IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jagath Samantha Godakanda
08-11-2022

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Summary of methodologies
  - Data collection –With Web scraping
  - Data Wrangling
  - Exploratory Data Analysis (EDA) with Data Visualization
  - EDA with SQL
  - Creating Interactive Map with Folium
  - Building Dashboards with Plotly Dash
  - Predictive Analysis using Classification Method
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive Analytics results -Maps and Dashboard
  - Predictive results

Introduction

- Project background and context
  - The main objective of this project is to predict if the Falcon 9 will successfully land on its first stage. Based on SpaceX website, the initial cost to launch Falcon 9 would cost 62 million dollars. On the other hand, other providers estimating around 165 million dollars for each launch. The difference of the cost enable SpaceX to reuse the first stage.

- Problems you want to find answers
  - The task is to find if the Space X Falcon 9 would successfully land on its first stage

Section 1

Methodology

Methodology

Executive Summary

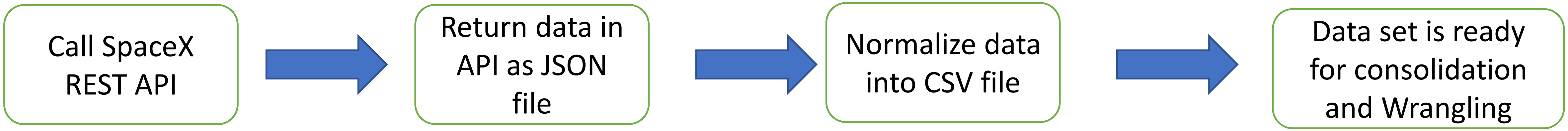
- Data collection methodology:
  - By the SpaceX REST API
  - Web Scrapping from Wikipedia
- Perform data wrangling
  - Machine Learning Methods – Encoding Data Fields
  - Removing null values from the variables
  - Remove unwanted variables from the data set.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

- Data collected from SpaceX REST API and web scraping
- Data in REST API include the data about launches, rocket used, Payload delivered, specification of launching and landings, and Landing outcomes whether it’s a successive or failure
- The Space X REST API URL is api.spacexdata.com/v4/
- Link to the Data gathered from Web Scraping is: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922

Data Collection –Flow Chart

Data Collection by SpaceX REST API



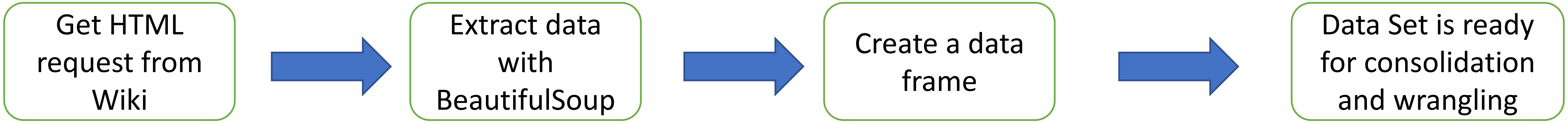
Call SpaceX REST API

Return data in API as JSON file

Normalize data into CSV file

Data set is ready for consolidation and Wrangling

Data Collection by Web Scraping



Get HTML request from Wiki

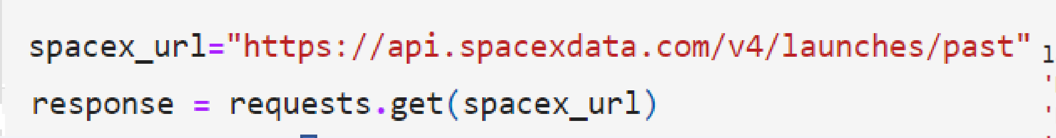
Extract data with BeautifulSoup

Create a data frame

Data Set is ready for consolidation and wrangling

Data Collection – SpaceX REST API

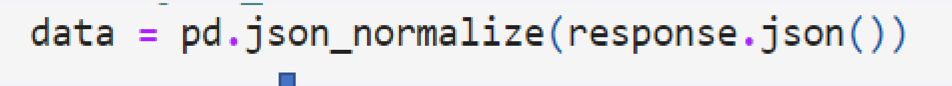
1. Getting Response from API



```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
```

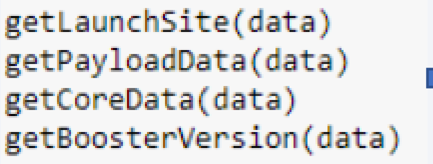
2. Converting Response to a JSON file

response = requests. get (static_json_url).json()



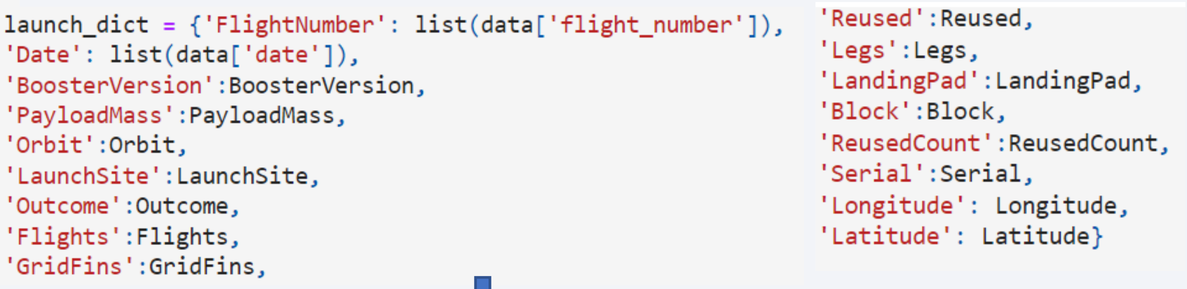
```
data = pd.json_normalize(response.json())
```

3. Data cleaning and transformation



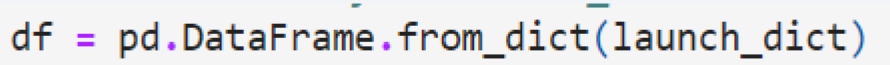
```
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
getBoosterVersion(data)
```

4. Create Data Dictionary



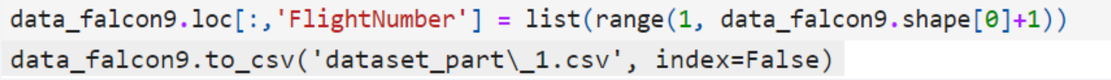
```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

5. Create data frame



```
df = pd.DataFrame.from_dict(launch_dict)
```

6. Filter data frame and export to CSV file

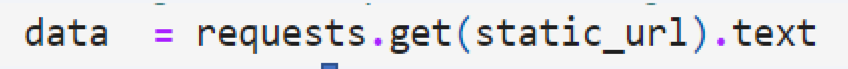


```
data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

Github Link to the code

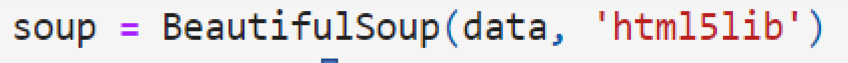
Data Collection – Web Scraping

1. Getting Response from HTML



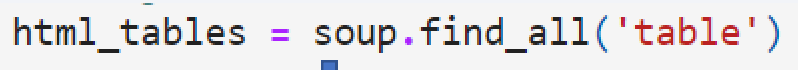
```
data  = requests.get(static_url).text
```

2. Create BeatifulSoup Object



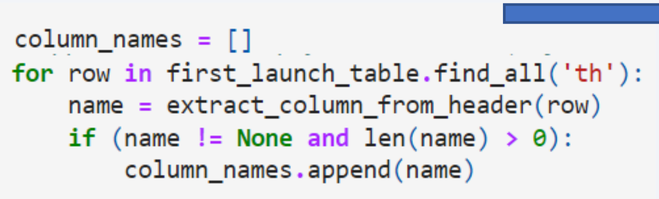
```
soup = BeautifulSoup(data, 'html5lib')
```

3. Find all Tables



```
html_tables = soup.find_all('table')
```

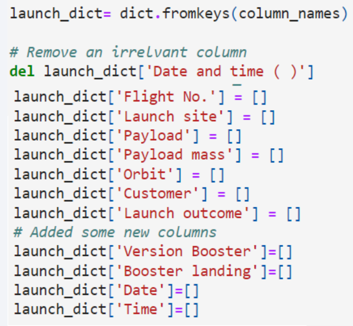
4. Getting column names



```
column_names = []
for row in first_launch_table.find_all('th'):
    name = extract_column_from_header(row)
    if (name != None and len(name) > 0):
        column_names.append(name)
```

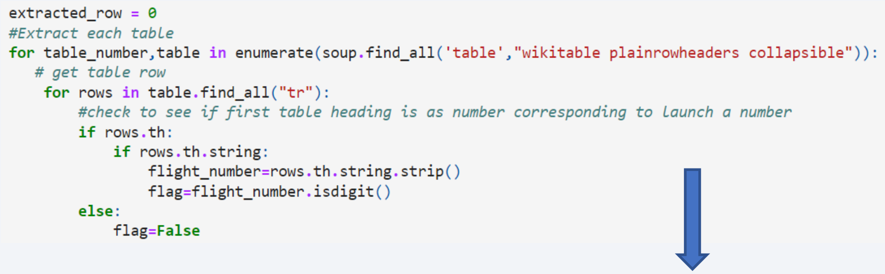
[Github Link to the code](#)

5. Create data dictionary



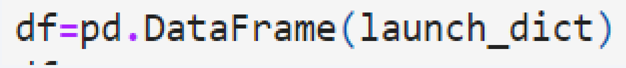
```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

6. Append data to keys (refer the link to the complete code)



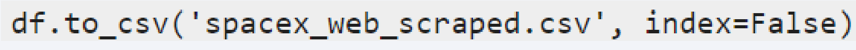
```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
```

7. Convert data dictionary to data frame



