**实验报告**

一．组员信息：标明组长，组员的分工信息

组员：陆莹莹（无分组，独自完成）

班级：软件183

学号：1806300156

二.作业题目和内容

实验一 《多源数据集成、清洗和统计》

题目

广州大学某班有同学100人，现要从两个数据源汇总学生数据。第一个数据源在数据库中，第二个数据源在txt文件中，两个数据源课程存在缺失、冗余和不一致性，请用C/C++/Java程序实现对两个数据源的一致性合并以及每个学生样本的数值量化。

● 数据库表：ID (int), 姓名(string), 家乡(string:限定为Beijing / Guangzhou / Shenzhen / Shanghai), 性别（string:boy/girl）、身高（float:单位是cm)）、课程1成绩（float）、课程2成绩（float）、...、课程10成绩(float)、体能测试成绩（string：bad/general/good/excellent）；其中课程1-课程5为百分制，课程6-课程10为十分制。

● txt文件：ID(string：6位学号)，性别（string:male/female）、身高（string:单位是m)）、课程1成绩（string）、课程2成绩（string）、...、课程10成绩(string)、体能测试成绩（string：差/一般/良好/优秀）；其中课程1-课程5为百分制，课程6-课程10为十分制。

两个数据源合并后读入内存，并统计：

1. 学生中家乡在Beijing的所有课程的平均成绩。

2. 学生中家乡在广州，课程1在80分以上，且课程9在9分以上的男同学的数量。(备注：该处做了修正，课程10数据为空，更改为课程9)

3. 比较广州和上海两地女生的平均体能测试成绩，哪个地区的更强些？

4. 学习成绩和体能测试成绩，两者的相关性是多少？（九门课的成绩分别与体能成绩计算相关性）

实验二 《数据统计和可视化》

题目

基于实验一中清洗后的数据练习统计和视化操作，100个同学（样本），每个同学有11门课程的成绩（11维的向量）；那么构成了一个100x11的数据矩阵。以你擅长的语言C/C++/Java/Python/Matlab，编程计算：

1. 请以课程1成绩为x轴，体能成绩为y轴，画出散点图。

2. 以5分为间隔，画出课程1的成绩直方图。

3. 对每门成绩进行z-score归一化，得到归一化的数据矩阵。

4. 计算出100x100的相关矩阵，并可视化出混淆矩阵。（为避免歧义，这里“协相关矩阵”进一步细化更正为100x100的相关矩阵，100为学生样本数目，视实际情况而定）

5. 根据相关矩阵，找到距离每个样本最近的三个样本，得到100x3的矩阵（每一行为对应三个样本的ID）输出到txt文件中，以\t,\n间隔。

实验三 《k-means聚类算法》

题目

用C++实现k-means聚类算法，

1. 对实验二中的z-score归一化的成绩数据进行测试，观察聚类为2类，3类，4类，5类的结果，观察得出什么结论？

2. 由老师给出测试数据，进行测试，并画出可视化出散点图，类中心，类半径，并分析聚为几类合适。

三．作业环境：文件说明，函数说明，调用的函数库以及涉及哪些技术

实验一：

文件说明：

函数说明：

# 定义一个函数查判断某id是否在d1中

def FindID\_d1(ID)

# 定义一个函数判断某ID是否在d2，有则返回下标

def FindID\_d2(ID)

调用的函数库：sys,xlrs,numpy

涉及的技术：导入txt；导入xlsx；排序函数；

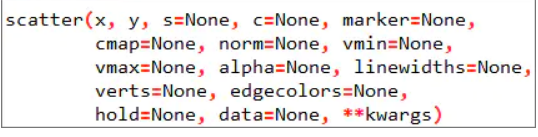
实验二：

文件说明：

函数说明：

map(nums.index, heapq.nlargest(4, nums)):# 找到每一行的4个最大值的下标

要绘制单个点我们需要使用scatter()函数；



x,y 形如shape(n,)的数组，可选值，

s 点的大小（也就是面积）默认20

c 点的颜色或颜色序列，默认蓝色。其它如c = 'r' (red); c = 'g' (green); c = 'k' (black) ; c = 'y'(yellow)

marker 形状，可选值，默认是圆

直方图绘制参数详解：

plt.hist(x, bins=10, range=None, normed=False, weights=None, cumulative=False, bottom=None, histtype='bar', align='mid', orientation='vertical', rwidth=None, log=False, color=None, label=None, stacked=False)

