Data Mining and Analysis

Course Design Report

Title : Analysing shop sales data

Name: Lin Jing

Faculty: Aliyun Big Data Application Academy

Instructor: Li Tongfei

Date: 2021-06-08

CONTENTS

Abstracts

In today's era of rapid technological development, the growth of information is also increasing exponentially. Once upon a time, shops sold goods with paper records, but nowadays it is converted into a short piece of data and stored in the computer. So in front of such a huge amount of data, the experience of the operation and maintenance staff to make decisions to improve sales performance is obviously a thing of the past. The use of data mining data analysis, through pandas, matplotlib and other methods will be a large amount of data through the form of charts and graphs in real time, in order to better put forward effective recommendations to improve the efficiency of shop operations and profitability. We also use discrete point detection to find out the outliers in the data and analyse in detail the origin of the outliers.

Keywords: data mining; graphs; pandas; matplotlib; discrete point detection;

I. General introduction

1.1 Background to the issue

This data is derived from the detailed sales data of a shop over a four-year period from 2015 to 2018. In order to improve the overall profit amount and O&M cost of the shop, what is reflected behind the data is mined here by analysing the sales, profit amount, cross-analysis, discount analysis and other methods from various dimensions.

1.2 Description of data

This case uses a total of 9959 rows and 21 columns of data. The names of the columns are 'Row ID', 'Order ID', 'Order Date', 'Ship Date', 'Mailing Method', 'Customer ID', 'Customer Name', 'Segmentation', 'City', 'Province/Autonomous Region', 'Country', 'Region', 'Product ID', 'Category', 'Sub-Category', 'Product Name', 'Sales', 'Quantity', 'Discount', 'Profit', 'District Manager'., 'Area Manager'. Exclude the unwanted column names as 'Row ID', 'Customer ID', 'Product ID', 'Product Name'.

1.3 Key technologies required

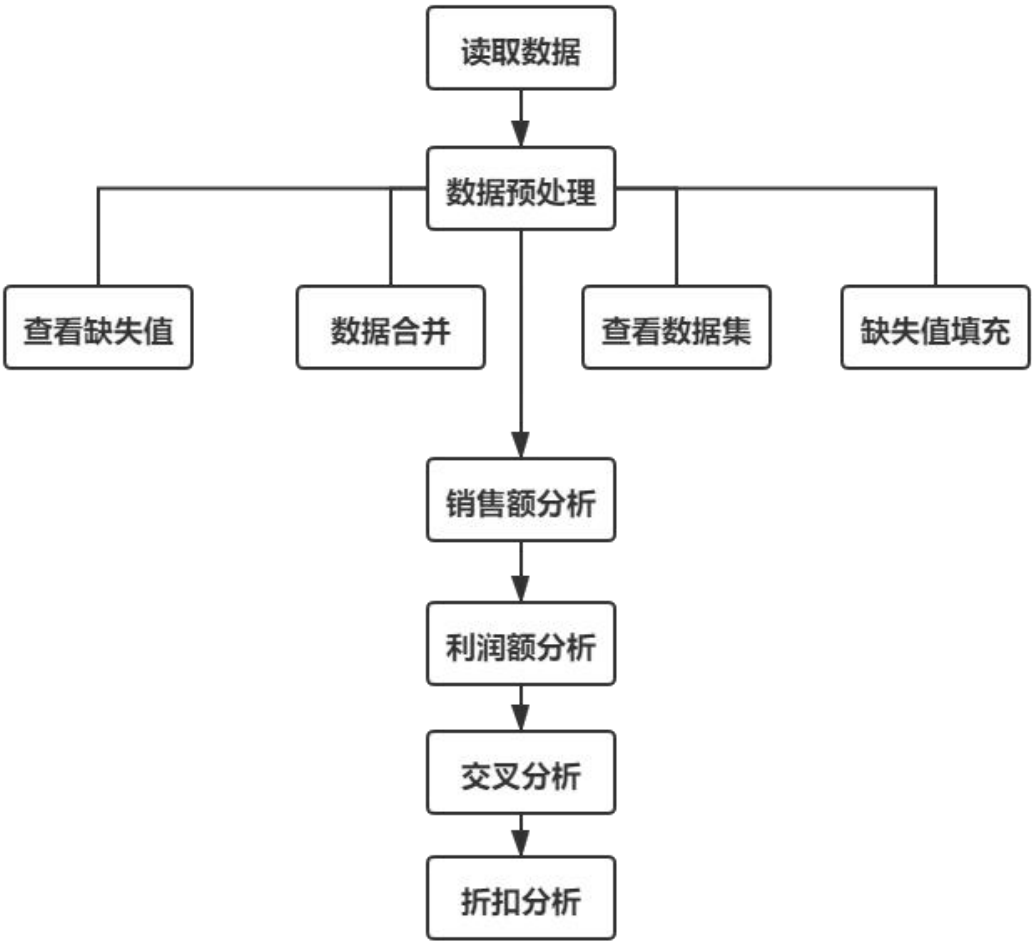
In this paper, we need to acquire data, clean and process the data, analyse it, chart it, etc. The following techniques are used in this process:

(1) Pandas: pandas is a tool based on NumPy, which was created to solve the task of data analysis. pandas incorporates a large number of libraries and a number of standard data models, which provide the tools needed to efficiently manipulate large datasets. pandas provides a large number of functions and methods that enable us to quickly and easily work with data. As you will soon discover, it is one of the key factors that make Python a powerful and efficient environment for analysing data.

(2) Matplotlib: Matplotlib is an external module for Python that provides plotting functionality. It was originally intended to mimic MatLab, so Matplotlib provides two interfaces for making graphs, one for object oriented programming, and one for the MatLab-like interface pyplot.

(3) Seaborn: Seaborn is a graphical visualisation python package based on matplotlib. It provides a highly interactive interface that facilitates the user to be able to make a variety of attractive statistical charts.

