

这是一份数据分析报告
针对某电子商品销售数据，进行探索性数据分析，用户画像和用户分群，通过分析找到门店长处与短板，给出参考建议，并用仪表盘展示

为了完成这份分析，我们将围绕以下几个部分进行分析

- 描述统计
- 消费人群分析&用户画像
- 产品分析
- 销售分析
 - 订单销售额分析
 - 用户分层

```
# from pylab import mpl
# mpl.rcParams['font.sans-serif'] = ['SimHei']
# mpl.rcParams['axes.unicode_minus'] = False
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import re
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
plt.rcParams['font.sans-serif']=['Microsoft YaHei']
```

```
data = pd.read_csv('/Users/orangeli/huxin/电子产品销售/电子产品销售分析.csv',encoding='utf-8',index_col=0,low_memory=False)
data.head()
```

```
.dataframe tbody tr th {
    vertical-align: top;
}

.dataframe thead th {
    text-align: right;
}
```

| | event_time | order_id | product_id | category_id | category_code | brand | price | |
|---|-------------------------|---------------------|---------------------|--------------|-----------------------------|---------|--------|------|
| 0 | 2020-04-24 11:50:39 UTC | 2294359932054536986 | 1515966223509089906 | 2.268105e+18 | electronics.tablet | samsung | 162.01 | 1.51 |
| 1 | 2020-04-24 11:50:39 UTC | 2294359932054536986 | 1515966223509089906 | 2.268105e+18 | electronics.tablet | samsung | 162.01 | 1.51 |
| 2 | 2020-04-24 14:37:43 UTC | 2294444024058086220 | 2273948319057183658 | 2.268105e+18 | electronics.audio.headphone | huawei | 77.52 | 1.51 |
| 3 | 2020-04-24 14:37:43 UTC | 2294444024058086220 | 2273948319057183658 | 2.268105e+18 | electronics.audio.headphone | huawei | 77.52 | 1.51 |
| 4 | 2020-04-24 19:16:21 UTC | 2294584263154074236 | 2273948316817424439 | 2.268105e+18 | NaN | karcher | 217.57 | 1.51 |

- event_time: 下单时间
- user_id: 用户编号
- order_id: 订单编号
- product_id: 产品编号
- category_id: 类别编号
- brand: 品牌
- price: 购买金额
- age: 年龄
- sex: 性别
- local: 省份

一、探索性数据分析

```
data.info()

<class 'pandas.core.frame.DataFrame'>
Index: 564169 entries, 0 to 2633520
```

```
Data columns (total 11 columns):
#      Column      Non-Null Count  Dtype
---  -
0      event_time   564169 non-null   object
1      order_id     564169 non-null   object
2      product_id   564169 non-null   int64
3      category_id  564169 non-null   float64
4      category_code 434799 non-null   object
5      brand        536945 non-null   object
6      price        564169 non-null   float64
7      user_id      564169 non-null   float64
8      age         564169 non-null   float64
9      sex         564169 non-null   object
10     local        564169 non-null   object
dtypes: float64(4), int64(1), object(6)
memory usage: 51.7+ MB
```

```
data.describe()
```

```
.dataframe tbody tr th {
    vertical-align: top;
}

.dataframe thead th {
    text-align: right;
}
```

| | product_id | category_id | price | user_id | age |
|-------|--------------|--------------|---------------|--------------|---------------|
| count | 5.641690e+05 | 5.641690e+05 | 564169.000000 | 5.641690e+05 | 564169.000000 |
| mean | 1.695711e+18 | 2.272919e+18 | 208.269324 | 1.515916e+18 | 33.184388 |
| std | 3.290688e+17 | 2.158282e+16 | 304.559875 | 2.377083e+07 | 10.122088 |
| min | 1.515966e+18 | 2.268105e+18 | 0.000000 | 1.515916e+18 | 16.000000 |
| 25% | 1.515966e+18 | 2.268105e+18 | 23.130000 | 1.515916e+18 | 24.000000 |
| 50% | 1.515966e+18 | 2.268105e+18 | 87.940000 | 1.515916e+18 | 33.000000 |
| 75% | 1.515966e+18 | 2.268105e+18 | 277.750000 | 1.515916e+18 | 42.000000 |
| max | 2.388434e+18 | 2.374499e+18 | 18328.680000 | 1.515916e+18 | 50.000000 |

