Vending Machine Analysis

By: Omar, Saif, Andrew, Ahmed, Aman

TABLE OF CONTENTS

01

Excel Data

02

Power BI Implementation

03

Python Mapping

04

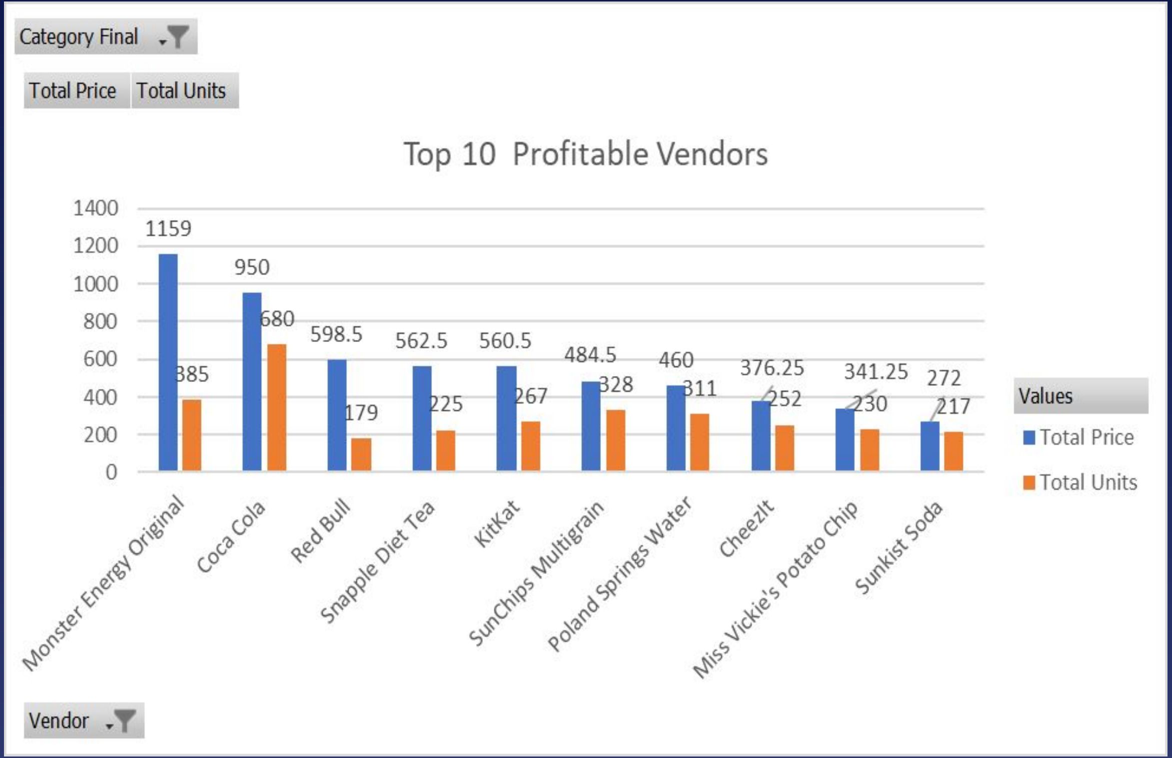
Python Predictive Analysis

Introduction & Problem Statement

1. How many of you have used vending machines this year?
   a. Did you ever wonder how companies know when to restock a vending machine?
2. ABC Vending is a small vending machine company with 5 vending machines across Mississauga
3. They Sell Four Categories of Products
   a. Carbonated Beverages
   b. Food
   c. Non Carbonated Beverages
   d. Water
4. In order to help ABC Vending with their business
   a. Executive Dashboard
   b. Logistic Mapping - Supply Chain
   c. Predictive Modelling Through Python

