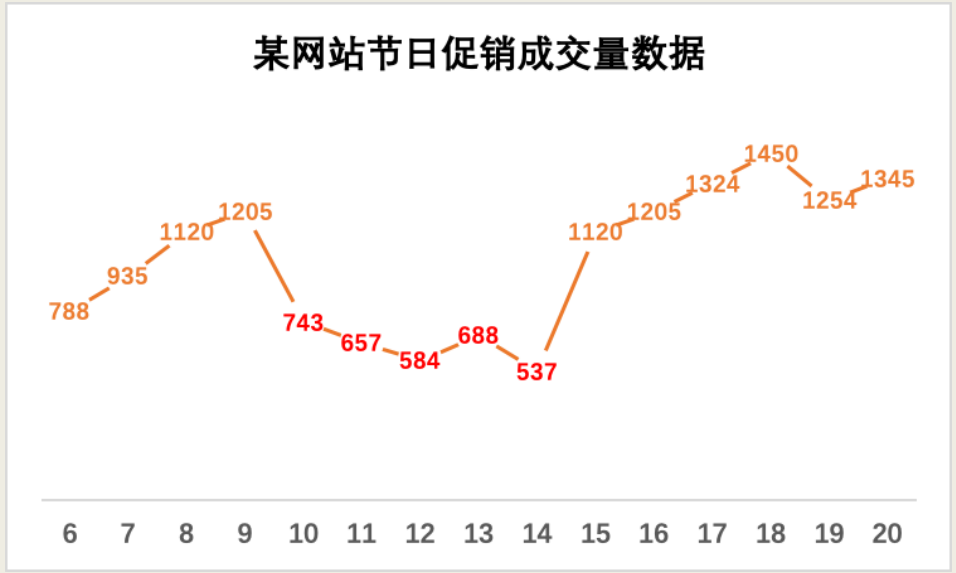
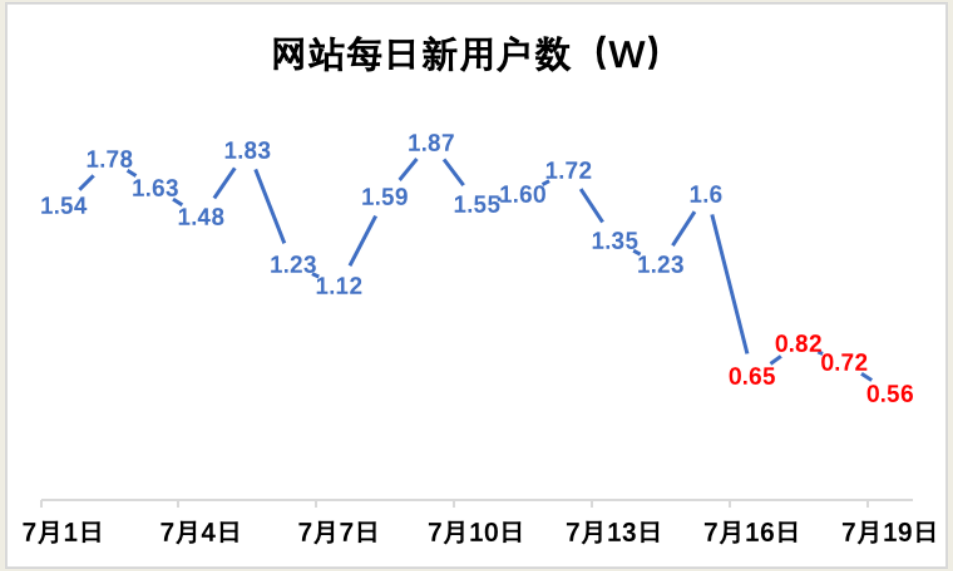
PYTHON指标异常预警

