

# 数据可视化

### 消费总额占比分析



### 人均消费占比分析



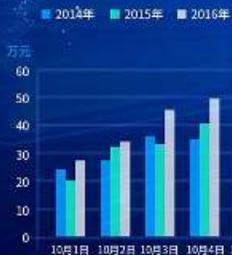
### 性别消费占比分析



### 区域消费占比分析



### 历年十一黄金周景区收入



### 十一黄金周人均消费对比



# 目录

## Contents

---

第一章

数据可视化概述

第二章

Matplotlib的基本使用

第三章

Matplotlib的进阶使用

第四章

...

第五章

...

# 第十章

## 银行产品客户 复购数据可视化 分析案例

1

数据查看与准备

2

客户基本特征分析

3

复购影响因素分析

4

可视化大屏

# 1. 数据查看与准备



本章的案例数据为阿里云天池大赛关于金融数据分析赛题（银行客户认购产品预测）所用数据\*，数据包括客户个人基本特征、银行营销数据以及宏观经济数据。本章选用公开的“train”子数据集文件。



字段	说明
age	年龄
job	工作
marital	婚姻
education	受教育程度
default	是否违约
housing	是否有房贷
loan	是否有贷款
contact	联系方式
month	上一次联系的月份
day_of_week	上一次联系是星期几
duration	上一次联系时长（秒）
campaign	联系次数
pdays	上一次联系后的间隔天数
previous	本次活动前联系次数
poutcome	之前的营销结果
emp_var_rate	就业变动率
cons_price_index	消费者价格指数
cons_conf_index	消费者信心指数
lending_rate3m	银行同业拆借率 3个月利率
nr_employed	雇员人数
subscribe	是否复购



## (1) 查看train数据集

```
import pandas as pd #导入模块pandas, 便于处理数据
file_path = r'D:\数据分析教材' #数据文件路径
data = pd.read_csv(file_path+'\\train.csv') #导入数据
data.head() #查看数据前5行
```

### 运行结果如图10.1

	id	age	job	marital	education	default	housing	loan	contact	month	...	campaign	pdays
0	1	51	admin.	divorced	professional.course	no	yes	yes	cellular	aug	...	1	112
1	2	50	services	married	high.school	unknown	yes	no	cellular	may	...	1	412
2	3	48	blue-collar	divorced	basic.9y	no	no	no	cellular	apr	...	0	1027
3	4	26	entrepreneur	single	high.school	yes	yes	yes	cellular	aug	...	26	998
4	5	45	admin.	single	university.degree	no	no	no	cellular	nov	...	1	240

	previous	poutcome	emp_var_rate	cons_price_index	cons_conf_index	lending_rate3m	nr_employed	subscribe
	2	failure	1.4	90.81	-35.53	0.69	5219.74	no
	2	nonexistent	-1.8	96.33	-40.58	4.05	4974.79	yes
	1	failure	-1.8	96.33	-44.74	1.50	5022.61	no
	0	nonexistent	1.4	97.08	-35.55	5.11	5222.87	yes
	4	success	-3.4	89.82	-33.83	1.17	4884.70	no



## (2) 查看是否存在缺失值

```
data.isnull().any() #查看数据是否存在缺失值
```

### 运行结果如图10.2

```
id                False
age               False
job              False
marital          False
education         False
default          False
housing          False
loan             False
contact          False
month            False
day_of_week      False
duration         False
campaign         False
pdays          False
previous         False
poutcome        False
emp_var_rate     False
cons_price_index False
cons_conf_index  False
lending_rate3m  False
nr_employed      False
subscribe        False
dtype: bool
```



### (3) 修改字段名

将数据集字段名以及“subscribe”字段数据转换成为中文。

#将数据列名更改为中文，方便数据处理

```
new_col = ['序号','年龄','工作','婚姻','受教育程度',
           '是否违约','是否有房贷','是否有贷款','联系方式',
           '上一次联系月份','上一次联系星期','上一次联系时长(秒)',
           '联系次数','上一次联系后的间隔天数','本次活动前联系次数',
           '之前营销结果','就业变动率','消费者价格指数','消费者信心指数',
           '银行同业拆借率 3个月利率','雇员人数','是否复购']
```

```
data.columns = new_col
```

