Machine Learning Engineer Nanodegree

Capstone Project Report

Customer Segmentation – Arvato Financial Solutions

KUSHAGRA AGRAWAL

July 06, 2020

Contents

Definition 3

Project Overview 3

Problem Statement 4

Evaluation Metrics 4

Analysis 5

Data Exploration and Preprocessing 5

Algorithms and Methodology 8

Results 10

Definition

Project Overview

Domain Background

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. Also, Arvato is wholly owned by Bertelsmann, which is a media, services and education company.

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 behavior from the data is providing valuable insights for the companies operating in customer centric marketing. Data Science and Machine Learning are immensely used now a days to fulfil business goals and to satisfy customers.

In this project, Arvato is helping a Mail-order company, which sells organic products in Germany, to understand its customers segments in order to identify next probable customers. The existing customer data and the demographic data of population in Germany are to be studied to understand different customer segments, and then building a system to make predictions on whether a person will be a customer or not based on the demographic data.

Dataset and Inputs

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).
- 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, organized 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 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 neighborhood in addition to their demographics. Customers data has three additional columns indicating their specifics about the mail order company. The Train and Test data have been provided to evaluate supervised learning algorithms.

Problem Statement

The problem statement can be formulated as, “Given the demographic data of a person, how can a mail order company acquire new customers in an efficient way”.
First, the demographic data of the general population and the customers is be studied with the help of unsupervised learning algorithms. The goal in this step is to identify segments in general population and segments in the existing customers, and then discovering what demographic features correspond to a person being a customer for the mail-order company.
Second, a supervised learning algorithm is be used to make predictions on whether a person is a probable customer or not, based on the demographic data.

I have used principal component analysis (PCA) technique for dimensionality reduction. Then, elbow curve will be used to identify the best number of clusters for KMeans algorithm. Finally, I will apply KMeans to make segmentation of population and customers and determine description of target cluster for the company.

Evaluation Metrics

The project is divided into two parts:

Customer Segmentation using Unsupervised Learning Algorithms

This part of the project uses a dimensionality reduction technique PCA to reduce the number of dimensions. 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. Also, in case of segmenting the customers into different clusters, an unsupervised learning algorithm like K-Means Clustering is proposed. Also, in this case the number of clusters is selected on the squared error i.e. the distance between all the clusters with the help of an elbow plot.

Customer Segmentation using Supervised Learning Algorithms

In the second part of the project, the task is to predict whether the mail-order company should approach a customer. Here the given 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. In this step evaluation metrics for classification can be used.
The class label distribution is highly imbalanced, in this binary classification problem there are 42,430 observations with label ‘0’ and only 532 observations with label ‘1’, as shown in Figure 1. 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 seem 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. Since, both these predictions are important for us.
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 better the performance of the model.

Also, the Kaggle competition page uses AUROC as the evaluation metric on the predictions on the test set.

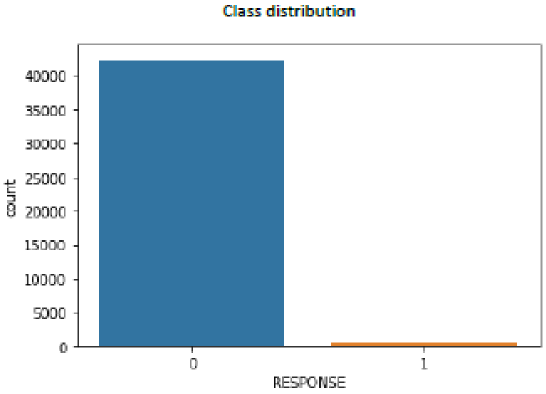


Fig 1: Class Imbalance

Analysis

Data Exploration and Preprocessing

The datasets given were loaded and checked for integrity, to contain the expected number of rows and columns as per description. The 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 single data preprocessing function, by calling these individual functions inside the main function.

