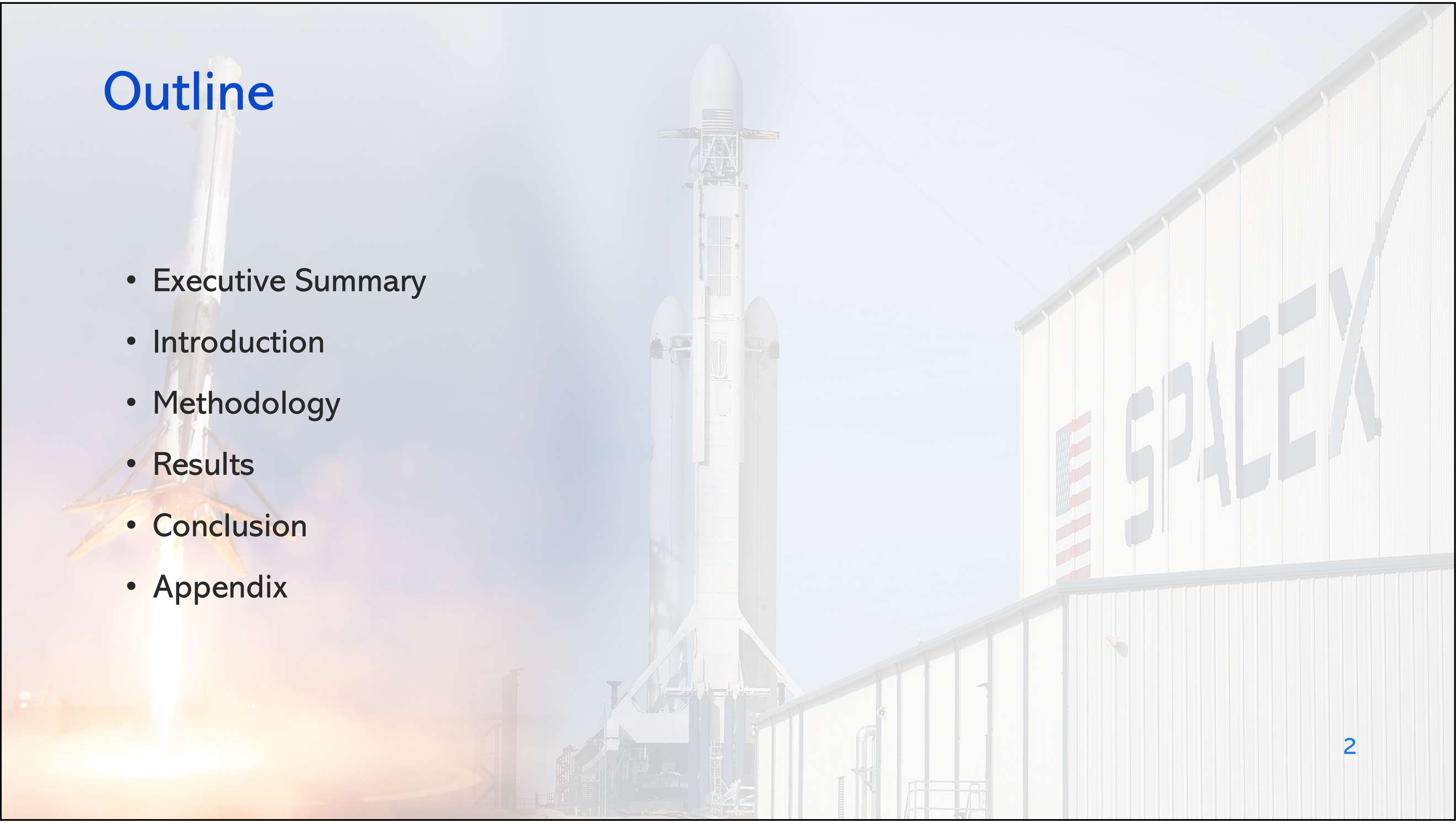
WINNING THE SPACE RACE WITH DATA SCIENCE

Simon Hunt
12/08/2022



Outline

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

Executive Summary

- Summary of methodologies

❖ Data Collection via API

❖ Data Collection using Web Scraping

❖ Data Wrangling

❖ Exploratory Data Analysis with SQL

❖ Exploratory Data Analysis including Data Visualization

❖ Interactive Visual Analytics via Folium

❖ Predictions with Machine Learning

- Summary of all results

❖ Results of Exploratory Data Analysis

❖ Screenshots of Interactive Analytics

❖ Results of Predictive Analytics

Introduction

- Project background and context

SpaceX was founded in 2002 by Elon Musk. Elon Musk has proven to be very influential across a number of different technical-related areas since this time. SpaceX was formed with the vision that commercial flight could become obtainable within the not-too-distant future. In 2022, it is reported that a Falcon 9 cost stands ay $97 million dollars. This is a fraction of the cost that would be required to launch a Nasa rocket for example. One of the key reasons for the cost saving is due to the fact that the rocket can reuse the first stage. The goal of this project is to produce a machine learning pipeline that will predict the outcome of the landing of the first stage in the future.

Problems you want to find answers

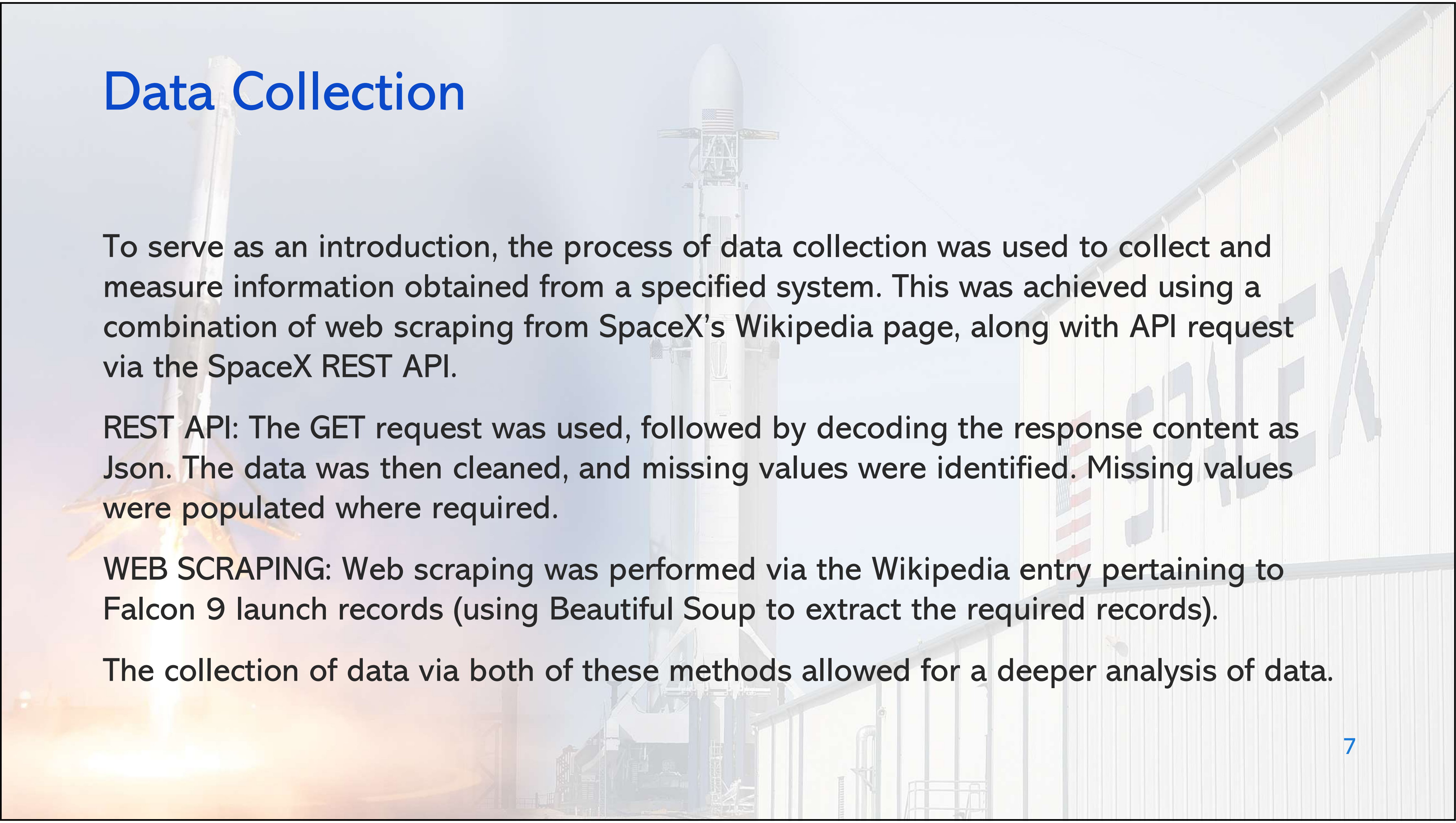
- ❖ What are the factors that will determine the successful landing of the rocket?
- ❖ Identifying relationships between various variables and determine how it will affect the outcome.
- ❖ Determining ideal conditions to ensure successful landing.

