Capstone Project

Machine Learning Engineer Nanodegree

Abhra Debroy

2nd Feb 2020

Customer Segmentation

for

Arvato Financial Services

Definition

Project Overview

This is a customer segmentation project for Arvato Financial Solutions, a Bertelsmann subsidiary. Arvato provides solution to run credit management effortlessly and efficiently, ultimately resulting in optimized financial performance for diverse industries. Arvato helps predict payment behavior and calculate or predict credit score for end-customers.

In this project we will analyze attributes of Arvato's existing customers and compare with demographic information of general population of Germany. Using knowledge from above analysis Mail-order company will devise efficient customer acquisition strategy and appropriate target for marketing campaign

Problem Statement

We need to work on two related problem statements for this project:

1. Create customer segmentation by analyze demographics information of existing customers of a mail-order company engaged by Arvato Financial and comparing it against demographics information of general population of Germany
2. Build a machine learning model to predicts whether an individual will respond to the marketing campaign by Mail-order company and test the model. For this we will use knowledge from [1] customer segmentation analysis and dataset with attributes from targets of a mail order campaign.

Analysis

Datasets and inputs

There are four data files associated with this project:

1. Udacity_AZDIAS_052018.csv: Demographics data for the general population of Germany
2. Udacity_CUSTOMERS_052018.csv: Demographics data for customers of a mail-order company
3. Udacity_MAILOUT_052018_TRAIN.csv: Demographics data for individuals who were targets of a marketing campaign
4. Udacity_MAILOUT_052018_TEST.csv: Demographics data for individuals who were targets of a marketing campaign.

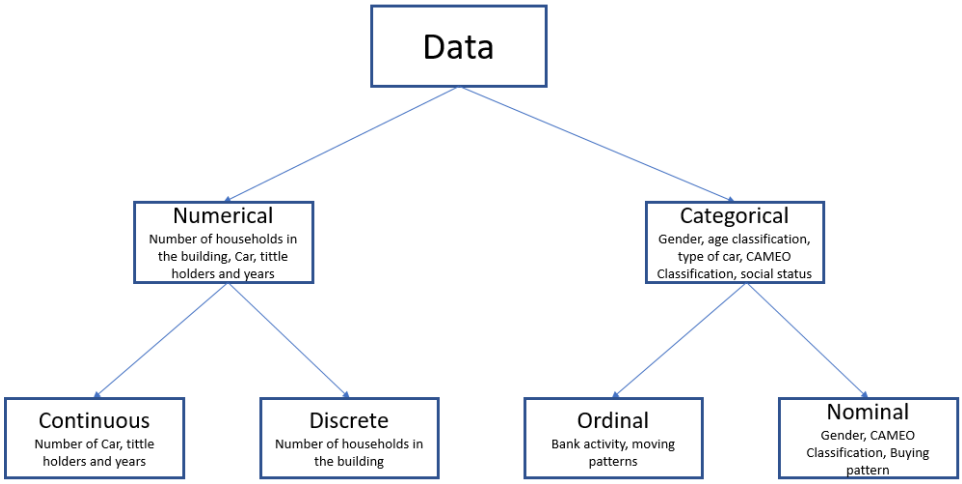
Data Exploration

Demographics data for the general population of Germany has 891 211 persons data (rows) x 366 features (columns).

Demographics data for customers of a mail-order company; 191 652 persons data (rows) x 369 features (columns). Customer data contains three extra columns ('CUSTOMER_GROUP', 'ONLINE_PURCHASE', and 'PRODUCT_GROUP'), which provide broad information about the customers.

Demographics data for individuals who were targets of a marketing campaign, is split into two (training and testing data) each having 42 982 persons data (rows) x 366 (columns). Training data has one extra columns for label for targets.

Dataset consists both Categorical and Numerical data.



Data Cleaning and Preprocessing

There are high number of missing data (range of 20% and more) which required to delete, replace or imputed with appropriate values.