- 订单金额平均消费208元，中位数在304元，说明用户消费金额相对比较稳定，但存在一定极值干扰

```
del data['category_code']
data.fillna('no_brand',inplace = True)
data = data.drop_duplicates('order_id')
data['event_time'] = data['event_time'].str.replace('1970','2020')
data['event_time'] = pd.to_datetime(data['event_time']).dt.to_period('D')
data = data.set_index('event_time',drop = False)
data['month'] = data['event_time'].dt.month
data.head(5)
```

```
.dataframe tbody tr th {
    vertical-align: top;
}

.dataframe thead th {
    text-align: right;
}
```

| | event_time | order_id | product_id | category_id | brand | price | user_id | age | sex | local | mon |
|------------|------------|---------------------|---------------------|--------------|---------|---------|--------------|------|-----|-------|-----|
| event_time | | | | | | | | | | | |
| 2020-04-24 | 2020-04-24 | 2294359932054536986 | 1515966223509089906 | 2.268105e+18 | samsung | 162.01 | 1.515916e+18 | 24.0 | 女 | 海南 | 4 |
| 2020-04-24 | 2020-04-24 | 2294444024058086220 | 2273948319057183658 | 2.268105e+18 | huawei | 77.52 | 1.515916e+18 | 38.0 | 女 | 北京 | 4 |
| 2020-04-24 | 2020-04-24 | 2294584263154074236 | 2273948316817424439 | 2.268105e+18 | karcher | 217.57 | 1.515916e+18 | 32.0 | 女 | 广东 | 4 |
| 2020-04-26 | 2020-04-26 | 2295716521449619559 | 1515966223509261697 | 2.268105e+18 | maestro | 39.33 | 1.515916e+18 | 20.0 | 男 | 重庆 | 4 |
| 2020-04-26 | 2020-04-26 | 2295740594749702229 | 1515966223509104892 | 2.268105e+18 | apple | 1387.01 | 1.515916e+18 | 21.0 | 男 | 北京 | 4 |

二、消费人群分析&用户画像

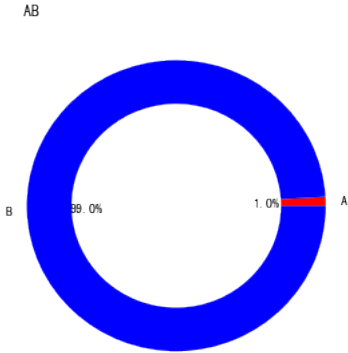
```
#按照每个用户的订单数，对用户进行分层，订单数>30，用户为A，订单数<30，用户为B
user_order=data.groupby('user_id')['order_id'].count().sort_values(ascending=False)
data_b = data.loc[data['user_id'].isin(user_order[user_order<30].index)]
data_b['user_level']='B'
data_a=data.loc[~data['user_id'].isin(list(data_b['user_id']))]
data_a['user_level']='A'
data_ab = pd.concat([data_a,data_b])
data_ab.head(5)
```

```
.dataframe tbody tr th {
    vertical-align: top;
}

.dataframe thead th {
    text-align: right;
}
```

