作者简介

HeoiJin：立志透过数据看清世界的产品策划，专注爬虫、数据分析、产品策划领域。

万物皆营销 | 资本永不眠 | 数据恒真理

一、项目准备

- 语言：Python 3.7
- IDE：Pycharm
- 相关库：pandas 0.25.3、matplotlib 3.2.1、pyecharts 1.6.2、seaborn 0.10.0
- 其他：chromedriver 83.0.4103.39、Edge 83.0.478.37
- 分析框架：5w2h，销售额=UV*转化率*客单价

PS：老规矩代码仅展示核心知识点部分，源码和数据集在文末

二、了解数据，梳理指标

数据字段梳理

数据集来自“和鲸”的天猫订单综合分析[1]，只有一个文件 report.csv，包含 7 个字段，共 28010 条数据，具体字段为：

- 订单编号
- 总金额：订单总金额，本文假设为商品的标价，共 866 种
- 买家实际支付金额：最终成交金额，分为已付款和未付款两种情况

- 已付款情况下：买家实际支付金额 = 总金额 - 退款金额
- 未付款情况下：买家实际支付金额 = 0
- 收货地址：买家的收货地址，记录维度为省市，共记录了 31 个省市
- 订单创建时间：2020 年 2 月 1 日 至 2020 年 2 月 29 日
- 订单付款时间：2020 年 2 月 1 日 至 2020 年 3 月 1 日
- 退款金额：付款后申请退款的金额，如果没有退款，退款金额为 0

指标维度梳理

在天猫母婴商品的分析当中，仅销售量作为结果指标，所有的分析围绕这个结果指标即可。但通过上面的字段梳理可知，除了成交金额作为结果指标外，还有一系列的过程指标，那么就需要对指标间的关系做逻辑梳理。

这里我们引入电商的分析中最经典的公式：**销售额 = UV * 转化率 * 客单价**

- 指标梳理：
- UV：在本数据集中，没有客户 id 作为 UV 数据，但我们可以把订单创建数量作为 UV 的数据
- 转化率：转化流程为订单创建 -> 订单付款 -> 订单成交 -> 订单全额成交
- 客单价：平均每单的售价，在本数据集当中，亦可以理解为各个产品的销量情况

- 维度梳理：
- 时间维度：（周/日）订单创建/付款时间
- 地域：各省市
- 产品：假设每一种金额对应唯一的产品时，总金额便可以作为产品品类的标识

数据清洗理

进行处理之前，先通过 info 函数对数据情况进行初步了解

```
In[2]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28010 entries, 0 to 28009
Data columns (total 7 columns):
订单编号          28010 non-null int64
总金额           28010 non-null float64
买家实际支付金额     28010 non-null float64
收货地址          28010 non-null object
订单创建时间        28010 non-null object
订单付款时间        24087 non-null object
退款金额          28010 non-null float64
dtypes: float64(3), int64(1), object(3)
memory usage: 1.5+ MB
```

观察可知，除订单付款时间之外，均没有缺失值。付款时间缺失的原因是用户在订单创建后跳失，缺失也是存在业务意义，暂不处理空值。

另外，订单创建时间和订单付款时间的格式是 object，需转化为时间格式，方便后续操作。Demo：

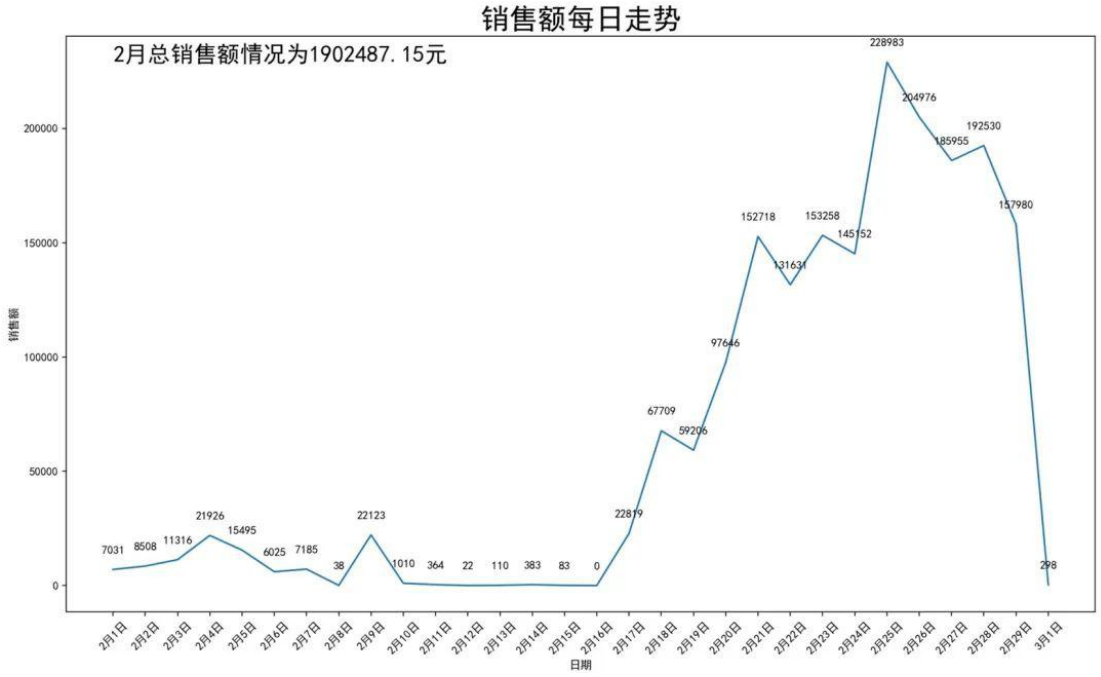
```
df['订单创建时间']=pd.to_datetime(df['订单创建时间'])
```