1. There are 45 no of features deleted based on criteria different of missing values (more than 20%), information is missing in *DIAS Attributes - Values 2017.xlsx* and redundant information. Name of these columns for deletion are store into list *columns_to_delete*

2. Deleted rows having more than 100 missing values to reduce data noise
3. There are data having invalid values like ('XX' and 'X'). These are replaced with NaN
4. Feature 'OST_WEST_KZ' has two values 'W' and 'O', replaced these with '0' and '1' correspondingly
5. Few features are of categorical having high cardinality, we choose to Label encode (example CAMEO_DEU_2015)
6. Numerical features (stored into list *numerical_col*) are imputed with median value
7. Categorical features (stored into list *categorical_col*) are imputed with Mode i.e. 'most_frequent' value
8. Range of data values with differs in the order of magnitude, we need to scale the features by standardization.

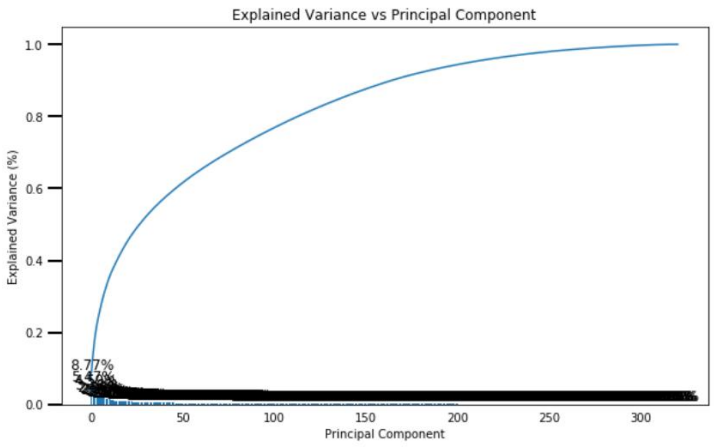
Solution Approach

Dimensionality Reduction

There are 350+ features in the dataset. Higher number of features increase the time and storage space required for data analysis. Dimensionality reduction is the process of reducing the dimensionality of the feature space with consideration of obtaining a set of important features. This helps remove redundant and irrelevant features without incurring much information loss. It also becomes easier to visualize the data.

Principal Component Analysis (PCA) is one of the most popular dimension reduction technique. It is a projection-based method which transforms the data by projecting it onto a set of orthogonal axes.

We performed PAC on our dataset. By this we reduced our features to 175 components from 350+ feature with 90% of information retained.



Above graph show that 175 components account for just over 90% of the variance. That means using these 175 components, we cab explain essential characteristics of the data.

Customer segmentation can be done with unsupervised learning method (example clustering algorithm K-mean) and second problem is a supervised learning method.

Evaluation metrics

Unsupervised learning method

For assessing unsupervised learning method, we do not have ground truth labels. Therefore, we can use Silhouette Coefficient for the model performance evaluation.

For deciding optimal number of clusters, I used both Elbow method and Silhouette score. In Elbow method, SSE calculation considers only intra cluster distance i.e. Mean square distance between each instance and its centroid. Silhouette Score consider both intra cluster distance and next closest cluster. That's why Silhouette score gave better result.

Supervised learning method

For supervised learning method, we can use Confusion Matrix, Accuracy, Precision, Recall and F1-Score, Area Under ROC for model evaluation.

For our problem, positive class consist of 1.25% of data. This is case of imbalance data problem. Standard measure like accuracy, precision, recall or F-score are not a reliably measures for performance evaluation of imbalance class. For this kind of problem, we recommend Area Under Receiver Operating Characteristic curve (AUROC). This curve can correctly assess both the class equally

The curve is a plot of *false positive rate* versus the *true positive rate* is) for a number of different training instances and choose a threshold that gives a desirable balance between the false positives and false negatives.

