the mail-order company acquire new customers?"

Customer's segmentation will be achieved using the demographics dataset of the whole population in Germany and the customers dataset.

The customer's dataset is a subset from the dataset of the whole population. The former will be a guide to the key attributes that classifies customers. Using unsupervised learning, it is possible to cluster the general population dataset and the customer's dataset and determine which cluster has the most number of customers in it.

Second, a supervised learning algorithm can be used to make predictions on whether a person is a probable customer or not, based on the demographic data.

Model Evaluation

As mentioned above, in the problem statement, the project will be divided into two parts:

a. Customer segmentation using “Unsupervised learning”

Using PCA as the dimensionality reduction technique, the explained variance ratio of each feature could be the reference in selecting the number of dimensions for the later steps. The minimum number of dimensions explaining as much variation as possible in the dataset can be chosen in this step.

An unsupervised learning algorithm like K-Means Clustering is proposed and the number of clusters is selected based on the squared error i.e. the distance between all the clusters with the help of an elbow plot.

b. Customer acquisition using “Supervised learning”

The mail-order company wants to target customers directly, and that can be achieved by a supervised learning model. The training data will be split into train and evaluation sets, the model will be trained on the training split and will be evaluated on the evaluation split (typical scenario).

The class label distribution is highly imbalanced; in this particular binary classification problem there are 42,430 observations with label ‘0’ and only 532 observations with label ‘1’, as shown in the figure below. For this problem, we need to be able to tell whether a person will be a future possible customer. AUROC metric which considers both true positive rate and false positive rate seems to be a good choice for this problem, since we want to be able to correctly predict both cases i.e. whether a person becomes a customer or not [3]. For this reason, Area Under Receiver Operating Characteristic (AUROC), has been selected as an evaluation metric. The AUROC gives an idea about overall performance of the model, where the curve is created by plotting True positive rate and False positive rate under different threshold settings. A good performing model will have an AUROC of 1. So higher the AUROC the better the performance of the model.

Kaggle uses AUROC as the evaluation metric as well—on the predictions on the test set.

[3] https://towardsdatascience.com/what-metrics-should-we-use-on-imbalanced-data-set-precision-recall-roc-e2e79252aeba

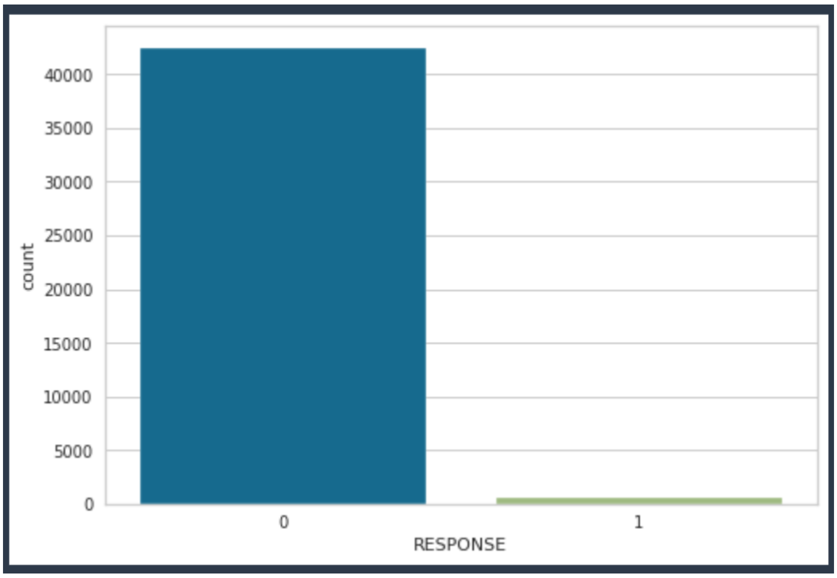


Figure 1. More 0's than 1's (Class Imbalance)

Data Analysis

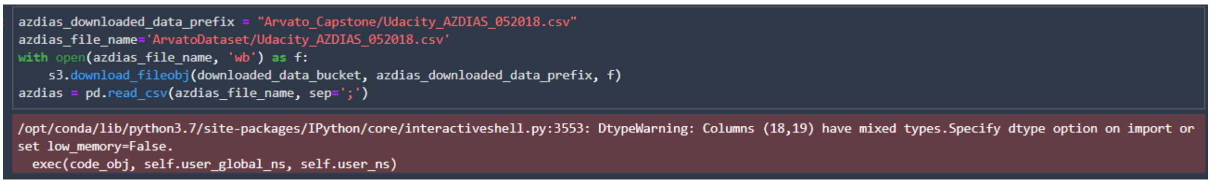
Data Preparation and Preprocessing

Preprocessing is done step by step and each step is done with the help of a helper function written for that specific step. This way, at the end it became easy to join all these functions into a single data preprocessing function, by calling these individual functions inside the main function.

