

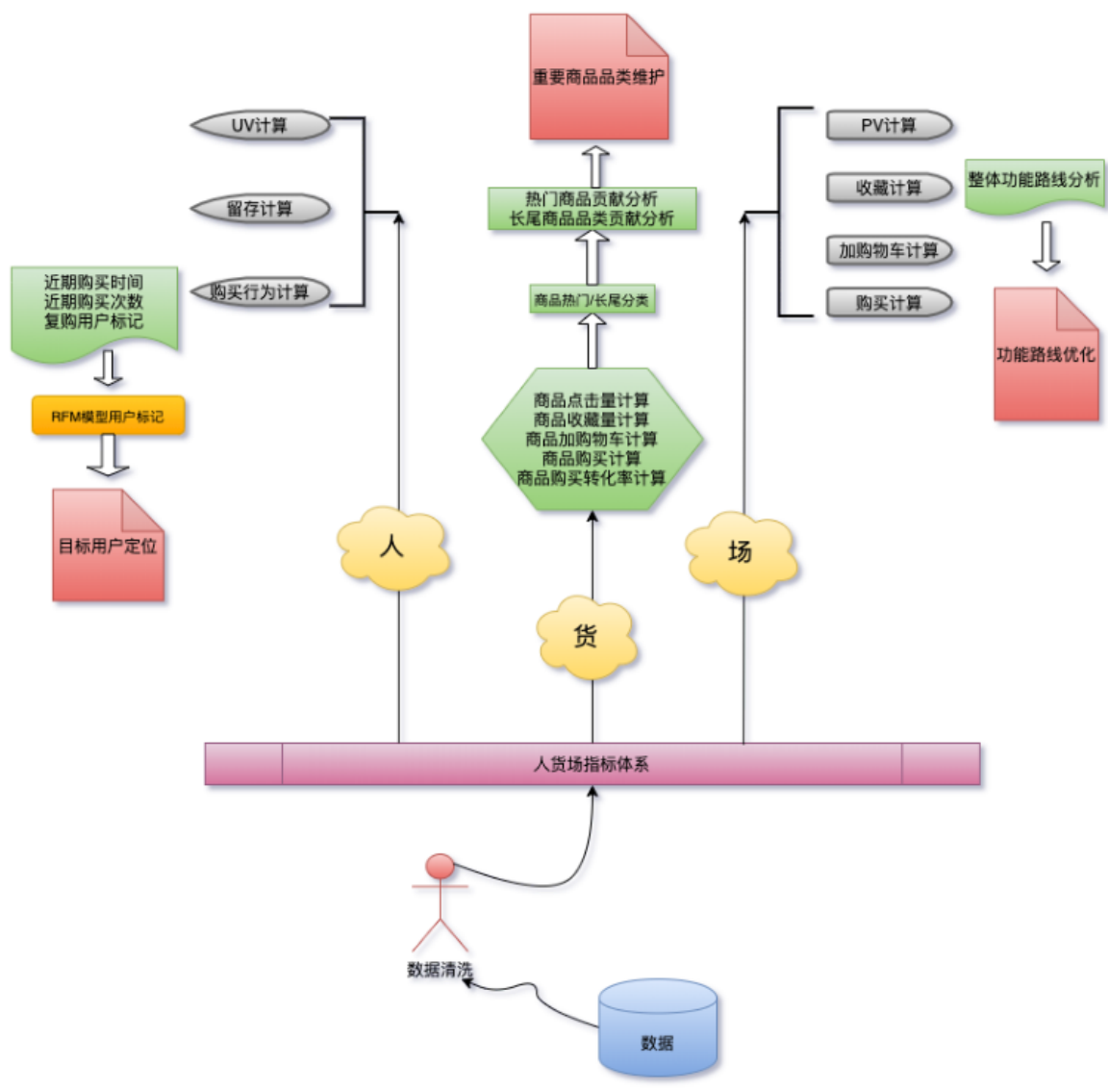
基于 MySQL 的电商用户、商品、平台价值分析

1、项目背景

如今，电商行业逐步向精细化运营发展，随着平台数据量的不断积累，通过数据分析来挖掘客户的潜在需求和消费偏好已经成为重要环节。

本项目基于某电商平台的用户行为数据，借助 MySQL 关系型数据库，探索用户行为规律，寻找高价值用户；分析商品特征，寻找高贡献商品；分析产品功能，优化产品路径。

1.1 分析流程



2、使用“人货场”拆解方式建立指标体系

目的：评价“用户、商品、平台”三者质量

人货场

a、【人】是整个运营的核心。所有营销措施都是围绕着，如何让更多的人有购买行为，让他们买的更多，买的更贵。所以我们需要分析，目前平台上的主力消费人群有哪些特征，他们对货品有哪些需求，他们活跃在哪些场，还有哪些有消费力的人不在平台上.....