#将subscribe(是否复购)的值更改为'复购'与'未复购',便于理解

```
data.loc[:, '是否复购'].replace({'yes':'复购','no':'未复购'}, inplace = True)
```

序号	年龄	工作	婚姻	受教育程度	是否违约	是否有房贷	是否有贷款	联系方式	上一次联系月份	...	联系次数	上一次联系后的间隔天数	本次活动前联系次数	之前营销结果	就业变动率	消费者价格指数	消费者信心指数	银行同业拆借率 3个月利率	雇员人数	是否复购
0	1	51	admin.	divorced	professional.course	no	yes	yes	cellular	aug ...	1	112	2	failure	1.4	90.81	-35.53	0.69	5219.74	未复购
1	2	50	services	married	high.school	unknown	yes	no	cellular	may ...	1	412	2	nonexistent	-1.8	96.33	-40.58	4.05	4974.79	复购
2	3	48	blue-collar	divorced	basic.9y	no	no	no	cellular	apr ...	0	1027	1	failure	-1.8	96.33	-44.74	1.50	5022.61	未复购
3	4	26	entrepreneur	single	high.school	yes	yes	yes	cellular	aug ...	26	998	0	nonexistent	1.4	97.08	-35.55	5.11	5222.87	复购
4	5	45	admin.	single	university.degree	no	no	no	cellular	nov ...	1	240	4	success	-3.4	89.82	-33.83	1.17	4884.70	未复购

