# **Consumer Purchase Behaviour Analysis Report**

**Problem Statement**

Predict the probability of a customer buying a certain product on the 50th week based on the historical purchase records of past 49 weeks.

**Initial Features in the given dataset:**

* i -> Customer ID
* j -> Product ID
* t -> Week Number
* price -> Price of one unit of a product in a certain week
* advertised -> Indicator to denote if a certain product was advertised or not in a certain week

**Initial Observations on the given dataset:**

* Each product had a fixed price on all the weeks that they were not advertised.
* Different prices were found for a product on the advertised weeks which suggests that different discounts were given on the same product on different weeks.

**Top 10 Customers purchase frequency data and plot:**



**Top 10 Products sold over 49 weeks:**

****

**Training Dataset Preparation:**

Since the data given in the input file only had the information of the product that were purchased by the customers over 49 weeks, data augmentation was done to include that product information which were not purchased by anyone over the period of 49 weeks. The augmented data was based on the assumption that there were fixed number of customers and products of values 2000 and 40 respectively as mentioned in the question. This data augmentation helped in generating the labels for the dataset where the data that was provided was labelled as 1 (purchased) and the data augmented was labelled as 0 (not purchased). The prices of the product in augmented data was selected from the given purchase data when the product was not advertised.

**Training Dataset Class Label Distribution:**

|  |  |
| --- | --- |
| **Labels** | **Row Count** |
| 0 | 3843498 |
| 1 | 76502 |

**Feature Engineering on training dataset:**

Since the data didn’t explicitly include any feature about a customer product preference, so three new features were introduced to include the customer product preferences.

***Feature 1 - prod\_adv\_prob***

This feature states the probability of a customer buying a product based on whether its advertised or not. The value was calculated on the following formula

P (customer i choosing product j based on advertised) =

count of choosing a product j by customer i if advertised or not

count of customer i choosing a product j

***Feature 2 - overall\_purchase\_prob***

This feature states the probability of a customer buying a certain product at all. The value was calculated on the following formula

P (customer i choosing product j) = count of choosing a product j by customer i

count of products purchased by customer i

***Feature 3 - adv\_prices***

This feature was derived from the product price and the advertised flag by multiplying them. This separates out the product prices which were only advertised in order state the preference weight for the advertised prices.

**Hence, the total number of features in the final training set becomes 8.**

**The correlation matrix of the training features with the labels is given below:**

****

**Test Dataset Preparation**

The testing dataset size is also fixed since we are trying to predict the probability of purchase of 2000 customers and 40 products of the 50th week. So, the dataset size is 2000 x 40 = 80000. The price of the 40 items is taken to be the fixed price from the input historical dataset and the price of the items which are advertised for the 50th week is computed based on the discount rate provided in the **promotion\_schedule.csv** file. Also, the advertised flag from the same file is included in the test dataset as a feature as well.

*N.B: There was one row missing for the product id 20 in the original* ***promotion\_schedule.csv*** *file. I have included that row with the assumption that it was not advertised on the 50th week.*

The derived features ***prod\_adv\_prob*** *and* ***overall\_purchase\_prob*** are taken from the historical data and the 3rd feature ***adv\_prices*** is calculated by multiplying the price and the advertised flag in the test dataset.

**Model Training and prediction:**

Even though this is a binary classification problem but the prediction to be done is the value of the probability of generating a label 1 (purchase) for each of the product for each customer. Logistic Regression has been used especially for this case since the goal of logistic regression using maximum likelihood estimation is to provide optimum estimates of Prob(Y=1|X). This model is extremely robust when classes are linearly separable which is the case. Logistic regression can also be regularized by penalizing coefficients with a tunable penalty strength. The hyper-parameter C which the inverse of the regularization strength has been tuned in order to avoid overfitting.

**Results:**

The predicted probabilities for all the combinations of 2000 customers and 40 products have been generated and saved in the **predictions.csv** file. Since there is no true labels present for the test dataset, the actual AUC score cannot be computed. However, I did a 100-Fold cross-validation to get the AUC score on the training set which is around 0.7.

So, it can be assumed that the test dataset will have the upper bound of AUC score as 0.7.

**Expected AUC Score <= 0.7**