Capstone Project: Machine Learning Engineer Nanodegree

Customer Segmentation Report for Arvato Financial Solutions

Submitted By:

JRA Supuli Jayaweera

on 7th August 2020

Table of Contents

1.0 Definition

1.1 Project Overview

1.1.1 Problem Domain and Origin

Arvato Financial Services, a mailorder company in Germany, is interested in analyzing the demographics data of its customers against the general population to identify which segments in the general population could be potential targets for their marketing plan. For this task, Arvato has provided demographics data of the general population at large as well as the data of the previous customers.
In addition to the modelling and identifying this customer base, the company is also interested to identify individuals who will respond favourably to the marketing campaign and become an Arvato customer.

1.1.2 Datasets and Inputs

This project provides four main datasets with identical demographics features.

1) ***Udacity_AZDIAS_052018.csv:*** Demographics data for the general population of Germany; 891 211 persons (rows) x 366 features (columns).
2) ***Udacity_CUSTOMERS_052018.csv****:* Demographics data for customers of a mail-order company; 191 652 persons (rows) x 369 features (columns). The extra 3 columns are for CUSTOMER_GROUP', 'ONLINE_PURCHASE', and 'PRODUCT_GROUP', which provide broad information about the customers.
3) ***Udacity_MAILOUT_052018_TRAIN.csv:*** Demographics data for individuals who were targets of a marketing campaign; 42 982 persons (rows) x 367 (columns).
4) ***Udacity_MAILOUT_052018_TEST.csv:*** Demographics data for individuals who were targets of a marketing campaign; 42 833 persons (rows) x 366 (columns).

In addition to the above four data files, two additional files were provided with details description of the attributes.

i) ***DIAS Information Levels - Attributes 2017.xlsx:*** A top-level list of attributes and descriptions, organized by informational category.
ii) ***DIAS Attributes - Values 2017.xlsx:*** A detailed mapping of data values for each feature in alphabetical order.

1.2 Problem Statement

The problem statement of the project can be divided into four main sections.

Section1: In this section, data preprocessing is done including data cleaning, reformatting data types and filling the missing data. Further, this section will decide which features to keep or drop and which features needs regrouping.

Section2: Identify the demographics features that describe and differentiate the company's existing customer base from the general population. This will be useful to explain which part of the general population is more likely to become the company's customers base.

Section3: In this section, develop a model to identify which individuals will become an actual customer for this company after the marketing campaign. For this task, supervised classification models like the random forest, decision trees and boosting models are considered. Once a successful model is developed, used it on a test dataset to predict the outcome of a selected individual being a customer.

Section4: In the final section, the predictions made on the test dataset will be submitted to the Kaggle competition to evaluate the performance.

1.3 Solution Statement

To achieve the goals of the project the following solution statement is proposed.

1) The data preprocessing and cleaning will be done using exploratory data analysis and the use of support documentation provided on attribute description.
2) After a dimensionality reduction of the features (using PCA), unsupervised learning techniques like k-means clustering will be used to perform customer segmentation. This will be used to identify which demographic features best describe the core customer base for the company.
3) Then, supervised classification algorithms will be used to develop a model to identify which individuals will be an actual customer after the marketing campaign. Decision trees, SVM, random forest, and boosting methods (AdaBoost and XGBoost) are selected as the initial classification algorithms. The model selection and tuning will be done against the training data provided. The model is used again on the test dataset to get predictions. The model evaluation is based on the area under the ROC curve which will be described in detail in this proposal later.

1.4 Metrics

1.4.1 Metric for unsupervised learning for customer segmentation

The customer segmentation was done using a k-means clustering method. Principle Component Analysis (PCA) was used to reduce the dimensionality of the original dataset and then k-means clustering method was used to cluster the population and customers. The optimum number of clusters were found using the elbow plot which calculates the sum of squared distances from each point to its assigned centre.

1.4.2 Metric for supervised learning for customer selection

The dataset provided for supervised learning is highly imbalanced and metric like accuracy cannot evaluate the performance of the model effectively. Thus, for this project, area under the receiver operating characteristic (AUC-ROC) curve is selected as the evaluation criteria. Furthermore, it is also the ranking criteria according to the Kaggle competition guidelines [1].

