***INSTACART ORDERS***

## **What is Instacart?**

**Introduction**

Instacart is an online grocery delivery service company that works with local stores to deliver groceries to your door. Through it, customers can select groceries online through a web application and within a couple of hours(or a timeframe of one’s choice ), the order gets delivered to the personal shopper.

**Data Collection source:**

Instacart Orders data has been taken from Kaggle. There were five datasets taken from the Kaggle website.

**Data Dictionary :**

**1. Orders dataset:**

#**order\_id**: order identifier

#**user\_id**: customer identifier

#**eval\_set**: which evaluation set this order belongs in (see SET described below)

#**order\_number**: the order sequence number for this user (1 = first, n = nth)

#**order\_dow**: the day of the week the order was placed on

#**order\_hour\_of\_day**: the hour of the day the order was placed on

#**days\_since\_prior**: days since the last order, capped at 30 (with NAs for order\_number = 1)

This is my understanding of the dataset structure:

* users are identified by user\_id in the orders csv file. Each row of the orders csv fil represents an order made by a user. Order are identified by order\_id;
* Each order of a user is characterized by an order\_number which specifies when it has been made with respect to the others of the same user;
* each order consists of a set of product each characterized by an add\_to\_cart\_order feature representing the sequence in which they have been added to the cart in that order;

**2. Products dataset :**

#**product\_id**: product identifier

#**product\_name**: name of the product

#**aisle\_id**: foreign key

#**department\_id**: foreign key

**3. Aisles dataset :**

#**aisle\_id**: aisle identifier

#**aisle**: the name of the aisle

**4. Departments dataset :**

#**department\_id**: department identifier

#**department**: the name of the department

**5**. **order\_products\_\_SET** :

#**order\_id**: foreign key

#**product\_id**: foreign key

#**add\_to\_cart\_order**: order in which each product was added to cart

#**reordered**: 1 if this product has been ordered by this user in the past, 0 otherwise

#where SET is one of the four following evaluation sets (eval\_set in orders):

#"**prior**": orders prior to that users most recent order

#"***train***": training data supplied to participants

#"**test**": test data reserved for machine learning competitions

Reading the datasets:

**import pandas as pd**

**aisles** = pd.read\_csv("../Capstone\_project/Instacart\_orders/aisles.csv")

**departments** = pd.read\_csv("../Capstone\_project/Instacart\_orders/departments.csv")

**prod\_p** = pd.read\_csv("../Capstone\_project/Instacart\_orders/order\_products\_\_prior.csv")

**prod\_tr** = pd.read\_csv("../Capstone\_project/Instacart\_orders/order\_products\_\_train.csv")

**products** = pd.read\_csv("../Capstone\_project/Instacart\_orders/products.csv")

**orders** = pd.read\_csv("../Capstone\_project/Instacart\_orders/orders.csv")

**Data Interpretation**

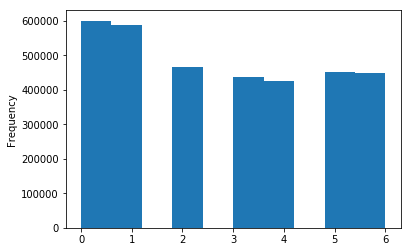
### **When do people order?**

Hour-of-Day

There is a clear effect of hour of day on order . Most orders are between 7.00-16.00 .

**Code:**

orders['order\_hour\_of\_day'].plot(kind = 'hist',figsize=(15,15),xticks =[i for i in range(0,24)])



Day-of-Week

There is clear effect of day\_of\_week, on the

orders. Most of the products have been ordered

on days 0 and 1, but as we don’t have information

of which value represents which day, we may assume

that it is weekend.

**Code:**

orders['order\_dow'].plot(kind = 'hist')

### 

### **When do they order again?**

People seem to order more often after about 9

days or approximately a week.

**Code:**

orders['days\_since\_prior\_order'].plot(kind = 'hist',figsize=(8,9),xticks =[i for i in range(0,30)])

### 

### 

### 

### 

### **How many prior orders are there?**

We can see that there are always at least 1 prior orders.

**Code:**

orders['order\_number'].plot(kind = 'hist',figsize=(10,10),xticks =[i for i in range(0,100,2)])

### **How many items do people buy?**

We can see that people most often order around 5 to 7 items from the

training set.

**Code:**

prod\_tr['add\_to\_cart\_order'].plot(kind='hist',xticks =[i for i in range(0,80,5)])



We can see that people most often order around 10 to 15 items from the

prior orders set.

**Code:**

prod\_p['add\_to\_cart\_order'].plot(kind='hist',xticks =[i for i in range(0,60,5)])

### **Top Selling Products**

Fruits like banana , strawberries, organic

spinach....are the most ordered products.

**Code:**

prod\_dep\_ais.groupby(['product\_name']).count()['order\_id'].sort\_values(ascending=False)[:10].plot(kind='bar')

**How often do people order same items again?**

60% of the orders have been reordered.

Code:

grouped = prod\_dep\_ais.groupby("reordered")["product\_id"].aggregate({'Total\_products': 'count'}).reset\_index()

grouped['Ratios'] = grouped["Total\_products"].apply(lambda x: x /grouped['Total\_products'].sum())

### 

### 

### 

### 

### 

### 

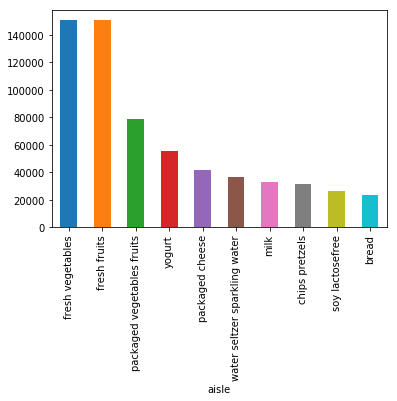
### **Top Selling Product Departments**

Most of the orders have been made from

produce department.

### 

### 

**Aisle Information**

Most of the orders have taken place from aisles-

Fresh vegetables and fresh fruits, so these two are topmost

Aisles.

Conclusion:

No overlap between users from train and test set

**test**: 75000 users, 1242497 orders, 16.566626666666668 orders per user

**train**: 131209 users, 2178586 orders, 16.60393722991563 orders per user

**Link to presentation slides:**

https://1drv.ms/p/s!AnyF9\_dv0S8xgW7aDGCJ8KWwlbrE