**课程预习报告**

电气2009班 U202012465 黄炼

本人签字：\_\_\_\_\_\_\_\_\_\_\_

## 1 预习目标

### 1.1对这门课的期望

在网上了解了一些神经网络在图像识别、语音识别等方面的应用，以及深度学习的一些应用后，对相关内容产生了兴趣。

期待能够学习到人工智能、神经网络等内容，利用机器学习完成一些有趣的项目，提高相关方面的能力。

### 1.2对人工智能的理解

人工智能是通过编程的方式，使机器拥有一个方面或者多个方面的专长能力，表现出存在一定的智能，甚至超过人的智能和能力。

期望：在弱人工智能方面，能够在某一方面拥有极强能力，实现对人类智能的超越（而不是功能类似于“人工智障”），从而减轻人的工作负担。

### 1.3预习计划

1. 学习python语言的基本使用

第一步：使用Anaconda为工具，配置python环境；

第二步：使用Pycharm作为IDE，用于编写代码；

第三步：参照B站视频，以及网上相关文档，模仿实践。

预期水平：学会python的基本使用，能够熟练运用IDE，能够运用python的基础功能写一些简单的程序。

1. 初步学习常用的用于数据处理的库

主要包括用于表格读取的xlrd，用于数据处理的numpy和pandas，用于绘图的matplotlib。

预期水平：了解他们是什么，学习最基本的一些使用方法，尝试应用于一些简单的程序中。

1. 预习项目：利用已学的内容，对一些成绩数据进行多维度分析。

## 2 基础知识预习

1. 学习python语言的基本使用

成功下载安装Anaconda和Pycharm，并配置了虚拟环境，环境中包含基础库，还安装了准备学习的库。

学会了如何安装、调用所需库，python的基本数据结构的使用，python的常用语法等。主要了解python与C++的不同点。

1. 初步学习常用的用于数据处理的库

了解了相关的库，尝试了网上的一些相关的简单代码。

## 3 预习项目设计(对应课程目标2，3)

### 3.1 项目选题

选题：成绩数据分析

原有的成绩数据在excel中，数据较为散乱，无法得出有效结论。本项目准备利用这些成绩数据进行分析，找出变化规律，同时也能熟悉python以及数据分析相关库的使用。

使用用于读取表格的xlrd库、用于数据处理的numpy库、用于数据处理的pandas库和用于画图的matplotlib库，编写python代码实现数据分析和可视化。

