Capstone Report

Customer Segmentation and Optimization of Customer Acquisition with Arvato Financial Solutions

Funing Tian
24th June 2021
Machine Learning Engineer Nanodegree
School of Artificial Intelligence

Contents

I. Definition

Project Overview
Arvato, wholly owned by Bertelsmann, is a services company that actively develops and implements innovative solutions for customers on a global scale. It provides services including customer support, information technology, logistics and finance [1]. As part of its services, Arvato is helping client companies get invaluable insights into client profiling and marketing.

In this project, we will employ the use of machine learning to deal with real-life data provided by Bertelsmann Arvato Analytics. More specifically, Arvato is helping a client mail-order company to better target next probable customers for its products.

To fulfill the goal, we will focus on customer segmentation, that is, characterizing customers segment of population based upon well-defined specific features [2]. Diving customers into groups based on common characteristics will enable our client company to market each group effectively and properly. Here, we will analyze demographics data of customers and the general German population. This will be followed by developing a supervised machine learning model to make predictions on whether a person will be a new customer.

Access to the comprehensive project files is given here:
https://github.com/tian570/Bertelsmann-Arvato-ML.

Datasets and Inputs
The project makes use of four data files:

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).

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, two description files have been provided to give attribute information:

5) DIAS Information Levels - Attributes 2017.xlsx: A top-level list of attributes and descriptions, organized by informational category.

6) DIAS Attributes - Values 2017.xlsx file: A detailed mapping of data values for each feature in alphabetical order.

Problem Statement

The problem that we will be investigating can be formulated as: "Given the access to German demographic profiles, how can the German mail-order company acquire new customers efficiently?"

This problem requires us to consider what we can do to predict with high accuracy whether a person with associated demographics data will be a new customer to our client company. Furthermore, how can we predict with confidence the probability of people with demographic profiles turning into future customers?

Evaluation Metrics

The selected supervised machine learning model will be used to make predictions on the mailout campaign data in competition through Kaggle [3]. The evaluation metric for this Kaggle competition is the Area Under the Receiver Operating Characteristics Curve (AUC for the ROC curve).

The decision of selecting the AUC-ROC curve to measure performance is due to the fact that it is one of the best performance evaluation techniques in imbalanced cases. In our case, the MAILOUR_TRAIN dataset displays large class imbalance, where most individuals did not respond to the mailout (Fig.1). Thus, predicting individual classes and using accuracy does not seem to be an appropriate performance evaluation method.

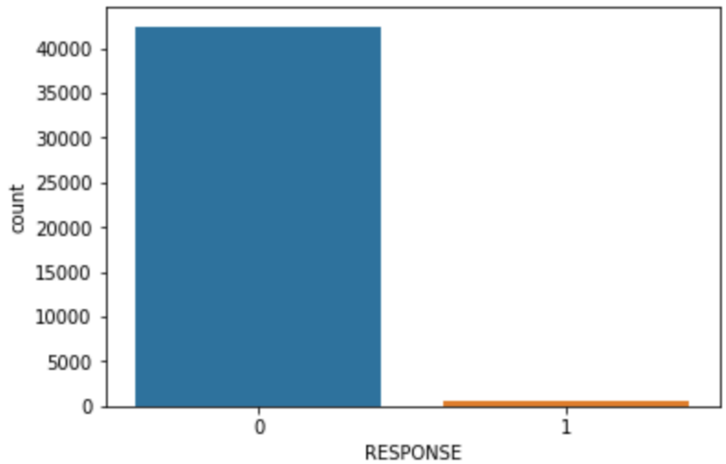


Figure 1. Counts of responses of individuals in the MAILOUT_TRAIN dataset.

The interpretation of the AUC-ROC curve is available in the Proposal file in the GitHub repository.

The final file that I have submitted to the associated Kaggle competition can be found under kaggle_submission/ in the GitHub repository.

II. Analysis

Data Exploration

Exploring the AZDIAS and CUSTOMERS datasets is the first step in this project. The two datasets contain the expected number of rows and columns as described. Each row of the two datasets represents a single person, but also includes information outside of individuals, including information about their household, building, and neighborhood. Compared with AZDIAS, CUSTOMERS contains three extra columns ('CUSTOMER_GROUP', 'ONLINE_PURCHASE', and 'PRODUCT_GROUP'), which provide broad information about the customers depicted in the file.