| | event_time | order_id | product_id | category_id | brand | price | user_id | age | sex | local | mont |
|------------|------------|---------------------|---------------------|--------------|----------|--------|--------------|------|-----|-------|------|
| event_time | | | | | | | | | | | |
| 2020-04-29 | 2020-04-29 | 2297988436574864215 | 1515966223509089486 | 2.268105e+18 | samsung | 115.72 | 1.515916e+18 | 47.0 | 女 | 重庆 | 4 |
| 2020-04-29 | 2020-04-29 | 2298027408663511168 | 1515966223510206441 | 2.268105e+18 | no_brand | 2.52 | 1.515916e+18 | 31.0 | 女 | 上海 | 4 |
| 2020-04-29 | 2020-04-29 | 2298114976126075117 | 1515966223509127845 | 2.268105e+18 | varta | 2.08 | 1.515916e+18 | 33.0 | 男 | 北京 | 4 |
| 2020-04-30 | 2020-04-30 | 2298477621916205325 | 1515966223509104136 | 2.268105e+18 | apple | 23.13 | 1.515916e+18 | 33.0 | 男 | 广东 | 4 |
| 2020-04-30 | 2020-04-30 | 2298590293420670991 | 1515966223509089483 | 2.268105e+18 | huawei | 92.34 | 1.515916e+18 | 19.0 | 女 | 江苏 | 4 |

```
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号
user_level_patio = data_ab.groupby('user_level')['user_id'].nunique()
plt.figure(figsize=(5,5))
lables = ['A','B']
sizes = [1,99]
colors = ['red','blue']
plt.title ('AB类客户占比图')
plt.pie(sizes,lables= lables, colors= colors,autopct='%1.1f%%',wedgeprops={'width':0.3})
plt.show()
```



```
level_user = data_ab.groupby('user_level')['order_id'].count()
level_price = data_ab.groupby('user_level')['price'].sum()
level_user_price = pd.concat([level_user,level_price],axis=1)
level_user_price
```

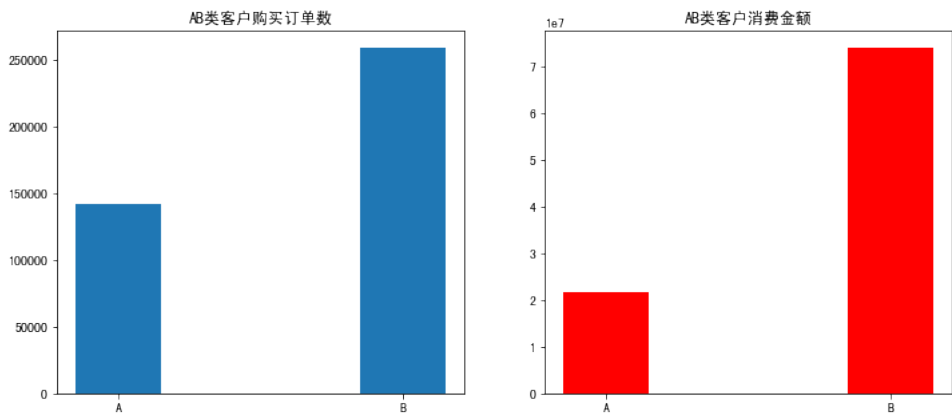
```
.dataframe tbody tr th {
    vertical-align: top;
}

.dataframe thead th {
    text-align: right;
}
```

| | order_id | price |
|------------|----------|--------------|
| user_level | | |
| A | 141636 | 2.170747e+07 |
| B | 258902 | 7.404613e+07 |

```
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号
fig,axs = plt.subplots(1,2,figsize=(12,5))
x=['A','B']
plt.title('AB类客户订单数')
level_user_x = level_user.index
level_user_y = level_user.values
level_price_x = level_price.index
level_price_y = level_price.values
bar_width = 0.3
ax1 = plt.subplot(1,2,1)
plt.title('AB类客户购买订单数')
plt.bar(level_user_x,level_user_y,bar_width)
ax2 = plt.subplot(1,2,2)
plt.title('AB类客户消费金额')
plt.bar(level_price_x,level_price_y,bar_width,facecolor = 'red')
```

<BarContainer object of 2 artists>



- 从上图发现，A类客户以1%的数量贡献了20%以上的销售额，可能存在异常数据，否则A类客户应被视为重点客户维系
- B类客户数量占比99%，下面有B类客户做重点分析

```
df = data_ab[data_ab.user_level == 'B']
df.info()
```