II. Analysing flow charts



III. Data analysis3.1 读取数据

It is divided into two tables, the first one is the orders table with 9959 data and 20 fields, and the second one is the salesperson table with six data and two fields.

The data is processed to delete the indicators that are not needed in this article, and the two tables are merged. As shown in Figure 3-1, 3-2.

```
In [2]:
1 data1 = pd.read_excel('data\data.xls',sheet_name='订单')
2 data2 = pd.read_excel('data\data.xls',sheet_name='销售人员')

In [3]:
1 data1.shape

Out[3]: (9959, 20)

In [4]:
1 data2.shape

Out[4]: (6, 2)

In [5]:
1 data1.columns

Out[5]: Index(['行 ID', '订单 ID', '订单日期', '发货日期', '邮寄方式', '客户 ID', '客户名称', '细分', '城市',
       '省/自治区', '国家', '地区', '产品 ID', '类别', '子类别', '产品名称', '销售额', '数量', '折扣',
       '利润'],
      dtype='object')

In [6]:
1 data2.columns

Out[6]: Index(['地区', '地区经理'], dtype='object')

In [7]:
1 data3=data1.drop(['行 ID','客户 ID','产品 ID','产品名称'],axis=1)
```

Figure 3-1

```
In [11]:
1 df=pd.merge(data3,data2,how='left',left_on='地区',right_on='地区')

In [12]:
1 df.shape

Out[12]: (9959, 17)

In [13]:
1 df.head()
```

Out[13]:

	订单 ID	订单日期	发货日期	邮寄方式	客户名称	细分	城市	省/自治区	国家	地区	类别	子类别	销售额	数量	折扣	利润	地区经理
0	US-2018-1357144	2018-04-27	2018-04-29	二级	曾惠	公司	杭州	浙江	中国	华东	办公用品	用品	129.696	2	0.4	-60.704	洪光
1	CN-2018-1973789	2018-06-15	2018-06-19	标准级	许安	消费者	内江	四川	中国	西南	办公用品	信封	125.440	2	0.0	42.560	白德伟
2	CN-2018-1973789	2018-06-15	2018-06-19	标准级	许安	消费者	内江	四川	中国	西南	办公用品	装订机	31.920	2	0.4	4.200	白德伟
3	US-2018-3017568	2018-12-09	2018-12-13	标准级	宋良	公司	镇江	江苏	中国	华东	办公用品	用品	321.216	4	0.4	-27.104	洪光
4	CN-2017-2975416	2017-05-31	2017-06-02	二级	万兰	消费者	汕头	广东	中国	中南	办公用品	器具	1375.920	3	0.0	550.200	范彩

Figure 3-2

3.2 Data preprocessing

3.2.1 View Missing Values

View the numerical data whether there are missing values, if there is a mean fill in the way to fill in the missing values. In this case, there are three missing values in the sales data, and the missing value is 0 after the mean value is filled, as shown in Figure 3-3.

```
In [14]:  1 df.isnull().sum()

Out[14]: 订单 ID      0
         订单日期      0
         发货日期      0
         邮寄方式      0
         客户名称      0
         细分        0
         城市        0
         省/自治区     0
         国家        0
         地区        0
         类别        0
         子类别       0
         销售额       3
         数量        0
         折扣        0
         利润        0
         地区经理      0
         dtype: int64

In [23]:  1 df=df.fillna(df['销售额'].mean())
          2 df.isnull().sum()

Out[23]: 订单 ID      0
         订单日期      0
         发货日期      0
         邮寄方式      0
         客户名称      0
         细分        0
         城市        0
         省/自治区     0
         国家        0
         地区        0
         类别        0
         子类别       0
         销售额       0
         数量        0
```

Figure 3-3

3.2.2 View a statistical description of the dataset

View various statistical characteristics of the full dataset, such as mean, variance, minimum maximum, and so on, as shown in Figure 3-4.

```
In [24]:  1 df.describe()
```

Out[24]:

	销售额	数量	折扣	利润
count	9959.000000	9959.000000	9959.000000	9959.000000
mean	1613.846855	3.768852	0.106406	215.638008
std	2641.165302	2.236739	0.187477	858.710532
min	13.440000	1.000000	0.000000	-7978.320000
25%	250.460000	2.000000	0.000000	7.756000
50%	637.000000	3.000000	0.000000	74.200000
75%	1785.210000	5.000000	0.200000	277.200000
max	35621.355000	14.000000	0.800000	10108.280000

Figure 3-4

3.3 Sales analysis

The total sales by year can be seen that sales are rising year by year and the rate of growth is getting bigger and bigger, which is good news for the mall and shows that the influence and social status of the mall is gradually expanding. It is believed that the sales can be doubled in 2019 relative to 2015, as shown in Figure 3-5.

In [26]:

```
plt.rcParams['font.sans-serif']=['Simhei']
seals_year = df['销售额'].groupby(df['订单日期'].dt.year).sum()
plt.bar(seals_year.index,height=seals_year.values,width=0.8,alpha = 0.5,color = 'b')
plt.xticks(seals_year.index,labels=['2015年','2016年','2017年','2018年'])
plt.ylabel('销售额（元）')
for a,b in zip(seals_year.index,seals_year.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b,ha='center', va= 'bottom',fontsize=10)
```

