IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

KHAWAR HASNAIN
28 JULY - 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result from Machine Learning Lab

Introduction

SpaceX has revolutionized the space industry by offering Falcon 9 rocket launches at just 62 million dollars, compared to competitors' prices of 165 million dollars. Their cost-saving innovation involves reusing the first stage of the rocket, leading to further reductions in launch expenses. As a data scientist for a SpaceX competitor, my crucial project is developing a machine learning pipeline to predict first stage landings for better bidding on rocket launches.

The problems included:

- Identifying all factors that influence the landing outcome.
- The relationship between each variables and how it is affecting the outcome.
- The best condition needed to increase the probability of successful landing.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API and Web Scrapping Methodology
- Perform data wrangling
  - One hot encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

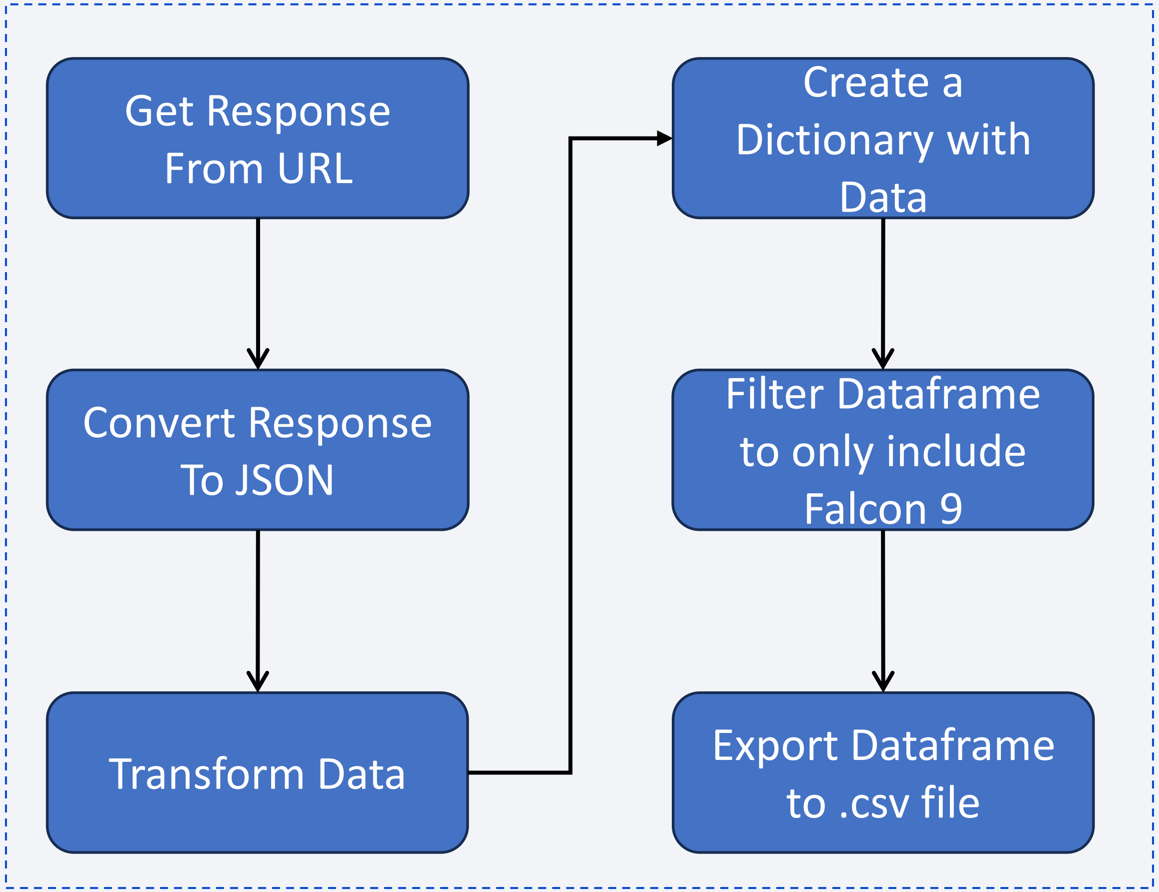
Data collection is the process of gathering and measuring information on targeted variables in an established system, which then enables one to answer relevant questions and evaluate outcomes.

As mentioned, the dataset was collected by REST API and Web Scrapping from Wikipedia For REST API, its started by using the get request. Then, we decoded the response content as Json and turn it into a pandas dataframe using json_normalize(). We then cleaned the data, checked for missing values and fill with whatever needed.

For web scrapping, we will use the BeautifulSoup to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for further analysis

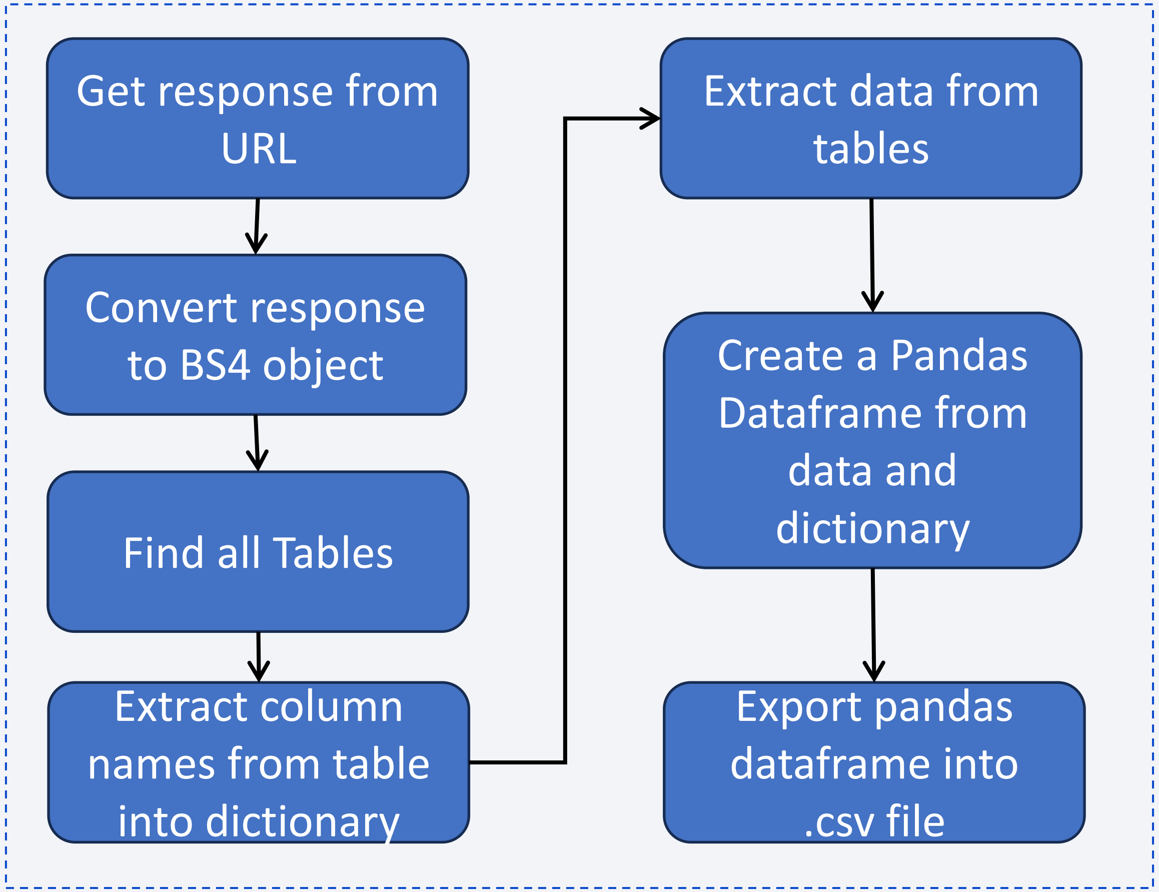
Data Collection – SpaceX API

- The API link to access Falcon9 Data is: API Link
- The github link to jupyter notebook is: Data Collection API Notebook


Get Response From URL
Convert Response To JSON
Transform Data
Create a Dictionary with Data
Filter Dataframe to only include Falcon 9
Export Dataframe to .csv file

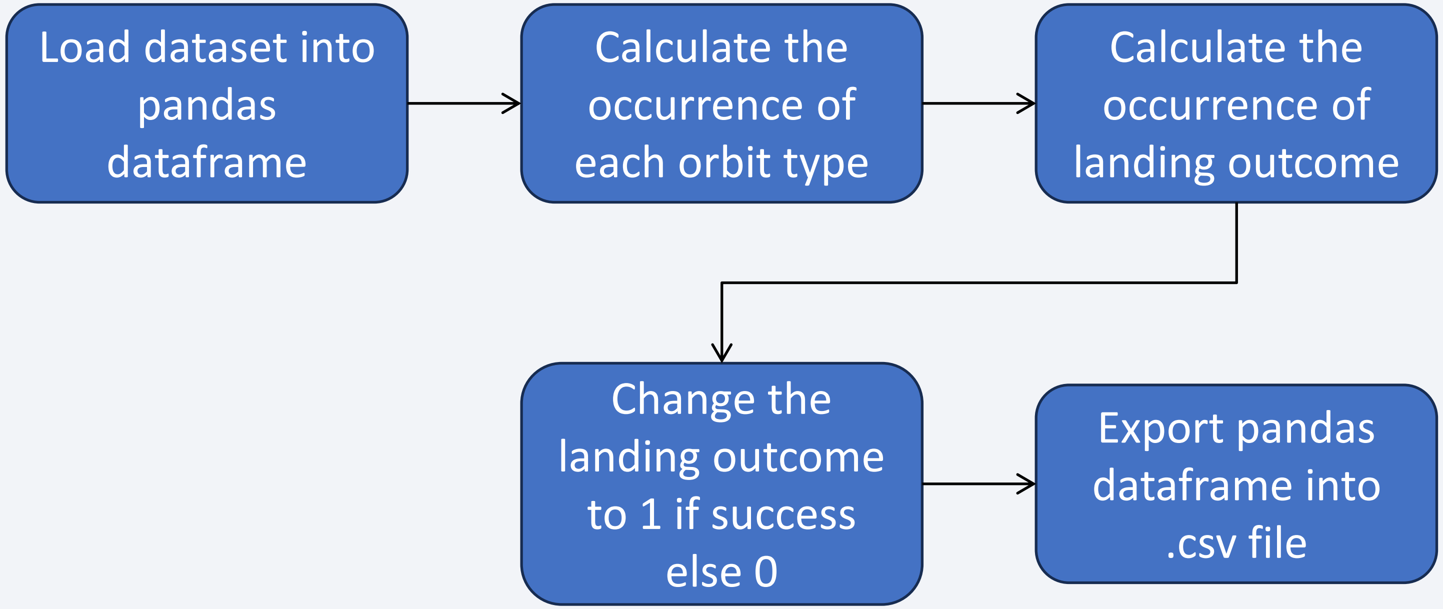
Data Collection - Scraping

- The data is scrapped from Wikipedia webpage: [Wikipedia page for Falcon 9 Launches]
- The Github url for jupyter notebook is: [Web Scrapping Notebook]