The MAILOUT train and test datasets share similar structures with the two datasets mentioned above. However, the MAILOUT dataset includes one additional column, which indicated whether or not each recipient became a customer of the company. For the "TRAIN" subset, this column has been retained, but in the "TEST" subset it has been removed.

The two metadata file helps in understanding what each feature represents and cleaning unknown values.

Data Pre-processing

Exploring data helps gain an insight into data points and features. However, since it is required to use complete data points to train a model, any missing or mis-recorded values requires to be cleaned. Here, we start with examining AZDIAS. The other three datasets will be pre-processed in a similar fashion.

1. Addressing unknown values

The first step is to deal with unknown representations across all the columns. In the attributes-values spreadsheet, unknown records are represented by one or multiple numerical values. These unknown values should be treated as missing entries. Thus, they were changed to NAN values.

2. Exploring missing values both column-wise and row-wise

After replacing unknown values with NAN values, missing values were studied both column-wise and row-wise.

We investigated the proportion of NAN values per column. It turned out that there were many missing values in the AZDIAS, especially eight features having more than 60% missing values (Fig.2). To figure out a threshold for dropping columns, we plotted number of columns at different percentage of missing values. Since most of the columns had percentage of missing values of less than 20% (Fig.3), we decided to take 20% as the cut-off.