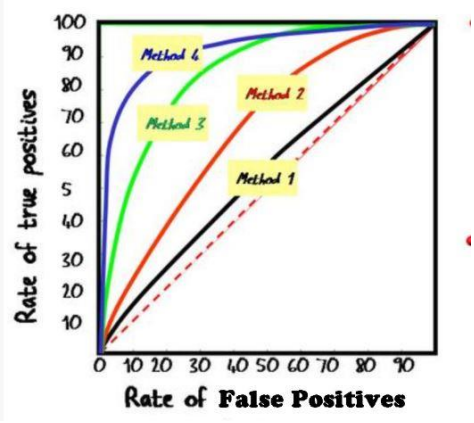

Figure1: Example of a ROC curve. [2]

As seen in Figure1, ROC is a graph used to plot the true positive rate (TPR, the proportion of actual customers that are labelled as so) against the false positive rate (FPR, the proportion of non-customers labelled as customers) at different threshold values. The TPR is referred to as the sensitivity or the recall while the FPR is also known as specificity. In an ideal situation, we want the TPR to be 1 and FPR to be 0. However, in a more practical setting, a trade off from these values are achieved to get the optimum threshold value. Model with the highest area under the roc (AUC-ROC) curve is selected as the best model for the classification.

2.0 Analysis and Methodology

2.1 Exploratory Data Analysis and Preprocessing

The original dataset for the population contained 891221 rows with 366 features while the customers dataset contained 191652 rows with 369 features. Three additional features on the customers correspond to 'customer_group', 'online_purchase' and 'product_group'. This section provides a detailed analysis of the data and the preprocessing steps carried out on the dataset before feeding it to the model.

2.1.1 Features with unknown value codes:

According to the additional information file, '***DIAS Attributes - Values 2017.xlsx'***, values/ranks are given for different categories of each feature. However, 233 features contained an unknown category represented by different values ('***Null_Values_DIAS_2'*** data file contains the summary of these features with its unknown value codes). These values and features were identified and then replaced with null values in both datasets.

2.1.2 Missing values:

Next, missing value composition of each feature was analysed and plotted as shown in Figure 2. For simplicity purposes, only the features with missing value percentage over 30% is considered for the plot. Features with more than 40% missing percentage (indicated by the red line) are selected as the cutoff point. Both datasets contained the same set of features that have more than 40% missing data and these 9 features were dropped from the dataset.