1. **Addressing mixed type columns**
   The warnings that came while loading the data are studied. The columns 18 and 19 have mixed features and some mis recorded values. The columns to be addressed are 'CAMEO_DEUG_2015 and 'CAMEO_INTL_2015'. Mis recorded values 'X' and 'XX' are replaced with NaN values in the data frame.
2. **Addressing 'unknown' values**
   Next step is to fix all the unknown values in the data frame. All the unknown values are replaced with NaN values. In total, there were 232 columns which had unknown values.
3. **Checking for common features**
   General Population data and Customers data are checked for common features.
   - 272 features are common between general population and customers data for which clear description is provided.
   - 3 features are only present in customers data.
   - 42 features have no description in the metadata.
4. **Addressing non-existent values in 'LP_*' columns:**
   Another problem with the given data lies in the values in the columns 'LP_FAMILIE_FEIN', 'LP_FAMILIE_GROB', 'LP_STATUS_FEIN', 'LP_STATUS_GROB', 'LP_LEBENSPHASE_FEIN' and

'LP_LEBENSPHASE_GROB'. These columns give the information about a person's family status, financial status and the life stage they are in.

- These columns have '0' as a value in the recorded data which does not relate to any category specified. These '0's has been converted to NaN values.
- There is too much information in the 'LP_LEBENSPHASE_FEIN' and 'LP_LEBENSPHASE_GROB' columns. The FEIN data have information about life stage and wealth information. The wealth information and life stage information are divided into two separate columns and saved.
- The columns 'LP_FAMILIE_FEIN' and 'LP_STATUS_FEIN' have been dropped since they have duplicate information that the corresponding '_GROB' columns consisted.

5. Re-encoding features

The below specified features have been re-engineered.

- WOHNLAGE: This column has mis-recorded values. All the mis-recorded values have been replaced with NaN.
- CAMEO_INTL_2015: This column contains information about the status of a person according to international standards. This column has been divided into two different columns to consist information about International Family status, International Wealth status.
- ANREDE_KZ: This column had 1,2 for male and female. It is re encoded to 0 for male and 1 for female.
- EINGEFUGT_AM: This column has the date on which the person has joined. This column has been converted to datetime data type and only year has been used as a feature.
- LNR: This column represents an ID given to each person and this feature is not used for analysis.

6. Missing Values

Next step is to study the missing values column wise and row wise.

- Row wise: The number of missing values per row is analyzed in this step. All the rows which have more than 50 missing features are dropped in this step. This resulted in dropping a total of 1,53,933 observations from general population data which originally contained 8,91,211 observations. And a total of 57,406 observations were dropped from customers data which originally contained 1,91,652 observations.
- Column wise: The percentage of missing values in each column is analyzed. A threshold of 30% was decided after analyzing the percentage missing value distribution. The columns that had more than 30% 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 are shown in Figure 2.