指标	细化指标	说明
浏览	PV	PageView 页面浏览量
	UV	Unique Visitor 独立访客数，一定时间内访问一个网站或页面的人数
流量质量	PV/UV	浏览深度
	ROI	投资回报率 = 年利润或年均利润 / 投资总额 x 100%
成交用户	新客数	当日激活或新增用户
	老客数	
	客单价	当日消费总额/客户数（新+老）。客单价应该维持稳定，突然多/少。后卖力特别强的，疫情影响客单价下降
	DAU	日活跃用户
	MAU	月活跃用户
	留存率	留存率 = 第t+n日活跃量 / t日激活量

b、【货】对应供给，涉及到了货品分层，哪些是红海，哪些是蓝海，如何进行动态调整，是做自营还是平台，以满足消费者需求

c、【场】消费者在什么场景下，以什么方式接触到这个商品。例如，淘宝、京东在导购，内容营销方面较弱，微信的 KOL 生态、小红书、微博的内容营销做的很好，但不在电商自己的场。如何做一个全场的打通，和消费者进行多点接触，比如社交和电商联动，来完成销售转化，这就是阿里和腾讯一直在讲的全域营销。

3、确认问题

本次分析的目的在于，通过用户行为数据分析，为以下问题提供解释和改进建议：

- 1) 基于漏斗模型的用户购买流程各环节分析指标，确定各个环节的转换率，以便于找到需要改进的环节
- 2) 商品分析：找出热销商品，研究热销商品特点
- 3) 基于 RFM 模型找出核心付费用户群，对这一部分用户进行精准营销

4、准备工作

4.1 数据读取

创建表结构，然后利用向导导入数据

```
CREATE TABLE o_retailers_trade_user
(
  user_id INT (9),
  item_id INT (9),
  behavior_type INT (1),
  user_geohash VARCHAR (14),
  item_category INT (9),
  time VARCHAR (13) #数据时间格式为2020-12-12 01
)
```

4.2 数据预处理

1、转换源数据的时间格式，增加日期列

```
#转换数据格式 date_time
ALTER TABLE o_retailers_trade_user ADD COLUMN date_time datetime null
# %H 表示0-23, %h 表示0-12
UPDATE o_retailers_trade_user SET date_time = STR_TO_DATE(time,'%Y-%m-%d %H')

#增加新列 dates
ALTER TABLE o_retailers_trade_user ADD COLUMN dates CHAR(10) NULL
UPDATE o_retailers_trade_user SET dates = DATE(date_time)

DESC o_retailers_trade_user
SELECT * FROM o_retailers_trade_user LIMIT 10
```

2、对源数据去重

```
-- 重复值处理：创建新表
CREATE TABLE temp_trade LIKE o_retailers_trade_user
INSERT INTO temp_trade SELECT DISTINCT * FROM o_retailers_trade_user

SELECT * FROM o_retailers_trade_user LIMIT 10
```

处理后的数据，取前10行，如下

信息	结果 1	剖析	状态				
user_id	item_id	behavior_type	user_geohash	item_category	time	date_time	dates
98047837	232431562	1	(Null)	4245	2019-12-06 02	2019-12-06 02:00:00	2019-12-06
97726136	383583590	1	(Null)	5894	2019-12-09 20	2019-12-09 20:00:00	2019-12-09
98607707	64749712	1	(Null)	2883	2019-12-18 11	2019-12-18 11:00:00	2019-12-18
98662432	320593836	1	96nn52n	6562	2019-12-06 10	2019-12-06 10:00:00	2019-12-06
98145908	290208520	1	(Null)	13926	2019-12-16 21	2019-12-16 21:00:00	2019-12-16
93784494	337869048	1	(Null)	3979	2019-12-03 20	2019-12-03 20:00:00	2019-12-03
94832743	105749725	1	(Null)	9559	2019-12-13 20	2019-12-13 20:00:00	2019-12-13
95290487	76866650	1	(Null)	10875	2019-11-27 16	2019-11-27 16:00:00	2019-11-27
96610296	161166643	1	(Null)	3064	2019-12-11 23	2019-12-11 23:00:00	2019-12-11
100684618	21751142	3	(Null)	2158	2019-12-05 23	2019-12-05 23:00:00	2019-12-05