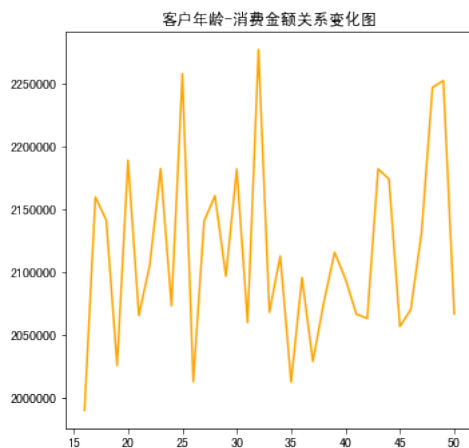
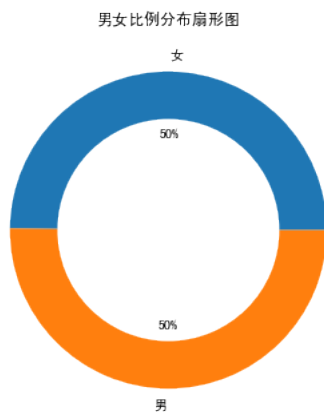
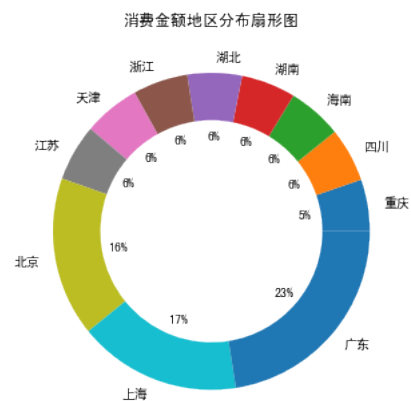
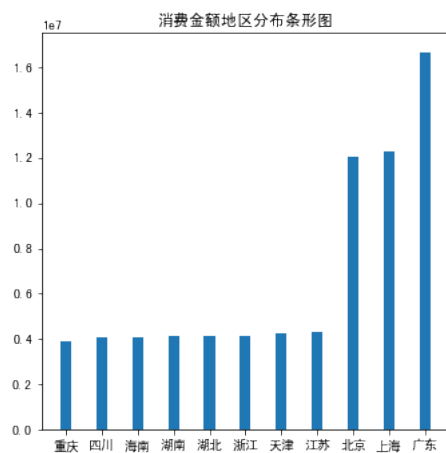
```
<class 'pandas.core.frame.DataFrame'>
PeriodIndex: 258902 entries, 2020-04-24 to 2020-11-21
Freq: D
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   event_time  258902 non-null  period[D]
1   order_id    258902 non-null  object
2   product_id  258902 non-null  int64
3   category_id 258902 non-null  float64
4   brand       258902 non-null  object
5   price       258902 non-null  float64
6   user_id     258902 non-null  float64
7   age         258902 non-null  float64
8   sex         258902 non-null  object
9   local       258902 non-null  object
10  month       258902 non-null  int64
11  user_level  258902 non-null  object
dtypes: float64(4), int64(2), object(5), period[D](1)
memory usage: 25.7+ MB
```

对B类客户做用户画像分析

```
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号
fig,axs = plt.subplots(2,2,figsize = (12,12))
```

```
ax1 = plt.subplot(2,2,1)
local_price = df.groupby('local')['price'].sum().sort_values()
lable1 = local_price.index
plt.bar(local_price.index,local_price.values,width = 0.3)
plt.title('消费金额地区分布条形图')
ax2 = plt.subplot(2,2,2)
plt.pie(local_price,labels=lable1,autopct='%0f%%',wedgeprops={'width':0.3})
plt.title('消费金额地区分布扇形图')
ax3 = plt.subplot(2,2,3)
sex_count = df.groupby('sex')['user_id'].nunique().rename('人数')
lable2 = sex_count.index
plt.pie(sex_count,labels=lable2,autopct='%0f%%',wedgeprops={'width':0.3})
plt.title('男女比例分布扇形图')
ax4 = plt.subplot(2,2,4)
age_price = df.groupby('age')['price'].sum()
lable3 = age_price.index
plt.ylabel='消费金额'
plt.xlabel = '年龄'
plt.plot(age_price,color = 'orange')
plt.title('客户年龄-消费金额关系变化图')
```

```
Text(0.5, 1.0, '客户年龄-消费金额关系变化图')
```



