Justin Kim

CareerFoundry
Data Immersion Portfolio
Instacart Analysis

Instacart Data Analysis

Context

Instacart is a grocery delivery and pick-up service that allows users to order groceries online from local stores.

Purpose

The project's purpose is to conduct an initial analysis of the data, exploring key insights and proposing strategies to enhance segmentation according to the specified criteria.

My Role

My role is a Data Analyst tasked with presenting stakeholders a strategy that will connect Instacart customers with the appropriate products to improve sales.

Tools Used: X jupyter

Key Questions from Instacart Stakeholders

01

The sales team needs insights on peak order days and hours to schedule ads during low-activity periods.

02

Are there peak spending times during the day that could impact product advertising choices?

03

How can the sales and marketing team regroup the price ranges?

04

Are there certain types of products that are more popular than others?

05

What is the ordering behavior among the wide variety of Instacart customers?

Data

“The Instacart Online Grocery Shopping Dataset 2017”, Accessed from www.instacart.com/datasets/grocery-shopping-2017 via Kaggle on [11/1/2023].

*The customers data and prices columns is fabricated for the sake of this assignment

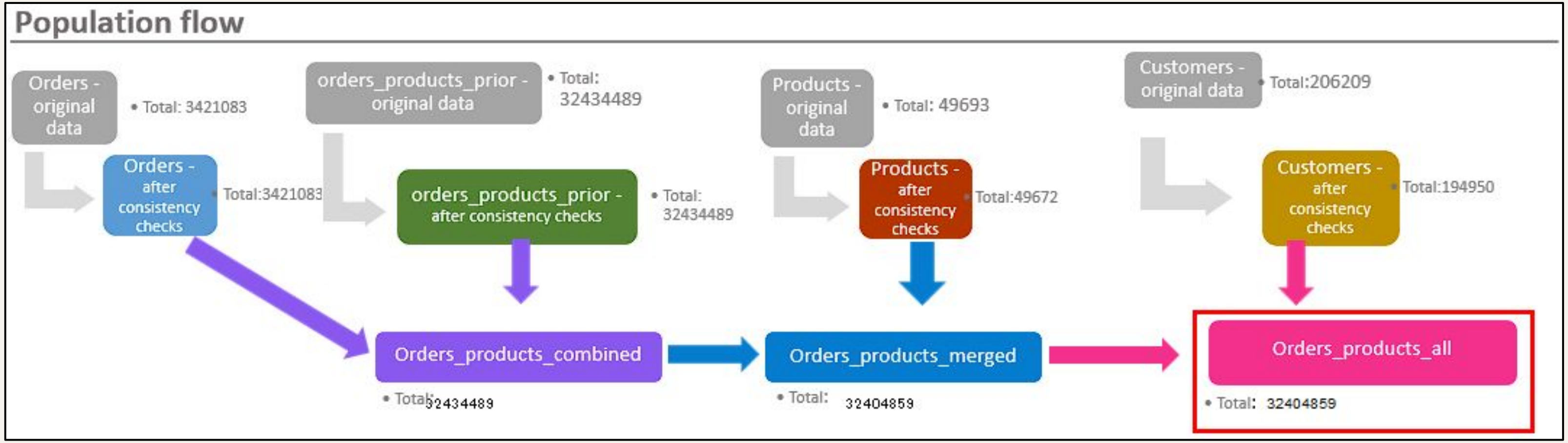
Data Dictionary

Skills

- Importing Libraries
- Importing and Exporting datasets
- Descriptive Analysis
- Data Wrangling
- Data Merging
- Deriving New Variables
- Aggregating Data
- Visualizations with Python Libraries
- Reporting with Excel