5、指标体系建设

5.1 用户指标体系

主要分析 PV、UV、PV/UV 即浏览深度（按日计算）、用户留存率、RFM用户模型。

1、PV、UV、PV/UV 即浏览深度（按日计算）

```
# PV按behavior_type=1计算，UV按distinct用户数计算

SELECT
    dates,
    COUNT(IF(behavior_type=1,user_id,NULL)) PV,
    COUNT(DISTINCT user_id) UV,
    ROUND(COUNT( IF(behavior_type=1,user_id,NULL))/COUNT(DISTINCT user_id),2)
    'PV/UV'
FROM temp_trade
GROUP BY dates
```

部分查询结果

信息	结果 1	结果 2	结果 3	剖析	状态
dates	PV	UV	PV/UV		
▶ 2019-11-18	1332	226	5.89		
2019-11-19	1286	222	5.79		
2019-11-20	1289	231	5.58		
2019-11-21	1247	232	5.38		
2019-11-22	1203	226	5.32		
2019-11-23	1516	241	6.29		
2019-11-24	1322	236	5.60		
2019-11-25	1395	243	5.74		
2019-11-26	1345	221	6.09		
2019-11-27	1331	232	5.74		
2019-11-28	1369	240	5.70		
2019-11-29	1478	243	6.08		
2019-11-30	1767	244	7.24		
2019-12-01	1372	245	5.60		
2019-12-02	1477	241	6.13		

2、最近一周的用户留存数、留存率

```
# 留存
WITH temp_table_trades as
(SELECT
    dates1 dates,
    COUNT(distinct user_id) num,
    COUNT(distinct if(DATEDIFF(dates2,dates1)=1,user_id,null)) remain1,
    COUNT(distinct if(DATEDIFF(dates2,dates1)=2,user_id,null)) remain2,
    COUNT(distinct if(DATEDIFF(dates2,dates1)=3,user_id,null)) remain3,
    COUNT(distinct if(DATEDIFF(dates2,dates1)=4,user_id,null)) remain4,
    COUNT(distinct if(DATEDIFF(dates2,dates1)=5,user_id,null)) remain5,
    COUNT(distinct if(DATEDIFF(dates2,dates1)=6,user_id,null)) remain6,
```

```

COUNT(distinct if(DATEDIFF(dates2,dates1)=7,user_id,null)) remain7
FROM
    (SELECT DISTINCT t1.user_id user_id,t1.dates dates1,t2.dates dates2
    FROM temp_trade t1
    LEFT JOIN temp_trade t2
    ON t1.user_id = t2.user_id
    WHERE t1.dates<=t2.dates)sub1
GROUP BY dates1)

SELECT dates,num,
CONCAT(ROUND((remain1/num)*100,2),'%') day1,
CONCAT(ROUND((remain2/num)*100,2),'%') day2,
CONCAT(ROUND((remain3/num)*100,2),'%') day3,
CONCAT(ROUND((remain4/num)*100,2),'%') day4,
CONCAT(ROUND((remain5/num)*100,2),'%') day5,
CONCAT(ROUND((remain6/num)*100,2),'%') day6,
CONCAT(ROUND((remain7/num)*100,2),'%') day7
from temp_table_trades

```

留存率的部分查询结果

信息	结果 1	结果 2	结果 3	剖析	状态				
dates	num	day1	day2	day3	day4	day5	day6	day7	
▶ 2019-11-18	226	65.04%	64.60%	60.18%	61.95%	63.27%	56.19%	60.18%	
2019-11-19	222	69.37%	63.51%	61.71%	63.51%	65.77%	65.32%	57.21%	
2019-11-20	231	63.20%	59.31%	61.90%	64.94%	62.77%	56.71%	57.14%	
2019-11-21	232	62.07%	62.93%	65.09%	65.52%	58.19%	61.21%	62.50%	
2019-11-22	226	69.47%	64.16%	65.49%	60.18%	64.60%	61.95%	63.27%	
2019-11-23	241	66.80%	66.39%	58.51%	61.00%	63.07%	65.56%	63.49%	
2019-11-24	236	69.07%	63.98%	66.53%	65.25%	66.53%	68.22%	65.25%	
2019-11-25	243	64.61%	63.37%	66.26%	63.37%	68.72%	62.96%	61.73%	
2019-11-26	221	67.87%	67.42%	65.16%	63.80%	61.54%	59.73%	56.56%	
2019-11-27	232	70.69%	66.38%	65.52%	67.24%	61.64%	59.48%	62.07%	
2019-11-28	240	67.92%	63.33%	64.17%	59.17%	61.25%	58.75%	57.50%	
2019-11-29	243	69.55%	67.49%	63.79%	60.49%	60.08%	62.96%	67.08%	
2019-11-30	244	63.52%	68.03%	64.75%	65.98%	64.34%	66.39%	64.75%	
2019-12-01	245	66.12%	60.41%	66.12%	67.35%	64.49%	62.45%	63.27%	
2019-12-02	241	67.22%	66.80%	67.22%	66.80%	65.56%	64.32%	64.73%	

