I. Definition

Project Overview

It’s crucial for any business to classify the customers as of different segments by specific characteristics. By understanding the behavior and the consumption logics of each segment, a company can target the right consumer groups and make effective advertisement campaign.

In this project, we will use machine learning techniques to analyze demographics data of customers of a mail-order company in Germany. The goal of this project is to create customers segments of population, and to build a model to predict weather a customer will respond to future campaign.

Domain Background

Machine learning methodologies have widely applied in business analysis, such as analyzing customer data and finding insights and patterns. For example, Torizuka [1] used the Random Forest algorithm for segmentation and concluded that the algorithm recognizes data with high accuracy within the presence of noise and outliers. Ezenkwu, Ozuomba and Kalu [2] applied clustering algorithm to discover subtle but tactical patterns for a large unlabeled dataset. In this project, we will also apply artificially intelligent algorithms to analyze the attributes of existing clients and employ machine learning models to help identify new potential clients.

Datasets and Inputs

There are four data files associated with this project that are provided by Arvato:

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).

We noticed a significant imbalance in *Udacity_MAILOUT_052018_TRAIN.csv*. Over 98% of the response was labeled as zero. Under this circumstance, simply using accuracy as a metric might cause over-fitting towards the non-response data.

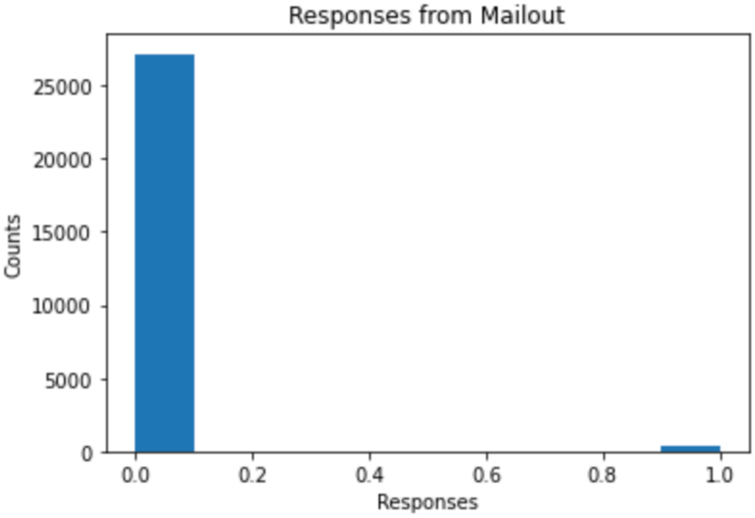


Figure 1: Distribution of mail-out responses

There are also two Excel spreadsheets, providing more information about the attributes in the data files:

DIAS Information Levels — Attributes 2017.xlsx: is a top-level list of attributes and descriptions, organized by the informational category.

DIAS Attributes — Values 2017.xlsx: is a detailed mapping of data values for each feature in alphabetical order.

Problem Statement

The main business questions for this project can be divided as follows:

- What are the segments of the general German population and customer population?
  We will implement unsupervised machine learning models to perform classification for the customers and the general population.
- How are these segments related to the Arvato customers database?
  By comparing the customer population to the general population, we could identify features that best describe the customers.
- What are the predictions of the targeted customers to the mail advertising campaign?
  A prediction model will then be created to identify individuals who are more likely to respond to the campaign.

Solution Statement

The solutions will be split into the following parts:

1. **Data Preprocessing**
   In this part, we need to perform exploratory data analysis for further analysis. Through EDA, we will be able to
   - Understand features and attributes of data
   - Handling missing values and outliners

   - Feature engineering
   - Data Transformation

2. **Consumer Segmentation Report**

   We first need to analyze general population (from `’Udacity_AZDIAS_052018.csv’`) and customers of mail-order company (from `’Udacity_CUSTOMERS_052018.csv’`). We will cluster the data using K-mean Algorithm to form segments of the population based on selected features. Since K-mean is exposed to the limitation of high-dimensional data, we will use PCA for dimensional reduction.