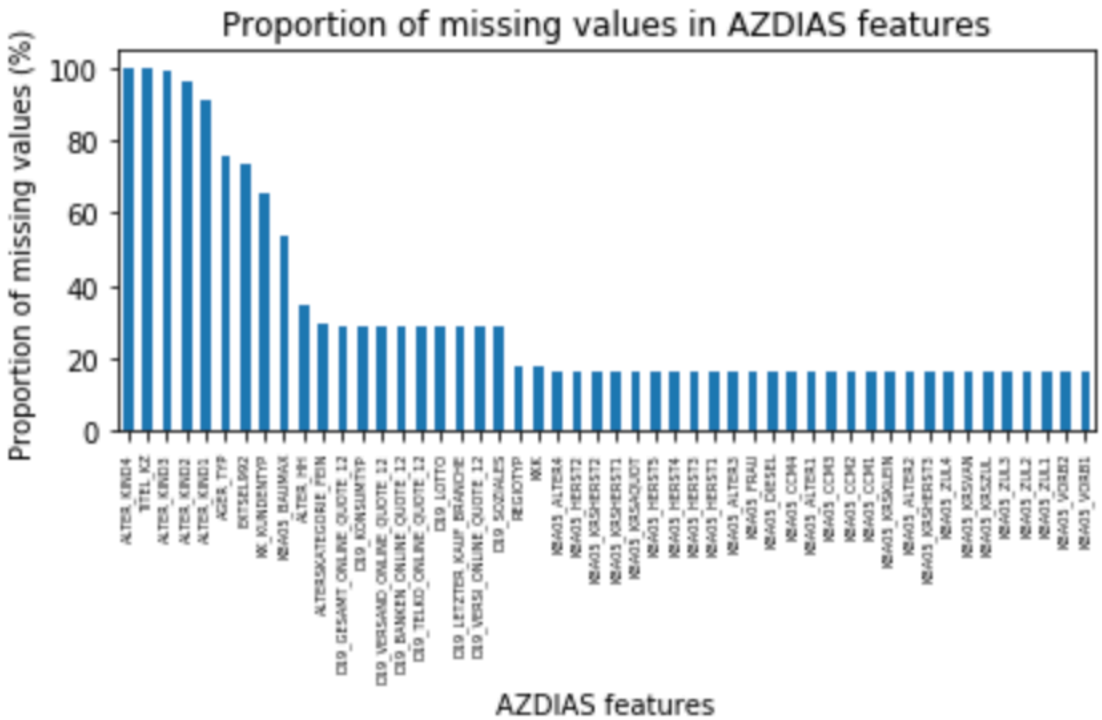


Figure 2. Percentage of missing values per ASDIAS features

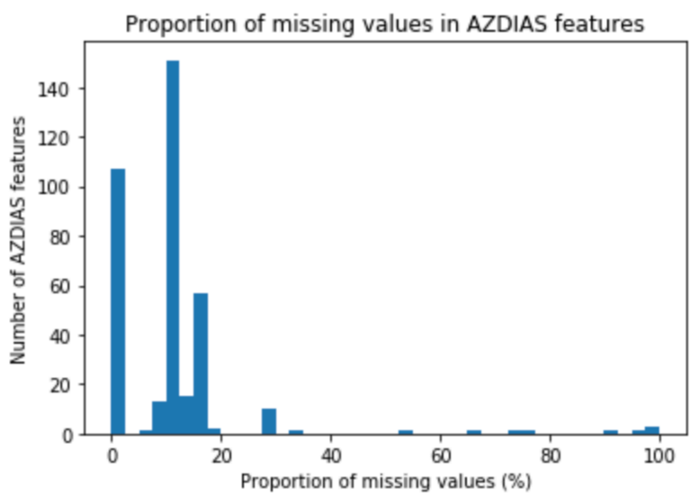


Figure 3. Number of columns for the proportion of missing values in AZDIAS.

We next explored missing values per row. Similarly, we were trying to figure out a threshold to drop rows containing too many missing values. Most of the columns had percentage of missing values of less than 10%. However, there are also a high number of rows with missing values greater than 60% (Fig.4). We decided to drop any rows that have over 10% missing values.

Figure 4. Number of rows for the proportion of missing values in AZDIAS.

3. Re-encoding categorical and mixed features

Some columns in the AZDIAS dataset contain categorical values. Some other may include mixed types. To prepare these for feature extraction, we'll want to convert these into numerical values.

During the exploration of categorical and mixed features, we managed to convert 'EINGEFUEGT_AM' to a datetime type and treat 'X' and 'XX' as NAN values.

For binary categorical variables that have non-numerical values, we need to convert them to numerical data. Specifically, 'OST_WEST_KZ' was the binary categorical variable that was converted to numerical values based upon the DIAS Attributes - Values 2017.xlsx file.

For multi-level categorical variables, they may increase the number of features for modeling. Therefore, we may consider dropping them. We dropped 'CAMEO_DE U_2015' and 'EINGEFUEGT_AM' in this step since both of them were with more t han 10 values.

In terms of mixed features, we need to convert them to numerical values. There exist 4 mixed features, including 'CAMEO_INTL_2015', 'LP_LEBENSPHASE_FEIN', 'LP_LEBENSPHASE_GROB' and 'PRAEGENDE_JUGENDJAHRE'. For these, we need to split them into multiple single-type features. 'CAMEO_INTL_2015' was re-encoded into 'CAMEO_INTL_2015_WEALTH_LEVEL' and 'CAMEO_INTL_2015_STATUS'. Features that started with LP had duplicated information. A function was developed to handle these features. The 4th mixed-feature column was

'PRAEGENDE_JUGENDJAHRE'. We re-encoded this feature to contain two values being Mainstream and Avantgarde, respectively.

4. Dropping highly correlated features
We will be implementing feature correlation to determine too high-correlated features since they many over-inflate the importance of a single feature. In this project, we defined too highly correlated features as having correlations with a column over 0.9. In this step, 15 features were dropped.

5. Completing missing values
After removing features and rows having proportion of missing values over the thresholds, there still exist missing values in the data. For the remaining missing values, we imputed them by using median values in each column.

6. Scaling features
Prior to applying PCA, we need to scale features to be of the same range since PCA may be influenced by variations in scales of features. A standard scaler was used to bring all the features to the same range.

III. Algorithms, Techniques and Methodology

Customer Segmentation
The goal of the first part of the project is to characterize the relationship between existing customers and the demographics of the German population, and to find out which types of people in the general population are more likely to become customers.

1. Dimensionality Reduction
Before feeding data into a machine learning model, a dimensionality reduction step is a necessity with the aim being forming a smaller set of features to better help separate our data. The technique that was used to reduce the number of features was the principal component analysis (PCA).

The idea of reducing dimensionality is that we want to keep a small number of features while retaining high explained data variance. To decide how many top components to include, it's helpful to look at how much data variance the components capture. From the two figures below (Fig.5 and Fig.6), we can conclude that 150 features can explain around 85% data variance. Further check on explained data variance by 150 principal components demonstrated that they could explain 86% data variance in AZDIAS and 85% data variance in CUSTOMERS.

Therefore, I decided to keep 150 principal components since they could retain the main information in both datasets.