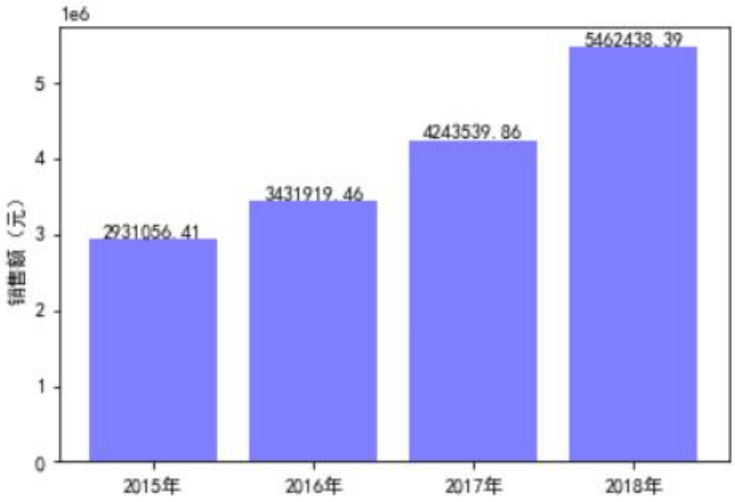


Figure 3-5

The total sales of provinces can be seen that Shandong, Guangdong and Heilongjiang are the top three provinces in terms of sales, followed by Liaoning and Henan. Guangdong and Shandong are both the top three provinces in terms of GDP in 2018, while Heilongjiang is ranked 22nd, but it is not normal for this shop to have so many sales in Heilongjiang. Maybe the shop has something that the people in Heilongjiang province need more, and the operators can focus on analysing how the items sold in the shop are related to the Heilongjiang province. As shown in Figure 3-6.

In [27]:

```
seals_pro= df['销售额'].groupby(df['省/自治区']).sum().sort_values()
plt.figure(figsize=(15,12))
plt.barh(seals_pro.index,width=seals_pro.values,height=0.5,alpha = 0.5,color = 'b')
plt.ylabel('省份')
plt.xlabel('销售额（元）')
for y, x in enumerate(seals_pro.values):
    plt.text(x+500, y-0.2, "%.2f" %x)
```

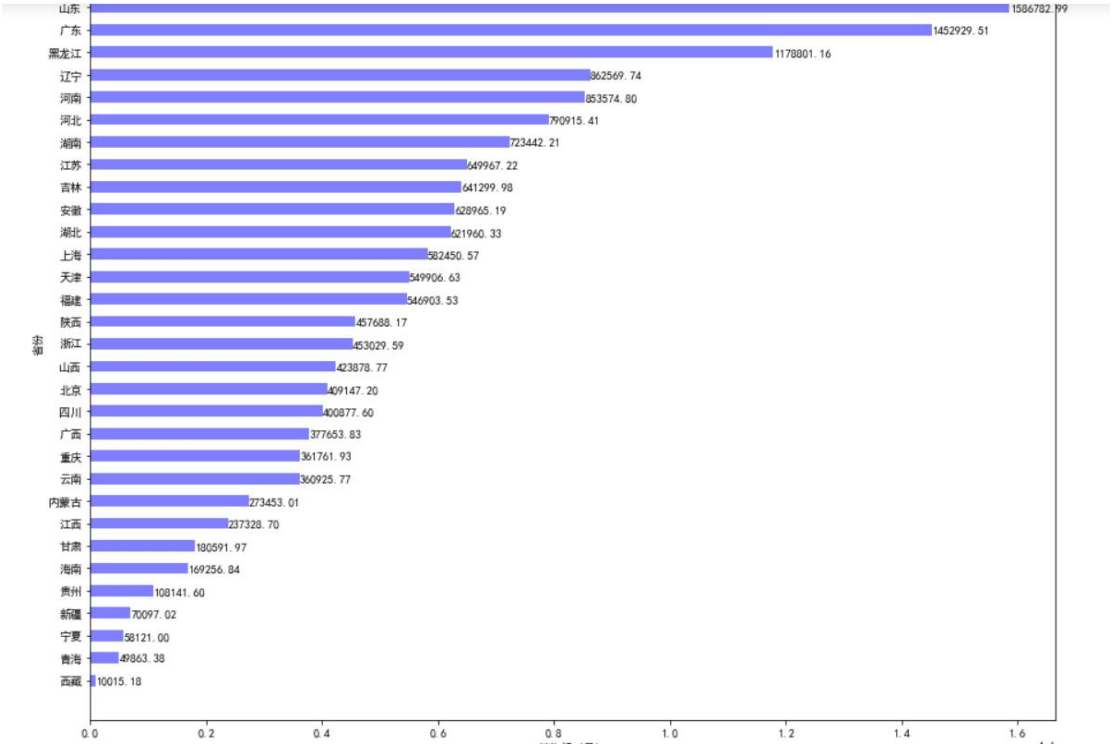


Figure 3-6

Then look at the sales staff's contribution to the sales volume. Hong Guang and Fan Cai managed by the region sales degree is the highest, can give them some performance incentives to motivate them to continue to work hard, while Yang Jian and Bai Dewei is the lowest amount, which may be related to the economic level of Northwest and Southwest region, consumption is not high so the total sales degree will be relatively low, we can consider a different sales strategy, with promotions and other ways to promote their consumption, increase sales. As shown in Figure 3-7.

In [28]:

```
seals_man= df['销售额'].groupby(df['地区经理']).sum().sort_values()
plt.bar(seals_man.index,height=seals_man.values,width=0.8,alpha = 0.5,color = 'r')
plt.ylabel('销售额（元）')
for a,b in zip(seals_man.index,seals_man.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
```

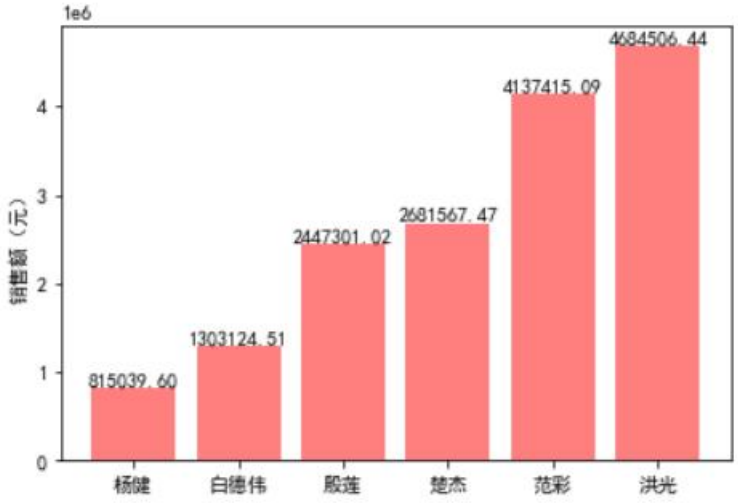


Figure 3-8 shows that furniture items such as chairs utensils bookshelves have the best sales degree, which is caused by the high demand and the unit price is not cheap.