Below are the issues that were faced during the data verification, integrity testing, and preprocessing (Before going into customer segmentation part).

Mixed Type Columns

The warnings that appeared while loading the data were studied.



```
azdias_downloaded_data_prefix = "Arvato_Capstone/Udacity_AZDIAS_052018.csv"
azdias_file_name='ArvatoDataset/Udacity_AZDIAS_052018.csv'
with open(azdias_file_name, 'wb') as f:
    s3.download_fileobj(downloaded_data_bucket, azdias_downloaded_data_prefix, f)
azdias = pd.read_csv(azdias_file_name, sep=';')
```

```
/opt/conda/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3553: DtypeWarning: Columns (18,19) have mixed types.Specify dtype option on import or
set low_memory=False.
  exec(code_obj, self.user_global_ns, self.user_ns)
```

Figure 2. The columns 18 and 19 contained mixed features

The Attribute-values excel sheet was used as a reference to understand what these columns represent and what values can these columns take.

- Addressed columns 'CAMEO_DEUG_2015' and 'CAMEO_INTL_2015'.

- Mis-recorded values – ‘X’, ‘XX’, are replaced with NaN values in the dataframe

Unknown Values

The ‘Attribute-values’ excel sheet contains the information about which columns contain unknown values and how they are entered specified in the dataset. With this information all the unknown values are replaced with NaN values in the dataframes. In total, there were 232 columns which contained unknown representations.

Missing LP_ Values

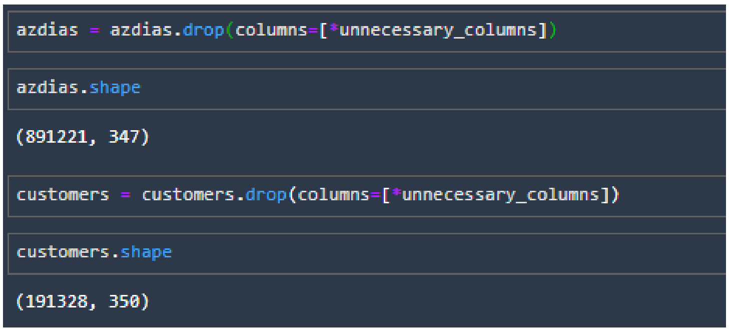
The values in the columns ‘LP_FAMILIE_FEIN’, ‘LP_FAMILIE_GROB’, ‘LP_STATUS_FEIN’, ‘LP_STATUS_GROB’, ‘LP_LEBENSPHASE_FEIN’ and ‘LP_LEBENSPHASE_GROB’ have missing values. These columns give the information about a person’s family status, financial status and the life stage they are in (as provided in the attributes info table).

- These columns contained ‘0’ as a value in the recorded data, which does not correspond to any category specified in the Attribute information data. These ‘0’s have been converted to NaN values.
- The ‘LP_LEBENSPHASE_FEIN’ and ‘LP_LEBENSPHASE_GROB’ have too much granular information packed into them. The FEIN data consisted of fine information about life stage and wealth information. This information has been divided to represent wealth information as one feature and life stage information as one feature and saved into the same two columns.
- The columns ‘LP_FAMILIE_FEIN’ and ‘LP_STATUS_FEIN’ have been dropped since they contained duplicate information that the corresponding ‘_GROB’ columns consisted of.