Get response from URL
Convert response to BS4 object
Find all Tables
Extract column names from table into dictionary
Extract data from tables
Create a Pandas Dataframe from data and dictionary
Export pandas dataframe into .csv file

Data Wrangling

- In the dataset we changes the outcomes to 1 and 0 to identify failed and successful launches.
- The github link for notebook: Data Wrangling Notebook



Load dataset into pandas dataframe

Calculate the occurrence of each orbit type

Calculate the occurrence of landing outcome

Change the landing outcome to 1 if success else 0

Export pandas dataframe into .csv file

EDA with Data Visualization

- Following charts were used to visualize data:
  - Scatter plot are very helpful in determining the correlation between different variables.
  - Bar graph shows the relationship between numeric and categorical values.
  - Line graph shows data variables and their trends. Line graph can help show global behavior and make predictions for unseen data.
- The github link for notebook: EDA Notebook

EDA with SQL

- Following queries were performed:
  - Displaying 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 records which will display the month names, faiilure landing_ouutcomes in drone ship, booster versions, launch_site for themonths in year 2015.
  - Rank the count of successful landiing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.
- Github Link: EDA SQL Notebook

Build an Interactive Map with Folium

- Folium map object is a map centered on NASA Johnson Space Center at Houson, Texas
  - Red circle at NASA Johnson Space Center's coordinate with label showing its name *(folium.Circle, folium.map.Marker).*
  - Red circles at each launch site coordinates with label showing launch site name *(folium.Circle, folium.map.Marker, folium.features.DivIcon).*
  - The grouping of points in a cluster to display multiple and different information for the same coordinates *(folium.plugins.MarkerCluster).*
  - Markers to show successful and unsuccessful landings. Green for successful landing and Red for unsuccessful landing. *(folium.map.Marker, folium.Icon).*
  - Markers to show distance between launch site to key locations (railway, highway, coastway, city) and plot a line between them*. (folium.map.Marker, folium.PolyLine, folium.features.DivIcon)*
- These objects are created in order to understand better the problem and the data. We can show easily all launch sites, their surroundings and the number of successful and unsuccessful landings.
- The Github link: Folium Map Notebook

Build a Dashboard with Plotly Dash

- Dashboard has dropdown, pie chart, rangeslider and scatter plot components
    - Dropdown allows a user to choose the launch site or all launch sites *(dash_core_components.Dropdown).*
    - Pie chart shows the total success and the total failure for the launch site chosen with the dropdown component *(plotly.express.pie).*
    - Rangeslider allows a user to select a payload mass in a fixed range *(dash_core_components.RangeSlider).*
    - Scatter chart shows the relationship between two variables, in particular Success vs Payload Mass *(plotly.express.scatter).*
- Github link: Plotly Dash App

Predictive Analysis (Classification)

- Data preparation
  - Load dataset.
  - Normalize data and split data into training and test sets.
- Model preparation
  - Selection of machine learning algorithms.
  - Set parameters for each algorithm to GridSearchCV.
  - Training GridSearchModel models with training dataset.
- Model evaluation
  - Get best hyperparameters for each type of model.
  - Compute accuracy for each model with test dataset and plot Confusion Matrix.
- Model comparison
  - Comparison of models according to their accuracy.
  - The model with the best accuracy will be chosen.
- Github Link: ML Notebook

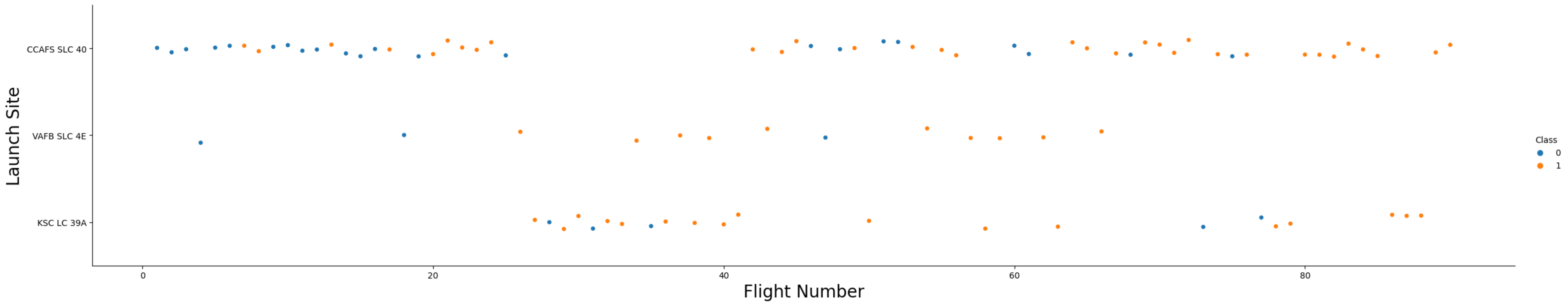
Results

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

Section 2

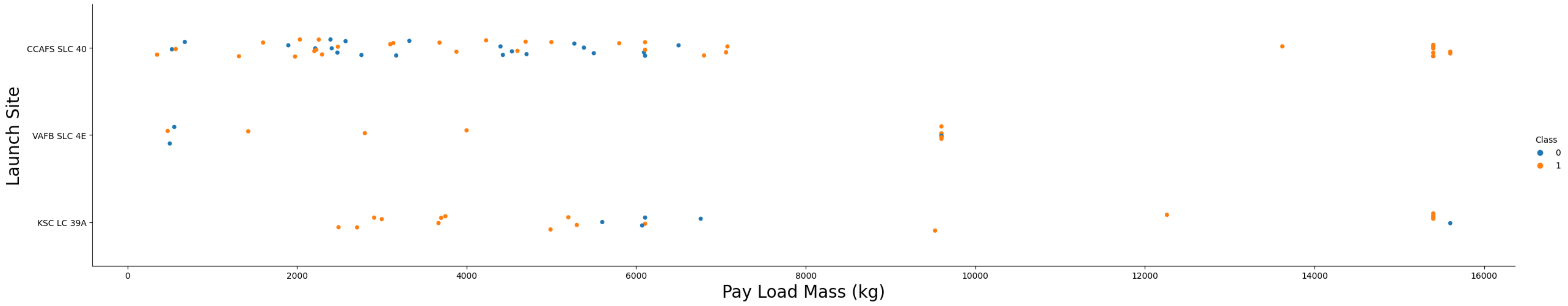
Insights drawn from EDA

Flight Number vs. Launch Site



- This scatter plot help us conclude that the success rate of launches on different launch sites is increasing.

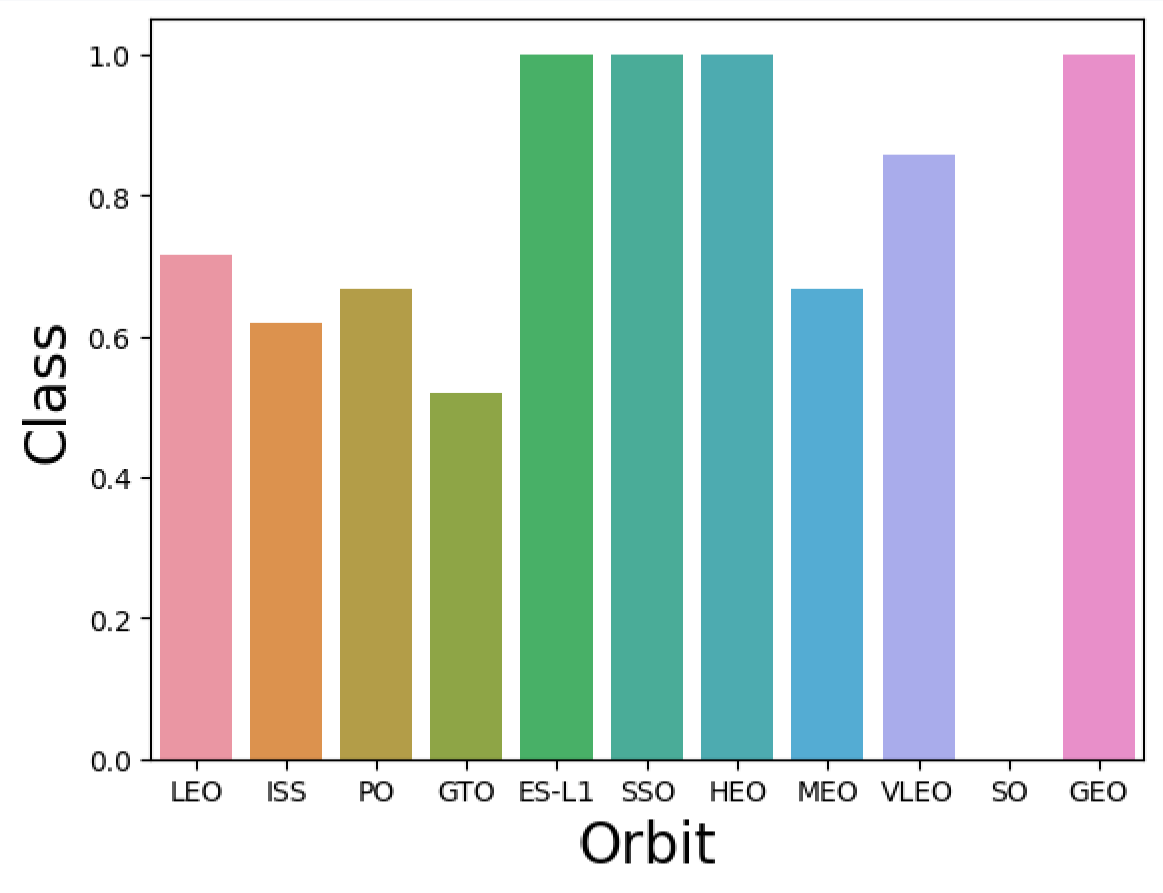
Payload vs. Launch Site



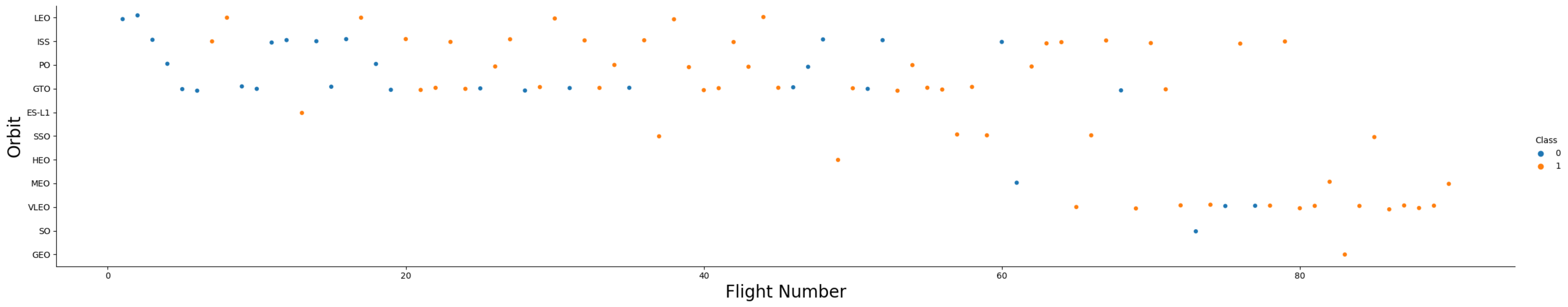
- Depending on the launch site, a heavier payload may be a consideration for a successful landing. On the other hand, a too heavy payload can make a landing fail.