```
df=pd.DataFrame(launch_dict)
```

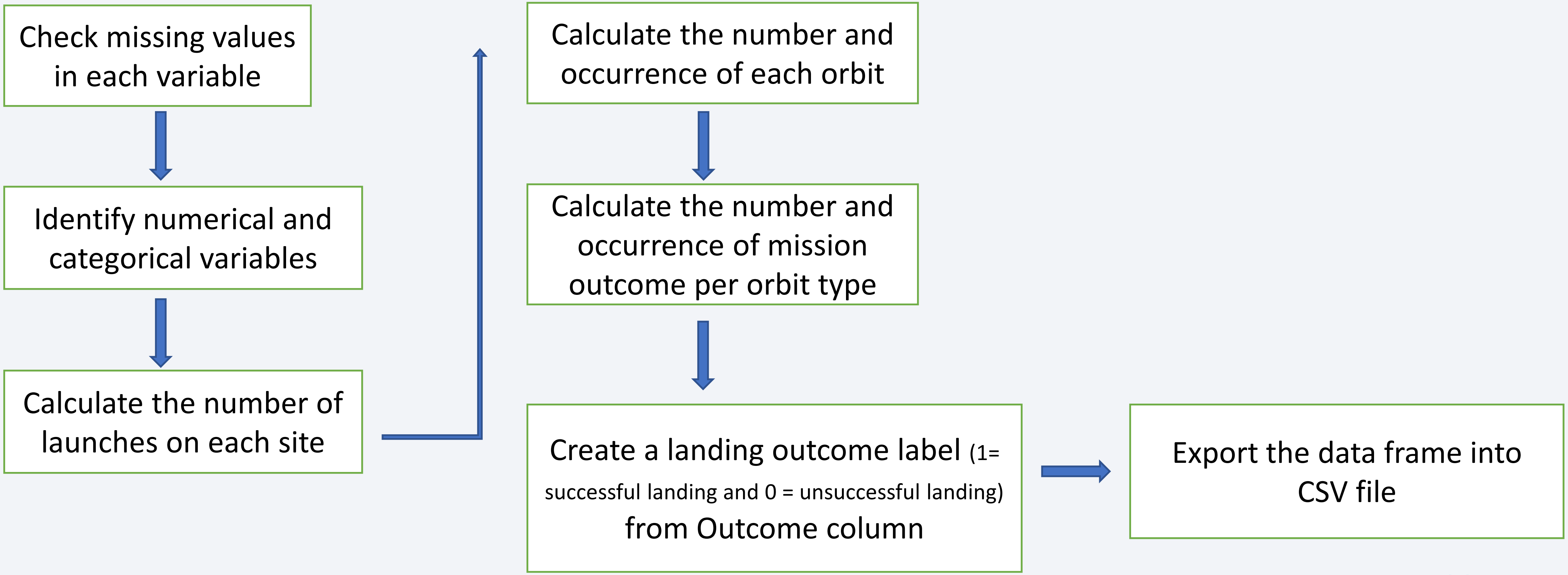
8. Convert data frame to CSV file