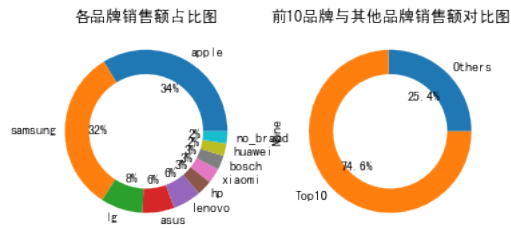
用户画像分析

- 北上广消费能力远超其他地区，消费金额是其他地区的2-3倍，且占了销售总额的一半以上，营销策略可以向北上广地区倾斜
- 对比消费人群的性别和年龄，可能存在异常数据，无其他明显特征

三、产品分析

```
brand_price = data_ab.groupby('brand')['price'].sum().sort_values(ascending=False).head(10)
top_sales = brand_price.sum()
ax1 = plt.subplot(1,2,1)
lable4 = brand_price.index
plt.title('各品牌销售额占比图')
plt.pie(brand_price,labels= lable4, autopct='%0f%%',wedgeprops={'width':0.3})
ax2 = plt.subplot(1,2,2)
plt.title('前10品牌与其他品牌销售额对比图')
pd.Series({'Others':data_ab['price'].sum()-top_sales,'Top10':top_sales}).plot(kind = 'pie',autopct='%0.1f%%',wedgeprops={'width':0.3},radius =1)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8b3ecd3b50>
```



结果与建议

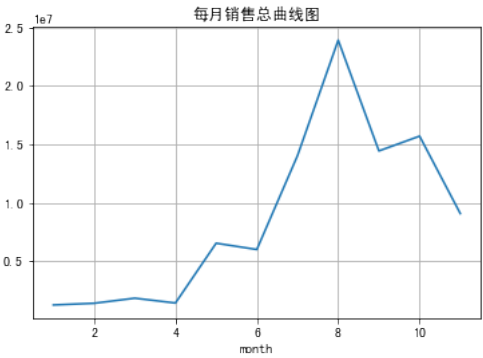
- 2020年手机品牌市场销售份额主要集中在前两大品牌三星和苹果，销售额占比为65%
- 对比2021年手机品牌销售市场份额（数据来源于网上）前三为华为、OPPO、小米，证明国产品牌崛起，可以深挖国产品牌机会，提高销售额

四、销售额分析

- 多维度交叉分析
- 用户分层

```
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号
plt.title('每月销售总曲线图')
data_ab.groupby('month').price.sum().plot(grid=True)
```

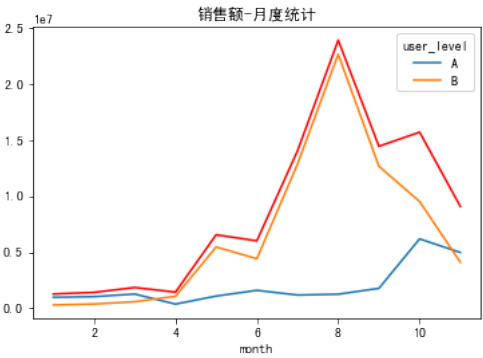
<matplotlib.axes._subplots.AxesSubplot at 0x7f8b2f808150>