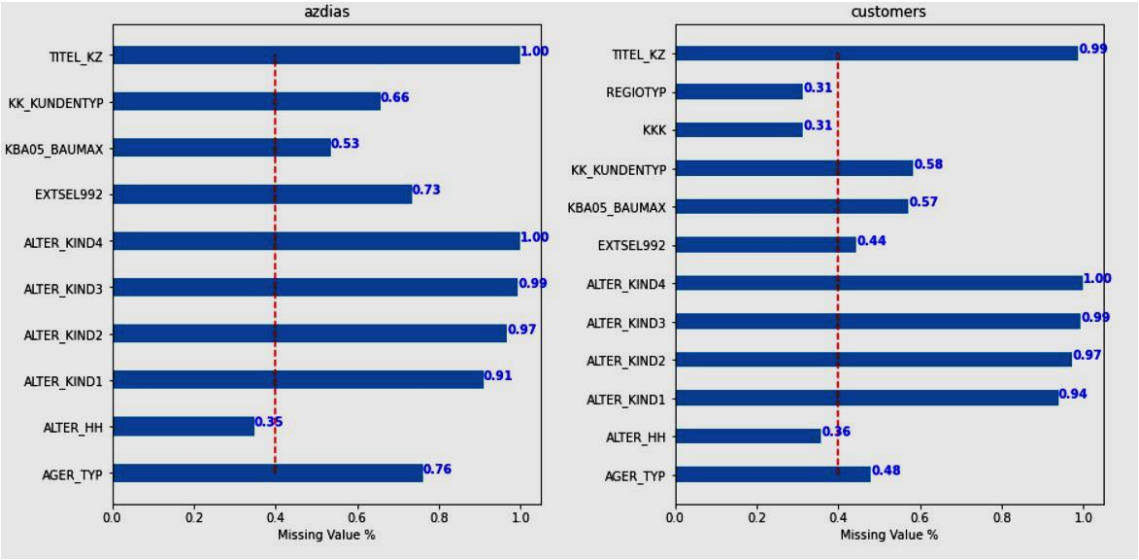

Figure2: Missing value percentage of features

In addition to the rows, missing data percentage is also evaluated in row wise (Figure3), and based on the values, a threshold value of 250 is selected. Any rows that do not have 250 values were dropped from the dataset. Overall, 105801 rows were dropped form the azdias and 51281 rows were dropped from customer datasets, respectively.

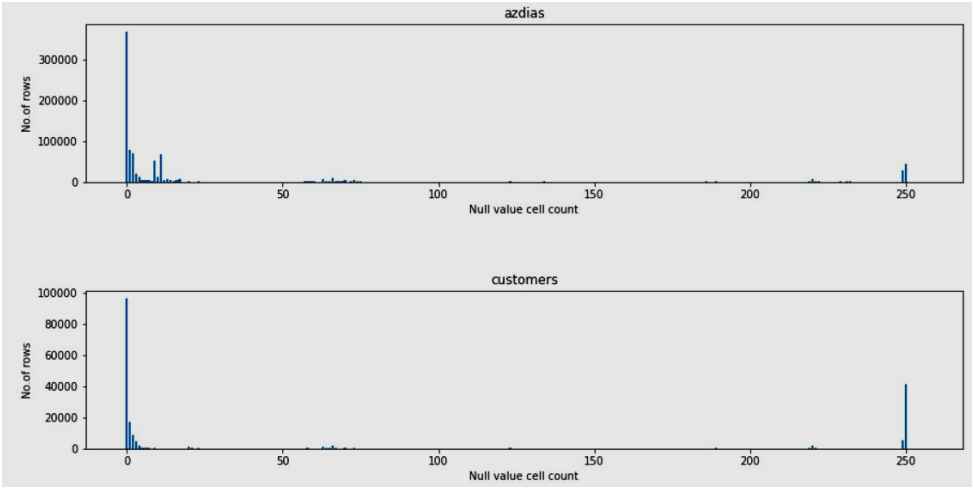

Figure3: Missing value percentage of rows

2.1.3 Features with objective data type:

Objective data include strings and mixed data. Upon inspection, the following 6 features had objective data type.

- ‘CAMEO_DEU_2015': This contained strings. (e.g. 1A, 2B , XX)
- 'CAMEO_DEUG_2015': This contained mixed data. (e.g. 6.0, 6 , X)
- 'CAMEO_INTL_2015': This contained mixed data. (e.g. 54.0, 54 , XX)

For above CAMEO_ features, the first characters of the string was extracted and mapped into integers. The X and XX were converted to NaNs.

- 'EINGEFUEGT_AM': This contained strings of a date and time (e.g. 1992-02-10 00:00:00). The first 4 characters (year) were extracted and converted to an int dtype.
- 'OST_WEST_KZ': This contains the string ‘W’ or ‘O’, which were mapped into integers.
- 'D19_LETZTER_KAUF_BRANCHE': Although no explicit explanation is provided about this feature by looking at the values, this might contain the group of the last transactional activity group. Since detail transactional activity of individuals is provided in other columns, it is decided to drop this.

2.1.4 Categorical features with high cardinality:

Most of the features are categorical type. But there were some features which had high cardinality. Figure4 shows the features with more than 15 unique values and its distribution.

Note:

- ‘LNR’ which corresponds to the ID of each person is dropped from both datasets.
- 'LP_LEBENSPHASE_FEIN' is the more detailed breakdown of 'LP_LEBENSPHASE_GROB'. Considering 'LP_LEBENSPHASE_FEIN' has over 40 categories, this is dropped from the df.
- 'GEBURTSJAHR' is the year of birth. But it contains value 0 which is not valid data. Thus, the 0 is converted to a Nan before plotting the distribution.