```
df.to_csv('spacex_web_scraped.csv', index=False)
```

Data Wrangling

- Perform some Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.



Check missing values in each variable

Identify numerical and categorical variables

Calculate the number of launches on each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per orbit type

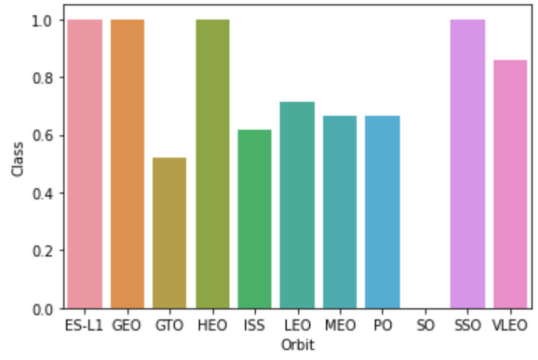
Create a landing outcome label (1= successful landing and 0 = unsuccessful landing) from Outcome column

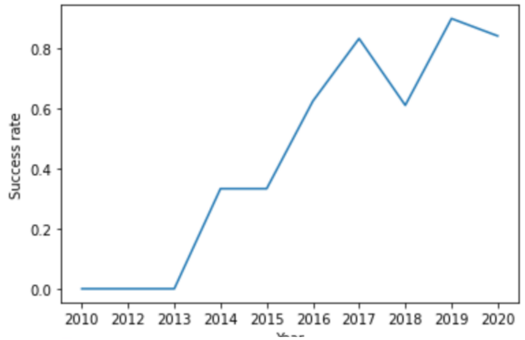
Export the data frame into CSV file

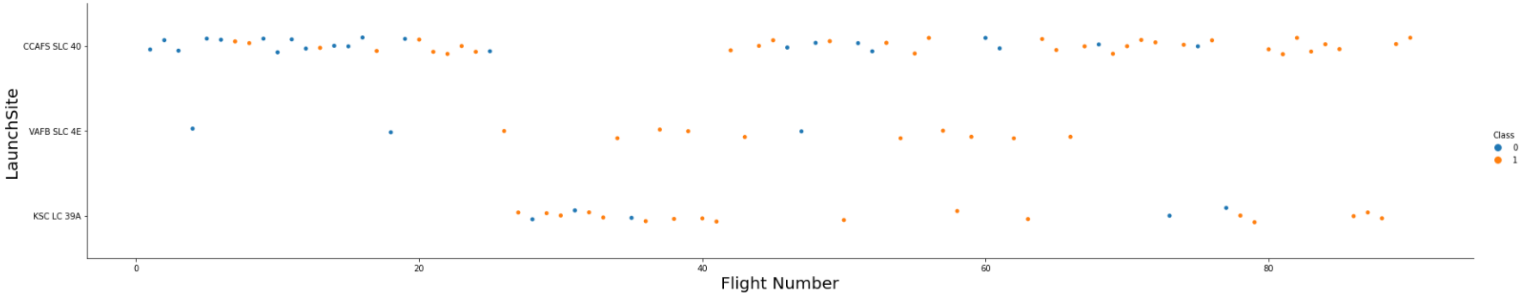
[Github Link to the code]

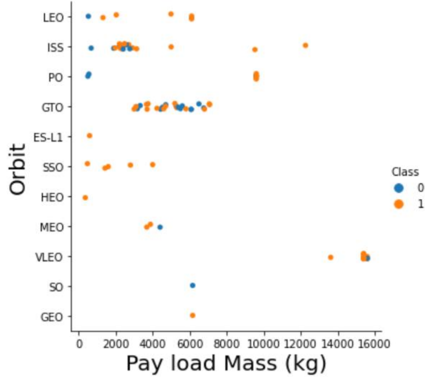
EDA with Data Visualization

- Scatter plot to Visualize the relationship between Flight Number and Pay load Mass (KG)
- Scatter plot to Visualize the relationship between Flight Number and Launch Site
- Scatter plot to Visualize the relationship between Payload and Launch Site
- Bar chart to determine the success rate of each orbit
- Scatter plot to Visualize the relationship between Flight Number and Orbit type
- Scatter plot to Visualize the relationship between Payload and Orbit type
- Line graph to Visualize the launch success yearly trend









[Github Link to the code]()

EDA with SQL

- SQL Queries Performed:
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'CCA’
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first successful landing outcome in ground pad was achieved.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass.
  - List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
  - Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

Github Link to the code

Build an Interactive Map with Folium

- Finding an optimal location for building a launch site certainly involves many factors and hopefully we could discover some of the factors by analyzing the existing launch site locations.
- First, created a folium Map object, with an initial center location to be NASA Johnson Space Center at Houston, Texas
- Created and added folium.Circle and folium.Marker for each launch site on the site map to explore the map by zoom-in/out the marked areas
- Mark the success/failed launches for each site on the map using MarkerCluster object [If a launch was successful (class=1), then we use a green marker and if a launch was failed, we use a red marker (class=0)]
- For each launch result in spacex_df data frame, added a folium.Marker to marker_cluster
- Calculate the distances between a launch site to its proximities by adding MousePosition on the map to get coordinate for a mouse over a point on the map

  After you plot distance lines to the proximities, you can answer the following questions easily:

  Are launch sites in close proximity to railways?

  Are launch sites in close proximity to highways?

  Are launch sites in close proximity to coastline?

  Do launch sites keep certain distance away from cities?