In [75]:

```
1 seals_kind= df['销售额'].groupby(df['子类别']).sum().sort_values()
2 plt.bar(seals_kind.index,height=seals_kind.values,width=0.5,alpha = 0.5,color = 'g')
3 plt.ylabel('销售额（元）')
4 for a,b in zip(seals_kind.index,seals_kind.values): # 添加数据标签
5     plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
```

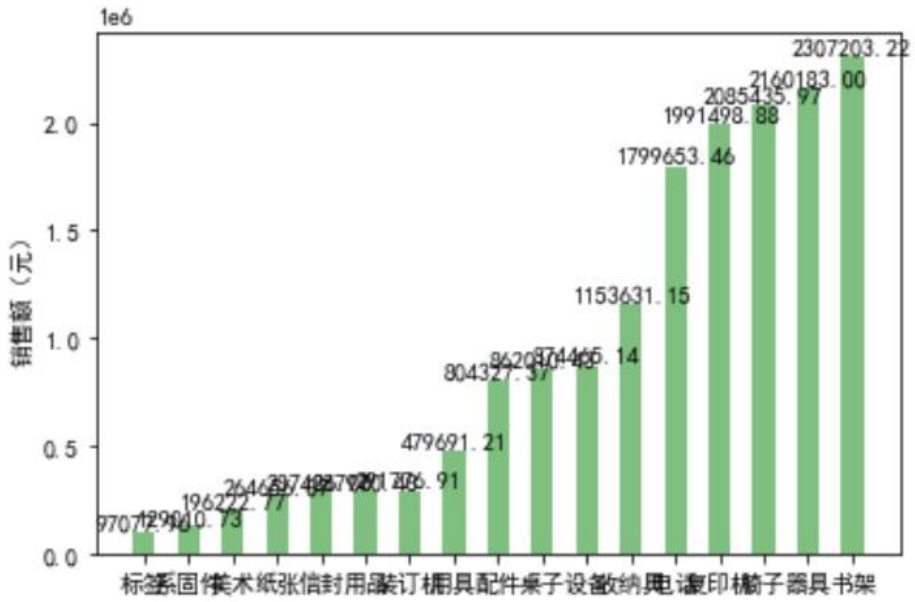


Figure 3-8

The impact of postal method on sales volume shows that most of them use standard class post and fewer use same day post, so it is possible to reach a favourable agreement with the courier company to make it cheaper whenever standard class

post is used. As shown in Figure 3-9.

```
In [76]:
seals_t= df['销售额'].groupby(df['邮寄方式']).sum().sort_values()
plt.bar(seals_t.index,height=seals_t.values,width=0.5,alpha = 0.5,color = 'r')
plt.ylabel('销售额（元）')
for a,b in zip(seals_t.index,seals_t.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=8)
```