Cleaning and Combining Data Sets

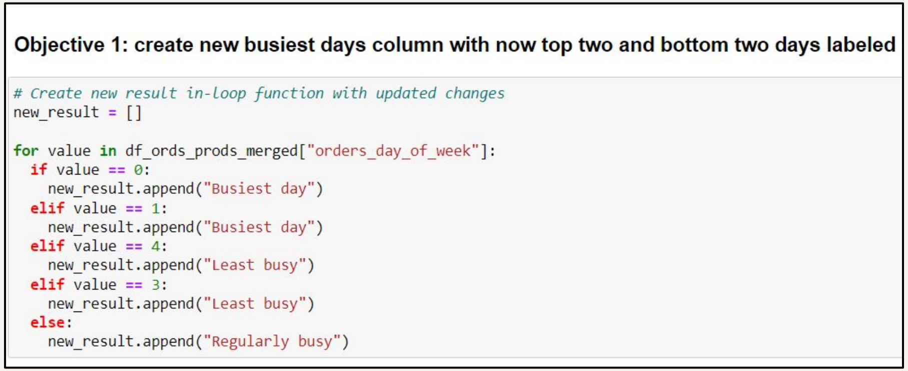
1. Clean the “Orders” dataset
2. Clean the “Products” dataset
3. Clean the “Customers” dataset
4. Merge everything together



Doing this part of the process allows minimal errors and produces the most accurate analysis to answer stakeholder questions.

Deriving New Variables

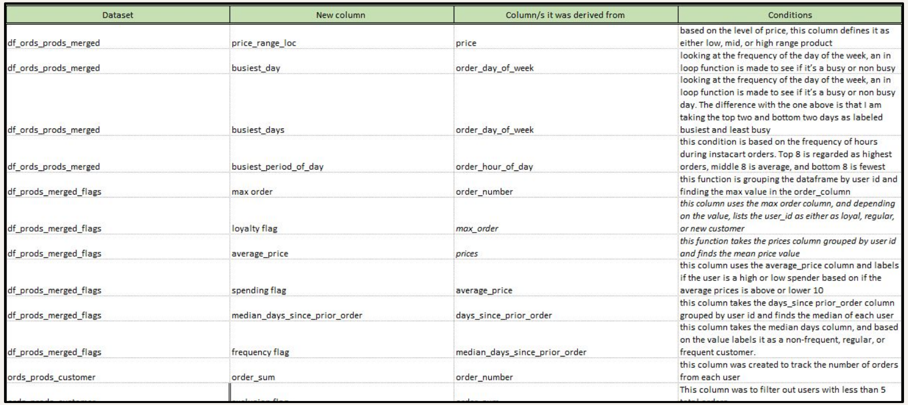
Employing merged columns within Jupyter Notebook, I crafted commands to generate new columns based on existing data, delineating key customer demographic details including age group, income level, region, and loyalty status.



Objective 1: create new busiest days column with now top two and bottom two days labeled

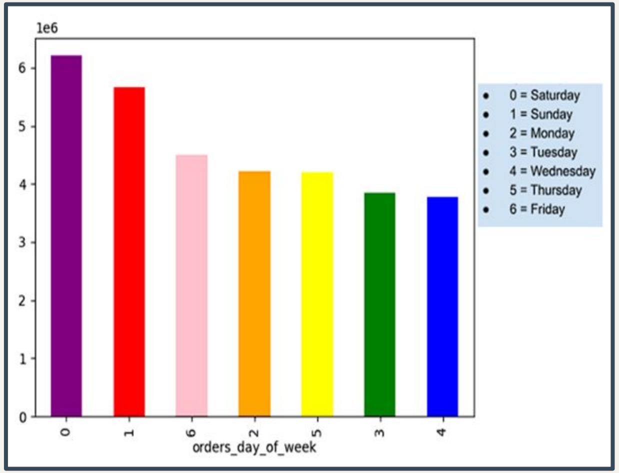
```
# Create new result in-loop function with updated changes
new_result = []

for value in df_ords_prods_merged["orders_day_of_week"]:
  if value == 0:
    new_result.append("Busiest day")
  elif value == 1:
    new_result.append("Busiest day")
  elif value == 4:
    new_result.append("Least busy")
  elif value == 3:
    new_result.append("Least busy")
  else:
    new_result.append("Regularly busy")
```

