Part of feature selection
预处理：

1. missing value 处理：将order数据集里面的days_since_piror_order 里面的nan全部填充为0
2. merge order_products_prior and orders 两个数据集—命名为merge1
3. Orders csv数据集之选择eval_set==prior的observations(用subset)—命名为orders_prior

user_id feature

✅ 1.1每个用户的订单数量

dataset: merge1
Variable: groupby：user_id
Method: order_number里面的最大值

✅ 1.2每个用户购买的商品总量

merge1
按照user_id分类
计算每一类的product_id的count（）

✅ 1.3每个用户订单的平均商品数

1.2每个用户购买的商品总量/1.1每个用户的订单数量

✅ 一周之内每个用户下单最频繁是哪天

Merge1
按照user_id， order_dow分类
订单数最多对应的星期

✅ 一天之内每个用户下单最频繁是在哪段时间

Merge1
按照user_id， order_hour_of_day分类
订单数最多对应的时间

✅ 每个用户的重复下单率

Merge1
按照user_id分类
选择reorder 变量 求平均值

✅ 每个用户购物频率（两次下单时间间隔）

orders_prior.csv
按照user_id分类
a=days_since_prior_order变量 求和sum()
b= order_id 变量 count（）
频率=a/b

Product_id feature

✅ 某产品购买次数

order_product_prior.csv
按照product_id分类
选择order_id变量,求个数count（）

✅ 某产品的重复下单率—feature_1_table

order_product_prior.csv
按照product_id分类
（reordered==1的情况）reordered这一列的均值

✅ department**的重复下单率**

a=merge department &product 根据department_id
b=merge a & feature_1_table
按照department分类，选择变量-产品的重复下单率（这个feature已经前面创建过）
Department的重复下单率=产品的重复下单率求和/产品的重复下单率总数

✅ 每个产品平均投放到购物车的位置

order_product_prior.csv
按照product分类
选择add_to_cart_order变量 mean()

==========

user_id & product_id feature

✅ 某产品被该用户购买的次数 times_bought

Merge1
按照user_id，product_id分类
选择order_id变量 求count（）

✅ 某产品被某用户的重复下单率

Merge1
按照user_id，product_id分类
【1】每个用户的订单数量
total_orders=按照user_id分类，对order_id求和
【2】该产品首次出现在该用户第几次订单中
first_order_number=按照user_id和product_id分类，选择order_number的最小值
【3】在这之后购买的订单数量
n=total_orders-first_order_number+1
【4】重复下单率uxp_reorder_ratio=某产品被该用户购买的次数 times_bought / n

✅ 某产品在某用户的最后4次订单里出现的几率 uxp_last_five； uxp_ratio_last_five

选择merge1数据集
按照user_id分类，创建新列：order_number_back
order_number.max()-order_number+1 倒序排列
选择order_number<=4的observation
按照user_id，product_id进行分类，计算count（）