共享单车数据集可视化

1 数据集选取

https://www.kaggle.com/pronto/cycle-share-dataset

2014-2016年间，西雅图的Pronto自行车共享系统由500辆自行车和54个车站组成，此数据集提供了在此期间的租用订单信息、共享单车信息以及当地的每日天气数据。

提出问题：

- 租用共享单车用户类型、性别、年龄分布情况如何?
- 用户特征、租用时间、天气情况等因素是否影响共享单车的租用，是如何影响的?

2 数据类型分析

2.1 导入工具包

```
import pandas as pd   # 导入数据框处理工具包

import datetime      # 导入处理时间工具包

import matplotlib.pyplot as plt       # 导入matplotlib工具包中绘图函数pyplot
%matplotlib inline
plt.style.use('ggplot') # 选择画图风格
plt.rcParams['font.sans-serif'] = ['Arial Unicode MS']

import warnings          # 忽略警告提示
warnings.filterwarnings('ignore')

from pylab import mpl
mpl.rcParams['font.sans-serif'] =['SimHei']     # 指定默认中文字体
mpl.rcParams['axes.unicode_minus'] = False     # 解决保存图像中负号‘-’显示为方块的问题
```

2.2 导入数据

```
trip = pd.read_csv("trip.csv",error_bad_lines=False, encoding = 'utf8', sep = ',')
weather = pd.read_csv("weather.csv",error_bad_lines=False, encoding = 'utf8', sep =
',')
```

数据存储格式为CSV文件，用记事本方式打开数据集并设置为中文编码（utf8）。

2.3 查看数据集信息

对数据集进行概览，查看字段数据类型及缺失值情况，如存在缺失值，应对缺失值加以处理再进行分析。

```
weather.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 689 entries, 0 to 688
Data columns (total 21 columns):
 #   Column                     Non-Null Count  Dtype  
---  ------                     --------------  -----  
 0   Date                       689 non-null    object 
 1   Max_Temperature_F          689 non-null    int64  
 2   Mean_Temperature_F         688 non-null    float64
 3   Min_TemperatureF           689 non-null    int64  
 4   Max_Dew_Point_F            689 non-null    int64  
 5   MeanDew_Point_F            689 non-null    int64  
 6   Min_Dewpoint_F             689 non-null    int64  
 7   Max_Humidity               689 non-null    int64  
 8   Mean_Humidity              689 non-null    int64  
 9   Min_Humidity               689 non-null    int64  
 10  Max_Sea_Level_Pressure_In  689 non-null    float64
 11  Mean_Sea_Level_Pressure_In 689 non-null    float64
 12  Min_Sea_Level_Pressure_In  689 non-null    float64
 13  Max_Visibility_Miles       689 non-null    int64  
 14  Mean_Visibility_Miles      689 non-null    int64  
 15  Min_Visibility_Miles       689 non-null    int64  
 16  Max_Wind_Speed_MPH         689 non-null    int64  
 17  Mean_Wind_Speed_MPH        689 non-null    int64  
 18  Max_Gust_Speed_MPH         504 non-null    object 
 19  Precipitation_In           689 non-null    float64
 20  Events                     328 non-null    object 
dtypes: float64(5), int64(13), object(3)
memory usage: 113.2+ KB
```

```
trip.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 286857 entries, 0 to 286856
Data columns (total 12 columns):
 #   Column             Non-Null Count   Dtype  
---  ------             --------------   -----  
 0   trip_id            286857 non-null  int64  
 1   starttime          286857 non-null  object 
 2   stoptime           286857 non-null  object 
 3   bikeid             286857 non-null  object 
 4   tripduration       286857 non-null  float64
 5   from_station_name  286857 non-null  object 
 6   to_station_name    286857 non-null  object 
 7   from_station_id    286857 non-null  object 
 8   to_station_id      286857 non-null  object 
 9   usertype           286857 non-null  object 
 10  gender             181557 non-null  object 
 11  birthyear          181553 non-null  float64
dtypes: float64(2), int64(1), object(9)
memory usage: 26.3+ MB
```