Success Rate vs. Orbit Type

- This bar chart shows us the success rate for different orbits.
- The orbits ES-L1, SSO, HEO and GEO has the highest success rate.

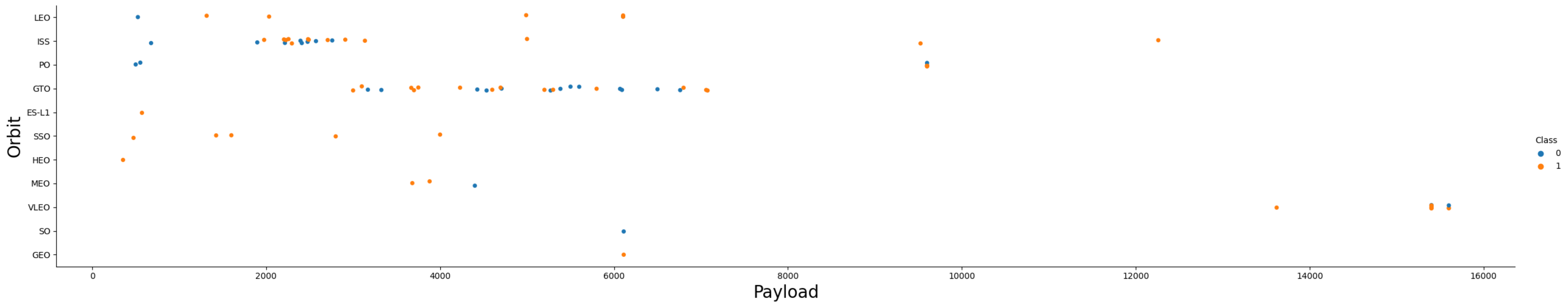


Flight Number vs. Orbit Type



- In 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

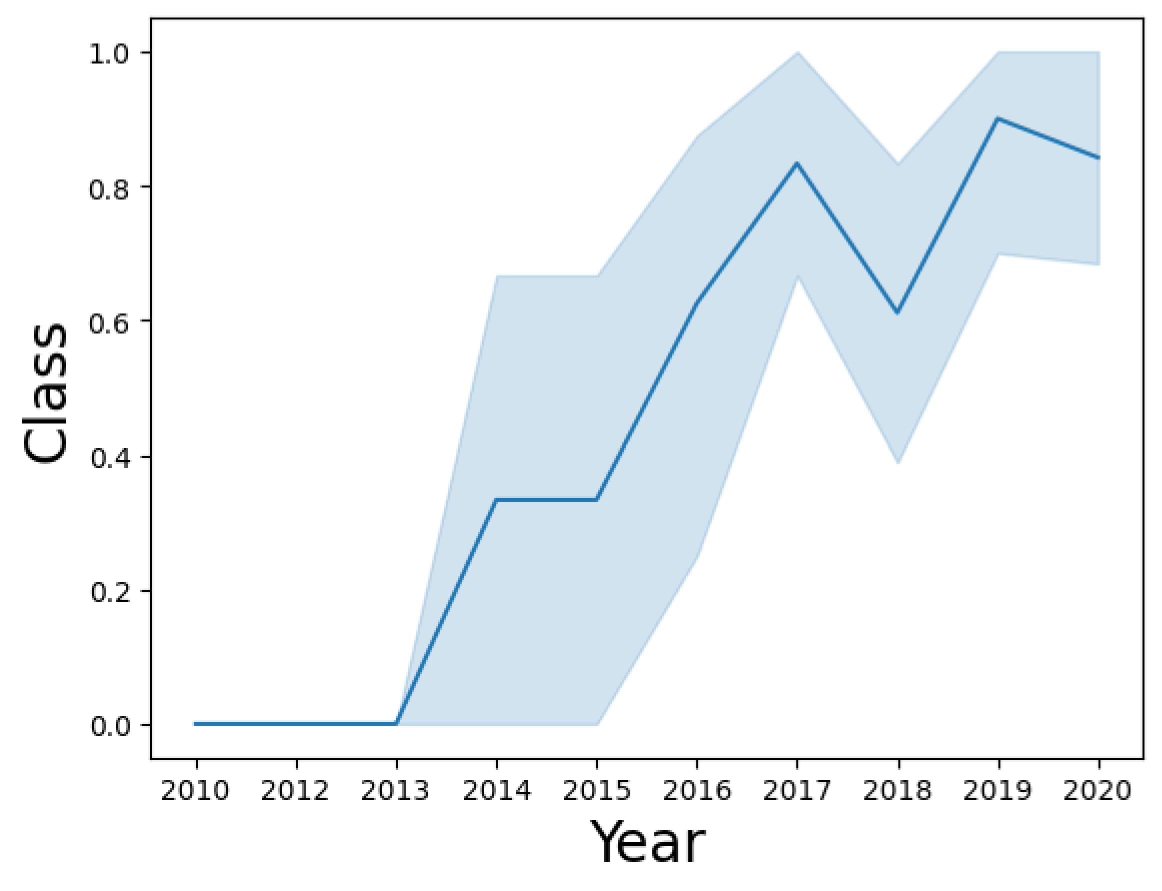


- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.

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

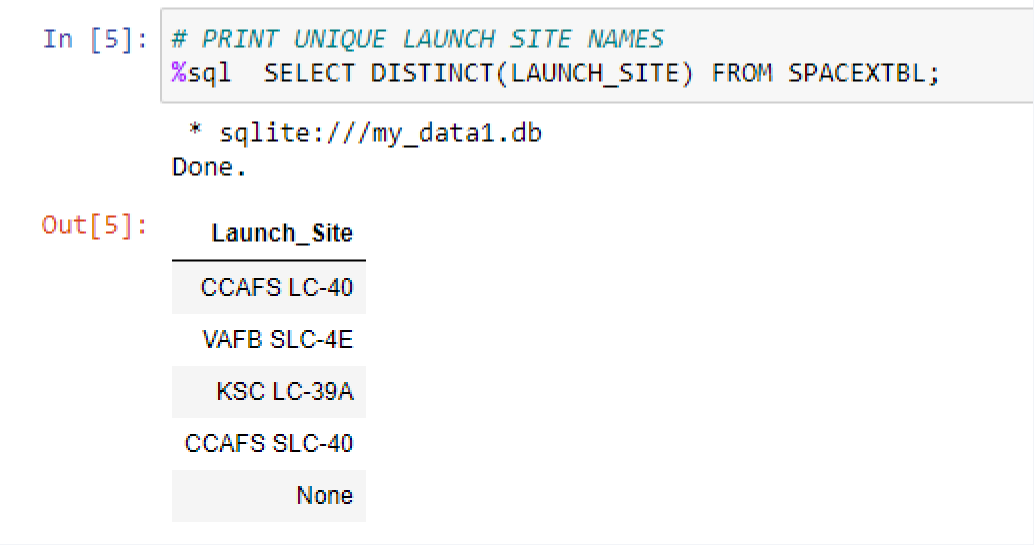
Launch Success Yearly Trend

- From the line plot we can see that the success rate kept increasing since 2013 till 2020.



All Launch Site Names

- This query returns a list of unique Launch Site names.



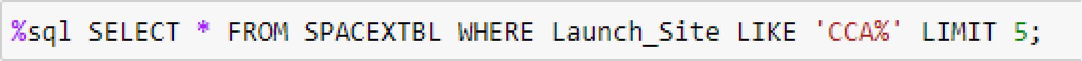
```
In [5]: # PRINT UNIQUE LAUNCH SITE NAMES
        %sql  SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL;

         * sqlite:///my_data1.db
        Done.
```

Out[5]:

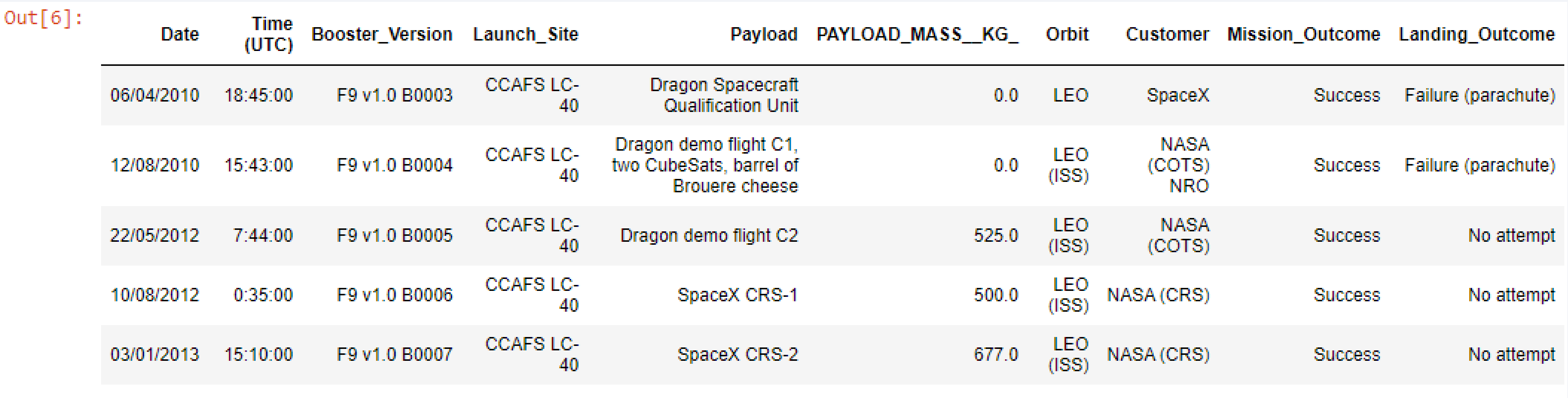
Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