SECTION 1
METHODOLOGY

Methodology

Executive Summary

- Data collection methodology:
  - In terms of how the data was collected, the SpaceX API was utilised in addition to web scraping Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)
- Perform data wrangling
  - Filtering. One-Hot Encoding used to prepare the data.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, fine tuning and evaluating classification models in order to achieve an optimum outcome.



Data Collection

To serve as an introduction, the process of data collection was used to collect and measure information obtained from a specified system. This was achieved using a combination of web scraping from SpaceX's Wikipedia page, along with API request via the SpaceX REST API.

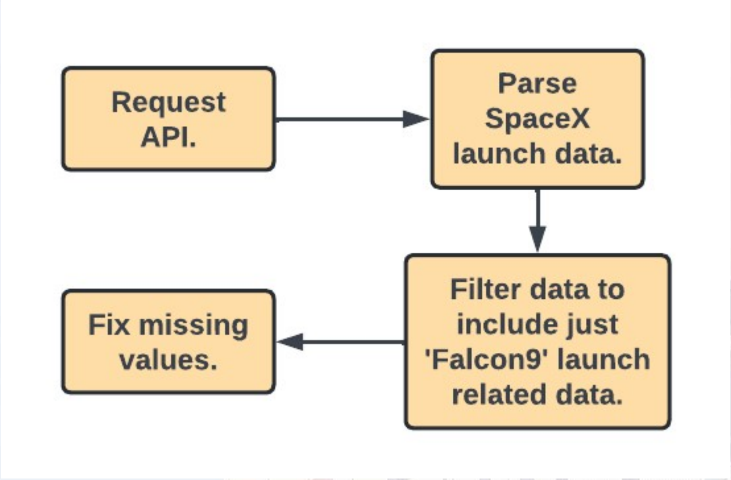
REST API: The GET request was used, followed by decoding the response content as Json. The data was then cleaned, and missing values were identified. Missing values were populated where required.

WEB SCRAPING: Web scraping was performed via the Wikipedia entry pertaining to Falcon 9 launch records (using Beautiful Soup to extract the required records).

The collection of data via both of these methods allowed for a deeper analysis of data.

Data Collection – SpaceX API

- In order to achieve this, the SpaceX API was used to collect the data (this data is publicly available). Please see flow chart to the right. The data was cleaned, and formatting and data wrangling operations were performed.
- GitHub URL: https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/WEEK-1_%20DATA%20COLLECTION%20API.ipynb



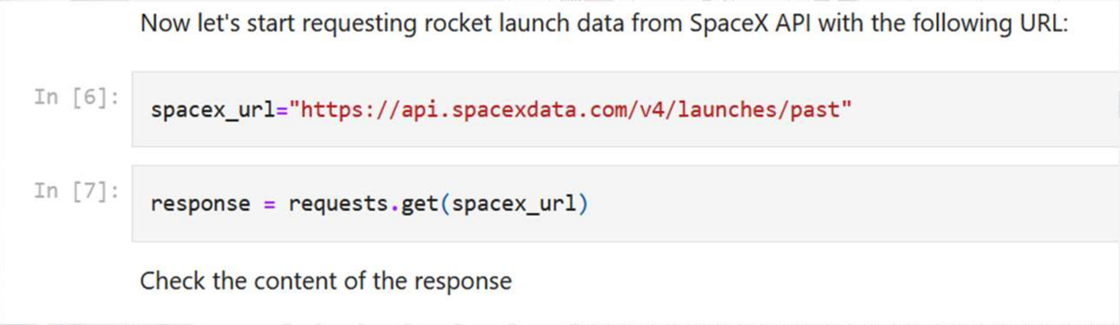
Request API.

Parse SpaceX launch data.

Filter data to include just 'Falcon9' launch related data.

Fix missing values.

Requesting data:



Now let's start requesting rocket launch data from SpaceX API with the following URL:

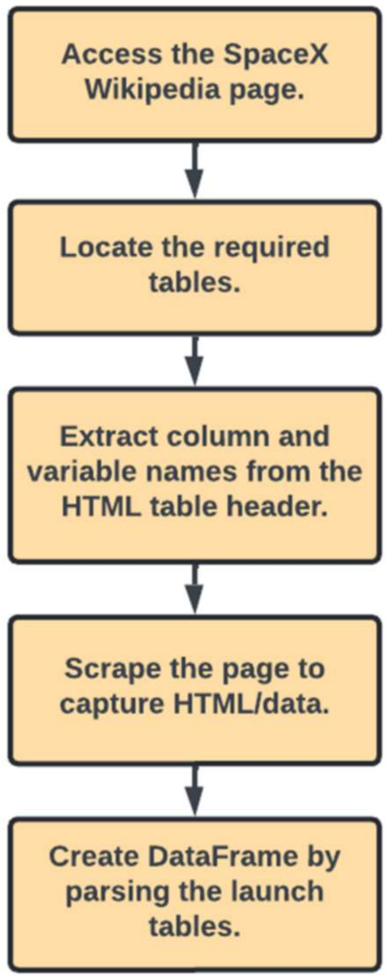
In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

Check the content of the response

Data Collection - Scraping

- Wikipedia was utilised as a source of data in relation to SpaceX launches. The tables from the Wikipedia entry were scraped by utilising BeautifulSoup, and converted into a Pandas DataFrame. The flowchart to the right details the process.
- GitHub URL: https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/WEEK-1_%20DATA%20COLLECTION%20%2B%20WEB%20SCRAPING.ipynb



Access the SpaceX Wikipedia page.

Locate the required tables.

Extract column and variable names from the HTML table header.

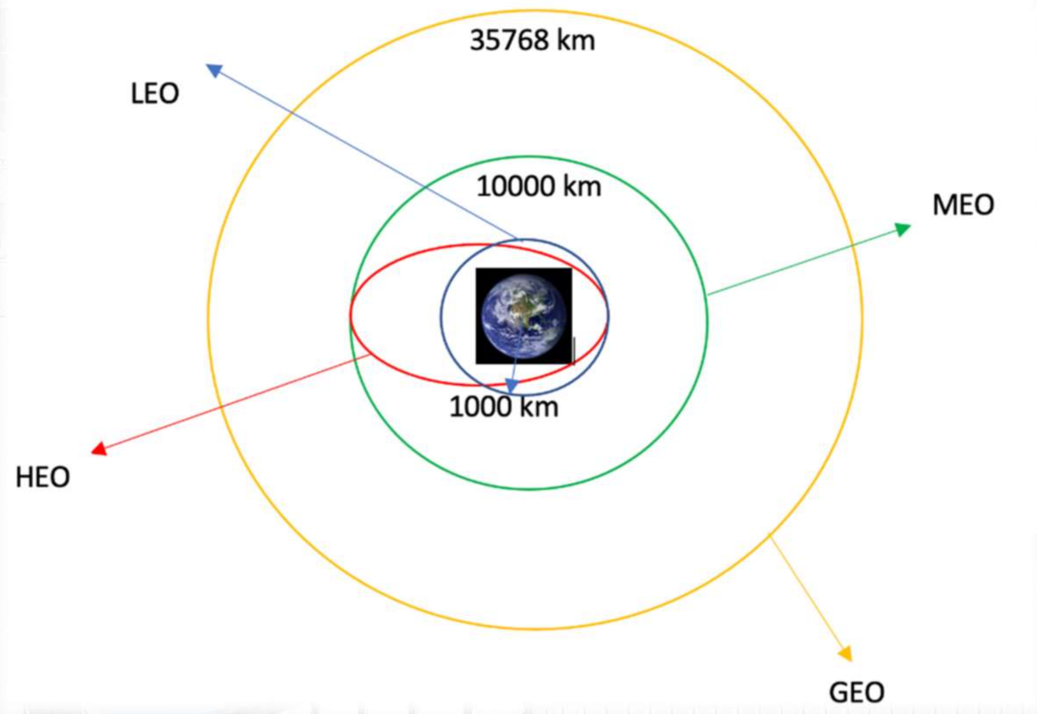
Scrape the page to capture HTML/data.