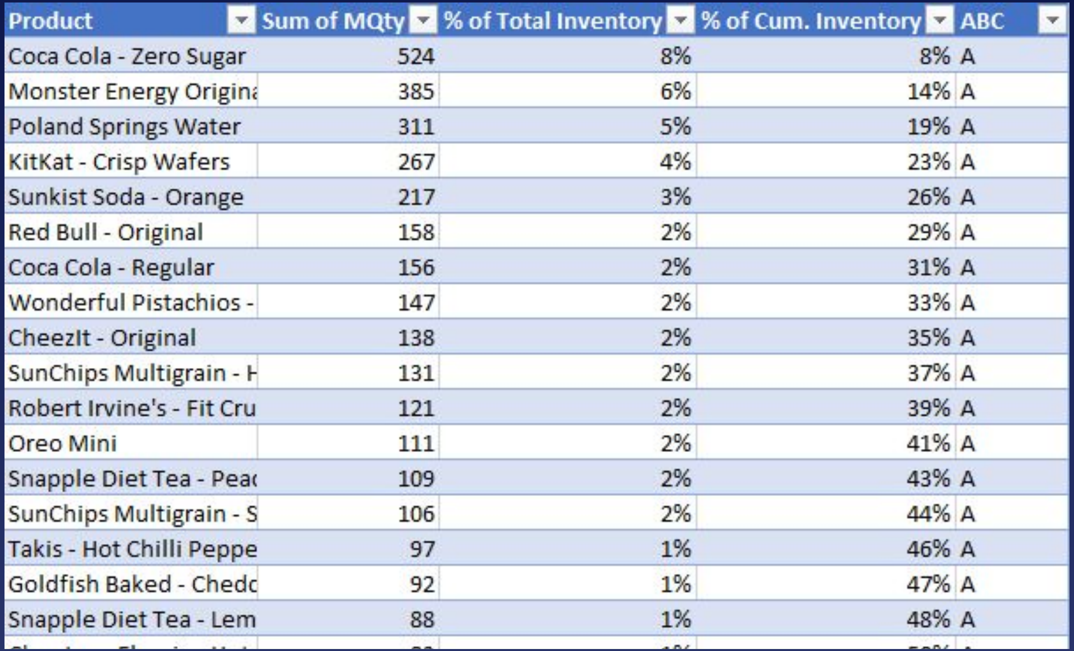
Excel Data Analysis

- The team used excel to slice and dice different data analytics techniques to achieve both exploratory data analysis and predictive modelling



ABC Analysis

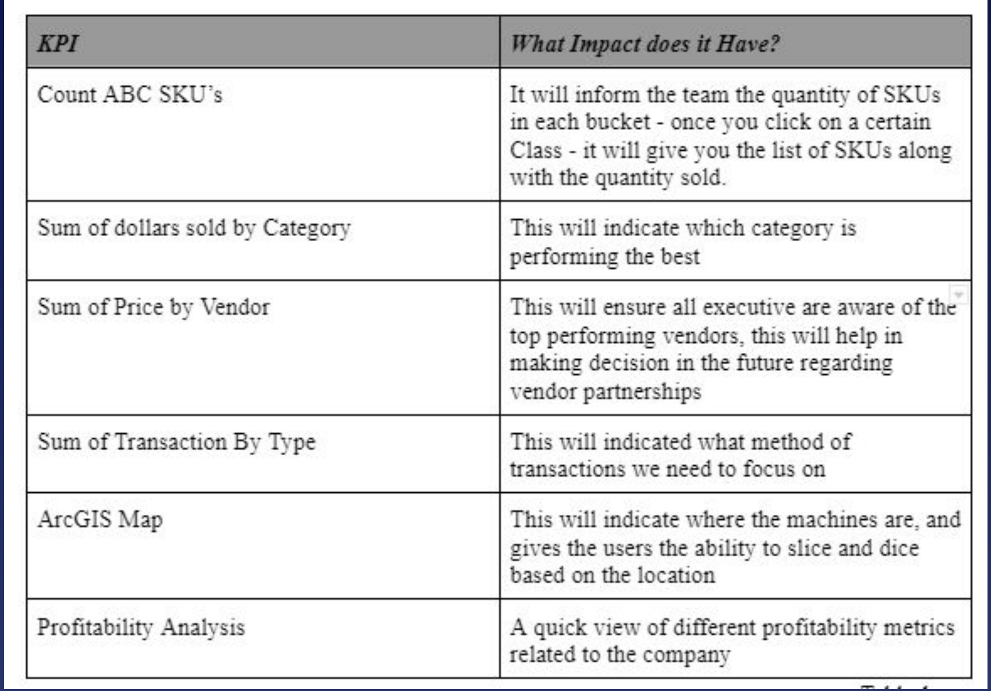
- For predictive modelling we utilized the ABC Analysis to use historical data to predict what SKUs can be categorized based on the % of Cumulative Inventory:



Product	Sum of MQty	% of Total Inventory	% of Cum. Inventory	ABC
Coca Cola - Zero Sugar	524	8%	8%	A
Monster Energy Origina	385	6%	14%	A
Poland Springs Water	311	5%	19%	A
KitKat - Crisp Wafers	267	4%	23%	A
Sunkist Soda - Orange	217	3%	26%	A
Red Bull - Original	158	2%	29%	A
Coca Cola - Regular	156	2%	31%	A
Wonderful Pistachios -	147	2%	33%	A
CheezIt - Original	138	2%	35%	A
SunChips Multigrain - H	131	2%	37%	A
Robert Irvine's - Fit Cru	121	2%	39%	A
Oreo Mini	111	2%	41%	A
Snapple Diet Tea - Peac	109	2%	43%	A
SunChips Multigrain - S	106	2%	44%	A
Takis - Hot Chilli Peppe	97	1%	46%	A
Goldfish Baked - Chedc	92	1%	47%	A
Snapple Diet Tea - Lem	88	1%	48%	A

Excel to PowerBI Transition

- We used both the exploratory data and predictive modelling done in Excel to create various KPIs that would help a Executive staff member make decision regarding the business in a strategic and astute manner.



KPI	*What Impact does it Have?*
Count ABC SKU's	It will inform the team the quantity of SKUs in each bucket - once you click on a certain Class - it will give you the list of SKUs along with the quantity sold.
Sum of dollars sold by Category	This will indicate which category is performing the best
Sum of Price by Vendor	This will ensure all executive are aware of the top performing vendors, this will help in making decision in the future regarding vendor partnerships
Sum of Transaction By Type	This will indicated what method of transactions we need to focus on
ArcGIS Map	This will indicate where the machines are, and gives the users the ability to slice and dice based on the location
Profitability Analysis	A quick view of different profitability metrics related to the company

To Access the Dashboard - Click Here.

Python Programming: Mapping

- Goal is to determine optimal geo-spatial coordinates for vending machines and the optimal positioning for the warehouse facility
- The vending machines were given arbitrary longitudes and latitudes
  - Each location was given a weight to determine the warehouse location
    - Red Marker (Brunswick Sq Mall), Purple Marker (Earle Asphalt), Blue Marker (EB Public Library), Yellow Marker (GuttenPlans), Green Marker (Warehouse)

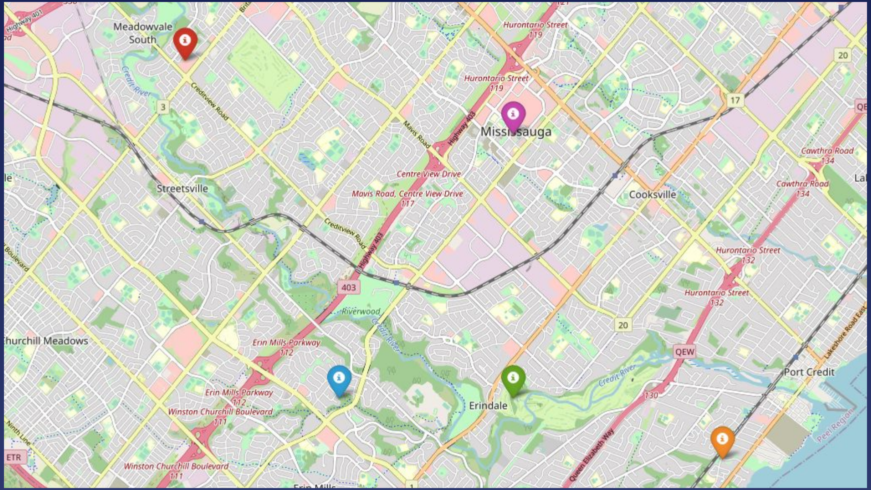


Figure: Vending Machines and Warehouse Mapped Out Using Folium Library

Python Programming: Mapping pt. 2

- Example presented is the optimal route between the Warehouse facility and Earle Asphalt:
  - Distance between two locations found to be 4.35 km or 2.70 miles
  - For drive mode assuming 50 km/h average speed it would take 5.22 minutes

