京东消费者行为数据分析

一、问题界定

1. What：用户的购买趋势如何？

2. Why：影响用户购买行为的因素有哪些？

3. How：京东的业务部门如何采取措施才能提升订单量？

二、数据收集与评估

1. 数据维度整合

类别	字段	重命名
用户维度	customer_id	用户 ID
	age_range	年龄分段
	gender	性别
	customer_register_date	用户注册日期
	customer_level	会员级别
	city_level	会员城市级别
购买行为维度	action_date	用户行为日期
	action_id	用户行为 ID
	type	用户行为类别
产品/店铺维度	brand	产品品牌
	product_id	产品 ID
	category	产品类别
	product_market_date	产品上市日期
	vender_id	商家 ID
	fans_number	粉丝数
	vip_number	会员数
	shop_register_date	开店日期
	shop_category	店铺主营类别
	shop_score	店铺评分
	shop_id	店铺 ID

经过初步观察，表中一共有 183828 行数据，20 个字段，总体上可分为三大维度，即用户、购买行为以及产品/店铺维度。

2. 缺失值分析

字段	缺失值数目	所占比例
年龄分段	83	0.05%
会员城市级别	125	0.07%
开店日期	71354	38.8%

由于 tableau 中数据的可视化操作可自行排除掉缺失值，因此这部分分析只用于后面利用 python 进行数据模型的构建，需要对缺失值进行处理。

通过简单的数据探索不难发现，最小的数据颗粒度为用户 ID（customer_id 字段），因此后续统计各个购买行为的用户数可基于此变量进行。

三、数据的整理和清洗

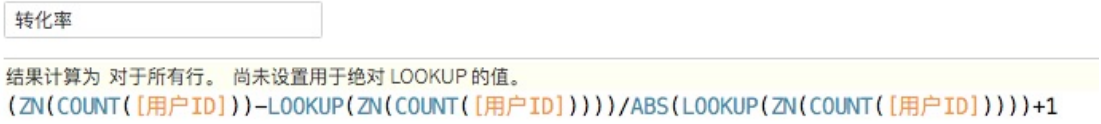
在使用 tableau 进行数据可视化前，我们需要将各个字段重命名（如表格所示），同时检查数据类型是否准确，如 ID 应为字符串，日期的话精确到天即可。此外，根据需要创建几个新的计算字段：

（1）用户注册时间（以年计）

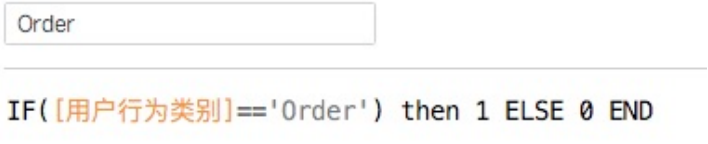
（2）产品上市时间（以年计）

（3）店铺年龄（以年计）

（4）转化率（%）：用于漏斗分析



（5）Order



其余对于缺失值的清洗，在数据模型部分详细阐述。

四、数据探索与可视化

此部分主要利用 tableau 进行探索和可视化，用以回答 what 和 why 的问题。

1. 用户购买趋势

1.1 变量选择

用户购买行为：用户行为类型 type 里的 Order 或直接用 Order 字段
时间维度：用户行为日期 action_date

1.2 呈现关系

呈现随时间的趋势变化

1.3 图表选择

使用折线图和漏斗分析图

1.4 业务结论

（1）基于 2018 年 2 月、3 月和 4 月的数据可以看出，用户购买行为在 2 月上旬呈现明显的下降趋势，而后逐渐回升并趋于平稳；
（2）通过漏斗分析这三个月不同用户行为间的转化率（基于浏览页总量计算），2 月和 3 月用户浏览页面的操作很多，但是没有转化为加入购物车的行为，少数用户直接下单购买，而 2 月用户购买行为的转化率在三个月里最低；4 月用户浏览量虽总量不高（这是因为 4 月只有前一周的数据，但几乎达到 3 月浏览量的 50%），但购买、评论、转粉的行为转化率得到提高。

2. 影响用户购买行为的因素

2.1 用户维度

2.1.1 变量选择

（1）年龄：年龄分段 age_range 字段（1，2，3，4，5，6 六个等级）
（2）性别：性别 gender 字段（男、女、未知）
（3）会员等级：会员级别 customer_level 字段（1，2，3，4，5，6，7 七个等级）
（4）会员所在城市：会员城市级别 city_level 字段（1，2，3，4，5，6 六个等级）
（5）用户注册时长：创建数据桶，2 年为一组，共 8 组
（6）用户购买行为：用户行为类型 type 里的 Order 或直接用 Order 字段

