Instacart的消費者購物分析

F74086056 曾鈺安

題目概述

- 根據InstaCart（生鮮雜貨代買代送平台）提供的顧客訂單歷史記錄，來預測對每一位消費者而言，他們曾購買過的商品裡，有哪些商品會出現在下一份訂單，即推測他們可能再次購買那些商品。

流程

- 數據分析
- 資料預處理
- 建模與預測

數據分析

數據說明

- 此競賽數據共有6個表格，其中包含了約20萬位顧客、將近5萬種商品以及約340萬筆訂單記錄。Instacart將所有訂單數據分為三部分：prior、train、test。
- 對每一位顧客而言，每一筆訂單會購買數種商品。我們可以從prior的訂單中找出顧客先前的消費行為，並對train和test預測顧客未來的消費行為。

數據說明

- 這個比賽可視為一個二元分類問題：對每一位顧客而言，每一項購買過的產品是否會再度購買（是=1；否=0）。接下來我們會從prior訂單中建立數項指標參數進行建模，預測reordered欄位。

主鍵Primary key
(根據prior訂單購買的產品)

參數X
(根據prior訂單建立)

train/test

future orders

欲預測的變數Y

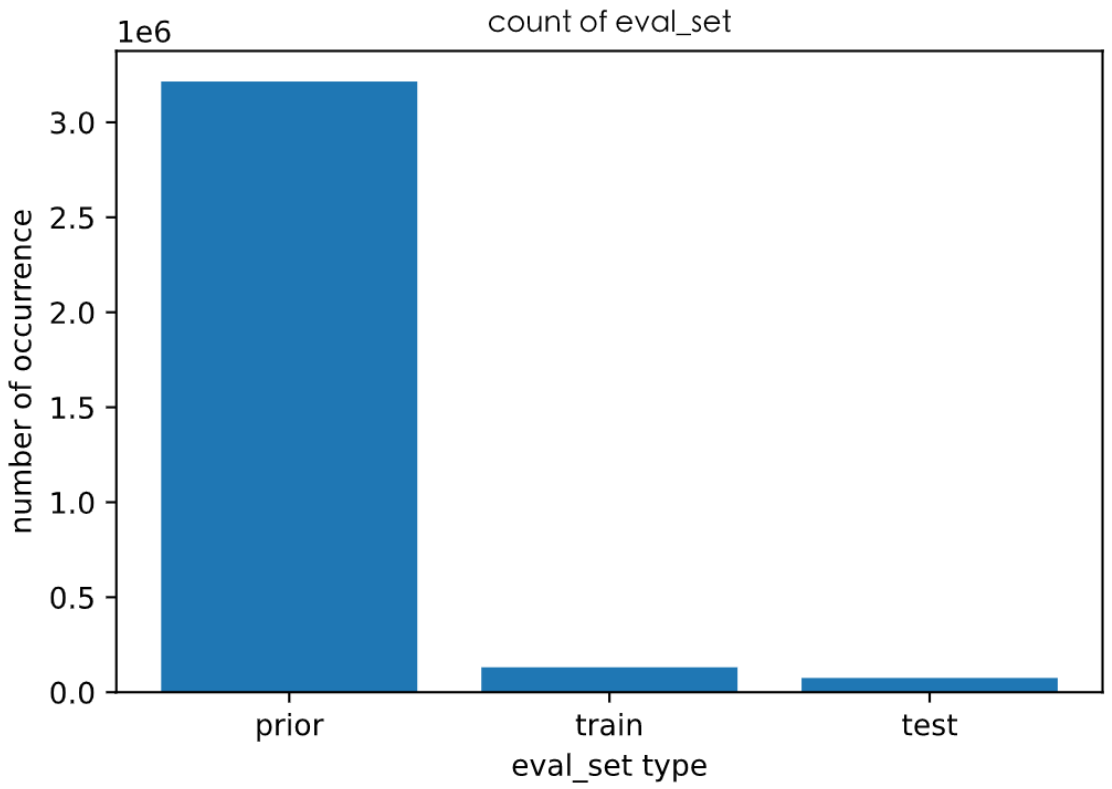
user_id	product_id	...	...	...	eval_set	order_id	reordered
1	196				train	1187899	1
1	10258				train	1187899	0
1	10486				train	1187899	1
1	10686				train	1187899	1
1	15435		...		train	1187899	0
1	12376				test	1187968	
2	11698				train	1256788	1
2	12495				train	1256788	1
2	14571				test	1257530	