2.4 删除冗余字段

删除冗余字段并重命名重要字段，减少内存的占用，加快数据处理速度，增加数据的可读性。

```
weather.drop(['Max_Temperature_F','Min_TemperatureF',
              'Max_Dew_Point_F','Min_Dewpoint_F',
              'Max_Humidity','Min_Humidity',
              'Max_Sea_Level_Pressure_In','Min_Sea_Level_Pressure_In',
              'Max_Visibility_Miles','Min_Visibility_Miles',
              'Max_Wind_Speed_MPH','Max_Gust_Speed_MPH','Events'],axis = 1,inplace =
True)

weather.rename(columns = {'Mean_Temperature_F':'Temperature',
                          'MeanDew_Point_F':'Dew_Point','Mean_Humidity':'humidity',
                          'Mean_Sea_Level_Pressure_In':'Sea_Pressure',
                          'Mean_Visibility_Miles':'Visibility_Miles',
                          'Mean_Wind_Speed_MPH':'Wind_Speed'},inplace = True)
```

```
trip.drop(['tripduration','from_station_name','to_station_name',
           'bikeid','from_station_id','to_station_id'],axis = 1,inplace = True)
```

2.5 字段含义说明

weather数据集字段含义：

- Date：日期
- Temperature：温度
- Dew_Point：露水值
- humidity：湿度
- Sea_Pressure：海平面气压
- Visibility_Miles：能见度
- Wind_Speed：风速
- Precipitation_In：降水量

trip数据集字段含义：

- trip_id：订单编号
- starttime：订单开始时间
- stoptime：订单结束时间
- usertype：用户类型
- gender：用户性别
- birthyear：用户出生日期

3 数据预处理

通过 2.3 可知，weather数据集中Date字段，trip数据集中starttime、stoptime字段均为object类型，需转换为时间序列。同时为trip创建新的字段，新增日期、年、月、日、小时、星期六个字段。

```
 #  转换为时间序列
trip['starttime'] = pd.to_datetime(trip.starttime)
trip['stoptime'] = pd.to_datetime(trip.stoptime)
weather['Date'] = pd.to_datetime(weather.Date)
 #  创建新的字段
trip['date'] = trip.starttime.astype('datetime64[D]')
trip['year'] = trip.date.apply(lambda x: x.year)
trip['month'] = trip.date.apply(lambda x: x.month)
trip['day'] = trip.date.apply(lambda x: x.day)
trip['hour'] = trip.starttime.apply(lambda x: x.hour)
trip['weekday'] = trip.starttime.apply(lambda x: x.weekday())
```

trip数据集中其他数据完整，而gender和birthyear两个字段存在空值且数量相等，通过与trip总数据概览对比发现，缺失值均来自临时用户数据。为对用户年龄分层，删除存在空值的行，并将birthyear转换成int型数据。

```
groupby_age = trip.dropna()
groupby_age['birthyear'] = groupby_age.birthyear.astype('int64')
groupby_age.info()
```

自定义函数对用户年龄分层，增加分层年龄age字段。

```
def func(x):
    if x > 1996:
        return '0-20'
    elif 1986 < x <= 1996:
        return '20-30'
    elif 1976 < x <= 1986:
        return '30-40'
    elif 1966 < x <= 1976:
        return '40-50'
    elif 1956 < x <= 1966:
        return '50-60'
    else :
        return '60+'

groupby_age['age'] = groupby_age.birthyear.apply(func)
```