False Positive Rate=FP/(FP+TN)

True Positive Rate=TP/(TP+FN)

Benchmarking and Model Evaluation

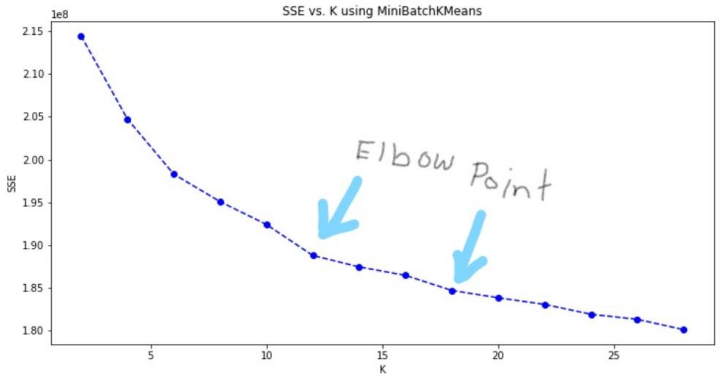
Unsupervised Clustering algorithm

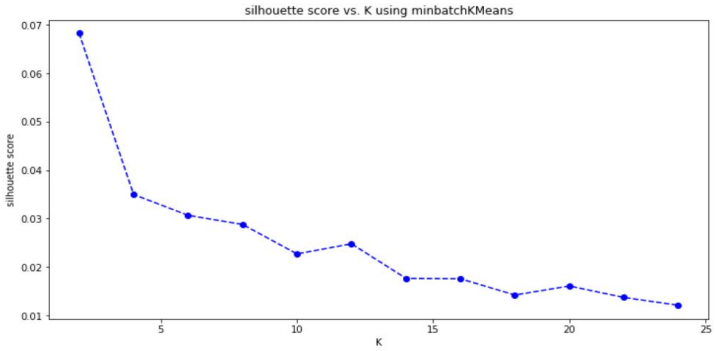
For customer segmentation problem I have chosen different clustering algorithms to benchmark performance and then chose one best performing algorithm to cluster demographic information for German population and customers of mail company.

I have started with MiniBatchKMeans clustering algorithm (centroid-based models). KMeans is the most common clustering algorithm because it is simple, easy to understand and implement. MniBatchKmeans is variant of KMeans but take less computation time while attempting to optimize same objective function. Algorithms I planned tp evaluate are few from below list:

- Centroid-based: KMeans
- Density-based: DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
- Expectation-maximization: Gaussian Mixture Models (GMM)

- Connectivity-based, or hierarchical: Hierarchical Clustering Algorithms

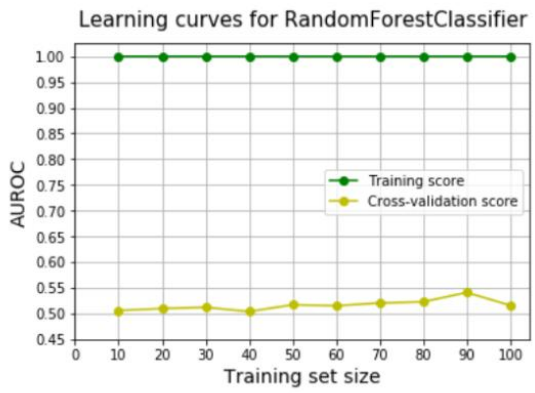




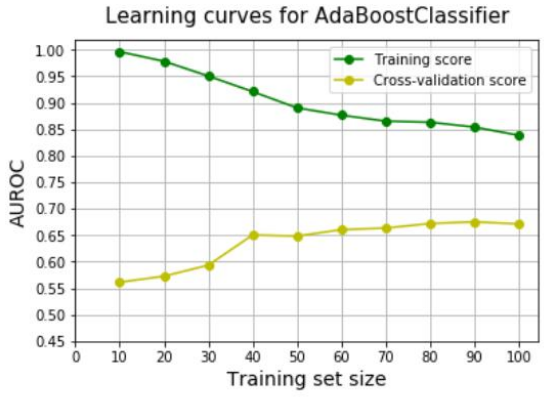
.