數據分析

- 首先，我們可以利用探索性分析（Exploring Data Analysis），繪製一些圖表或統計數據來了解資料分布狀況，或欄位之間的關係。

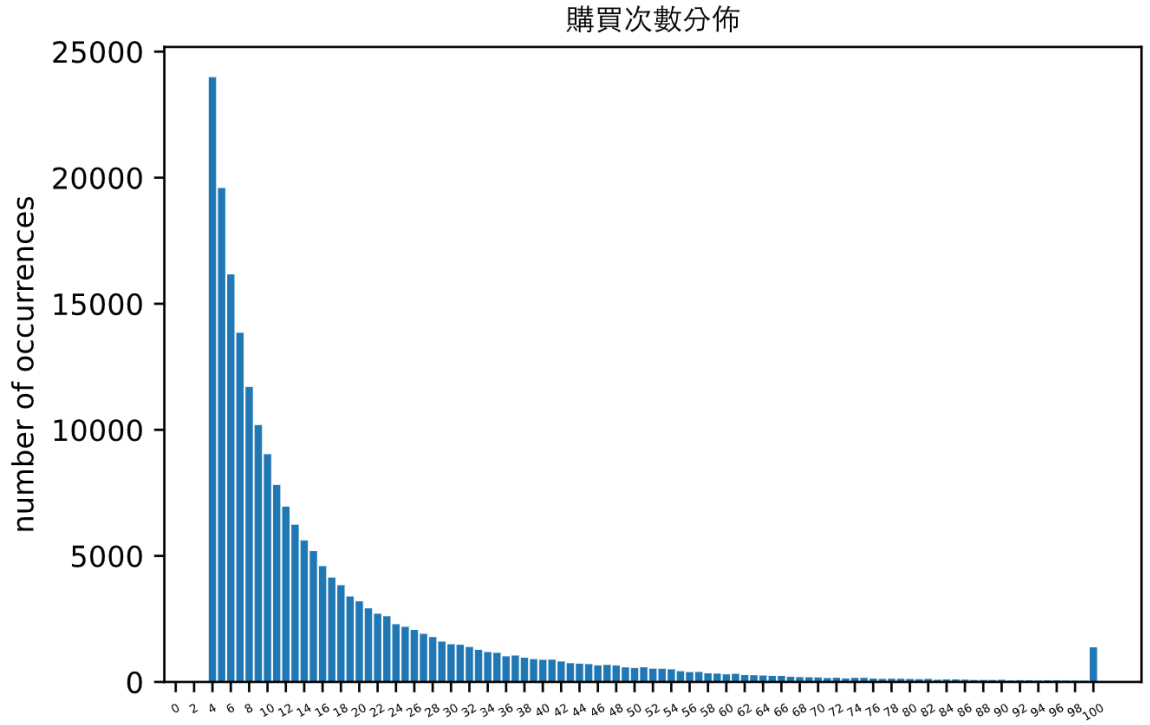
數據分析

- 此資料集共有3,421,083筆訂單記錄，其中包含206,209位顧客。而資料集群的分佈如下圖。



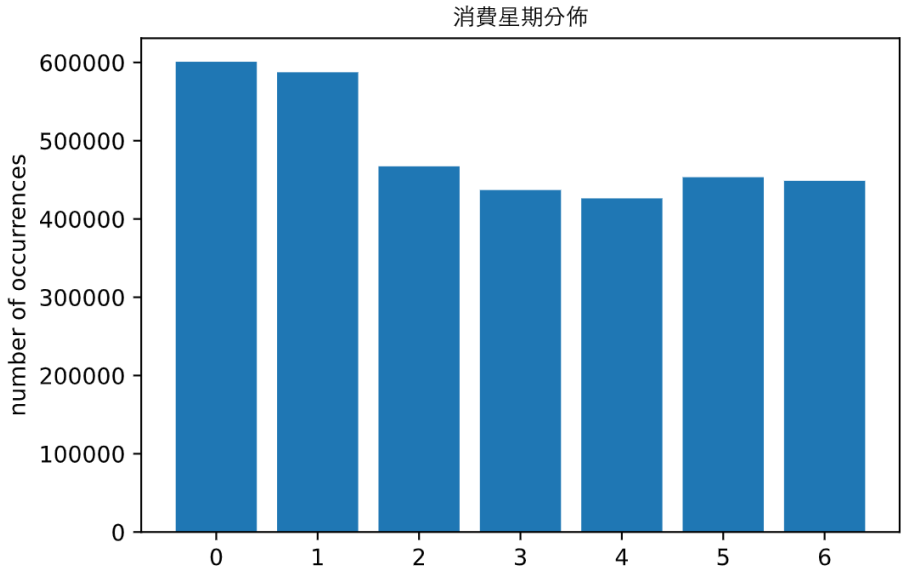
數據分析

- 接著我們查看每一筆訂單的購買商品數量分布。
- 我們可以從圖中觀察出，最低購買次數為4筆，最高則為100筆，而擁有100筆訂單記錄特別多，推測應該是Instacart官方在數據集中選取了不少含有100筆以上訂單的顧客。