2.1.2 呈现关系

不同类别的比较关系

2.1.3 图表选择

（1）年龄分段与购买行为：条形图
（2）性别与购买行为：饼图
（3）会员级别与购买行为：条形图
（4）会员城市级别与购买行为：条形图
（5）用户注册时长与购买行为：条形图

2.1.4 业务结论

（1）年龄在第 5、6 分段的用户购买力较强于其他组，而第 3 年龄段用户几乎没有下单购买，只停留在页面浏览阶段，说明中老年人是购买主力军，而对中青年吸引力不够；
（2）男性用户购买能力略高于女性用户，男性用户在食物、美妆、衣服等店铺消费巨多，而女性用户则青睐美妆、家用电器和食物类店铺；
（3）等级为 1、5、6、7 的会员购买行为占主导，而 2、3、4 等级的用户几乎不消费，说明新用户和老用户具有更高的消费潜力；
（4）等级为 2、6 的城市购买力不足，其他等级占主导，说明用户的购买行为有地域差别；
（5）用户注册时长越短，购买行为数越多，说明用户的粘度随着时间的增加而降低。

2.2 店铺/产品维度

2.2.1 变量选择

（1）商家评分：店铺评分 shop_score 字段
（2）商家开店时长：店铺年龄字段，以年计
（3）商家粉丝数：粉丝数 fans_number 字段
（4）商家会员数：会员数 vip_number 字段
（5）商家类别：店铺主营类别 shop_category 字段
（6）产品上市时间：以年计
（7）产品品牌：产品品牌 brand 字段
（8）产品类别：产品类别 category 字段
（9）用户购买行为：用户行为类型 type 里的 Order 或直接用 Order 字段

2.2.2 呈现关系

不同类别的比较关系

2.2.3 图表选择

（1）商家评分与购买行为：条形图

（2）商家开店时长与购买行为：条形图
（3）商家粉丝数与购买行为：条形图
（4）商家会员数与购买行为：条形图
（5）商家类别与购买行为：树地图/气泡图/词云图
（6）产品上市时间与购买行为：条形图
（7）产品品牌与购买行为：树地图/气泡图/词云图
（8）产品类别与购买行为：树地图/气泡图/词云图

2.2.4 业务结论

（1）商家的普遍评分在 8.5-10 之间波动，且用户购买行为与店铺评分无明显相关性；
（2）店铺的店龄在 0-6 年之间波动，而用户下单量主要集中在新店铺（<3 年），而 5 年以上的老店铺订单量不多；
（3）用户倾向于在粉丝数多的店铺购买产品；
（4）同样，店铺会员数越多，订单量也越多；
（5）成交量最多的店铺类别 Top5 为美妆类、食品类、衣服类、家用电器类、珠宝首饰类；
（6）用户普遍倾向于新上市的产品，上市超过 3 年的产品订单数明显下降；
（7）用户更倾向于购买其他品牌，可以侧面反映出，用户不再过分追求“大牌”；
（8）用户购买最多的产品类别为大衣、茶叶、面霜、精华、项链、基础化妆品等。

五、数据分析模型

此部分用来回答 how 的问题，使用 tableau 和 python 共同完成。

1. 问题拆解与模型选择

（1）京东的用户画像和特征-->聚类分析

（2）识别更容易产生购买行为的用户-->逻辑回归

2. 聚类分析

此处简单用 tableau 的聚类分析来建立模型。

2.1 变量选择

首先要用两个度量变量将各个用户以散点的形式定位在坐标系里，还要尽可能的分散，这里选用用户注册时长和产品上市时间两个度量，筛选出产生购买行为的用户，同时基于前几部分的分析添加其他变量（如年龄分段、产品类别、产品品牌、店铺评分、店铺会员数、店铺粉丝数），通过自带的聚类分析功能，将购买用户分成 3 个群体，群体特征如下：

		中心						最常见		
群集	项数	平均值 产品上市时间	平均值 用户注册时间	平均值 会员数	平均值 粉丝数	平均值 店铺评分		年龄分段	产品类别	产品品牌
群集 1	5850	0.87211	1.641	1.8831e+05	1.0035e+05	9.3738		3	Xbox	Kisses
群集 2	1883	3.4724	2.7759	2.443e+05	1.0388e+05	9.4406		4	Phone	Calbee
群集 3	2953	1.145	5.8335	2.0759e+05	1.2563e+05	9.4219		4	Skirt	Nike
未建立群集	0									