### 3.2 程序设计

1. 读取表格中的数据

1.1 使用xlrd库读入数据，将各列存入列表中

1.2 print数据，发现存在空值和不必要数据

1.3 删除空值和不必要数据

2. 使用numpy进行数据分析

2.1 使用numpy的库函数求取数据特征值

2.2 对前50名的数据求取特征值

2.3 使用matplotlib的库函数以表格形式展示结果

3. 使用matplotlib绘制数据分布情况

3.1 使用pandas的库函数将成绩数据按区间分割

3.2 使用matplotlib的库函数展示成绩分布

### 3.3 运行结果

程序见附录OpenDocument文本，所用数据见附录Microsoft Excel Worksheet。

将程序在Pycharm中运行，期望得到特征值数据表和成绩分布图。

得到的结果如下图1、图2所示。

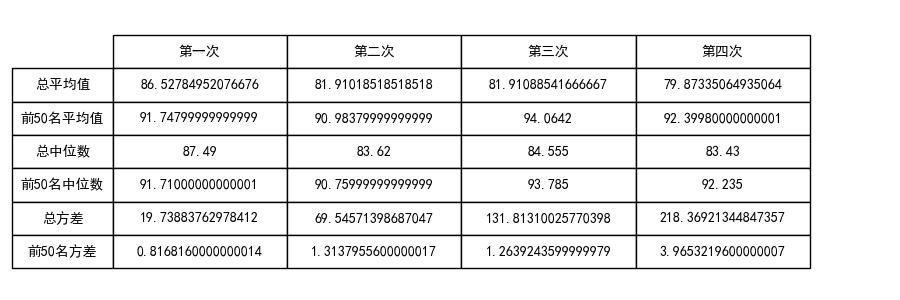


图1 数据特征值表格

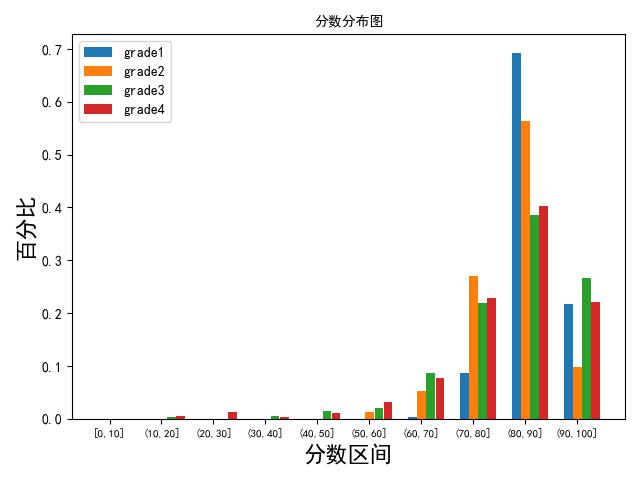


图2 成绩分布图

结果分析：

分析表可知，

总平均值和中位数，第一次最高，后三次次较低；

前50名平均值和中位数，先减小后增大再减小，但波动不大；

总方差不断增大；

前50名方差，第一次较小，第二、三次增大且接近，第四次继续增大。

分析图可知，

(80,90]分范围的人数最多，且第一次的成绩最突出，后三次低分段不断增多；

第二次与第一次比较，(90,100]分数段人数减少而(70,80]分数段人数增多；

第三次和第四次分布接近，与第一次比较，(80,90]分数段人数较少，主要变为(70,80]和(90,100]分数段。

## 4 预习总结

了解了人工智能相关背景知识。

学习了python的环境配置方法和基本使用方法，了解了数据分析相关的库。

利用所学知识和技能完成了一个简单的数据分析项目。

### 附录

1. *# -\*- coding: utf-8 -\*-*
2. *# 分别导入用于读取表格的xlrd库、用于数据处理的numpy库,用于数据处理的pandas库和用于画图的matplotlib库*
3. import xlrd
4. import numpy
5. import matplotlib.pyplot as plt
6. import pandas
7. *# 读取表格，得到第一页的有效数据*
8. data = xlrd.open\_workbook("data.xls")
9. table = data.sheet\_by\_name('Sheet1')
10. *# 将成绩和排名分别存放在数组中*
11. rank1 = table.col\_values(0)
12. grade1 = table.col\_values(1)
13. rank2 = table.col\_values(2)
14. grade2 = table.col\_values(3)
15. rank3 = table.col\_values(4)
16. grade3 = table.col\_values(5)
17. rank4 = table.col\_values(6)
18. grade4 = table.col\_values(7)
19. print(rank1)
20. *# 由于表格第一行不是数据，删除*
21. rank1.pop(0)
22. rank2.pop(0)
23. rank3.pop(0)
24. rank4.pop(0)
25. grade1.pop(0)
26. grade2.pop(0)
27. grade3.pop(0)
28. grade4.pop(0)
29. *# 由于三次人数不同，列表中有空值，删除*
30. while '' in rank1:
31. rank1.remove('')
32. while '' in rank2:
33. rank2.remove('')
34. while '' in rank3:
35. rank3.remove('')
36. while '' in rank4:
37. rank4.remove('')
38. while '' in grade1:
39. grade1.remove('')
40. while '' in grade2:
41. grade2.remove('')
42. while '' in grade3:
43. grade3.remove('')
44. while '' in grade4:
45. grade4.remove('')
46. *# 求三次成绩的平均值，中位数和方差*
47. avg1 = numpy.average(grade1)
48. avg2 = numpy.average(grade2)
49. avg3 = numpy.average(grade3)
50. avg4 = numpy.average(grade4)
51. mid1 = numpy.median(grade1)
52. mid2 = numpy.median(grade2)
53. mid3 = numpy.median(grade3)
54. mid4 = numpy.median(grade4)
55. var1 = numpy.var(grade1)
56. var2 = numpy.var(grade2)
57. var3 = numpy.var(grade3)
58. var4 = numpy.var(grade4)
59. *# 求前50名成绩的平均值，中位数和方差*
60. avg11 = numpy.average(grade1[:50])
61. avg21 = numpy.average(grade2[:50])
62. avg31 = numpy.average(grade3[:50])
63. avg41 = numpy.average(grade4[:50])
64. mid11 = numpy.median(grade1[:50])
65. mid21 = numpy.median(grade2[:50])
66. mid31 = numpy.median(grade3[:50])
67. mid41 = numpy.median(grade4[:50])
68. var11 = numpy.var(grade1[:50])
69. var21 = numpy.var(grade2[:50])
70. var31 = numpy.var(grade3[:50])
71. var41 = numpy.var(grade4[:50])
72. *# 用来正常显示中文标签*