5 rows × 22 columns

## 2. 客户基本特征分析

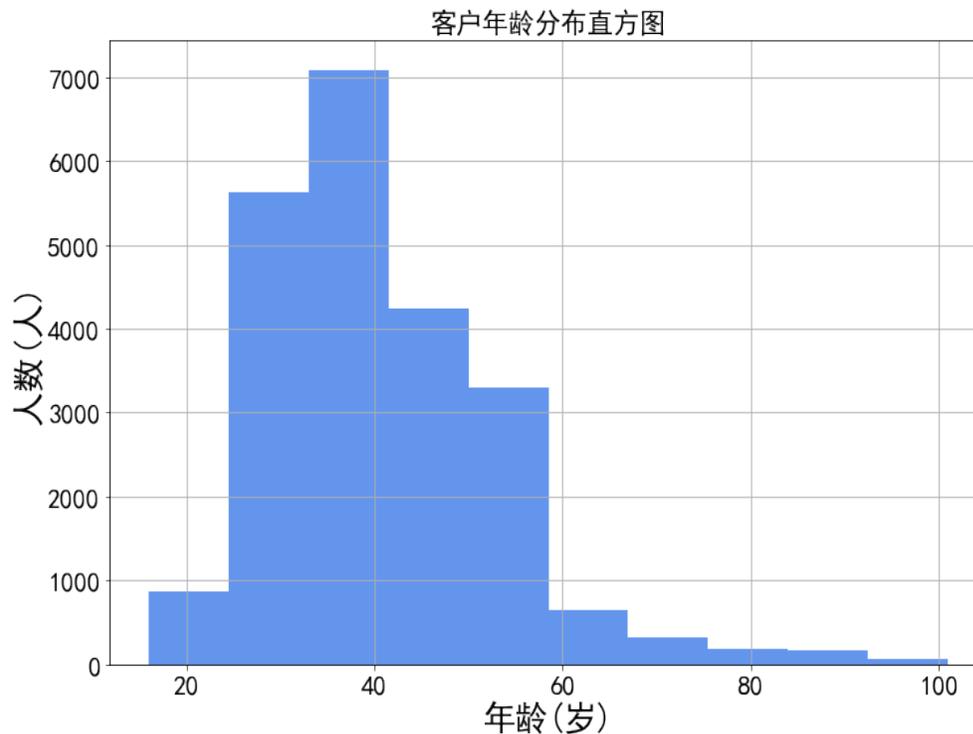


## (1) 客户年龄特征分析

数据集中关于客户基本特征的相关数据包含了客户的年龄数据，若想要直观地查看客户群体的年龄分布规律，可以利用Matplotlib库绘制客户年龄直方图。

```
import matplotlib.pyplot as plt #导入matplotlib.pyplot模块绘图
plt.rcParams['font.family'] = 'SimHei' #将字体设置为黑体
plt.rcParams['axes.unicode_minus'] = False #显示负号

plt.figure(figsize=(12,8),dpi=72) #设置图片大小
plt.title('客户年龄分布直方图',fontsize=20) #设置标题，并将字体大小设置为20
plt.hist(data['年龄'],color='cornflowerblue') #绘制直方图，并设置颜色
plt.grid(True) #显示网格
plt.xlabel('年龄(岁)',size=25) #设置x轴
plt.ylabel('人数(人)',size=25) #设置y轴
plt.xticks(size=20) #设置x轴刻度，并将字体大小设置为20
plt.yticks(size=20) #将y轴刻度字体设置为20
plt.show() #显示图像
```

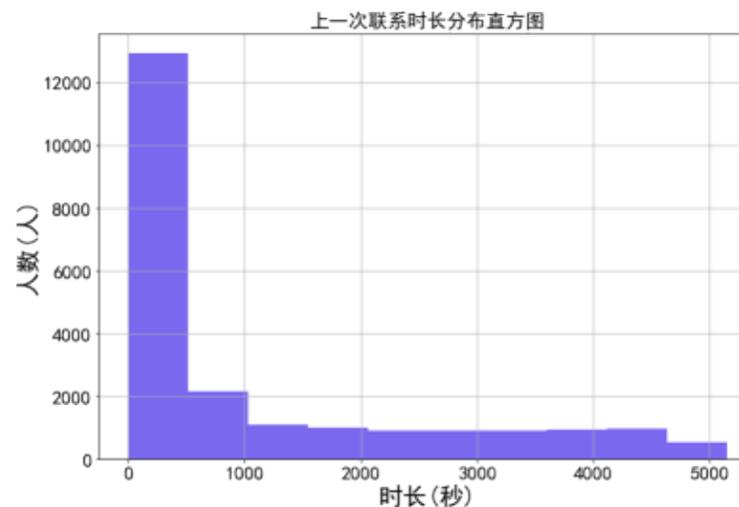




## (2) 客户年龄特征分析

数据集中银行与客户的上一次联系时长数据为一组连续型数据，为了查看上一次联系时长的分布情况，可以利用Matplotlib库绘制联系时长直方图。

```
plt.figure(figsize=(12,8),dpi=72) #设置图片大小
plt.title('上一次联系时长分布直方图',fontsize =20) #设置标题，并将字体大小设置为20
plt.hist(data['上一次联系时长(秒)'],color = 'mediumslateblue') #绘制直方图，并设置颜色
plt.grid(True) #显示网格
plt.xlabel('时长(秒)',size = 25) #设置x轴
plt.ylabel('人数(人)',size = 25) #设置y轴
plt.xticks(size = 20) #设置x轴刻度，并将字体大小设置为20
plt.yticks(size = 20) #将y轴刻度，并将字体大小设置为20
plt.show() #显示图像
```