4 数据可视化分析

4.1 用户维度

```
fig = plt.subplots(figsize=(16,12))

ax1 = plt.subplot2grid((2,2), (0,0),colspan=1)
df1 = trip.groupby(by = ['usertype']).usertype.count()
df1.plot.pie(startangle =90,autopct = '%.1f%%',ax =ax1)
ax1.set_title('2014-2016西雅图Pronto共享单车用户类型占比')
```

```
ax2 = plt.subplot2grid((2,2), (0,1),colspan=2)
df2 = trip.groupby(by = ['gender']).gender.count()
df2.plot.pie(startangle =90,autopct = '%.1f%%',ax =ax2)
ax2.set_title('2014-2016西雅图Pronto共享单车用户性别占比')

ax3 = plt.subplot2grid((2,2), (1,0),colspan=2)
df3 = groupby_age.groupby(by = ['age']).age.count()
df3.plot.bar(ax =ax3)
ax3.set_title('2014-2016西雅图Pronto共享单车用户年龄段占比')
```

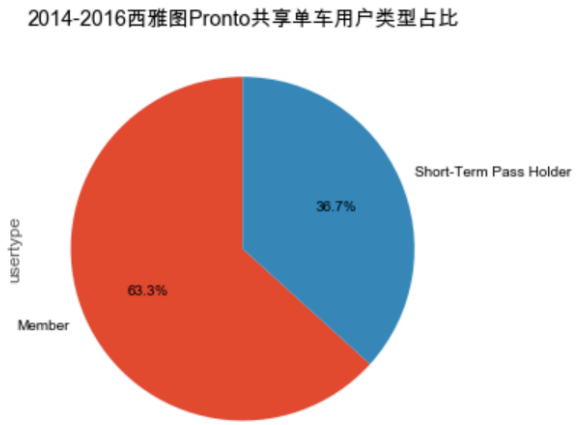


图1

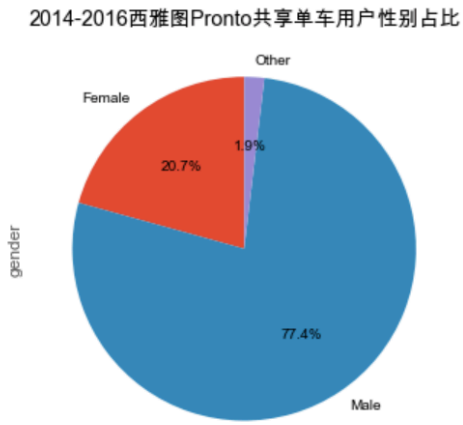


图2

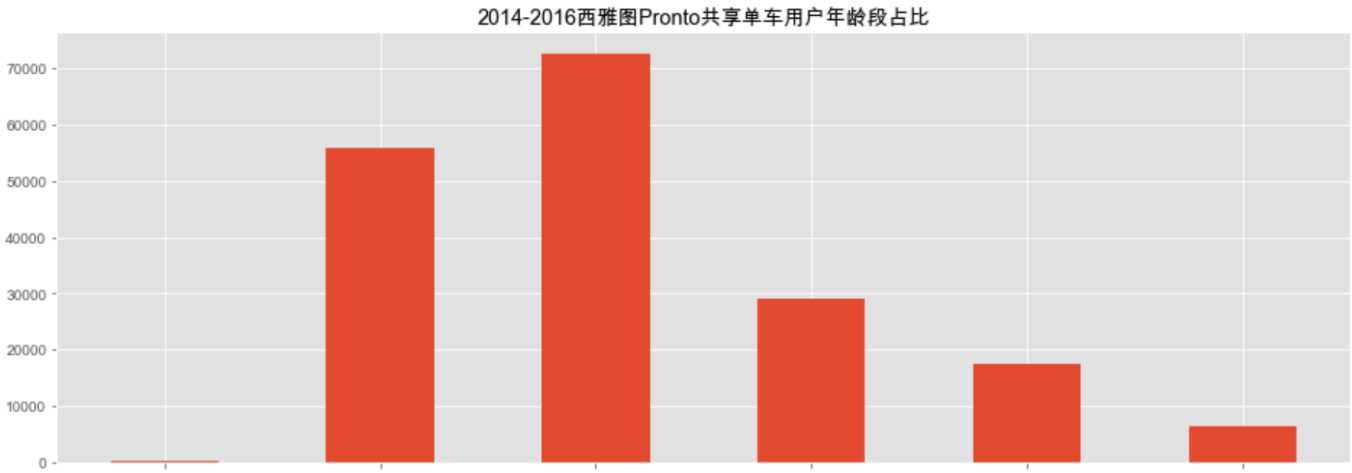


图3

结果分析说明

从图1可以看出,会员租车人数比非会员租车人数多，大概比例为6：4左右。

从图2可以看出，男性用户与女性用户比例为4：1左右。

从图3可以看出，30-40岁用户租车数量更多，其次为20-30岁，租车用户年龄主要集中在20-60岁之间，可推测租车主要目的是上下班通勤。

4.2 时间维度

按照日期、年、月、日、星期、小时聚合trip数据集，统计每小时订单数，形成一个时间维度，重设索引并对字段重命名。

```
groupby_year_month_day_hour = trip.groupby(by =
['date','year','month','day','hour','weekday']).trip_id.count()
gg1 = groupby_year_month_day_hour.reset_index()
gg1.columns = ['date','year','month','day','hour','weekday','count']
```

为gg1数据表增加一个季节（season）字段。

```
def get_season(x):
    if 1 <= x <= 3:
        return 'Spi'
    elif 4 <= x <= 6:
        return 'Sum'
    elif 7 <= x <= 9:
        return 'Fall'
    else:
        return 'Win'
gg1['season'] = gg1.month.apply(get_season)