先看结果指标，确定现状

对于任何商业组织及其行为，最终目的都是获利，而在不考虑成本的情况下，收入便是最重要的结果指标。在本数据集中，第一步自然是要知道销售额的情况。

实现方法：

1. 以订单付款时间为分组条件，对买家实际支付金额求和
2. 用 matplotlib 绘制折线图



从上图可得知几个重点信息：

1. 整体的销售额为 190 万
2. 4 号出现局部峰值，5-8 号持续型下降，每日成交额低于万级水平
3. 10-16 日共一周时间的销售额几乎低于千级，需要特别留意数据的真实性
4. 17 日后出现持续型增长，25 日出现本月峰值

但仅凭上面的信息并不足以支撑决策，因此我们对案例增加些背景假设：

1. 本月的销售额目标是 220 万
2. 除 10-16 日之外，所有数据的采集均没有错误
3. 10-16 日的实际日均销售额为 2 万

在增加假设后，我们可以得知本月的真实销售额为 200 万，距离目标还差 20 万。接下来将从公式中的三个指标来拆解是什么环节出现问题，应该如何提升销售额。

三、拆解结果指标，进一步锁定问题

用户行为路径整体转化率

从字段梳理中可以得知用户行为路径为：订单创建 -> 订单付款 -> 订单成交 -> 订单全额成交。而转化率的计算方法有两种：

- 绝对转化率：每一个环节的订单数除以初始环节的订单数
- 相对转化率：每一个环节的订单数除以上一个环节的订单数

两种计算方式有各自的适用场景，个人理解在了解整体情况时绝对转化率更适合，而加入维度进行对比时，相对转化率则更适合。因此本环节中，使用绝对转化率进行计算。

实现方式：

正常的 groupby 函数并不能帮助我们对特定列进行复杂的筛选，因此需要手动计算各环节的订单数。

1. 求出各环节的订单数

- 订单创建/付款：计次
- 订单实际成交：筛选出买家实际支付金额大于 0 的所有行
- 订单全额成交：筛选出买家实际支付金额等于总金额的所有行

- 求转化率：用本环节的订单数除以订单创建数

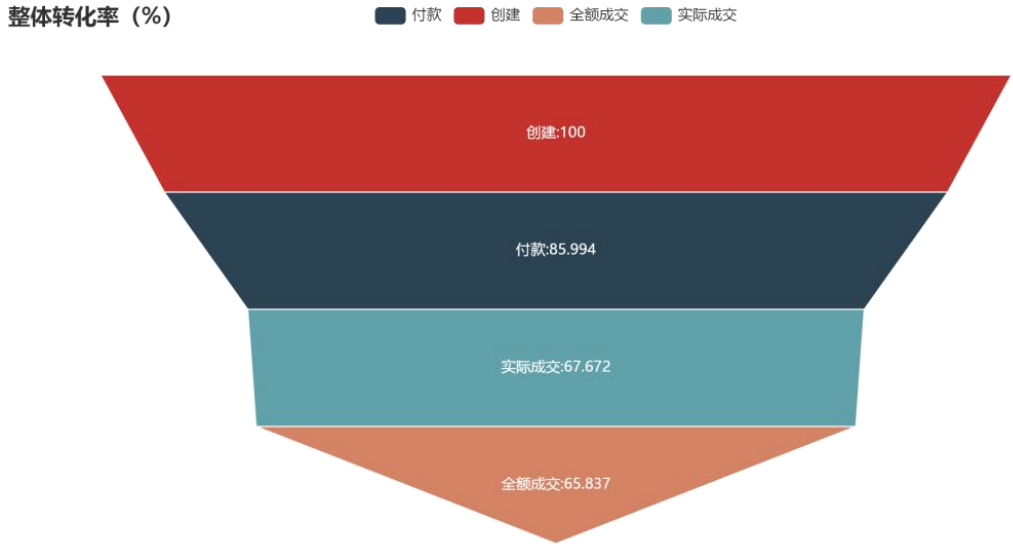
- 用 pyecharts 绘制漏斗图

```
rates = pd.Series({
    '创建':df['订单创建时间'].count(),
    '付款':df['订单付款时间'].count(),
    '实际成交':df[df['买家实际支付金额']>0].shape[0],