Launch Site Names Begin with 'CCA'

- This query find 5 records where launch sites begin with \`CCA\`.



```
%sql SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
```

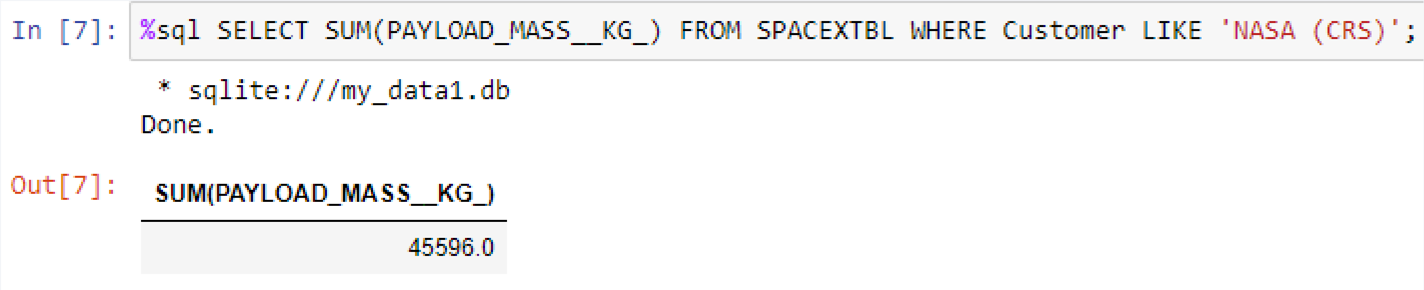


Out[6]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- This query calculate the total payload carried by boosters from SpaceX customer NASA.



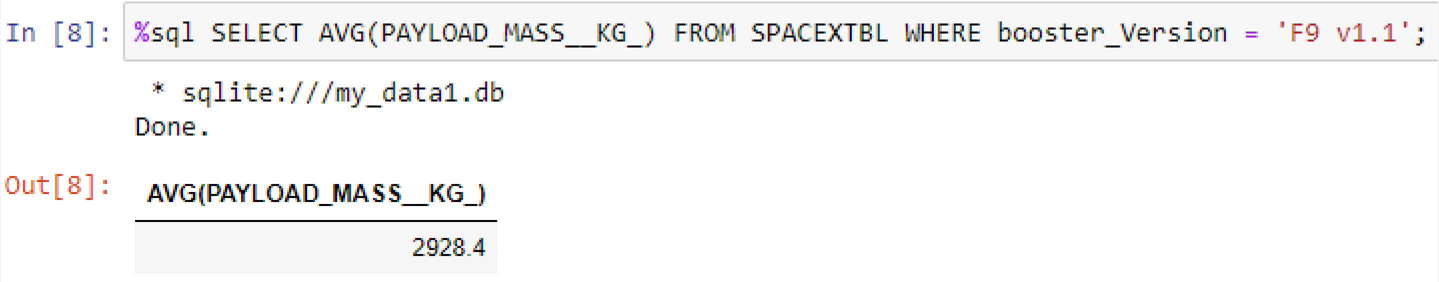
```
In [7]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer LIKE 'NASA (CRS)';

         * sqlite:///my_data1.db
        Done.
```

Out[7]:

SUM(PAYLOAD_MASS__KG_)
45596.0

Average Payload Mass by F9 v1.1

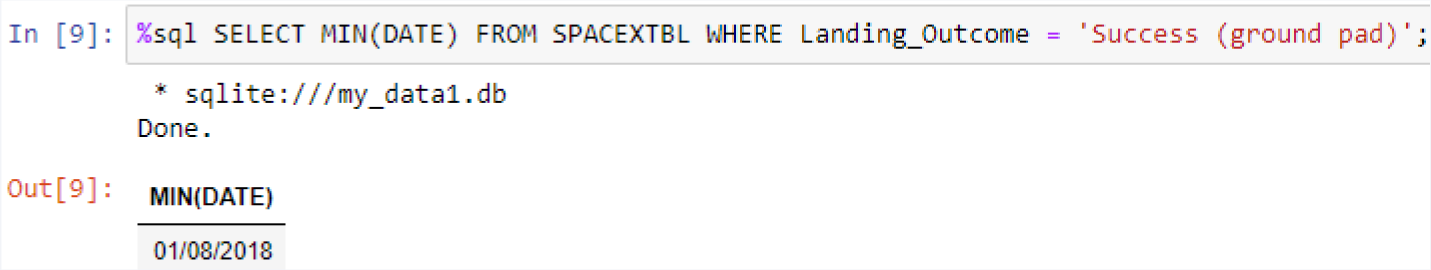
- This query calculate the average payload mass carried by booster version F9 v1.1.



```
In [8]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE booster_Version = 'F9 v1.1';

         * sqlite:///my_data1.db
        Done.

Out[8]:  AVG(PAYLOAD_MASS__KG_)
                        2928.4
```

First Successful Ground Landing Date

- This query prints the date of the first successful landing outcome on ground pad using the MIN function.

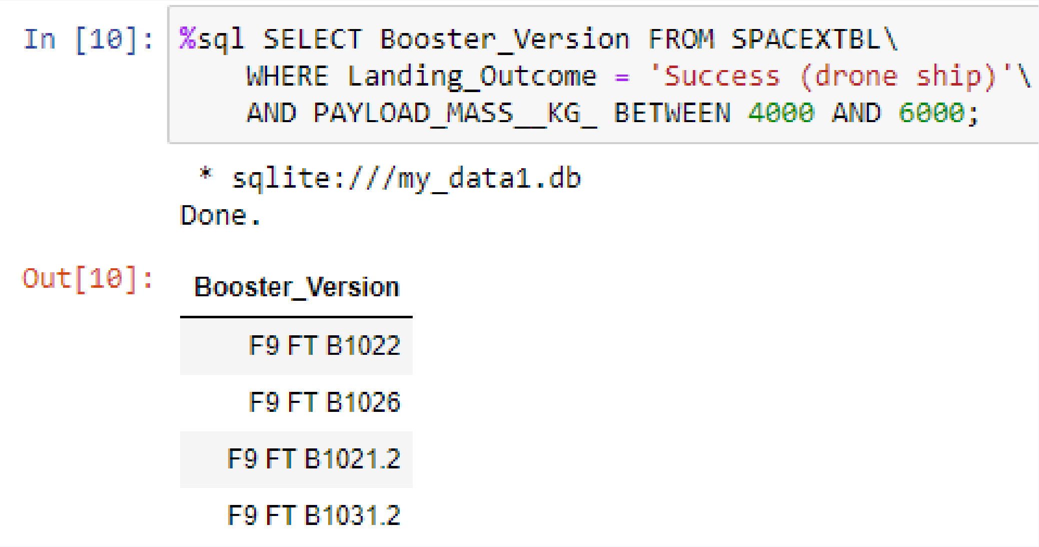


```
In [9]: %sql SELECT MIN(DATE) FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)';

 * sqlite:///my_data1.db
Done.

Out[9]:
```

MIN(DATE)
01/08/2018

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 using BETWEEN.

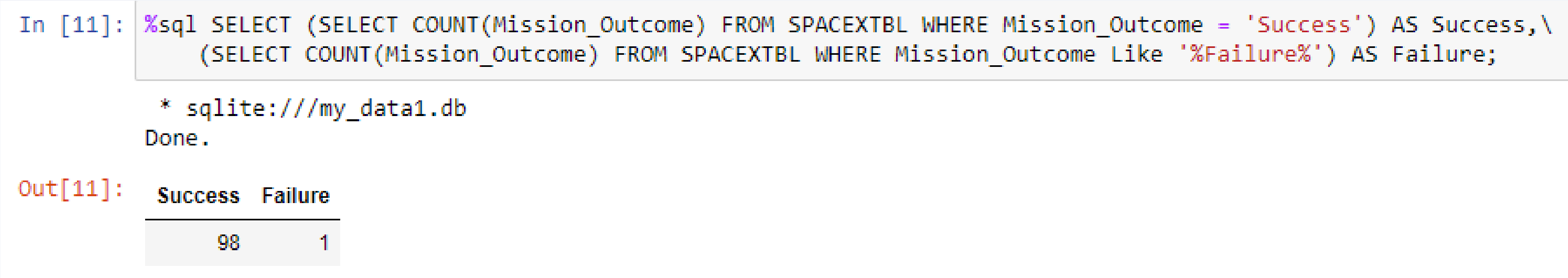


```
In [10]: %sql SELECT Booster_Version FROM SPACEXTBL\
             WHERE Landing_Outcome = 'Success (drone ship)'\
             AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;

          * sqlite:///my_data1.db
         Done.

Out[10]:
```

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

Total Number of Successful and Failure Mission Outcomes

- This query calculates the total number of successful and failure mission outcomes