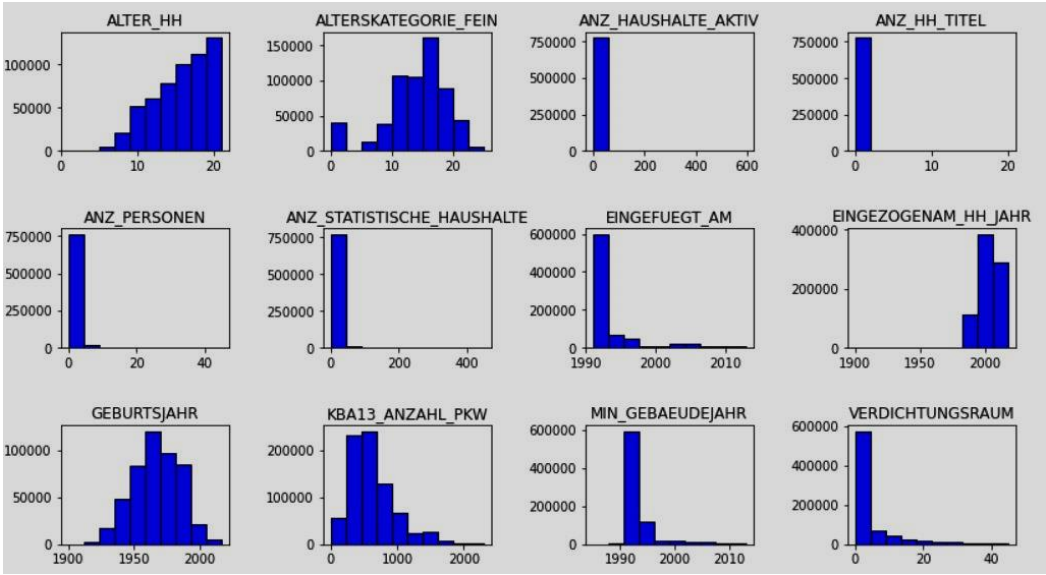

Figure4: Distribution plot of the high cardinal categorical features

From the distribution graphs, we can see ANZ_HAUSHALTE_AKTIV','ANZ_HH_TITEL', 'ANZ_STATISTISCHE_HAUSHALTE', 'VERDICHTUNGSRAUM' 'ANZ_PERSONEN', 'EINGEFUEGT_AM', and 'MIN_GEBAEUDEJAHR' features are highly skewed. Also, 'EINGEFUEGT_AM','EINGEZOGENAM_HH_JAHR','MIN_GEBAEUDEJAHR' and 'GEBURTSJAHR' have year values for data.

All these features are reconstrued to categorical bins of levels <=10. The bin sizes/limits are decided by the azdias data. Once the bins are selected customers data is transformed according to those limits. For non-skewed data, bin limits are deiced based on the quantile method.

2.1.5 Imputing missing values:
The final step in data preprocessing is to impute the missing value with the most frequent value for each feature. For each feature, frequent value is selected by the azdias dataset since it represents the population at large and that value is used to fill the customers missing data as well. Most frequent value for each feature was selected from the azdias dataset and these values were used to impute the missing values of azdias and customers dataset.

2.1.5 Feature scaling:
To scale the features MinMax scaler is used. The scaler parameters were defined using the azdias and then customers dataset was transformed accordingly.

2.2 Methodology

2.2.1 Algorithms and Techniques
Algorithms and Technique used in this project are discussed under the following two segments- customer segmentation and customer identification.

2.2.1.1 Customer Segmentation
In this section, unsupervised learning (clustering algorithms) methods are used to describe the relationship between the demographics of the company's existing customers and the general population of Germany.

- PCA:

Considering the cleaned and preprocessed data has 354 features, the first step is to apply principal component analysis (PCA). Applying PCA can improve the performance of the clustering algorithms as well as reduce the noise. Figure5 shows the cumulative explained variance ratio of the principal components of the azdias dataset. Based on that, we can see over 90% of the total variance is covered by the top 170 components. Thus, we will transform both datasets into these top 170 components.