    '全额成交':df[df['买家实际支付金额']==df['总金额']].shape[0],
},name='订单量').to_frame()
# 绝对转化率=各环节订单数/订单创建数
rates['整体转化率']=rates['订单量'].apply(lambda x: round(x*100/rates.iloc[0,0],3))
# 可视化部分
c=(
    Funnel()
    .add(
        '转化率',
        [list(z) for z in zip(rates.index,rates['整体转化率'])],
        # 设置标签位置及数据展现形式
        label_opts=opts.LabelOpts(position='inside',formatter='{b}:{c}')
    )
```

```
    .set_global_opts(title_opts=opts.TitleOpts(title='整体转化率'))
)
# 转存
make_snapshot(snapshot,c.render(),'转化率.png')
```

```
-----各环节绝对转化率(%)-----

          订单量     整体转化率
创建      28010   100.000
付款      24087    85.994
实际成交   18955    67.672
全额成交   18441    65.837
```



转化率

根据 **TrustData** 的报告显示，淘宝平时的订单成功率（指提交订单支付的支付成功率）为 97.4%[2]。若以此作为标准，本次分析当中的付款转化率 85.99%低于预期标准，实际成交及全额成交的环节转化率甚至不到 7 成。属于比较低的水平。

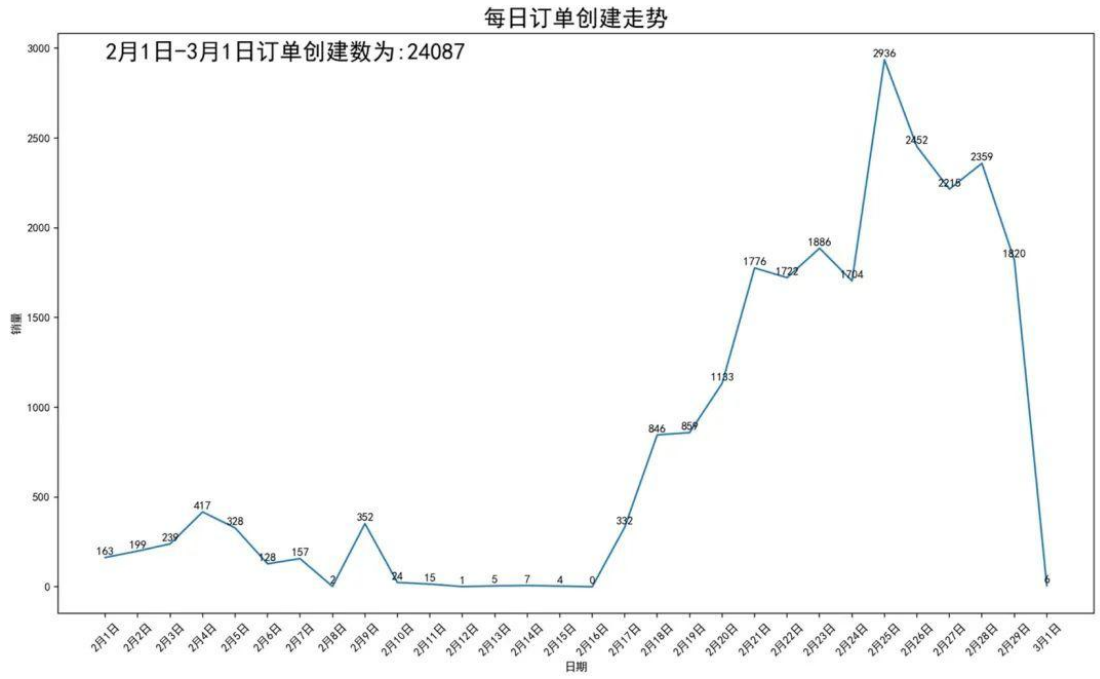
下一步的分析将锁定哪些区域转化率低，具有什么特征。

订单创建量每日走势

实现方法：

1. 以订单付款时间（频率：日）为分组条件，对订单创建时间计次

2. 用 matplotlib 绘制折线图



按照现有的转化率情况进行估算，10-16 日的日均订单创建量约为 300+，即 2 月的整体订单创建数为 2 万 6 千左右。在转化率及客单价均不变的情况下，订单创建数至少要增加多 6k 才能达到目标值。

