# 数据准备

|  |  |
| --- | --- |
| **数据来源** | vipmodel.ds\_user\_brand\_category\_goods |
| **时间跨度** | **120days**  (dt BETWEEN get\_dt\_date('${dt}', -121) AND get\_dt\_date('${dt}', -2)) |
| **特征构造** | **bool\_click**( IF(sum(sales\_money)=0,0,1) AS bool\_buy);  **bool\_buy**( IF(sum(sales\_money)=0,0,1) AS bool\_buy) |
| **数据总量描述** | **可视化未完成** |

-----BASE DATA-----

DROP TABLE IF EXISTS temp\_ds.lh\_itemcf\_120\_${dt};

CREATE TABLE temp\_ds.lh\_itemcf\_120\_${dt} AS

SELECT

user\_id,goods\_id,ct1\_id,brand\_store\_sn,

max(brand\_store\_name) AS brand\_store\_name,

min(CASE brand\_store\_level

WHEN 'AD' THEN '1'

WHEN 'AA' THEN '2'

WHEN 'A+' THEN '3'

WHEN 'A' THEN '4'

WHEN 'A-' THEN '5'

WHEN 'B+' THEN '6'

WHEN 'B' THEN '7'

WHEN 'C' THEN '8'

ELSE '9' END) AS brand\_store\_level,

max(ct1\_name) AS ct1\_name,

max(dt) maxdt,

sum(pv) AS pv,

sum(case when sales\_qty > 0 then 1 else 0 end) AS sales\_qty,

sum(case when is\_add\_cart > 0 then 1 else 0 end) AS add\_cart\_qty,

sum(case when is\_goods\_like > 0 then 1 else 0 end) AS goods\_like\_qty,

sum(sales\_money) AS sales\_money,

sum((floor(datediff(add\_date,get\_date('${dt}', -91))/7)+1)/(datediff(get\_date('${dt}', -2),get\_date('${dt}', -91))/7+1)\*pv) AS user\_pt\_brand\_id\_sum\_pv\_time\_decay,

IF(sum(sales\_money)=0,0,1) AS bool\_buy,

IF(sum(pv)=0,0,1) AS bool\_click

FROM vipmodel.ds\_user\_brand\_category\_goods

WHERE dt BETWEEN get\_dt\_date('${dt}', -121) AND get\_dt\_date('${dt}', -2)

AND user\_id IS NOT NULL

AND user\_id > 0

AND goods\_id IS NOT NULL

GROUP BY user\_id,goods\_id,brand\_store\_sn,ct1\_id;

# 数据预处理

## 2.1 去噪

缺失值：赋零值

错误值：user\_id为负的部分（剔除）

## 归一化

对研究数据特征bool\_click进行归一化（bool\_buy同理）

IF(sum(pv)=0,0,1) AS bool\_click

## 降维

过滤掉无用户点击行为（购买）行为的用户以及未被点击（购买）的物品

--bool\_click

----USER DEGREE---

DROP TABLE IF EXISTS temp\_ds.lh\_itemcf\_user\_${dt};

CREATE TABLE temp\_ds.lh\_itemcf\_user\_${dt} AS

SELECT goods\_id,user\_id

,count(DISTINCT goods\_id) AS user\_link

FROM vipmodel.ds\_user\_brand\_category\_goods

WHERE dt BETWEEN get\_dt\_date('${dt}', -121) AND get\_dt\_date('${dt}', -2)

AND user\_id > 0

AND pv<>0

GROUP BY goods\_id,user\_id;

----ITEM DEGREE---

DROP TABLE IF EXISTS temp\_ds.lh\_itemcf\_item\_${dt};

CREATE TABLE temp\_ds.lh\_itemcf\_item\_${dt} AS

SELECT goods\_id

,count(DISTINCT user\_id) AS item\_link

FROM vipmodel.ds\_user\_brand\_category\_goods

WHERE dt BETWEEN get\_dt\_date('${dt}', -121) AND get\_dt\_date('${dt}', -2)

AND user\_id >0

AND pv<>0

GROUP BY goods\_id;

--------BASE FEATURE-----

DROP TABLE IF EXISTS temp\_ds.lh\_itemcf\_feature\_${dt};

CREATE TABLE temp\_ds.lh\_itemcf\_feature\_${dt} AS

SELECT a.\*

,COALESCE(b.user\_link,0) AS user\_link

,COALESCE(c.item\_link,0) AS item\_link

FROM temp\_ds.lh\_itemcf\_120\_${dt} a

LEFT OUTER JOIN temp\_ds.lh\_itemcf\_user\_${dt} b

ON a.user\_id=b.user\_id and a.goods\_id=b.goods\_id

LEFT OUTER JOIN temp\_ds.lh\_itemcf\_item\_${dt} c

ON a.goods\_id=c.goods\_id;

# 物品相似度计算[[1]](#footnote-1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **方法** | **欧式距离** | 皮尔森 | **COSINE** | **JACCARD** |
| **公式** |  |  |  |  |

--USER X ITEM PURCHASED DATA TO ITEM X ITEM MATRIX

----MATRIX CALCULATE FEATURE----

DROP TABLE IF EXISTS temp\_ds.lh\_itemcf\_matrix\_feature\_${dt};

CREATE TABLE temp\_ds.lh\_itemcf\_matrix\_feature\_${dt} AS

SELECT a.user\_id AS user\_id

,a.user\_link AS user\_link

,a.goods\_id AS goods\_id\_a

,b.goods\_id AS goods\_id\_b

,a.bool\_buy AS bool\_buy\_a

,b.bool\_buy AS bool\_buy\_b

,a.item\_link AS item\_link\_a

,b.item\_link AS item\_link\_b

FROM (SELECT \*

FROM temp\_ds.lh\_itemcf\_feature\_${dt}

WHERE user\_link<>0

AND item\_link<>0 ) a

JOIN (SELECT \*

FROM temp\_ds.lh\_itemcf\_feature\_${dt}

WHERE user\_link<>0

AND item\_link<>0 ) b

ON a.user\_id=b.user\_id

# 评估

随机抽取1%用户，利用TOP-N推荐，看用户是否购买，计算召回率：

1. 对于用户点击次数最多的十件物品进行物品相似扩散形成用户推荐物品列表；
2. 同理，对于用户购买次数最多的十件商品进行物品相似扩散形成用户推荐物品列表。

数据总体可视化

1. :A、B两个物品同时出现次数；：A物品出现次数（同理） [↑](#footnote-ref-1)