常见的指标异常案例

网站节日大促、服务器宕机用户付款失败
14点用户进线，发现异常，修复后成交量恢复正常

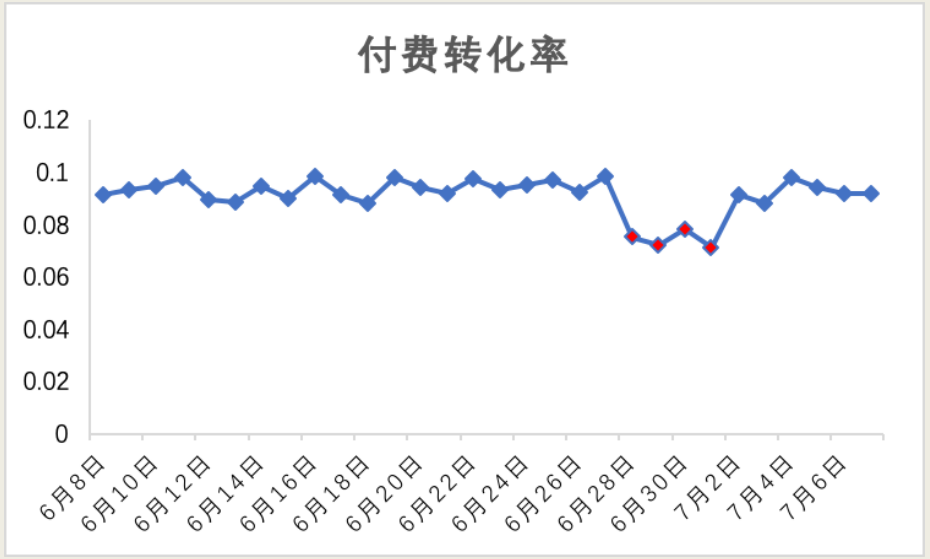


APP新用户数锐减，查询后发现APP新版本注册页面闪退，用户无法注册

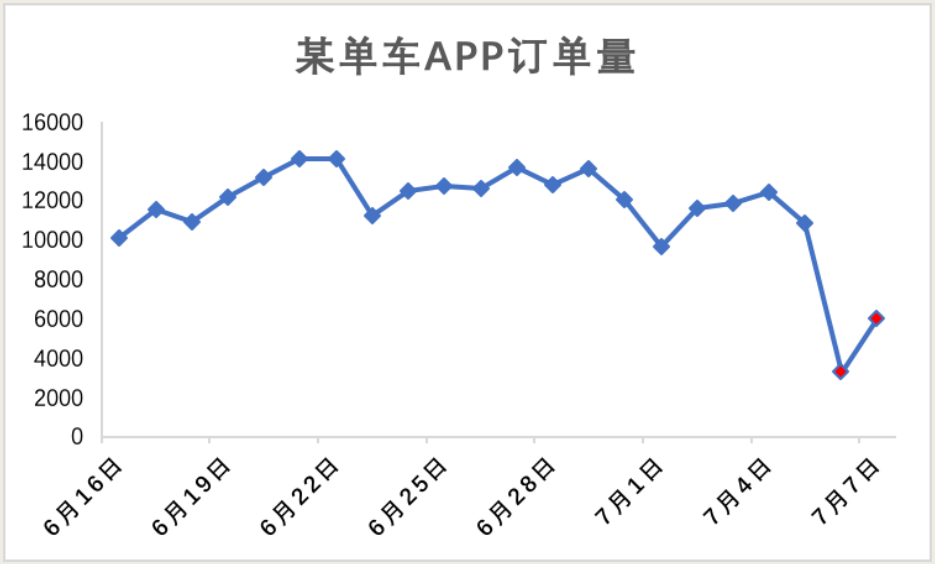


常见的指标异常案例

监控到游戏付费转化率下降，分析发现是因为更换活动素材引起的，把素材切回原来的版本，转化率恢复