值得注意 17 号后，订单创建量不断上升，峰值甚至接近上升前峰值的 7 倍。下一步的分析将通过不同区域的订单创建情况，找到哪些区域需要提升订单创建数。

实际交易量前 10 的产品情况

实现方法：

1. 筛选买家实际支付金额大于 0 的所有行

2. 对总金额所在列求次，并降序排序

3. 求出所有品类的占比情况

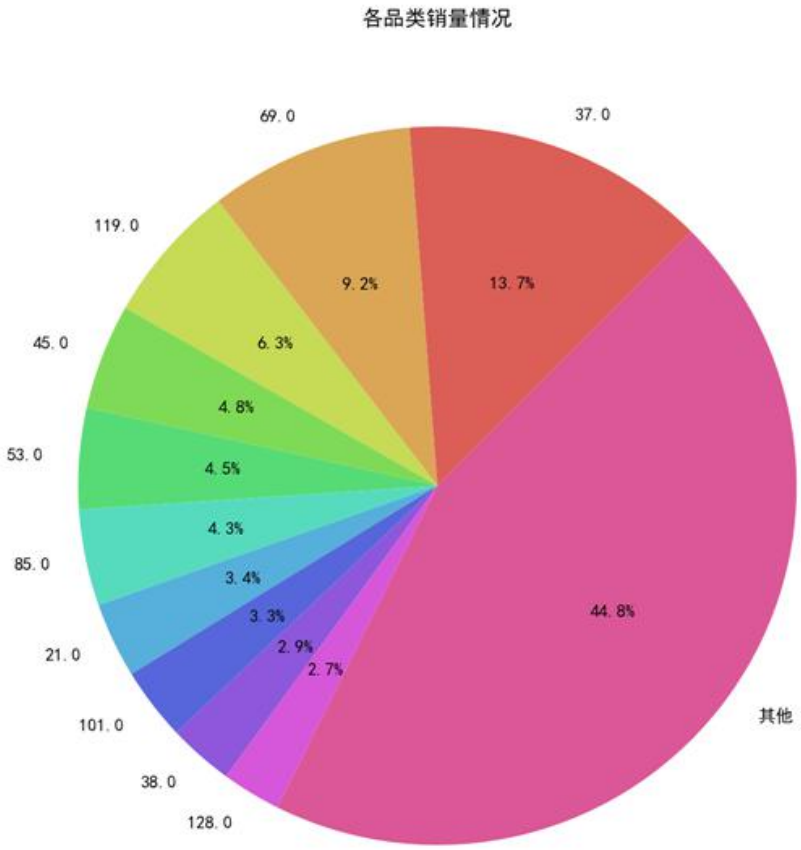
4. 将除销量前 10 之外的其他产品合并为其他品类

5. 利用 matplotlib 绘制饼图

```
df=df[df['买家实际支付金额']>0]
hot_=pd.DataFrame({
    '销量':df['总金额'].value_counts(),
    '全额付款销量':df[df['买家实际支付金额']==df['总金额']]['总金额'].value_counts()
}).sort_values(by='销量',ascending=False)
hot_['全额付款占比']=hot_['全额付款销量']/hot_['销量']
hot_['占比(%)']=hot_['销量'].apply(lambda x : round((x/hot_['销量'].sum())*100,2))
```

--------------------热卖前20--------------------

	销量	全额付款销量	总销量占比(%)	销售额	销售额占比(%)
37.0	2654	2628.0	14.00	97694.0	5.14
69.0	1776	1737.0	9.37	120851.0	6.35
119.0	1221	1213.0	6.44	145014.0	7.62
45.0	919	905.0	4.85	40966.0	2.15
53.0	875	863.0	4.62	46099.0	2.42
85.0	837	819.0	4.42	70148.0	3.69
21.0	654	650.0	3.45	13680.0	0.72
101.0	633	615.0	3.34	62763.0	3.30
38.0	566	561.0	2.99	21468.7	1.13
128.0	514	493.0	2.71	64644.0	3.40
76.0	425	416.0	2.24	31981.0	1.68
160.0	332	320.0	1.75	52471.0	2.76
43.0	318	315.0	1.68	13650.0	0.72
168.0	291	288.0	1.54	48723.0	2.56
112.0	274	265.0	1.45	30332.0	1.59
34.9	270	268.0	1.42	9378.1	0.49
104.0	264	254.0	1.39	27109.0	1.42
77.0	255	247.0	1.35	19238.0	1.01
81.0	209	206.0	1.10	16701.0	0.88
64.8	207	199.0	1.09	13054.8	0.69