3、RFM 模型分析 (recency、frequency、monetary)

R: 用户最近一次购买的时间,

F: 一段时间内 (一个月) 用户购买的频率,

M: 一段时间内 (一个月) 用户购买的金额。

(由于源数据没有购买金额的数据, 所以这里只分析 R、F)

由于 R、F 计算出来不好直接比较, 所以将 R、F 统一量纲, 映射到 1-5 之间, 分数越高用户价值越高

R: 依据距离的远近给用户评分, 时间越近分数越高, 评分范围 (1-5), 对应购买时间与最大日期差值 (8+, 7-8, 5-6, 3-4, 1-2)

F: 依据近期购买频次划分, 频次越高分数越高, 评分范围 (1-5), 对应购买频次 (1-2, 3-4, 5-6, 7-8, 8+)

R部分

```
DROP view if EXISTS user_recency;
CREATE VIEW user_recency
AS
SELECT user_id,MAX(dates) dates
FROM temp_trade
WHERE behavior_type=2
GROUP BY user_id
ORDER BY dates DESC
```

划分r等级，按近期购买时间划分，时间越近分数越高

```
DROP VIEW IF EXISTS r_level;
CREATE VIEW r_level
AS
SELECT *,
datediff('2019-12-18',dates) r_num,
CASE
    WHEN datediff('2019-12-18',dates)<=2 THEN 5
    WHEN datediff('2019-12-18',dates)<=4 THEN 4
    WHEN datediff('2019-12-18',dates)<=6 THEN 3
    WHEN datediff('2019-12-18',dates)<=8 THEN 2
    ELSE 1 END r_value
FROM user_recency
```

F部分

```
DROP VIEW IF EXISTS user_frequency;
CREATE VIEW user_frequency
AS
SELECT user_id,COUNT(*) f_num
FROM temp_trade
WHERE behavior_type=2
GROUP BY user_id
ORDER BY f_num DESC
```

划分f等级，按近期购买频次划分，频次越高分数越高

```
DROP VIEW IF EXISTS f_level;
CREATE VIEW f_level
AS
SELECT *,
CASE
    WHEN f_num<=2 THEN 1
    WHEN f_num<=4 THEN 2
    WHEN f_num<=6 THEN 3
    WHEN f_num<=8 THEN 4
    ELSE 5 END f_value
FROM user_frequency
```

整合 R、F 等级，给用户划分类别（由于没有M值，所以只有 4 种客户分类）

- 重要高价值用户：消费时间近、频率高
- 重要唤回客户：消费时间远、频率高
- 重要深耕客户：消费时间近、频率低

-- 重要挽留客户：消费时间远、频率低

以 R、F 的平均分为界限，划分用户的时间远近、频次高低

```
# R、F平均值
#SELECT AVG(r_value) r_avg FROM r_level;    -- r_avg=2.7939
#SELECT AVG(f_value) f_avg FROM f_level; -- f_avg=2.2606

DROP VIEW IF EXISTS RFM_inall;
CREATE VIEW RFM_inall
AS
SELECT r.user_id,r_value,f_value,
CASE
    WHEN r_value>2.7939 and f_value>2.2606 THEN '重要高价值用户'
    WHEN r_value<2.7939 and f_value>2.2606 THEN '重要唤回用户'
    WHEN r_value>2.7939 and f_value<2.2606 THEN '重要深耕用户'
    WHEN r_value<2.7939 and f_value<2.2606 THEN '重要挽回用户'
END user_class
FROM r_level r,f_level f
WHERE r.user_id=f.user_id;

#查询各个分类用户的数目

SELECT user_class,count(user_id) num
FROM rfm_inall
GROUP BY user_class
```