2.2 模型解读

（1）群集 1：注册时间短的新用户，较为年轻，追求新产品，热爱电子产品，不在意店铺评分以及粉丝数；
（2）群集 2：注册时间适中，较为年长，不过分追求新产品，关注店铺评分；
（3）群集 3：注册时间较长，较为年长，但喜欢新产品，购买衣物较多，偏好运动品牌。

3. 逻辑回归模型

使用 Python 的 sklearn 库来建立模型。

3.1 数据概览

```
In [3]: data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 183828 entries, 0 to 183827
Data columns (total 20 columns):
customer_id              183828 non-null int64
product_id               183828 non-null int64
action_date              183828 non-null datetime64[ns]
action_id                183828 non-null int64
type                     183828 non-null object
age_range                183745 non-null float64
gender                   183828 non-null object
customer_register_date   183828 non-null datetime64[ns]
customer_level           183828 non-null int64
city_level               183703 non-null float64
brand                    183828 non-null object
shop_id                  183828 non-null int64
category                 183828 non-null object
product_market_date      183828 non-null datetime64[ns]
vender_id                183828 non-null int64
fans_number              183828 non-null int64
vip_number               183828 non-null int64
shop_register_date       112474 non-null datetime64[ns]
shop_category            183828 non-null object
shop_score               183828 non-null float64
dtypes: datetime64[ns](4), float64(3), int64(8), object(5)
memory usage: 28.1+ MB
```

字段 type、gender、brand、category、shop_category 属于类别变量，需要特殊处理，尤其是 type，要将 order 分出来；字段 age_range、city_level 和 shop_register_date 有缺失值，需要清洗，为方便，将包含缺失值的行去掉。同时增加 is_male，customer_register_duration，shop_age，product_launch_duration 字

段（同 tableau 中的计算）。

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 112384 entries, 0 to 183824
Data columns (total 25 columns):
customer_id                  112384 non-null int64
product_id                   112384 non-null int64
action_date                  112384 non-null datetime64[ns]
action_id                    112384 non-null int64
type                         112384 non-null object
age_range                    112384 non-null float64
gender                       112384 non-null object
customer_register_date       112384 non-null datetime64[ns]
customer_level               112384 non-null int64
city_level                   112384 non-null float64
brand                        112384 non-null object
shop_id                      112384 non-null int64
category                     112384 non-null object
product_market_date          112384 non-null datetime64[ns]
vender_id                    112384 non-null int64
fans_number                  112384 non-null int64
vip_number                   112384 non-null int64
shop_register_date           112384 non-null datetime64[ns]
shop_category                112384 non-null object
shop_score                   112384 non-null float64
Order                        112384 non-null uint8
is_male                      112384 non-null uint8
customer_register_duration   112384 non-null float64
shop_age                     112384 non-null float64
product_launch_duration      112384 non-null float64
dtypes: datetime64[ns](4), float64(6), int64(8), object(5), uint8(2)
memory usage: 25.8+ MB
```

3.2 单变量分析

3.2.1 数字型变量

对数字型变量进行简单的描述指标评估：

	age_range	customer_level	city_level	fans_number	vip_number	shop_score	Order	is_male	customer_register_duration	shop_age	product_launch_duration
count	112384.000000	112384.000000	112384.000000	1.123840e+05	1.123840e+05	112384.000000	112384.000000	112384.000000	112384.000000	112384.000000	112384.000000
mean	4.614260	4.790264	3.300007	1.131981e+05	1.654101e+05	9.342639	0.095058	0.648767	3.450409	2.319193	1.428339
std	1.579324	2.371815	1.424654	3.050072e+05	3.225609e+05	1.228837	0.293296	0.477358	2.428765	1.476927	1.225100
min	1.000000	1.000000	1.000000	0.000000e+00	0.000000e+00	-1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	4.000000	1.000000	3.000000	7.762000e+03	1.970000e+04	9.360763	0.000000	0.000000	1.419178	1.005479	0.504110
50%	5.000000	5.000000	4.000000	3.098900e+04	6.538050e+04	9.490974	0.000000	1.000000	3.063014	2.183562	1.063014
75%	6.000000	7.000000	4.000000	1.135410e+05	1.889352e+05	9.617728	0.000000	1.000000	5.342466	3.276712	1.989726
max	6.000000	7.000000	6.000000	9.293487e+06	1.384168e+07	10.000000	1.000000	1.000000	14.750685	7.008219	7.347945

将数据按 Order 分组后求均值，可以看出，是否购买产品的两组间 city_level、vip_number、customer_register_duration、shop_age、is_male 间均值有明显差别。

Order	age_range	customer_level	city_level	fans_number	vip_number	shop_score	is_male	customer_register_duration	shop_age	product_launch_duration
0	4.611548	4.819717	3.299574	113747.035280	161409.890906	9.336739	0.658597	3.497562	2.332347	1.430638
1	4.640082	4.509876	3.304128	107972.532154	203491.487129	9.398805	0.555181	3.001518	2.193964	1.406453

3.2.2 类别型变量

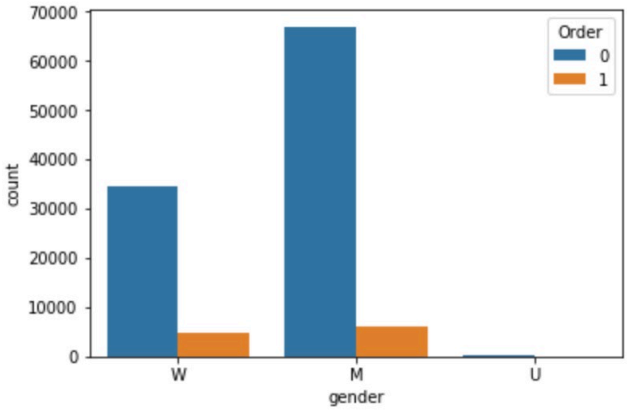
对类别型变量，简单看下各自有多少个分类，占比如何，同时对 Order 进行

分组，查看组别间的差异。

（1）性别