销量前 10 的产品销量已经占到了总销量的 55%。在分析这类榜单类或者占比类的图表时，以下信息非常关键：

- 整体排行：是否有哪些产品排在前列/没有排在前列是出乎意料的事
- 趋势：销量排行变化情况，什么产品增长最快，什么产品表现平庸，什么产品开始下滑
- 产品布局：
- 走量类的产品销量是否多于品牌类产品
- 同一产品线的高中低端产品的销量分布情况，探究客户的消费需求偏向

这部分的分析考验数分们对业务中产品布局、产品策略规划等业务知识的熟悉程度。目前我们并没有更多数据和标准支撑分析，不再展开。

初次汇报

与业务部门的首次汇报沟通，我们并不需要明确所有细分维度的问题，重点在于拉齐认知，让业务部门对现状有客观的理解，并商讨出下一步的分析方向即可。我们通过 5w2h 框架对现有信息进行整理：

- 现状：2 月整体的销售额为 200 万，距离目标 220 万还差 20 万
- 原因：转化率及 UV 不达标，低价商品占比高：

1. 转化率情况：付款成功率为 85.99%，低于行业水平的 97.4%，实际成交和全额成交的转化率分别为 67.67%和 65.83%。

2. UV 情况/订单创建情况：2 月订单创建数为 2 万 6 千个。在转化率及客单价不变的情况下，需要增加至 3 万 2 千个才能达到目标值。

3. 客单价/品类销售情况：目前销售前 5 的产品分别为 37（2654 件）、69（1776 件）、119（1221 件）、45（919 件）、53（875 件），销量前 10 的产品已占总销量的 55%。

在拉齐认知之后，便可以进一步规划深入分析的方向。我们可以先与业务团队沟通，将他们认为的造成问题的原因以 MECE 原则整理为决策树，再逐点假设检验。而本文的着重，是通过多维度多指标的交叉分析，找到更具体的问题。因此，根据上述问题及现有数据，我们下一步分析方向为：