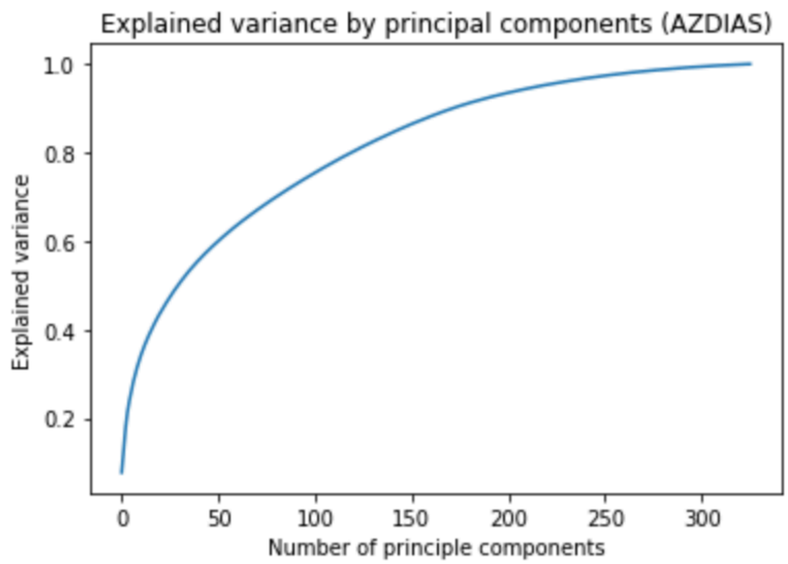


Figure 5. cumulative explained data variance by principal components in AZDIAS.

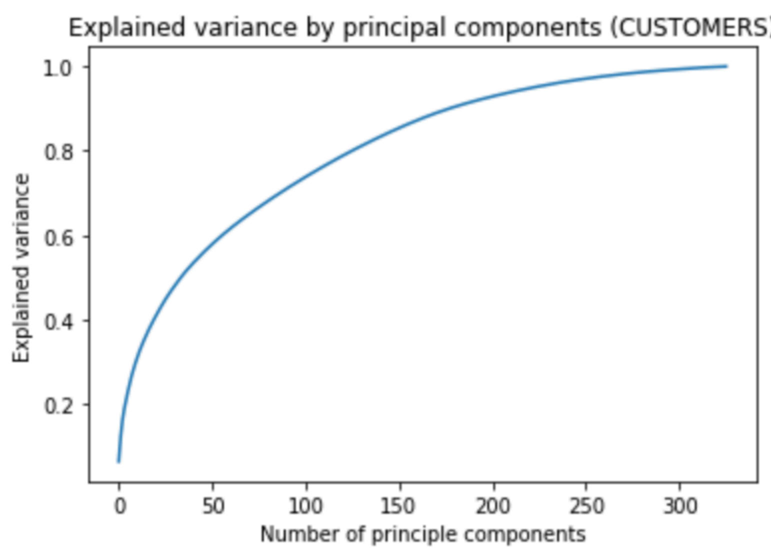


Figure 6. Cumulative explained data variance by principal components in CUSTOMERS.

2. Clustering

After having PCA attributes set up, the next is to divide the general population and the customer base into different segments. The unsupervised clustering algorithm, k-means, is chosen to segment the general population and customers. We want to select a k such that data points in a single cluster are close together but that there are enough clusters to effectively separate the data. After trying several values for k, the centroid distance typically reaches some "elbow"; it stops decreasing at a sharp rate and this indicates a good value of k.

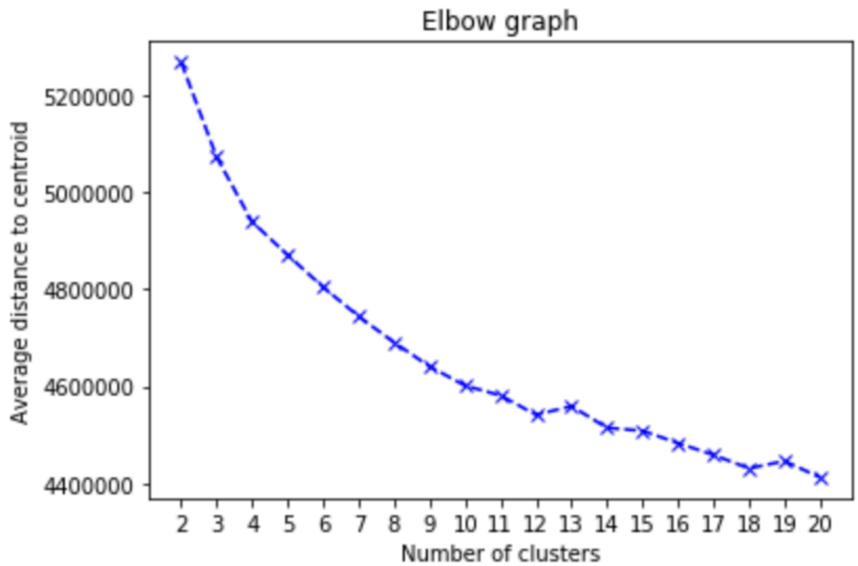


Figure 7. Elbow graph showing within-cluster distance for different number of clusters