7月6～7日，由于下暴雨，单车APP订单量暴跌



指标异常预警的意义

- 提高发现问题的效率，及时跟进

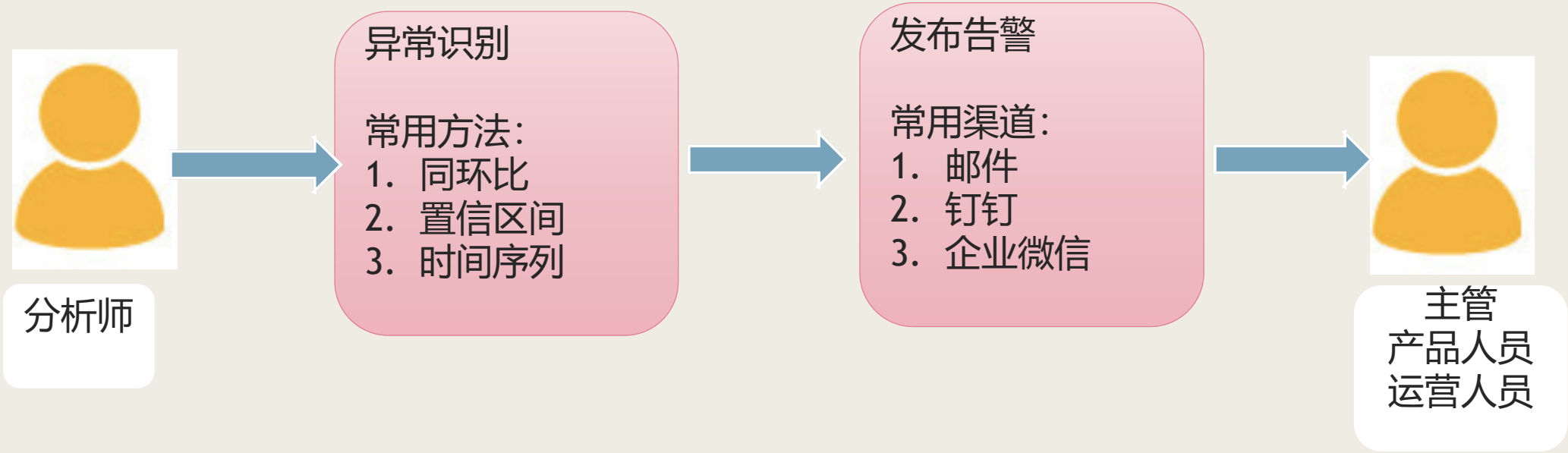
- 指标预警的本质：**把不符合预期的值挑出来**
  1. *预测今天的数值*
  2. *对比现实和预测值*

- 哪些场景、指标适合做监控
  1. *KPI指标（活跃用户数、总营收、订单比数）*
  2. *各环节基础数据（APP打开次数、添加购物车次数）*
  3. *转化率（注册成功率=注册成功数l 注册页面uv）*