3. **Model Training**

   We will perform a number of classification algorithms first (e.g. Random Forest, XGBoost, Gradient Boosting, etc.), then compare their performances with predefined metrics (e.g. WSS, ROC, etc.) and select best-performing model to predict potential customers.

Evaluation Metrics

One of the most commonly seen metrics for unsupervised learning is Within-cluster sum of square (WSS). WSS calculates sum of squared distance between centroids and data points within each cluster.

Another metric we would like to apply is Area Under Curve (AUC) for receiver operating characteristic curve (ROC). The ROC curve shows the true positive rate against the false positive rate. Since the training data is highly imbalanced, simply using accuracy as a metric might cause over-fitting towards the non-response data. The ROC_AUC is often much more meaningful than accuracy for imbalanced classification problems.

II. Analysis

Data Exploration and Visualization

Overview of *AZDIAS* Dataset

- Int64Index: 37436 entries, 0 to 37435
- Columns: 366 entries, LNR to ALTERSKATEGORIE_GROB
- dtypes: float64(289), int64(71), object(6)
- memory usage: 104.8+ MB

Overview of *CUSTOMERS* Dataset

- Int64Index: 39545 entries, 0 to 39544
- Columns: 369 entries, LNR to ALTERSKATEGORIE_GROB
- dtypes: float64(276), int64(85), object(8)
- memory usage: 111.6+ MB

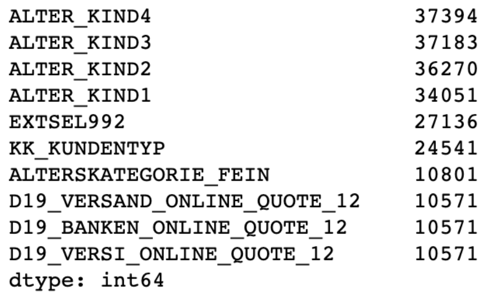
We can conclude following information for these two datasets:

- Column number mismatched

  There are 366 attributes in AZDIAS dataset and 369 attributes in CUSTOMERS Dataset. In order to ensure the consistency of our data sample, we need to remove the extra 3 columns in CUSTOMERS data.
- Mixed data types

  We can see that there are 3 different data types in both samples. We need to convert the object to the appropriate data types.
- Missing values

  We noticed that there are many missing values in both datasets. We listed top 10 attributes with most missing values.



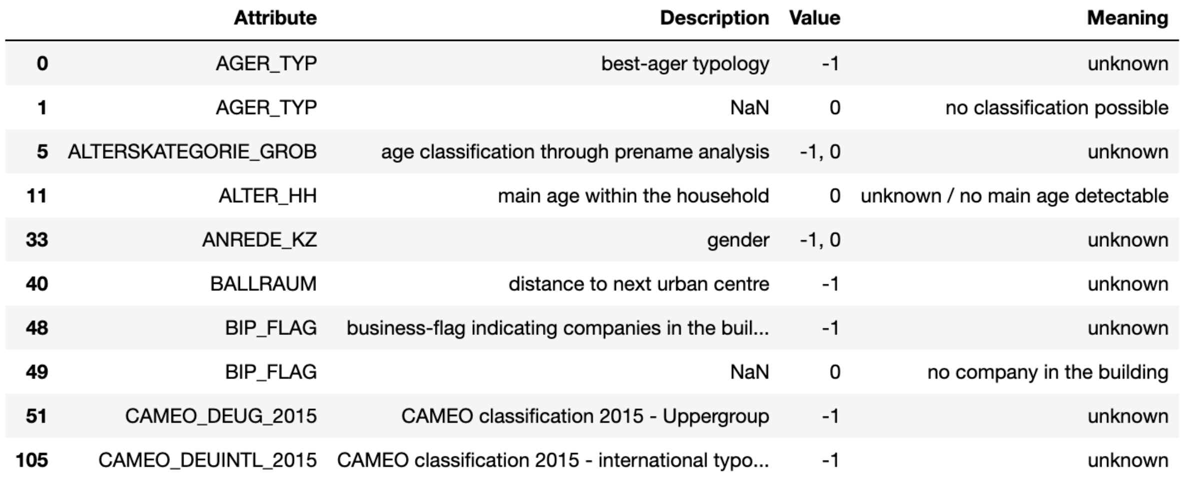
```
ALTER_KIND4                    37394
ALTER_KIND3                    37183
ALTER_KIND2                    36270
ALTER_KIND1                    34051
EXTSEL992                      27136
KK_KUNDENTYP                   24541
ALTERSKATEGORIE_FEIN           10801
D19_VERSAND_ONLINE_QUOTE_12    10571
D19_BANKEN_ONLINE_QUOTE_12     10571
D19_VERSI_ONLINE_QUOTE_12      10571
dtype: int64
```

Figure 2: Top 10 attributes with most NaNs (AZDIAS)

Missing data reduces the statistical power of the predictions. Removing attributes with a large proportion of missing values could improve the robustness of the prediction models.

- Unknown attributes

  We observed a large number of attributes with unknown or no values in *DIAS Attributes — Values 2017.xlsx.* Here are 10 examples of such attributes:



	Attribute	Description	Value	Meaning
0	AGER_TYP	best-ager typology	-1	unknown
1	AGER_TYP	NaN	0	no classification possible
5	ALTERSKATEGORIE_GROB	age classification through prename analysis	-1, 0	unknown
11	ALTER_HH	main age within the household	0	unknown / no main age detectable
33	ANREDE_KZ	gender	-1, 0	unknown
40	BALLRAUM	distance to next urban centre	-1	unknown
48	BIP_FLAG	business-flag indicating companies in the buil...	-1	unknown
49	BIP_FLAG	NaN	0	no company in the building
51	CAMEO_DEUG_2015	CAMEO classification 2015 - Uppergroup	-1	unknown
105	CAMEO_DEUINTL_2015	CAMEO classification 2015 - international typo...	-1	unknown

Figure 3: Attributes with unknown or no values

We need to convert such values with no meaning to NaNs to remove the statistical influence of these values over the final predictions.

Algorithms and Techniques

Customer Segmentation

Since there are 323 remaining attributes in both datasets, we will use PCA to perform dimension reduction before we implement K-means clustering. PCA is a method that only selects linearly independent variables which explain the most variability in the data. The number of components can be chosen based on the cumulative explained variance. In this project, I will choose based on 99% of explained variance to ensure a high level of accuracy. PCA could help us find the features that are of great importance and most relevant in segmenting data.

Then, we need to divide the data into different segments that share similar features. One of the most commonly used unsupervised classification model is K-means clustering. The idea of choosing the number of clusters is to have the minimal WSS. We will use Elbow graph to help us select an appropriate number of clusters that minimizes the WSS.

Customer prediction

In this part, we will implement several supervised learning models to identify customers that are most likely to response to the mail advertising campaign. This is a binary classification problem with expected predictions yes or no.

As there are many advanced classifiers we can choose. We will perform each one of them and select the best-performing classifier for this project. Some proposed models are:

- Gradient Boosting: a generalization of boosting to arbitrary differentiable loss functions. Gradient Boosting can be used for both regression and classification problems.
- XGBoost: derived from Gradient Boosting but it improved the efficiency of regularization objective function.
- Random Forest: combines several methods of classification.

- CatBoost: a relatively new open-source algorithm. Its core edges is the ability to handle a variety of data types.
- LightGBM: derived from Gradient Boosting. It is designed to have faster training speed and higher efficiency.

Benchmark Model

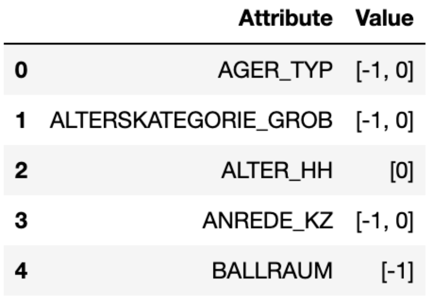
We selected the benchmark model to be the logistic regression model. This simple classification model will establish initial results upon which further improvements can be made to assess the relative improvements.

After fitting our data to the logistic regression model, we obtained an ROC_AUC score of 0.6762.

III. Methodology

Data Preprocessing

1. **Convert values with unknown meaning to NaNs**
   Understand the meanings of attributes are always the first step in data exploration. To remove the invalid data, we need to convert attribute values with unknown meaning or no classification possible to NaN values. We create a data frame that aggregates the name of the attributes and values with no or unknown meaning.



	Attribute	Value
0	AGER_TYP	[-1, 0]
1	ALTERSKATEGORIE_GROB	[-1, 0]
2	ALTER_HH	[0]
3	ANREDE_KZ	[-1, 0]
4	BALLRAUM	[-1]

Figure 4: example of attributes and values with unknown meaning

2. **Handle missing values**

   Then, we noticed that there are many missing values in *AZDIAS* and *CUSTOMERS* datasets. The final numbers of missing values were analyzed for each attribute. The analysis demonstrates that most of the attributes have empty values less than 30%. However, there are 41 attributes with NaNs more than 30%. We will remove these attributes from our datasets.

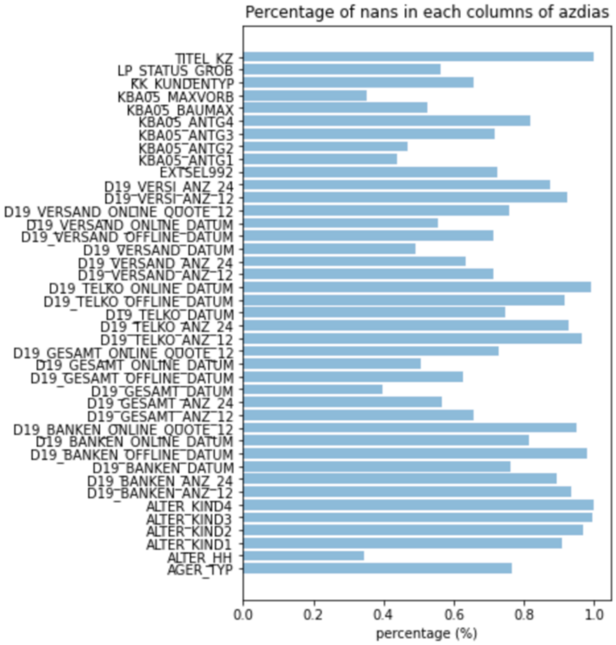


Figure 5: Attributes with more than 30% of missing values

We also studied the distribution of missing values in each row. According to the following figure, we can see that the majority of the distribution falls below 30%. Therefore, we will remove the rows containing missing values greater than 30%.

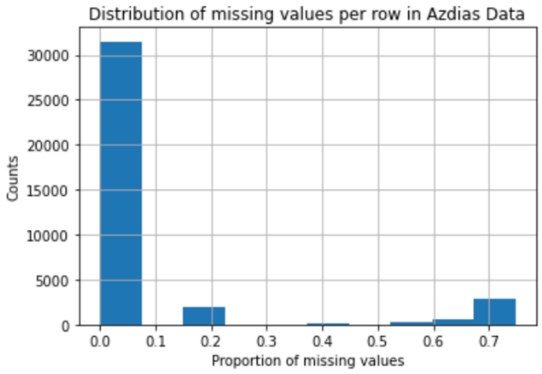


Figure 6: Distribution of missing values per row

3. **Convert data type**

   There are 6 attributes whose data types are object. We need to convert these data to the proper data type.

   - ‘CAMEO_DEUG_2015’, ‘CAMEO_INTL_2015’: reencode ‘X’ and ‘XX’ respectively to NaNs.
   - ‘OST_WEST_KZ’: reencode ‘O’ to 0 and ‘W’ to 1.
   - ‘CAMEO_DEU_2015’: reencode ‘1A’ to ‘9E’ with 1 to 43
   - Drop ‘EINGEFUEGT_AM’ and ‘D19_LETZTER_KAUF_BRANCHE’ since there no attribute meaning for these two features.

4. **Feature scaling and normalization**

   Before we are moving into unsupervised learning, we need to perform feature scaling to avoid tilting by outliners. In this step, all the remaining missing values will be filled with means. All values for each attribute will be standardized with a normal distribution.

Implementation

PCA

We plotted cumulative explained variance vs number of principal components for entire *AZDIAS* dataset.

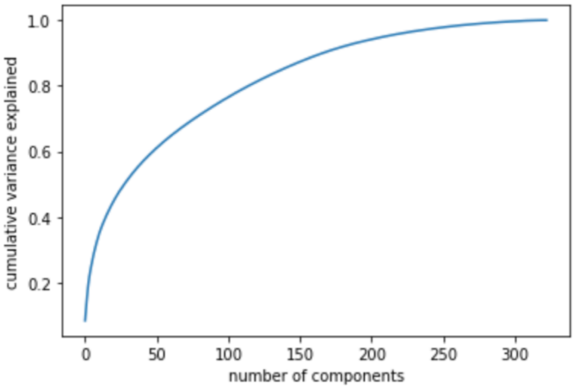


Figure 7: The explained variance of PCA components (AZDIAS)

We want to choose the number of components that can explain over 90% of the variance. After calculation, we decided to include first 166 components to our analysis which captures over 90% of the explained variance.

K-Means Clustering & Elbow Method

We implemented K-means clustering through 'sklearn'. By looping through 1 to 20, we created an Elbow graph of WSS versus number of clusters. We decided to include 8 clusters in total since additional clusters only slightly reduce the inertia of K-means.

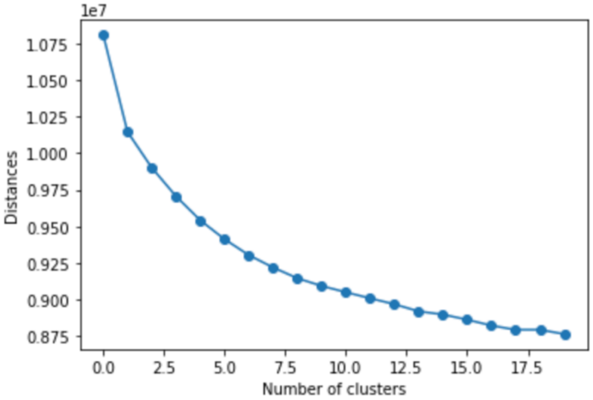


Figure 8: Elbow graph for K-means

We then applied the same procedures to CUSTOMERS dataset and compared the distributions of the two population.

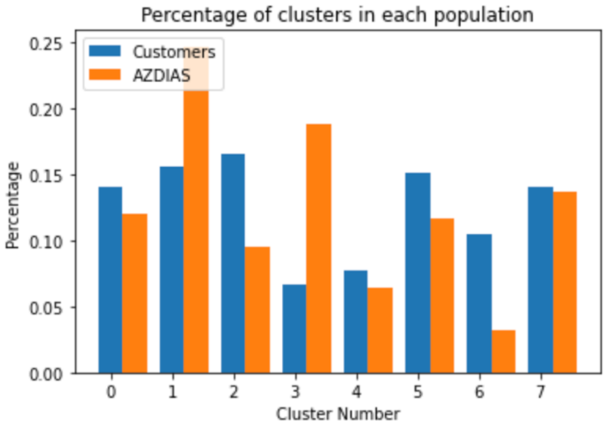


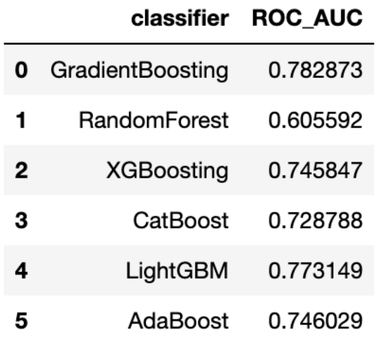
Figure 9: distribution of clusters in each population

From the graph above, we noticed the distributions of general population and customers population varies. In order to better understand the characteristics of customers population, we want to target some clusters that have a high proportion of customers over its entire population. There are a few of clusters worth attention from us.

For example, Cluster 2 has a relatively low percentage of general population but the highest percentage of customer population. We may conclude that individuals in Cluster 2 have a high probability to become our potential customers. On the other hand, Cluster 3 has the second highest proportion in general population, but lowest percentage in customer population. Therefore, we may want to ignore Cluster 3 for mailing campaign.

Classification

The final piece of the machine learning procedures is using classification model to predict which segments are most likely to become our future customers. The training data contains one additional column named ‘response’ while we need to predict it for the testing dataset. The performance of each classifier is shown as below.



	classifier	ROC_AUC
0	GradientBoosting	0.782873
1	RandomForest	0.605592
2	XGBoosting	0.745847
3	CatBoost	0.728788
4	LightGBM	0.773149
5	AdaBoost	0.746029

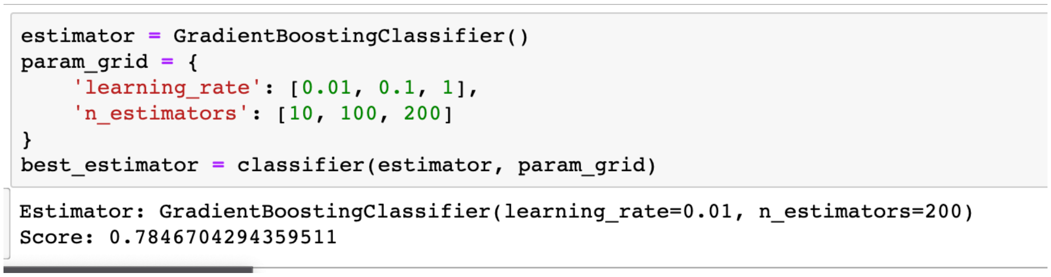
Figure 10: ROC_AUC scores for each classifier

Among the training model, Gradient Boosting has the highest ROC_AUC score of 0.7829 while LightGBM has the second highest score of 0.7731. The scores of these two classifiers outstand the remaining models.

Refinement

To test the general performance of classifiers on our data, we used the default hyperparameters. And now we would like to apply hyperparameter tuning to enhance the performance of predictions and to decide which model to use as our final classifier.

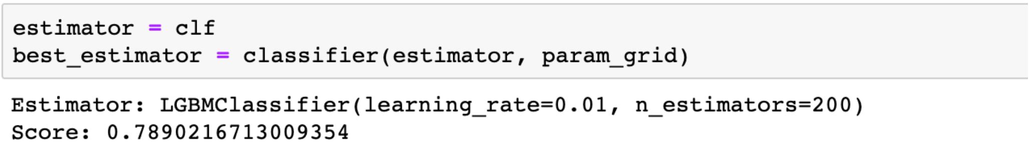
We used grid search, GridSearchCV, to perform hyperparameter tuning on Gradient Boosting and LightGBM.



```
estimator = GradientBoostingClassifier()
param_grid = {
    'learning_rate': [0.01, 0.1, 1],
    'n_estimators': [10, 100, 200]
}
best_estimator = classifier(estimator, param_grid)
```

```
Estimator: GradientBoostingClassifier(learning_rate=0.01, n_estimators=200)
Score: 0.7846704294359511
```

Figure 11: Best hyperparameters for Gradient Boosting



```
estimator = clf
best_estimator = classifier(estimator, param_grid)
```

```
Estimator: LGBMClassifier(learning_rate=0.01, n_estimators=200)
Score: 0.7890216713009354
```

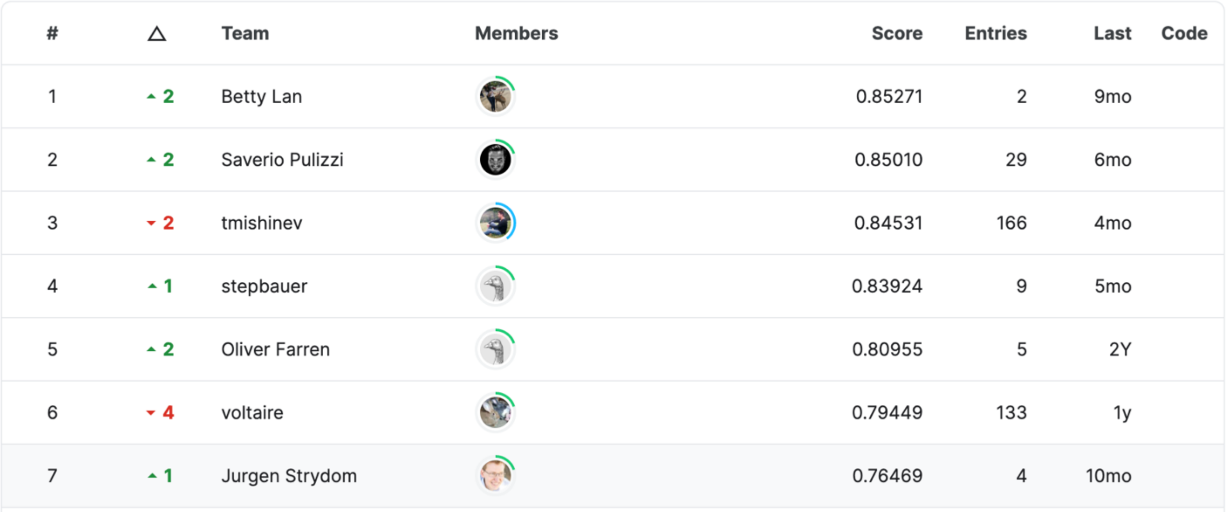
Figure 12: Best hyperparameters of LightGBM

After hyperparameter tuing, LightGBM has a higher ROC_AUC score of 0.789. The final classifier we chose is LightGBM.

IV. Results

Model Evaluation and Validation

Our model was finalized with a ROC_AUC score of 0.789. Comparing with the private Leaderboard in Kaggle. This score is ranked at about 8th place.



#	△	Team	Members	Score	Entries	Last	Code
1	▲ 2	Betty Lan		0.85271	2	9mo	
2	▲ 2	Saverio Pulizzi		0.85010	29	6mo	
3	▼ 2	tmishinev		0.84531	166	4mo	
4	▲ 1	stepbauer		0.83924	9	5mo	
5	▲ 2	Oliver Farren		0.80955	5	2Y	
6	▼ 4	voltaire		0.79449	133	1y	
7	▲ 1	Jurgen Strydom		0.76469	4	10mo	

Justification

Comparing with the benchmark classifier, the logistic regression model, the ROC_AUC score got substantial improvement. From 0.6762 to 0.789, the performance of the classifiers was improved by 16.68%. We can conclude that our final model is significant in searching potential customers.

V. Conclusion

In this project, we performed a thorough data analysis and machine learning modeling for a mail-order company in Germany. At the end of the project, we were able to predict whether or not individuals will respond to the mailing campaign. There are some interesting findings we would like to discuss:

- The data provided has a large dimension. With over 300 attributes, it requires extensive efforts to understand these data. For a future improvement, we can make a more detailed analysis to explore the attributes of the data. For example, are the attributes over-represented or under-represented the entire population?
- In this project, we compared the performance of 6 classifiers. The best performing model is with a ROC_AUC score of . A potential improvement can be made by exploring a better alternative for handling imbalanced data.
- The method we implement to handle missing values is dropping columns and rows with over 30% of the NaNs, and filling the remaining missing values with sample means. There are several alternatives we could explore: 1) test different threshold for dropping data; 2) fill the empty value with most frequent data or certain quantiles.