1. 各个省市的成交额分布情况

2. 各个省市的订单创建量变化情况

3. 各个省市的成交转化率变化情况

4. 各个省市的各品类销量及转化率情况

四、多维度交叉分析，找到解题思路

省市间的销量会有明显的差异，且在实际的商业环境当中，我们的资源有限，并不能照顾到所有的区域，一定要有取舍。

因此我们先通过了解各个省市的实际成交额情况来判断哪些省市是重点区域，再针对重点区域对比分析三个重要指标的情况，从而定位到更具体的问题，找到针对性的解决思路。

各个省市销售额情况

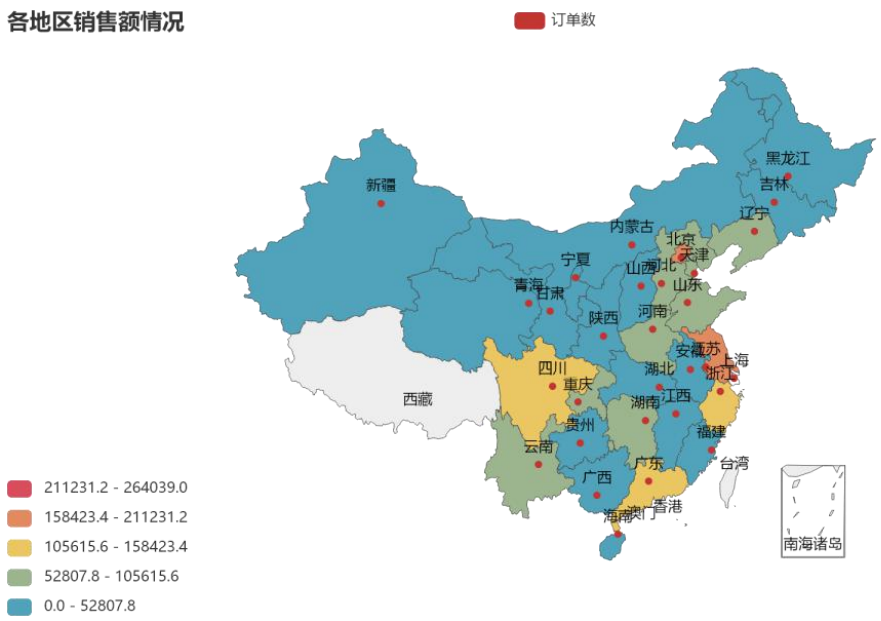
实现方法：

1. 挑选出实际支付金额大于 0 的所有行
2. 以收货地址为分组条件，对买家实际支付金额求和，并降序排序
3. 用 pyecharts 绘制地图分布

```
amount=df[df['买家实际支付金额'] > 0].groupby('收货地址')['买家实际支付金额'].sum().sort_values(ascending=False)

# 处理省份名称为 pyecharts 可识别的形式
_x=[i.replace('省','').replace('自治区','') for i in amount.index]
_x=[x if len(x)<4 else x[:2] for x in _x]

# 计算最大值作为 pyecharts 色块分组中的最大值
max_=int(amount['实际成交数'].max())
c=(
    Map()
    .add(
        '订单数',[list(i) for i in zip(_x, amount['实际成交数'].to_list())],'china'
    )
    .set_global_opts(
        title_opts=opts.TitleOpts(title='各地区实际成交订单数'),
        visualmap_opts=opts.VisualMapOpts(max_=max_,is_piecewise=True)
    )
)
```