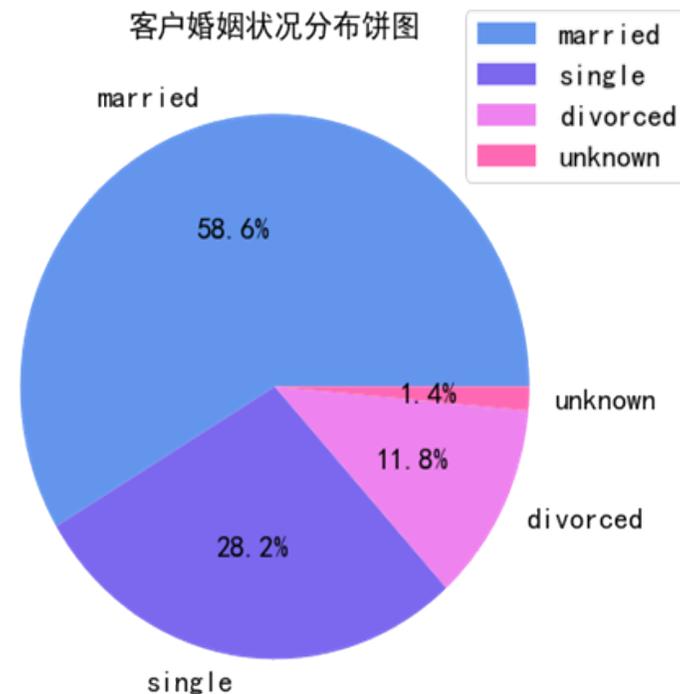




### (3) 客户婚姻状况分析

数据集中客户婚姻字段，有married、single、divorced、unknown四个类别，可以利用Matplotlib库中的pie()函数绘制饼图。

```
plt.figure(figsize=(12,8),dpi=72) #设置图片大小
plt.title('婚姻状况统计饼图',fontsize=20) #设置标题，并将字体大小设置为20
y = data['婚姻'].value_counts() #对“婚姻”字段进行分类计数
plt.pie(y,
        labels=y.index, #设置标签为婚姻状况的4个类别
        autopct='%3.1f%%', #显示小数点后1位
        colors=['cornflowerblue','mediumslateblue','violet','hotpink'], #设置饼图颜色
        textprops={'fontsize':20, 'color':'black'}) #设置标签字体大小及颜色
plt.legend(loc=[0.8,0.8],fontsize=20) #设置图例，指定位置为右上角
plt.show() #显示图像
```

