Javier Echavarri Trillo, 205574
Arne Berresheim, 230829

REPORT PROJECT 2

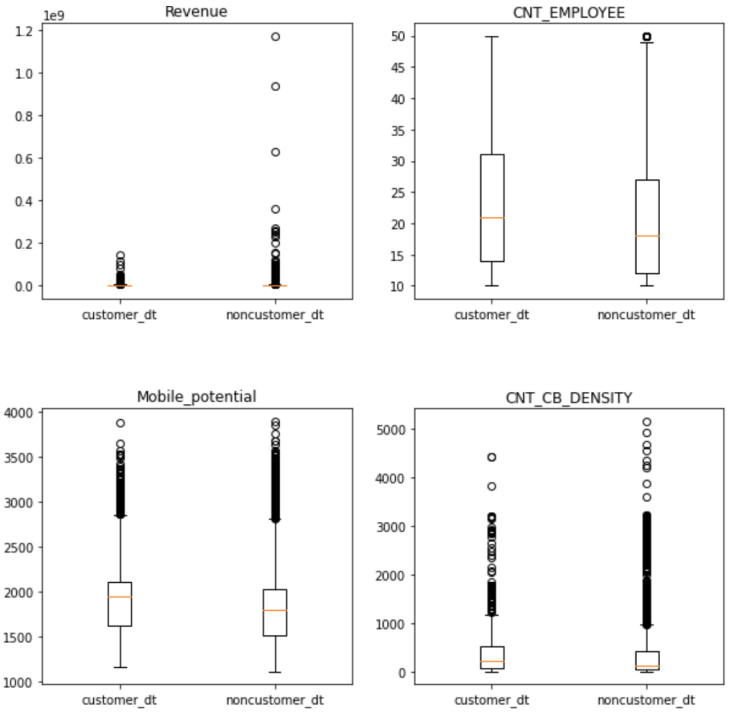
EX1:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13335 entries, 0 to 13334
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   City                13335 non-null  object
 1   Customer_Flag       13335 non-null  int64
 2   Revenue             8589 non-null   float64
 3   [illegible]         non-null        float64
 4   Legal_Form_Code     13229 non-null  float64
 5   CNT_EMPLOYEE        13335 non-null  int64
 6   CNT_CB_DENSITY      10265 non-null  float64
 7   CNT_CB_MOB_DENSITY  10265 non-null  float64
 8   CNT_CB_FN_DENSITY   10265 non-null  float64
 9   Mobile_potential    13335 non-null  float64
dtypes: float64(7), int64(2), object(1)
memory usage: 1.0+ MB
```

click to scroll output; double click to hide

These are the datatypes of the dataset. We can see that the variable with the most nulls is Revenue and the ones with less are City, Customer_Flag, Sector, CNT_EMPLOYEE, and mobile potential. CIty is an object of type string.

EX3:

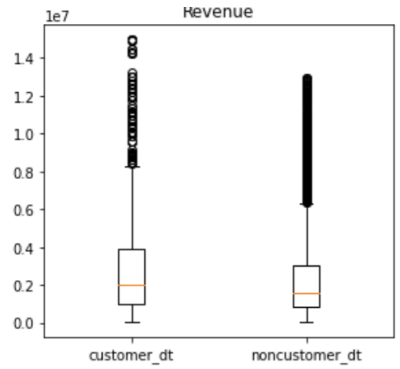


We can see that the revenue is basically the same with the exception of more outliers for non_customer_dt. On CNT_EMPLOYEE the only differences are the outlier and a lower median and quartiles for non_customer_dt. On Mobile_potential lower quartiles and median for non_customer_dt (both have a lot of outliers). On CNT_CB_DENSITY both have outliers but median and quartiles are lower on non_customer_dt.

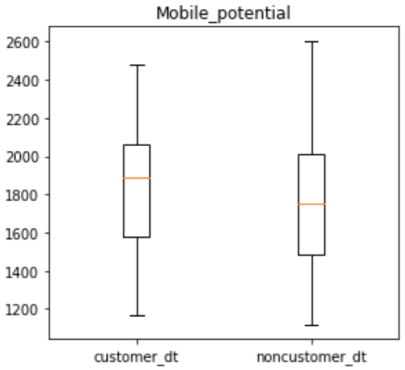
The quartiles for **Mobile_potential** on **customer_dt** are 1621.055686, 1948.437661, 2116.474074
The quartiles for **Mobile_potential** on **noncustomer_dt** are 1513.383597, 1797.054278, 2035.082840
The quartiles for **Revenue** on **customer_dt** are 1047500.0, 2200000.0, 4195000.0
The quartiles for **Revenue** on **noncustomer_dt** are 902986.0, 1750000.0, 3501123.5

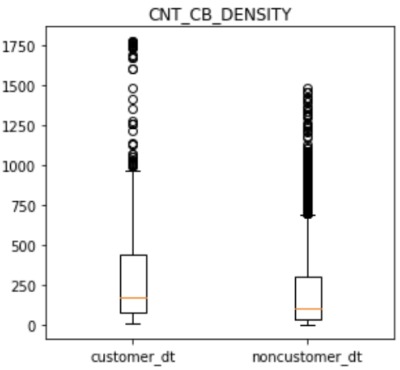
Javier Echavarri Trillo, 205574
Arne Berresheim, 230829

EX4:



We can see that maximums are lower and the minimums are higher, so eliminating some datapoints has narrowed the values and reduced the number of outliers.



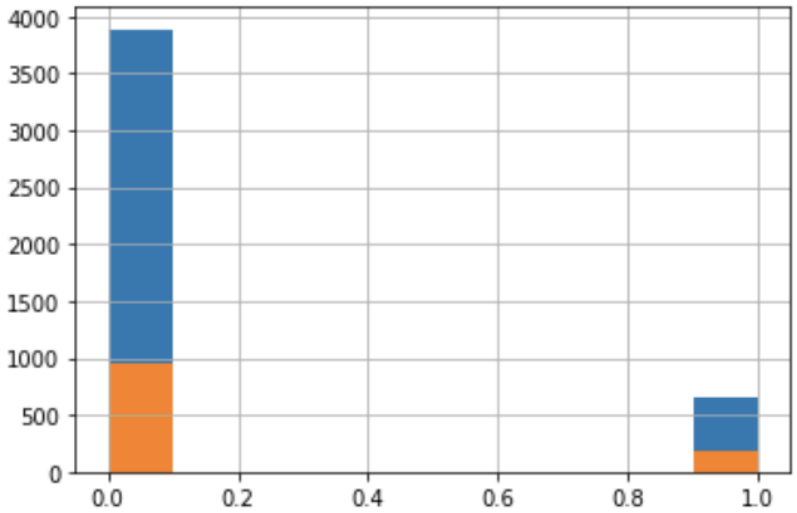


EX6:

length of X_train: 4536
length of X_test: 1135

It is clear that the length of the X_train dataset is an 80% of the full length dataset which is 6571.

EX7:



They are imbalanced as we see on the graph. If we want to improve the quality of the classification it needs to be more balanced.

.

Javier Echavarri Trillo, 205574
Arne Berresheim, 230829

EX12:

```
##################### SVC #####################

###### TEST DATASET ######
Accuracy:  0.49404761904761907

Confusion Matrix :
 [[100  65]
 [105  66]]

Classification Report:
              precision    recall  f1-score   support

           0       0.49      0.61      0.54       165
           1       0.50      0.39      0.44       171

    accuracy                           0.49       336
   macro avg       0.50      0.50      0.49       336
weighted avg       0.50      0.49      0.49       336
```

```
##################### DECISION TREE #####################

###### TEST DATASET ######
Accuracy:  0.5357142857142857

Confusion Matrix :
 [[ 77  88]
 [ 68 103]]

Classification Report:
              precision    recall  f1-score   support

           0       0.53      0.47      0.50       165
           1       0.54      0.60      0.57       171

    accuracy                           0.54       336
   macro avg       0.54      0.53      0.53       336
weighted avg       0.54      0.54      0.53       336
```

The decision tree classifier has better accuracy on both classes by a little margin. The recall is also higher in the decision tree. Generally we would use the decision tree, because it has higher accuracy and recall.

EX13:

```
############### Voting Ensemble ###############
Accuraccy:  0.5238095238095238

Confusion Matrix:
  [[132  33]
 [127  44]]

Classification Report:
              precision    recall  f1-score   support

           0       0.51      0.80      0.62       165
           1       0.57      0.26      0.35       171

    accuracy                           0.52       336
   macro avg       0.54      0.53      0.49       336
weighted avg       0.54      0.52      0.49       336
```

When wanting to predict the class 0 this model scores a lot better than the other two, because of its high recall, which is 80%. On the other hand though if we look at the recall of class 1 (26%) this is only a better model than the Decision Tree if you are only predicting for class 0.

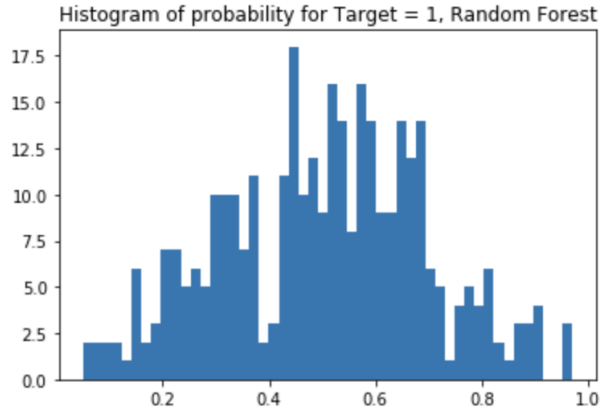
EX14:

```
############### Random Forest ###############
Accuraccy:  0.625

Confusion Matrix:
  [[101  64]
 [ 62 109]]