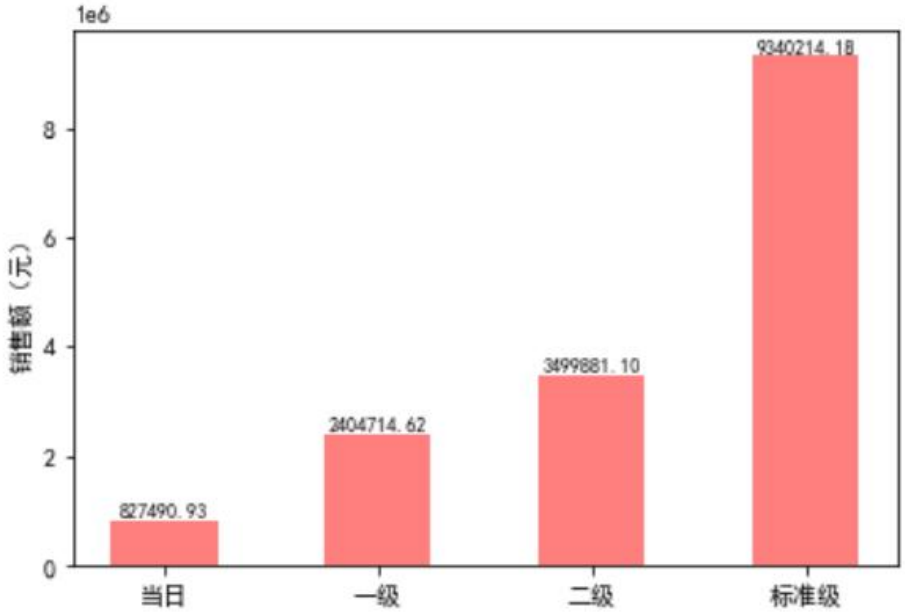


Figure 3-9

3.4 Profitability analysis

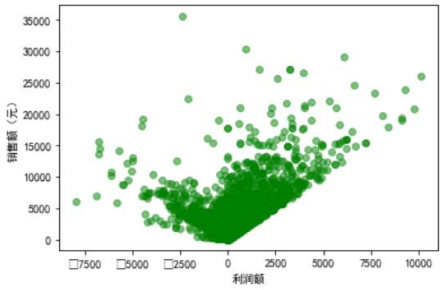
3.4.1 Discrete point detection

Firstly, check if there is any linear relationship between sales and profit, plot sales and profit on a scatter plot and find that there is no particular linear relationship and there is a discrete point. Sales are very high but the profit is a loss, which is very unfavourable to the operation of the shop, the query found to be a kind of table, each sold at a loss of 2375 yuan, so you can consider the price of this product is higher or no longer sell this product. In order to stop the loss in time. As shown in Figure 3-10.

```
In [24]:
1 plt.scatter(x=df['利润'],y=df['销售额'],alpha=0.5,color = 'g')
2 plt.xlabel('利润额')
3 plt.ylabel('销售额（元）')

Out[24]: Text(0, 0.5, '销售额（元）')

C:\Users\25010\AppData\Roaming\Python\Python37\site-packages\matplotlib\backends\backend_agg.py:238: RuntimeWarning: Glyph
current font.
  font.set_text(s, 0.0, flags=flags)
C:\Users\25010\AppData\Roaming\Python\Python37\site-packages\matplotlib\backends\backend_agg.py:201: RuntimeWarning: Glyph
current font.
  font.set_text(s, 0, flags=flags)
```




```
In [27]:
1 df['销售额'].max()

Out[27]: 35621.354999999996
```

Figure 3-10

3.4.2 Change in the amount of profit over time

This can be shown through Figure 3-11. The amount of profit is increasing but the growth rate of 2017-2018 is decreasing, not like the sales where the growth rate has been getting bigger, this may be due to the fact that most of the goods sold during the year are not actually profitable, later on the marketers can develop new selling strategies to ensure that the growth rate is increasing.

```
In [28]:
1  prf = df['利润'].groupby(df['订单日期'].dt.year).sum()
2  plt.bar(prf.index,height=prf.values,width=0.8,alpha = 0.5,color = 'b')
3  plt.xticks(prf.index,labels=['2015年','2016年','2017年','2018年'])
4  plt.ylabel('利润额（元）')
5  for a,b in zip(prf.index,prf.values): # 添加数据标签
6      plt.text(a, b+0.05,
7               '%.2f' % b,
8               ha='center',
9               va= 'bottom',
10              fontsize=10)
```

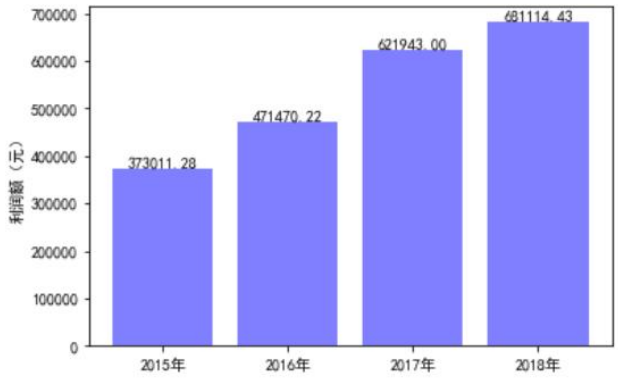


Figure 3-11

3.4.3 Impact of sales force on profitability

As can be seen in Figure 3-12. Although Hong Quang is greater than Pham Cai in terms of sales, the actual amount of profit is indeed greater for Pham Cai than for Hong Quang, so for the contribution of the employees to the company, it is actually Pham Cai who is a little bit more outstanding and is the number one in the amount of profit even though he does not have as many sales.

```
In [34]:
prf_man= df['利润'].groupby(df['地区经理']).sum().sort_values()
plt.bar(prf_man.index,height=prf_man.values,width=0.8,alpha = 0.5,color = 'r')
plt.ylabel('利润额（元）')
for a,b in zip(prf_man.index,prf_man.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
```

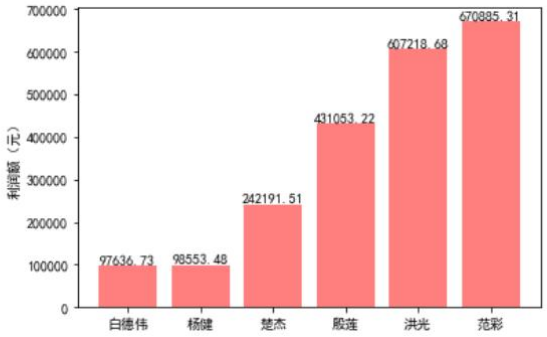


Figure 3-13

3.4.4 Profitability of commodity groups

All of the items are basically positive in terms of profit amount, only the table and the art knife are at a loss, and the table is at a huge loss. This has to do with the cost of buying, and selling at a lower price will naturally be a loss. It is recommended to raise the price of the table if you still want to sell it. As shown in Figure 3-14.

In [29]:

```
prf_kind= df['利润'].groupby(df['子类别']).sum().sort_values()
plt.bar(prf_kind.index,height=prf_kind.values,width=0.5,alpha = 0.5,color = 'g')
plt.ylabel('利润额（元）')
for a,b in zip(prf_kind.index,prf_kind.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
```

```
C:\Users\25010\AppData\Roaming\Python\Python37\site-packages\matplotlib\backends\backend_agg.p
current font.
  font.set_text(s, 0.0, flags=flags)
C:\Users\25010\AppData\Roaming\Python\Python37\site-packages\matplotlib\backends\backend_agg.p
current font.
  font.set_text(s, 0, flags=flags)
```

Figure 3-14

3.5 cross-tabulation analysis

3.5.1 Cross-tabulation of provinces, categories and profits

Through the cross-analysis table of provincial category profits can be seen, Inner Mongolia, Sichuan, Zhejiang, Hubei, Gansu, Liaoning and other regions of the three commodity category profits are negative, which represents these areas of the business is the most unprofitable, and need to immediately change the strategy in order to make the profitability can be increased in the future. As shown in Figure 3-15.