73. plt.rcParams['font.sans-serif'] = ['SimHei']
74. *# 将结果输出为表格进行比较*
75. row = ["总平均值", "前50名平均值", "总中位数", "前50名中位数", "总方差", "前50名方差"]
76. col = ["第一次", "第二次", "第三次", "第四次"]
77. vals = [[avg1, avg2, avg3, avg4], [avg11, avg21, avg31, avg41], [mid1, mid2, mid3, mid4],
78. [mid11, mid21, mid31, mid41], [var1, var2, var3, var4], [var11, var21, var31, var41]]
79. plt.figure(figsize=(9, 3))
80. tab = plt.table(cellText=vals,
81. colLabels=col,
82. rowLabels=row,
83. loc='center',
84. cellLoc='center',
85. rowLoc='center')
86. tab.scale(1, 2)
87. plt.axis('off')  *# 取消表格的坐标轴显示*
88. *# 通过pandas库的cut函数分割区间，并统计分数区间的人数*
89. bin\_grade = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
90. a1 = pandas.cut(x=grade1, bins=bin\_grade, retbins=False, labels=[
91. u"[0,10]", u"(10,20]", u"(20,30]", u"(30,40]", u"(40,50]", u"(50,60]",
92. u"(60,70]", u"(70,80]", u"(80,90]", u"(90,100]"])
93. b1 = a1.value\_counts()
94. a2 = pandas.cut(x=grade2, bins=bin\_grade, retbins=False, labels=[
95. u"[0,10]", u"(10,20]", u"(20,30]", u"(30,40]", u"(40,50]", u"(50,60]",
96. u"(60,70]", u"(70,80]", u"(80,90]", u"(90,100]"])
97. b2 = a2.value\_counts()
98. a3 = pandas.cut(x=grade3, bins=bin\_grade, retbins=False, labels=[
99. u"[0,10]", u"(10,20]", u"(20,30]", u"(30,40]", u"(40,50]", u"(50,60]",
100. u"(60,70]", u"(70,80]", u"(80,90]", u"(90,100]"])
101. b3 = a3.value\_counts()
102. a4 = pandas.cut(x=grade4, bins=bin\_grade, retbins=False, labels=[
103. u"[0,10]", u"(10,20]", u"(20,30]", u"(30,40]", u"(40,50]", u"(50,60]",
104. u"(60,70]", u"(70,80]", u"(80,90]", u"(90,100]"])
105. b4 = a4.value\_counts()
106. *# 将区间都人数提取为列表，作为纵坐标*
107. y1 = b1.values
108. y2 = b2.values
109. y3 = b3.values
110. y4 = b4.values
111. *# 横向标签*
112. labels = ["[0,10]", "(10,20]", "(20,30]", "(30,40]", "(40,50]", "(50,60]",
113. "(60,70]", "(70,80]", "(80,90]", "(90,100]"]
114. x = numpy.arange(len(labels)\*1.2,step = 1.2)  *# 标签位置,包含10组*
115. width = 0.2  *# 柱状图的宽度*
116. fig, ax = plt.subplots()  *# 创建组合图*
117. *# 设置四组绘图数据*
118. rects1 = ax.bar(x - width + 0.01, y1, width, label="grade1")
119. rects2 = ax.bar(x + 0.02, y2, width, label="grade2")
120. rects3 = ax.bar(x + width + 0.03, y3, width, label="grade3")
121. rects4 = ax.bar(x + width\*2 + 0.04, y4, width, label="grade4")
122. *# 为x轴、y轴、标题等添加文本*
123. ax.set\_ylabel('人数', fontsize=16, fontproperties='SimHei')
124. ax.set\_xlabel('分数区间', fontsize=16, fontproperties='SimHei')
125. ax.set\_title('分数分布图', fontproperties='SimHei')
126. ax.set\_xticks(x)
127. ax.set\_xticklabels(labels, fontsize=8)
128. ax.legend()  *# 添加图例*
129. fig.tight\_layout()
130. plt.show()
131. *# 分析表可以知道，总平均值和中位数，第一次最高，后两次较低；*
132. *# 前50名平均值和中位数，三次先减小后增大；*
133. *# 总方差不断增大；前50名方差，第一次较小，后两次较大且接近*
134. *# 第四次是后加的数据，方差较第三次增大，其他变化不大*
135. *# 分析图可知(80,90]分范围的人数最多，且第一次的成绩最突出，后两次低分段不断增多*
136. *# 第二次与第一次比较，(90,100]分数段人数减少而(70,80]分数段人数增多*
137. *# 第三次与第一次比较，(80,90]分数段人数较少，主要变为(70,80]和(90,100]分数段*
138. *# 第四次是后来加入的数据，和第三次接近*