```
data1.gender.value_counts(1)
```

```
M    0.648767
W    0.348555
U    0.002678
Name: gender, dtype: float64
```

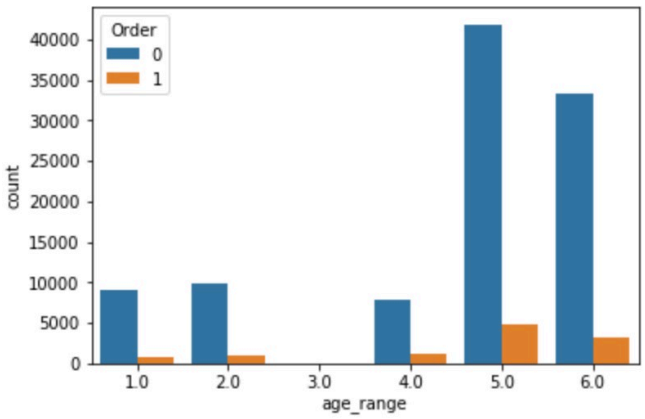


不同性别间，Order=1 没有明显差异。

（2）年龄分段

```
data1.age_range.value_counts(1)
```

```
5.0    0.414232
6.0    0.324717
2.0    0.094844
1.0    0.086560
4.0    0.079611
3.0    0.000036
Name: age_range, dtype: float64
```

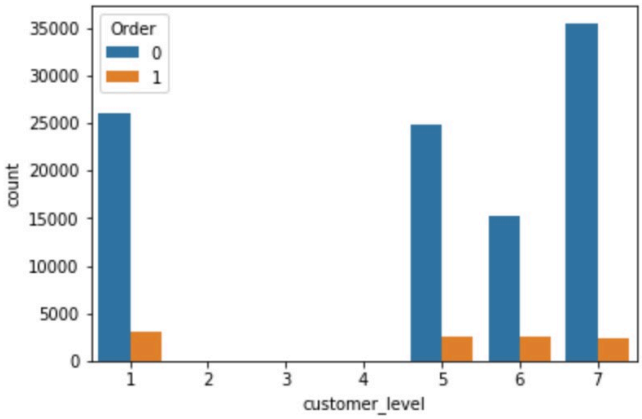


age_range=5 或 6 时，Order=1 明显高于其他组。

（3）会员级别

```
data1.customer_level.value_counts(1)
```

```
7    0.337432
1    0.259957
5    0.244012
6    0.157211
4    0.000801
3    0.000578
2    0.000009
Name: customer_level, dtype: float64
```

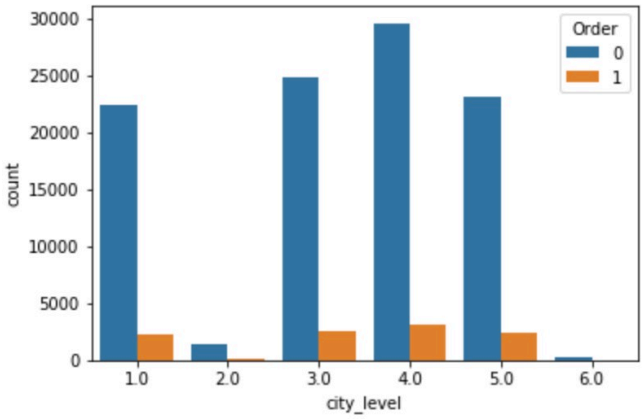


customer_level=1，5，6，7 间 Order=1 无明显差别，但明显高于 customer_level=2，3，4。

（4）会员城市级别