最后各个分类的用户数量如下

信息	结果 1	结果 2	剖析	状态
	user_class	num		
▶	重要深耕用户	44		
	重要高价值用户	44		
	重要唤回用户	11		
	重要挽回用户	66		

5.2 商品指标体系

商品的浏览量、收藏量、加购量、购买量，购买转化（购买用户与浏览用户的比值）

```
# 商品点击量、收藏量、加购量、购买次数、购买转化
SELECT item_id,
COUNT(if(behavior_type=1,item_id,NULL)) pv,
COUNT(if(behavior_type=4,item_id,NULL)) liken,
COUNT(if(behavior_type=3,item_id,NULL)) addn,
COUNT(if(behavior_type=2,item_id,NULL)) payn,
COUNT(distinct if(behavior_type=2,user_id,NULL))/COUNT(DISTINCT user_id) payv
FROM temp_trade
GROUP BY item_id
ORDER BY payn DESC;
```

查询部分结构如下

信息	解释 1	结果 1	结果 2	剖析	状态	
item_id		pv	likn	addn	payn	payv
396972151		2	0	0	2	0.5000
262644230		0	0	0	2	1.0000
289585954		1	0	0	2	1.0000
279902408		1	0	0	1	1.0000
280916644		1	0	0	1	1.0000
281409852		0	1	0	1	1.0000
281796945		0	0	0	1	1.0000
282268542		3	0	0	1	1.0000
282450136		1	0	0	1	1.0000
282510225		0	0	0	1	1.0000
283137404		0	0	0	1	1.0000
283370309		0	0	0	1	1.0000
283596456		0	0	0	1	1.0000
283828646		0	0	0	1	1.0000

品类的浏览量、收藏量、加购量、购买量，购买转化（购买用户与浏览用户的比值）

```
# 品类点击量、收藏量、加购量、购买次数、购买转化
SELECT item_category,
COUNT(if(behavior_type=1,item_id,NULL)) pv,
COUNT(if(behavior_type=4,item_id,NULL)) liken,
COUNT(if(behavior_type=3,item_id,NULL)) addn,
COUNT(if(behavior_type=2,item_id,NULL)) payn,
COUNT(distinct if(behavior_type=2,user_id,NULL))/COUNT(distinct user_id) payv
FROM temp_trade
GROUP BY item_category
ORDER BY payn DESC;
```

查询的部分截图如下

信息	解释 1	结果 1	结果 2	剖析	状态	
item_category		pv	likn	addn	payn	payv
▶	13230	1481	2	35	43	0.1250
	5894	1336	2	20	36	0.1236
	1863	1461	11	41	34	0.1137
	6513	1030	7	29	31	0.1344
	11279	767	4	22	25	0.1259
	10894	491	2	15	20	0.1261
	5027	1498	2	27	19	0.0732
	2825	608	0	11	15	0.0930
	5399	1131	4	18	15	0.0488
	3628	316	6	6	13	0.0800
	4370	493	4	15	13	0.0897
	6054	320	1	10	12	0.1375
	10392	308	6	9	11	0.1064
	5395	403	0	5	10	0.0676
	5503	347	4	6	9	0.0659

5.3 平台指标体系

平台的浏览数、收藏数、加购物车数、购买数、购买转化（平台当日的所有用户中有购买转化的用户比）

#点击次数、收藏次数、加购物车次数、购买次数、购买转化（平台当日的所有用户中有购买转化的用户比）

```
SELECT dates,COUNT(*) num,
COUNT(if(behavior_type=1,user_id,NULL)) pv,
COUNT(if(behavior_type=4,user_id,NULL)) liken,
COUNT(if(behavior_type=3,user_id,NULL)) addn,
COUNT(if(behavior_type=2,user_id,NULL)) payn,
COUNT(distinct if(behavior_type=2,user_id,NULL))/COUNT(DISTINCT user_id) payv
FROM temp_trade
GROUP BY dates
ORDER BY payn DESC;
```