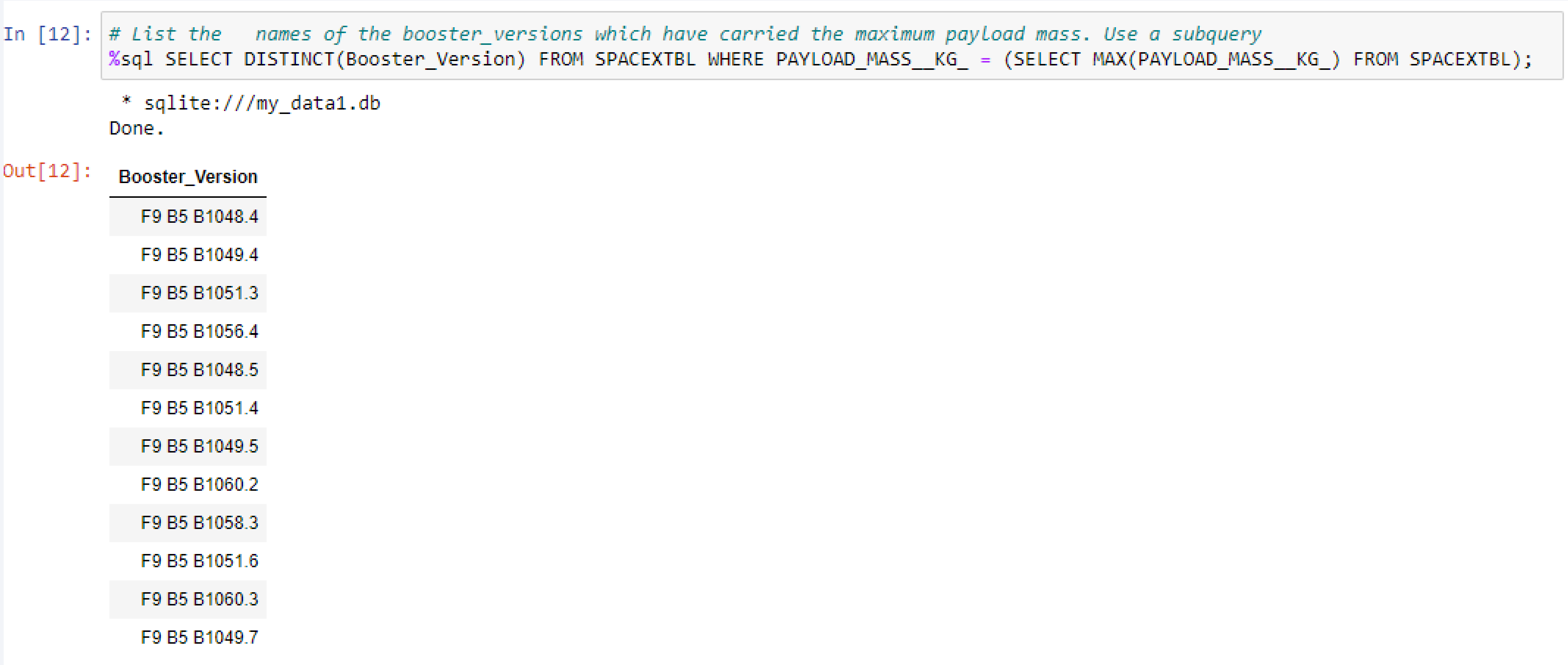
```
In [11]: %sql SELECT (SELECT COUNT(Mission_Outcome) FROM SPACEXTBL WHERE Mission_Outcome = 'Success') AS Success,\
             (SELECT COUNT(Mission_Outcome) FROM SPACEXTBL WHERE Mission_Outcome Like '%Failure%') AS Failure;

          * sqlite:///my_data1.db
         Done.
```

Out[11]:

Success	Failure
98	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass using subquery and MAX.



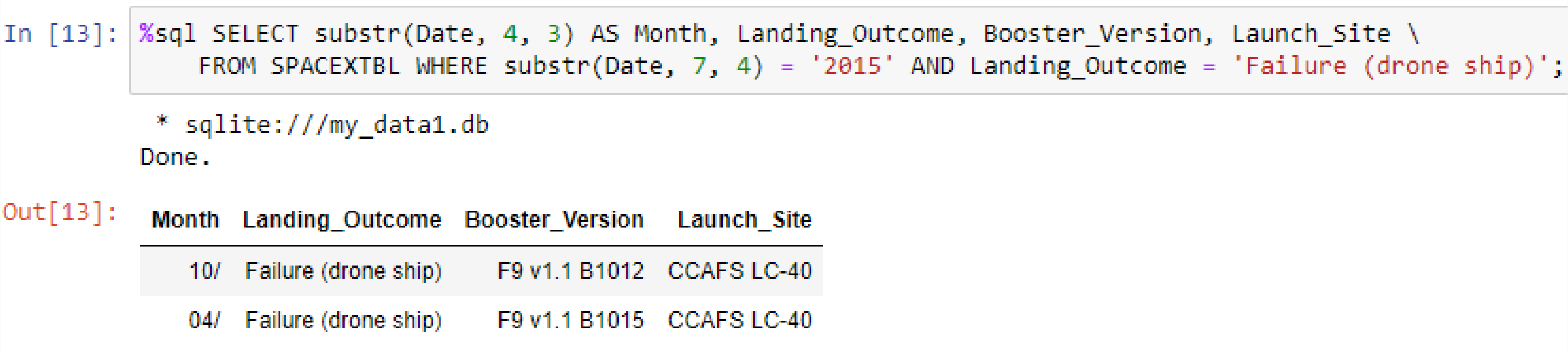
```
In [12]: # List the   names of the booster_versions which have carried the maximum payload mass. Use a subquery
         %sql SELECT DISTINCT(Booster_Version) FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);

          * sqlite:///my_data1.db
         Done.
```

Out[12]:

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015 using substr to compare dates as sqlite does not support dates.



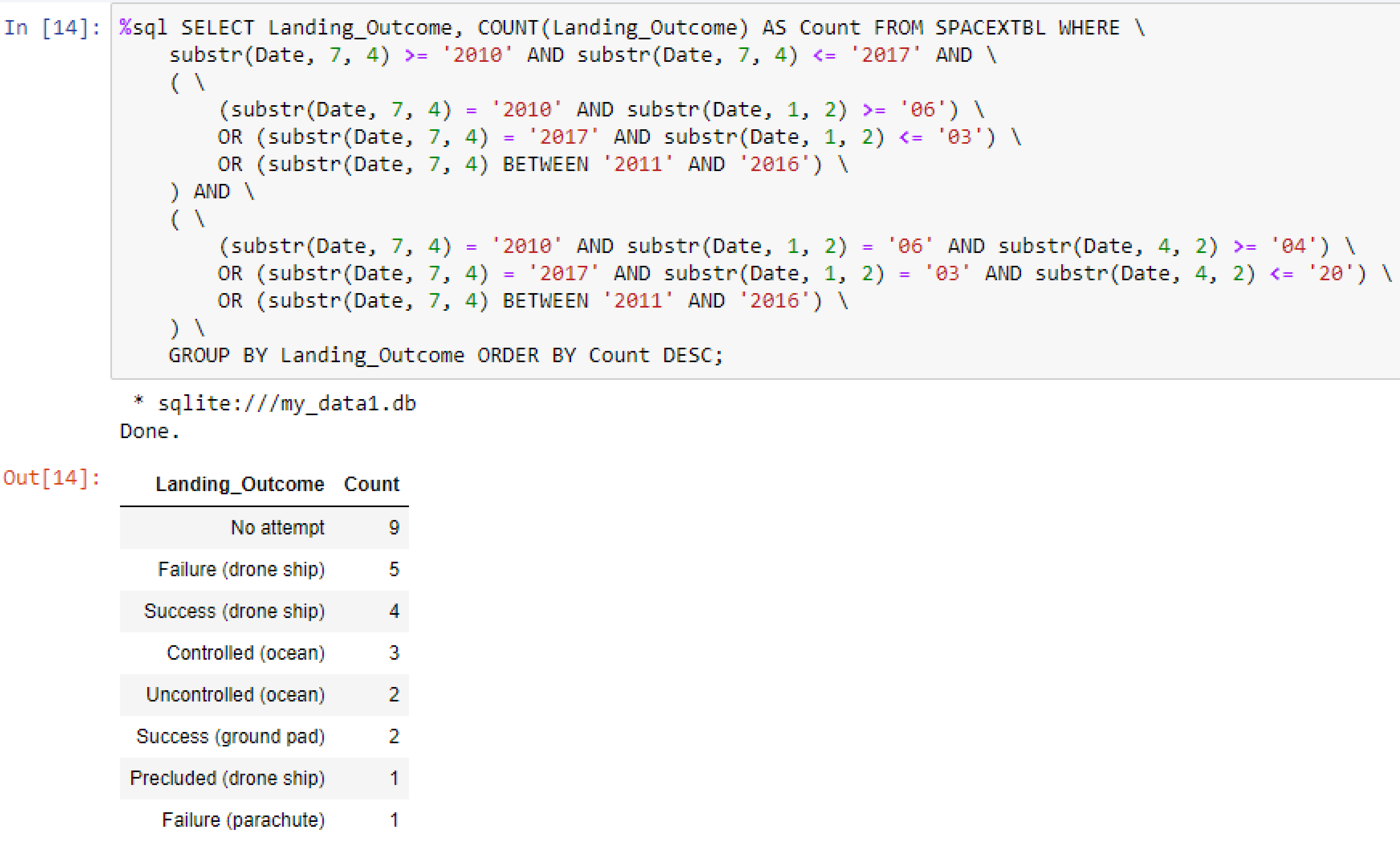
```
In [13]: %sql SELECT substr(Date, 4, 3) AS Month, Landing_Outcome, Booster_Version, Launch_Site \
             FROM SPACEXTBL WHERE substr(Date, 7, 4) = '2015' AND Landing_Outcome = 'Failure (drone ship)';

          * sqlite:///my_data1.db
         Done.
```

Out[13]:

Month	Landing_Outcome	Booster_Version	Launch_Site
10/	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04/	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) using GROUP BY and between the date 2010-06-04 and 2017-03-20, in descending order using ORDER BY.



```
In [14]: %sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS Count FROM SPACEXTBL WHERE \
             substr(Date, 7, 4) >= '2010' AND substr(Date, 7, 4) <= '2017' AND \
             ( \
                 (substr(Date, 7, 4) = '2010' AND substr(Date, 1, 2) >= '06') \
                 OR (substr(Date, 7, 4) = '2017' AND substr(Date, 1, 2) <= '03') \
                 OR (substr(Date, 7, 4) BETWEEN '2011' AND '2016') \
             ) AND \
             ( \
                 (substr(Date, 7, 4) = '2010' AND substr(Date, 1, 2) = '06' AND substr(Date, 4, 2) >= '04') \
                 OR (substr(Date, 7, 4) = '2017' AND substr(Date, 1, 2) = '03' AND substr(Date, 4, 2) <= '20') \
                 OR (substr(Date, 7, 4) BETWEEN '2011' AND '2016') \
             ) \
             GROUP BY Landing_Outcome ORDER BY Count DESC;

          * sqlite:///my_data1.db
         Done.

Out[14]:
```