```

完成绘图

```
fig = plt.subplots(figsize=(18,18))

ax1 = plt.subplot2grid((3,2),(0,0))
gg1['year_month'] = gg1.date.astype('datetime64[M]')
df4 = gg1.groupby('year_month').sum()['count']
df4.plot( linestyle = 'dashed', marker = 'o', ax = ax1 )
ax1.set_title('2014-2016西雅图Pronto共享单车每月使用量')

ax2 = plt.subplot2grid((3,2),(0,1))
df41 = gg1[['month','count']]
df41.boxplot( by='month', ax = ax2)
ax2.set_title('2014-2016西雅图Pronto共享单车按月使用量')
ax2.set_xticklabels(['Jan','Feb','Mar','Apr','May','June','July','Aug','Sept','Oct','No
v','Dec'], rotation='horizontal')

ax3 = plt.subplot2grid((3,2),(1,0))
df5 = gg1.groupby(['hour','weekday']).mean().unstack()['count']
df5.rename( columns = {0:'Mon',1:'Tue',2:'Wed',3:'Thu',4:'Fri',5:'Sat',6:'Sun'},
inplace = True )
df5.plot(linestyle = 'dashed', marker = 'o', ax =ax3 )
ax3.set_title('2014-2016西雅图Pronto共享单车按星期每小时平均使用量')

ax4 = plt.subplot2grid((3,2),(1,1))
```

```
df51 = gg1[['weekday','count']]
df51.boxplot( by='weekday', ax = ax4)
ax4.set_title('2014-2016西雅图Pronto共享单车按星期使用量')
ax4.set_xticklabels(['Mon','Tue','Wed','Thu','Fri','Sat','Sun'], rotation='horizontal')

ax5 = plt.subplot2grid((3,2),(2,0))
df6 = gg1.groupby(by = ['hour','season']).mean().unstack()['count']
df6.plot( linestyle = 'dashed', marker='o', ax = ax5 )
ax5.set_title('2014-2016西雅图Pronto共享单车按季度每小时平均使用量')

ax6 = plt.subplot2grid((3,2),(2,1))
df61 = gg1[['hour','count']]
df61.boxplot( by='hour', ax = ax6)
ax6.set_title('2014-2016西雅图Pronto共享单车每小时使用量')
```