Github Link to the code

Build a Dashboard with Plotly Dash

- SpaceX Launch Records Dashboard has dropdown, pie chart, Payload Mass range slider and scatter plot components
- Dropdown allows a users to choose specific launch site or all launch sites
- Pie chart shows the Total Success Launches by all launch sites or selected launch site from the drop-down menu
- Rangeslider allows users to select a payload mass in a fixed range
- Scatter chart shows the Correlation Between Payload Mass(KG) and number of Success/Failure landings for All sites or selected launch site from the drop-down menu

Github Link to the code

Predictive Analysis (Classification)

1. Data preparation

   - Load dataset
   - Normalize data
   - Split dataset into training and test sets.

2. Model preparation

   - Selection of machine learning algorithms
   - Set parameters for each algorithm
   - Training models with training dataset

3. Model evaluation

   - Get best hyperparameters for each type of model
   - Compute accuracy for each model
   - Plot Confusion Matrix

4. Model comparison

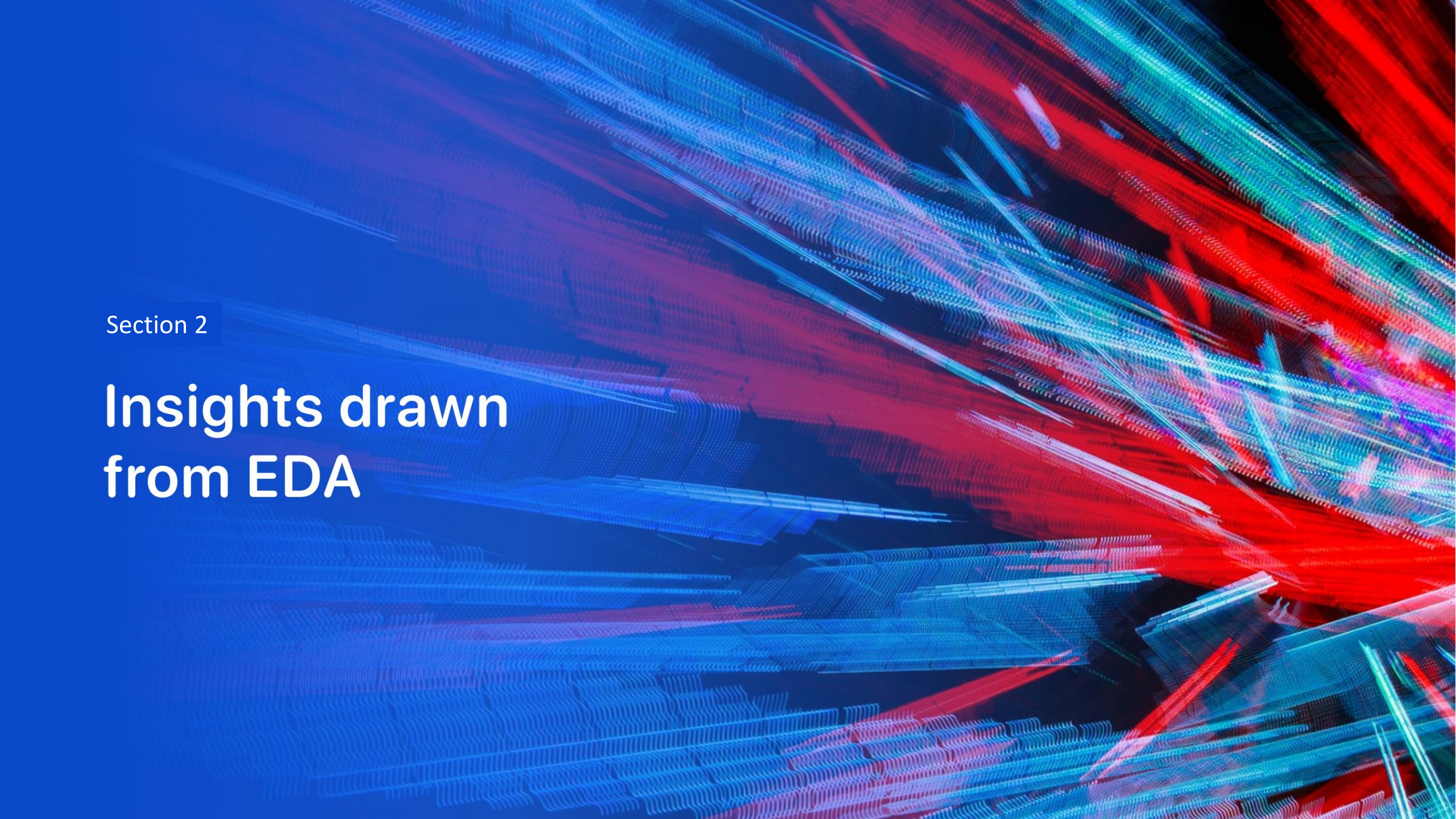
   - Compare the models according to their accuracy
   - Select the best model with the best accuracy

Results

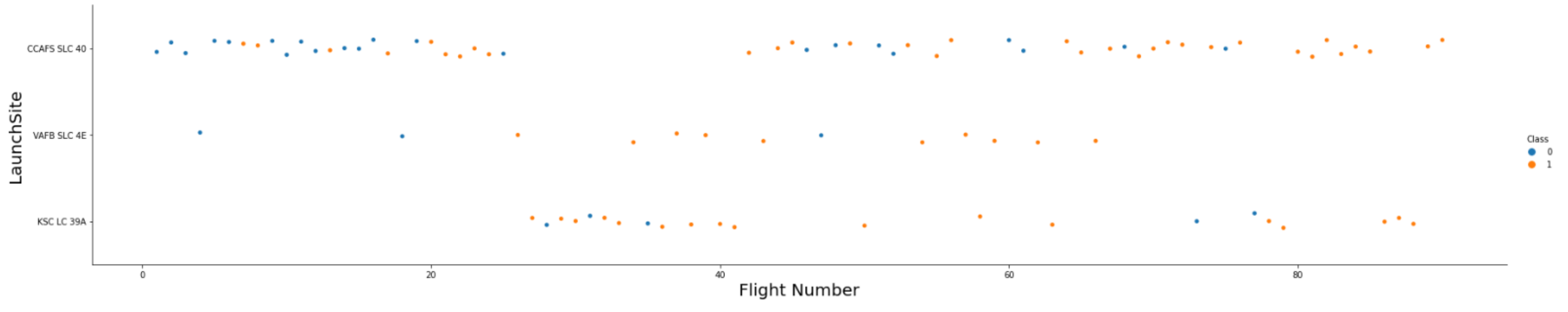
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

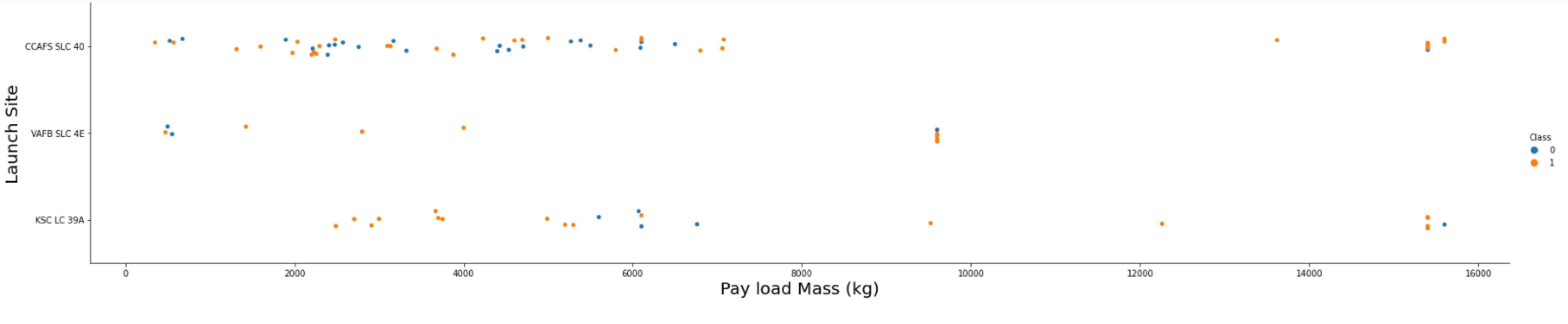


Flight Number vs. Launch Site



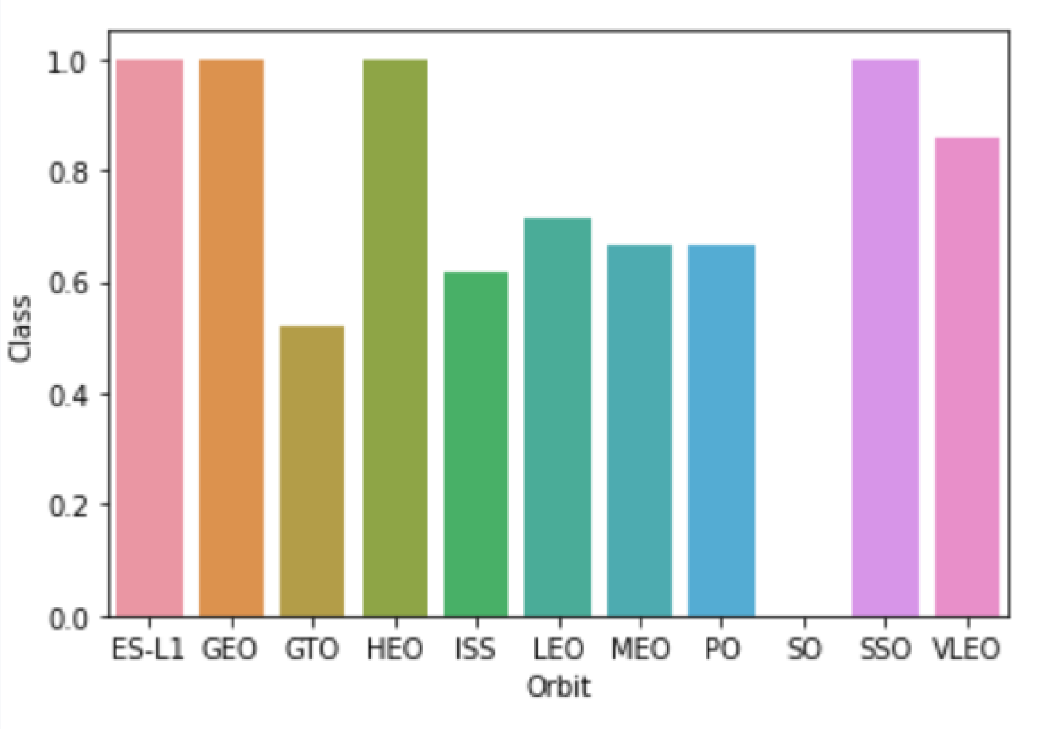
Launch site CCAFS SLC-40 has a significant higher launches than other sites
Also, the success rate of all sites are significantly higher than unsuccessful launches.

Payload vs. Launch Site



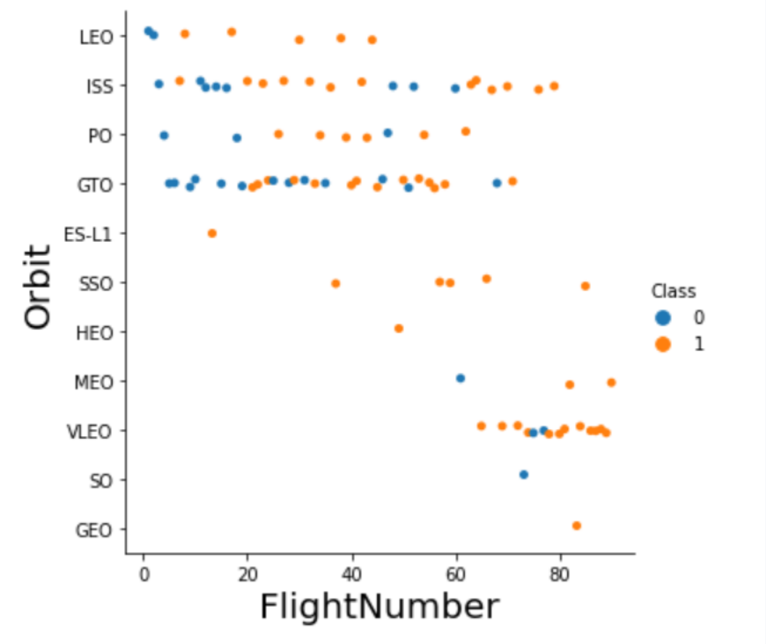
VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000)

Success Rate vs. Orbit Type



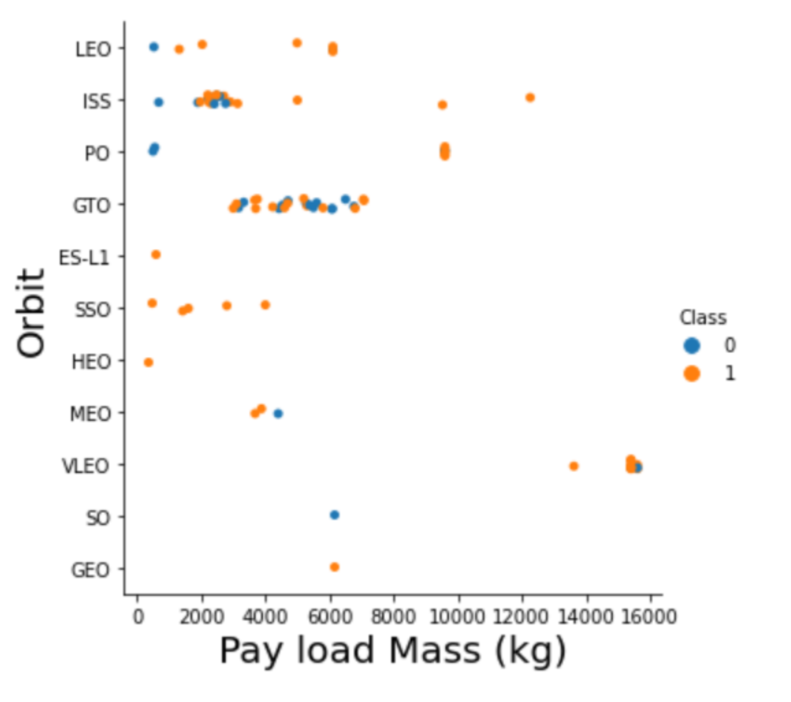
Rockets launched by the orbits ES-L1, GEO,HEO and SSO has a higher success rate than other orbits.

Flight Number vs. Orbit Type



The LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

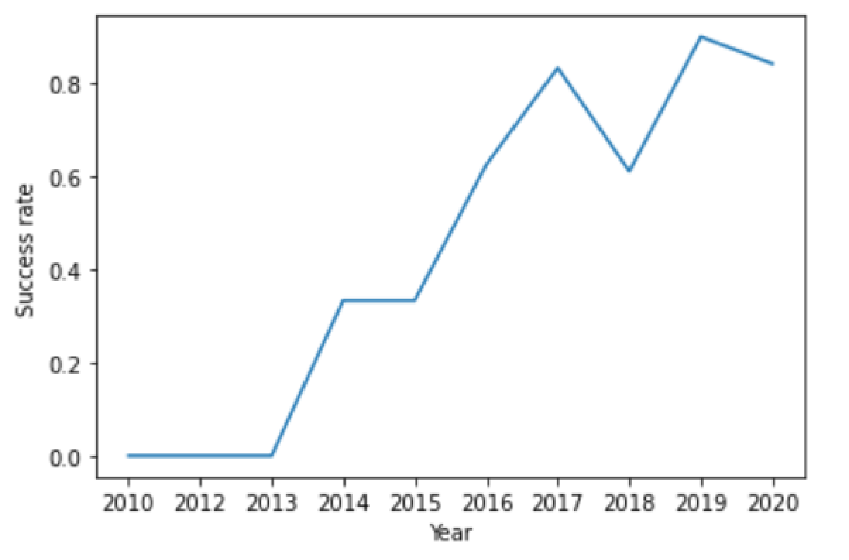
Payload vs. Orbit Type



With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

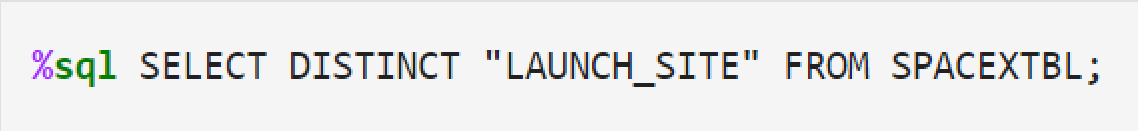
However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

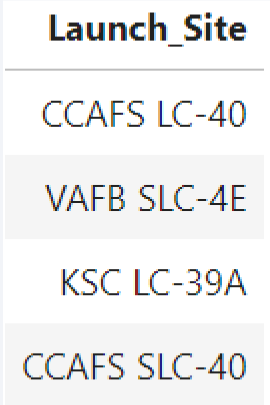


The success rate since 2013 kept increasing till 2020 due to the advancement of technology

All Launch Site Names



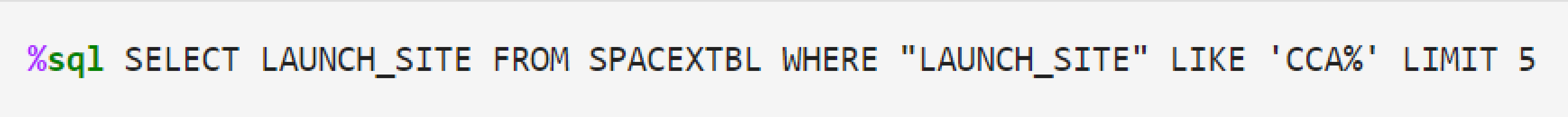
```
%sql SELECT DISTINCT "LAUNCH_SITE" FROM SPACEXTBL;
```



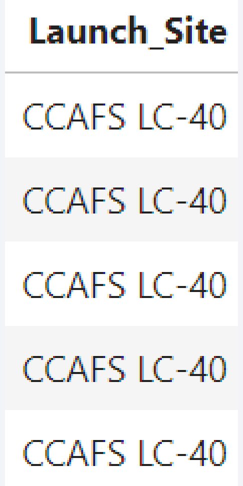
Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Used Unique function to remove duplicate launch sites

Launch Site Names Begin with 'CCA'



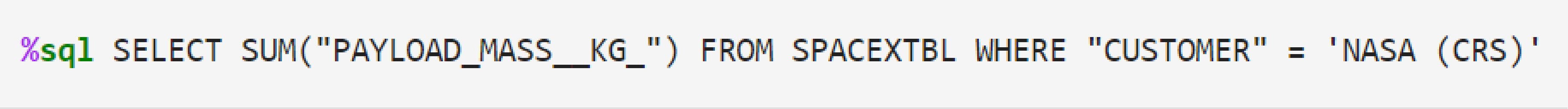
```
%sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE "LAUNCH_SITE" LIKE 'CCA%' LIMIT 5
```



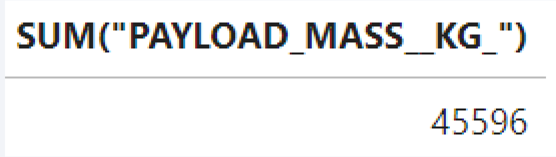
Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Used LIKE ‘CCA%’ to filter the Launch site names starts with CCA