为什么用Python

- 代码可复用性好
- 学习成本较低

完整的指标预警 = 异常识别 + 告警



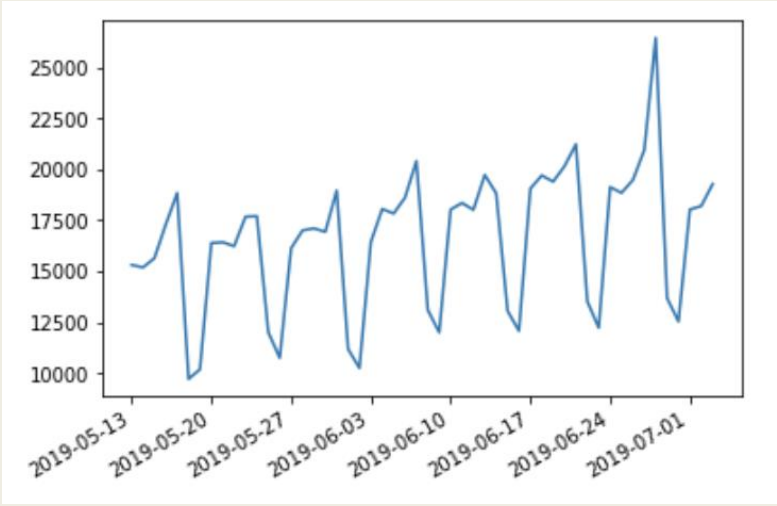
用Python实现同比异常判断

同比异常预警流程：

1. 根据历史数据确定同比范围
    1. 求出历史同比数值
    2. 结合业务判断，选取正常波动范围
2. 同比超出波动范围，预警

某网站每日完单数据异常监控

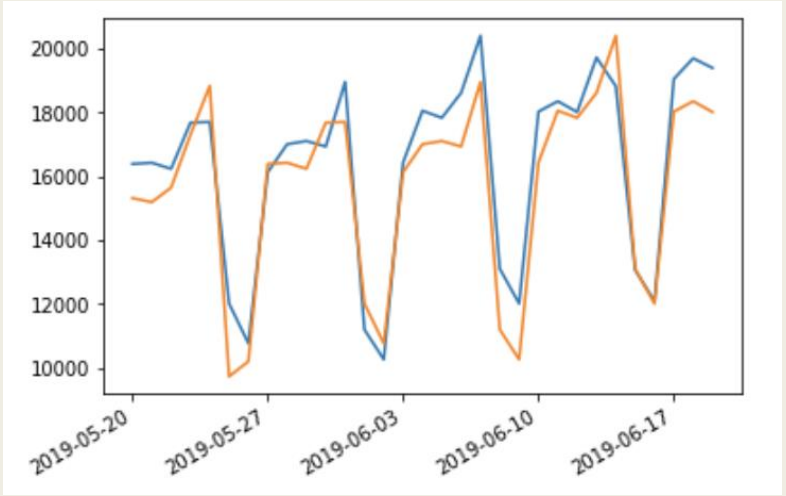
	dt	finish_order
0	2019-05-13	15315.7
1	2019-05-14	15190.4
2	2019-05-15	15643.4
3	2019-05-16	17297.0
4	2019-05-17	18837.5
5	2019-05-18	9723.8
6	2019-05-19	10199.4
7	2019-05-20	16387.7
8	2019-05-21	16425.1
9	2019-05-22	16234.2

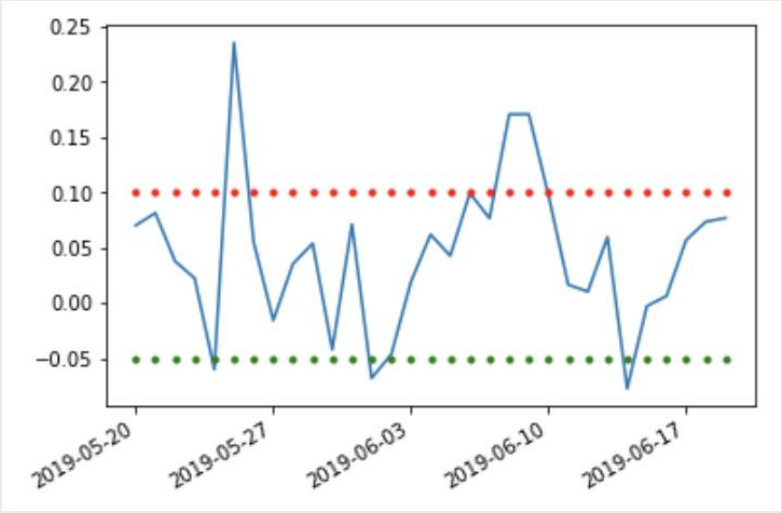


用Python实现同比异常判断：确定同比波动范围

计算同比数值、绘制图形观察

	dt	finish_order	lastweek_day	lastweek_value	w2w
7	2019-05-20	16387.7	2019-05-13	15315.7	0.069994
8	2019-05-21	16425.1	2019-05-14	15190.4	0.081282
9	2019-05-22	16234.2	2019-05-15	15643.4	0.037767
10	2019-05-23	17680.1	2019-05-16	17297.0	0.022148
11	2019-05-24	17703.3	2019-05-17	18837.5	-0.060210
12	2019-05-25	12012.7	2019-05-18	9723.8	0.235392
13	2019-05-26	10761.6	2019-05-19	10199.4	0.055121
14	2019-05-27	16127.8	2019-05-20	16387.7	-0.015859
15	2019-05-28	17004.3	2019-05-21	16425.1	0.035263
16	2019-05-29	17106.0	2019-05-22	16234.2	0.053701
17	2019-05-30	16932.8	2019-05-23	17680.1	-0.042268






```
df2.shape
```

```
(31, 5)
```