```
data1.city_level.value_counts(1)
```

```
4.0    0.291536
3.0    0.244964
5.0    0.226785
1.0    0.220200
2.0    0.013561
6.0    0.002954
Name: city_level, dtype: float64
```

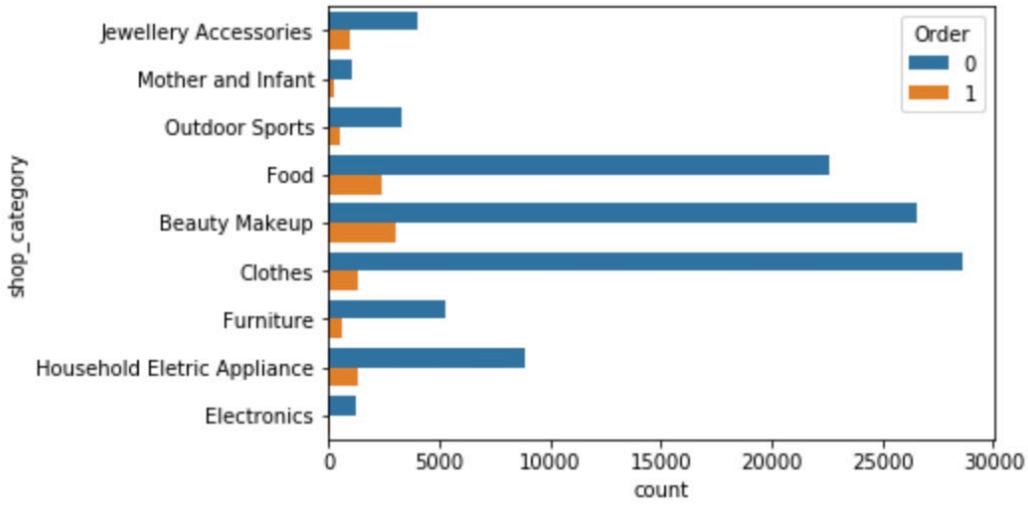


city_level=1，3，4，5 中 Order=1 明显高于 city_level=2，6。

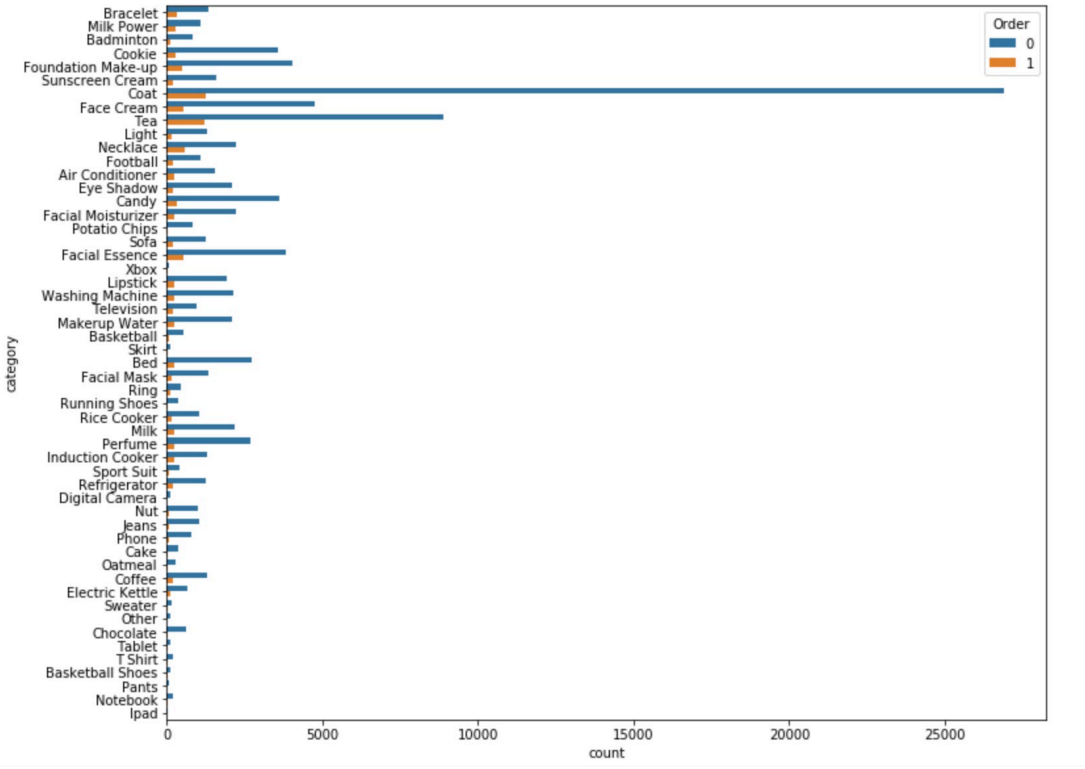
（5）店铺营业类别

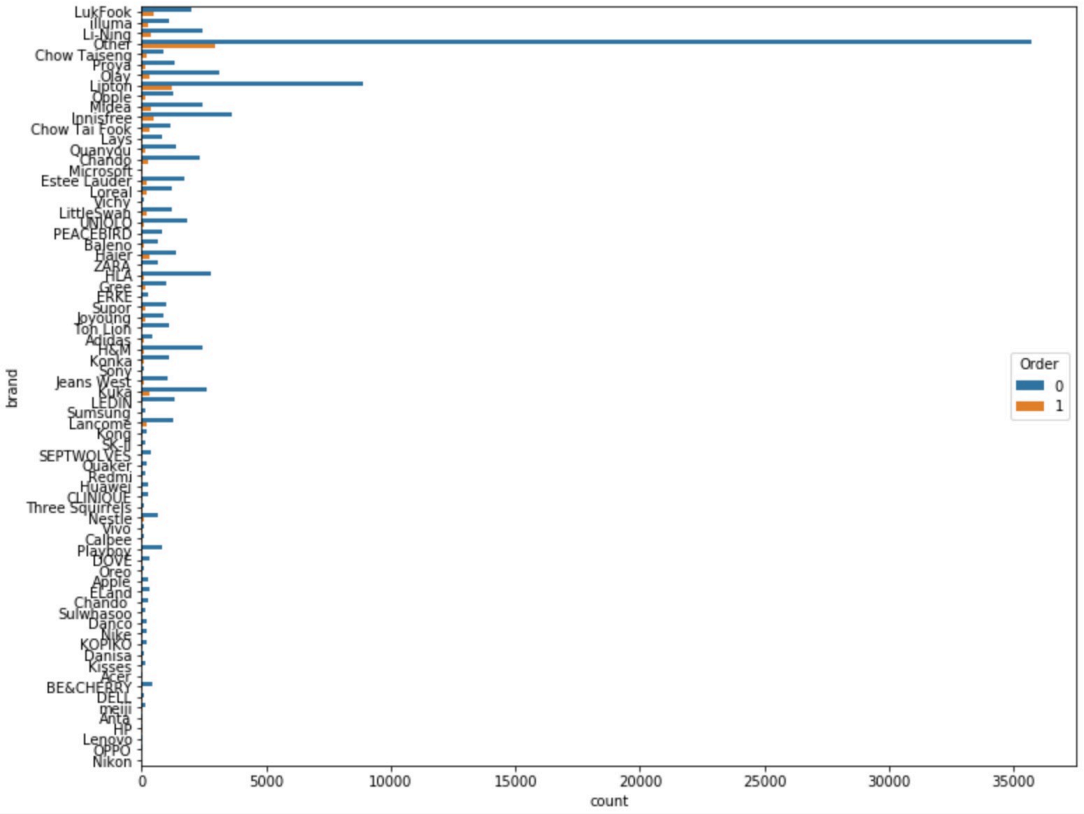
```
data1.shop_category.value_counts(1)
```