Create DataFrame by parsing the launch tables.

Data Wrangling

- To begin, Exploratory Data Analysis was performed on the data set. Launch per site summaries, orbit occurrences and mission outcomes per type of orbit were then calculated. The landing outcome label was then created. I then created a landing outcome label and exported the results as a CSV file.

- GitHub URL: https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/WEEK-1_%20DATA%20WRANGLING.ipynb



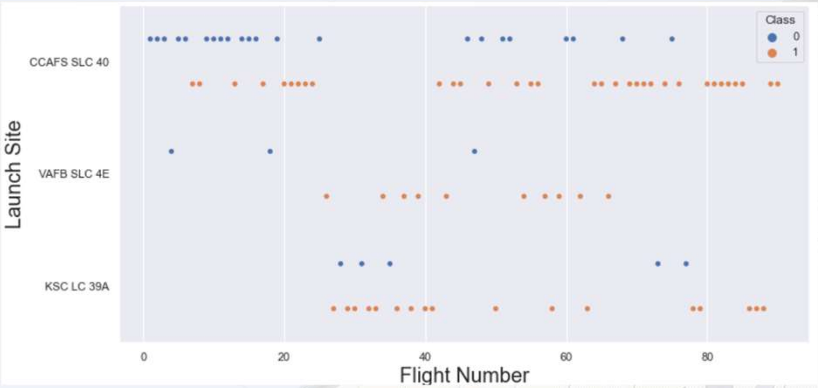
EDA with Data Visualization

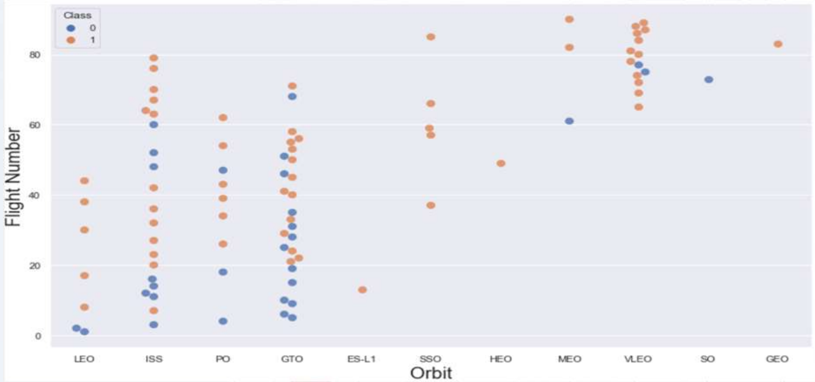
The following charts were created:

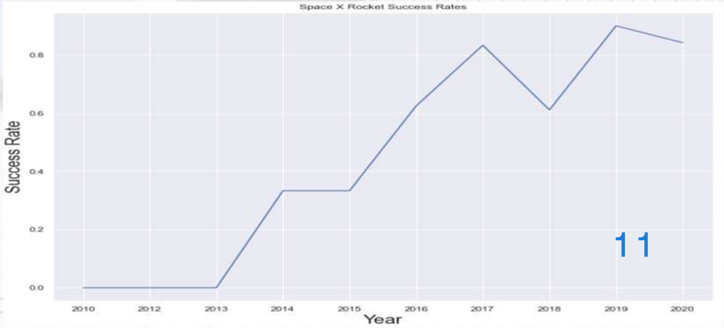
Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit Type vs. Success Rate, Flight Number vs. Orbit Type, Payload Mass vs. Orbit Type, Success Rate Yearly Trend

Scatter plots illustrate the relationships between variables. They are useful for identifying potential correlations, which can be incorporated into machine learning models. Bar charts compare different discrete categories, highlighting the relationships between these categories and specific measured values. Line charts depict trends over time, providing a clear view of how data changes across a timeline.

- GitHub URL: https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/WEEK-2_%20EDA%20WITH%20VISUALISATION.ipynb







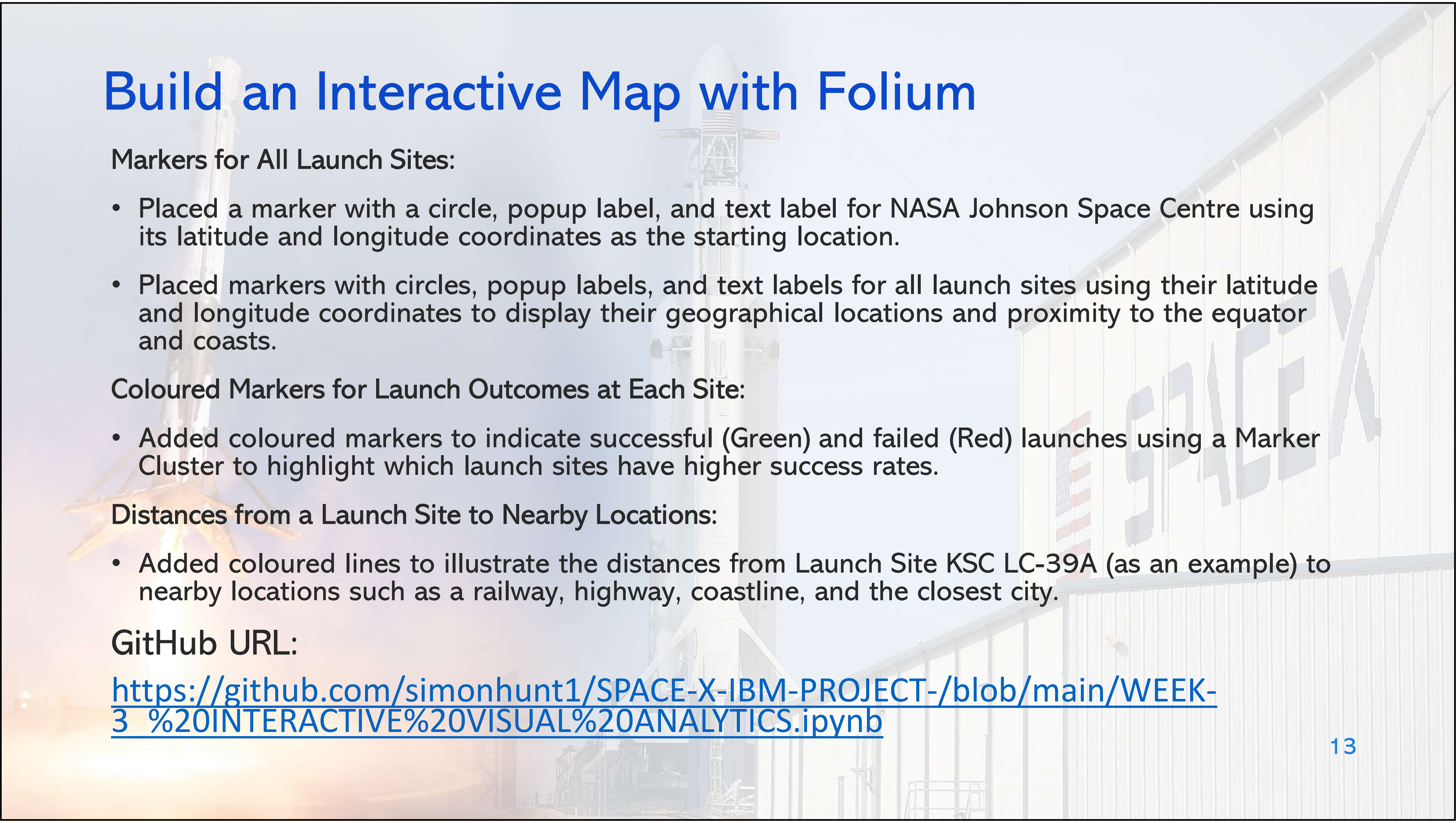
EDA with SQL

Executed SQL queries included:

- Retrieving the unique launch site names from space missions
- Displaying five records with launch sites starting with 'CCA'
- Calculating the total payload mass for boosters launched by NASA (CRS)
- Finding the average payload mass for booster version F9 v1.1
- Identifying the date of the first successful ground pad landing
- Listing boosters that successfully landed on a drone ship with payload masses between 4000 and 6000
- Counting the total number of successful and failed mission outcomes
- Listing the booster versions that carried the maximum payload massReporting the failed drone ship landings, their booster versions, and launch site names for 2015
- Ranking landing outcomes (e.g., Failure on drone ship, Success on ground pad) from 2010-06-04 to 2017-03-20 in descending order

GitHub URL:

https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/WEEK-2_%20DATA%20ANALYSIS%20USING%20SQL.ipynb



Build an Interactive Map with Folium

Markers for All Launch Sites:

- Placed a marker with a circle, popup label, and text label for NASA Johnson Space Centre using its latitude and longitude coordinates as the starting location.
- Placed markers with circles, popup labels, and text labels for all launch sites using their latitude and longitude coordinates to display their geographical locations and proximity to the equator and coasts.