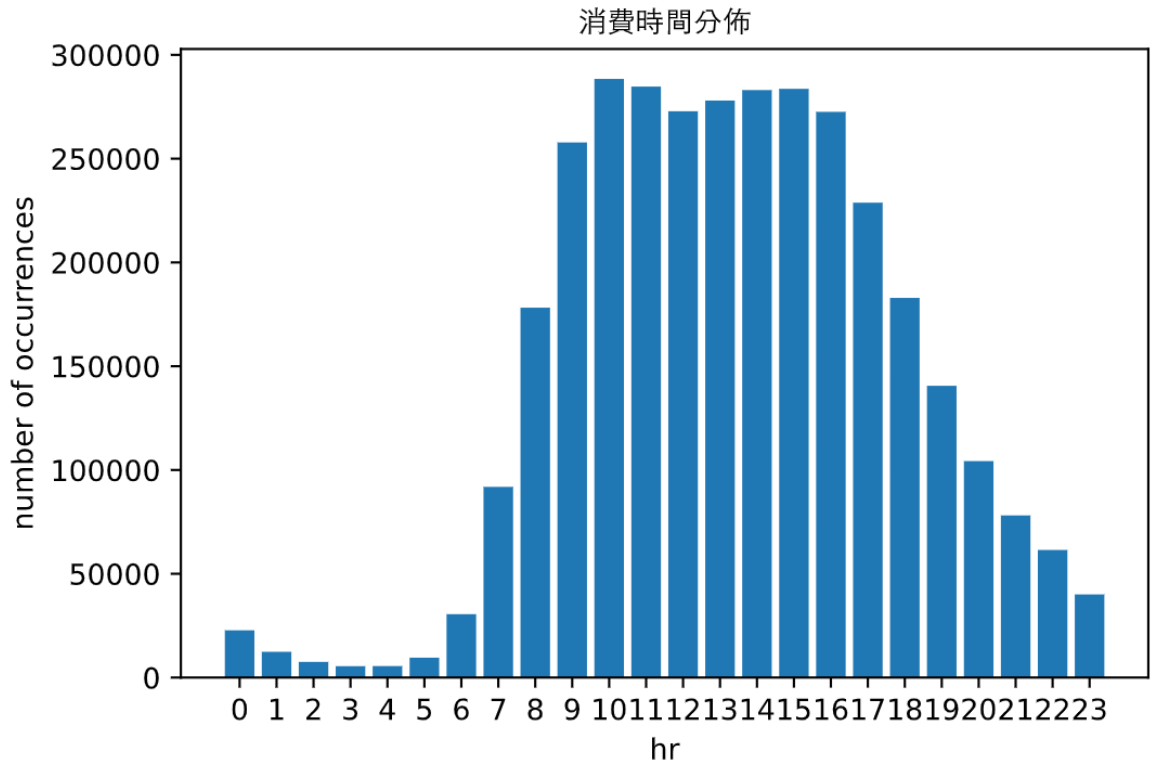
數據分析

- 接著分析顧客的消費時間習慣。官方數據的說明中，並未提供 **order_row** 欄位中0~6所代表的星期數，所以依據常理推測，大多數顧客的消費行為多集中在週末，因此我們可將0視為星期六，1視為星期日，剩下則以此類推。
- 由此得知禮拜六產生最多消費行為，而禮拜三則產生最少消費行為。



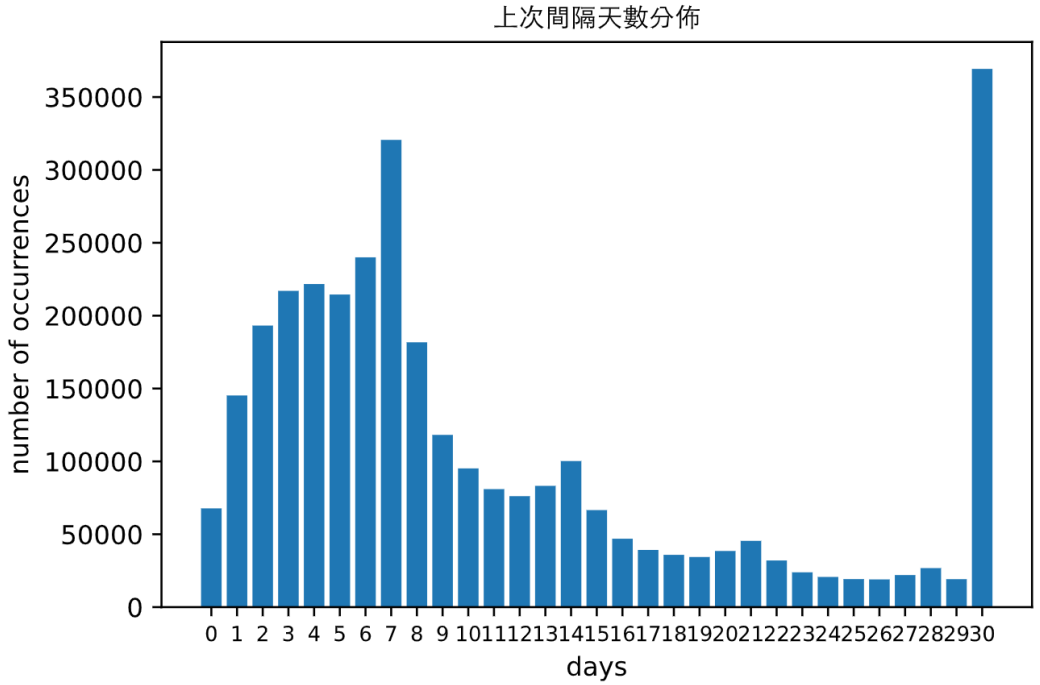
數據分析

- 而在一天之間，消費行為多數集中在早上10點開始至下午4點，而其中又以12點左右的消費記錄較少，推測應為中午吃飯時間，因此較少人進行消費。



數據分析

- 接著分析消費頻率。7天是出現最多次的消費頻率，和我們一般生活認知的消費模式相近，多數人會在數天至一個禮拜採購一次民生用品。而其中30天的消費頻率出現次數異常的高，推測一樣為官方將距離上次消費已經超過30天以上的數據一律合併為30天。



資料預處理

定義問題

- 使用XGBoost（eXtreme Gradient Boosting）算法進行建模。XGBoost是目前Kaggle競賽中最常見到的算法，同時也是多數得獎者所使用的模型。
- 我們要從prior訂單中的欄位（參數X）建立指標，並以這些指標建模來預測 reordered 欄位（變數Y）。