Missing Values

Column wise

The percentage of missing values in each column is analysed. The columns which had missing values in customers data also seem to have missing data in the general population data and the distribution of the missing data per column is similar between these two. A threshold of 20% was decided after analysing the percentage missing value distribution. The columns that had more than 20% missing values were dropped from both customers data and general population data. A total of 11 columns have been dropped in this step, the columns that have been dropped.



```
azdias = azdias.drop(columns=[*unnecessary_columns])

azdias.shape

(891221, 347)

customers = customers.drop(columns=[*unnecessary_columns])

customers.shape

(191328, 350)
```

Figure 3. Table shapes after dropping unnecessary columns

Row wise

The number of missing values per row is analysed. All the observations with more than 40 missing features are dropped. The resulted dataset sizes are shown below

Feature Encoding

- EINGEFUGT_AM: This column represents the date on which the person has joined or the date the entry was made. This column has been converted to datetime column and only year has been extracted as a feature.
- ANREDE_KZ: This represents the Gender, which was encoded with values 1,2 for male and female, is encoded to contain 0-male and 1-female.
- CAMEO_INTL_2015: This column contained information about the status of a person according to international standards. This column has been divided into two different columns to consist of information about International Family status, International Wealth status.
- WOHNLAGE: This column also has mis recorded values. These values were replaced with NaNs.
- LNR: This column corresponds to an ID given to each person and this feature has been neglected for the analysis.

Imputing

The data still has some missing values. These missing values have been replaced with the most frequently occurred observation in each feature. Since the data corresponds to population in general, imputing the missing values with most frequent observations has been selected.

Scaling Features

A standard scaler is used to bring all the features to the same range. This is done in order to eliminate feature dominance when applying dimensionality reduction.

Unsupervised learning Implementation (Customer Segmentation)

In order to compare the general population and customers (to predict future customers), dividing the general population and the customers into different segments has to be made and that is achieved using unsupervised learning techniques that will be described in this part of the report.

The company's existing customers data was available to understand and compare each feature in the customers data and the general population data. This requires a lot of analysis and this process is time consuming because not all the features will be important in determining the customer behaviour. Also, there might exist some complex interactions between these features which resulted in the person being a customer. A hand coded analysis like this would consume a lot of time resulting in no fruitful results.

An example for that type of analysis is as shown below in Figure 4, the figure describes the distribution of customers (taken from the customers dataset) with regard to the "D19_GESAMT_ANZ_24" attribute.

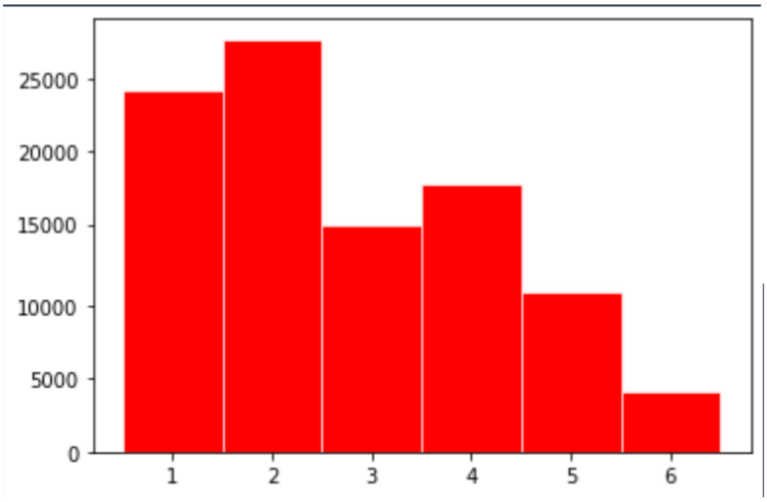


Figure 4. Customers distribution according to transaction activity TOTAL POOL in the last 24 months

Figure 4 shows that most of the customers occupy the values [1, 2, 3, and 4]. Which means (from the attributes value table), that most of the customers tend to have low transactional activity.

But doing that for all the attributes will be cumbersome, right?

So instead, an approach to segment the customers and general population into different parts using unsupervised learning algorithms was chosen. The Principal Component Analysis (PCA) was performed on the given data to reduce the number of dimensions. Since there were 341 features after the data cleaning and feature engineering step, there is a need to understand which features will be able to explain the variance in the dataset. This is done with the help of PCA and the resulting explained variance plot is shown in Figure 5.

