数据科学基础第一次作业

王晨曦

2018 年 4 月 2 日

1 问题重述

用可视化手段初步探索 adult 数据集的基本情况。

2 解答

解答分为两部分，第一部分为各属性的分布情况展示，第二部分为属性之间的相互关系。数据读取、处理和图像绘制的代码附在最后。

2.1 各属性分布情况

2.1.1 年龄分布

如图1所示，将数据集中的年龄以 16 岁为起点，90 岁为终点，15 年为间隔划分成 5 个年龄段，统计每个年龄段的人数在总人数中占的比重，发现该地图 31-45 岁年龄段的人数最多，其次是 46-60 岁年龄段和 16-30 岁年龄段。

2.1.2 工作类别

如图2所示，将每种工作的人数占的比例绘制成饼图，发现将近三分之二的人为 Private 类型。

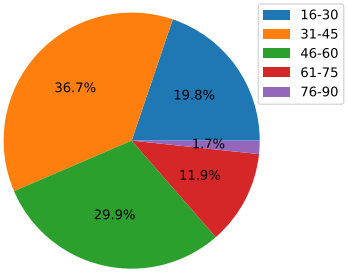

图 1: 年龄分布

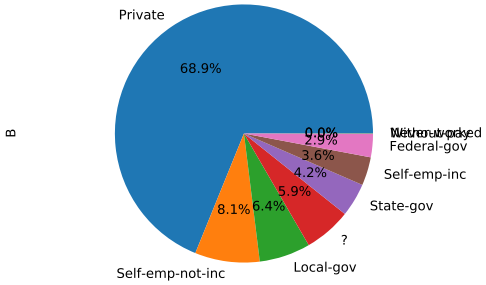

图 2: 工作类别

2.1.3 教育程度

如图3所示，将数据集中人群的教育程度绘制成柱状图，发现高中和大学水平的人居多。

2.1.4 受教育时间

如图4所示，将不同受教育时间对应的人数绘制成柱状图，大部分人的受教育时间为9-12年。

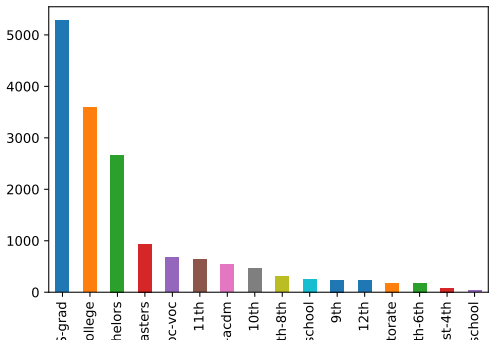

图 3: 教育程度

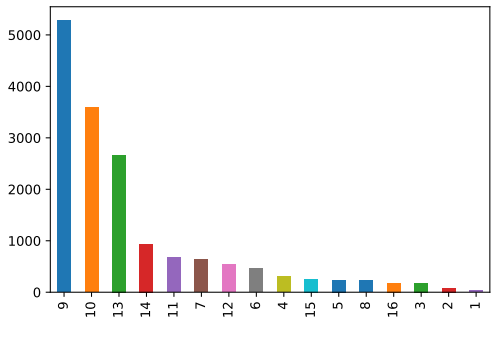

图 4: 受教育时间

2.1.5 婚姻状况

如图5所示，将婚姻状况按人数绘制成饼图，发现将近一半人已婚，单身和离婚的比例也很大。

2.1.6 家庭关系

如图6所示，将家庭关系按人数绘制成饼图，发现丈夫和未婚的关系居多，这与婚姻状况和性别比例相呼应。

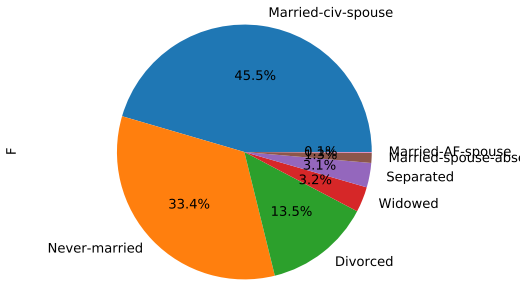

图 5: 婚姻状况

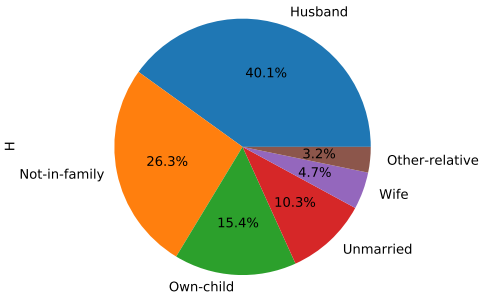

图 6: 家庭关系

2.1.7 职业分布

如图7所示，将不同职业的人数用柱状图绘制出来，发现六种职业在人群中占了很大比重。

2.1.8 种族分布

如图8所示，将种族人数用柱状图绘制出来，发现白种人最多，其他人种很稀少，为了方便查看，纵坐标改为对数表示。

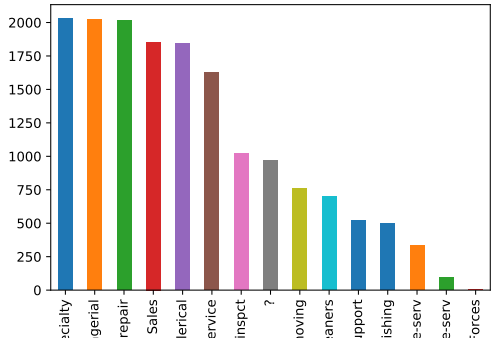

图 7: 职业分布

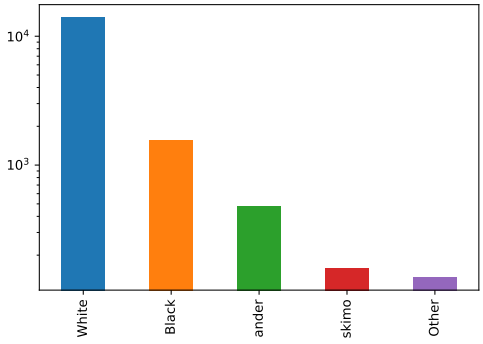

图 8: 种族分布

2.2 属性间相互关系

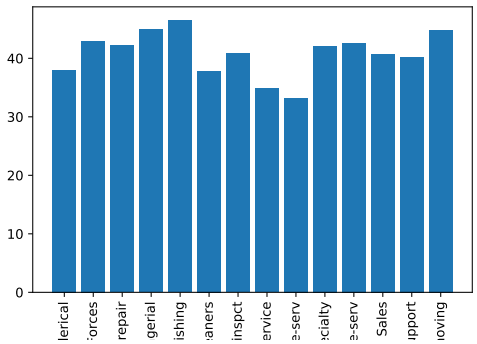

图 9: 工作时间与职业的关系

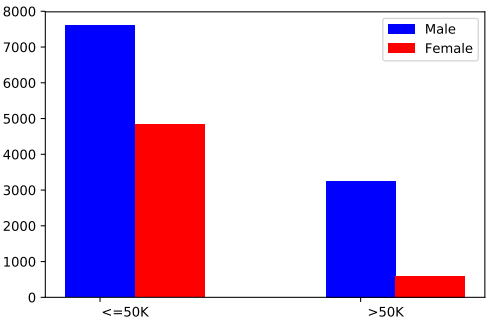

图 10: 性别和收入的关系

2.2.1 工作时间与职业的关系

如图9所示，将每种职业对应的平均工作时间用柱状图绘制出来，发现大部分职业平均工作时间相差不大，少数工作的时间较为极端。

2.2.2 性别与收入的关系

如图10所示，把男性和女性的收入水平画在同一张饼图上，发现该地图男性人数明显大于女性人数，并且男性高收入人数大于女性高收入人数。

3 程序源码

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("./adult.txt",names=list('ABCDEFGHIJKLMNO'))


age = df['A'].values
age_cnt = [0 for _ in range(5)]
age_labels = ['16-30','31-45','46-60','61-75','76-90']
for i in range(5):
    mask = (age>i*15+15) & (age<=i*15+30)
    age_cnt[i] = age[mask].sum()
plt.pie(age_cnt,autopct='%.1f%%')
plt.axis('equal')
plt.legend(age_labels)
plt.savefig('./figures/age.pdf')