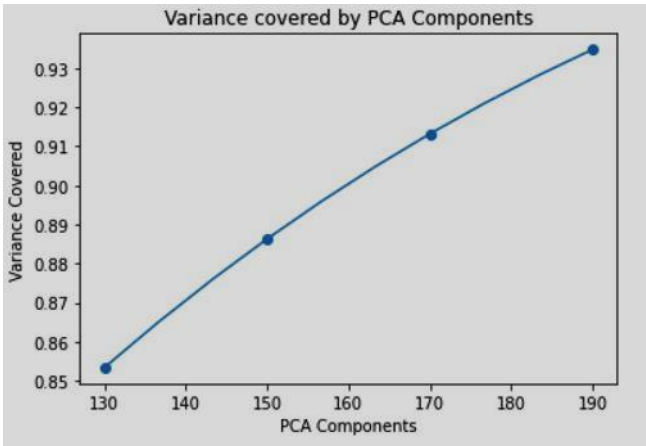

Figure5: Cumulative explained variance ratio plot of PCA

Figure6 shows the weight distribution of the highest positive and negative contributing features for the top 4 PCA components.

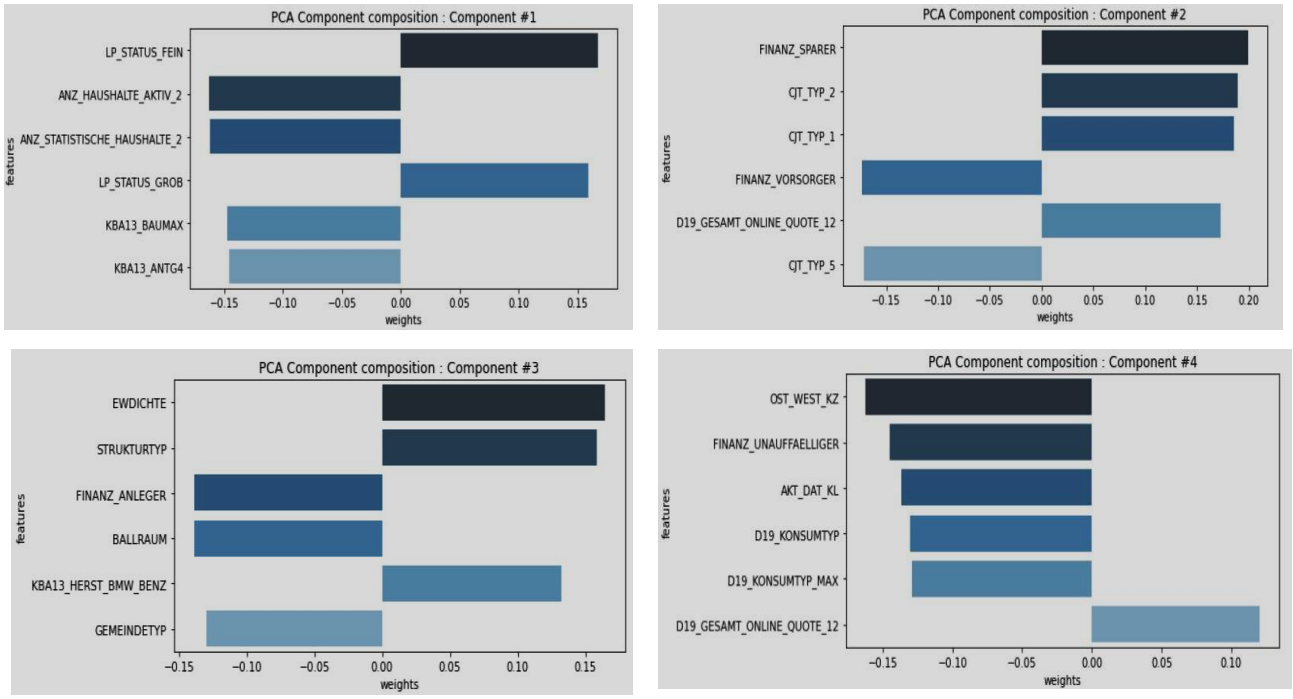

Figure6: Analysis of PCA component composition

PCA Component 1 has a high positive weight for LP_STATUS_FEIN (LP_STATUS_GROB is the rough estimate of the same feature) which represents the social status of an individual.
The largest negative weights are for ANZ_HAUSHALTE_AKTIV, KBA13_BAUMAX and KBA13_ANTG4. Although the exact explanation is not provided for all features, based on other similar naming variables it can be said that 'HAUSHALTE' features represent the number of household details and 'KBA13' features represent the share of cars.