2020年初销量较低，从4月份开始显著增长，8月快速下滑，下滑率达59%
下面从多维度对比分析，对销售额下滑原因进行分析挖掘，

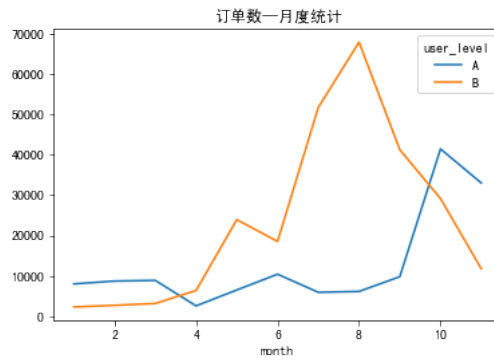
```
data_ab.groupby(['month','user_level'])['price'].sum().unstack().plot()
data_ab.groupby('month')['price'].sum().rename('A+B').plot(color='r')
plt.title('销售额-月度统计')
```

Text(0.5, 1.0, '销售额-月度统计')



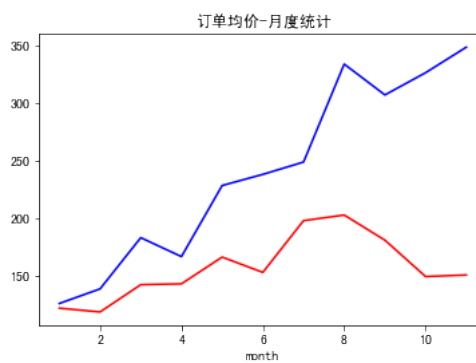
```
data_ab.groupby(['month','user_level'])['order_id'].count().unstack().plot()
plt.title('订单数-月度统计')
```

Text(0.5, 1.0, '订单数-月度统计')



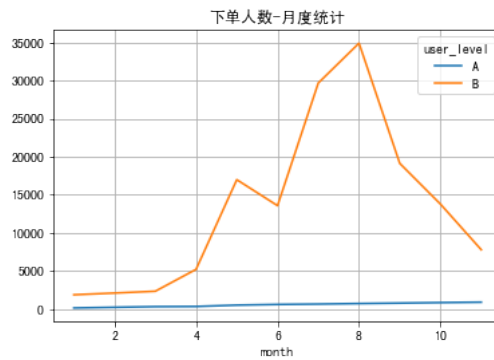
```
(data_a.groupby('month')['price'].sum()/data_a.groupby('month')['order_id'].count()).rename('A').plot(color='red')
(data_b.groupby('month')['price'].sum()/data_b.groupby('month')['order_id'].count()).rename('B').plot(color='blue')
plt.title('订单均价-月度统计')
```

Text(0.5, 1.0, '订单均价-月度统计')



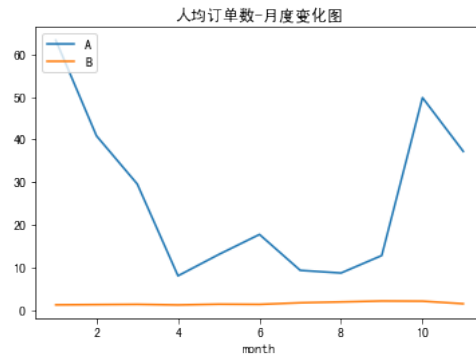
```
data_ab.groupby(['month', 'user_level']).user_id.nunique().unstack().plot(grid=True)
plt.title('下单人数-月度统计')
```

Text(0.5, 1.0, '下单人数-月度统计')



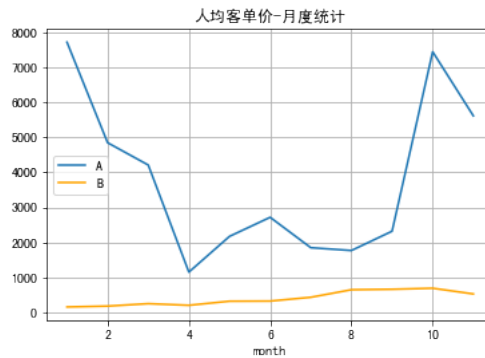
```
(data_a.groupby('month')['order_id'].count()/data_a.groupby('month').user_id.nunique()).rename('A').plot(grid=True).legend(loc=1)
(data_b.groupby('month')['order_id'].count()/data_b.groupby('month').user_id.nunique()).rename('B').plot().legend(loc=2)
plt.title('人均订单数-月度变化图')
```