Supervised Classification algorithm

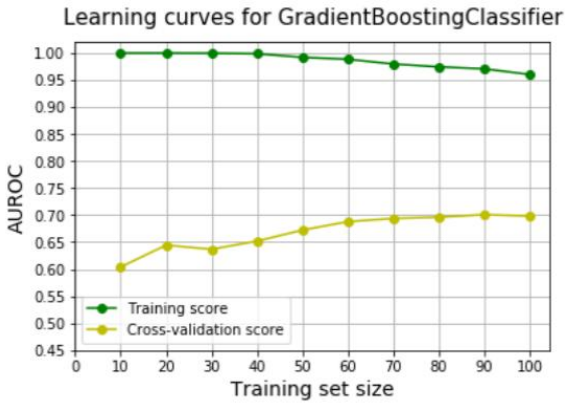
I have evaluated few models using learning curve and chose best model based on Area Under ROC (AUROC). I have selected a Random Forest, few Ensemble classifiers which are insensitive to imbalance data (like cost-sensitive learning) example – AdaBoostClassifier and GradientBoostingClassifier and a SVM model



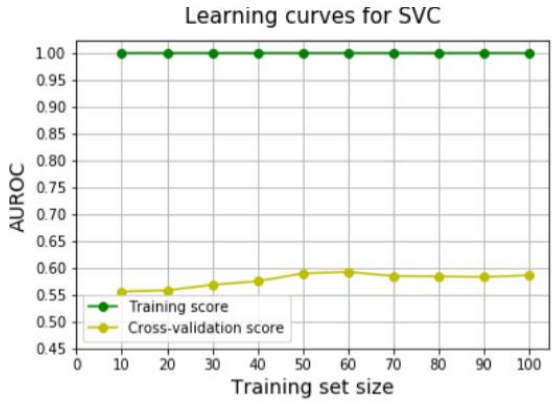

```
Roc_auc train score = 1.0
Roc_auc validation score = 0.52
```




```
Roc_auc train score = 0.84
Roc_auc validation score = 0.67
```




```
Roc_auc train score = 0.96
Roc_auc validation score = 0.7
```




```
Roc_auc train score = 1.0
Roc_auc validation score = 0.59
```

Results of Model Evaluation

Unsupervised Clustering algorithm

MiniBatchKMeans which is fast and less consuming, gave fare idea about the number of clusters required for the problem. Thereafter, I have tried other three algorithm KMeans, Gaussian Mixture Model (GMM) and AgglomerativeClustering.

AgglomerativeClustering was not suitable for large data like ours. I was getting Memory Error. Comparing KMeans and Gaussian Mixture Model, KMeans gives better Silhouette score. Therefore, for final customer segmentation clustering I have used KMeans crusting on the PCA reduced dataset.

Supervised Classification algorithm

For RandomForestClassifier - AUROC for training set is 1 and validation set in 0.52. Validation has more error. This is case of overfitting. Model is highly biased. It was fastest algorithm.

AdaBoostClassifier : AUROC for training is 0.84 and validation is 0.67. Both the curve has converged well in learning curve graph. This is relatively better model than RandomForestClassifier.

GradientBoostingClassifier : AUROC for training is 0.96 and validation is 0.7. Both the curve has not fully converged. This model has highest ROC for both training and validation. This model takes more execution time than AdaBoostClassifier.

SVC: AUROC for training is 1 and validation is 0.59. Validation has more error. This is the case of overfitting. Model is highly biased. This model has highest execution time. This is worst performing model.

AdaBoostClassifier and GradientBoostingClassifier has almost similar AUROC for validation set. But learning curve for AdaBoostClassifier merges better than GradientBoostingClassifier and also has less execution time. Therefore, final model for supervise classification I have chosen AdaBoostClassifier.

Thereafter, we used GridSearchCV for hyperparameters tuning for AdaBoostClassifier.

Further Improvements for data handling

Apart from what achieved in the project, there could be other possible improvement for handling imbalance class classification problem as described below:

1. Random under-sampling of the majority class
2. Random over-sampling of the minority class
3. Random under-sampling may lead to potential loss of information as lot of data instances are taken way. In this case we can perform a informed under-sampling by finding out the distribution of data first and selectively removing majority class.
4. SMOTE: Synthetic Minority Over-sampling Technique has been designed to generate new samples that are coherent with the minor class distribution. The main idea is to oversample with synthetically generated data points that are not too different from the minority class data