Total Payload Mass



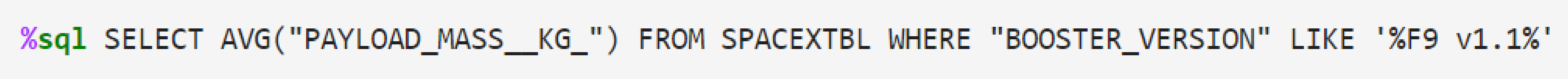
```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "CUSTOMER" = 'NASA (CRS)'
```



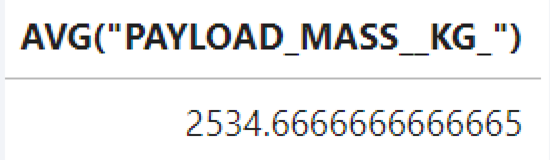
SUM("PAYLOAD_MASS__KG_")
45596

Total payload mass carried by boosters launched by NASA (CRS)

Average Payload Mass by F9 v1.1



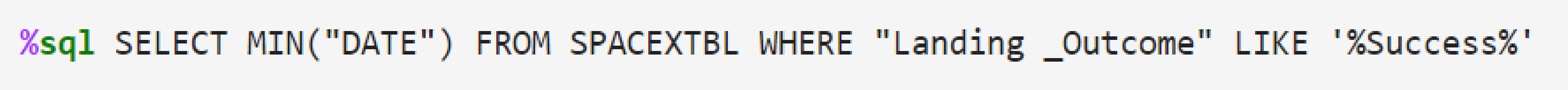
```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "BOOSTER_VERSION" LIKE '%F9 v1.1%'
```



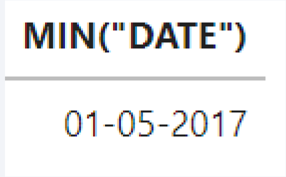
AVG("PAYLOAD_MASS__KG_")
2534.6666666666665

This is the average of all payload masses with booster versions contains ‘F9 v1.1’ substring

First Successful Ground Landing Date



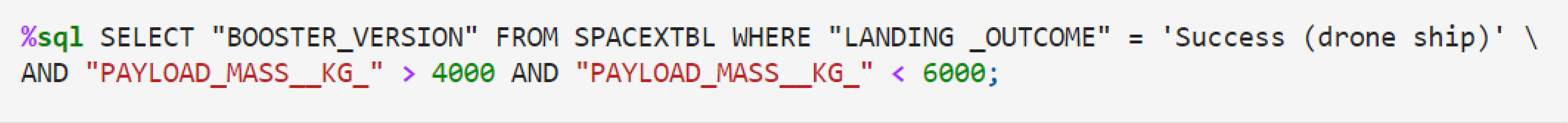
```
%sql SELECT MIN("DATE") FROM SPACEXTBL WHERE "Landing _Outcome" LIKE '%Success%'
```



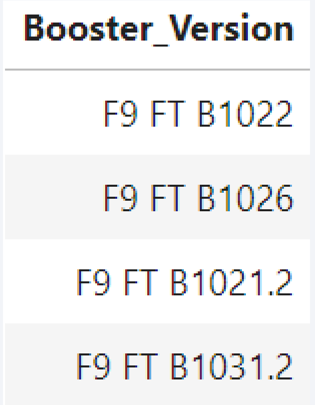
MIN("DATE")
01-05-2017

This is the oldest or earliest date of successful landing.

Successful Drone Ship Landing with Payload between 4000 and 6000



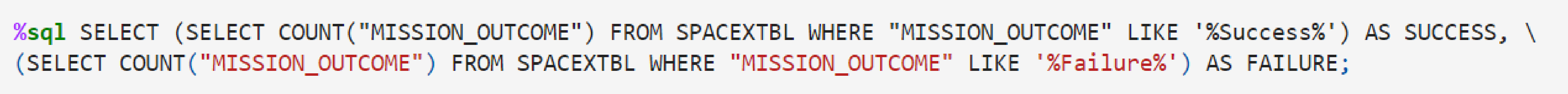
```
%sql SELECT "BOOSTER_VERSION" FROM SPACEXTBL WHERE "LANDING _OUTCOME" = 'Success (drone ship)' \
AND "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS__KG_" < 6000;
```



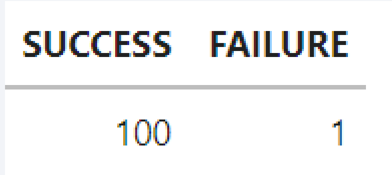
Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

The booster version where landing was successful and payload mass is between 4000 and 6000 kg.

Total Number of Successful and Failure Mission Outcomes



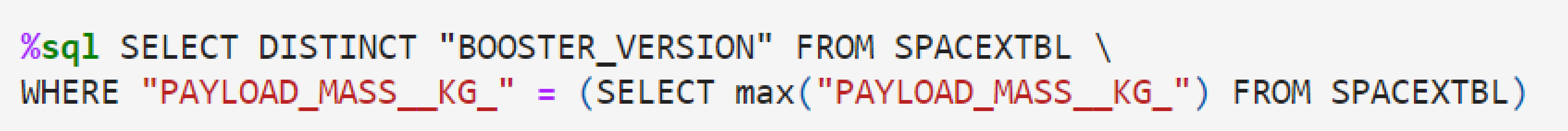
```
%sql SELECT (SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Success%') AS SUCCESS, \
(SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Failure%') AS FAILURE;
```



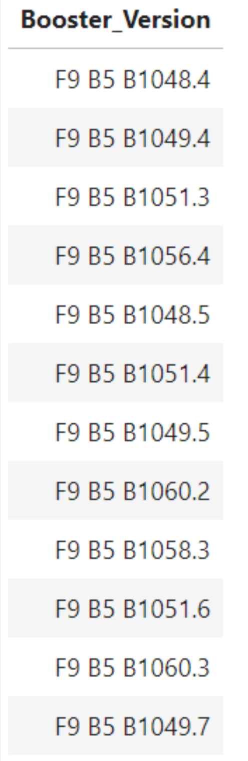
SUCCESS	FAILURE
100	1

Have used sub queries to get the success and failure outcomes. First sub query filtered the successful outcome while second sub query gives the failure outcome.

Boosters Carried Maximum Payload



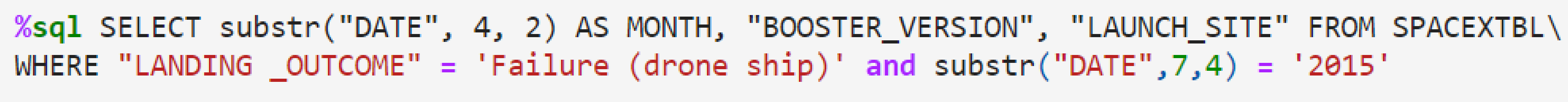
```
%sql SELECT DISTINCT "BOOSTER_VERSION" FROM SPACEXTBL \
WHERE "PAYLOAD_MASS__KG_" = (SELECT max("PAYLOAD_MASS__KG_") FROM SPACEXTBL)
```



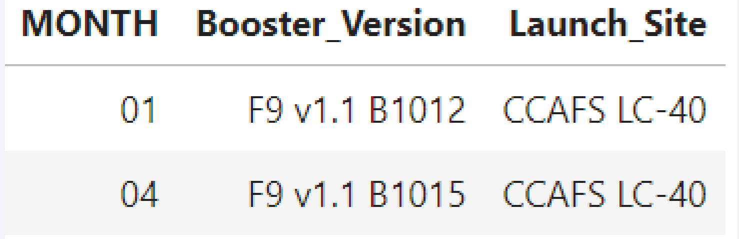
Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

First select the Maximum payload mass using the sub query and then select the unique booster versions that carry maximum payload mass.

2015 Launch Records



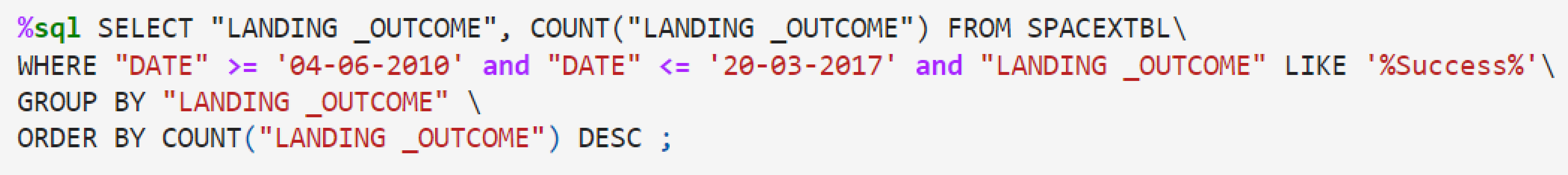
```
%sql SELECT substr("DATE", 4, 2) AS MONTH, "BOOSTER_VERSION", "LAUNCH_SITE" FROM SPACEXTBL\
WHERE "LANDING _OUTCOME" = 'Failure (drone ship)' and substr("DATE",7,4) = '2015'
```



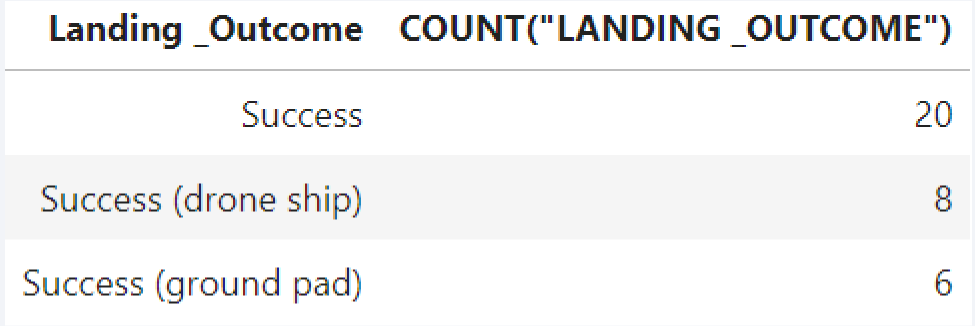
MONTH	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

The result table shows the month, booster version, launch site where landing was unsuccessful where the date somewhere in 2015.

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20