From the elbow graph (Fig. 7), we can see that within-cluster distances decrease at a high rate until reaching 12 clusters. This indicates that there is enough separation to distinguish the data points in each cluster, but also that we included enough clusters so that the data points aren’t extremely far away from each cluster. Thus, we came to the conclusion that 12 was the optimal number of clusters.

We next looked into clusters and compared the general population and customer base for each of the 12 clusters. We can know that some clusters were overrepresented by customers whereas others didn't (Fig. 8). This gives us an idea of better targeting future customers.

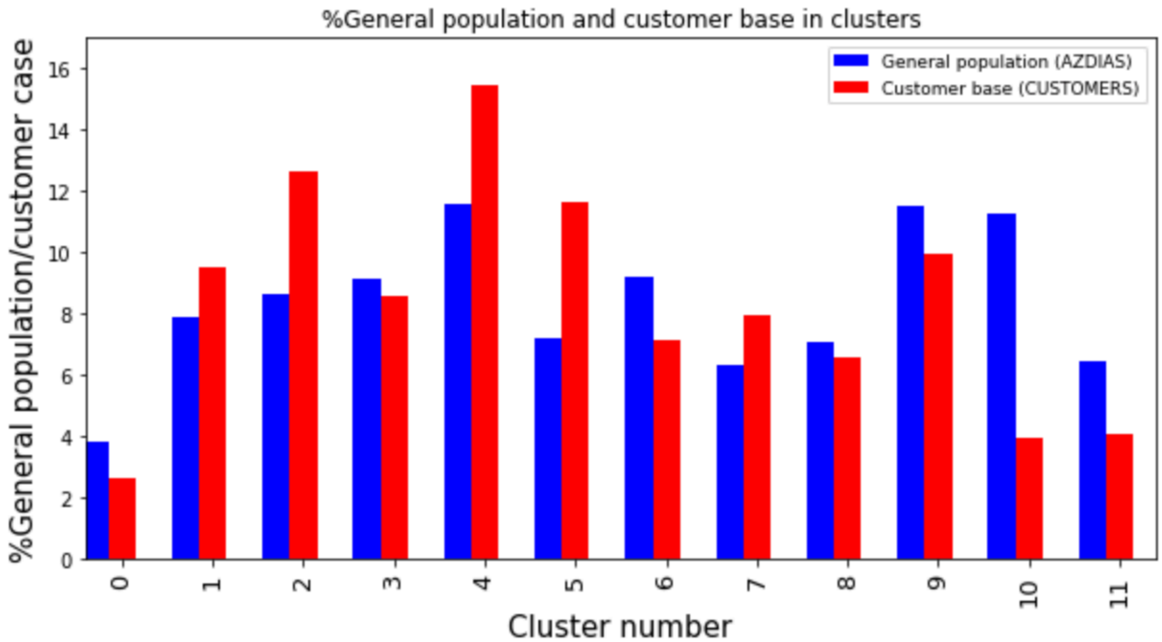


Figure 8. Percentage of general population vs customer base in each cluster.

Customer Acquisition

The second part of this project is to use supervised learning techniques to predict potential customers based on demographic data. We will be using the MAILOUT train and test datasets. As mentioned, the MAILOUT dataset has one additional column included in the training dataset, which indicated whether or not a person will become a customer of the company. The two datasets were pre-processed and scaled using similar strategies applied to AZDIAS and CUSTOMERS. Additionally, in the MAILOUT_TRAIN dataset, we observed large class imbalance (Fig.1). To deal with this, we resampled the minority class, leading to the same number of observations in both response classes.