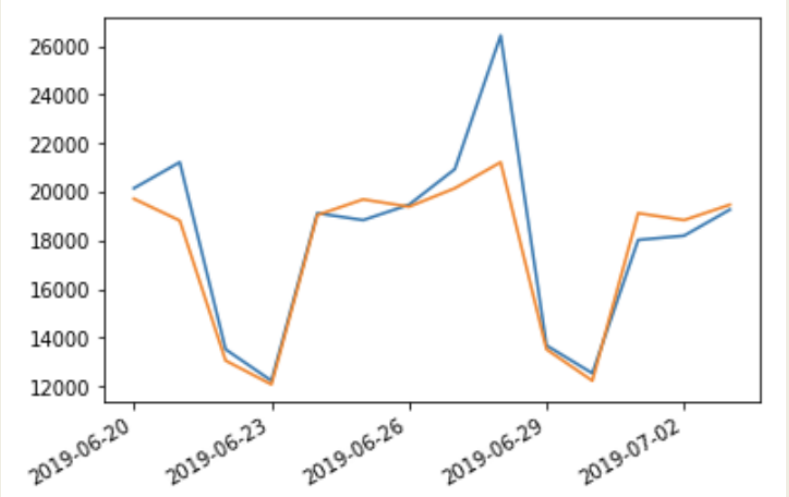
```
df2.loc[(df2.w2w>0.1)|(df2.w2w<-0.05)].shape
```