从图及数据可以得知，销售额前 5 分别为上海、北京、江苏省、广东省、浙江省，下面的分析将针对这五个省市开展，其他区域的分析思路类似。

重点城市各项指标变化情况

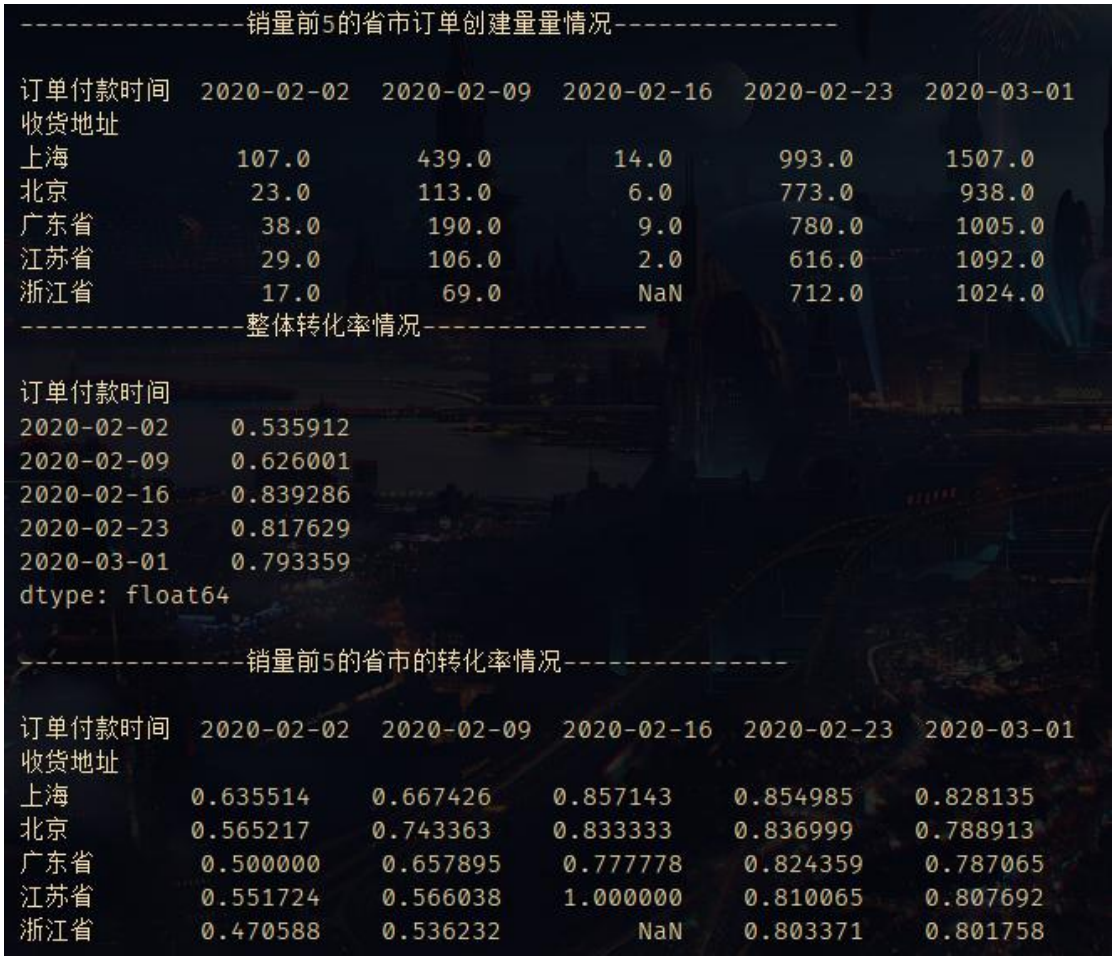
实现方法：

1. 新建列“成交情况”，记录所有实际成交（买家实际支付金额大于 0）的订单。如有实际成交则赋值为 1，否则为 0
2. 利用收货地址和订单付款时间（频率：周）分组，对订单创建时间计次并命名为订单创建数，对成交情况求和并命名为订单成交数
3. 筛选销量前五的省市：[上海，广东省，北京，江苏省，浙江省]
4. 求出实际成交的转化率

```
# 找到有实际成交的所有行
df['成交情况']=0
```

```
df.loc[df[df['买家实际支付金额']>0].index.to_list(),'成交情况']=1
df.loc[df[df['买家实际支付金额']==df['总金额']].index.to_list(),'全价成交']=1

# 分组求和
df_=df.groupby(by=['收货地址',pd.Grouper(key='订单付款时间',freq='W')]).agg(
    订单创建数=('订单创建时间', 'count'),
    订单成交数=('成交情况', 'sum')
)
df_head3=df_.loc[['上海','广东省','北京','江苏省','浙江省']]
# 求转化率
df_head3['实际成交转化率']=df_head3['订单成交数']/df_head3['订单创建数']
print(f'{"-" * 15}销量前 5 的省市订单创建量量情况{"-" * 15}\n')
print(df_head3['订单创建数'].unstack())
print(f'\n{"-" * 15}销量前 5 的省市的转化率情况{"-" * 15}\n')
print(df_head3['实际成交转化率'].unstack())
```



----------销量前5的省市订单创建量量情况----------

订单付款时间	2020-02-02	2020-02-09	2020-02-16	2020-02-23	2020-03-01
收货地址					
上海	107.0	439.0	14.0	993.0	1507.0
北京	23.0	113.0	6.0	773.0	938.0
广东省	38.0	190.0	9.0	780.0	1005.0
江苏省	29.0	106.0	2.0	616.0	1092.0
浙江省	17.0	69.0	NaN	712.0	1024.0

----------整体转化率情况----------

订单付款时间	
2020-02-02	0.535912
2020-02-09	0.626001
2020-02-16	0.839286
2020-02-23	0.817629
2020-03-01	0.793359

dtype: float64

----------销量前5的省市的转化率情况----------

订单付款时间	2020-02-02	2020-02-09	2020-02-16	2020-02-23	2020-03-01
收货地址					
上海	0.635514	0.667426	0.857143	0.854985	0.828135
北京	0.565217	0.743363	0.833333	0.836999	0.788913
广东省	0.500000	0.657895	0.777778	0.824359	0.787065
江苏省	0.551724	0.566038	1.000000	0.810065	0.807692
浙江省	0.470588	0.536232	NaN	0.803371	0.801758

从上图中我们可以得知几个信息：

1. 在春节过后，疫情最为严重的那段时间，每座城市的订单创建数及最终成交的转化率都是非常低，几乎是每创建两个订单，就有一个订单被退回。而在疫情逐渐明朗后，各项指标均有回升，但依然低于标准值 97%。

