Skoltech

Skolkovo Institute of Science and Technology

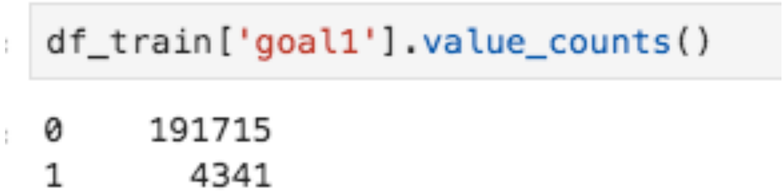
Introduction to DC

Team members:
Kundyz Onlabek, Ekaterina Zharkova, Elena Nazarova

Using data science methods, our team wants to solve the problem of one European company OneTwoTrip, namely, to predict the likelihood of a flight ticket return and determine the likelihood of a customer purchasing an additional service. This model can help the company:

- Predict how many people will be on board the aircraft - calculate the aircraft load
- Reduce costs - if you predict that a person will hand over sick, then you can sell more tickets than seats on the plane
- Develop a competent financial model of the company

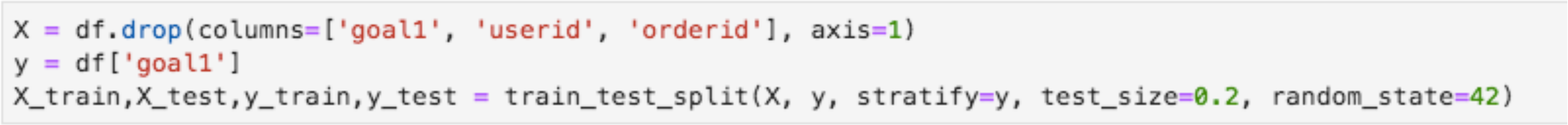
We see that there are a lot of 0 and few units of data in "goal1"



```
df_train['goal1'].value_counts()
```

```
0    191715
1      4341
```

Therefore, we split the data into split and trein using "stratification"

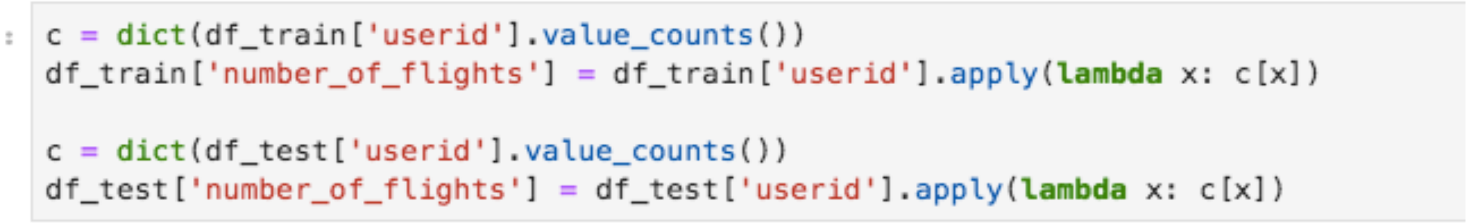


```
X = df.drop(columns=['goal1', 'userid', 'orderid'], axis=1)
y = df['goal1']
X_train,X_test,y_train,y_test = train_test_split(X, y, stratify=y, test_size=0.2, random_state=42)
```

Next, we count the models using weight = "balanced":

Model	Roc-auc result
Logistic regression	0.664 ± 0.012
Random forest classifier	0.653 ± 0.013
Decision tree classifier	0.606 ± 0.011
XGBoost	0.683 ± 0.011

Next, we decided to guess the features and add new features



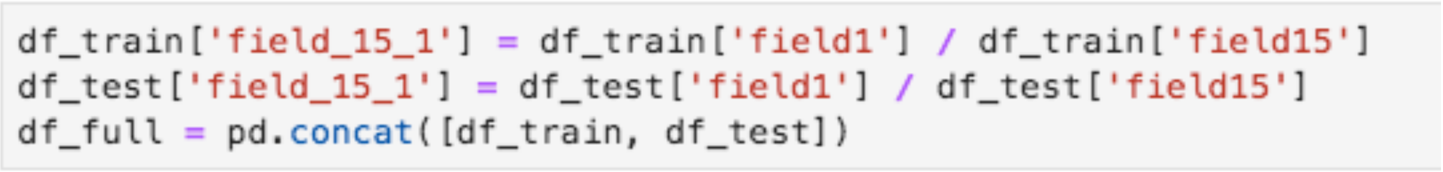
```
c = dict(df_train['userid'].value_counts())
df_train['number_of_flights'] = df_train['userid'].apply(lambda x: c[x])

c = dict(df_test['userid'].value_counts())
df_test['number_of_flights'] = df_test['userid'].apply(lambda x: c[x])
```

"field4" is most likely the order number or count of number

"field5" is a sign of the first order

'field1' is the price

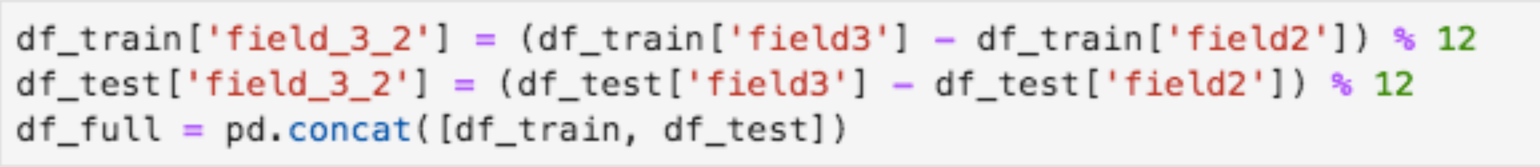
'field15' is the number of tickets purchased by the user



```
df_train['field_15_1'] = df_train['field1'] / df_train['field15']
df_test['field_15_1'] = df_test['field1'] / df_test['field15']
df_full = pd.concat([df_train, df_test])
```

"field2" is the month of purchase

"field3" is the month of departure of the aircraft

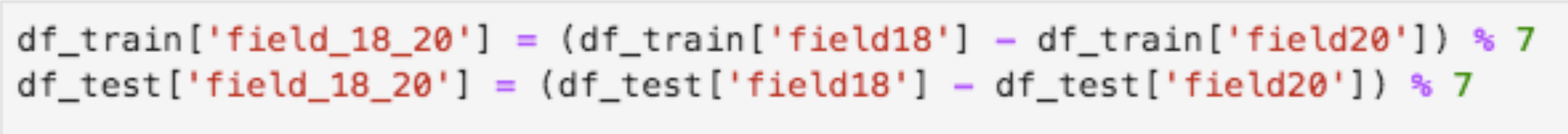


```
df_train['field_3_2'] = (df_train['field3'] - df_train['field2']) % 12
df_test['field_3_2'] = (df_test['field3'] - df_test['field2']) % 12
df_full = pd.concat([df_train, df_test])
```

"field16" is the difference between two dates (days)

"field20" is the day of the week on which the plane departs

"field20" is the day of the week of purchase



```
df_train['field_18_20'] = (df_train['field18'] - df_train['field20']) % 7
df_test['field_18_20'] = (df_test['field18'] - df_test['field20']) % 7
```

Next, we once again decided to apply the models, but using Grid Search

Model	Roc-auc result
Logistic regression	0.663 ± 0.012
Random forest classifier	0.624 ± 0.009
Decision tree classifier	0.555 ± 0.003
XGBoost	0.613 ± 0.005