Coloured Markers for Launch Outcomes at Each Site:

- Added coloured markers to indicate successful (Green) and failed (Red) launches using a Marker Cluster to highlight which launch sites have higher success rates.

Distances from a Launch Site to Nearby Locations:

- Added coloured lines to illustrate the distances from Launch Site KSC LC-39A (as an example) to nearby locations such as a railway, highway, coastline, and the closest city.

GitHub URL:

https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/WEEK-3_%20INTERACTIVE%20VISUAL%20ANALYTICS.ipynb

Build a Dashboard with Plotly Dash

Launch Sites Dropdown Menu:

- Implemented a dropdown menu to allow for the selection of different launch sites.

Pie Chart Displaying Successful Launches (All Sites/Specific Site):

- Included a pie chart to illustrate the total count of successful launches across all sites, and the Success vs. Failed counts for a selected launch site.

Payload Mass Range Slider:

- Introduced a slider to adjust the payload mass range.

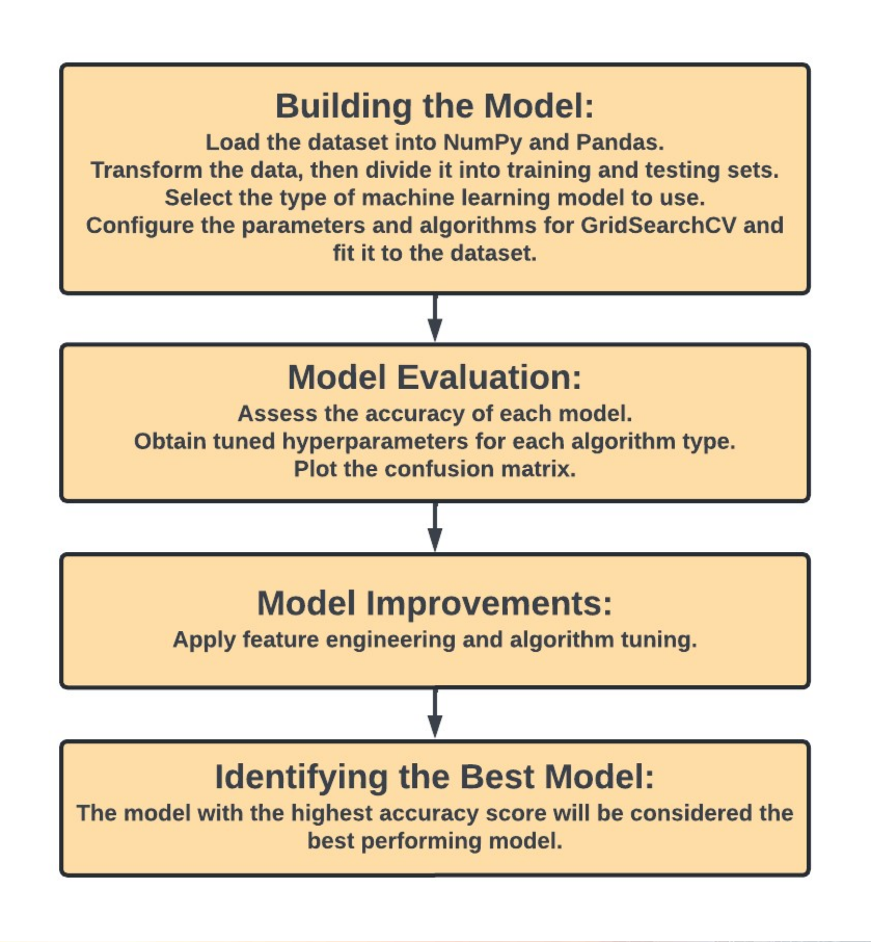
Scatter Plot of Payload Mass vs. Success Rate for Various Booster Versions:

- Created a scatter plot to display the relationship between payload mass and launch success rates across different booster versions.

GitHub URL:

https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/SPACEX%20APP.py

Predictive Analysis (Classification)



Building the Model:
Load the dataset into NumPy and Pandas.
Transform the data, then divide it into training and testing sets.
Select the type of machine learning model to use.
Configure the parameters and algorithms for GridSearchCV and fit it to the dataset.

Model Evaluation:
Assess the accuracy of each model.
Obtain tuned hyperparameters for each algorithm type.
Plot the confusion matrix.

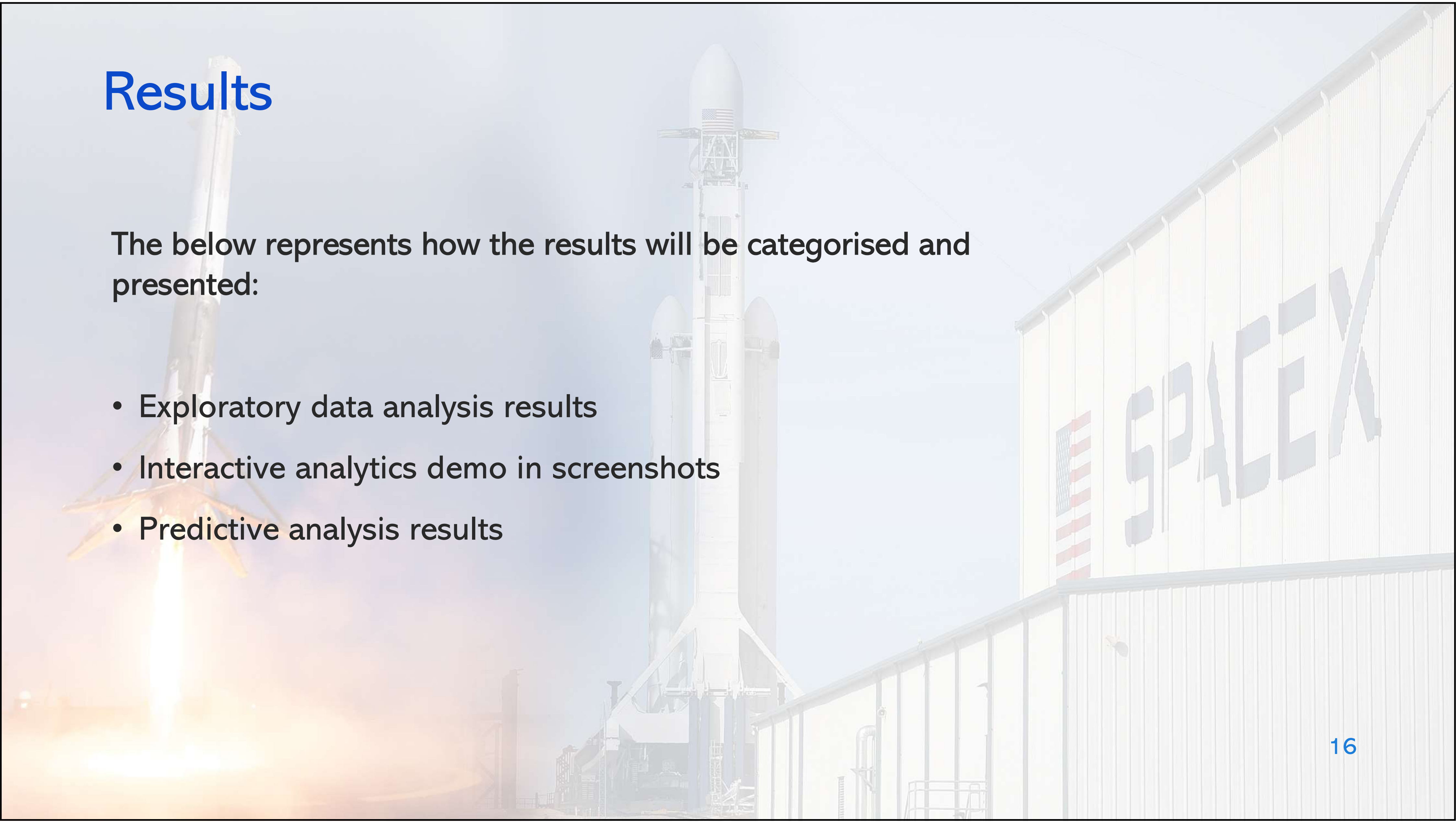
Model Improvements:
Apply feature engineering and algorithm tuning.