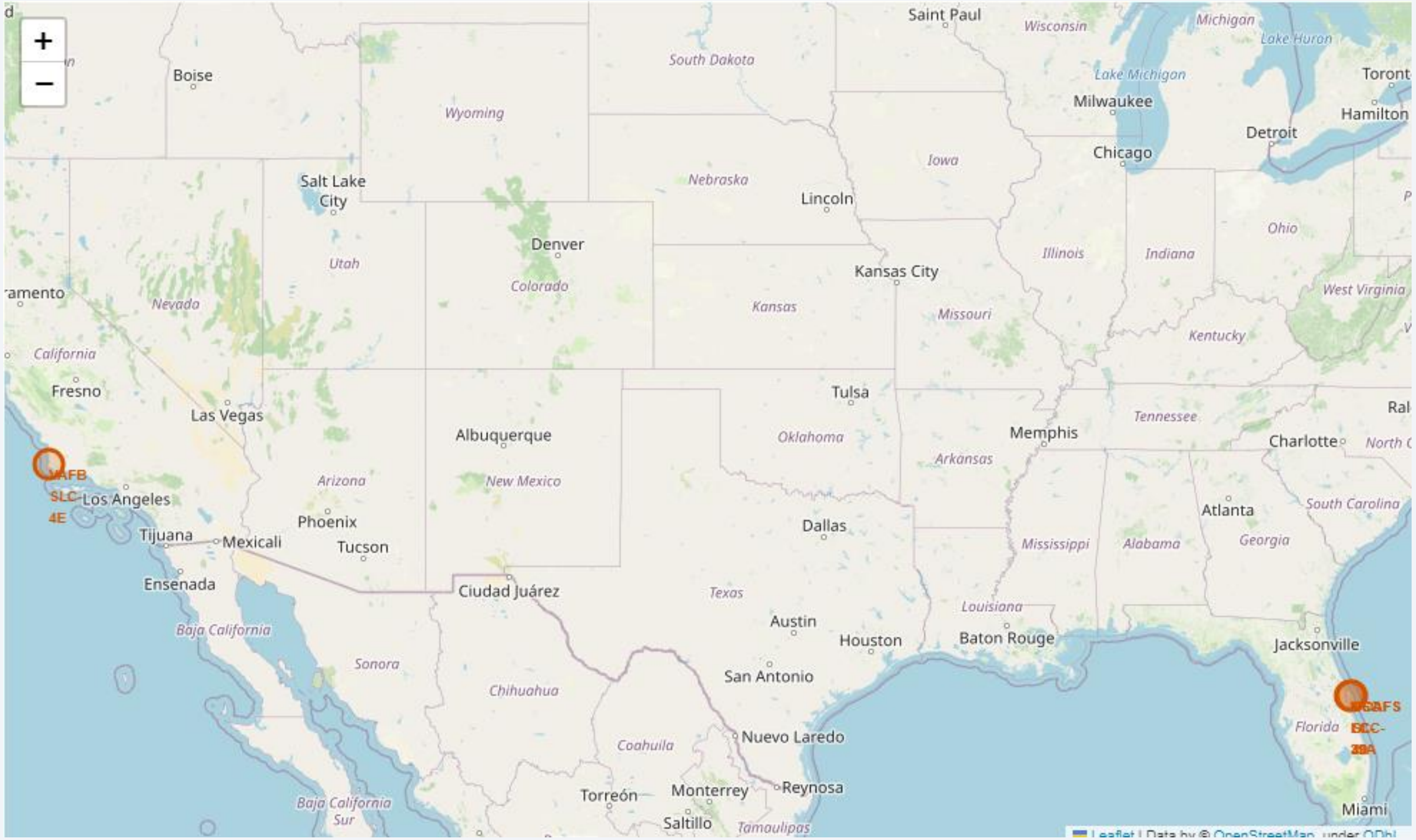
Landing_Outcome	Count
No attempt	9
Failure (drone ship)	5
Success (drone ship)	4
Controlled (ocean)	3
Uncontrolled (ocean)	2
Success (ground pad)	2
Precluded (drone ship)	1
Failure (parachute)	1

Section 3

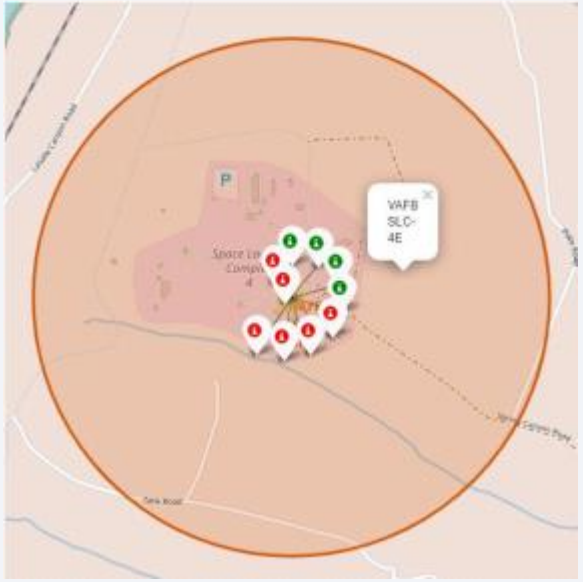
Launch Sites Proximities Analysis

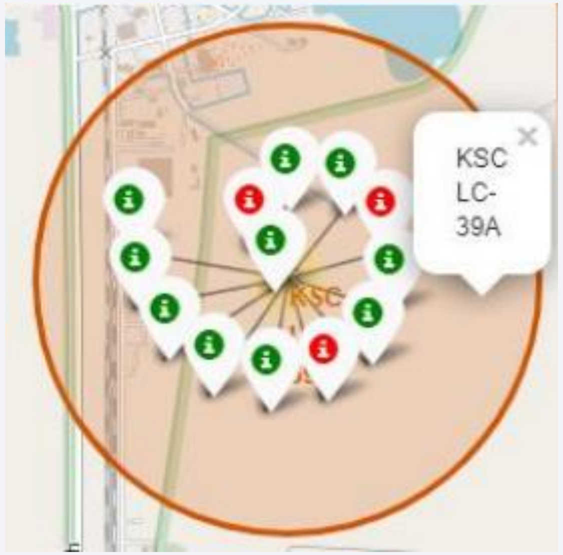
Folium Map – Ground Stations

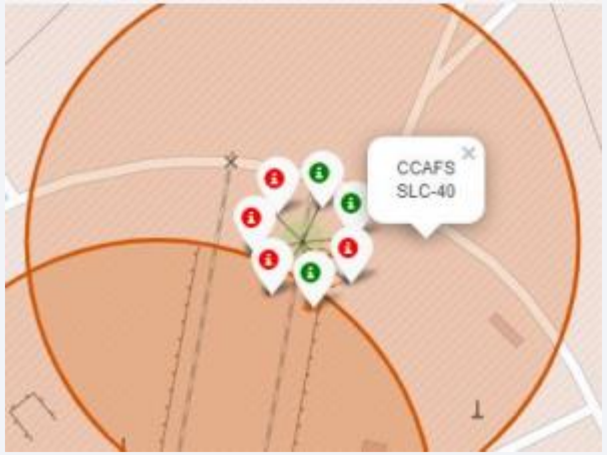
- From this Folium map we can clearly see that launch sites for SpaceX Falcon 9’s rocket are located on opposite coasts of USA.

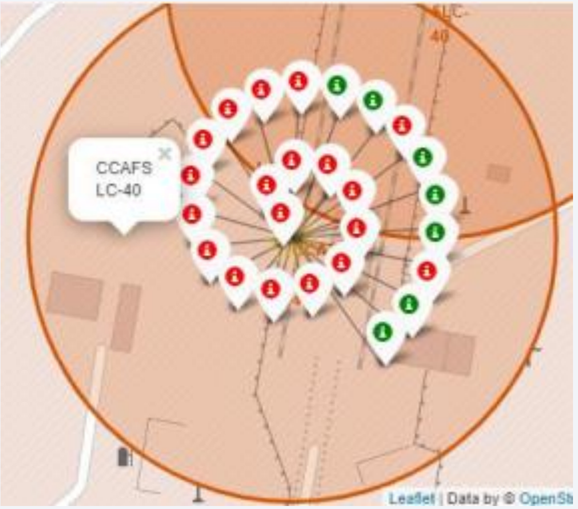


Folium map – Color Labeled Markers



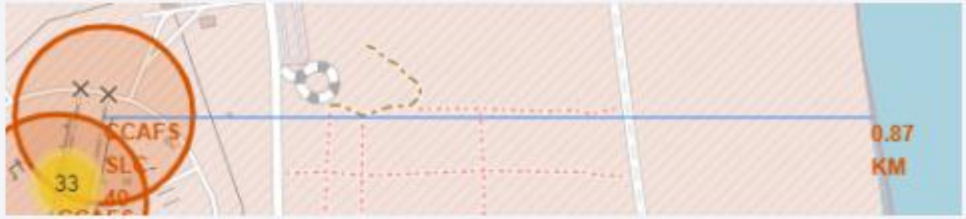


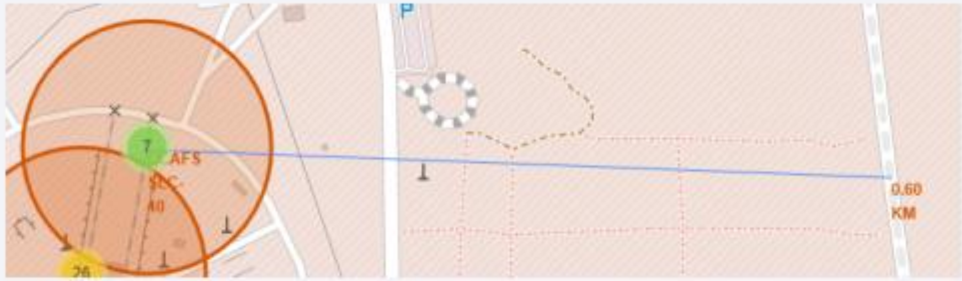


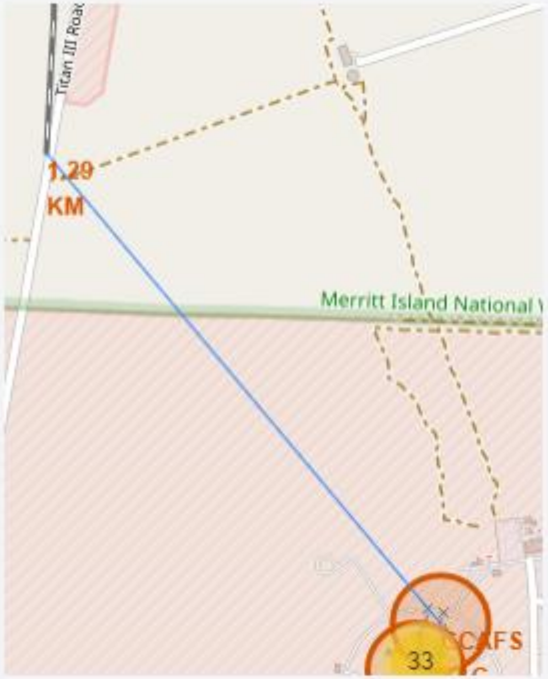


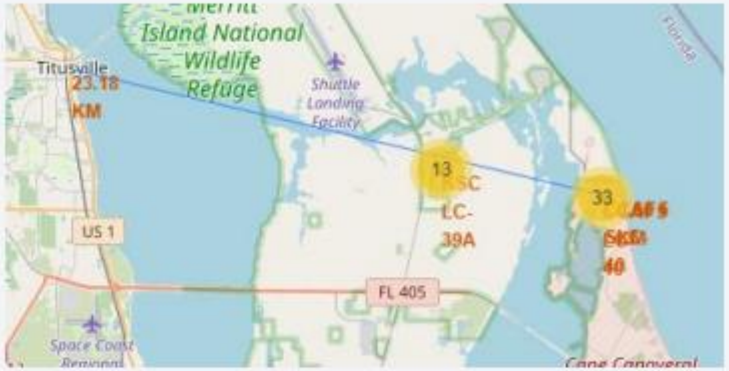
- The Red marker shows failed launch attempt and the Green shows successful launches.
- We can see that the launch site KSC LC-39A has more successful launch ratio.