类别	办公用品	家具	技术
省/自治区			
上海	35229.460	40216.428	46204.200
云南	20973.820	42095.592	23569.756
内蒙古	-15815.184	-24077.928	-17814.776
北京	24618.860	42062.580	25280.500
吉林	26063.436	50022.735	76972.000
四川	-6415.164	-45722.796	-37349.564
天津	38900.960	27610.170	51192.960
宁夏	7177.100	-2732.380	4092.900
安徽	43450.120	44779.210	60799.480
山东	128908.920	123901.988	132652.100
山西	27574.540	54483.030	25005.820
广东	97420.260	120960.133	119614.600
广西	31990.644	18530.085	34203.540
新疆	10644.200	1118.600	2843.260
江苏	1968.176	-76255.340	-33315.856
江西	8801.660	22714.580	16290.820
河北	61986.680	48722.625	61322.380
河南	40575.696	55416.543	103536.440
浙江	-24613.064	-53659.088	-53456.844
海南	12818.176	12201.875	14702.072
湖北	-20040.496	-47230.260	-64761.592
湖南	61279.764	44644.285	50811.880
甘肃	-190.708	-25144.420	-17347.064
福建	21840.420	61755.652	59005.660
西藏	908.040	358.540	NaN
贵州	8217.300	6644.540	4136.580
辽宁	-25614.736	-67684.064	-74739.924
重庆	21359.660	20986.756	22085.336
陕西	27876.800	45376.128	32561.760
青海	4126.080	6285.860	1865.360
黑龙江	85618.932	90353.970	81199.160

Figure 3-15

This is converted to a heat map to visualise the amount of profit in each category for each region. It can be seen that Shanxi Guangdong has high profits in all categories. As shown in Figure 3-16.

```
In [39]:
plt.figure(figsize=(10,10), dpi= 80)
#cmap = sns.cubehelix_palette(start = 1.5, rot = 3, gamma=0.8, as_cmap = True)
sns.heatmap(tem,annot=True,annot_kws={'size':9,'weight':'bold', 'color':'black'},cmap='rainbow')
plt.show()
```

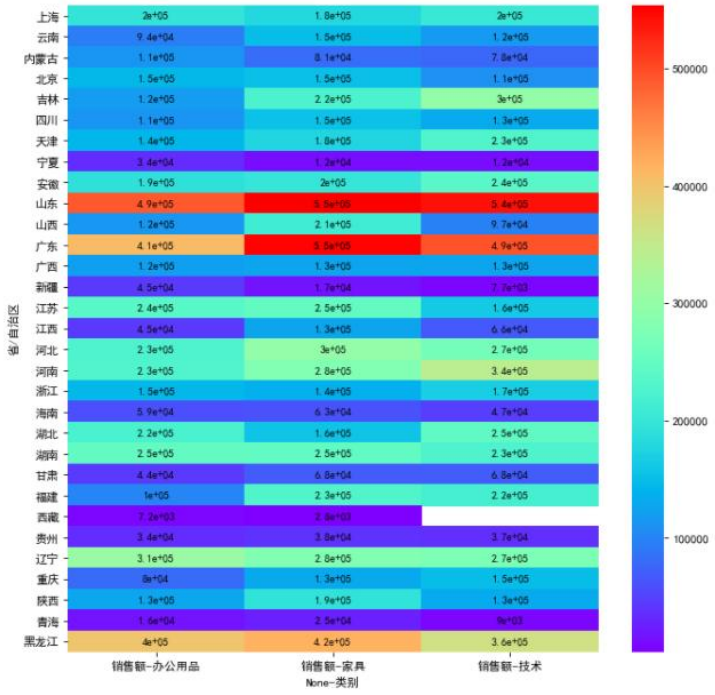


Figure 3-16

3.5.2 Area, Manager, Profit Cross-tabulation

Plotting the regions, managers, and profits in a cross-tabulation allows you to visualise the profits and sales of each regional manager's corresponding region for each commodity category. Northwest office supplies have a higher amount of profit, the Northeast is more average, North China furniture is more profitable, and East China is more technologically advanced because of Shanghai, a technologically advanced city. The amount of profit in Southwest China are less, because of the remoteness of the place, and South Central China is higher, especially for technical goods. As shown in Figure 3-17.

In [44]:
```
tem5 = pd.pivot_table(data=df,values=['利润','销售额'],index=['地区经理','地区','类别'],aggfunc='sum')
tem5
```

Out[44]:

地区经理	地区	类别	利润	销售额
杨健	西北	办公用品	49633.472	2.691927e+05
		家具	24903.788	3.162124e+05
		技术	24016.216	2.309564e+05
楚杰	东北	办公用品	86067.632	8.257765e+05
		家具	72692.641	9.206984e+05
		技术	83431.236	9.361960e+05
殷莲	华北	办公用品	137265.856	7.458135e+05
		家具	148800.477	9.197439e+05
		技术	144986.884	7.817436e+05
洪光	华东	办公用品	215585.692	1.408629e+06
		家具	163453.430	1.676224e+06
		技术	228179.560	1.600575e+06
白德伟	西南	办公用品	49272.216	3.476926e+05
		家具	29303.092	5.015337e+05
		技术	19061.420	4.538982e+05
范彩	中南	办公用品	219815.484	1.270911e+06
		家具	199582.201	1.399928e+06
		技术	251487.628	1.466576e+06

Figure 3-17

3.6 Discount analysis

3.6.1 Category Discounts

Tables are the most discounted for all categories of supplies, which is why the profit on tables is negative, the discount is too high and the consequences are not taken into account. The discount on tables should be reduced in the future. This is shown in Figure 3-18.