work_cls = df['B']
work_cls.value_counts().plot.pie(autopct='%.1f%%')
plt.axis('equal')
plt.savefig('./figures/work_cls.pdf')


edu_lv = df['D']
edu_lv.value_counts().plot.bar()
plt.savefig('./figures/edu_lv.pdf')


edu_time = df['E']
edu_time.value_counts().plot.bar()
plt.savefig('./figures/edu_time.pdf')


marry_stat = df['F']
marry_stat.value_counts().plot.pie(autopct='%.1f%%')
plt.axis('equal')
plt.savefig('./figures/marry_stat.pdf')


business = df['G']
business.value_counts().plot.bar()
plt.savefig('./figures/business.pdf')

```

```
46
47 home_rela = df['H']
48 home_rela.value_counts().plot.pie(autopct='%.1f%%')
49 plt.axis('equal')
50 plt.savefig('./figures/home_rela.pdf')
51
52
53 race = df['I']
54 race.value_counts().plot.bar(logy=True)
55
56
57 business = df['G']
58 work_time = df['M']
59 business = business.values
60 work_time = work_time.values
61 business_cls = np.unique(business)
62 business_cls = business_cls[business_cls!=' ?']
63 avg_work_time = []
64 for obj in business_cls:
65     mask = (business==obj)
66     avg_work_time.append(np.mean(work_time[mask]))
67 plt.bar(business_cls,avg_work_time)
68 plt.xticks(rotation=90)
69 plt.savefig('./figures/business_worktime.pdf')
70
71
72 sex = df['J'].values
73 income = df['O'].values
74 sum_male = [np.sum((sex==' Male')&(income==' <=50K.')), np.sum((sex==' Male')&(
                                        income==' >50K.'))]
75 sum_female = [np.sum((sex==' Female')&(income==' <=50K.')), np.sum((sex=='
                                        Female')&(income==' >50K.'))]
76 plt.bar([0,1.5],sum_male,width=0.4,label='Male',fc='b',tick_label=['
                                                  <=50K','            >50K'])
77 plt.bar([0.4,1.9],sum_female,width=0.4,label='Female',fc='r')
78 plt.legend()
79 plt.savefig('./figures/sex_income.pdf')
```