Folium Map – Distances between CCAFS SLC-40 and its proximities





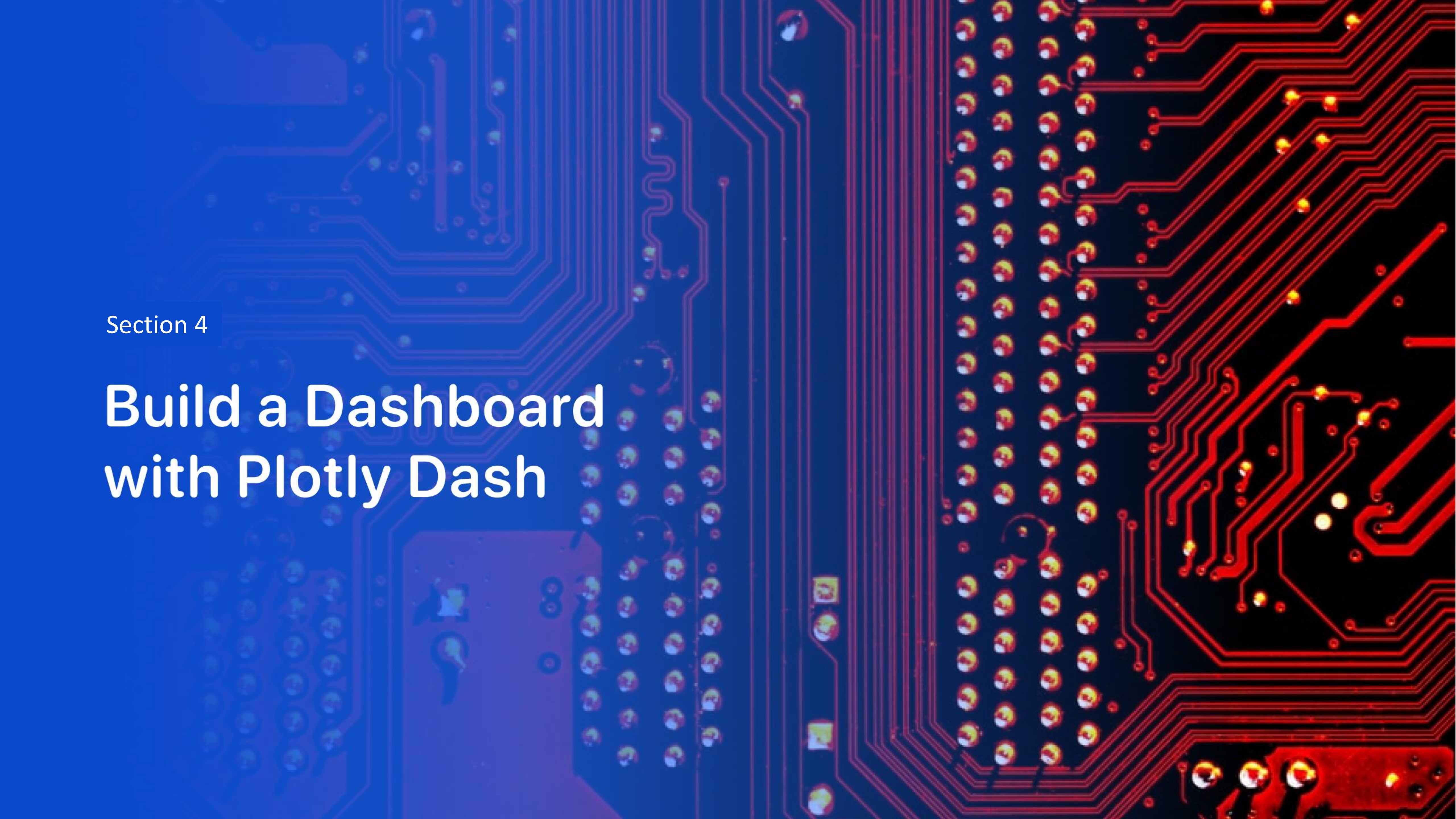




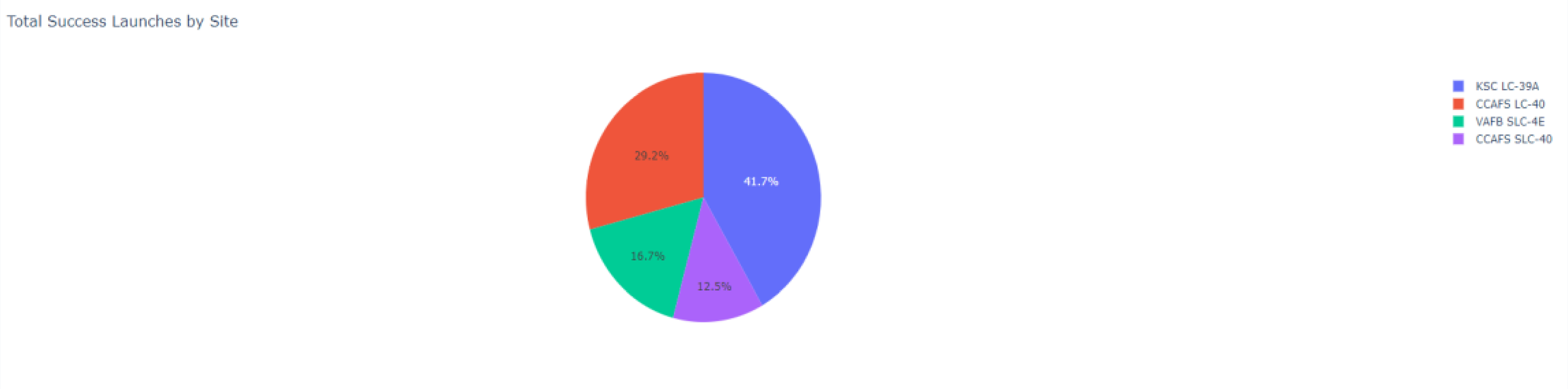
- Is CCAFS SLC-40 in close proximity to railways ? Yes
- Is CCAFS SLC-40 in close proximity to highways ? Yes
- Is CCAFS SLC-40 in close proximity to coastline ? Yes
- Do CCAFS SLC-40 keeps certain distance away from cities ? No

Section 4

Build a Dashboard with Plotly Dash

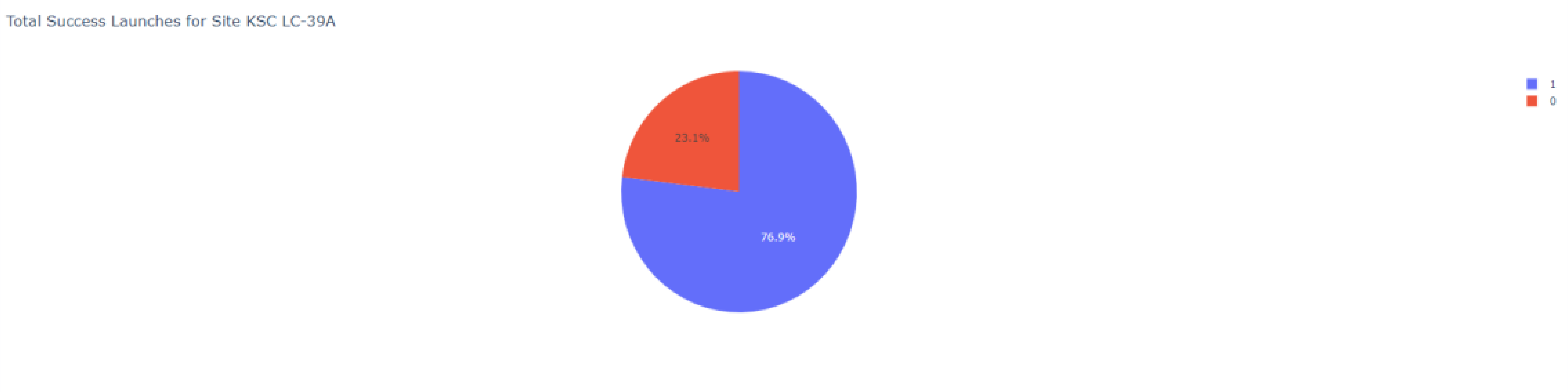


Total success by Site



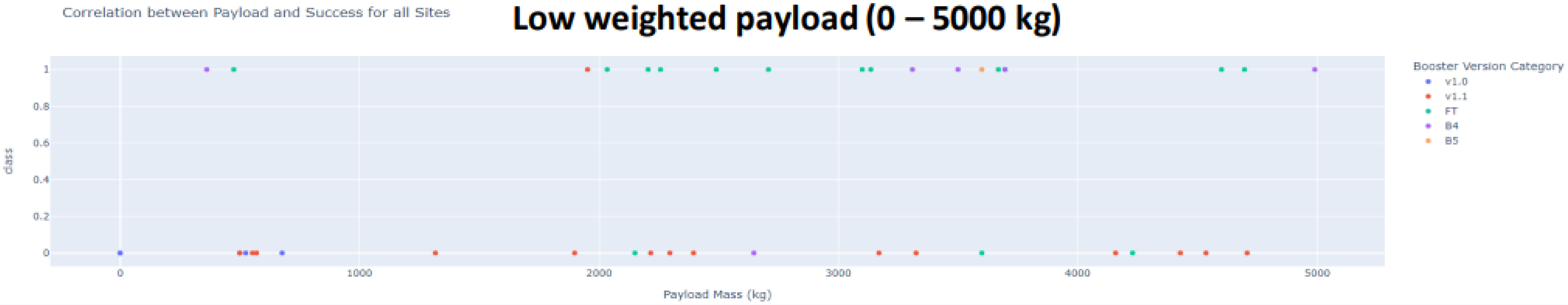
- We see that KSC LC-39A has the best success rate in launches.