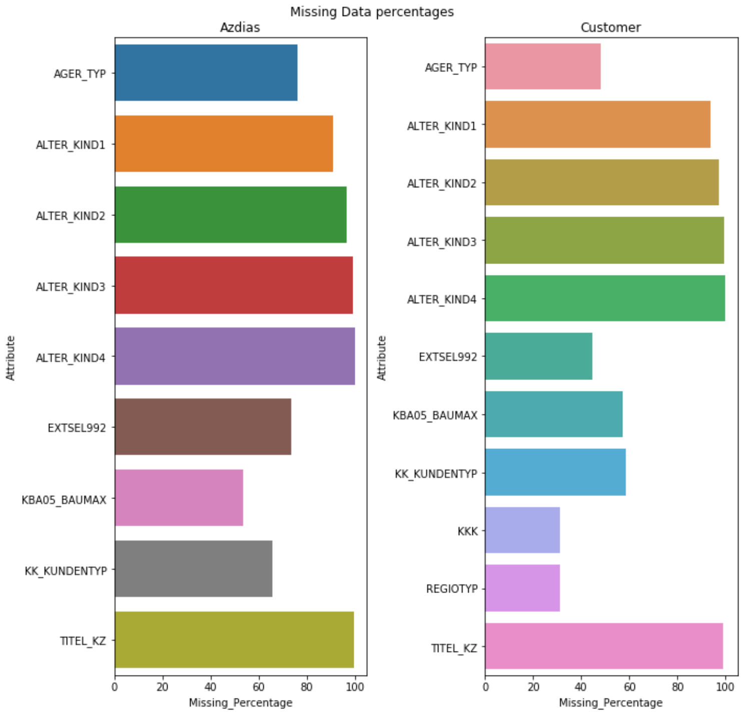


Fig 2: Columns with more than 30% missing values

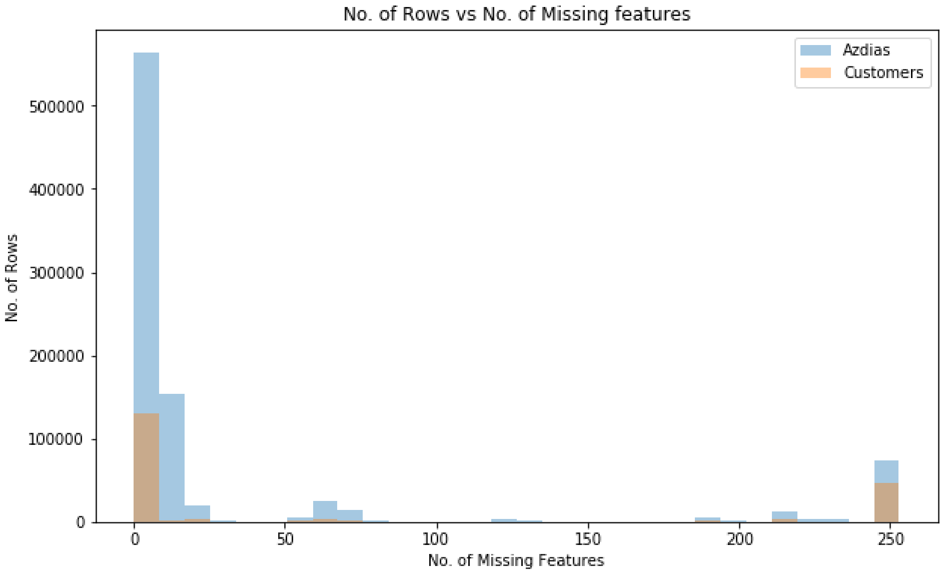


Fig 3: Distribution of missing features

7. **Imputing Missing Values**
   Even after removing features and rows which had missing values, the data still has missing values. These missing values is replaced by the values which occur most frequently in a column. Since the data corresponds to population in general, imputing the missing values with most frequent observations has been selected.
8. **Feature scaling**
   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.

Algorithms and Methodology

Customer Segmentation

The aim of this part of the project is to compare the general population and customers to determine future customers for this the general population and customers is divided into different segments. This requires a lot of analysis and the process is time consuming. Also, there might exist some complex interactions between these features which resulted in the person being a customer.

Dimensionality Reduction

The Principal Component Analysis (PCA) was performed on the given data to reduce the number of dimensions. PCA can be thought of as fitting a p-dimensional ellipsoid to the data, where each axis of the ellipsoid represents a principal component. If some axis of the ellipsoid is small, then the variance along that axis is also small, and by omitting that axis and its corresponding principal component from our representation of the dataset, we lose only an equally small amount of information. Since there were 353 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 4. As seen in the below figure, 90% of the variance can be explained with only 150 components that is why I have reduced the dimension to 150.

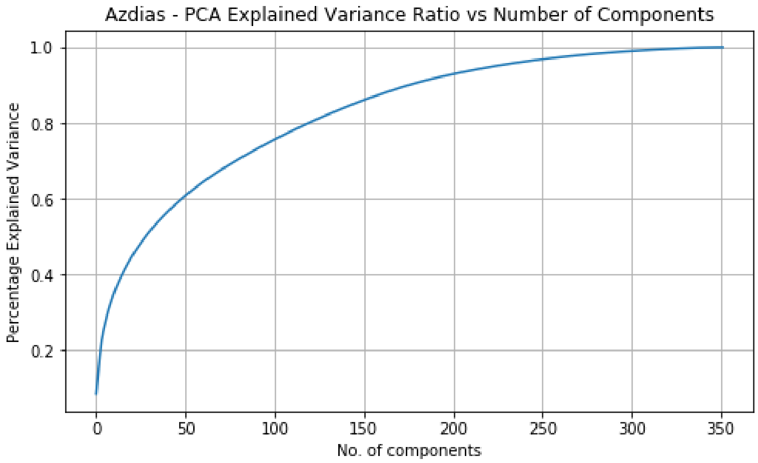


Fig 5: PCA Explained Variance Plot

Clustering

After the dimensionality reduction, the next step is to divide the general population and customer population into different segments. K-Means clustering algorithm has been chosen for this task. Since it 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. 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 minimize 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.