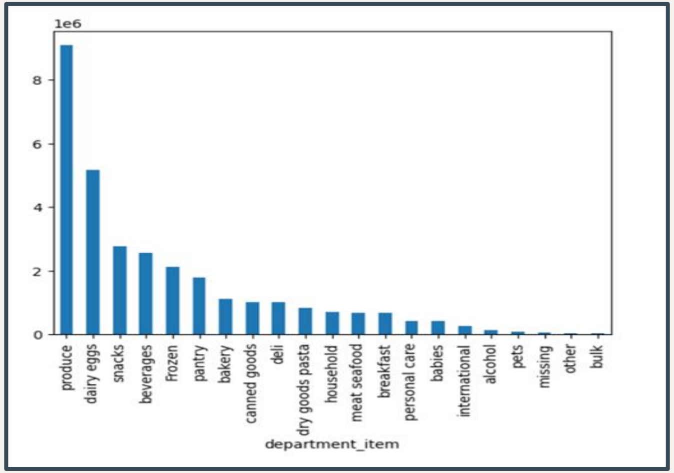


Dataset	New column	Column/s it was derived from	Conditions
df_ords_prods_merged	price_range_loc	price	based on the level of price, this column defines it as either low, mid, or high range product
df_ords_prods_merged	busiest_day	order_day_of_week	looking at the frequency of the day of the week, an in loop function is made to see if it's a busy or non busy
df_ords_prods_merged	busiest_days	order_day_of_week	looking at the frequency of the day of the week, an in loop function is made to see if it's a busy or non busy day. The difference with the one above is that I am taking the top two and bottom two days as labeled busiest and least busy
df_ords_prods_merged	busiest_period_of_day	order_hour_of_day	this condition is based on the frequency of hours during instacart orders. Top 8 is regarded as highest orders, middle 8 is average, and bottom 8 is fewest
df_prods_merged_flags	max order	order_number	this function is grouping the dataframe by user id and finding the max value in the order_column
df_prods_merged_flags	loyalty flag	*max_order*	*this column uses the max order column, and depending on the value, lists the user_id as either as loyal, regular, or new customer*
df_prods_merged_flags	average_price	*prices*	*this function takes the prices column grouped by user id and finds the mean price value*
df_prods_merged_flags	spending flag	average_price	this column uses the average_price column and labels if the user is a high or low spender based on if the average prices is above or lower 10
df_prods_merged_flags	median_days_since_prior_order	days_since_prior_order	this column takes the days_since prior_order column grouped by user id and finds the median of each user
df_prods_merged_flags	frequency flag	median_days_since_prior_order	this column takes the median days column, and based on the value labels it as a non-frequent, regular, or frequent customer.
ords_prods_customer	order_sum	order_number	this column was created to track the number of orders from each user
ords_prods_customer	exclusion flag	order_sum	This column was to filter out users with less than 5 total orders

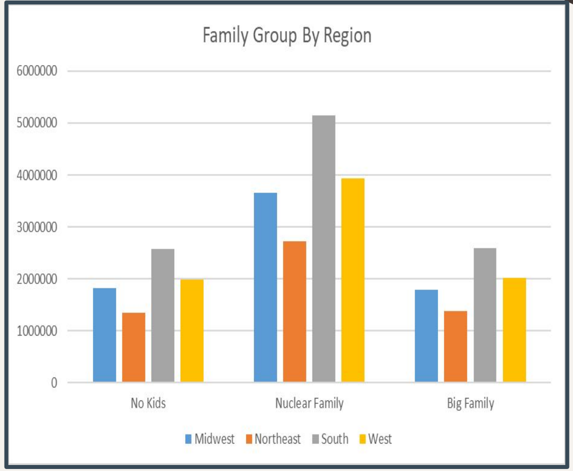
Recommendations



The visual representations above depict the distribution of orders across the week. I recommend that the sales team targets advertisements from Monday to Thursday, leveraging periods of lower order volume to extend customer engagement.



This bar chart visualization compares the top-selling items on Instacart, revealing that produce is the leading item among all customers. An observed consistency in top items suggests a preference for everyday groceries essential for balanced meals. I recommend the implementation of Instacart bundle deals comprising these top-selling items.



This bar chart depicts family situations by region, with nuclear families (two kids) being predominant among Instacart customers, indicating a focus on family meals and school lunches; my recommendation is to implement targeted campaigns featuring items for both parents and children.

Justin Kim

Email: Jjoonhokim8@gmail.com

Phone: (562)-376-0651

LinkedIn

Additional link to access presentation, query log, and data dictionary:

Github Link