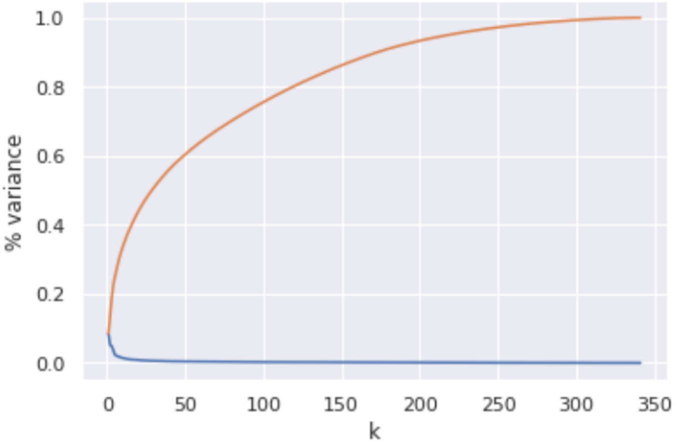
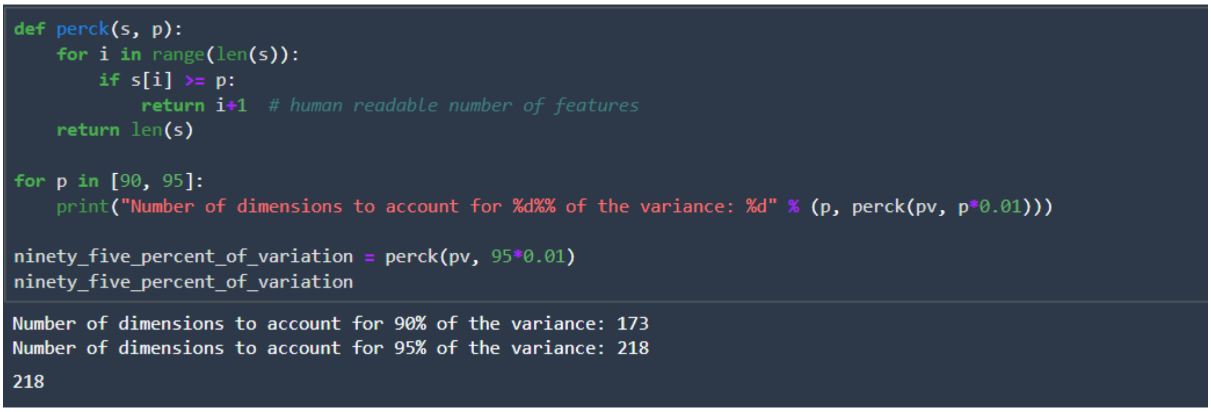


Figure 5. Variation presentation VS k number of features

As seen from the figure, ninety five percent of the variations are represented in about 210 of the features. And to give accurate measures, below is the accurate number of features, which is 218.



```
def perck(s, p):
    for i in range(len(s)):
        if s[i] >= p:
            return i+1  # human readable number of features
    return len(s)

for p in [90, 95]:
    print("Number of dimensions to account for %d%% of the variance: %d" % (p, perck(pv, p*0.01)))

ninety_five_percent_of_variation = perck(pv, 95*0.01)
ninety_five_percent_of_variation
```

```
Number of dimensions to account for 90% of the variance: 173
Number of dimensions to account for 95% of the variance: 218

218
```

Figure 6. Fcn determines the number of dimensions that represents most of the variations

Population Clustering

Next is to divide the general population and customer population into different clusters. K-Means clustering measures the distance between two observations to assign a cluster. This algorithm will help us in separating the general population with the help of the reduced features into a specified number of clusters in a very simple approach. And use this cluster information to understand the similarities in the general population and customer data. The number of clusters is a hyperparameter when working with clustering algorithms. The basic idea behind the clustering algorithms is to select the number of clusters to minimise the intra-cluster variation. Which means the points in one cluster are as close as possible to each other. There is no definitive way of selecting the number of clusters, we can either intuitively select a specific number of clusters or perform an analysis and then select the number of clusters. Here, an elbow plot has been used to decide the number of clusters for the K-Means algorithm. The elbow plot plots the Sum of Squared distances in each cluster for the specified list of number of clusters[4].

This plot helps in understanding how the number of clusters affect the intra-cluster distances. The optimal number of clusters can be the number where the sum of squares of distances starts to plateau. The number of clusters in this case is chosen to be ‘7’, since the sum of squares of distances stops decreasing at a higher rate at this point as shown in Figure 7.