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 ‘8’, since the sum of squares of distances stops decreasing at a higher rate at this point as shown in Figure 6.

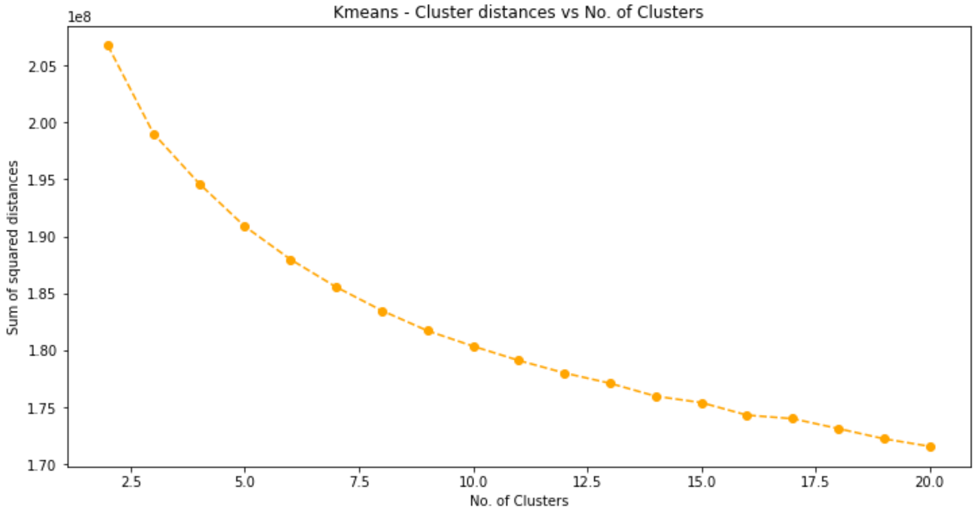


Fig 6: K-means Elbow Plot

Customer Acquisition

The second part of the project is to use supervised learning algorithms 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 data.

Benchmark Model

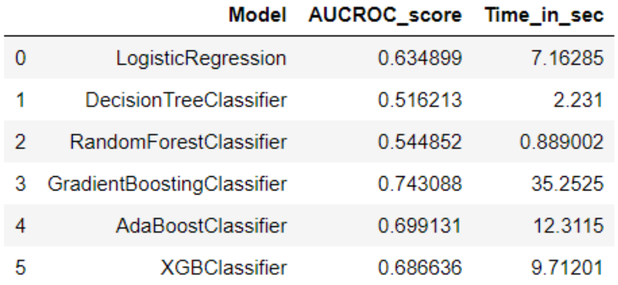
It is the model based on the performance of which rest of the models will be compared. The data is split into train and validation splits and a logistic regression model was trained on unscaled training data and evaluated on the unscaled validation data. **The AUROC score obtained for Logistic Regression model is 0.6341.**

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 validation split. The algorithms that have been selected for this step are:

- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier
- Gradient Boosting Classifier
- AdaBoost Classifier
- XG Boost Classifier

All the selected algorithms can be used for classification tasks. The performance of all the algorithms have been compared with each other and with the benchmark set in the previous step. The comparison can be seen in Figure 7.



	Model	AUCROC_score	Time_in_sec
0	LogisticRegression	0.634899	7.16285
1	DecisionTreeClassifier	0.516213	2.231
2	RandomForestClassifier	0.544852	0.889002
3	GradientBoostingClassifier	0.743088	35.2525
4	AdaBoostClassifier	0.699131	12.3115
5	XGBClassifier	0.686636	9.71201

Fig 7: Comparison of Performance on Scaled Data

The models are trained with default hyperparameters. There is no change in the performance of Logistic Regression. Performance of Decision Tree and Random Forest Classifier is not good. Best performance is of Gradient Boosting Classifier, but it takes too much time to train. AdaBoost and XGBoost have good performance and takes less time to train therefore they are chosen for hyperparameter tuning.