图4

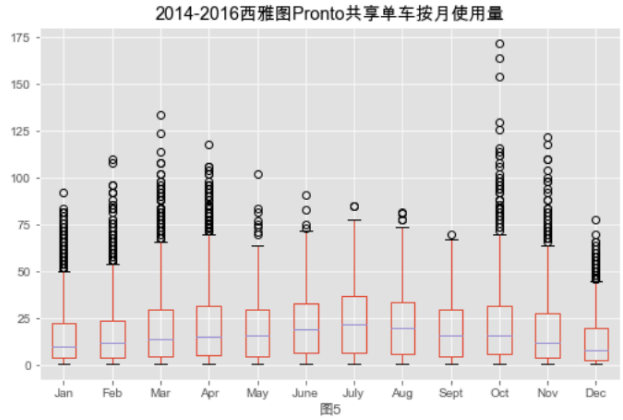


图5

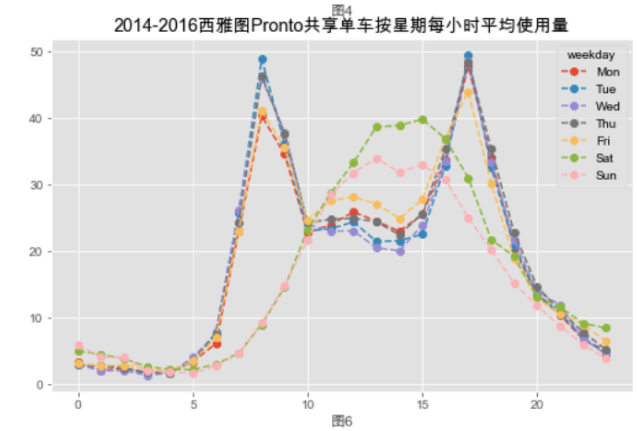


图6

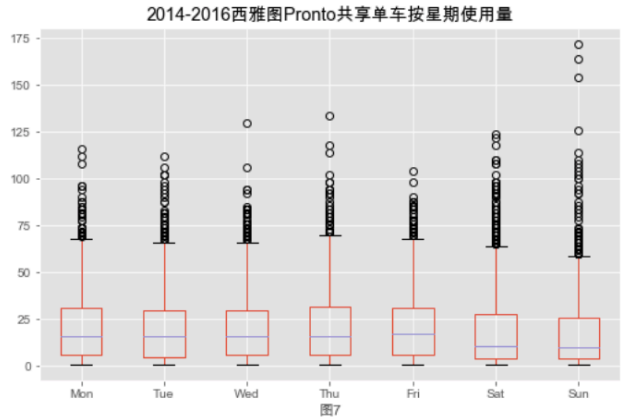


图7

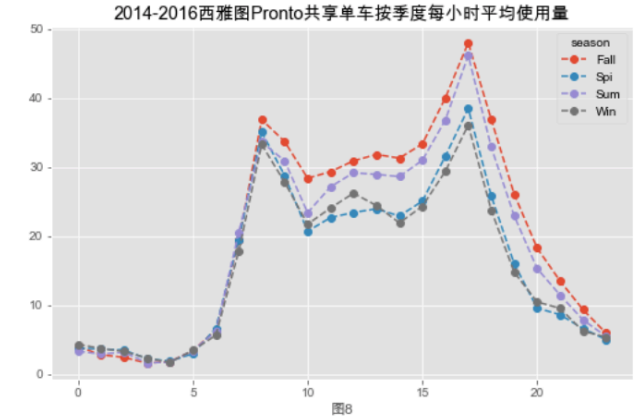


图8

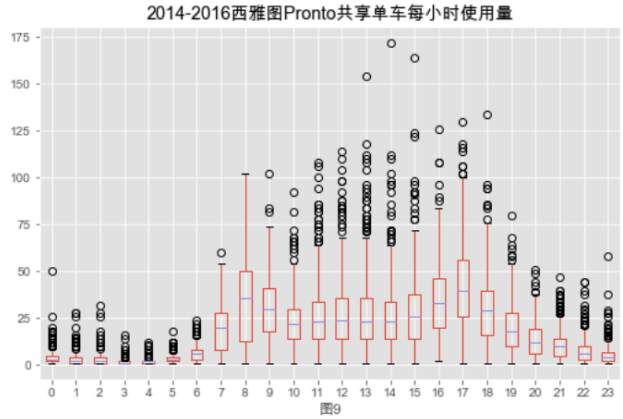


图9

结果分析
从图4、图5可以看出，租车数量随月份而变化，时间是影响租车数量的重要因素。2016年比2015年同期租车数量有所下降，可能是竞争对手、补贴力度、宣传渠道、用户忠诚度等原因造成的。由于样本数据太少，不能进行分析。

