Capstone Project Proposal

Create a Customer Segmentation Report for Arvato Financial Solutions

Project overview

Customer acquisition is very important to the growth of a business. How to efficiently bring new customers and new clients to your business is one of main objects for the marketing teams around the world.

In this project, I will work with customers datasets from Arvato, and the goal is to create user segments and predict if a customer will response to the campaign.

Problem & Solution Statements

The ultimate goal of this project is to train a model to predict if a customer would like to response to a campaign. I will split the project into several tasks.

1. Download data and understand data (attributions).
2. Data preparation and feature engineering. In this part, we shall analyze data and deal with missing values.
3. User segmentation. To solve this problem, we will analyze the general population data and customer data, and use unsupervised machine learning algorithm to performance user segmentations, which could identify the users that mostly likely to become customers. We will use K-MEAN algorithm to make customers segmentations of population and then to describe the cluster of population who most likely convert. As we all know that one of the limitation of K-mean is high dimensional data. To solve this problem, we will use PCA to reduce data dimensionality first.
4. Train a classification model on AWS SageMaker to determine if a user likely to response the campaign. In this part, we will choose several binary classification algorithms, then compare their performances, finally choose the best one to predict who is very likely to become a customer. We shall work with the following common classification algorithms: random forest, gradient boosting, ect.

Datasets and inputs

In this project, we will work with 4 datasets.

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).

Evaluation metrics

One of the most common evaluation metrics for binary classification problem is *Area Under ROC Curve*. It presents the ability of a model to discriminate between positive and negative classes. An area of 1.0 represents a model could predict positive classes perfectly. And an area of 0.5 represents that the prediction of a model is as good as random. We shall use 0.5 as a *Benchmark*. Here is the definition of the ROC curve [2].

> A ROC Curve is a plot of the true positive rate and the false positive rate for a given set of probability predictions at different thresholds used to map the probabilities to class labels.

Benchmark model

In the Kaggle competition, the highest AUC for the ROC curve score is 0.84739. We shall set it to be the benchmark.

Project design

Data preprocessing

Missing values

There are lots of missing values in the datasets, which is needed to be handles. We could fill these nulls with mean or mode.

Attributes study

In order to understand the dataset, some time is required to spend on datasets' attributes study. If there are too many missing values in one column, we could drop it since it does not contain enough

useful information.

Encoding Data

Since most machine learning models only accept numerical variables, we shall convert these categorical variables to numbers such that the model is able to understand and extract valuable information.

Training & Testing

K-Mean clustering

1. Data normalization.
2. Use PCA for dimensionality reduction
3. Apply K-Mean clustering
4. Use Elbow to choose the number of clusters
5. Compare Customers data with AZDIAS data and interpret the results.

Classification Problem

1. Split dataset into training and validating sets.
2. Train selected classification models, such as, random forest, gradient boosting, ect.
3. Tune the parameters for each model and choose the best one based on AUC score.

1. A New Method for Dimensionality Reduction using K-Means Clustering Algorithm for High Dimensional Data Sets Clustering Algorithm for High Dimensional Data Set

2. Metrics To Evaluate Machine Learning Algorithms in Python