We have tried the following supervised learning algorithms (each with performances):

1) BenchMark Model: Logistic Regression

```
Time taken: 79.48352980613708
AUROC score: 0.7869803539944817
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='warn',
          n_jobs=None, penalty='l2', random_state=10, solver='warn',
          tol=0.0001, verbose=0, warm_start=False)
```

2) Random Forest Classifier

```
Time taken: 18.93514609336853
AUROC score: 0.9934712850584364
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=None,
            oob_score=False, random_state=10, verbose=0, warm_start=False)
```

3) Gradient Boosting Classifier

```
Time taken: 942.5874569416046
AUROC score: 0.9024049565159769
GradientBoostingClassifier(criterion='friedman_mse', init=None,
              learning_rate=0.1, loss='deviance', max_depth=3,
              max_features=None, max_leaf_nodes=None,
              min_impurity_decrease=0.0, min_impurity_split=None,
              min_samples_leaf=1, min_samples_split=2,
              min_weight_fraction_leaf=0.0, n_estimators=100,
              n_iter_no_change=None, presort='auto', random_state=10,
              subsample=1.0, tol=0.0001, validation_fraction=0.1,
              verbose=0, warm_start=False)
```

4) Light GBM Classifier

```
Time taken: 35.285934925079346
AUROC score: 0.9874326898738096
LGBMClassifier(boosting_type='gbdt', class_weight=None, colsample_bytree=1.0,
        importance_type='split', learning_rate=0.1, max_depth=-1,
        min_child_samples=20, min_child_weight=0.001, min_split_gain=0.0,
        n_estimators=100, n_jobs=-1, num_leaves=31, objective=None,
        random_state=10, reg_alpha=0.0, reg_lambda=0.0, silent=True,
        subsample=1.0, subsample_for_bin=200000, subsample_freq=0)
```

5) Ada Boost Classifier

```
Time taken: 89.2452142238617
AUROC score: 0.8185195296641141
AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None,
          learning_rate=1.0, n_estimators=50, random_state=10)
```

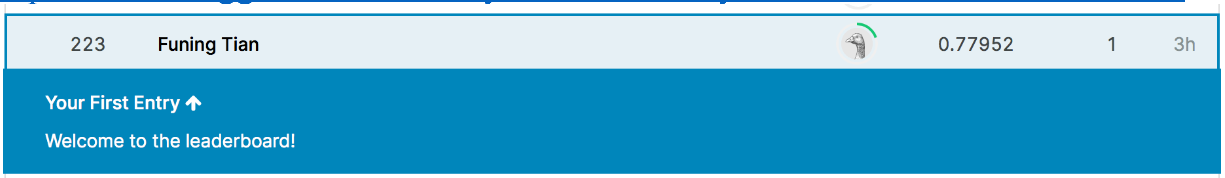
6) XG Boost Classifier

```
Time taken: 1299.9323909282684
AUROC score: 0.896656927390466
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
       colsample_bynode=1, colsample_bytree=1, gamma=0, learning_rate=0.1,
       max_delta_step=0, max_depth=3, min_child_weight=1, missing=None,
       n_estimators=100, n_jobs=1, nthread=None,
       objective='binary:logistic', random_state=10, reg_alpha=0,
       reg_lambda=1, scale_pos_weight=1, seed=None, silent=None,
       subsample=1, verbosity=1)
```

IV. Results
After hyperparameter tuning, the best trained model was the Gradient Boosting Classifier algorithm with an AUROC score being 0.90.

Final predictions were made on the preprocessed and scaled MAILOUT_TEST dataset. Among 42,833 individuals listed in the test dataset, 9,115 individuals are likely to become a customer of the mail-order company, which accounts for approximately 21% of the test individuals.

The predictions on the test dataset were submitted to Kaggle.
https://www.kaggle.com/c/udacity-arvato-identify-customers/leaderboard#score



V. Improvements and Future Steps
Although I got one of the high scores, there is still room for improvement. We may consider including the following steps in future improvements:

1. Addressing more multi-level categorical features and one-hot encoding them
2. Removing outliers
3. Dealing with class imbalance using down-sampling methods
4. Setting different ranges of hyperparameters

References
[1] Arvato. *Wikipedia*. https://en.wikipedia.org/wiki/Arvato
[2] Customer Segmentation. *Wikipedia*.
https://en.wikipedia.org/wiki/Market_segmentation
[3] Udacity+Arvato: Identify Customer Segments. *Kaggle*.
https://www.kaggle.com/c/udacity-arvato-identify-customers/overview/evaluation