x：指定要绘制直方图的数据。

bins：指定直方图条形的个数。

range：指定直方图数据的上下界，默认包含绘图数据的最大值和最小值。

normed：是否将直方图的频数转换成频率。

weights：该参数可为每一个数据点设置权重。

cumulative：是否需要计算累计频数或频率。

bottom：可以为直方图的每个条形添加基准线，默认为0。

histtype：指定直方图的类型，默认为bar，除此之外，还有barstacked、step和stepfilled

align：设置条形边界值的对齐方式，默认为mid，另外还有left和right

orientation：设置直方图的摆放方向，默认为垂直方向

rwidth：设置直方图条形的宽度

log：是否需要对绘图数据进行log变换

color：设置直方图的填充色

edgecolor：设置直方图边框色

label：设置直方图的标签，可通过legend展示其图例

stacked：当有多个数据时，是否需要将直方图呈堆叠摆放，默认水平摆放

调用的函数库：sys, xlrs, numpy, pyplot, pylab

涉及的技术：python画散点图和直方图

Python中生成并绘制混淆矩阵（confusion matrix）--从博客中复制

首先import一些必要的库：

from sklearn.metrics import confusion\_matrix # 生成混淆矩阵函数

import matplotlib.pyplot as plt # 绘图库

import numpy as np

import tensorflow as tf

然后定义绘制混淆矩阵函数：

def plot\_confusion\_matrix(cm, labels\_name, title):

cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis] # 归一化

plt.imshow(cm, interpolation='nearest') # 在特定的窗口上显示图像

plt.title(title) # 图像标题

plt.colorbar()

num\_local = np.array(range(len(labels\_name)))

plt.xticks(num\_local, labels\_name, rotation=90) # 将标签印在x轴坐标上

plt.yticks(num\_local, labels\_name) # 将标签印在y轴坐标上

plt.ylabel('True label')

plt.xlabel('Predicted label')

生成混淆矩阵：

其中pred\_y为预测值，y\_为网络输出预测值，test\_x为测试输入值，test\_y为测试真实值。

（本程序中标签为one-hot形式，故使用tf.argmax(y\_, 1)和tf.argmax(test\_y, 1)，若标签为普通列表形式，请直接使用y\_和test\_y）

pred\_y = session.run(tf.argmax(y\_, 1), feed\_dict={X: test\_x})

cm = confusion\_matrix(np.argmax(test\_y, 1), pred\_y,)

print(cm)

绘制混淆矩阵热图并显示：

plot\_confusion\_matrix(cm, labels\_name, "HAR Confusion Matrix")

# plt.savefig('/HAR\_cm.png', format='png')

plt.show()

四．难题与解决

实验一用的语言是python，因为用python处理数据会更为方便。难题就是我还没有学习过python，所以我先用了一段时间来学习它，我才开始进行实验。Python涉及的知识点也很多，在短时间内我并没有掌握的很好，所以我仍需要花更多时间去学习，去巩固，去加深。

对于实验一的数据合并，我一开始并没有头绪，实际上数据合并的实现并不容易，但也不会非常难。所以我先想清楚大致的过程，然后再开始实现。

对于混淆矩阵，我没有掌握，不知道如何下手。搜寻资料了解到初步浅显的知识，但不知如何运用在本次实验中。

五．总结

对于数据的处理不是一个简单的过程，因此凸显出数据处理技术的重要性。通过实验对数据处理有更进一步的理解。

本次实验学习到了散点图和直方图的编写。散点图经常用来显示分布或者比较几个变量的相关性或者分组。直方图(Histogram)，又称质量分布图，是一种统计报告图，由一系列高度不等的纵向条纹或线段表示数据分布的情况。 一般用横轴表示数据类型，纵轴表示分布情况。

本次实验新学习python语言，Python是一种跨平台的计算机程序设计语言。 是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。我体会到：Python的设计哲学是“优雅”、“明确”、“简单”。里面有许多很方便的方法，有时候会比c++更为容易实现。

在实验过程中我也更为理解本个课程的课程内容和核心，一开始我对机器学习与数据挖掘只是表面上的的理解，现在能更为深入地了解本个课程，收获匪浅。