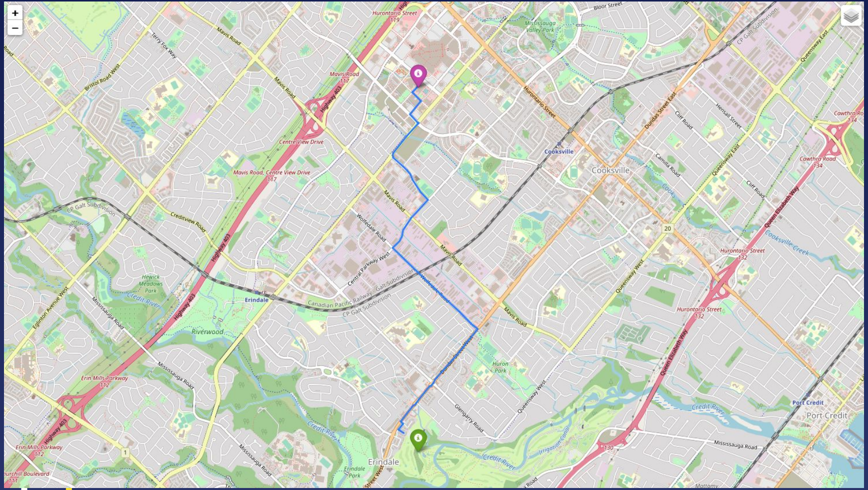


Figure: Optimal Route Using ‘drive’ mode

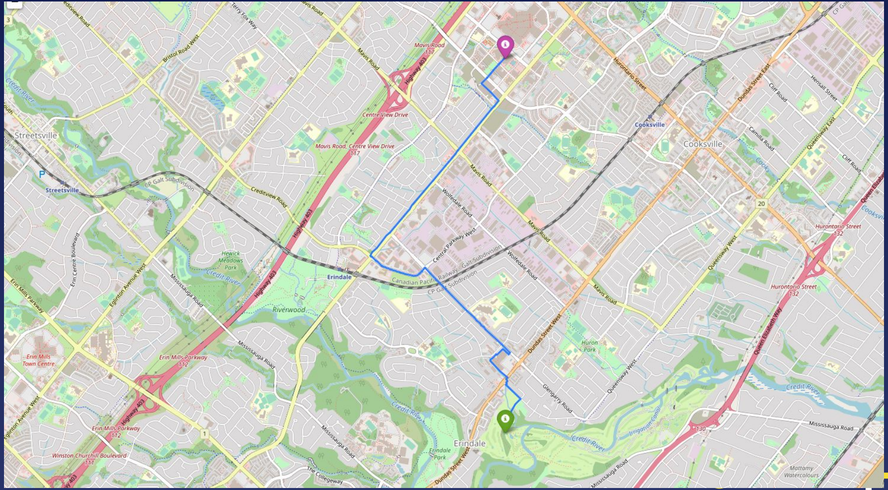
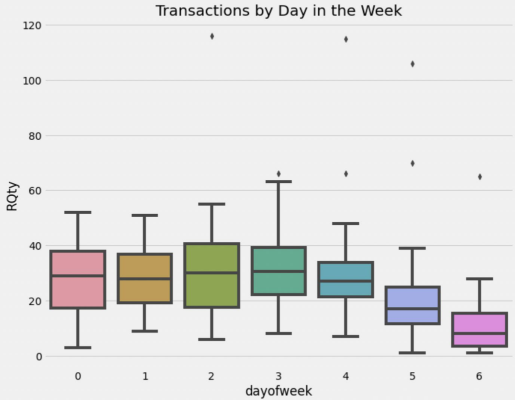
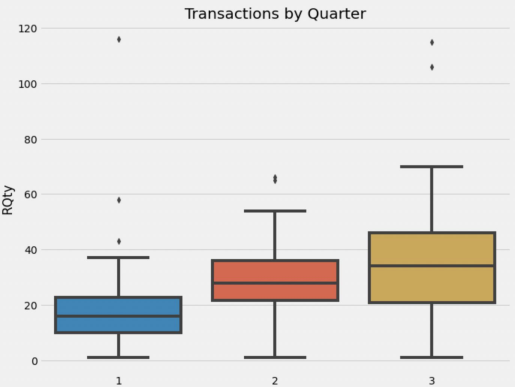
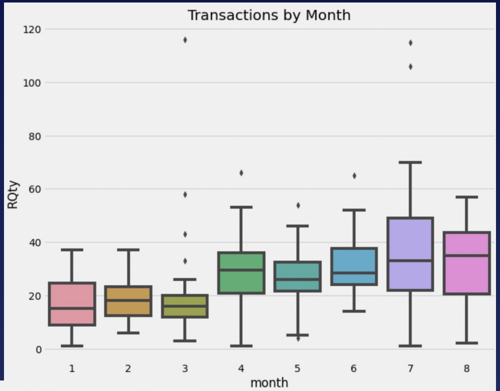