Classification Report:
              precision    recall  f1-score   support

           0       0.62      0.61      0.62       165
           1       0.63      0.64      0.63       171

    accuracy                           0.62       336
   macro avg       0.62      0.62      0.62       336
weighted avg       0.62      0.62      0.62       336
```

This model is overall the best of all that we have seen so far, because of high precision and recall for both classes.

Javier Echavarri Trillo, 205574
Arne Berresheim, 230829

EX15:

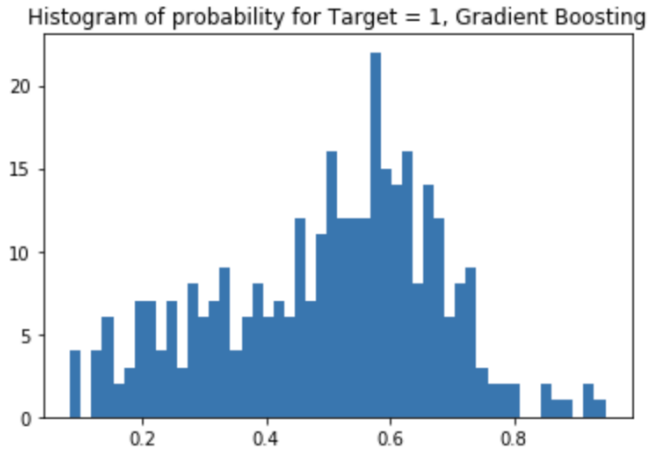


EX16:

```
############### Gradient Boosting ###############
Accuracy:  0.6190476190476191

Confusion Matrix:
  [[ 91  74]
 [ 54 117]]

Classification Report:
                precision    recall  f1-score   support

           0       0.63      0.55      0.59       165
           1       0.61      0.68      0.65       171

    accuracy                           0.62       336
   macro avg       0.62      0.62      0.62       336
weighted avg       0.62      0.62      0.62       336
```

This model has similar precision for the two classes in comparison to the random forest. The recall is higher for class 1, but lower for class 0. Overall we decided that the Random Forest is better, because it has an overall higher score.

EX17:



If we have a look at the two histograms we can see that the one of the Random Forest has a lot of values around 0.5 without a clear distinction between the two classes. This is better visible in the histogram of the Gradient Boosting. Here we can see a clear drop in numbers at about 0.46. Therefore we will choose the Gradient Boosting model.

Javier Echavarri Trillo, 205574
Arne Berresheim, 230829

EX18:

According to our observations from the exercise before and some experimenting with the value we changed the cutoff to 0.46. We will send them 84 customers as this is the value of falsely predicted customers according to our model. Meaning that these are the non customers that according to our model could be customers.

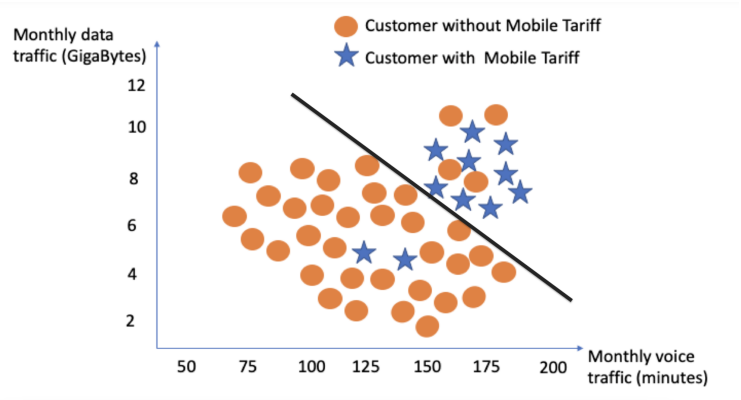
EX19:

```
[['Mobile_potential', 0.18877197405200938],
 ['CNT_CB_DENSITY', 0.15649270078402414],
 ['City_coded', 0.13243986573475086],
 ['CNT_CB_FN_DENS', 0.11602890774407222],
 ['Sector', 0.10179753573398068],
 ['Revenue', 0.0978701939251656],
 ['Legal_Form_Code', 0.0773243578733948],
 ['CNT_EMPLOYEE', 0.07496209220217445],
 ['CNT_CB_MOB_DENSITY', 0.05431237195042798]]
```

Here we can see the top 3 important features are ‘Mobile Potential’, ‘CNT_CB_DENSITY’ and ‘City_coded’.

EX20:

- Target = 1 is for the customers with mobile tariffs and Target = 0 for the customers without a mobile tariff
- Yes, we would add these variables, because with them we could see if the customers with a tariff actually have more data and voice consumption and whether their expenses are higher. Also it is easier to classify customers without a mobile tariff that have high cost and high usage for whom it would make sense to get a mobile tariff.
- Unbalanced, as the tariff is not very popular, meaning that there will be more customers without a mobile tariff than with one.
- The pattern that we can detect here is that Target = 1 customers have the highest monthly voice traffic (150-200) and also the highest monthly data traffic (6-11). There are also some outliers which are not in this area
- .



- The customers to be called are the ones which are above the plane. There are four orange data points in that area
- These are calculated from the data points with the plane that we can see above.

	precision	recall
0	0.94	0.88
1	0.69	0.82

Javier Echavarri Trillo, 205574
Arne Berresheim, 230829

We hereby declare that, except for the code provided by the course instructors, all of our code, report, and figures were produced by ourselves