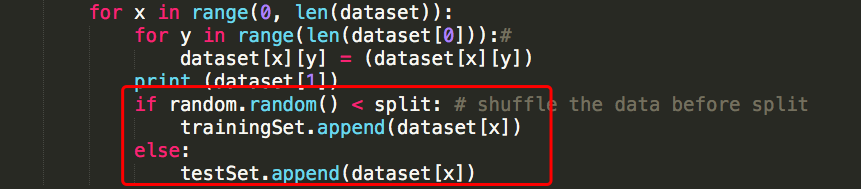
几点note

1. Q1 .py中在数据划分部分,split的作用 如果说你是为了把数据集分成train:test = split : 1-split，依靠你的代码，最后划分的数据集 train 和 test 也并不满足这个比例。 或许当train的数量足够多的情况下，这种方式可以保证最后划分的结果比例符合。 如果我猜错了你的用意，直接skip掉我的code 就好了。



我定义了一个函数

load\_data(file,split,trainingSet=[], testSet=[])

你可以通过在主函数 main 中任意 注释掉一个load 函数来运行

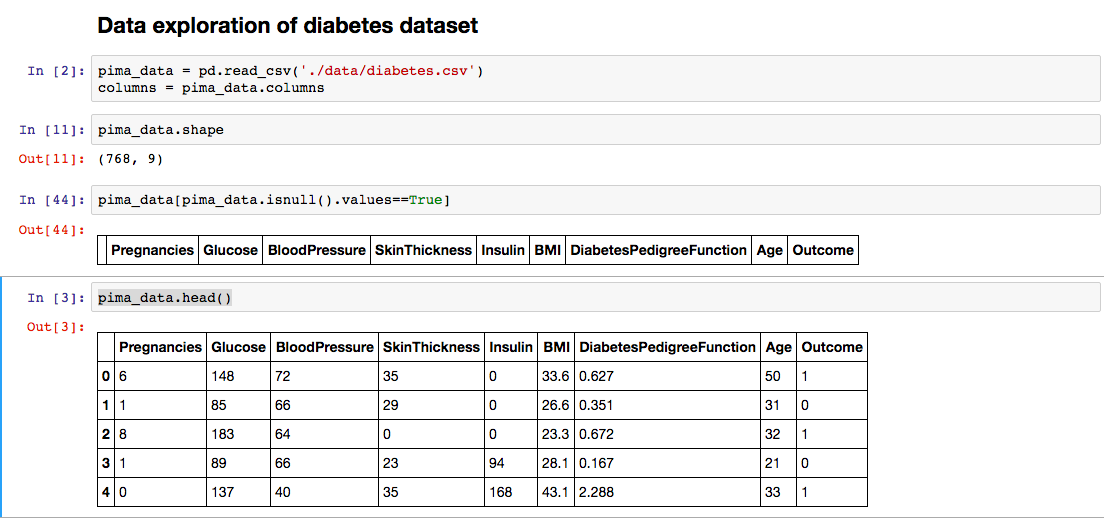
1. 另外我加了一个数据预处理的方式 ，你原来用是standardscaler，他所做的归一化是把feature中每一个x 通过 即 减去均值然后除以标准差来归一。

我加了一种是MinMaxScaler ，这种原理是把每一个feature的值域都限定到(0,1)之间。 在第一个数据集上表现良好，运行准确率高于standardscaler。

