# KaggleGarantiBBVA

January 23, 2021

[1]:

```
import csv
from sklearn.ensemble import RandomForestClassifier
import pandas as pd
#from pylab import *
#import numpy as np
from sklearn.model_selection import train_test_split
#import sys
from imblearn.over_sampling import SMOTE
from sklearn.ensemble import RandomForestRegressor

from sklearn.feature_selection import SelectFromModel
import numpy as np
from sklearn import metrics
```

[2]:

```
df1=pd.read_csv('C:/Users/Asus/KaggleProject1github/train.csv')
```

[3]:

```
X=df1.drop(['f2'], axis='columns')
X['f2']=pd.read_csv('C:/Users/Asus/KaggleProject1github/datesf.csv')
```

[4]:

```
learn=X.drop(['f3'], axis='columns')
find=X['f3']
learn=learn.fillna(0)
learn=learn.drop(['f1'], axis='columns')
```

[5]:

```
df2=pd.read_csv('C:/Users/Asus/KaggleProject1github/test.csv')
test=df2.drop(['f2'], axis='columns')
test['f2']=391


test=test.fillna(0)
test=test.drop(['f1'],axis='columns')

learn=learn.drop(['f29'], axis='columns')
```

[6]:

```
clf=RandomForestRegressor(n_estimators=50)

clf.fit(learn,find)
```

```
[6]: RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
                          max_depth=None, max_features='auto', max_leaf_nodes=None,
                          max_samples=None, min_impurity_decrease=0.0,
                          min_impurity_split=None, min_samples_leaf=1,
                          min_samples_split=2, min_weight_fraction_leaf=0.0,
                          n_estimators=50, n_jobs=None, oob_score=False,
                          random_state=None, verbose=0, warm_start=False)
```

```
[7]: #print(test.shape)
     #print(learn.shape)
```

```
[8]: y_pred = clf.predict(test)
```

```
[9]: result=pd.DataFrame()
     result['Id']=df2['f1'].values
     result['Predicted']=pd.DataFrame(y_pred)
     df=result
     df=df.sort_values('Id')
```

```
[10]: import tkinter as tk
      from tkinter import filedialog
      from pandas import DataFrame
      root= tk.Tk()

      canvas1 = tk.Canvas(root, width = 300, height = 300, bg = 'lightsteelblue2',␣
       ↪relief = 'raised')
      canvas1.pack()

      def exportCSV ():
          global df

          export_file_path = filedialog.asksaveasfilename(defaultextension='.csv')
          df.to_csv (export_file_path, index = False, header=True)

      saveAsButton_CSV = tk.Button(text='Export CSV', command=exportCSV, bg='green',␣
       ↪fg='white', font=('helvetica', 12, 'bold'))
      canvas1.create_window(150, 150, window=saveAsButton_CSV)

      root.mainloop()
```

```
[ ]:
```

```
[ ]:
```

```
[11]: # print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test,␣
       ↪y_pred)))
```

```
#we used in testing but didn't used it in the end due to having actual test␣
 ↪data.
```

[ ]:

```
```

[12]:

```
a=X['f1']
```

[13]:

```
array=[]
for i in a:
    array.append(i)
array.sort()
```

[14]:

```
#X.info()
```

[15]:

```
df2=pd.read_csv('C:/Users/Asus/KaggleProject1github/test.csv')
df2=pd.DataFrame(df2)
#df2.info()
```

[16]:

```
a=df2['f1']
```

[17]:

```
array2=[]
for i in a:
    array2.append(i)
array2.sort()
```

[ ]:

```
```