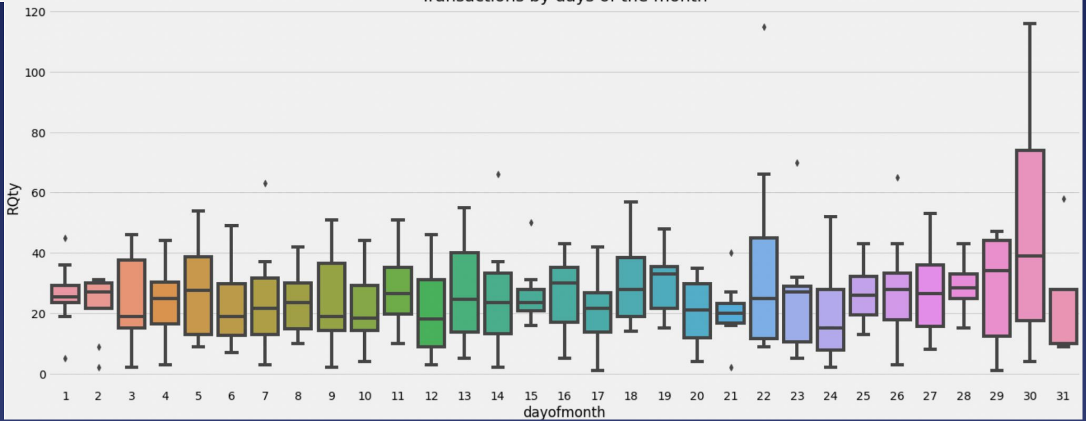
Figure: Optimal Route Using ‘bike’ mode

Python Initial Analysis

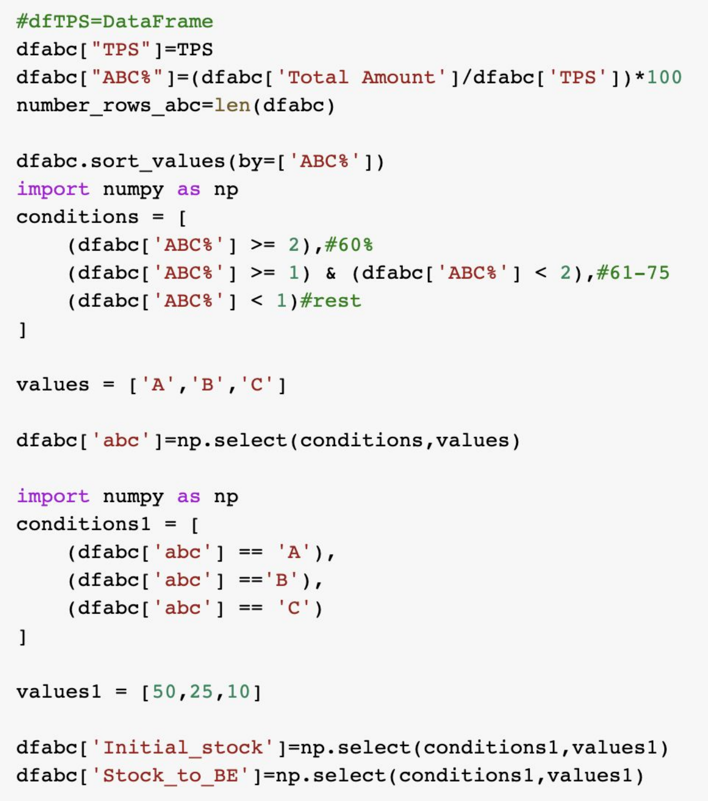




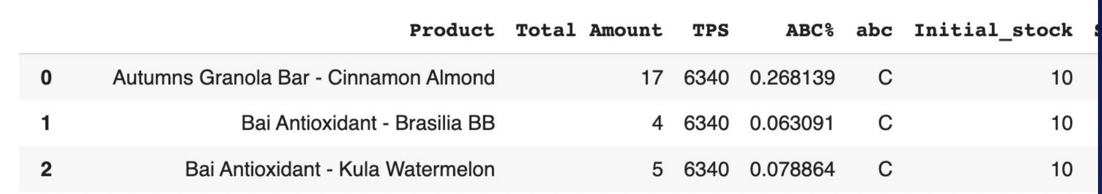




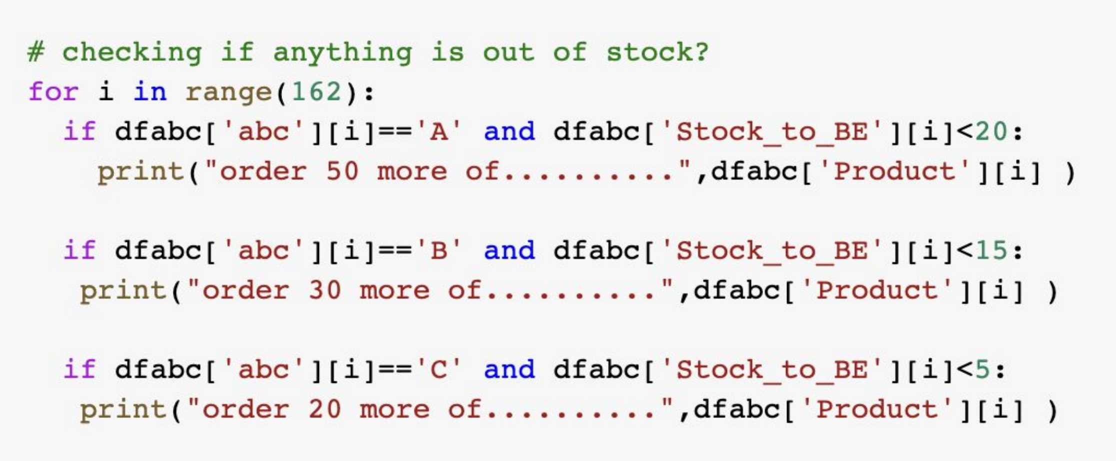
Python Automatic Inventory System



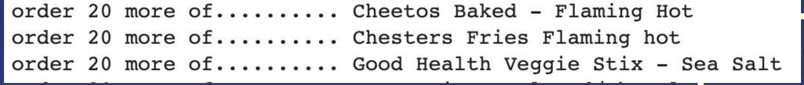
```
#dfTPS=DataFrame
dfabc["TPS"]=TPS
dfabc["ABC%"]=(dfabc['Total Amount']/dfabc['TPS'])*100
number_rows_abc=len(dfabc)

dfabc.sort_values(by=['ABC%'])
import numpy as np
conditions = [
    (dfabc['ABC%'] >= 2),#60%
    (dfabc['ABC%'] >= 1) & (dfabc['ABC%'] < 2),#61-75
    (dfabc['ABC%'] < 1)#rest
]

values = ['A','B','C']

dfabc['abc']=np.select(conditions,values)

import numpy as np
conditions1 = [
    (dfabc['abc'] == 'A'),
    (dfabc['abc'] =='B'),
    (dfabc['abc'] == 'C')
]

values1 = [50,25,10]

dfabc['Initial_stock']=np.select(conditions1,values1)
dfabc['Stock_to_BE']=np.select(conditions1,values1)
```



	Product	Total Amount	TPS	ABC%	abc	Initial_stock
0	Autumns Granola Bar - Cinnamon Almond	17	6340	0.268139	C	10
1	Bai Antioxidant - Brasilia BB	4	6340	0.063091	C	10
2	Bai Antioxidant - Kula Watermelon	5	6340	0.078864	C	10

