压缩包中包含有

训练源代码：classification.ipynb文件和一个classification.py文件

测试接口：test_interface_classification.ipynb 文件

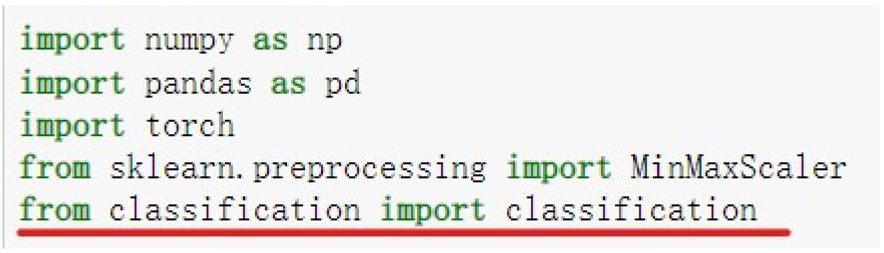
报告：Report_classification.pdf

模型：model_classification.pkl

训练集：3_train_classification.csv

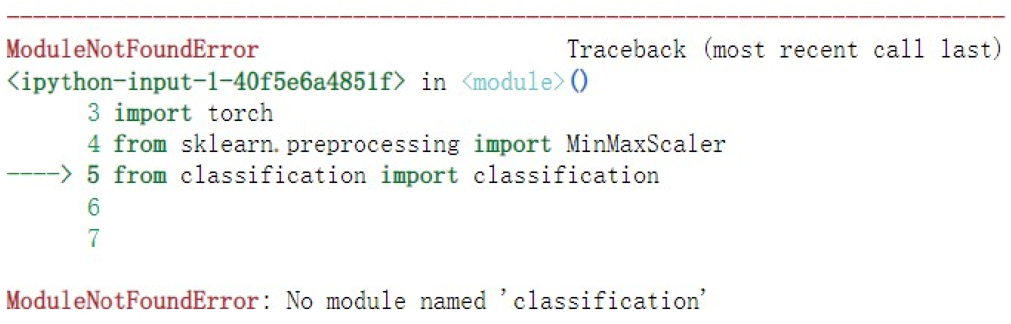
总共6个文件

如果用jupyter notebook打开测试接口，由于我们测试接口中导入了classification.py中的模块



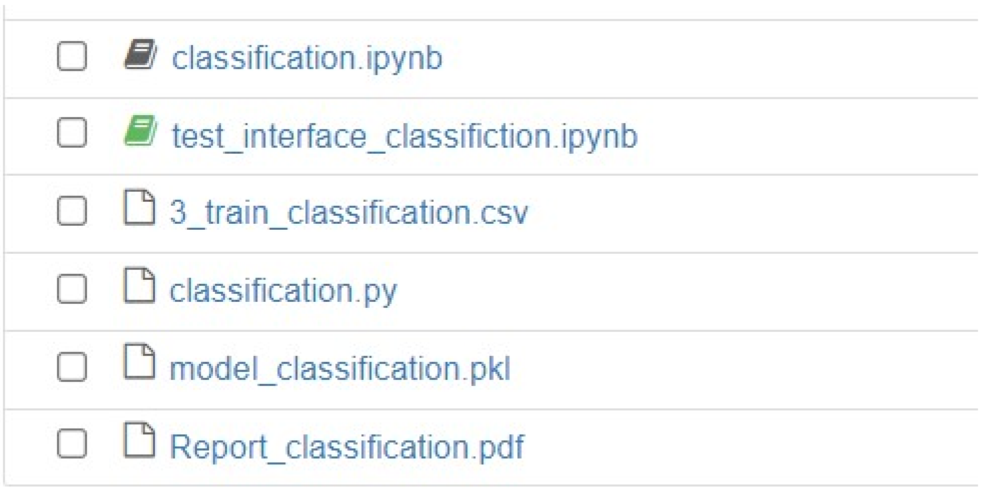
```
import numpy as np
import pandas as pd
import torch
from sklearn.preprocessing import MinMaxScaler
from classification import classification
```

因此，使用jupyter notebook打开上传test_interface_classification.ipynb时一定要把classification.py也上传过去，否则会报错，如图



```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-1-40f5e6a4851f> in <module>()
      3 import torch
      4 from sklearn.preprocessing import MinMaxScaler
----> 5 from classification import classification
      6
      7

ModuleNotFoundError: No module named 'classification'
```