Identifying the Best Model:
The model with the highest accuracy score will be considered the best performing model.

GitHub URL:

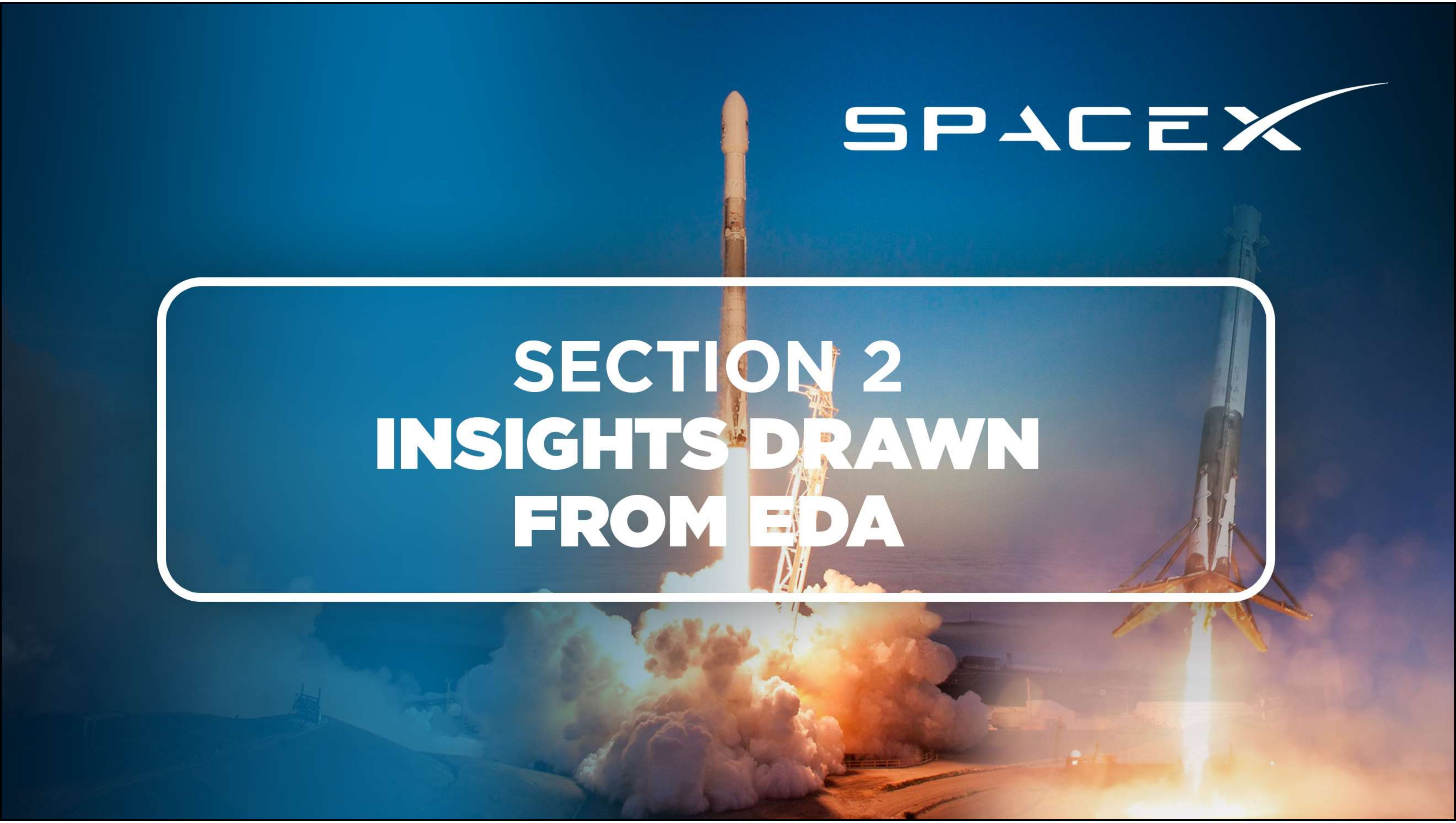
https://github.com/simonhunt1/SPACE-X-IBM-PROJECT-/blob/main/SPACEX%20APP.py

Results



The below represents how the results will be categorised and presented:

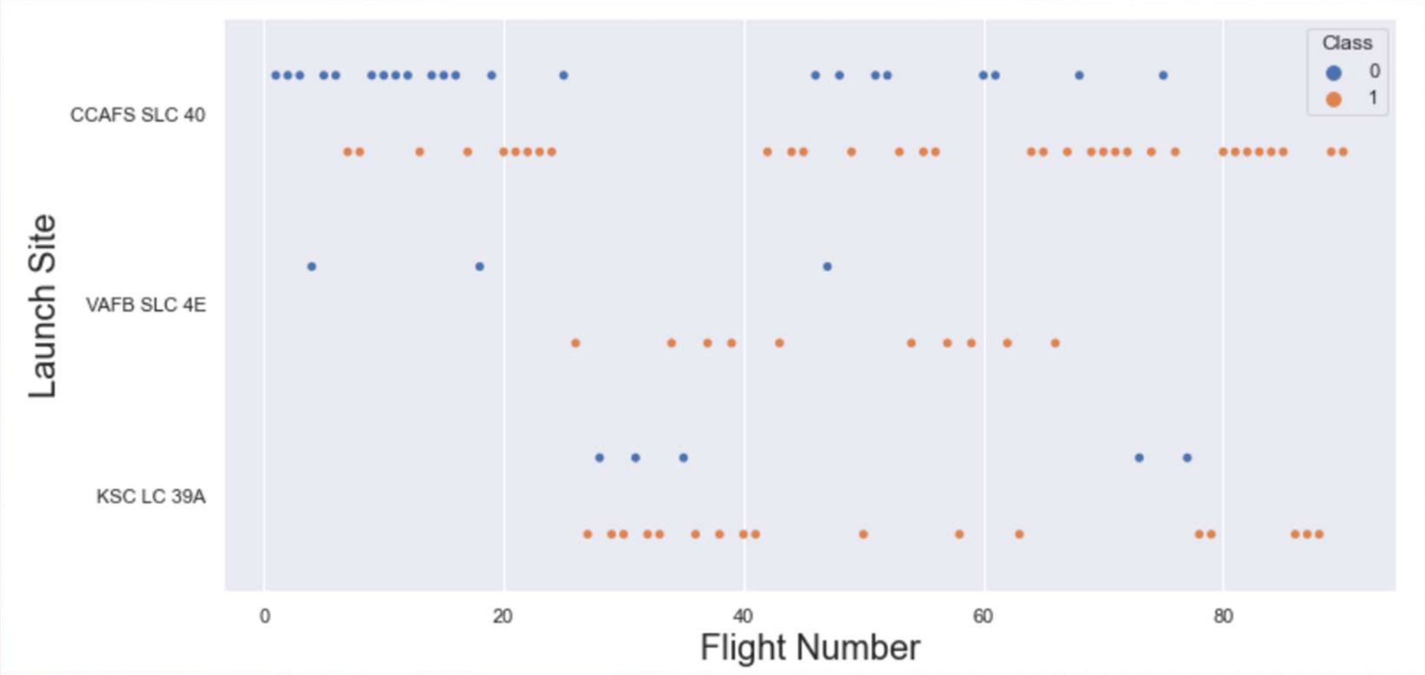
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