PCA Component 2 has a high positive weight for FINANZ_SPARER(financial typology: money saver), CJT_TYP_2 and CJT_TYP_1 (Customer-Journey-Typology relating to the preferred information and buying channels of consumers) and D19_GESAMT_ONLINE_QUOTE_12 which represents and amount of online transactions within all transactions respectively.
The largest negative weights are for FINANZ_VORSORGER (financial typology: be prepared) and CJT_TYP_5.

PCA Component 3 has a high positive weight for EWDICHTE (density of inhabitants per square kilometer), STRUKTURTYP and KBA13_HERST_BMW_BENZ (share of BMW & Mercedes Benz within the PLZ8). The largest negative weights are for FINANZ_ANLEGER (financial typology: investor), BALLRAUM (distance to the next metropole) and GEMEINDTYP.

PCA Component 4 has a high positive weight for D19_GESAMT_ONLINE_QUOTE_12. The largest negative weights are for OST_WEST_KZ (flag indicating the former GDR/FRG), FINANZ_UNAUFFAELLIGER (financial typology: unremarkable), AKT_DAT_KL, D19_KONSUMTYP (consumption type) and D19_KONSUMTYP_MAX.

- Cluster size:

Too high clusters points will fail to capture the patterns of clusters and overfit. To select the optimum cluster size elbow method is used. This is a common heuristic mathematical optimization method in which it calculates the sum of squared distances from each point to its assigned centre [3]. The elbow act as a cutoff point where the additional clusters will not significantly reduce the cost. Figure5 shows the elbow plot the PCA transformed dataset. Based on the graph, cluster size of 7 is selected.

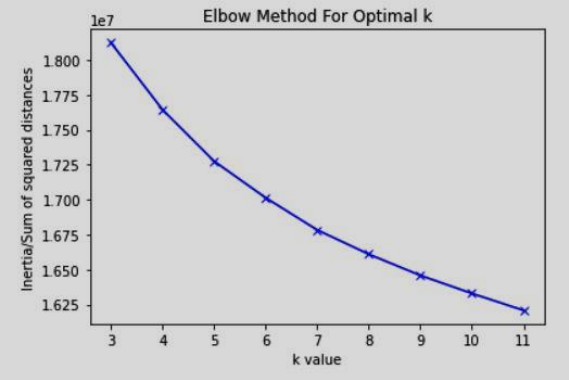

Figure7: Elbow Plot

- Clustering:

Once the cluster size is decided, k-means clustering algorithm is used to cluster the population and customers. K-means is a relatively easy to implement method for this task, and it can scale up to large dataset. Figure8a shows the distribution of people in each cluster while Figure8b shows the ratio of customers/general population in each cluster.

