**Machine Learning Engineer Nano degree**
Capstone Proposal

Nyaribo Maseru
January 20th, 2020

# Customer Segmentation Using Data and Machine Learning Algorithms For Arvato Financial Solutions

## Domain Background

This project is a partial fulfilment for Udacity's Machine Learning NanoDegree program. The project is based on real-life data science problem provided by Bertelsmann Arvato Analytics.In this project we are provided with demographic datasets of customers of a mail-order company in Germany, and demographic data of the general population of Germany.

The mission of the project is to make predictions based on data and machine learning algorithms on individuals who are mostly likely to become customers for a mail order sales company in Germany, instead of solely relying on gut feeling and intuition from senior experienced managers.

## Problem Statement

The problem is how the mail-order company is to increase efficiency in customer acquisition process. Thus, instead of the company reaching out to all people in Germany, and then targeting them with marketing campaigns, it can just use the trained model to reach out to the people identified as becoming most likely new customers, and then to do targeted advertising.

## Datasets and Inputs

The data that i shall use has been provided by Bertelsmann Arvato Analytics, and represents a real-life data science task. The description or "Get to Know the Data" below is from Udacity.

There are four data files associated with this project:

1. Udacity_AZDIAS_052018.csv: Demographics data for the general population of Germany; 891211 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).

Each row of the demographics files represents a single person, but also includes information outside of individuals, including information about their household, building, and neighbourhood. Use the information from the first two files to figure out how customers ("CUSTOMERS") are similar to or differ from the general population at large ("AZDIAS"), then use your analysis to make predictions on the other two files ("MAILOUT"), predicting which recipients are most likely to become a customer for the mail-order company.

The "CUSTOMERS" file contains three extra columns (‘CUSTOMER_GROUP', 'ONLINE_PURCHASE', and 'PRODUCT_GROUP'), which provide broad information about the customers depicted in the file. The original "MAILOUT" file included one additional column, "RESPONSE", 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; it is against that withheld column that your final predictions will be assessed in the Kaggle competition.

**Solution Statement**

The approach to the problem stated above shall be in two parts as follows;

Part 1: Use unsupervised learning techniques to perform customer segmentation, identifying the parts of the population that best describe the core customer base of the company.

This shall involve analysis of demographics data for customers of a mail-order sales company in Germany, comparing it against demographics information for the general population. Then, use the findings to apply on a third dataset with demographics information for targets of a marketing campaign for the company.

Part 2: Use a supervised model to predict which individuals are most likely to convert into becoming customers for a marketing campaign for the company.

We know the goal of the mail-order company is to increase efficiency towards customer acquisition for targeted advertising. The model design is optimised to know users who should NOT be targeted. That is, we want to have as few false positives (0s classified as 1s) as possible.

Conversely, if the mail-order sales company asks for an application that will show ALL potential customers to target, even if it means a higher a number of false positives, then we'd want as few negatives as possible.

**Benchmark Model**

For Part 1 of the solution , I shall use PCA and K-means model predictors to generate cluster labels that identify the correlations between features that i shall then use on Part 2, to build a predication model to predict a response score for each customer.

While for Part 2 of the solution , I shall train a binary classifier into one of two groups of classes: target or No_target. I shall use Amazon SageMaker's LinearLearner. LinearLearner offers the hyper-parameter binary_classifier_model which is the model evaluation criteria for the training dataset, that also sorts out imbalanced training data.

**Evaluation Metrics**

To train according to specific product demands and goals, we do not want to optimise for accuracy only. Instead, we want to optimise for a metric that can help us decrease the number of false positives or negatives. In light of this, we want to build a model that has as many true positives and as few false negatives, as possible.

This corresponds to a model with a high recall: true positives / (true positives + false negatives). The matrix to use for the model shall be Recall.

I will assume that performance on a training set will be within about 5% of the performance on a test set. So, for a recall of about 85%, I'll aim for a bit higher, 90%.

In this section, propose at least one evaluation metric that can be used to quantify the performance of both the benchmark model and the solution model. The evaluation metric(s) you propose should be appropriate given the context of the data, the problem statement, and the intended solution. Describe how the evaluation metric(s) are derived and provide an example of their mathematical representations (if applicable). Complex evaluation metrics should be clearly defined and quantifiable (can be expressed in mathematical or logical terms).

**Project Design**

To implement Part 1 of the Solution problem; segmentation, i'll go through a number of steps:
1. Data loading and exploration
2. Data cleaning and pre-processing

3. Dimensionality reduction with PCA
4. Feature engineering and data transformation
5. Clustering transformed data with k-means
6. Extracting trained model attributes and visualising k clusters

To implement Part 2 of the Solution problem; predict potential customer, i'll go through a number of steps:

1. Upload data to s3
2. Defining and training a LinearLearner, binary classifier
3. Making improvements on the model
4. Evaluating and comparing model test performance

**Reference**

In this project i leveraged most of the materials from the Machine Learning Engineer Nanodegree classroom and student community at Udacity to complete the project. I am grateful to Udacity and Bertelsmann Arvato Analytics, for the datasets training for this project.

1. https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc
2. https://sagemaker.readthedocs.io/en/stable/linear_learner.html#sagemaker.LinearLearner
3. https://medium.com/@jaouad.eddadsi/finding-new-customers-using-data-and-machine-learning-algorithms-5da8bbeae798
4. https://medium.com/@venkateshrajagopalan86/customer-segmentation-for-arvato-financial-services-42ac87870b3c