从图6、图7可以看出，工作日租车情况大致相同，周末有所下降；工作日与周末租车需求大致相反，结合图7看出，租车主要目的是上下班通勤，次要目的是出行租车。

从图8、图9可以看出，季节对租车人数有一定影响，春季、冬季租车数量有所下降，一天内租车人数随时间发生变化，且趋势与季节时间是相同的。图中早8点 和晚17点为租车高峰期，显然是上下班通勤租车；中午12点有一个租车小高峰，推测是外出就餐造成的。

从以上分析可知，时间是影响租车数量的重要因素，而天气与时间密切相关，天气包括温度、湿度、降雨量、能见度、风速等因素。

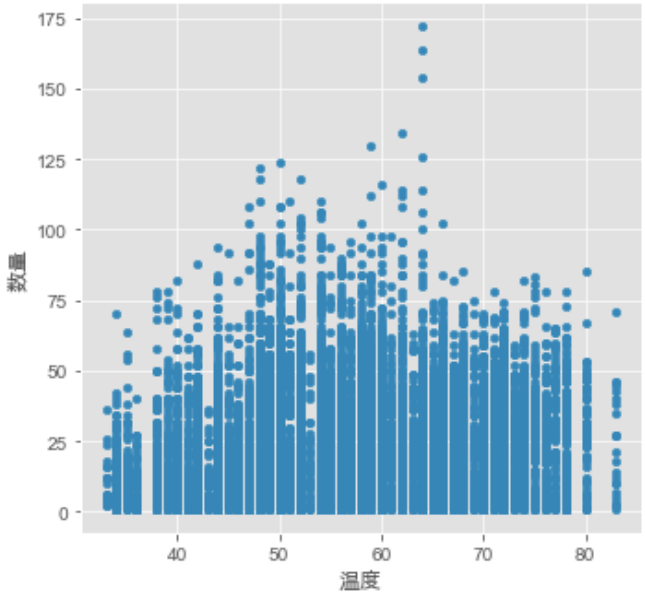
4.3 天气维度

把weather数据表和中间表gg1分类数据集进行关联，分析天气因素对租车量的影响。

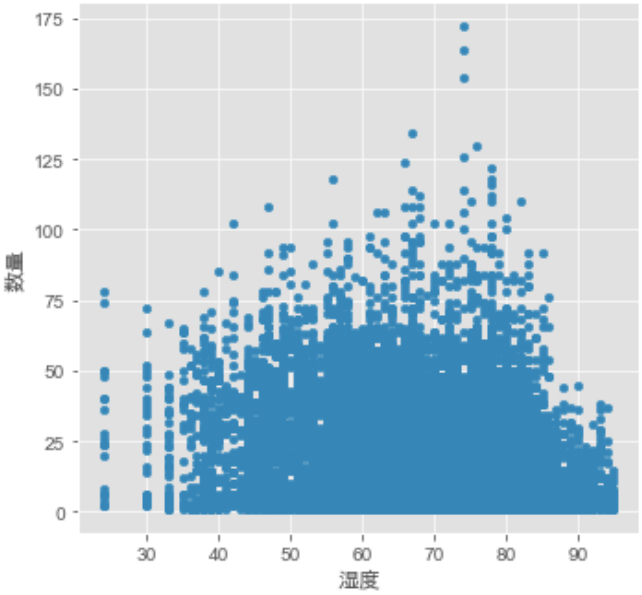
```
merged = gg1.merge(right=weather,how='inner',left_on='date',right_on='Date')

fig = plt.subplots(figsize=(12,18))

ax1 = plt.subplot2grid((3,2),(0,0))
merged.plot.scatter(x='Temperature',y='count',ax = ax1 )
ax1.set_title('2014-2016西雅图Pronto共享单车不同温度使用量')
ax1.set_xlabel('温度')
ax1.set_ylabel('数量')

ax2 = plt.subplot2grid((3,2),(0,1))
merged.plot.scatter(x='humidity',y='count',ax = ax2 )
ax2.set_title('2014-2016西雅图Pronto共享单车不同湿度使用量')
ax2.set_xlabel('湿度')
ax2.set_ylabel('数量')

ax3 = plt.subplot2grid((3,2),(1,0))
merged.plot.scatter(x='Sea_Pressure',y='count',ax = ax3 )
ax3.set_title('2014-2016西雅图Pronto共享单车不同海平面压力使用量')
ax3.set_xlabel('海平面压力')
ax3.set_ylabel('数量')

ax4 = plt.subplot2grid((3,2),(1,1))
merged.plot.scatter(x='Visibility_Miles',y='count',ax = ax4 )
ax4.set_title('2014-2016西雅图Pronto共享单车不同能见度使用量')
ax4.set_xlabel('能见度')
ax4.set_ylabel('数量')

ax5 = plt.subplot2grid((3,2),(2,0))
merged.query('Wind_Speed < 20').plot.scatter(x='Wind_Speed',y='count',ax = ax5 )
ax5.set_title('2014-2016西雅图Pronto共享单车不同风速使用量')
```

```
ax5.set_xlabel('风速')
ax5.set_ylabel('数量')

ax6 = plt.subplot2grid((3,2),(2,1))
merged.query("Precipitation_In < 1.25").plot.scatter(x='Precipitation_In',y='count',ax
= ax6 )
ax6.set_title('2014-2016西雅图Pronto共享单车不同降雨量使用量')
ax6.set_xlabel('降雨量')
ax6.set_ylabel('数量')
```