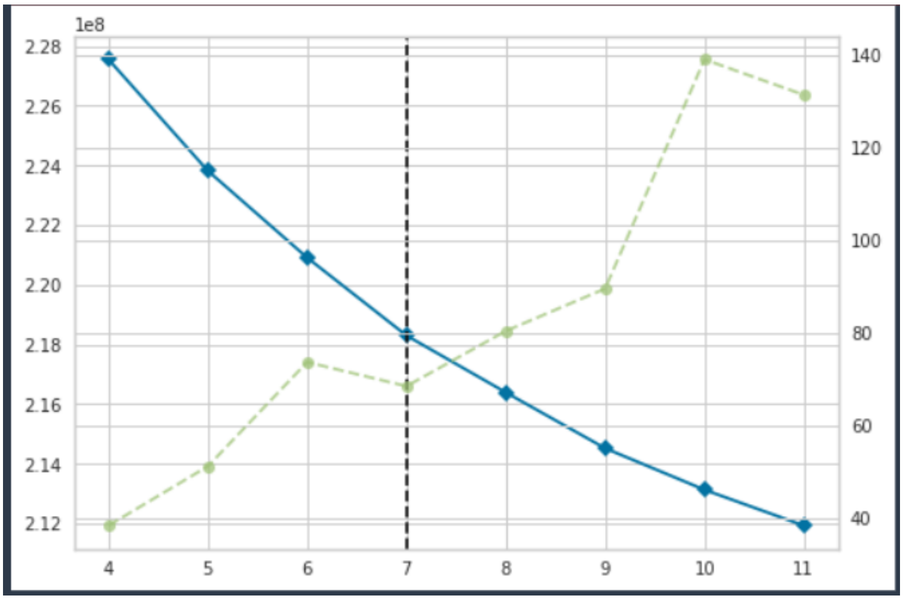


Figure 7. The elbow curve shows the optimal number of clusters

[4] https://www.datanovia.com/en/lessons/determining-the-optimal-number-of-clusters-3-must-know-methods/

Figure 8 represents the proportions of population coming into each cluster. The distribution of the general population is close to uniform (although not perfectly uniform), meaning that the general population has been uniformly clustered into 7 segments. But the customer population seems to be coming from the clusters 2, 4, 5, 6. We can further confirm this by taking the ratio of proportions of customers segments and general population segments as shown in Figure 9.