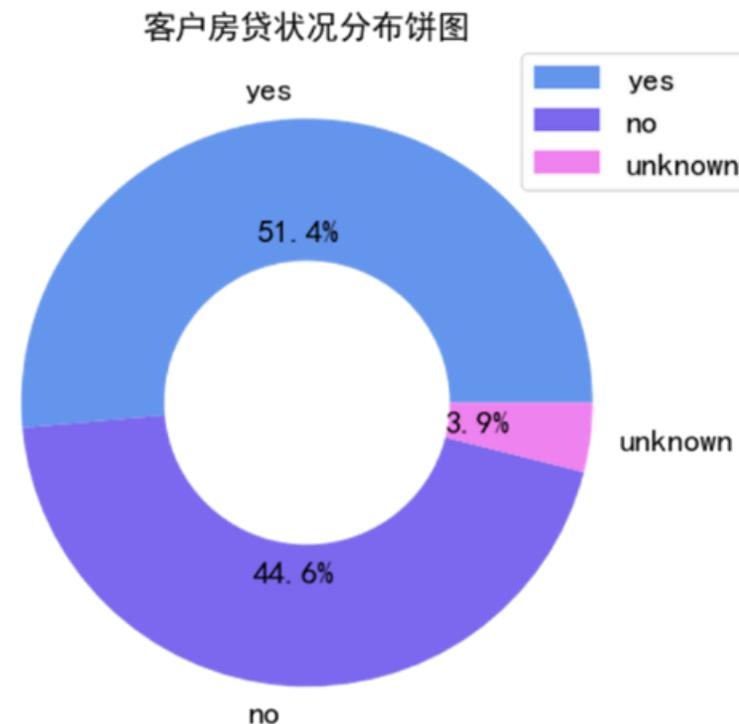




#### (4) 客户房贷状况分析

数据集中客户是“否有房贷”字段有yes、no、unknown三个类别，想得知客户群体中承担房贷人群的分布比例，通过Matpoylib库中的pie()函数绘制环形图。

```
plt.figure(figsize=(12,8), dpi=72) #设置图片大小
plt.title('房贷状况统计饼图',fontsize=20) #设置标题，并将字体大小设置为20
y = data['是否有房贷'].value_counts() #对“是否有房贷”字段进行分类计数
plt.pie(y,
        labels=y.index, #设置标签为房贷状况的3个类别
        autopct='%3.1f%%', #显示小数点后1位
        colors=['cornflowerblue','mediumslateblue','violet'], #设置环形图颜色
        textprops={'fontsize':20, 'color':'black'}, #设置标签字体大小及颜色
        wedgeprops={'width':0.5}) #设置环形图宽度
plt.legend(loc=[0.8,0.8],fontsize=20) #设置图例，指定位置为右上角
plt.show() #显示图像
```





## (5) 客户违约情况分析

客户违约情况字段有yes、no、unknown三个类别，可以利用玫瑰图进行可视化。玫瑰图通过每块扇形代表数值的大小，面积越大则表示数值越大，可以有效地凸显不同类别之间的差异。利用pyecharts.charts模块中Pie()函数进行绘制。

```

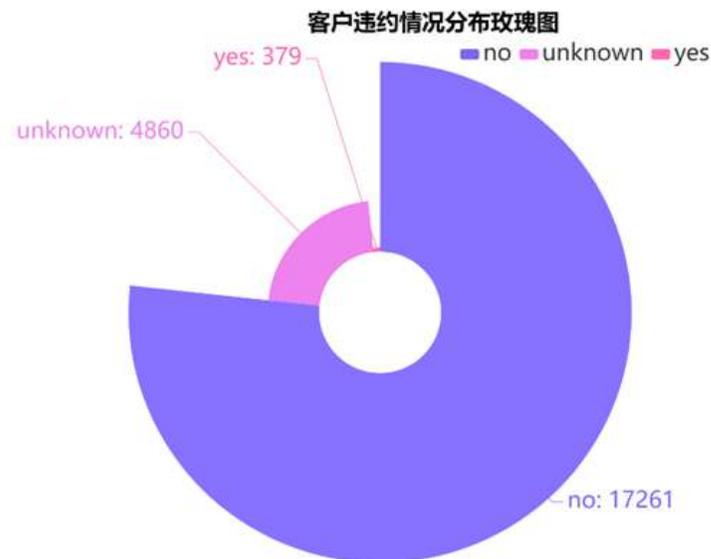
from pyecharts.charts import Pie #导入pyecharts.charts模块用于绘图
from pyecharts import options as opts #导入options模块设置格式

y = data['是否违约'].value_counts() #对违约情况进行计数

pie1 = Pie(init_opts=opts.InitOpts(width='1000px', #设置图像大小及背景颜色
                                   height='800px', bg_color='white'))

pie1.add(series_name = '是否违约',
          data_pair = [list(i) for i in zip(y.index,y)], #设置数据
          radius=['20%', '80%'], #设置环形大小
          center=['50%', '50%'], #设置玫瑰图在画布上的位置
          rosetype='radius' #将图片展示为玫瑰图
).set_series_opts(label_opts=opts.LabelOpts(position='poutside', #将标签显示位置设置于图外部
                                             formatter='{b}: {c}', font_size=25) #设置标签文字格式
).set_global_opts(title_opts=opts.TitleOpts(title='违约情况玫瑰图', #设置图像标题
                                             pos_left='400', #设置标题左右位置
                                             pos_top='20', #设置标题上下位置
                                             #设置标题字体颜色及大小
                                             title_textstyle_opts=opts.TextStyleOpts(color='black', font_size=25)),
                  legend_opts=opts.LegendOpts(is_show=True, #设置显示图例
                                             pos_left='600', #设置图例左右位置
                                             pos_top='2', #设置图例上下位置
                                             textstyle_opts=opts.TextStyleOpts(font_size=25)) #设置图例大小
).set_colors(['mediumslateblue', 'violet', 'hotpink'] #设置图形颜色
).render('违约情况玫瑰图.html') #设置文件名称

```



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：  
<https://d.book118.com/068031123014006063>