查询结果如下

信息	结果 1	结果 2	剖析	状态			
dates	num	p	v	liken	addn	payn	payv
▶ 2019-12-12	2855	2620	81	107	47	0.0926	
2019-11-20	1371	1289	11	31	40	0.1039	
2019-12-10	1836	1731	15	51	39	0.0984	
2019-12-11	1849	1729	9	72	39	0.0772	
2019-11-21	1343	1247	8	51	37	0.1078	
2019-11-30	1861	1767	20	41	33	0.0943	
2019-12-02	1562	1477	8	44	33	0.0622	
2019-12-18	1363	1282	17	31	33	0.1027	
2019-11-29	1581	1478	12	59	32	0.0905	
2019-12-08	1604	1513	10	49	32	0.1012	
2019-11-25	1501	1395	21	54	31	0.0864	
2019-11-19	1368	1286	15	37	30	0.0811	
2019-12-05	1481	1412	10	30	29	0.0909	
2019-12-17	1452	1367	12	44	29	0.0756	
2019-12-14	1555	1480	17	30	28	0.0936	

分析用户在购买商品前的行为路径

```
#用户购买行为路径分析
#DROP VIEW IF EXISTS user_way;
#CREATE VIEW user_way
#AS
SELECT user_way,COUNT(*) num FROM
(SELECT user_id,item_id,GROUP_CONCAT(behavior_type ORDER BY date_time SEPARATOR
'-') user_way
FROM temp_trade
GROUP BY user_id,item_id
HAVING user_way REGEXP '2+')sub1
GROUP BY SUBSTRING_INDEX(user_way,'2',1);
```

有购买行为的不同行为路径的用户数

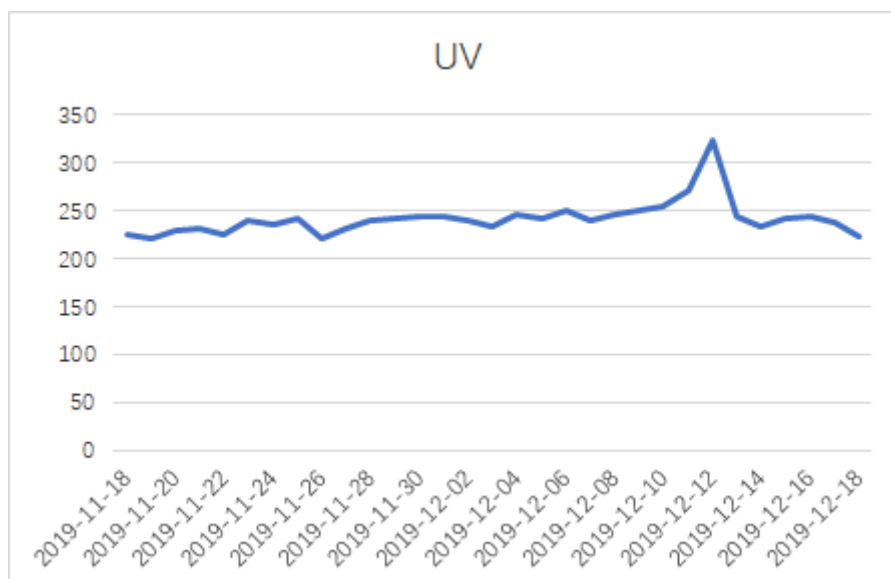
信息	结果 1	结果 2	剖析	状态
	user_way	num		
▶	2	730		
	1-2	109		
	1-1-2	12		
	4-2	2		
	3-2	2		
	1-1-1-2	2		
	3-1-2	1		

6、结论

将指标体系建设中的查询结果导出，结合数据可视化效果，做如下分析

6.1 用户分析

1、UV 异常分析：每日UV 数据中，明显异常点为双十二期间的活动造成，该影响为已知影响。



2、UV 周环比的分析：日常周环比数据大多大于0，说明用户呈一定上升趋势。但数据中存在 11-26、12-2、12-7、12-16等数据为下降数据，需要结合其他数据进行原因分析。