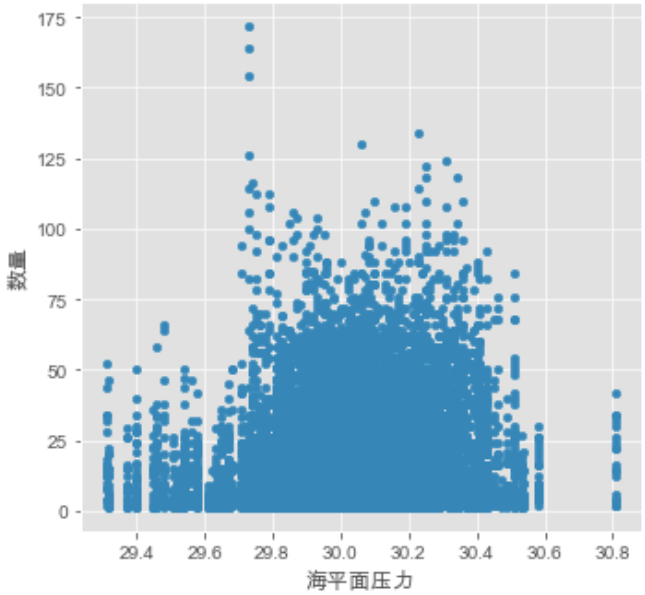
2014-2016西雅图Pronto共享单车不同温度使用量



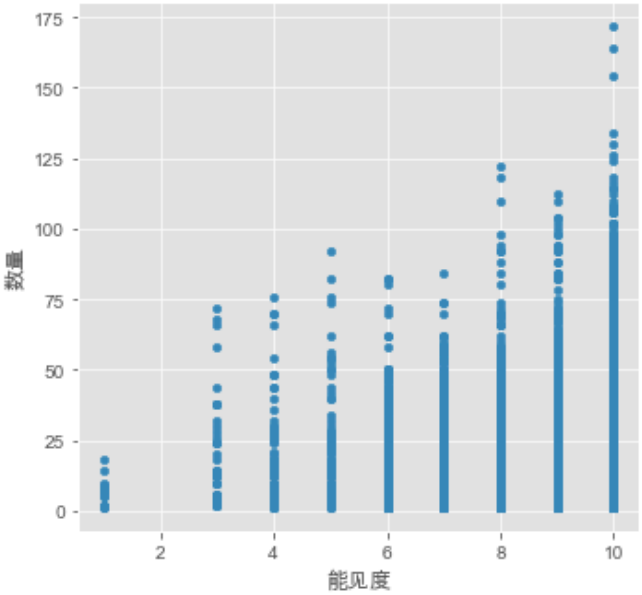
2014-2016西雅图Pronto共享单车不同湿度使用量



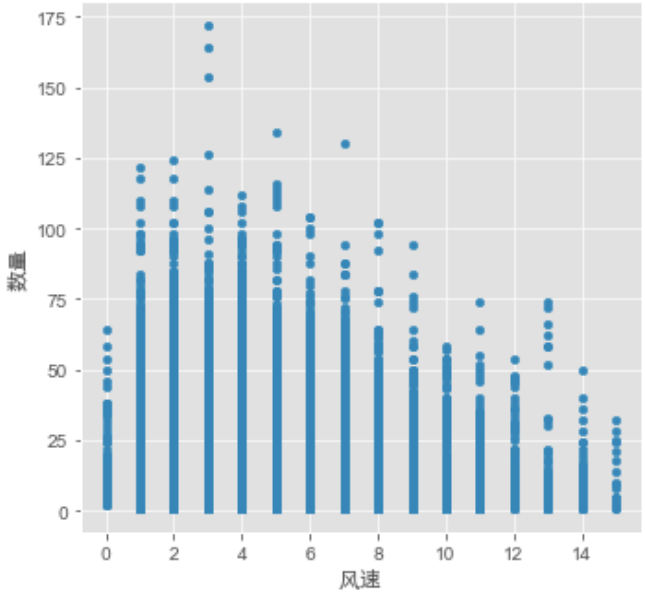
2014-2016西雅图Pronto共享单车不同海平面压力使用量



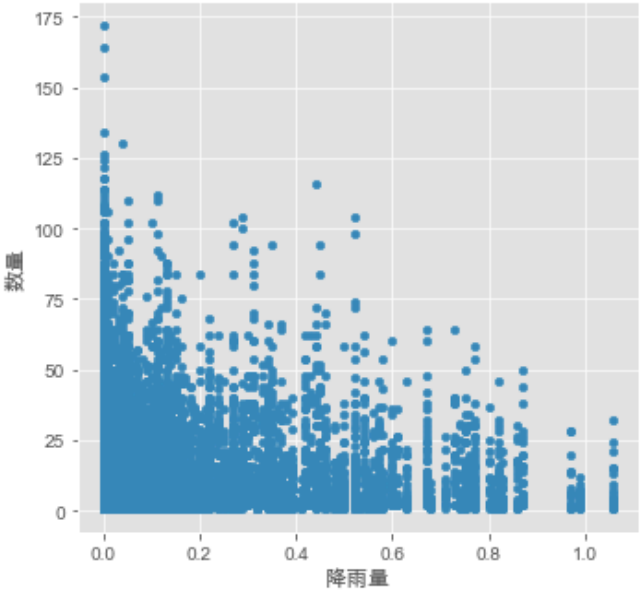
2014-2016西雅图Pronto共享单车不同能见度使用量



2014-2016西雅图Pronto共享单车不同风速使用量



2014-2016西雅图Pronto共享单车不同降雨量使用量



结果分析

温度在40-80华氏度范围内租车需求较高，温度过高过低均抑制租车需求。

湿度在35-85范围内租车需求较高，湿度过高过低同样抑制租车需求。

海平面压力在29.7-30.4范围内租车需求较高，海平面压力过高过低也抑制租车需求。

能见度越高租车需求越高，风速越大租车需求越低，降雨量越大租车需求越低。