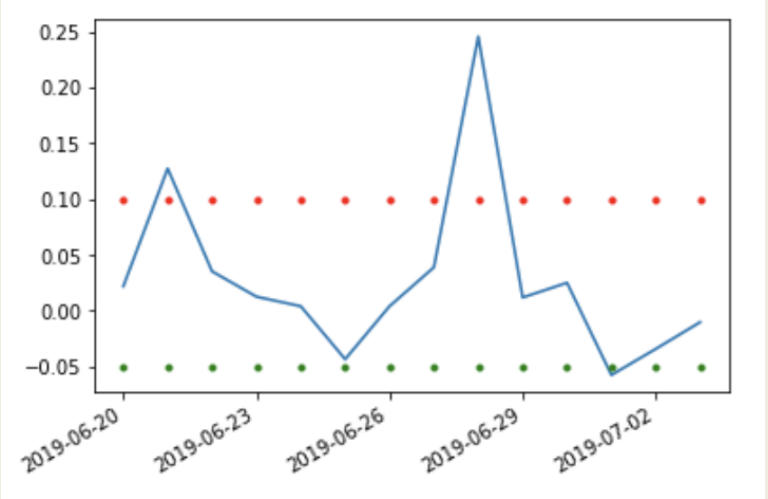
```
(6, 8)
```

用Python实现同比异常判断：判断是否超出同比波动范围

计算同比数值、判断是否超出上下波动范围

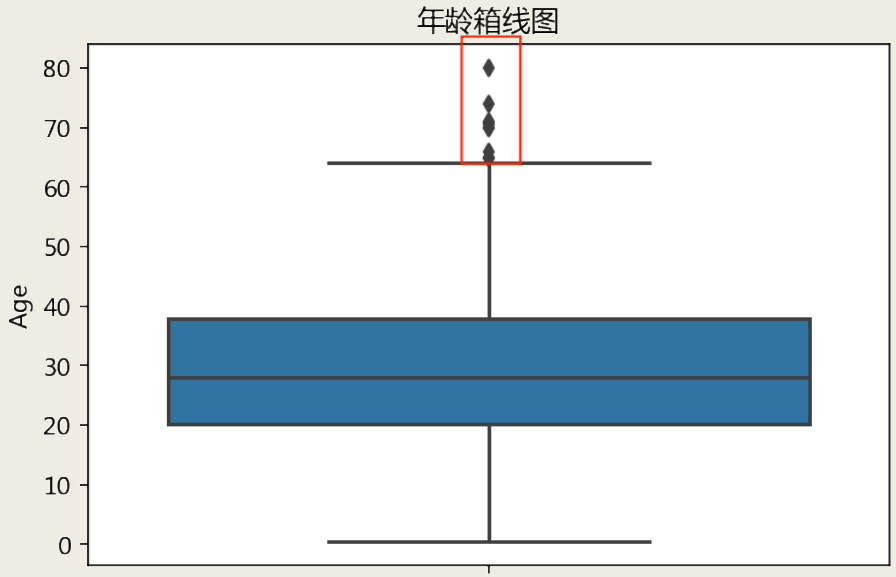
	dt	finish_order	lastweek_day	lastweek_value	w2w	upper_range	lower_range	is_abnormal
38	2019-06-20	20156.1	2019-06-13	19722.1	0.022006	0.1	-0.05	0
39	2019-06-21	21226.2	2019-06-14	18825.6	0.127518	0.1	-0.05	1
40	2019-06-22	13526.1	2019-06-15	13063.6	0.035404	0.1	-0.05	0
41	2019-06-23	12236.4	2019-06-16	12083.1	0.012687	0.1	-0.05	0
42	2019-06-24	19129.9	2019-06-17	19051.3	0.004126	0.1	-0.05	0
43	2019-06-25	18844.7	2019-06-18	19698.5	-0.043343	0.1	-0.05	0
44	2019-06-26	19476.0	2019-06-19	19395.9	0.004130	0.1	-0.05	0
45	2019-06-27	20942.8	2019-06-20	20156.1	0.039030	0.1	-0.05	0
46	2019-06-28	26443.9	2019-06-21	21226.2	0.245814	0.1	-0.05	1
47	2019-06-29	13687.8	2019-06-22	13526.1	0.011955	0.1	-0.05	0
48	2019-06-30	12543.9	2019-06-23	12236.4	0.025130	0.1	-0.05	0
49	2019-07-01	18030.2	2019-06-24	19129.9	-0.057486	0.1	-0.05	1
50	2019-07-02	18201.9	2019-06-25	18844.7	-0.034110	0.1	-0.05	0
51	2019-07-03	19278.2	2019-06-26	19476.0	-0.010156	0.1	-0.05	0





用数据本身的特征确定波动范围——箱线图

```
#箱线图


sns.boxplot(data=df, y='Age')
plt.title('年龄箱线图')
```



函数：sns.boxplot()

箱线图识别异常点的原理：

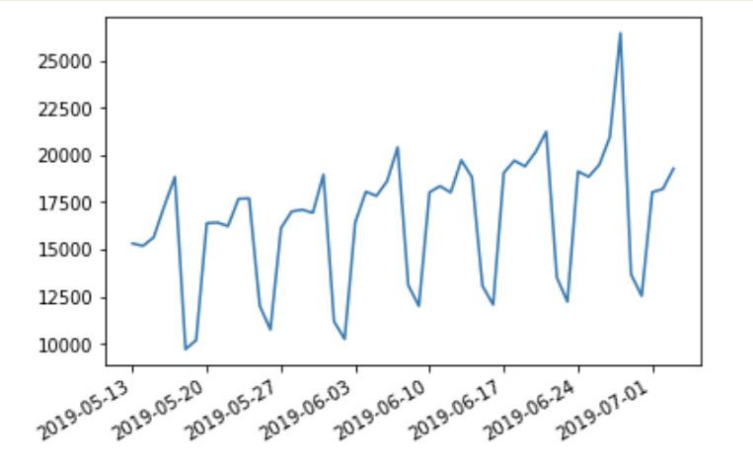
Q1:下四分位数(25%)
Q3:上四分位数(75%)
IQR:四分位距，IQR=Q3-Q1