```
%sql SELECT "LANDING _OUTCOME", COUNT("LANDING _OUTCOME") FROM SPACEXTBL\
WHERE "DATE" >= '04-06-2010' and "DATE" <= '20-03-2017' and "LANDING _OUTCOME" LIKE '%Success%'\
GROUP BY "LANDING _OUTCOME" \
ORDER BY COUNT("LANDING _OUTCOME") DESC ;
```



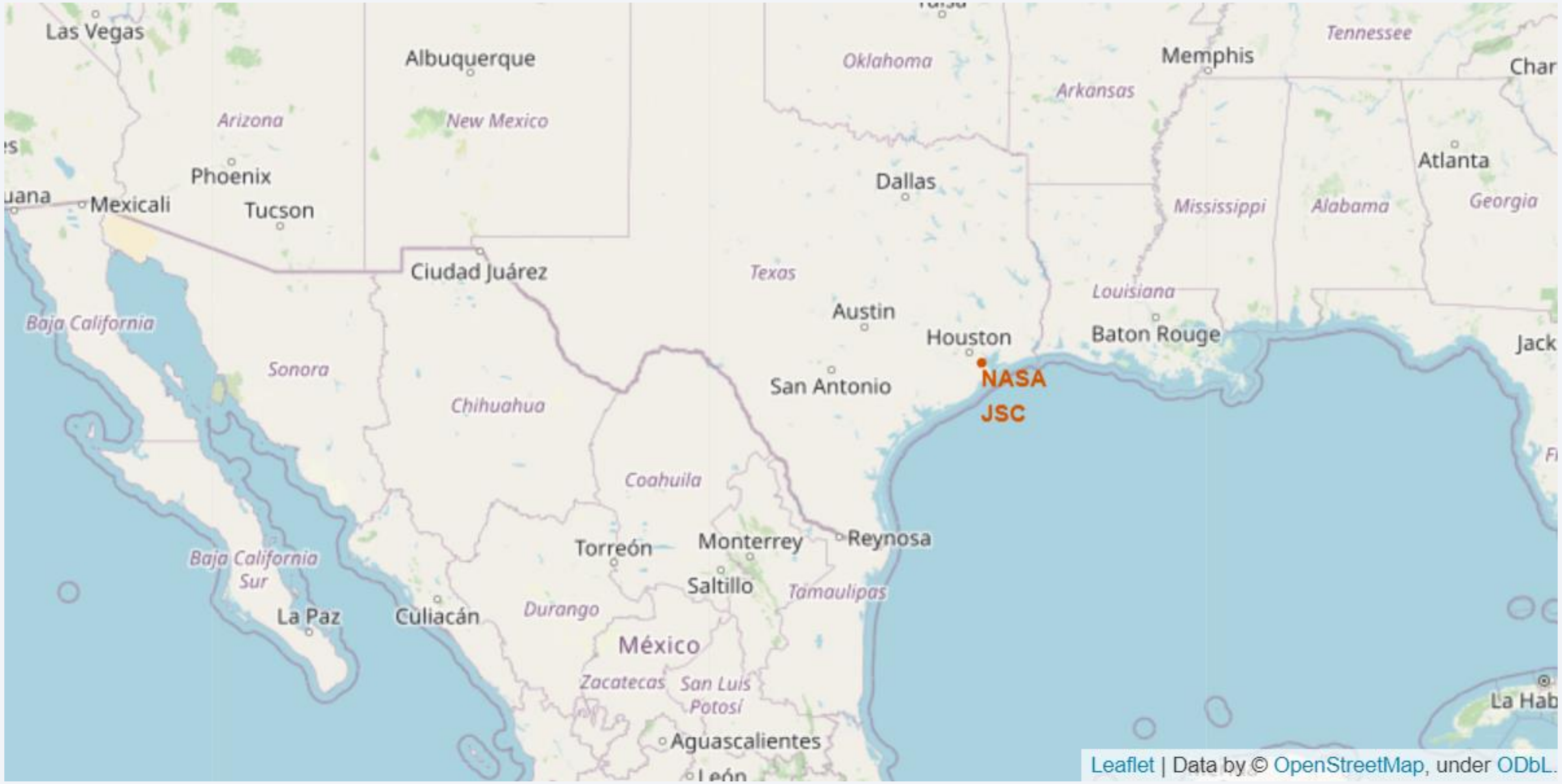
Landing _Outcome	COUNT("LANDING _OUTCOME")
Success	20
Success (drone ship)	8
Success (ground pad)	6

This frequency table shows the landing outcomes and their count where mission was successful, and date is between 04/06/2010 and 20/03/2017.

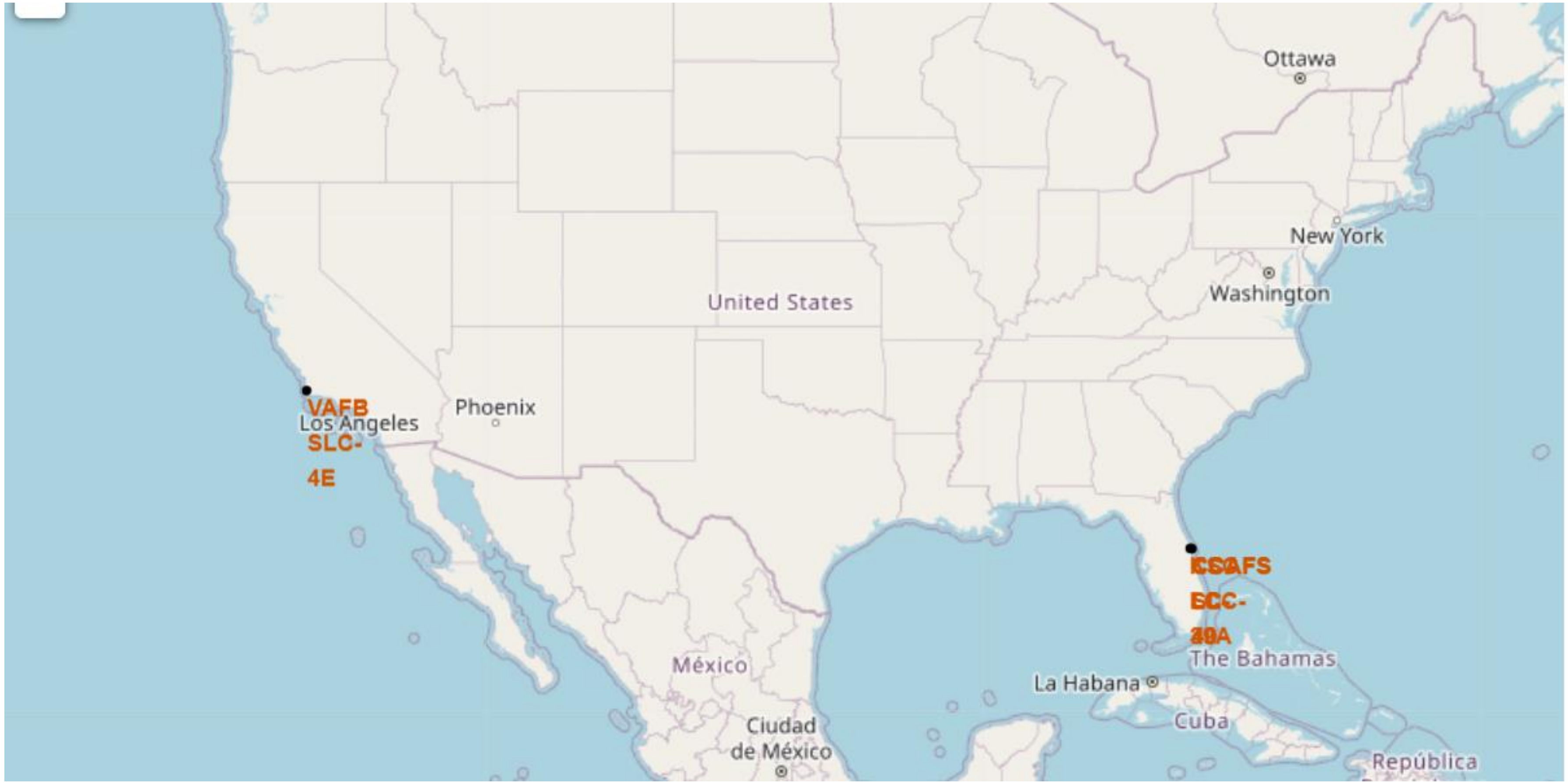
Section 3

Launch Sites Proximities Analysis