Total success launches for Site KSC LC-39A

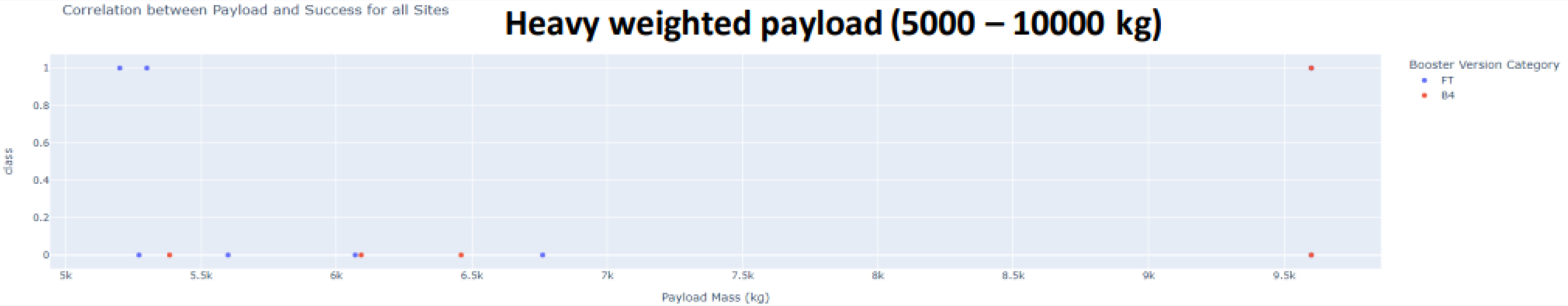


- We can see that the success rate for launch site KSC LC-39A is 76.9% higher than other launch sites.

Payload mass vs Outcome for all sites with different payload mass selected



Low weighted payload (0 – 5000 kg)



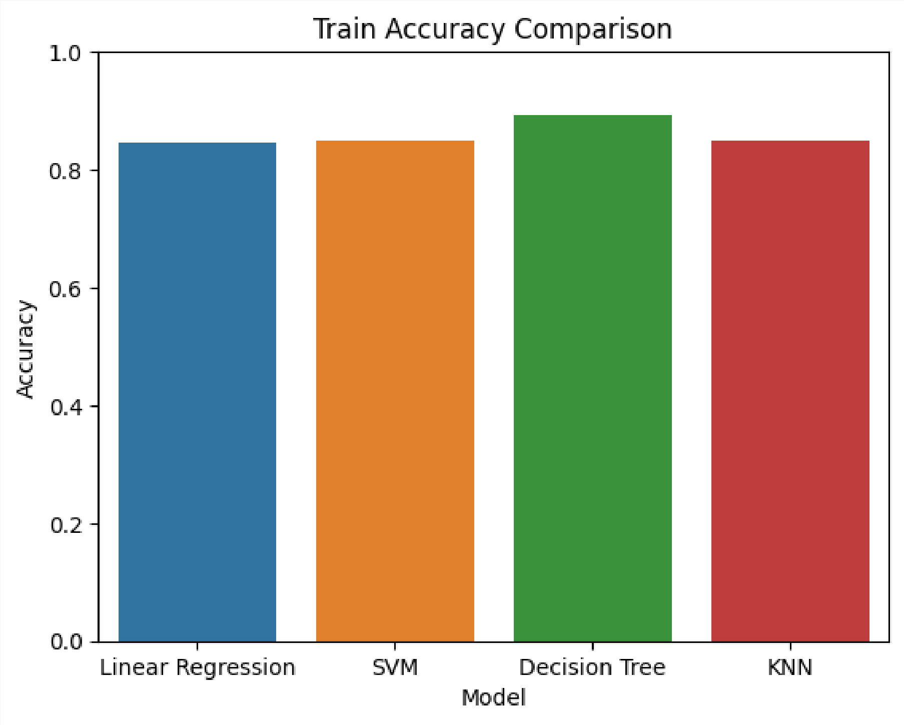
Heavy weighted payload (5000 – 10000 kg)

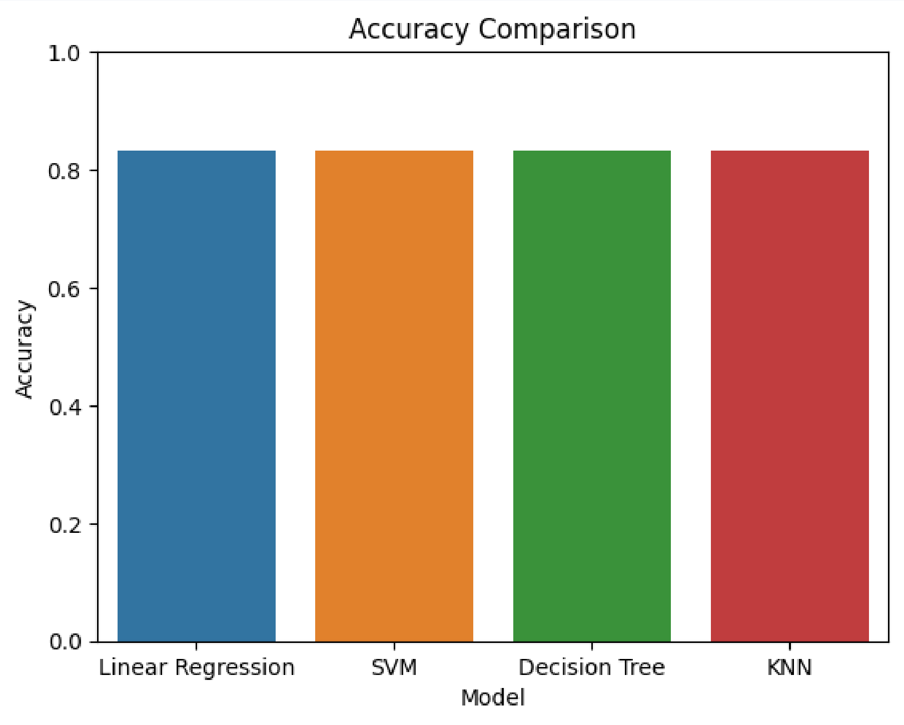
- The success rate for most booster version is high in low payloads as compared to heavy payloads.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

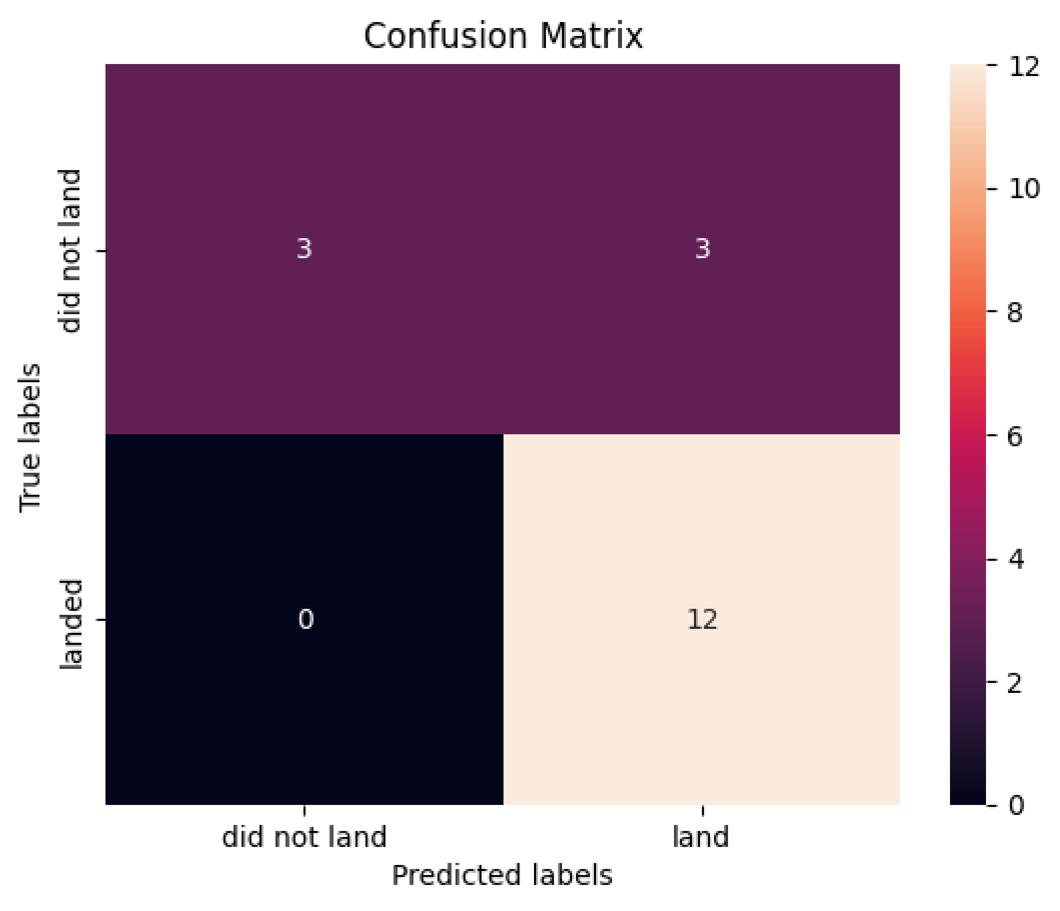




- From the visualization we can conclude that all the models have the same accuracy score in test dataset as any other model. But in train data accuracy score Decision Trees have higher accuracy.

Confusion Matrix

- As the accuracy score is similar across all algorithms so the confusion matrix will also be similar.
- The problem here are the false positive.



Conclusions

Mission success can be attributed to various factors, including the launch site, the orbit, and notably, the number of previous launches. It is reasonable to assume that the accumulation of knowledge between launches contributes to the transition from launch failures to successful missions.

- Among the orbits considered, GEO, HEO, SSO, and ES-L1 have shown the highest success rates.
- Depending on the specific orbit, the payload mass may significantly impact the mission's success. Certain orbits necessitate lighter or heavier payload masses. Generally, missions with lighter payloads tend to perform better than those with heavier payloads.
- Presently, there is no clear explanation as to why certain launch sites outperform others. For instance, KSC LC-39A is recognized as the most successful launch site. To gain insights into this matter, it may be valuable to acquire atmospheric or other relevant data.
- After analyzing the dataset, we determined the Decision Tree Algorithm to be the most suitable model, despite all models having identical test accuracy. The Decision Tree Algorithm was preferred due to its superior train accuracy.

Appendix

- Notebooks and all of the relevant dataset can found in this github repository: https://github.com/justA-Noobdev/data-science-capstone.git

Thank you!