1. 计算运行时间用了time 模块 单位是s 秒， 参考这个链接

<http://www.runoob.com/python/python-date-time.html>

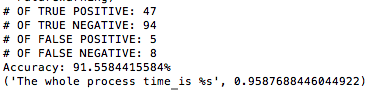
1. 建议下载一个jupyter 可以通过安装anaconda来安装 效果如下



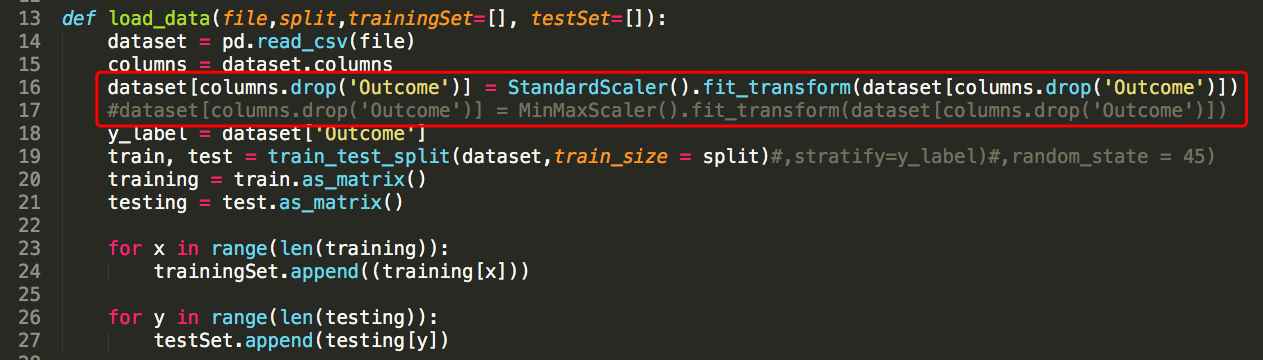


运行步骤

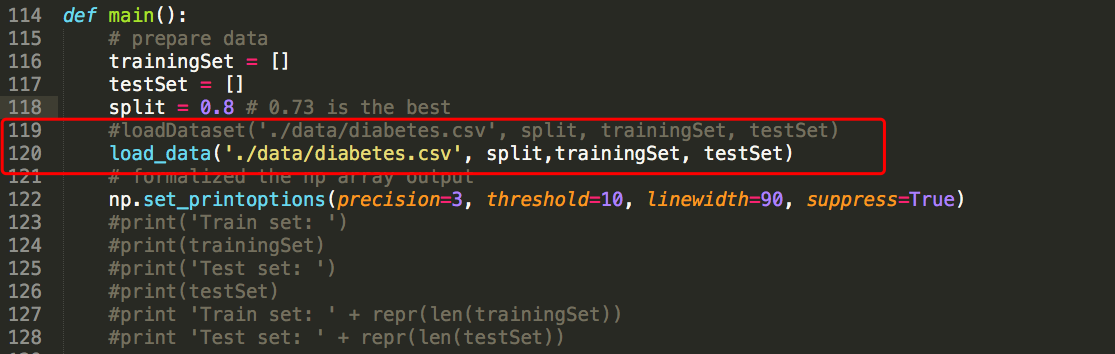
针对 Dataset 1 : python Q1.py



* 1. 如果你想要试一下minmaxscaler,把代码里红框处 上面注释掉，下面的# 去了 就可以



* 1. 如果你想要用自己的划分数据集函数



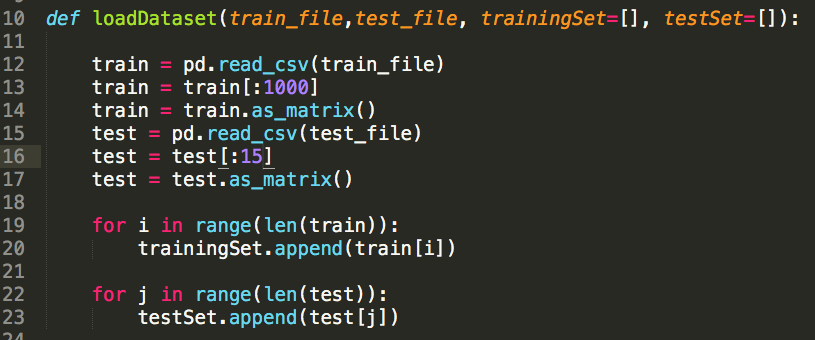
红框下面的代码注释掉，上面的”#”去掉

2 针对 Dataset 2:

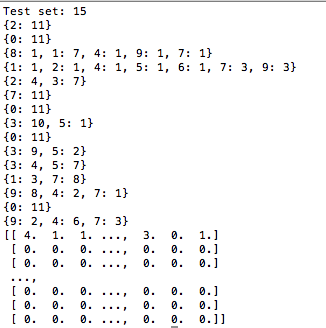
2.1 Python Q2.py

Q2.py 是我针对你的KNN实现改的版本 ，运行和Q1一样，只不过比较花时间。

我用了训练数据中的前1000行和测试数据的前15行作为数据集测试了 整个代码没有问题，正式运行的时候需要把这两行注释了。 不过正式运行，结果似乎还出不来，太久了 :<

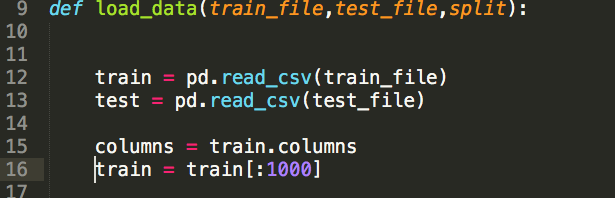


测试的输出结果

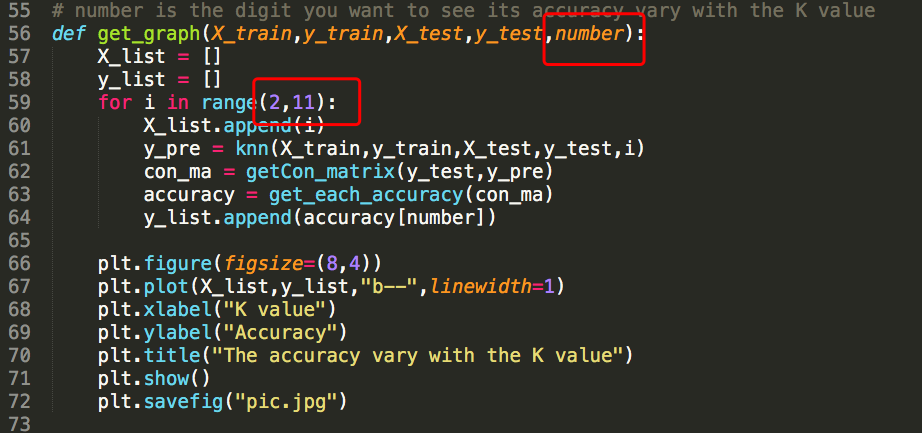


2.2 调用scikit-learn 库中的knn 代码是ref.py 直接运行就可以

注意，你第一次直接运行 是为了检查整个程序在你电脑上运行是否有问题，所以在第16行我写的是只取前面1000行数据作为数据集，当你正式运行的时候，应该把这一行注释了。



2.3 如果你要改KNN 中K的值 以及 你想要看另外一个 number accuracy的变化，改ref.py文件中下面两个红框。 Number 就是你想要看的数字, （2，11）意思是从2开始到10(python 中range 都是到前一个位置为止)的K值 下 研究accuracy 的变化曲线。



3. report中的图都是用data\_visualize.py中的代码画的，下了jupyter的话，打开.ipynb那个文件更直观一点，up to u.

此外，在mnist数据集上， 我没有尝试那两个scaler因为数据集太大，apply这两个scaler 也要很久 出不来结果，再者感觉上，所有值的值域都是0,1 用不用scaler应该影响不大，当然感兴趣的话 最好还是尝试一下。