2. 上海的订单创建数及实际成交数都是最多的，转化率的表现也是最好的，因此作为标杆。

3. 北京及广东是重点区域。两个区域是除上海之外销量最多的，但转化率基本低于平均值，特别是广东，虽订单创建量排名第二，但销售额却排名第四。

4. 江苏和浙江为次重要区域。两个区域虽订单创建量相对较低，但转化率接近均值，销售额情况也较为乐观，江苏的销售额甚至高于广东。

下一步，我们结合产品维度，对比上海、北京、广东这三个区域的在 2 月 17-3 月 1 日期间不同品类销售情况及对应的转化率情况，从而找出具有针对性的优化建议。

结合产品维度交叉分析

实现思路：

1. 提取收货地址在北上、广东且时间在 2 月 17-3 月 1 日之间的所有数据
2. 新建列“成交情况”，用于记录所有实际成交（买家实际支付金额大于 0）的订单。如有实际成交则赋值 1，否则赋值 0
3. 以收货地址和订单付款时间（频率为周）为分组条件，对订单创建时间计次并命名为订单创建数，对成交情况求和并命名为订单成交数
4. 筛选每个省市的销量前 5 的产品
5. 求转化率

```
df=df[(df['收货地址'].isin(['北京','上海','广东省']))&(df['订单创建时间']>'2020-02-16')]
df['成交情况']=0
df.loc[df[df['买家实际支付金额']>0].index.to_list(),'成交情况']=1
df_=df.groupby(by=['收货地址','总金额']).agg(
    订单创建数=('订单创建时间', 'count'),
```

```
    订单成交数=('成交情况', 'sum')
)
# 筛选每个省市的销量前 5
df1=df_.reset_index().groupby('收货地址').apply(lambda x: x.nlargest(5,'订单创建数',keep='all')).set_index(['收货地址','总金额'])
a=df1['订单成交数'].unstack()
a.loc['上海',[21,53]]=df_.loc['上海'].loc[[21,53]]['订单成交数']

b=df1['订单创建数'].unstack()
b.loc['上海',[21,53]]=df_.loc['上海'].loc[[21,53]]['订单创建数']
```

```
----------------重点省市销量前5产品的订单创建量量情况----------------

总金额    21.0   37.0   45.0   53.0   69.0   85.0   119.0
收货地址
上海      79.0  352.0  129.0  112.0  257.0  134.0  212.0
北京       NaN  239.0    NaN   78.0  163.0  116.0  193.0
广东省    138.0  403.0    NaN  102.0  211.0    NaN  132.0

----------------重点省市销量前5产品的转化率情况----------------

总金额       21.0       37.0       45.0       53.0       69.0       85.0       119.0
收货地址
上海     0.594937   0.721591   0.775194   0.794643   0.793774   0.798507   0.745283
北京          NaN   0.669456        NaN   0.782051   0.742331   0.801724   0.740933
广东省    0.514493   0.595533        NaN   0.754902   0.682464        NaN   0.628788
```

对照我们的标杆上海区域，为北京、广东两个区域的不同产品提出不同的优化目标，例如：

北京：

- 针对 37 产品：在价格不变的情况下，提升 100 个订单创建数，成交转化率提升至 70%

广东：

- 针对 69 产品：在价格及订单创建数量不变的情况下，提升成交转化率至 75%

至于如何提升，我们需要对比三个区域在业务活动上的差异，比如渠道差异、推广活动差异、用户行为引导差异等等。在得知了业务活动上的差异后，我们就可以做针对性的 ABtest 或者其他的调整，最终从 UV、转化率、客单价三个方面入手提升收入。

五、复盘&反思

文章思路复盘

1. 我们从销售额情况入手了解 2 月的整体经营情况，发现销售额离目标约有 20 万元的差距
2. 通过一个公式三个指标拆解销售额，梳理 UV、转化率、客单价各项指标的情况，定位进一步分析的方向
3. 对不同区域的销售额情况进行简单了解，筛选核心的区域，加入产品维度进行多维度多指标交叉分析，制定有针对性的优化目标

当然，文章作为案例还是有很多值得斟酌的地方，特别是在针对产品销量提升的建议上，并没有考虑到产品的布局情况，这可能会出现同产品线中，低利润商品销量上升但高利润商品销量下降，从而导致最终销售额下降的怪圈。

反思

标准！标准！标准！

数据本身没有任何意义，**数据+标准才是价值的所在。**这篇文章鸽了这么久，就是因为一直没有找到合适的标准去衡量分析结果的好坏。从业务的角度看，没有了标准，就变成了一篇为了分析而分析，毫无商业价值的报告。

对于没有条件的孩子，走起来比站在原地强。

意识到因为没有标准导致分析无法深入从而一直拖稿的问题后，我做了两件事：

- **看行业分析报告找标准。**尽管行业报告的真实性有待考究，仅凭一句样本没有代表性，就能推翻的行业报告的结论。以此作为判断标准实属下策，但下策总比束手无策强。
- **按照现有思路先把文章写出来。**受到自己上一篇文章的影响，想在动手前明确完整的分析思路，不走多余的路。但实际数据分析并非一蹴而就，而是一个不断优化迭代的过程，先完成再完美。

那么本次分享就到这里啦，希望各路大神不吝赐教。

完整代码及数据集

- github：https://github.com/heoijin/BAproject
- 奶牛快传（推荐）：https://alltodata.cowtransfer.com/s/87c0fd7e2a084d
- 百度网盘：https://pan.baidu.com/s/1sRhbgi-VpU16bllCVISV2A 提取码：q4xs

参考文章

[1]数据

集: *https://www.kesci.com/home/project/5eb60fd0366f4d002d7792d5/fork*

[2]

TrustData 信诺数据：2015 年双十一中国移动互联网电商行业发展分析报

告: *https://www.useit.com.cn/thread-10658-1-1.html*