**2019.0.24总结**部分总结在1.23文档中

**第三周编程题： 带有一个隐藏层的平面数据分类**

**需要的软件包**

numpy：是用Python进行科学计算的基本软件包。

sklearn：为数据挖掘和数据分析提供的简单高效的工具。

matplotlib ：是一个用于在Python中绘制图表的库。

testCases：提供了一些测试示例来评估函数的正确性，参见下载的资料或者在底部查看它的代码。

planar\_utils ：提供了在这个任务中使用的各种有用的功能，参见下载的资料或者在底部查看它的代码。

import 和from import 区别

import:导入整个库，如果要使用其中一个类要加上对应的类名限制。

from import：直接导入库里面的特定类

1. 实例：>>> import datetime #导入整个库
2. >>> print(datetime.datetime.now())

>>> from datetime import datetime #从某库中只导入某类

>>> print(datetime.now())

from import \* #导入全部函数

若库中有类，使用方法类名.函数（）

若库中没有类，使用方法直接调用函数（）

导入包：

*import* numpy *as* np  
*import* matplotlib.pyplot *as* plt  
*from* testCases *import* \*  
*import* sklearn  
*import* sklearn.datasets  
*import* sklearn.linear\_model  
*from* planar\_utils *import* plot\_decision\_boundary, sigmoid, load\_planar\_dataset, load\_extra\_datasets  
np.random.seed(1) #设置一个固定的随机种子，以保证接下来的步骤中我们的结果是一致的。

加载数据集到X,Y中

X, Y = load\_planar\_dataset()  
#plt.scatter(X[0, :], X[1, :], c=np.squeeze(Y), s=40, cmap=plt.cm.Spectral) #绘制散点图  
#plt.show()  
shape\_X = X.shape  
shape\_Y = Y.shape  
m = Y.shape[1]#训练集里面的数量  
  
*print* ("X的维度为："+str(shape\_X))  
*print* ("Y的维度为："+str(shape\_Y))  
*print* ("数据集里面的数据有："+str(m)+"个")