SECTION 2
INSIGHTS DRAWN FROM EDA

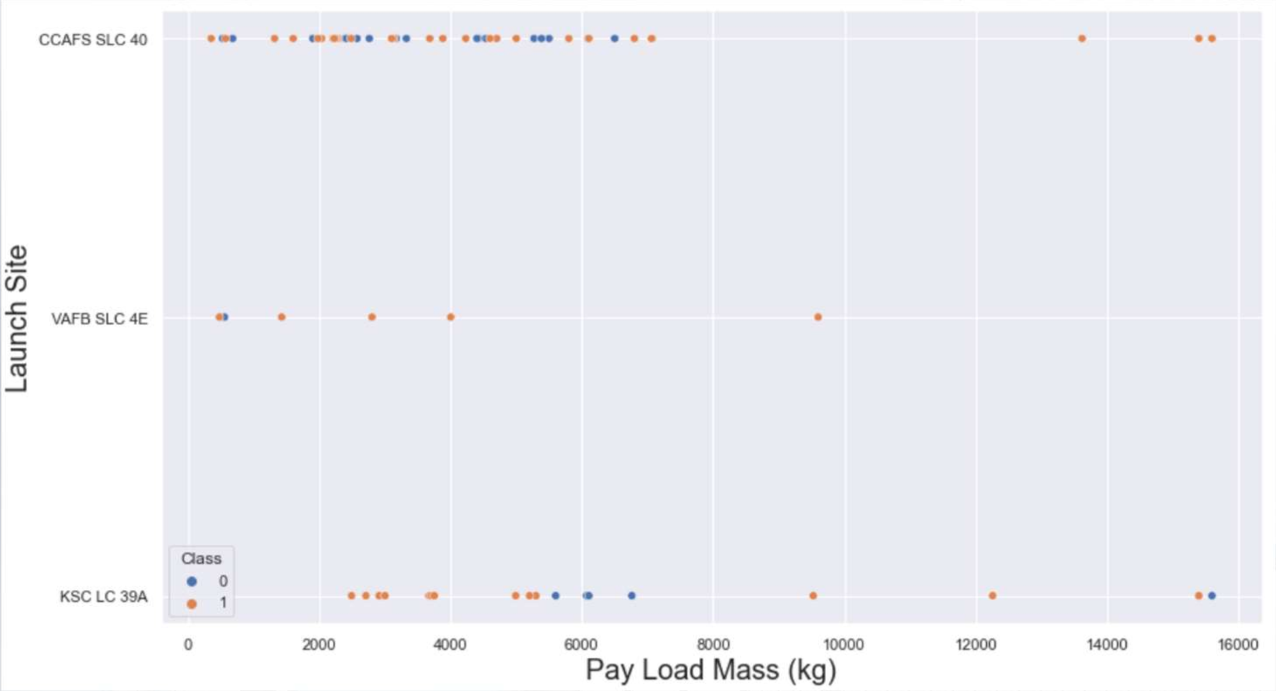
Flight Number vs. Launch Site

This scatter plot indicates a correlation between the number of flights from a launch site and its success rate—the more flights, the higher the success rate. However, the CCAFS SLC40 site deviates from this pattern, showing the weakest correlation.



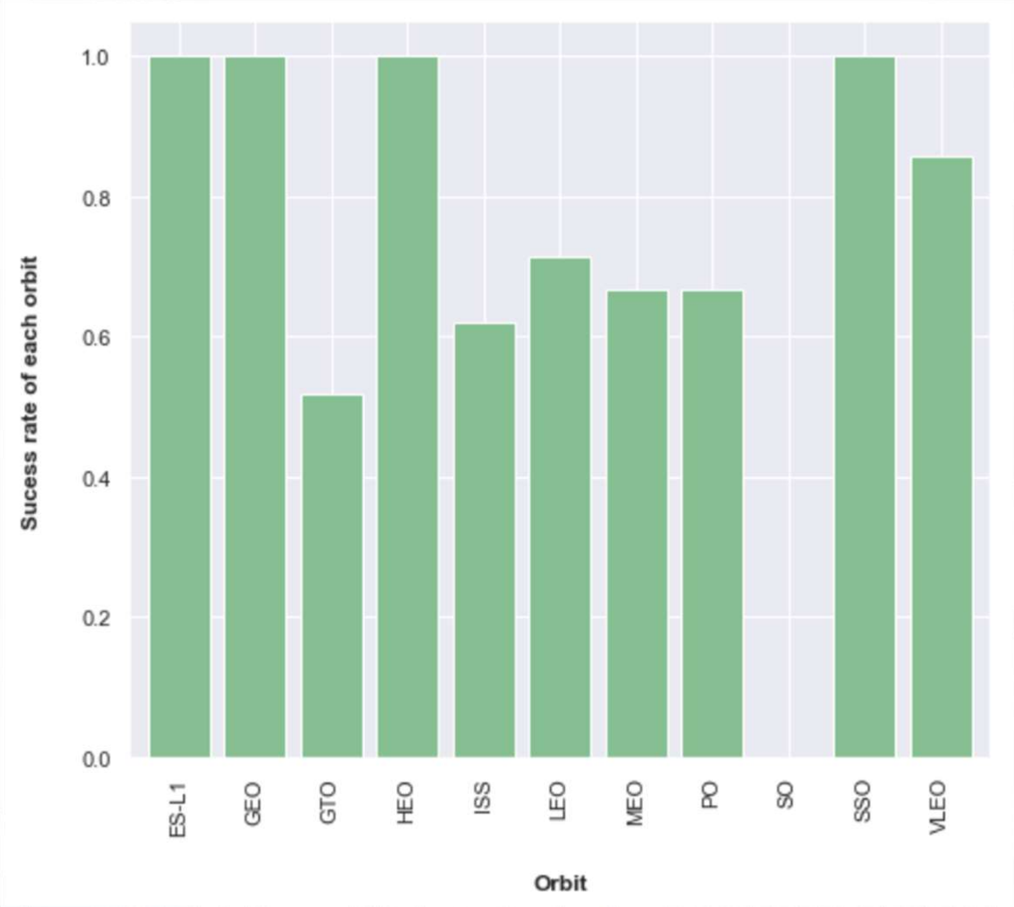
Payload vs. Launch Site

- This scatter plot demonstrates that payloads exceeding 7000kg have a significantly higher probability of successful launches. However, there is no evident pattern suggesting that the success rate is influenced by the launch site based on payload mass.



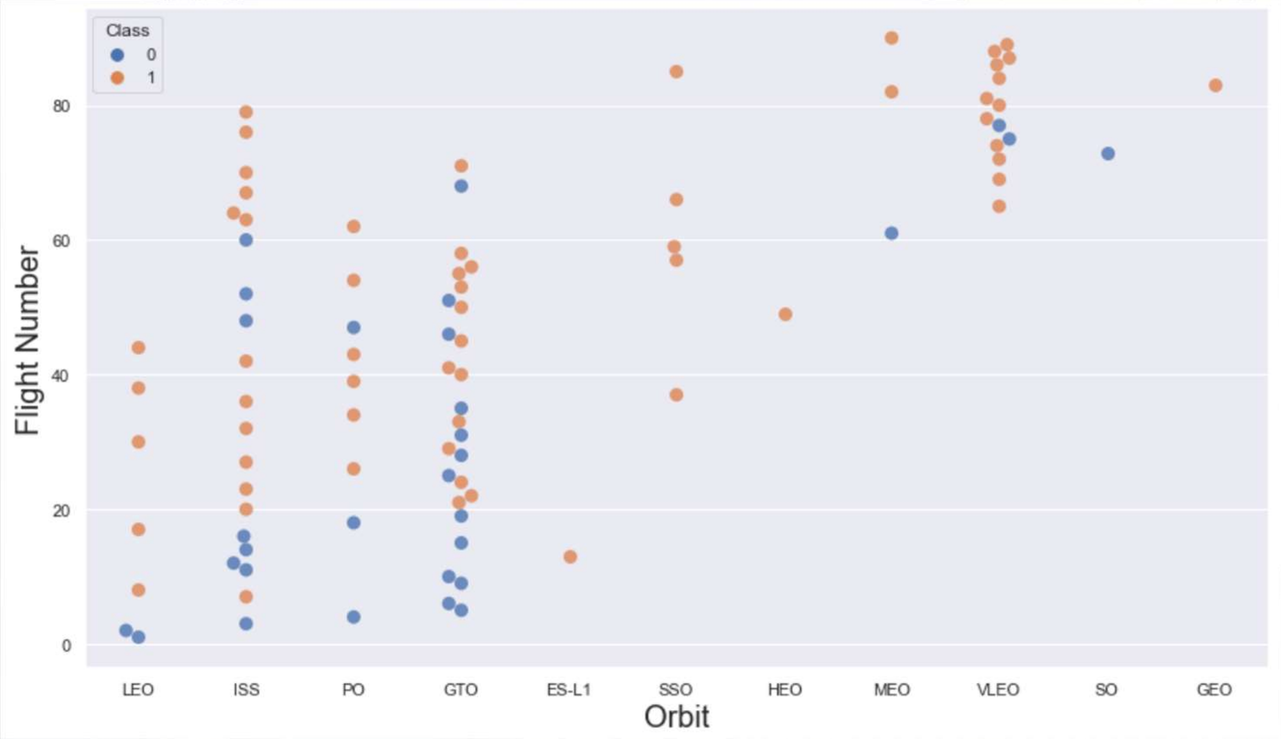
Success Rate vs. Orbit Type

This figure illustrates the impact of different orbits on landing outcomes. Certain orbits, such as SSO, HEO, GEO, and ES-L1, have a 100% success rate, while the SO orbit has a 0% success rate. However, a closer examination reveals that some orbits, like GEO, SO, HEO, and ES-L1, only have a single occurrence. This indicates that more data is required to identify any patterns or trends before drawing definitive conclusions.