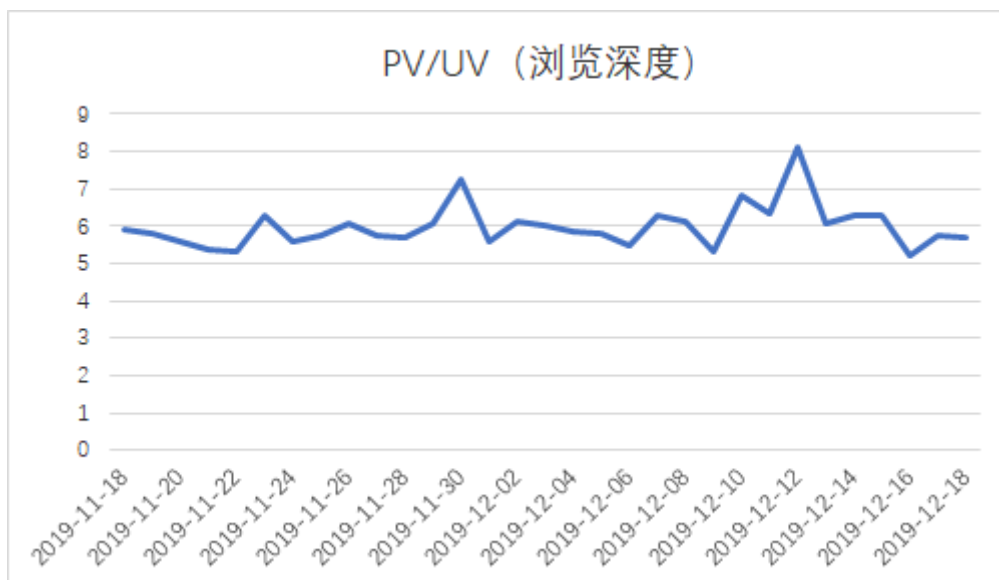
其中双十二活动后，12-16之后的用户周环比会相应下降，为正常原因。

对于双十二之前日期的周环比下降，猜测可能的原因有：

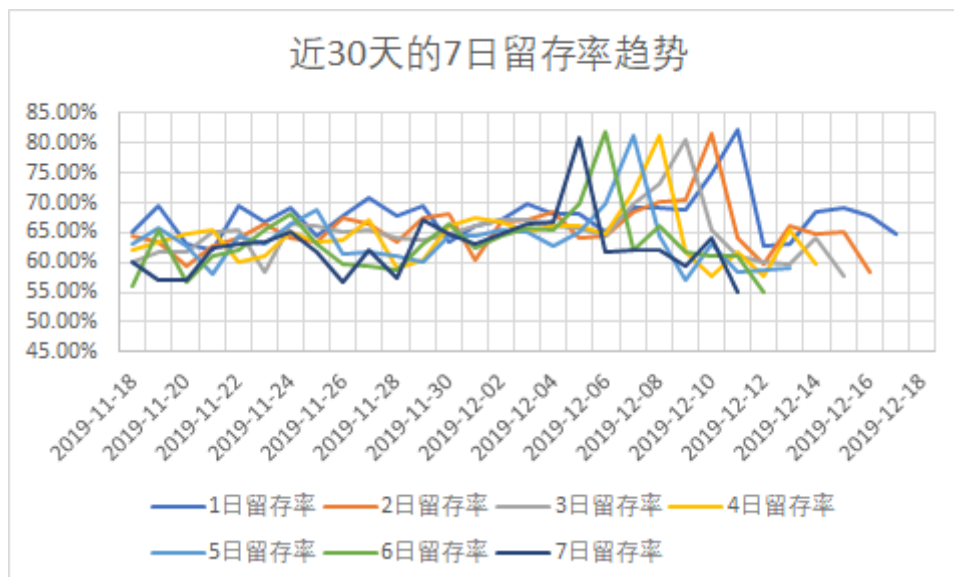
内部问题：产品bug（网站bug）、策略原因（周年庆活动结束了）、营销原因（代言人换了）等