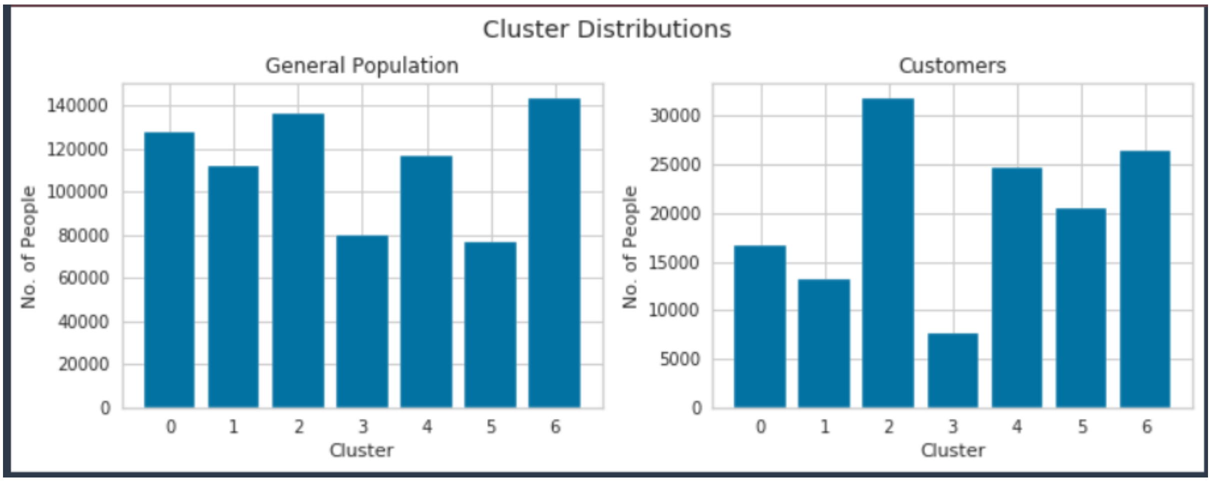


Figure 8. Cluster proportions

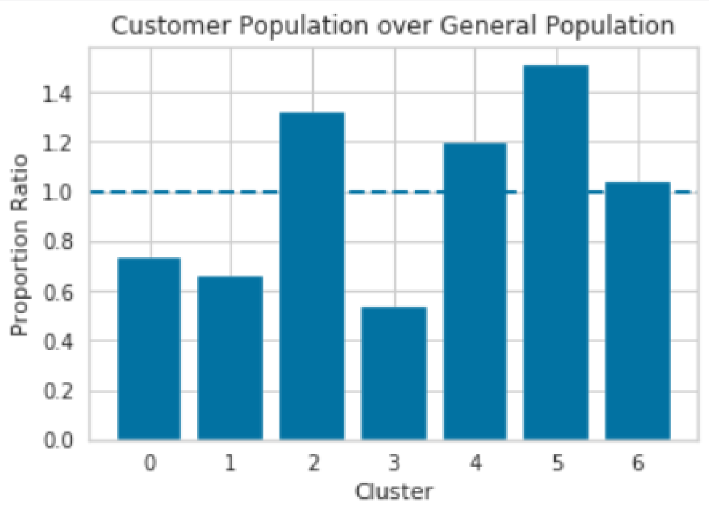
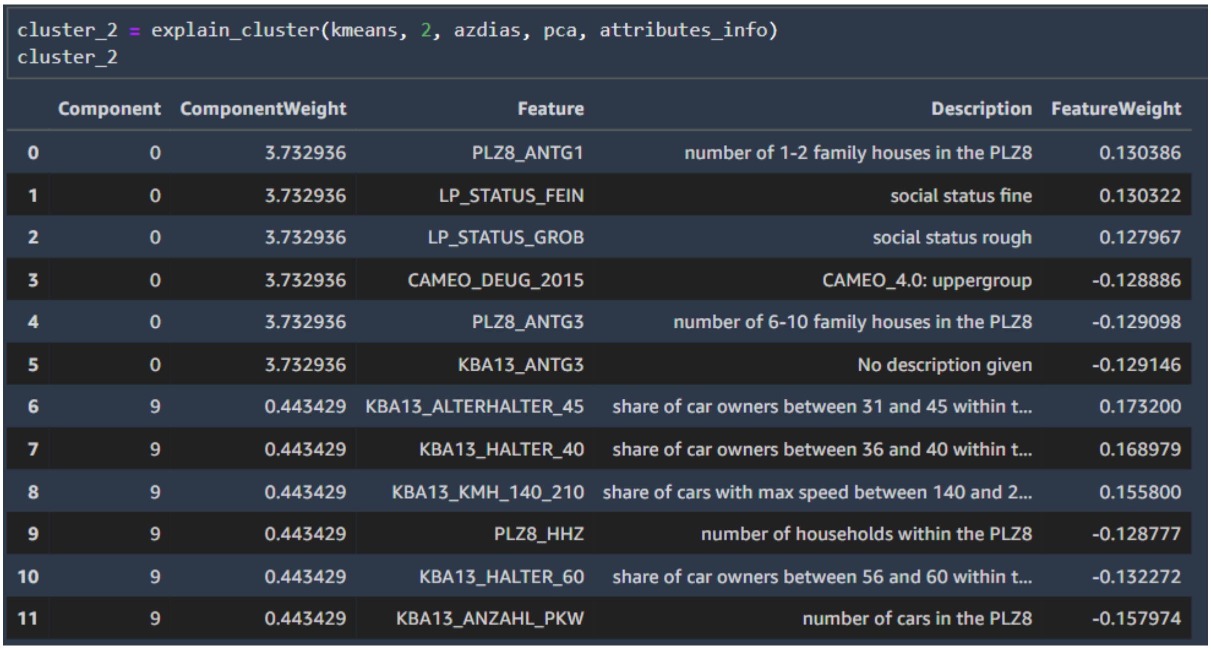


Figure 9. Cluster Proportion ratio

If the ratio of proportions is greater than 1 that means this cluster has a greater number of customers in the existing population and has a potential to have more future customers. Whereas if the ratio is less than 1 that means these clusters have the least possibility to have future customers.

Similar to what we have done with PCA components, we can understand each cluster by analysing what components make up each cluster and what main features make up these components. An example is shown in Figure 10.



```
cluster_2 = explain_cluster(kmeans, 2, azdias, pca, attributes_info)
cluster_2
```