In [33]:
```
disct_kid= df['折扣额'].groupby(df['子类别']).sum().sort_values()
plt.bar(disct_kid.index,height=disct_kid.values,width=0.8,alpha = 0.5,color = 'r')
plt.ylabel('折扣总额（元）')
for a,b in zip(disct_kid.index,disct_kid.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
```

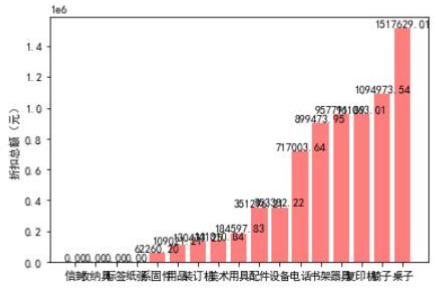


Figure 3-18

3.6.2 Provincial Discounts

The discounts given to each province are different, with the biggest discounts being given to Liaoning, Jiangsu, Hubei, Zhejiang, and Sichuan. These provinces

enjoy very high discounts compared to others. However, Figure 3-16 shows that although these cities have high discounts and high sales, their profits are negative, which is equivalent to charity business. If the shop does not deal with this issue of activity strength in time, the profit amount will be lower and lower in the future. As shown in Figure 3-19.

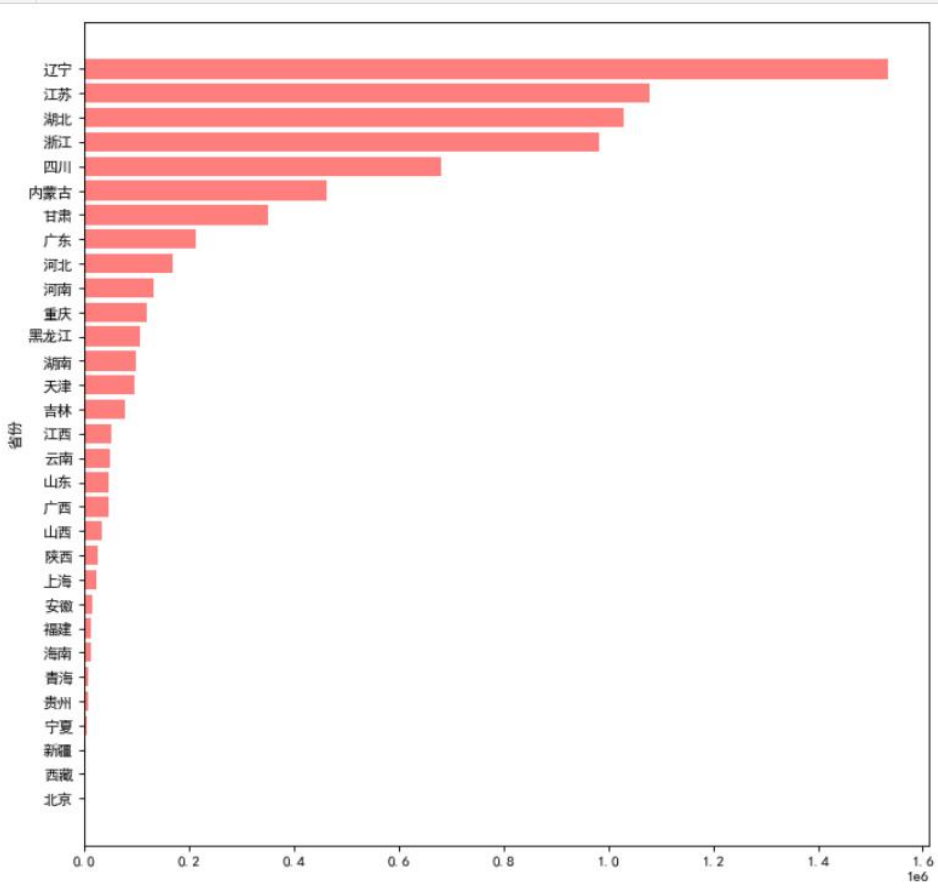


Figure 3-19

IV. Summary

Although we analyse the knowledge of a simple shop sales data, but it is more important to look through the phenomenon to see the essence. For a shop can not only look at its sales, but also combined with the cost of integrated down to see the final profit. First of all, we should make awards to a few employees with high profits, and hold a meeting to see how to deal with the southwest and northwest of the low-profit areas of the corresponding personnel and activities to adjust the strategy. Secondly, we need to adjust the discounts and original prices of goods, especially the price of the table, and its need to be adjusted. Because relative to other categories of goods will not have so much loss, but

the table is the only loss of the most commodities. There is also the fact that the activities in several cities that have been losing money need to be rectified, not as much as the current discounts, otherwise it will only remain in a loss-making state. It's not about selling more that means good, it's about focusing on actual revenue. Cities with high profit margins can consider providing more supply of goods and manpower, maintaining good relationships with customers, and offering some promotional activities from time to time to continue to expand sales and profit margins. This will allow the shop to make higher overall profits and margins in the years to come and avoid sales that would result in a loss.

V. Personal reflections

After this data mining project more let me realise the importance of data mining and data analysis in the business field, if you want to get a better conclusion through human statistical analysis need to spend a lot of manpower and time, but if the use of python in the pandas, matplotlib and other libraries to do data mining data analysis, will be very convenient. Using a variety of data processing methods, and data analysis methods, cross-combination of various indicators for comprehensive analysis, and show the form of graphs can be the fastest and most intuitive for the analysts to see some of the strengths and weaknesses hidden behind the data. This also strengthens my will to learn this knowledge in the future.

Acknowledgements

I would like to thank Mr Li Tongfei for his teaching in the Python Data Analytics and Enterprise Big Data courses. Without the teacher's vivid classroom, I would not have the passion to learn data analysis. In this data analysis course design, I have encountered a lot of trouble, but flipping through the content of the courseware used by the teacher in class, thinking of the knowledge taught by the teacher in class, combined with all kinds of information on the Internet, all the problems were solved as if the water pump was opened. I hope that in the future study, I can always keep the heart of a child, with a pious and modest attitude to face all the problems, so as to walk up the ladder of success!