Text(0.5, 1.0, '人均订单数-月度变化图')



```
(data_a.groupby('month')['price'].sum()/data_a.groupby('month').user_id.nunique()).rename('A').plot(grid=True).legend(loc=1)
(data_b.groupby('month')['price'].sum()/data_b.groupby('month').user_id.nunique()).rename('B').plot(grid=True,color='orange').legend(loc=6)
plt.title('人均客单价-月度统计')
```

Text(0.5, 1.0, '人均客单价-月度统计')



结论与建议：

- B类客户下单人数、订单量下降是造成销售额下降的主要原因
- B类客户的订单均价和人均客单价一直保持上升趋势
- B类客户的人均订单数和客单价也在稳步上升
- 从订单数和下单人数可以发现，4-8月有显现的流量导入，8月以后流量较少，同时会员的价值在不断释放

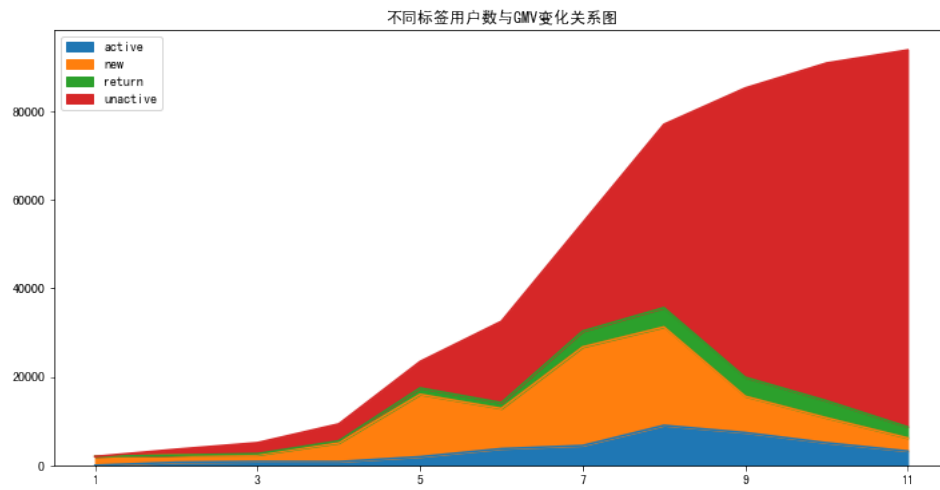
#用户分层

```
def active_status(x):
    status = []
    for i in range(11):
        if x[i] == 0:
            if i == 0:
                status.append('unreg')
            else:
                status.append('unreg') if status[i-1] == 'unreg' else status.append('unactive')
        else:
            if i == 0:
                status.append('new')
            else:
                if status[i-1] == 'unreg':
                    status.append('new')
                else:
                    status.append('return') if status[i-1] == 'unactive' else status.append('active')
    return pd.Series(status, index = columns_month)

columns_month = data['month'].sort_values().astype('str').unique()
order_record = data.pivot_table(index = 'user_id', columns='month', values = 'price', aggfunc='sum').fillna(0).applymap(lambda i: 1 if i>0 else 0)
order_record.columns = data['month'].sort_values().astype('str').unique()
user_active_status = order_record.apply(active_status,axis = 1)
```

```
user_active_status.apply(lambda x: pd.value_counts(x)).drop('unreg').fillna(0).T.plot(kind='area',figsize=(12,6))
plt.title('不同标签用户数与GMV变化关系图')
```

Text(0.5, 1.0, '不同标签用户数与GMV变化关系图')



- 4-8月销售额增加主要是新用户的持续导入
- 8月以后新用户数急剧下降，不活跃用户急剧增加，销售额大幅下滑
- 有部分会员成功转化为活跃会员，这也是在11月新用降至4月水平时，销售远高4月的重要原因
- 可以复盘8月后新用户降低的原因，同时从ROI的角度复盘4-8月运营策略，以及策略是否可以持续