	Component	ComponentWeight	Feature	Description	FeatureWeight
0	0	3.732936	PLZ8_ANTG1	number of 1-2 family houses in the PLZ8	0.130386
1	0	3.732936	LP_STATUS_FEIN	social status fine	0.130322
2	0	3.732936	LP_STATUS_GROB	social status rough	0.127967
3	0	3.732936	CAMEO_DEUG_2015	CAMEO_4.0: uppergroup	-0.128886
4	0	3.732936	PLZ8_ANTG3	number of 6-10 family houses in the PLZ8	-0.129098
5	0	3.732936	KBA13_ANTG3	No description given	-0.129146
6	9	0.443429	KBA13_ALTERHALTER_45	share of car owners between 31 and 45 within t...	0.173200
7	9	0.443429	KBA13_HALTER_40	share of car owners between 36 and 40 within t...	0.168979
8	9	0.443429	KBA13_KMH_140_210	share of cars with max speed between 140 and 2...	0.155800
9	9	0.443429	PLZ8_HHZ	number of households within the PLZ8	-0.128777
10	9	0.443429	KBA13_HALTER_60	share of car owners between 56 and 60 within t...	-0.132272
11	9	0.443429	KBA13_ANZAHL_PKW	number of cars in the PLZ8	-0.157974

Figure 10. Cluster #2 - Principle components and component features

The two components that make up this cluster '0' and '9'. That means this cluster corresponds to people who like to live in neighbourhoods having a smaller number of houses and the houses with a smaller number of families, which can be seen from the feature weights given to the corresponding feature in each component. Also, these people tend to live in neighbourhoods which have 31-40 car owners (seen from component 9 feature weights) and like to live where there are a smaller number of cars (seen from last element in the table).

Supervised Learning Implementation (Customer Acquisition)

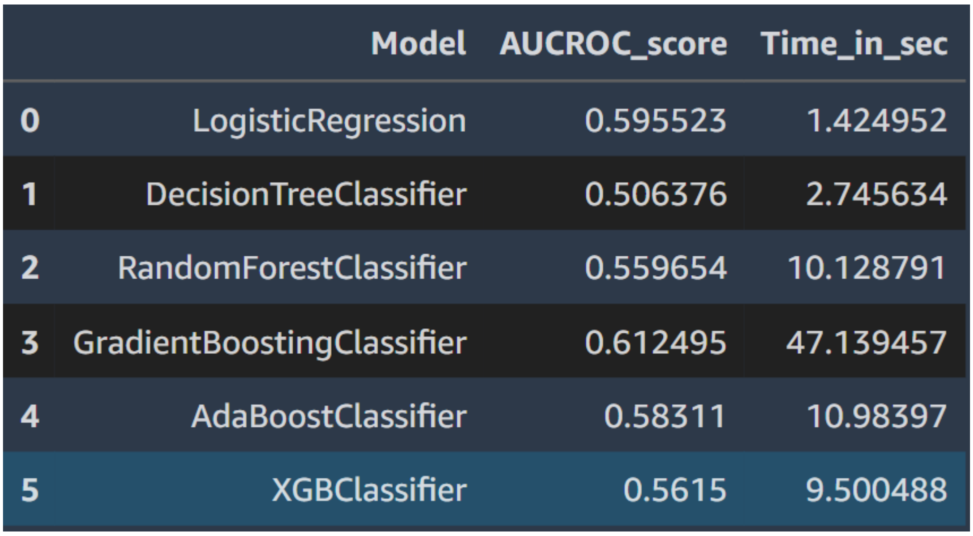
Supervised learning is used to predict whether a person will be a customer or not based on the demographic data. The file 'Udacity_MAILOUT_052018_TRAIN.csv' is provided with the same features as the general population and customers demographic data. An extra column 'RESPONSE' has been provided with this data. The response column indicates whether this person was a customer or not. This data has been cleaned by following similar cleaning and processing steps that were followed for general population and customer datasets.

Benchmark

Comparing the results from future steps in order to evaluate the used models. The data is split into train and validation splits and a logistic regression model was trained on unscaled training data and evaluated on the unscaled validation data.
Benchmarking the logistic regression model resulted with a score - (0.647 AUROC Score).

More models Baseline performance

After setting the benchmark, the data has been scaled with the standard scaler and is split into training and validation split. Different algorithms have been trained on the training split and have been evaluated on the validation split. The algorithms that have been selected with their AUCROC Score and time elapse for training and prediction:



	Model	AUCROC_score	Time_in_sec
0	LogisticRegression	0.595523	1.424952
1	DecisionTreeClassifier	0.506376	2.745634
2	RandomForestClassifier	0.559654	10.128791
3	GradientBoostingClassifier	0.612495	47.139457
4	AdaBoostClassifier	0.58311	10.98397
5	XGBClassifier	0.5615	9.500488

Figure 11. Supervised learning models performance comparison