VI. Appendix

运行环境 anaconda jupyter notebook。

```
import pandas  as pd
import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns
data1 = pd.read_excel('data\data.xls',sheet_name='订单')
data2 = pd.read_excel('data\data.xls',sheet_name='销售人员')
data1.shape
data2.shape
data1.columns
data2.columns
data3=data1.drop(['行 ID','客户 ID','产品 ID','产品名称'],axis=1)
data3.head()
data3.shape
data2.head(10)
df=pd.merge(data3,data2,how='left',left_on='地区',right_on='地区')
df.shape

df.head()
df.sample(10)
df.isnull().sum()
df=df.fillna(df['销售额'].mean())
df.isnull().sum()
df.describe()
plt.rcParams['font.sans-serif']=['Simhei']
seals_year = df['销售额'].groupby(df['订单日期'].dt.year).sum()
plt.bar(seals_year.index,height=seals_year.values,width=0.8,alpha = 0.5,color = 'b')
plt.xticks(seals_year.index,labels=['2015 年','2016 年','2017 年','2018 年'])
plt.ylabel('销售额（元）')
for a,b in zip(seals_year.index,seals_year.values): # 添加数据标签
        plt.text(a, b+0.05,'%.2f' % b,ha='center', va= 'bottom',fontsize=10)
        seals_pro= df['销售额'].groupby(df['省/自治区']).sum().sort_values()
        plt.figure(figsize=(15,12))
        plt.barh(seals_pro.index,width=seals_pro.values,height=0.5,alpha = 0.5,color = 'b')
        plt.ylabel('省份')
        plt.xlabel('销售额（元）')
        for y, x in enumerate(seals_pro.values):
            plt.text(x+500, y-0.2, "%.2f" %x)
        seals_man= df['销售额'].groupby(df['地区经理']).sum().sort_values()
        plt.bar(seals_man.index,height=seals_man.values,width=0.8,alpha = 0.5,color = 'r')
        plt.ylabel('销售额（元）')
        for a,b in zip(seals_man.index,seals_man.values): # 添加数据标签
            plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
        seals_kind= df['销售额'].groupby(df['子类别']).sum().sort_values()
        plt.bar(seals_kind.index,height=seals_kind.values,width=0.5,alpha = 0.5,color = 'g')
        plt.ylabel('销售额（元）')
        for a,b in zip(seals_kind.index,seals_kind.values): # 添加数据标签
            plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
        seals_t= df['销售额'].groupby(df['邮寄方式']).sum().sort_values()
        plt.bar(seals_t.index,height=seals_t.values,width=0.5,alpha = 0.5,color = 'r')
        plt.ylabel('销售额（元）')
        for a,b in zip(seals_t.index,seals_t.values): # 添加数据标签
            plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=8)
        plt.scatter(x=df['利润'],y=df['销售额'],alpha=0.5,color = 'g')
```

```
plt.xlabel('利润额')
plt.ylabel('销售额（元）')
df['销售额'].max()
prf = df['利润'].groupby(df['订单日期'].dt.year).sum()
plt.bar(prf.index,height=prf.values,width=0.8,alpha = 0.5,color = 'b')
plt.xticks(prf.index,labels=['2015 年','2016 年','2017 年','2018 年'])
plt.ylabel('利润额（元）')
for a,b in zip(prf.index,prf.values): # 添加数据标签
    plt.text(a, b+0.05,
             '%.2f' % b,
             ha='center',
             va= 'bottom',
             fontsize=10)
prf_man= df['利润'].groupby(df['地区经理']).sum().sort_values()
plt.bar(prf_man.index,height=prf_man.values,width=0.8,alpha = 0.5,color = 'r')
plt.ylabel('利润额（元）')
for a,b in zip(prf_man.index,prf_man.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
prf_p= df['利润'].groupby(df['省/自治区']).sum().sort_values()
plt.figure(figsize=(10,12))
plt.barh(prf_p.index,width=prf_p.values,height=0.5,alpha = 0.5,color = 'b')
plt.ylabel('省份')
plt.xlabel('利润额（元）')
prf_kind= df['利润'].groupby(df['子类别']).sum().sort_values()
plt.bar(prf_kind.index,height=prf_kind.values,width=0.5,alpha = 0.5,color = 'g')
plt.ylabel('利润额（元）')
for a,b in zip(prf_kind.index,prf_kind.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
tem = pd.pivot_table(data=df,values=['销售额'],index=['省/自治区'],columns=['类别'],aggfunc='sum')
Tem
plt.figure(figsize=(10,10), dpi= 80)
#cmap = sns.cubehelix_palette(start = 1.5, rot = 3, gamma=0.8, as_cmap = True)
sns.heatmap(tem,annot=True,annot_kws={'size':9,'weight':'bold',
'color':'black'},cmap='rainbow')
plt.show()
tem5 = pd.pivot_table(data=df,values=['利润','销售额'],index=['地区经理','地区','类别'],aggfunc='sum')
tem5
df['折扣额']=df['折扣']*df['销售额']*df['数量']
disct_kid= df['折扣额'].groupby(df['子类别']).sum().sort_values()
plt.bar(disct_kid.index,height=disct_kid.values,width=0.8,alpha = 0.5,color = 'r')
plt.ylabel('折扣总额（元）')
for a,b in zip(disct_kid.index,disct_kid.values): # 添加数据标签
    plt.text(a, b+0.05,'%.2f' % b, ha='center',va= 'bottom',fontsize=10)
disct_pro= df['折扣额'].groupby(df['省/自治区']).sum().sort_values()
plt.figure(figsize=(10,10), dpi= 80)
plt.barh(disct_pro.index,width=disct_pro.values,height=0.8,alpha = 0.5,color = 'r')
plt.ylabel('省份')
plt.show()
```