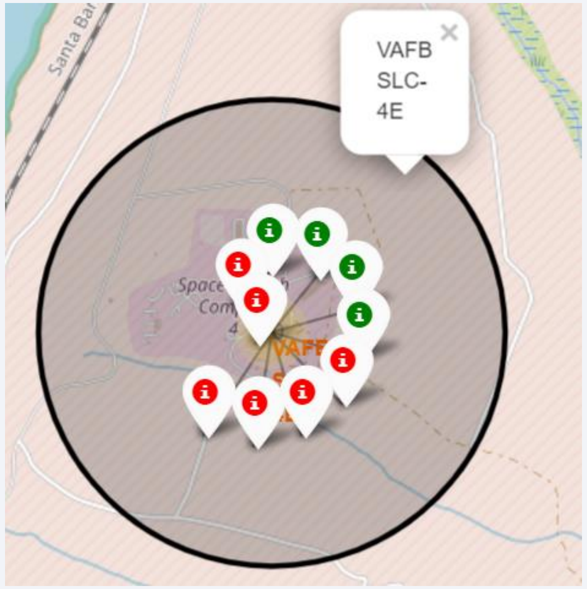
Folium Map- NASA Johnson Space Center at Houston, Texas

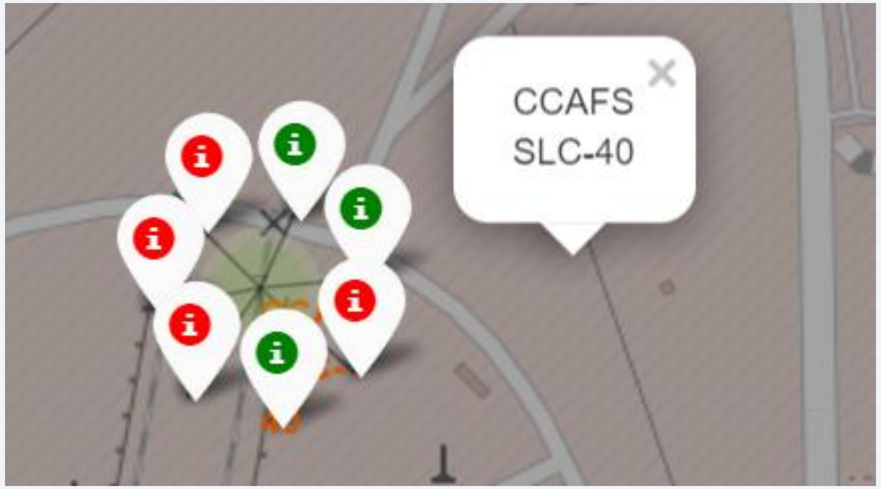


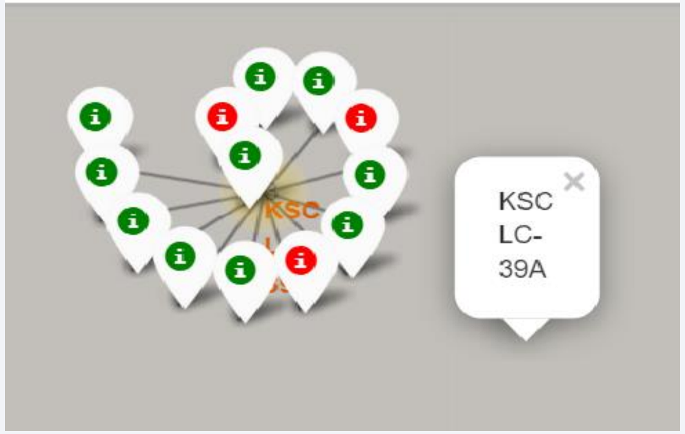
Folium Map with all Ground stations

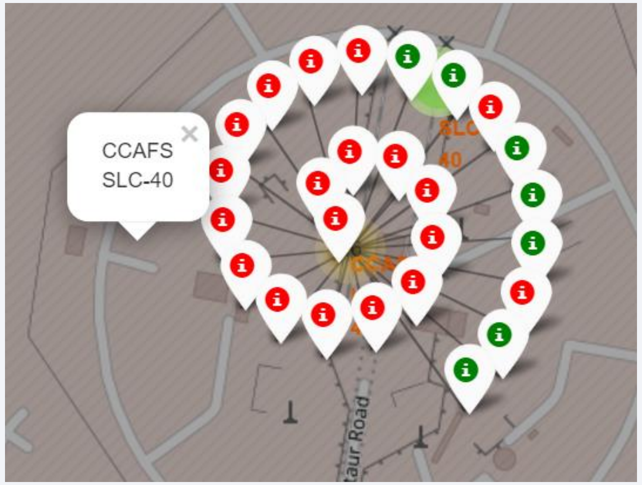


Folium map – Color Labeled Markers on the sites



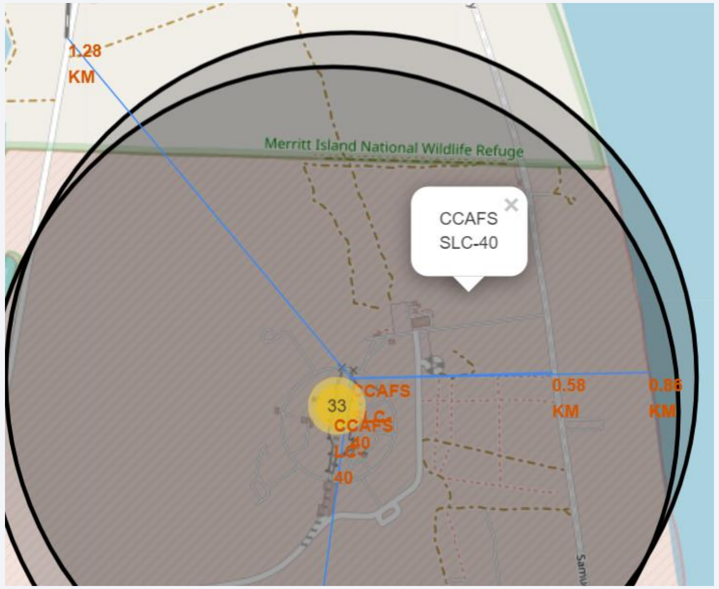


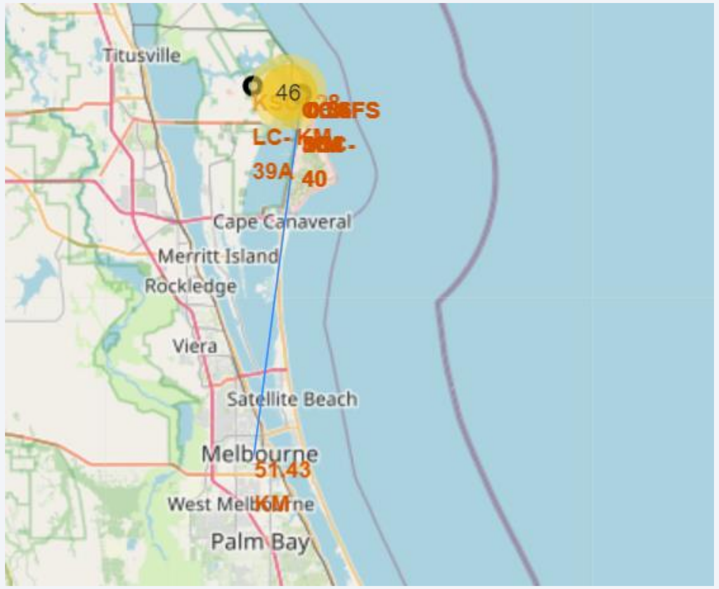


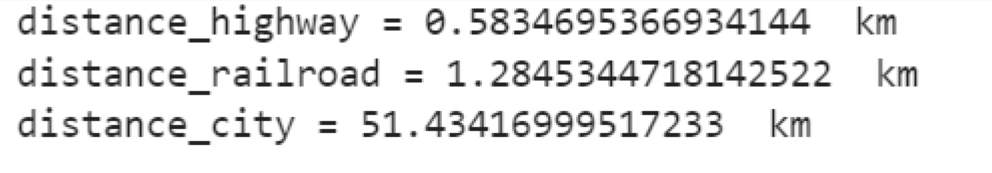


In each site, successful launches shows by green color and failure launches shows in red color.

Distances between CCAFS SLC-40 and its proximities

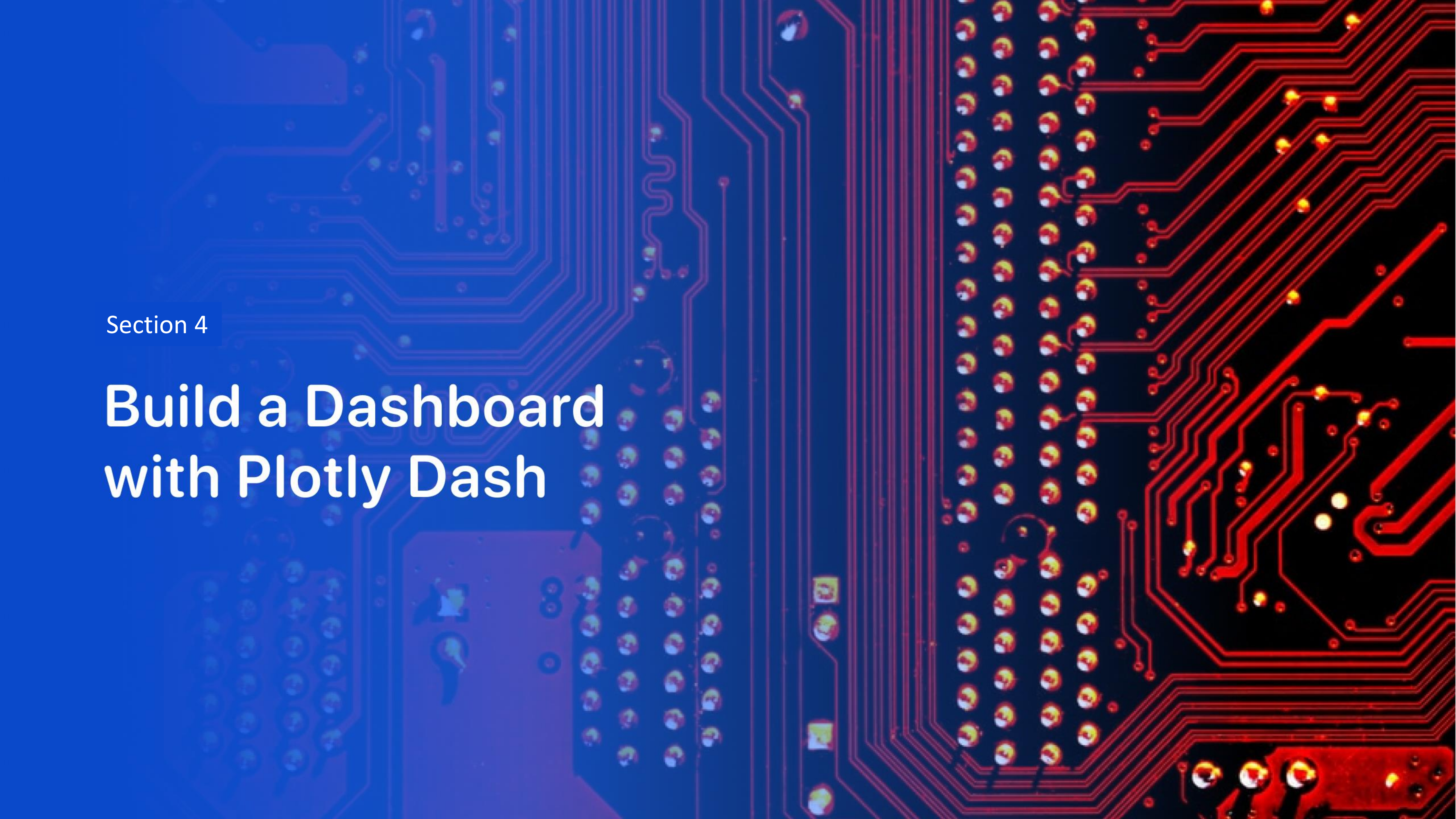