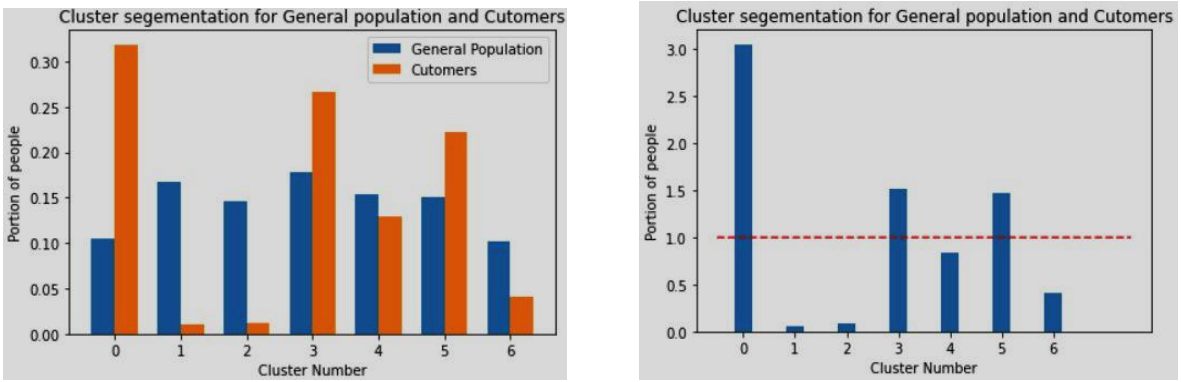

Figure8: a) No. of people in each cluster, b) ratio of customers/population in each cluster

From the Figure8a graph, it can be seen, the general population is approximately evenly distributed among the 7 clusters. From these clusters, customers are mostly located in cluster 0, 3 and 5 while cluster 1,2 and 6 has the lowest number of customer representation. From the ratio of customers to general population provides (Figure8b) an insight into which cluster of the population should be targeted by Arvato. If this ratio is more than 1 (clusters 0,3 and 5), it indicates that there is a higher portion of customers present in those clusters and most importantly higher probability that people in that cluster will become future customers.

Next, the top 5 PCA components for each cluster was analysed.

- Top 5 PCA components for the cluster 0 are: [3, 1, 2, 4, 6]
- Top 5 PCA components for the cluster 1 are: [4, 3, 1, 2, 11]
- Top 5 PCA components for the cluster 2 are: [1, 2, 8, 7, 5]
- Top 5 PCA components for the cluster 3 are: [1, 2, 4, 6, 11]
- Top 5 PCA components for the cluster 4 are: [1, 2, 4, 3, 11]
- Top 5 PCA components for the cluster 5 are: [2, 1, 3, 11, 8]
- Top 5 PCA components for the cluster 6 are: [2, 4, 3, 1, 10]

Almost all the top PCA components for each cluster falls in the range 1-4. These were explained in detail under PCA component analysis. Further, PCA component 11 is in the top 5 in 4 of the clusters (cluster 1,3,4,5). More PCA components have been analysed in the notebook.

2.2.1.2 Customer Identification

In this section, a classification model is built to identify whether a given person will respond favourably to the mailout campaign. Two datasets were provided for this. The training dataset has the same features as the customers with an additional column for responses. The final goal is to use the trained classification model on the test dataset to predict the response.
Upon inspection of bost the training and test data, it was observed both datasets can be preprocessed in the same way as the above mentioned azdias dataset. Once the dataset is cleaned, supervised classification algorithms like the random forest, Decision tree and Boosting methods were used to select the best performing model.

2.2.2 Benchmark

Due to the nature of the data provided, a random forest (RF) model is selected as the benchmark model as it has high accuracy for classification among current algorithms and runs efficiently on large databases [4]. The performance of each model is evaluated based on the AUC-ROC score. The training data was split into training and validation and the **benchmark score of 0.5881** was obtained using the RF model on the unscaled features.

2.2.3 Supervised learning

With the benchmark score, the next step is to use selected supervised classification models like random forest, XGBoost, Decision tree, AdaBoost and GradientBoosting to evaluate the performance. These models were applied to the scaled dataset. Table1 shows the performance of each model.

Table1: Performance of the classification models

Model	Score (AUC-ROC)	Execution Time (s)
RandomForest	0.5869	8.2988
GradientBoosting	0.7382	47.4300
AdaBoost	0.7152	11.0260
DecisionTree	0.5105	2.1959
XGBoost	**0.7405**	**19.0304**

Based on the above performance scores, XGB classifier has shown the best accuracy with 0.74. Time duration for XGB is reasonable and much less than the Gradientboosting. Although GradientBoosting has an accuracy slightly less than XGB, the execution time is too long (2.5 times XGboost). RF and Decision tree are both very fast, but the accuracy is too low (<60%).

Thus, XGBoost method is selected for this classification and it performed well compared to the benchmark model even without any refinement.