```
Clothes                        0.266675
Beauty Makeup                  0.263320
Food                           0.223128
Household Eletric Appliance    0.091428
Furniture                      0.052160
Jewellery Accessories          0.044989
Outdoor Sports                 0.034258
Electronics                    0.012030
Mother and Infant              0.012012
Name: shop_category, dtype: float64
```



（6）产品类别和产品品牌




LukFook
Illuma
Li-Ning
Other
Chow Taiseng
Proya
Olay
Lipton
Apple
Midea
Innisfree
Chow Tai Fook
Lays
Quanyou
Chando
Microsoft
Estee Lauder
Loreal
Vichy
LittleSwan
UNIQLO
PEACEBIRD
Baleno
Haier
ZARA
HLA
Gree
ERKE
Suipor
Joyoung
Toh Lion
Adidas
H&M
Konka
sony
Jeans West
Kuka
LEDIN
Sumsung
Lancome
Kong
SKII
SEPTWOLVES
Quaker
Redmi
Huawei
CLINIQUE
Three Squirrels
Nestle
Vivo
Calbee
Playboy
DOVE
Oreo
Apple
ELand
Chando
Sulwhasoo
Danco
Nike
KOPIKO
Danisa
Kisses
Acer
BE&CHEERY
DELL
meiji
Anta
HP
Lenovo
OPPO
Nikon
0
5000
10000
15000
20000
25000
30000
35000
count
brand
Order
0
1