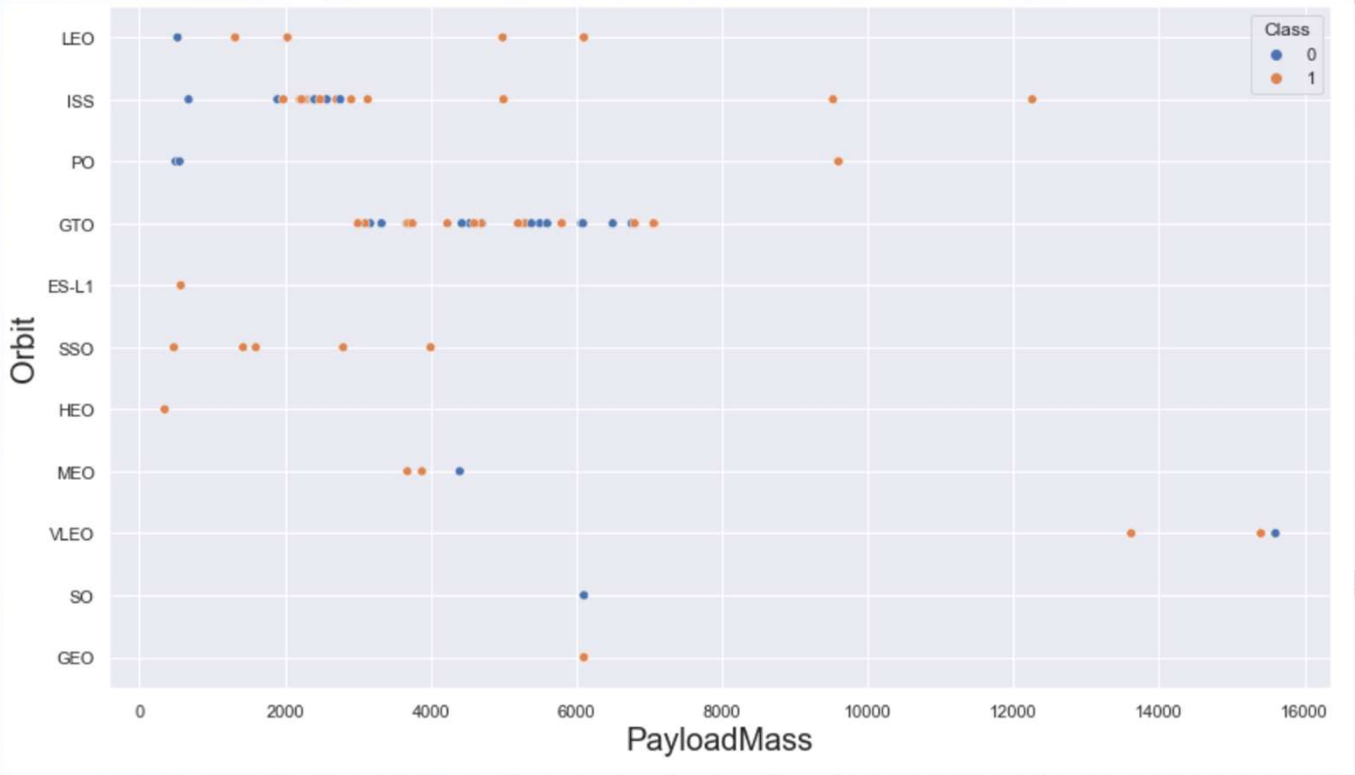
Flight Number vs. Orbit Type

This scatter plot demonstrates that, in general, higher flight numbers correlate with increased success rates for each orbit, particularly for the LEO orbit. However, the GTO orbit shows no discernible relationship between these variables. Orbits with only one occurrence should be excluded from this analysis, as additional data is needed for a conclusive statement.



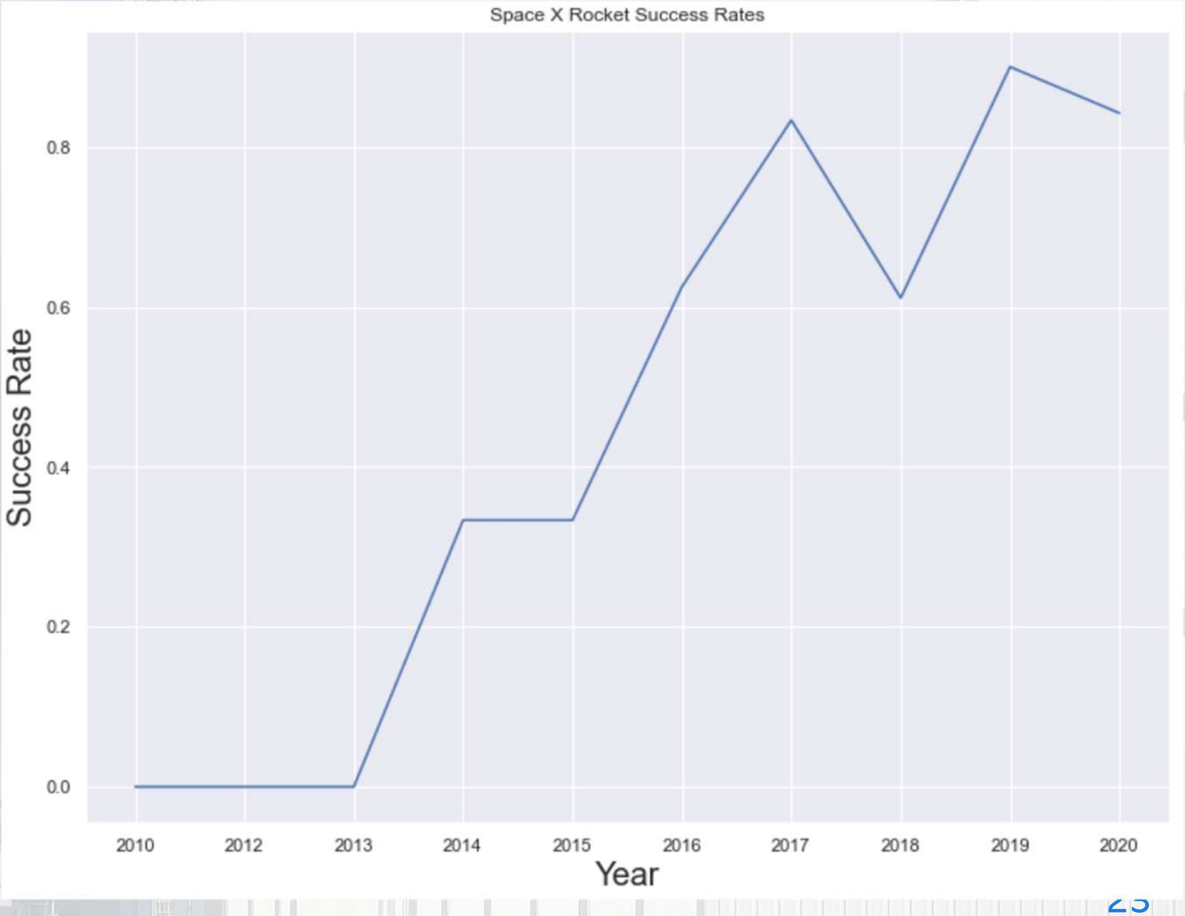
Payload vs. Orbit Type

Heavier payloads positively affect LEO, ISS, and P0 orbits. Conversely, they negatively impact MEO and VLEO orbits. For GTO orbit, no clear relationship between payload and success is evident. Additionally, SO, GEO, and HEO orbits require more data to identify any potential patterns or trends.



