**实验目录**

1.     基于逻辑回归的鸢尾花多分类预测

**实验内容**

1.     基于逻辑回归的鸢尾花多分类预测

知识点

1)       逻辑回归可以处理多分类问题

2)      逻辑回归解决多分类问题的方法包括OVR和SoftMax

3)      通常情况下SoftMax方法好于OVR

实验目的

1)       学习建立逻辑回归模型预测三种鸢尾花分类

2)      学习使用OVR方法建立逻辑回归模型

3)      学习使用SoftMax方法建立逻辑回归模型

实验步骤

1)     打开Jupyter，并新建python工程

2)     读取数据

1.        Jupyter输入代码后，使用shift+enter执行，下同。

2.       鸢尾花数据集是一个非常著名的数据集，数据包括三种鸢尾花花萼、花瓣的长宽（厘米计量），每种鸢尾花包括50个样本，共150个样本。字段如下:

sepal\_l: 花萼长度，连续变量，厘米计量

sepal\_w: 花萼宽度，连续变量，厘米计量

petal\_l：花瓣长度，连续变量，厘米计量

petal\_w：花瓣宽度，连续变量，厘米计量

classes：鸢尾花种类，离散变量，（0-2）

3.       使用pandas读取csv文件

[Code 001]:

*import pandas as pd*

*data\_iris = pd.read\_csv('/root/experiment/datas/iris.csv',index\_col=0)*

*# 查看数据维度*

*data\_iris.shape*

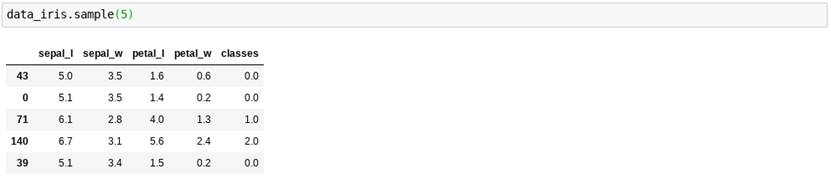


3)     描述性分析与可视化分析

1.        查看数据的随机五项

[Code 002]:

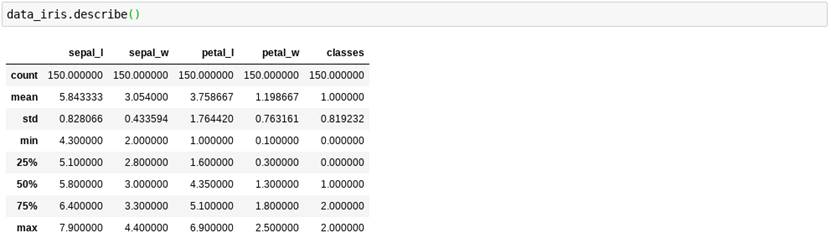
*data\_iris.sample(5)*



2.       查看数据统计描述

[Code 003]:

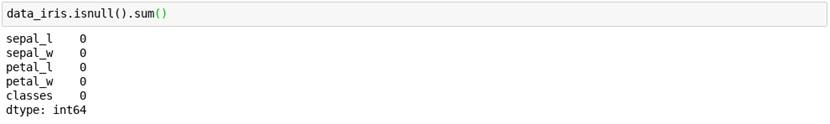
*data\_iris.describe()*



3.       查看数据缺失值

[Code 004]:

*data\_iris.isnull().sum()*



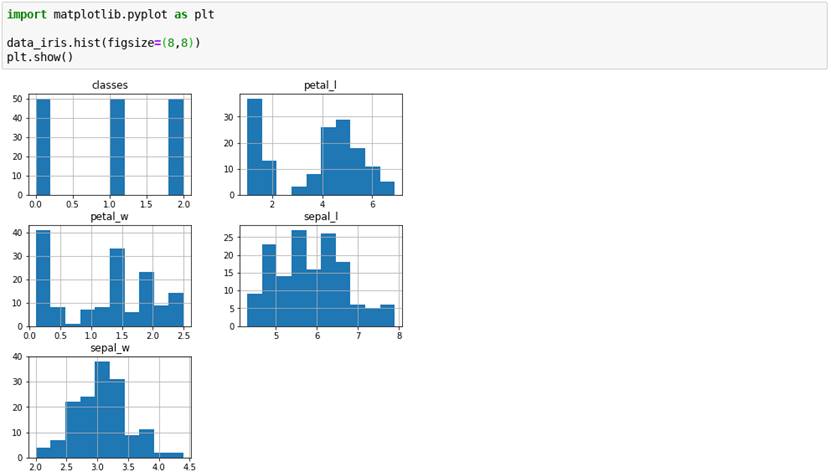
4.       查看数据分布（绘图时，由于jupyter的问题，执行时可能需重复执行才能显示绘图结果，下同）

[Code 005]:

*import matplotlib.pyplot as plt*

*data\_iris.hist(figsize=(8,8))*

*plt.show()*



5.       数据可视化分析

[Code 006]:

*plt.figure(figsize=(16,6))*

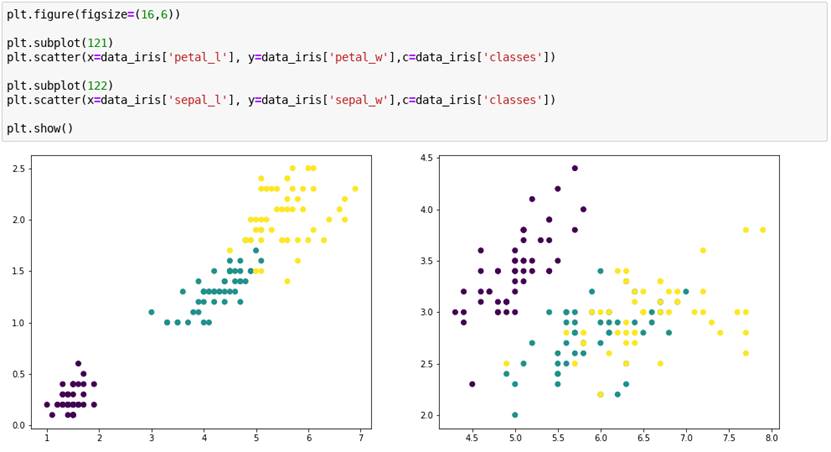
*plt.subplot(121)*

*plt.scatter(x=data\_iris['petal\_l'], y=data\_iris['petal\_w'],c=data\_iris['classes'])*

*plt.subplot(122)*

*plt.scatter(x=data\_iris['sepal\_l'], y=data\_iris['sepal\_w'],c=data\_iris['classes'])*

*plt.show()*



4)     数据预处理

1.        划分自变量和因变量，训练集和测试集

[Code 007]:

*# 划分自变量和因变量*

*X = data\_iris.iloc[:,:-1]*

*y = data\_iris.iloc[:,-1]*

*# 划分训练集和测试集*

*from sklearn.model\_selection import train\_test\_split*

*X\_tr,X\_ts,y\_tr,y\_ts = train\_test\_split(X,y,test\_size=0.2)*

*X\_tr.shape,X\_ts.shape*



5)     建立模型

1.        使用OVR方式建立逻辑回归模型

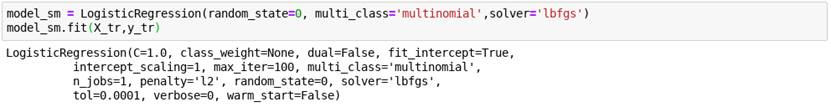
[Code 008]:

*？？？？？*

？？？？

2.       使用SoftMax方式建立逻辑回归模型

[Code 009]:



6)     模型预测与评估

1.        预测并查看混淆矩阵

[Code 010]:

*# 对测试集进行预测*

*y\_ovr\_pred = model\_ovr.predict(X\_ts)*

*y\_sm\_pred = model\_sm.predict(X\_ts)*

*# 查看OVR预测混淆矩阵*

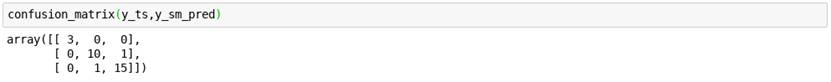
*？？？？*

？？？？

2.       查看SoftMax预测混淆矩阵

[Code 011]:

*confusion\_matrix(y\_ts,y\_sm\_pred)*



3.       查看混淆矩阵评分

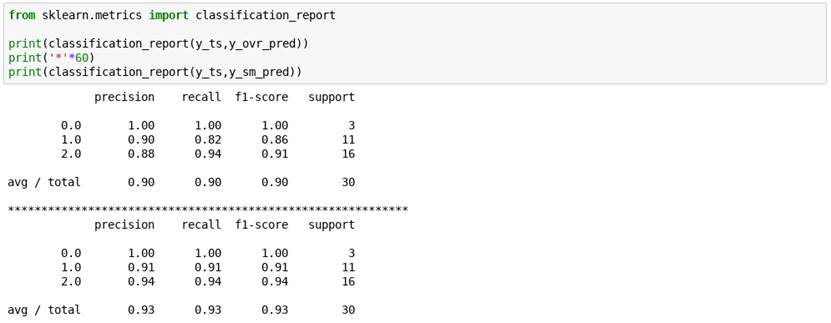
[Code 012]:

*from sklearn.metrics import classification\_report*

*print(classification\_report(y\_ts,y\_ovr\_pred))*

*print('\*'\*60)*

*print(classification\_report(y\_ts,y\_sm\_pred))*



4.       10折交叉验证计算OVR模型准确率

[Code 013]:



5.       10折交叉验证计算SoftMax模型准确率

[Code 014]:

*？？？*

7)     实验结论与总结

？？？？？？