```
data1.category.value_counts(1)

Coat                0.250516
Tea                 0.089924
Face Cream          0.047240
Foundation Make-up  0.040388
Facial Essence      0.038956
Candy               0.035121
Cookie              0.034258
Bed                 0.026365
Perfume             0.026000
Necklace            0.025137
Facial Moisturizer  0.021765
Milk                0.021435
Washing Machine     0.021151
Makerup Water       0.020519
Eye Shadow          0.020403
Lipstick            0.019140
Air Conditioner     0.015821
Sunscreen Cream     0.015634
Bracelet            0.014993
Induction Cooker    0.013676
Facial Mask         0.013276
Coffee              0.013125
Refrigerator        0.013116
Sofa                0.013000
Light               0.012795
Milk Power          0.012012
Football            0.011256
Rice Cooker         0.010642
Television          0.010108
Jeans               0.009717
Nut                 0.009681
Badminton           0.008569
Potatio Chips       0.007901
Phone               0.007439
Electric Kettle     0.006914
Chocolate           0.005543
Basketball          0.005348
Ring                0.004858
Sport Suit          0.004378
Running Shoes       0.003630
Cake                0.003452
Oatmeal             0.002687
T Shirt             0.002091
Notebook            0.001806
Sweater             0.001370
Skirt               0.001272
Tablet              0.001112
Other               0.001103
Basketball Shoes    0.001077
Digital Camera      0.000934
Pants               0.000605
Xbox                0.000561
Ipad                0.000178
Name: category, dtype: float64
```

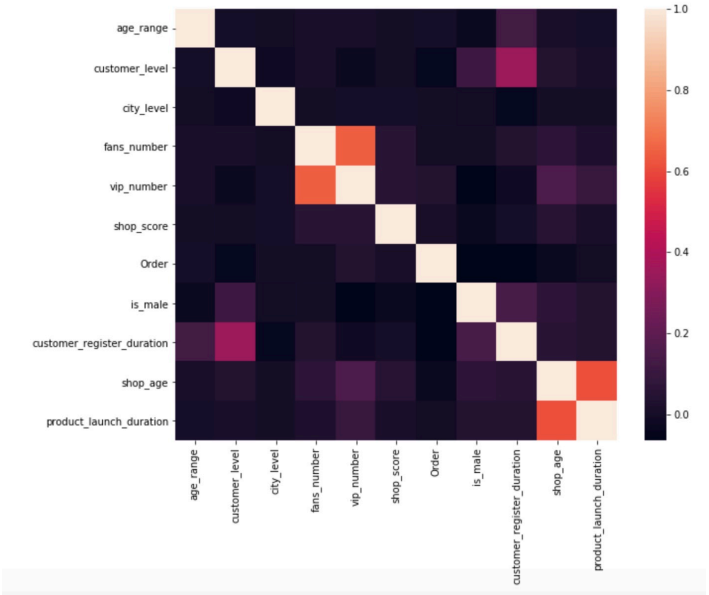
```
data1.brand.value_counts(1)