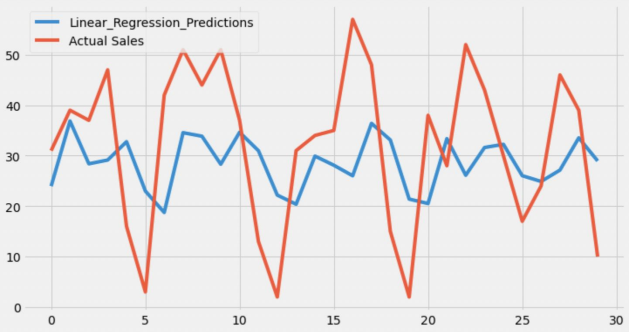


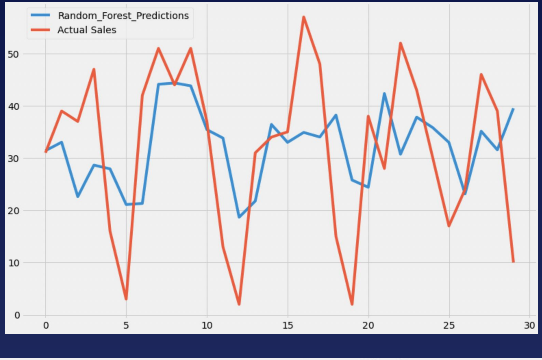
```
# checking if anything is out of stock?
for i in range(162):
  if dfabc['abc'][i]=='A' and dfabc['Stock_to_BE'][i]<20:
    print("order 50 more of..........",dfabc['Product'][i] )

  if dfabc['abc'][i]=='B' and dfabc['Stock_to_BE'][i]<15:
   print("order 30 more of..........",dfabc['Product'][i] )

  if dfabc['abc'][i]=='C' and dfabc['Stock_to_BE'][i]<5:
   print("order 20 more of..........",dfabc['Product'][i] )
```

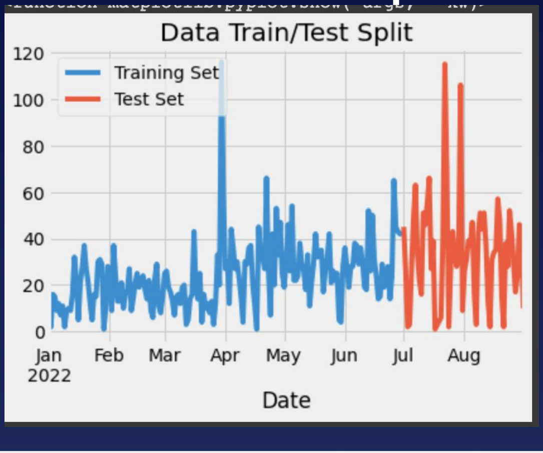


```
order 20 more of.......... Cheetos Baked - Flaming Hot
order 20 more of.......... Chesters Fries Flaming hot
order 20 more of.......... Good Health Veggie Stix - Sea Salt
```

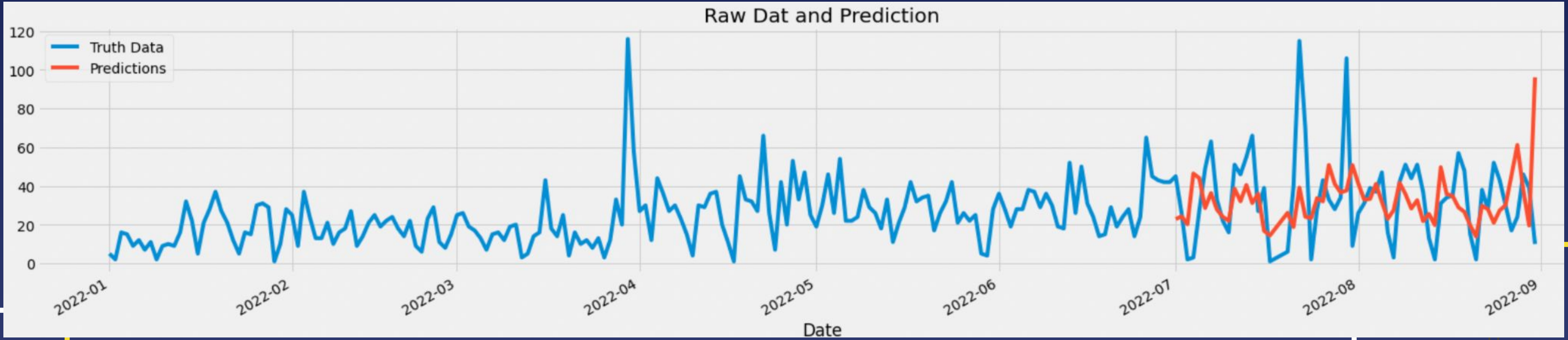
Python Predictive Modelling







```
<function matplotlib.pyplot.show(*args, **kw)>
```



Conclusion

ABC vending machine with 8 months dataset, improvements were achieved through some descriptive and predictive analysis.

Five fronts were maintained including:

- Creating a executive Dashboard.
- Creating Different graphs for monitoring
- Rcoil Analysis (Most sales in that vending machine Row)
- Mapping (Optimal route)
- Machine learning to predict future demand