Hyperparameter Tuning

The selected algorithms, AdaBoost and XGBoost classifiers have been tuned with the help of a Grid Search. A set of hyperparameters for both the algorithms have been selected for tuning and a grid search has been performed for both the algorithms to determine the best performing models.

Results

After the hyperparameter tuning, the performance on the validation data with best models resulted in an improvement. The score for AdaBoost is 0.7431 and for XGBoost is 0.7478.

XGBoost and AdaBoost performs better than Logistic regression because there are a lot of imbalanced class and they tend to give more weight to those observations while training thus increasing the accuracy.

Since the algorithms used here are tree-based models, these algorithms can be analyzed further for the importance these models have given to each feature.

- Adaboost:
  The feature importance's for Adaboost model is shown in Figure 8. The feature 'D19_SOZIALES' is having the highest importance which follows by other features.

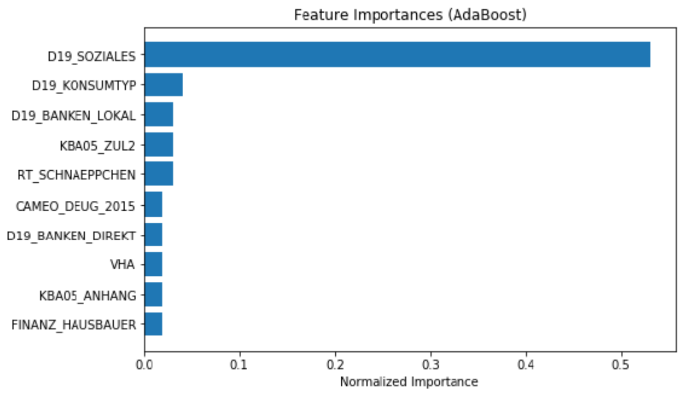


Fig 8: Adaboost feature importance

- XGBoost:
  The feature importance's for XGBoost model is shown in Figure 9. The feature 'D19_SOZIALES' is having the highest importance which follows by other features.

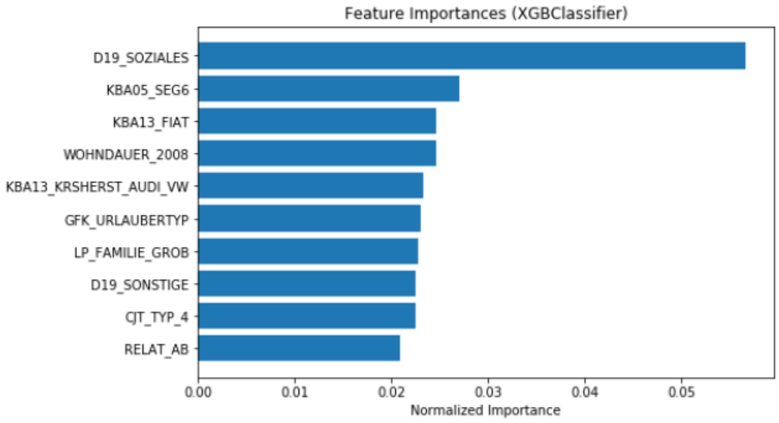


Fig 9: XGBoost feature importance

Both the algorithms have given the highest importance to ‘D19_SOZIALES’ feature. There is no description given in the attribute information files. The feature importance’s with the XGboost model seem to be well distributed when compared to AdaBoost model. This might be due to the way these algorithms are designed, Adaboost improves upon weak learners by identifying short comings in the highly weighted data points, whereas the XGBoost algorithm improves upon the weak learners with the help of gradients coming from an objective function.

Prediction on Test Data

The final predictions were made on the test data which was provided in the file ‘Udacity_MAILOUT_052018_TEST.csv’. The same pre-processing steps were performed to clean the data. This data was scaled with the scaler which was fit on the training data.