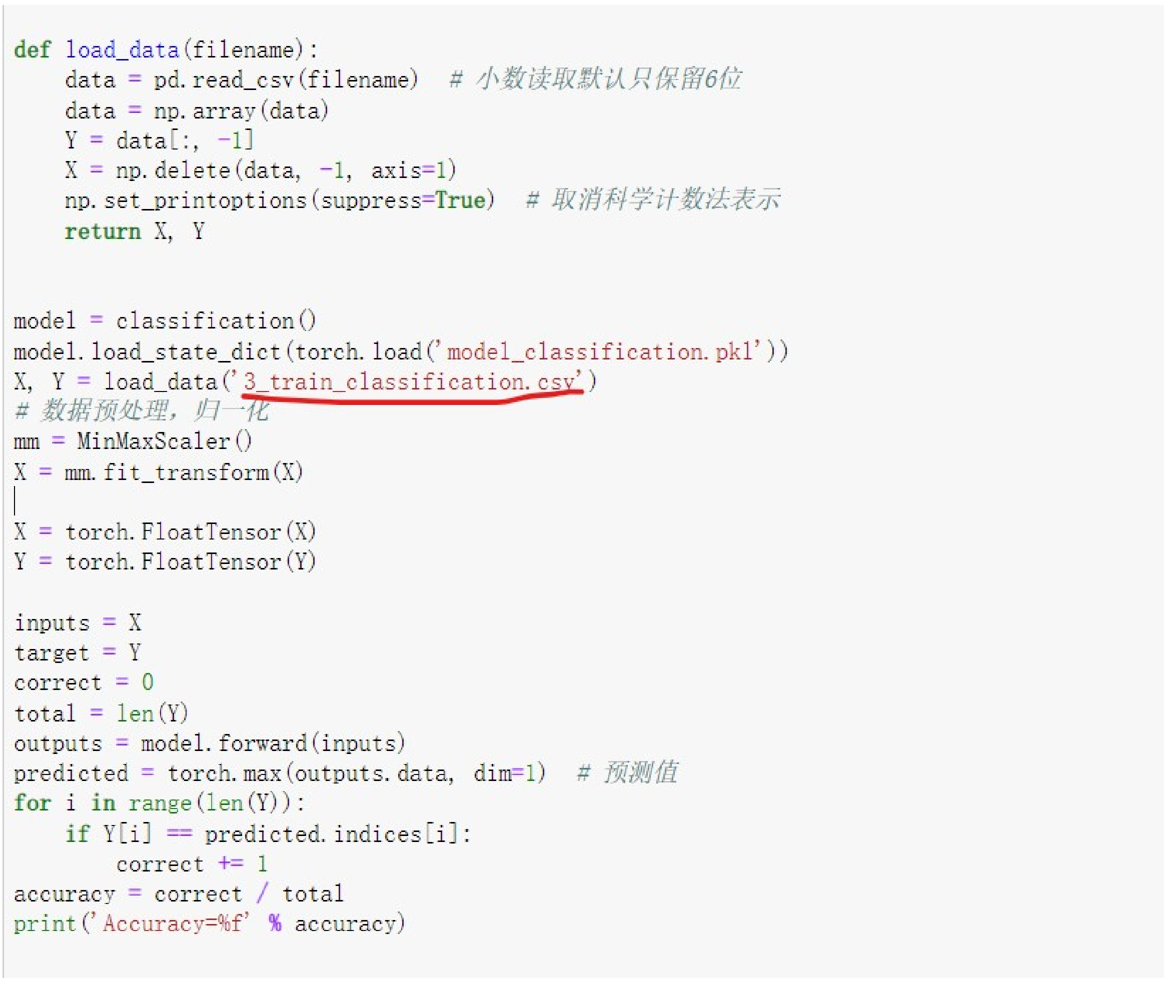
最后jupyter notebook文件夹中应包含这些文件，**再加上一个测试集**共7个文件



```
classification.ipynb
test_interface_classifiction.ipynb
3_train_classification.csv
classification.py
model_classification.pkl
Report_classification.pdf
```

导入数据步骤

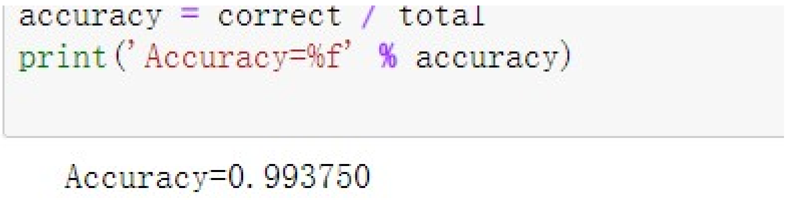
1、打开test_interface_classification.ipynb文件



```python
def load_data(filename):
    data = pd.read_csv(filename)  # 小数读取默认只保留6位
    data = np.array(data)
    Y = data[:, -1]
    X = np.delete(data, -1, axis=1)
    np.set_printoptions(suppress=True)  # 取消科学计数法表示
    return X, Y


model = classification()
model.load_state_dict(torch.load('model_classification.pkl'))
X, Y = load_data('3_train_classification.csv')
# 数据预处理，归一化
mm = MinMaxScaler()
X = mm.fit_transform(X)

X = torch.FloatTensor(X)
Y = torch.FloatTensor(Y)

inputs = X
target = Y
correct = 0
total = len(Y)
outputs = model.forward(inputs)
predicted = torch.max(outputs.data, dim=1)  # 预测值
for i in range(len(Y)):
    if Y[i] == predicted.indices[i]:
        correct += 1
accuracy = correct / total
print('Accuracy=%f' % accuracy)
```

在该处填写测试集的文件名（或者添加测试集路径）

点击运行，会输出Acurracy



```python
accuracy = correct / total
print('Accuracy=%f' % accuracy)
```

```
Accuracy=0.993750
```