```
distance_highway = 0.5834695366934144  km
distance_railroad = 1.2845344718142522  km
distance_city = 51.43416999517233  km
```

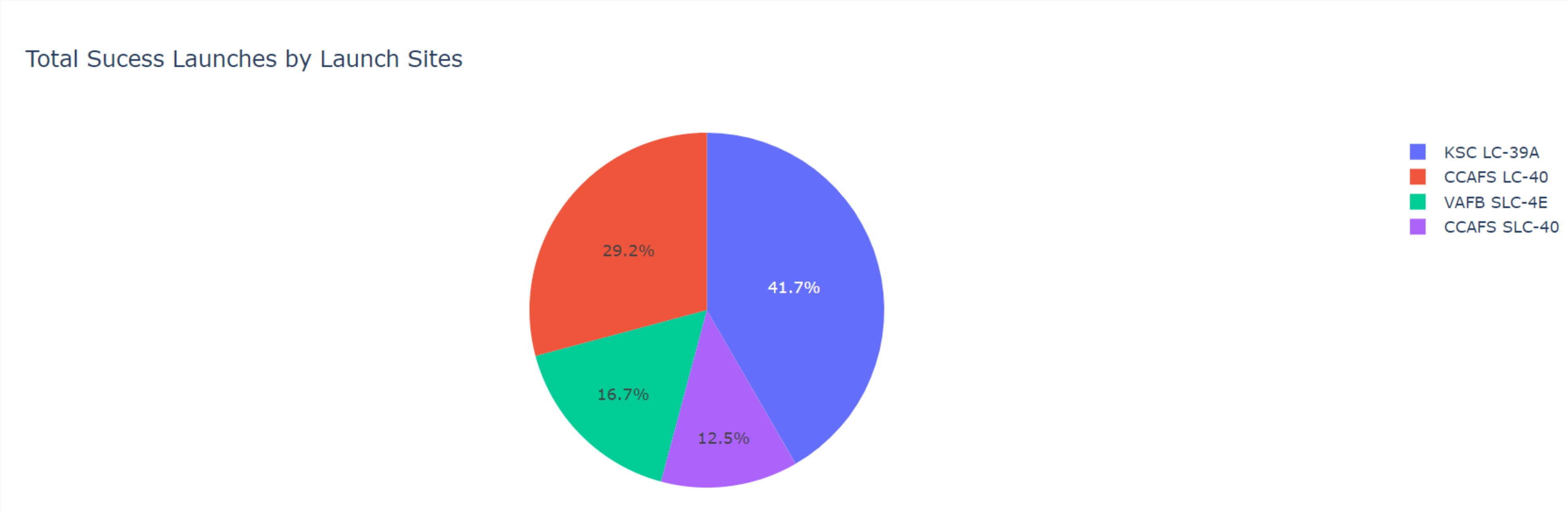
Is CCAFS SLC-40 in close proximity to railways ? Yes
Is CCAFS SLC-40 in close proximity to highways ? Yes
Is CCAFS SLC-40 in close proximity to coastline ? Yes
Do CCAFS SLC-40 keeps certain distance away from cities ? No



Section 4

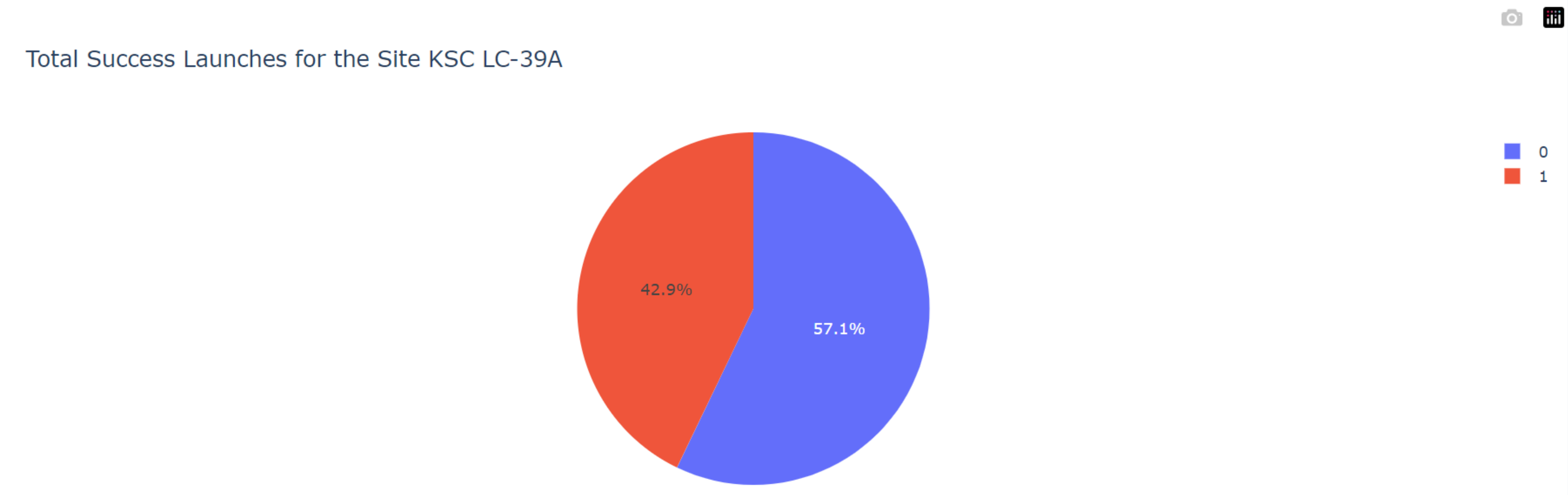
Build a Dashboard with Plotly Dash

Dashboard – Total success launches by Site



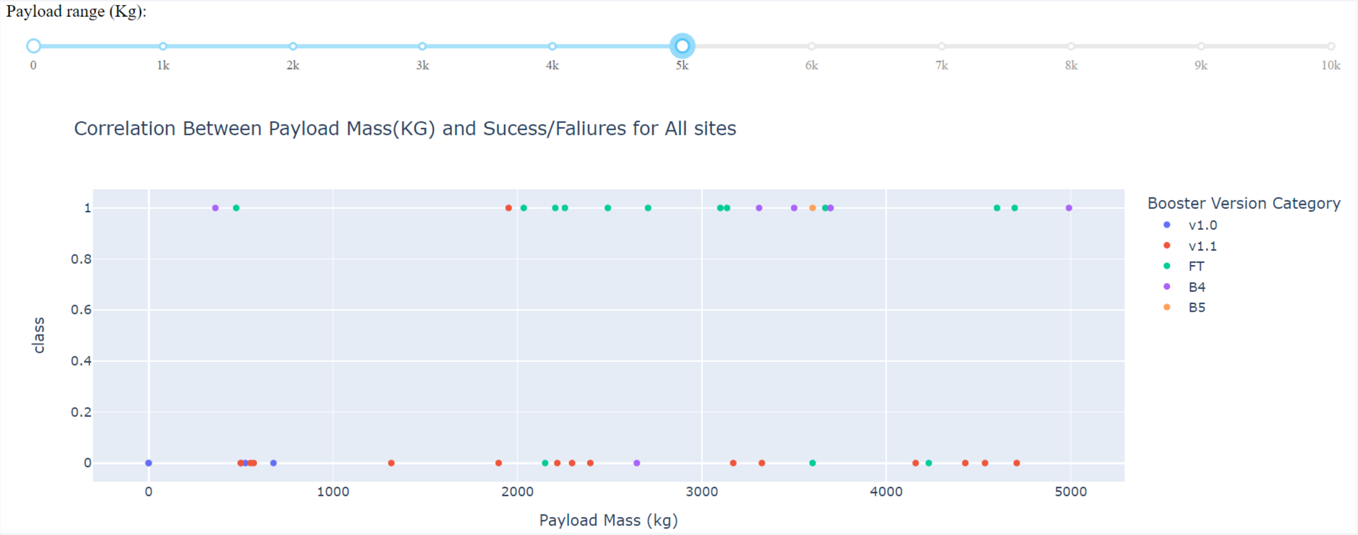
KSC LC-39A has the best success rate of launches which is 41.7%

Dashboard – Total success launches for Site KSC LC-39A



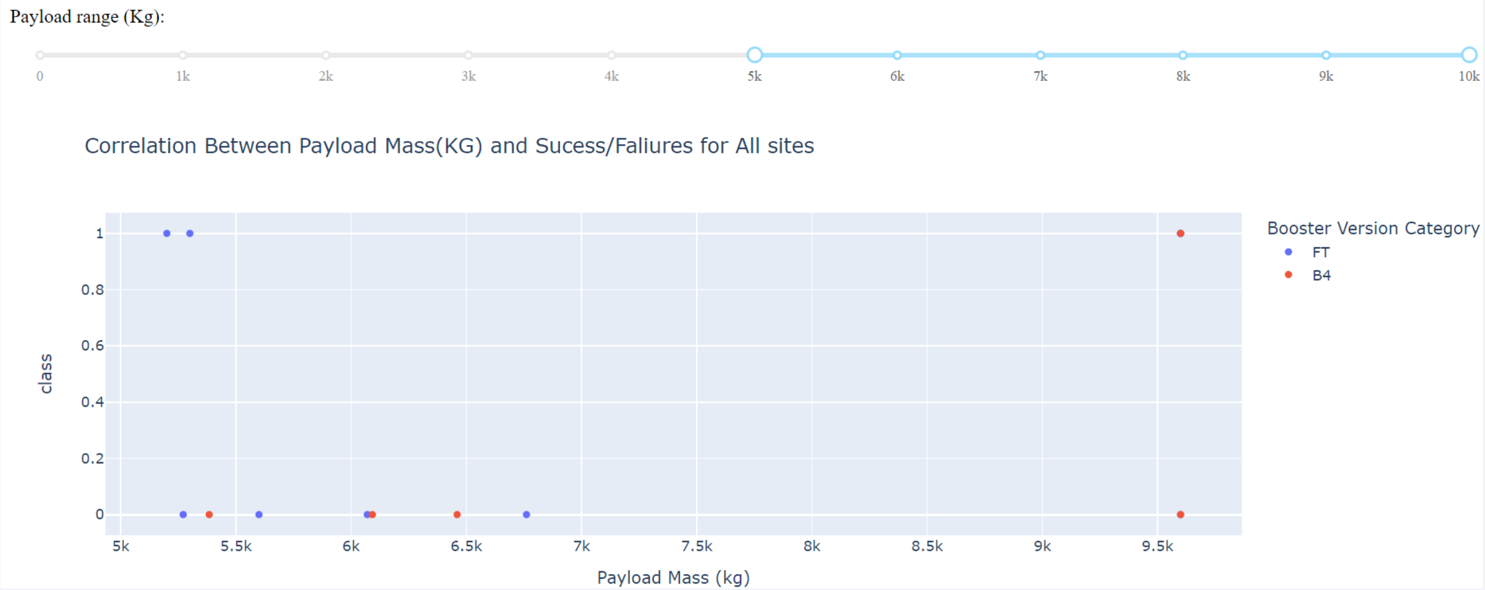
The site KSC LC-39A has a 76.9% success rate while 23.1% failure rate for launches.

Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider



First scatter plot shows the outcome for low weight (0-5000 Kg) launches whereas the second plot shows the outcome for high weight (5000-10000 Kg).

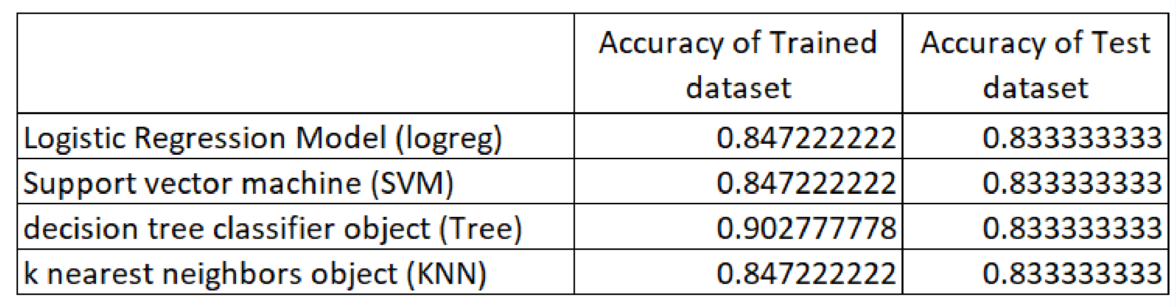
We can clearly see that the success rate for low weight launches are higher than the launches with high weights for all sites.



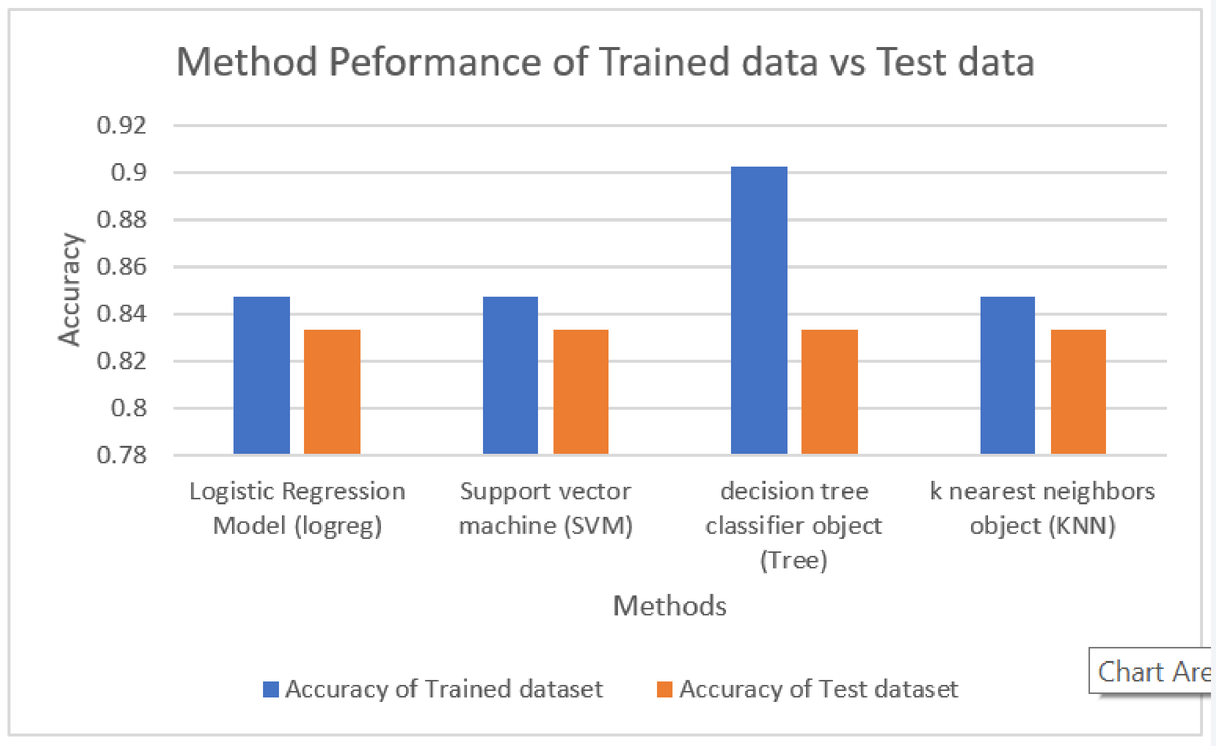
Section 5

Predictive Analysis (Classification)

Classification Accuracy

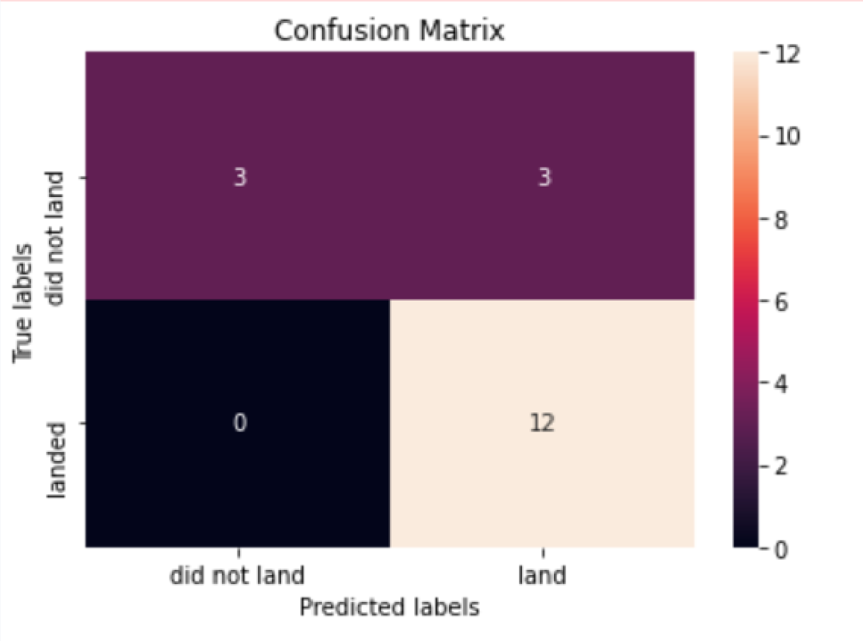


	Accuracy of Trained dataset	Accuracy of Test dataset
Logistic Regression Model (logreg)	0.847222222	0.833333333
Support vector machine (SVM)	0.847222222	0.833333333
decision tree classifier object (Tree)	0.902777778	0.833333333
k nearest neighbors object (KNN)	0.847222222	0.833333333



- Accuracy of Test data for all three methods performs same as shown in the graph which is 83.33%.
- Therefore, as the best model we select the decision tree method as it has a higher accuracy for the trained data which is 90.27%

Confusion Matrix



Confusion matrices for all four methods are same as shown above matrix. Seems like all four methods generate false positive results based on the results of equal accuracies.

Conclusions

- Output of a mission can be depended on the location of the launch site, orbit, launch history, and lessons learned from the previous launches.
- GEO, HEO, SSO, ES-L1 are the orbits that shows a best success rates on launching and landing.
- Based on the results launches with low weighted payloads perform better than the launches with heavy weighted payloads.
- Based on the results KSC LC-39A launch site identified as a best launch site to make success launch.
- The Decision Tree Algorithm has been selected as a best model even though the accuracy of test data are identical with other three methods. However, Decision Tree Method have proven higher accuracy on the training data. Therefore, we selected the Decision Tree Algorithm as a best model.

Appendix

- Link to the Data Collection API
- Link to the Data Collection with Web scraping
- Link to the Exploratory Data Analysis (EDA)
- Link to the EDA with SQL
- Link to the EDA with Data Visualization
- Link to the launch site analysis by Folium Maps
- Dashboard by Plotly Dash
- Github link to the code for the dashboard
- Dash for the sites with highest launch ratio
- Link to the SpaceX machine learning and prediction
- Link to the PDF of Machine Learning Outputs

Thank you!