外部原因：竞品活动问题（其他平台大酬宾）、政治环境问题（进口商品限制）、舆情口碑问题（平台商品爆出质量问题）等

3、通过 PV/UV 数据可以看到，页面的平均浏览深度为6左右；



4、留存率大致在65%左右，峰值值的出现原因主要是，双十二当天用户都回到平台上活跃

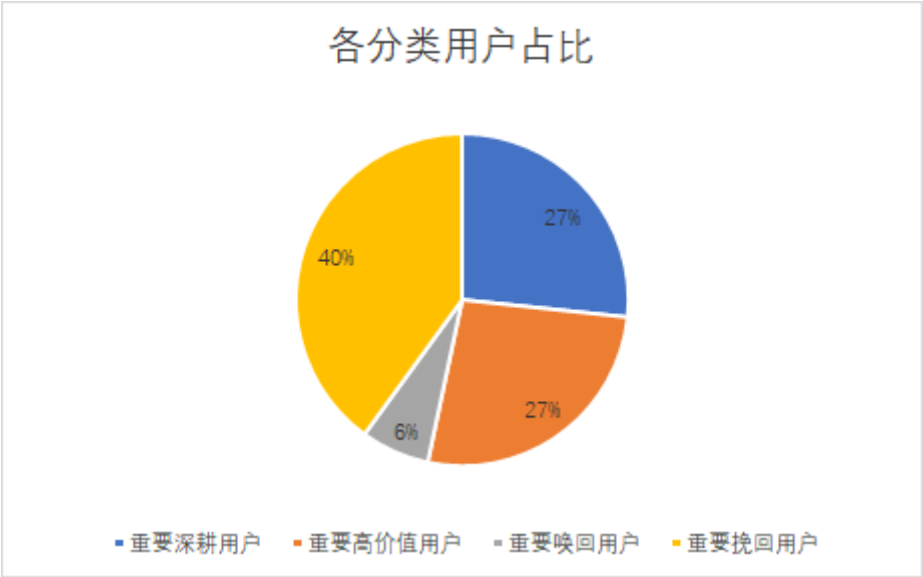


6.2 用户精细化运营分析

通过 RFM 模型中的用户最近一次购买时间，用户消费频次分析，分拆得到下列重要用户。

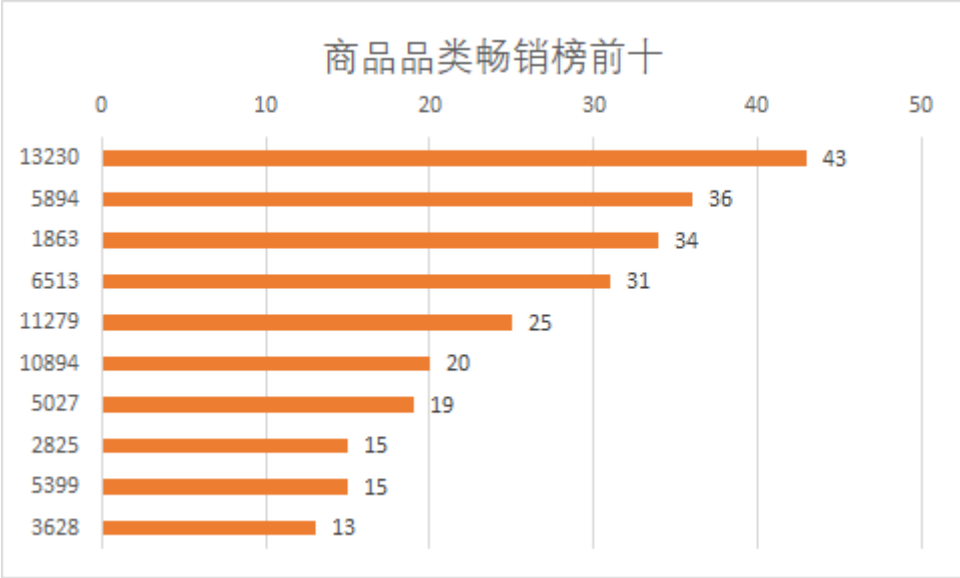
可以在后续精细化运营场景中直接使用细分用户，做差异化运营：

- 1、高价值用户：做VIP服务设计，增加用户粘性，同时通过设计优惠券提升客户消费
- 2、深耕用户：做广告、推送刺激用户，提升消费频次
- 3、挽留客户：做优惠券、签到送礼策略，增加用户粘性
- 4、唤回客户：做定向广告、短信召回策略，尝试召回客户



6.3 商品分析

热销商品品类如下图所示



其中 '5027'、'5399' 的购买转化率较低，需要结合更多数据进行进一步解读。（可能原因：品类自有特性导致用户购买较低，比如非必需品、奢侈品等等）

item_category	pv	liken	addn	payn	payv
13230	1481	2	35	43	12.50%
5894	1336	2	20	36	12.36%
1863	1461	11	41	34	11.37%
6513	1030	7	29	31	13.44%
11279	767	4	22	25	12.59%
10894	491	2	15	20	12.61%
5027	1498	2	27	19	7.32%
2825	608	0	11	15	9.30%
5399	1131	4	18	15	4.88%
3628	316	6	6	13	8.00%

6.4 产品功能路径分析

下表为用户的购买路径，可以发现用户主要以直接购买为主，添加购物车的购买在主要购买路径中数量较少。后续产品的加购功能和产品收藏功能还需结合更多的数据来做改进方案

user_way	num
2	730
1-2	109
1-1-2	12
4-2	2
3-2	2
1-1-1-2	2
3-1-2	1