上限=Q3+1.5*IQR
下限=Q1 -1.5*IQR

如果最大值没有超过Q3+1.5*IQR，则延长线上限为最大值，延长线下限同理

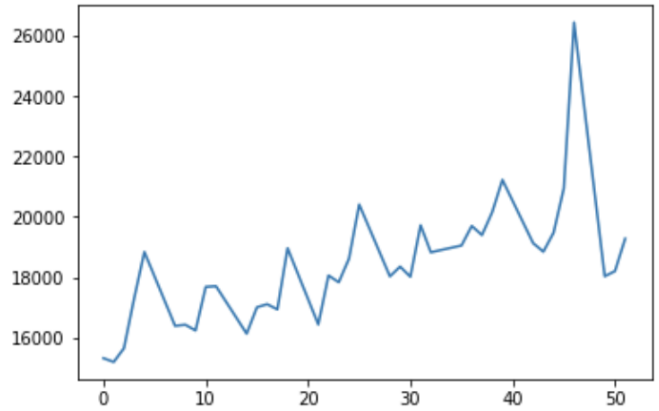
```
Q1 = np.percentile(df[col], 25) #取出指定分位数
Q3 = np.percentile(df[col], 75)
IQR = Q3-Q1
upper = Q3+1.5*IQR
lower = Q1-1.5*IQR
```

用数据本身的特征确定波动范围——箱线图



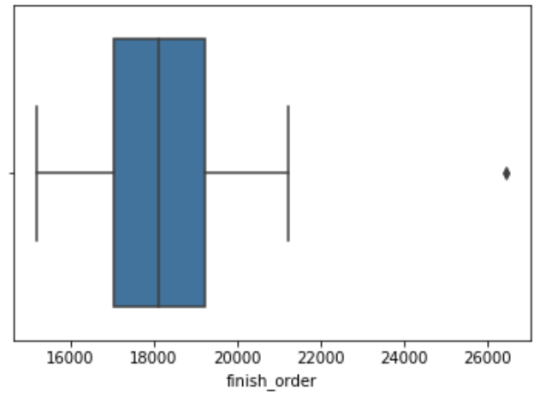

```
df.loc[(df.is_workday==1)].finish_order.plot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1a276965f8>
```




```
sns.boxplot(df.loc[(df.is_workday==1)].finish_order)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x10fc5ce80>
```