2.2.3 Refinement

To further improve the XGBoost model, hyperparameters were trained using GridSearchCV. The tested and selected hyperparameters are shown in Figure9.

Figure9: Tested and selected hyperparameters for XGBoost model

The accuracy on the validation dataset was 0.7386 which is approximately similar to the original XGBoost model. However, the test score using the hyperparameter tuned model was significantly better as explained in the next section.

3.0 Results

3.1 Model Evaluation

The selected model for this classification task is the hyperparameters tuned XGBoost model. To evaluate the performance of the model, the test dataset was preprocessed in the same was the train data/azdias and the model was applied. Figure10 shows the AUC-ROC accuracy obtained for the model.

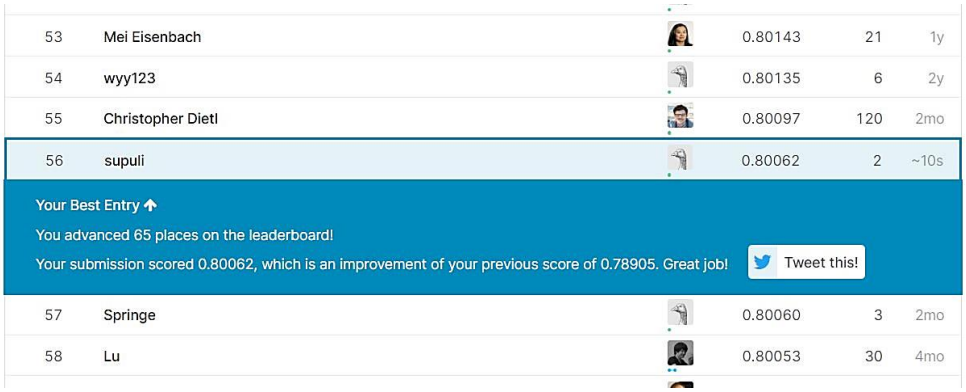

Figure10: Performance of the model on the Kaggle leaderboard

Table2 summarizes the selected model parameters and their performance. Based on the performance values, it is evident that the tuned XBGbost model is significantly better than the benchmark model. Considering the data imbalance and the high volumes of missing data it is challenging to get high accuracy scores. Based on the Kaggle leaderboard the highest score

achieved so far is 0.8106. Compared to that value, the score obtained in this model (0.8006) is justifiable.

Table2: Parameters and Performance of the benchmark and XGBoost model

	Benchmark Model-RandomForest	XGBoost Model-Baseline	XGBoost Model-Tuned
Model Parameters	criterion='gini', max_depth=None, max_features='auto', max_leaf_nodes=None, max_samples=None, bootstrap=True, min_samples_leaf=1, min_samples_split=2, n_estimators=100, oob_score=False,	booster='gbtree', gamma=0, learning_rate=0.1, max_delta_step=0, max_depth=3, min_child_weight=1, n_estimators=100, objective='binary:logistic'	booster='gbtree', eta=0.01 gamma=0.1, learning_rate=0.1, max_delta_step=0, max_depth=3, min_child_weight=1, n_estimators=100, objective='binary:logistic'
Validation Score	0.5881	0.7405	0.7386
Test Score	-	0.78905	**0.80062**

3.2 Further Testing

Suggestion to further test the model are as follows:

1) Adding more meaningful features and through feature selection
2) Handle the imbalance nature of the dataset through techniques like under-sampling and over-sampling.
3) Trying out new models like neural networks and bagging methods.

4.0 References

[1] Udacity+Arvato: Identify Customer Segments. [accessed 2020 Jul 11]
https://www.kaggle.com/c/udacity-arvato-identify-customers/overview/evaluation

[2] Sparks Machine Learning Library Tutorial, Binary Classification [accesses 2020 Jul 26]
http://web.cs.ucla.edu/~mtgarip/linear.html

[3] Wikipedia, Elbow method (clustering) [accesses 2020 Jul 20]
https://en.wikipedia.org/wiki/Elbow_method_(clustering)

[4] Breiman L, Cutler A. Random Forests. [accessed 2020 Jul 11]
https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm