DATA SCIENCE INTERNSHIP - DATA GLACIER

Project: Bank Marketing (Campaign) -- Group Project

Group Name: **Datazoids**

Name: **Efe KARASIL - Sefa Sözer**

E-mail: **ekarasil@sabanciniv.edu - sefasozer9@gmail.com**

Country: **Turkey - Turkey**

College: **Sabancı University - Trakya University**

Specialization : **Data Science**

Problem Description:

ABC Bank wants to sell its term deposit product to customers and before launching the product they want to develop a model which helps them in understanding whether a particular customer will buy their product or not (based on customer's past interaction with bank or other Financial Institution).

Business Understanding:

Bank wants to use the ML model to shortlist customers whose chances of buying the product is more so that their marketing channel (telemarketing, SMS/email marketing etc) can focus only on those customers whose chances of buying the product is more.

This will save resources and their time (which is directly involved in the cost (resource billing)).

The data is related to direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed.

The classification goal is to predict if the client will subscribe (yes/no) to a term deposit (variable y).

Citation Request:

This dataset is publicly available for research. The details are described in [Moro et al., 2014].

Please include this citation if you plan to use this database:

[Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, In press, http://dx.doi.org/10.1016/j.dss.2014.03.001

Available at: [pdf] http://dx.doi.org/10.1016/j.dss.2014.03.001

[bib] http://www3.dsi.uminho.pt/pcortez/bib/2014-dss.txt

1. Title: Bank Marketing (with social/economic context)

2. Sources

Created by: Sérgio Moro (ISCTE-IUL), Paulo Cortez (Univ. Minho) and Paulo Rita (ISCTE-IUL) @ 2014

3. Past Usage:

The full dataset (bank-additional-full.csv) was described and analyzed in:

S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems (2014), doi:10.1016/j.dss.2014.03.001.

4. Relevant Information:

This dataset is based on the "Bank Marketing" UCI dataset (please check the description at: http://archive.ics.uci.edu/ml/datasets/Bank+Marketing).

The data is enriched by the addition of five new social and economic features/attributes (nationwide indicators from a ~10M population country), published by the Banco de Portugal and publicly available at: https://www.bportugal.pt/estatisticasweb.

This dataset is almost identical to the one used in [Moro et al., 2014] (it does not include all attributes due to privacy concerns).

Using the rminer package and R tool (http://cran.r-project.org/web/packages/rminer/), we found that the addition of the five new social and economic attributes (made available here) lead to substantial improvement in the prediction of a success, even when the duration of the call is not included. Note: the file can be read in R using:
d=read.table("bank-additional-full.csv",header=TRUE,sep=";")

The binary classification goal is to predict if the client will subscribe to a bank term deposit (variable y).

5. Number of Instances: 41188 for bank-additional-full.csv

6. Number of Attributes: 20 + output attribute.

7. Attribute information:

For more information, read [Moro et al., 2014].

Input variables:

bank client data:

1 - age (numeric)

2 - job : type of job (categorical: "admin.","blue-collar","entrepreneur","housemaid","management","retired","self-employed", "services","student","technician","unemployed","unknown")

3 - marital : marital status (categorical: "divorced","married","single","unknown"; note: "divorced" means divorced or widowed)

4 - education (categorical: "basic.4y","basic.6y","basic.9y","high.school","illiterate","professional.course","university.degree","unknown")

5 - default: has credit in default? (categorical: "no","yes","unknown")

6 - housing: has housing loan? (categorical: "no","yes","unknown")

7 - loan: has personal loan? (categorical: "no","yes","unknown")

related with the last contact of the current campaign:

8 - contact: contact communication type (categorical: "cellular","telephone")

9 - month: last contact month of year (categorical: "jan", "feb", "mar", ..., "nov", "dec")

10 - day_of_week: last contact day of the week (categorical: "mon","tue","wed","thu","fri")

11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y="no"). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus,

this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

 # other attributes:

12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)

13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)

14 - previous: number of contacts performed before this campaign and for this client (numeric)

15 - poutcome: outcome of the previous marketing campaign (categorical: "failure","nonexistent","success")

social and economic context attributes

16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)

17 - cons.price.idx: consumer price index - monthly indicator (numeric)

18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)

19 - euribor3m: euribor 3 month rate - daily indicator (numeric)

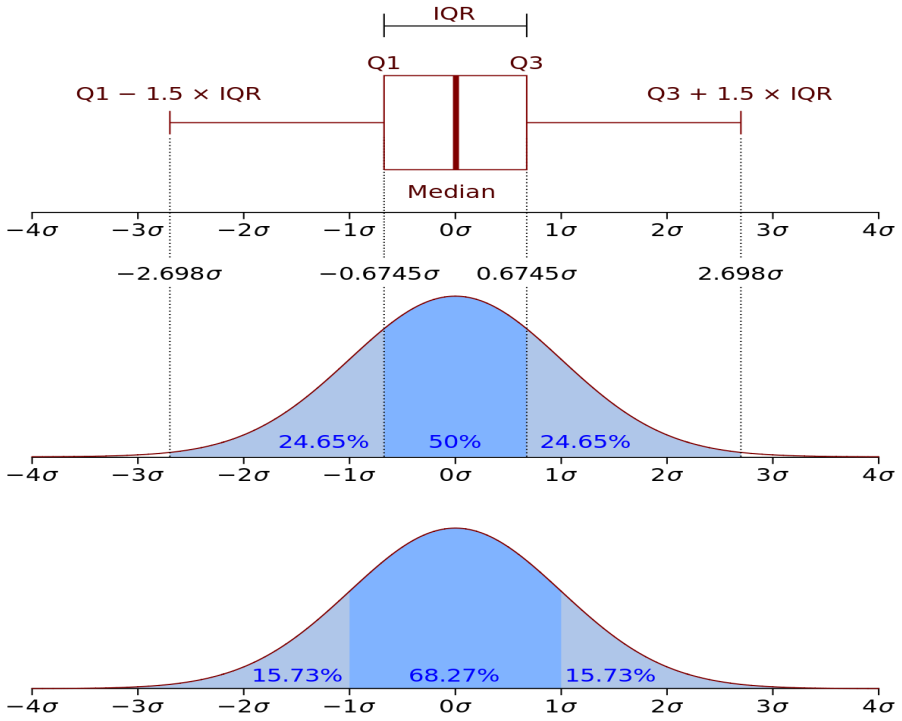
20 - nr.employed: number of employees - quarterly indicator (numeric)

Output variable (desired target):

21 - y - has the client subscribed a term deposit? (binary: "yes","no")

8. Missing Attribute Values: There are several missing values in some categorical attributes, all coded with the "unknown" label. These missing values can be treated as a possible class label or using deletion or imputation techniques.

Outlier Detection and Handling



IQR is used to measure variability by dividing a data set into quartiles. The data is sorted in ascending order and split into 4 equal parts. Q1, Q2, Q3 called first, second and third quartiles are the values which separate the 4 equal parts.

- Q1 represents the 25th percentile of the data.
- Q2 represents the 50th percentile of the data.
- Q3 represents the 75th percentile of the data.

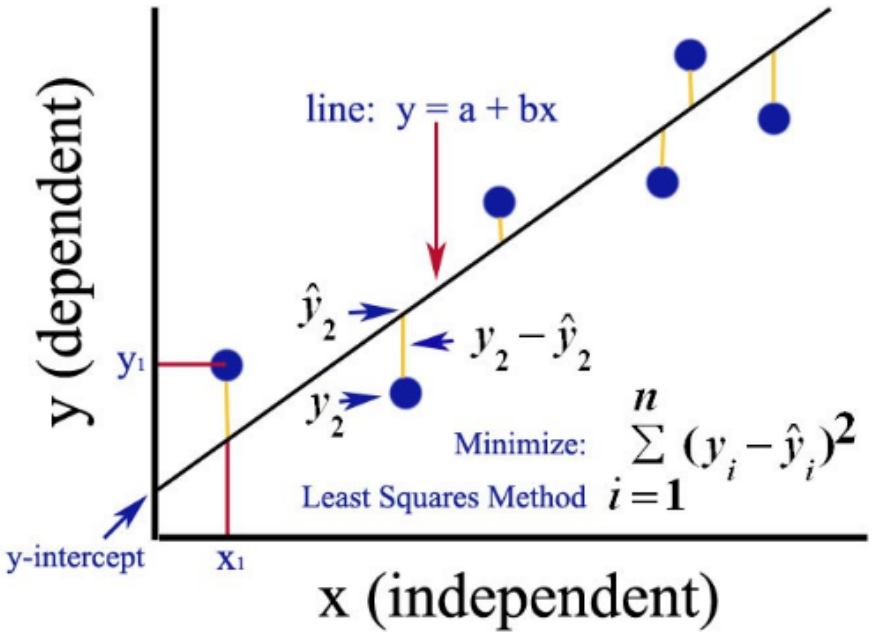
IQR(The interquartile range) method was used for outlier detection and handling. The interquartile range method defines outliers as values larger than Q3 + 1.5 * IQR or the values smaller than Q1 – 1.5 * IQR. Outlier detection and handling was used for numerical values.

Feature Importance

OLS method was used for feature importance and deleting the columns-features from the dataset for better results in ML models. The ordinary least squares (OLS) method can be defined as a linear regression technique that is used to estimate the unknown parameters in a model. The method relies on minimizing the sum of squared residuals between the actual and predicted values. The residual can be defined as the difference between the actual value and the predicted value. Another word for residual can be error. In mathematical terms, this can be written as:

Minimize ∑(yi – ŷi)^2

where yi is the actual value, ŷi is the predicted value. A simple linear regression model used for determining the value of the response variable, ŷ, can be represented as the following equation. Note the method discussed in this blog can as well be applied to multivariate linear regression models.



ŷ = bX + a

where ŷ is the predicted value, a is the intercept, and b is the slope of the line. The coefficients b and a can also be called the coefficients of determination. The OLS method can be used to estimate the unknown parameters (b and a) by minimizing the sum of squared residuals. In other words, the OLS method finds the best-fit line for the data by minimizing the sum of squared errors or residuals between the actual and predicted values. The sum of squared residuals is also termed the sum of squared error (SSE).

This method is also known as the least-squares method for regression or linear regression.

At the end of this method, **'Age', 'Job', 'Default', 'Housing'** and **'Campaign'** columns-features have been dropped.

Preparation summary before Machine Learning models

- Getting rid of outliers with IQR method from numerical features
    - Actually, both data with outliers and without outliers were tried to be used in models for comparison. The original data consists of the information of approximately 40 thousand customers, and after the outliers in all the data are processed, the information of approximately 30 thousand customers remains.
    - After both training data and prediction of the model obtained from training data with test data, there is a significant increase of 5% in the accuracy scores when the outliers were excluded. For example, from 80% to 85 in training data, from 90 to 95 in test data.

  - If the company wants to minimize cost, getting rid of outliers before modeling can be a better idea, because getting to more customers while having less accuracy in predictions can be high cost in time, money, data processing etc.

- Leaving 'unknown' values as they are in the categorical features to be processed as another category for their specific column-feature
- Using encoder to make categorical values numerical to prepare for machine learning models
- Using the ordinary least squares (OLS) method for feature selection, elimination
- Dropping "Duration" feature which should not be known beforehand to predict desired target "y" feature
  - Since the main target is to predict if a customer wants to subscribe for the bank term deposit or not, and contact that customer according to that, it can be understood that the bank did not contact the customer and do not have the "duration" data yet.

Model Building

- ☐ It was thought to be that classification models provide better results, since our desired output column for the machine learning models is a categorical value
- ☐ Therefore, we used classification models.
- ☐ One of the methods was splitting the data to train-valid-test to evaluate the chosen models and prevent data leak by training the model and evaluating it using the training and validating sets.
- ☐ Cross validation was used to evaluate the model, or hyperparameter, the model has to be trained from scratch, each time, without reusing the training result from previous attempts. The result of the cross validation gives us the optimized model.
- ☐ We made the learning data 75% the test data 25%. Then we tried the following

models on these data.

- ☐ Multiple classification metrics were utilized to examine the model. It included accuracy,Mean squared error and ROC-AUC.

The following algorithms selected for this classification problem include:

- **Linear Algorithms:**
  - ❖ **Logistic Regression (LR) (Base Model)**
  - ❖ **Linear Discriminant Analysis (LDA).**

- **Nonlinear Algorithms:**
  - ❖ **Classification and Regression Trees (CART)**
  - ❖ **Gaussian Naive Bayes (NB)**
  - ❖ **k-Nearest Neighbors (KNN)**

- **Ensemble Methods:**

***Boosting Methods:**

- ❖ **AdaBoost (AB)**
- ❖ **Gradient Boosting (GBM)**
- ❖ **XGBClassifier(XGB)**

***Bagging Methods:**

- ❖ **Random Forests (RF)**
- ❖ **Extra Trees (ET)**

- ❖ After a nice preprocessing of the data, we embed it into machine learning models. We have tried all branches of machine learning and observed that ensemble models give better results than Linear and Non-Linear models. Among the Ensemble models, the XGBoost, Adaboost and Gradient Boosting models are very close in accuracy metrics.

Here are the results of the machine learning models we used:

Ensemble Methods:

AdaBoost (AB):	**0.860 (0.021)**
Gradient Boosting (GBM):	**0.863 (0.020)**
Random Forests (RF):	**0.813 (0.025)**
Extra Trees (ET):	**0.753 (0.023)**
XGBClassifier(XGB):	**0.864 (0.021)**
CatBoost(CB):	**0.846 (0.014)**

Linear and Non-Linear Models:

Logistic Regression(LR):	**0.825(0.023)**
Linear Discriminant Analysis (LDA):	**0.830 (0.020)**
k-Nearest Neighbors (KNN):	**0.773 (0.024)**
Classification and Regression Trees (CART):	**0.683(0.026)**
Gaussian Naive Bayes (NB):	**0.835 (0.021)**

These results are obtained with k-fold(10) validation of **training data, the average and standard deviation of scores are given.

Applying Models

Almost every model was also subjected to Bayesian optimization methods while tuning their hyperparameters.

Bayesian optimization:

Bayesian optimization is a global optimization method for noisy black-box functions. Applied to hyperparameter optimization, Bayesian optimization builds a probabilistic model of the function mapping from hyperparameter values to the objective evaluated on a validation set. By iteratively evaluating a promising hyperparameter configuration based on the current model, and then updating it, Bayesian optimization aims to gather observations revealing as much information as possible about this function and, in particular, the location of the optimum. It tries to balance exploration (hyperparameters for which the outcome is most uncertain) and exploitation (hyperparameters expected close to the optimum). In practice, Bayesian optimization has been shown to obtain better results in fewer evaluations compared to grid search and random search, due to the ability to reason about the quality of experiments before they are run.

Making Prediction with models and testing:

Although Linear and Non-Linear models were much less successful in training data when compared to Ensemble models, predictions(with machine learning) for these models were also made. Then, compared with the test data.

	Accuracy score	Mean squared error
Logistic Regression(LR):	0.944	(0.055)
Linear Discriminant Analysis (LDA):	0.930	(0.069)
k-Nearest Neighbors (KNN):	0.941	(0.058)
Classification and Regression Trees (CART):	0.939	(0.060)
Gaussian Naive Bayes (NB):	0.816	(0.184)

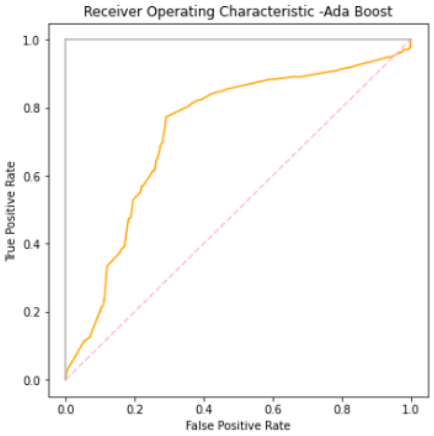
After Linear and Non-Linear models, Ensemble Methods used for testing models with predictions.

	Accuracy score	Mean squared error
AdaBoost (AB):	0.946	(0.053)
Gradient Boosting (GBM):	0.947	(0.053)
Random Forests (RF):	0.943	(0.057)
Extra Trees (ET):	0.941	(0.058)
XGBClassifier(XGB):	0.947	(0.053)
CatBoost(CB):	0.944	(0.055)

Model predictions and their score after evaluated with test data

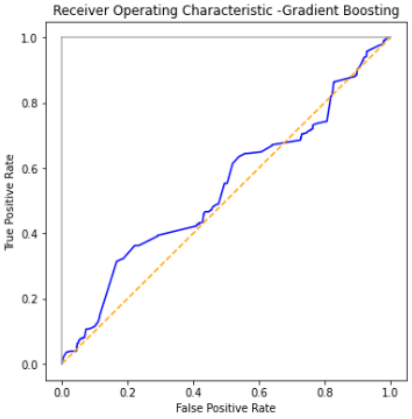
Ada Boost - Default

Area under the ROC curve: 0.726



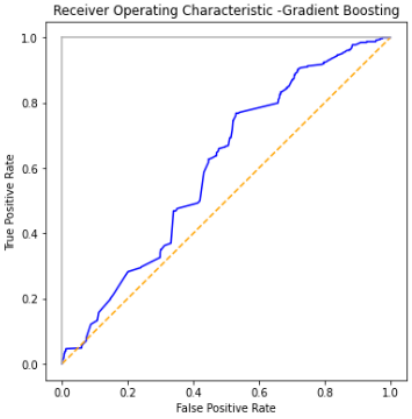
Ada Boost - After hyperparameter tuning

Area under the ROC curve: 0.731



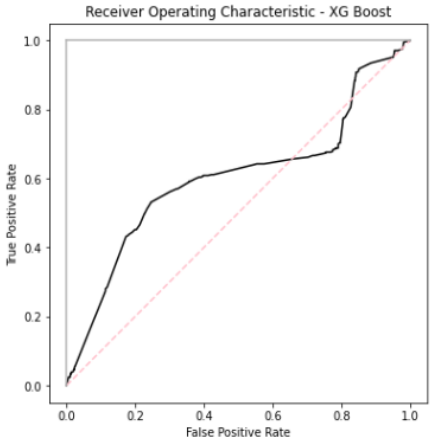
Gradient Boosting - Default

Area under the ROC curve: 0.538



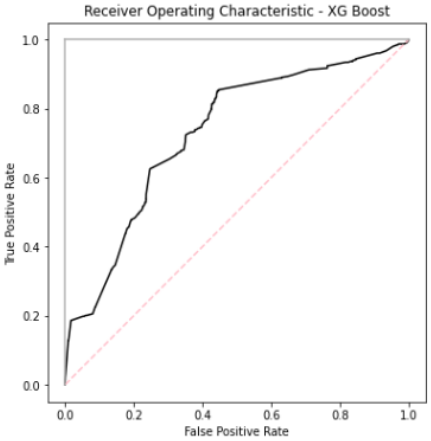
Gradient Boosting- After hyperparameter tuning

Area under the ROC curve: 0.603



XG Boost - Default

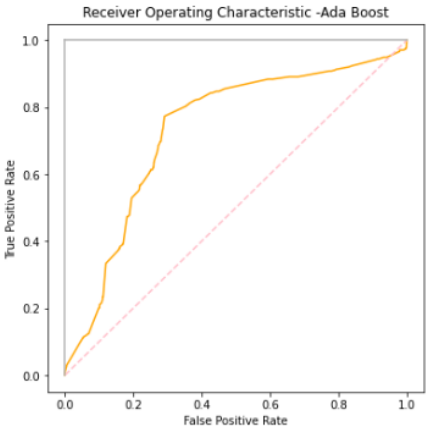
Area under the ROC curve: 0.601



XG Boost - After hyperparameter tuning

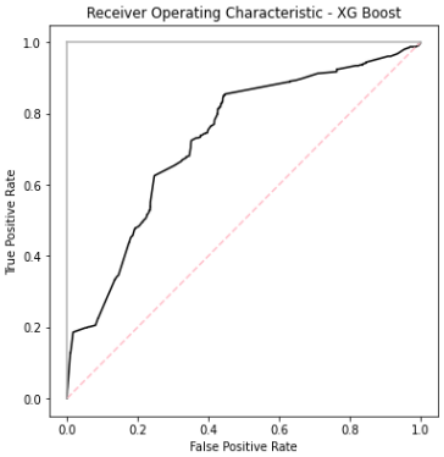
Area under the ROC curve: 0.728

When we look at the 3 model ROC curves we can see that Gradient Boosting is not as efficient as the other two(XG Boost, ADA Boost). However, both XG Boost and ADA Boost has very similar and good results in the final as well. So, time efficiency of these two models can be used for choosing the best one.



Ada Boost - After hyperparameter tuning

Area under the ROC curve: 0.731



XG Boost - After hyperparameter tuning

Area under the ROC curve: 0.728

Time comparison:

	AdaBoost (AB)	XGBClassifier(XGB)
The spent time for measuring the accuracies for training data process(with k-fold validation learning) :	47 sec	31 sec
The spent time for measuring the accuracies, fitting the models, prediction and comparing with test data:	5 sec	3 sec

Final Model Selection

After all these steps, XG Boost was the model that gave the best prediction results in a fair time. In most evaluations throughout the whole process this model was the one of the top ones or the one that is the best. However, it should be noted that XG Boost with the hyperparameters optimized with Bayesian optimization, not with the default parameters made the model more effective and correct.