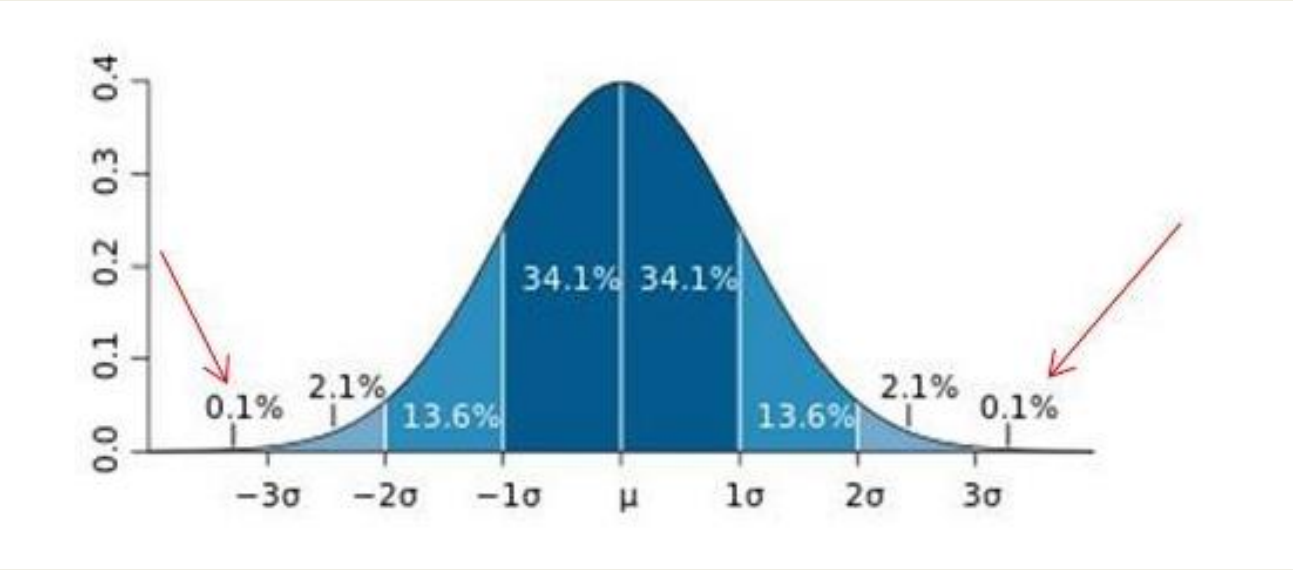


箱线图异常识别的特点：

只能识别波动极大的异常点

用数据本身的特征确定波动范围——3倍标准差

- 3倍标准差（数据需服从正态分布）

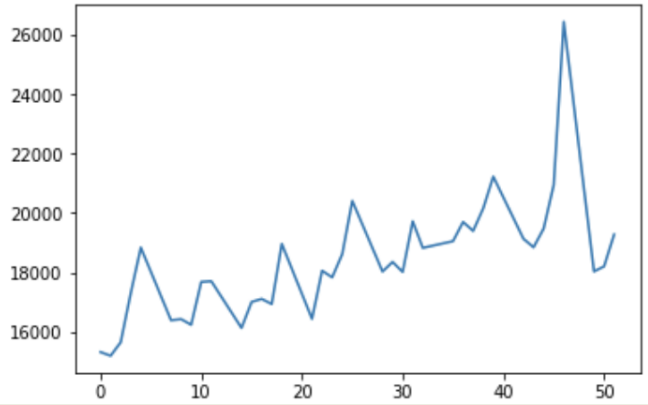


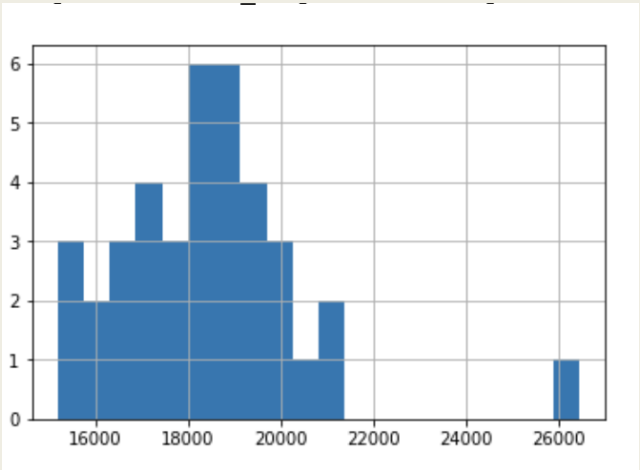

```
#3倍标准差

mean = df2.Fare.mean()
std = df2.Fare.std() #标准差

upper = mean+3*std #上限
lower = mean-3*std #下限
```

用数据本身的特征确定波动范围——3倍标准差

```
df.loc[(df.is_workday==1)].finish_order.plot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1a276965f8>
```






```
#计算均值与标准差
mean = df.loc[(df.is_workday==1)].finish_order.mean()
std = df.loc[(df.is_workday==1)].finish_order.std()

mean,std
```

```
(18342.326315789473, 2029.360405656304)
```

```
#仅能识别极端异常值

upper_3std = mean + 3*std
lower_3std = mean - 3*std

upper_3std, lower_3std
```

```
(24430.407532758385, 12254.24509882056)
```

```
upper_2std = mean + 2*std
lower_2std = mean - 2*std

upper_2std, lower_2std
```

```
(22401.04712710208, 14283.605504476865)
```

只能识别波动极大的异常点

参考资料

- 10分钟入门pandas，https://blog.csdn.net/wangshuang1631/article/details/52276189
- Python钉钉机器人报警，https://www.cnblogs.com/wangzhouyi/p/9724659.html
- Python邮件发送报表，https://blog.csdn.net/u012111465/article/details/82713561
- Python用ARIMA做时间序列预测，https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/

代码实践，请大家打开电脑