資料預處理——商品

1. 將aisles, department合併至 products
2. 將orders及order_prior做inner join，合併為 orders_products
3. 利用 orders_products 建立表格 prd，包含每一項商品的：
    1) 被重複購買的次數 (prod_reorder)
    2) 被購買的次數 (prod_order)
    3) 被第一次購買的次數 (prod_1st_orders)
    4) 被第二次購買的次數 (prod_2nd_orders)
4. 利用3.來產生以下欄位：
    1) 被第二度購買機率 (reorder_prob):
    prod_2nd_orders/prod_1st_orders
    2) reorder_times: prod_reorder/prod_1st_orders +1
    3) 被重複購買比例 (reorder_ratio): prod_reorder/prod_orders

資料預處理——顧客

1. 篩選資料集標籤為prior的訂單，建立表格 users，包含每一位顧客的：
    1) 訂單數量 (user_orders)
    2) 距離上一次購買天數的總和 (period)
    3) 距離上一次購買天數的平均 (mean_days_prior)
2. 建立表格 us，包含每一位使用者的：
    1) 購買的產品數量總和 (user_total_products)
    2) 重複購買率 (user_reorder_ratio)
    3) 購買的產品種類總和 (user_distinct_products)

資料預處理——顧客

3. 將 us 對 users做inner join，並新增欄位：
    1) 平均單次購買的產品數量 (user_avg_basket)：user_total_products/user_orders
4. 篩選表格 order 中 eval_set 欄位非 prior 的數據，並選取以下欄位：
    1) user_id
    2) order_id
    3) eval_set
    4) days_since_prior_order
5. 對 users 做 inner join

資料預處理——顧客及其購買的商品

1. 利用 orders_products建立表格 dt，包含每一位顧客對每一項商品的：
    1) 購買該商品的次數 (up_orders)
    2) 第一次購買該商品的訂單次序 (up_1st_order)
    3) 最後一次購買該商品的訂單次序 (up_last_order)
    4) 平均將該商品放入購物車的順序 (up_avg_cart_pos)
2. 將表格 dt 及表格 prd 做inner join
3. 將表格 dt 及表格 users 做inner join

資料預處理——顧客及其購買的商品

4. 在表格 dt 新增以下欄位，包含每一位顧客對每一項商品的：
   1) 平均購買該商品的次數 (up_order_rate): up_orders/user_orders
   2) 連續沒有購買該商品的次數 (up_orders_since_last)
   3) up_order_rate_since_1st: up_orders/(user_orders – up_1st_order+1)
5. 將表格 dt 和 order_train 中的以下欄位做inner joint:
   1) user_id
   2) product_id
   3) reordered

建模與預測

建模與預測

- 從表格 dt 中根據eval_set欄位，建立訓練集，最後共採用19個因子進行建模：

1. 針對所有商品：prod_orders、reorder_prob、reorder_times、reorder_ratio
2. 針對所有顧客：user_orders、period、mean_days_prior、user_total_products、user_reorder_ratio、user_distinct_products、user_avg_basket、days_since_prior_order
3. 針對顧客及其購買的商品：up_orders、up_1st_order、up_last_order、up_avg_cart_pos- up_order_rate、up_orders_since_last、up_order_rate_since_1st

建模

- 從訓練集中挑選10%作為子訓練集，採用Binary Logistics Regression建模，訓練次數為80次，訓練參數如下：

1. objective: binary:logistic
2. eval_metric: logloss
3. eta: 0.1
4. max_depth: 6
5. min_child_weight: 10
6. gamma: 0.7
7. subsample: 0.76
8. colsample_bytree: 0.95
9. alpha: 0.05
10. lambda: 10

預測

- 將建立好的模型，對測試集預測reordered欄位的機率，若reordered機率大於0.21的產品則視為會再度購買，其餘則視為不會再度購買。
- 從submission中找出沒有在測試集裡的user_id，並視為沒有任何再度購買的商品，以”None”填補，最後便可上傳數據。