Classification on a diabetic dataset

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
```

Load the data and label file and observer their features

```
train_data=pd.read_csv("./traindata.txt",header=None,sep=' ')
train_label=pd.read_csv("./trainlabel.txt",header=None)
print(train_data.head(3))
print(train_label.head(3))
```

```
   0    1   2   3  4     5      6   7
0  6  148  72  35  0  33.6  0.627  50
1  1   85  66  29  0  26.6  0.351  31
2  8  183  64   0  0  23.3  0.672  32
   0
0  1
1  0
2  1
```

This is a **binary classification problem** and the **dimension of the training data is 8**. Therefore, I decide to build a model based on **RandomForestClassifier since it has a good performance on imbalance dataset**. Then I convert the dataframe to numpy array and split the training set.

```
train_data=train_data.to_numpy()
train_label=train_label.to_numpy()
x_train,x_test,y_train,y_test=train_test_split(train_data,train_label,test_size=
0.2,random_state=0)
```

Build the RandomForestClassifier and begin to train the model.

```
clf = RandomForestClassifier(n_estimators=50,random_state=2)
clf = clf.fit(x_train, y_train.ravel())
```

Evaluate the model performance on the valid set.

```
clf.score(x_test,y_test)
```

```
0.8083333333333333
```

Finally I test the model performance on the test set, and write the testlabel to a txt file

```
test_data=pd.read_csv("./testdata.txt",header=None,sep=' ').to_numpy()
test_label=clf.predict(test_data)
with open ('testlabel.txt','w') as f:
    for i in test_label:
        f.write(str(i)+'\n')
```