Other           0.344017
Lipton          0.089924
Innisfree       0.036696
Olay            0.030592
Kuka            0.025742
HLA             0.025262
Li-Ning         0.025173
Midea           0.024639
Chando          0.022815
H&M             0.022574
LukFook         0.022396
UNIQLO          0.017093
Estee Lauder    0.016764
Haier           0.014842
Quanyou         0.013623
Proya           0.013134
Chow Tai Fook   0.013089
Lancome         0.013027
Opple           0.012795
LittleSwan      0.012502
LEDIN           0.012288
Loreal          0.012235
illuma          0.012012
Konka           0.010562
Supor           0.010188
Ton Lion        0.010082
Gree            0.009797
Jeans West      0.009752
Chow Taiseng    0.009503
Joyoung         0.008898
                   ...
Chando          0.002669
CLINIQUE        0.002420
Huawei          0.002322
Apple           0.002189
KOPIKO          0.001966
Quaker          0.001922
Kong            0.001797
Nike            0.001726
Danco           0.001655
Redmi           0.001353
Sulwhasoo       0.001335
Sumsung         0.001326
SK-II           0.001272
meiji           0.001272
Three Squirrels 0.001157
Kisses          0.001121
Vichy           0.001112
DELL            0.001094
Danisa          0.001014
Sony            0.000917
Vivo            0.000881
Oreo            0.000783
Calbee          0.000765
Microsoft       0.000561
HP              0.000356
Anta            0.000285
Acer            0.000222
OPPO            0.000151
Lenovo          0.000133
Nikon           0.000018
Name: brand, Length: 72, dtype: float64
```

3.3 相关性分析

对所有数值变量与 Order 进行相关性分析并用热力图显示。

	Order
Order	1.000000
vip_number	0.038264
shop_score	0.014814
age_range	0.005299
city_level	0.000938
fans_number	-0.005553
product_launch_duration	-0.005790
shop_age	-0.027481
customer_level	-0.038315
customer_register_duration	-0.059902
is_male	-0.063541



可以看出，用户购买行为与店铺会员数、店铺评分、年龄层正相关，其余负相关；店铺会员数和店铺评分与用户购买行为的相关系数较高。

3.4 数据分析模型

处理后的数据，Order 中 1 占 10%左右，适合训练模型。

```
data1.Order.value_counts()
```

```
0    101701
1     10683
Name: Order, dtype: int64
```

3.4.1 变量的选择

优先选取与用户购买行为相关性高的变量作为自变量，如 vip_number、shop_score 、 fans_number 、 customer_register_duration 、 shop_age 、 is_male 、

customer_level；因变量为 Order。

3.4.2 训练集与测试集分割

调用 sklearn.model_selection 里面的 train_test_split 函数对自变量、因变量进行分割，比例为测试集：训练集=0.3:0.7。

3.4.3 数据标准化

首先要对数据进行标准化，选择 sklearn.preprocessing 里面的 scale 函数对自变量进行标准化处理，而后训练模型。

3.4.4 模型选择

调用 sklearn.linear_model 里面的 LogisticRegression 模型，参数均为默认值。

3.4.5 模型评估

（1）利用训练集和测试集给模型打分，即准确率（accuracy）。

```
lr.score(X_test, y_test)
```

```
0.9033100011863804
```

```
lr.score(X_train, y_train)
```

```
0.9055270249656785
```

（2）调用 sklearn.metrics 里面的 classification_report 计算查准率（precision）和查全率（recall）。

	precision	recall	f1-score	support
order=0	0.90	1.00	0.95	30459
order=1	0.20	0.00	0.00	3257
avg / total	0.84	0.90	0.86	33716

precision 表示为模型预测用户产生购买行为，预测准确的概率。

recall 表示为用户产生购买行为时，模型预测准确的概率。

可以发现，对于用户不产生购买行为的情况（order=0）查准率和查全率均很高，然而当用户产生购买行为时，模型不能完全查出真实用户购买情况，同时模型判断用户会发生购买行为，大约只有 25%的准确度。因此该模型需要优化。

（3）模型参数解读

```
print(lr.coef_, lr.intercept_)
```

```
[[ 0.23342826  0.05595638 -0.20944254 -0.158877   -0.10231849 -0.16852815
  -0.05341331]] [-2.31128553]
```

由于各变量取值范围较大，因此不好评判自变量是如何具体影响因变量的；正系数为正向影响，且系数越大，程度越大。

3.4.6 模型优化

（1）增大测试集的比例至 50%

发现用户产生购买行为的查准率略有提高，但不明显；查全率仍没有提高。

	precision	recall	f1-score	support
order=0	0.91	1.00	0.95	50883
order=1	0.25	0.00	0.00	5309
avg / total	0.84	0.91	0.86	56192

训练集准确率下降而测试集准确率上升，但效果不明显。

```
lr.score(X_test, y_test)
```

```
0.9054135820045558
```

```
lr.score(X_train, y_train)
```

```
0.9042924259681093
```

（2）修改自变量

经过反复测试，并没有很大提高，这可能与前期数据的特征工程这一部分没有做好有关，也可能本身数据集不够完善。

（3）尝试 KNN 模型（k=5）

经过试验，对于用户产生购买行为时，模型预测准确的概率升高，但是模型预测用户会购买的准确率稍有下降。

	precision	recall	f1-score	support
order=0	0.90	0.99	0.95	30459
order=1	0.18	0.02	0.04	3257
avg / total	0.83	0.90	0.86	33716