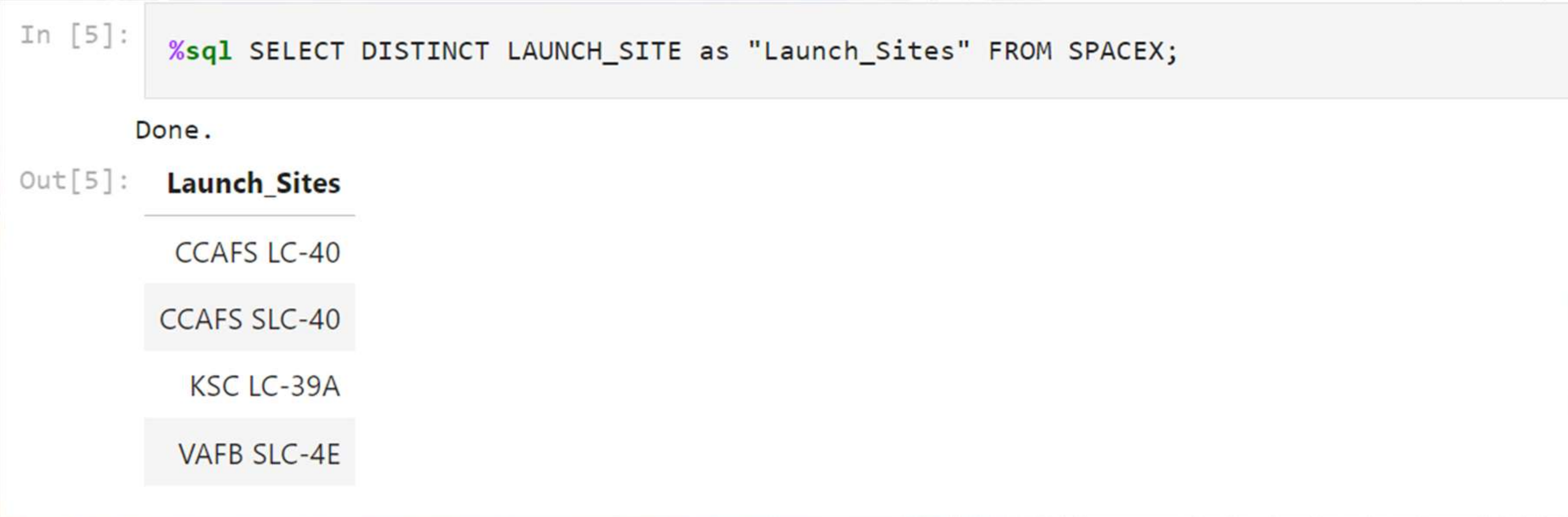
Launch Success Yearly Trend

As per the line graph, these figures clearly show an upward trend from 2013 to 2020.



All Launch Site Names

I utilised the keyword DISTINCT to display only the unique launch sites from the SpaceX data.



In [5]:
```
%sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEX;
```

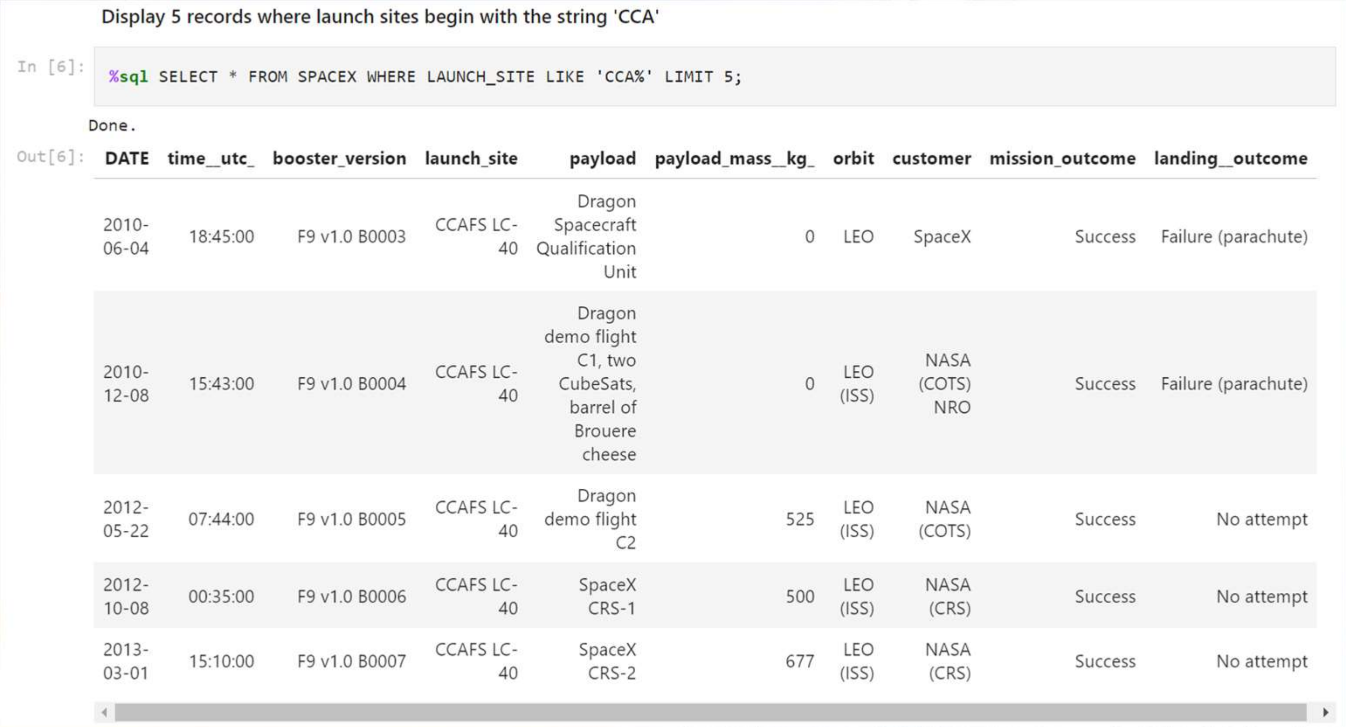
```
Done.
```

Out[5]:

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

Launch Site Names Begin with 'CCA'

- The below query was used to present the 5 records whereby launch sites begin with “CCA”.



Display 5 records where launch sites begin with the string 'CCA'

In [6]:

```
%sql SELECT * FROM SPACEX WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

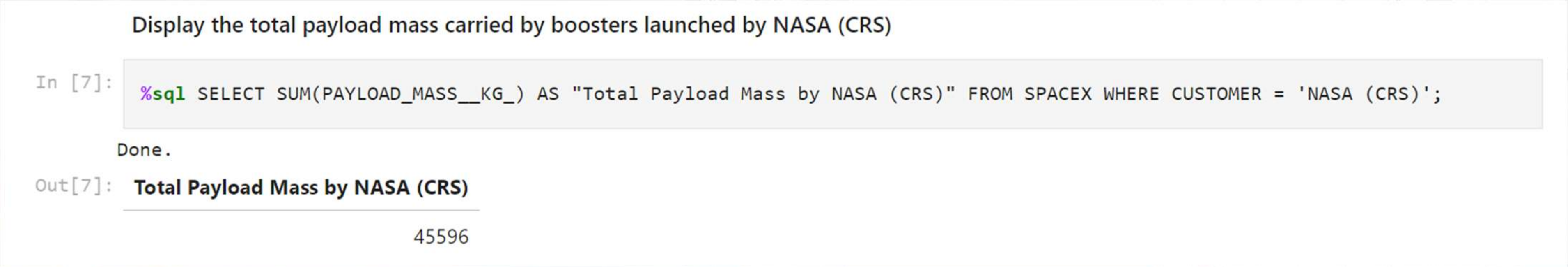
```
Done.
```

Out[6]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

The below query was utilised to calculate the total payload carried by boosters from NASA (45596).



Display the total payload mass carried by boosters launched by NASA (CRS)

```
In [7]: %sql SELECT SUM(PAYLOAD_MASS__KG_) AS "Total Payload Mass by NASA (CRS)" FROM SPACEX WHERE CUSTOMER = 'NASA (CRS)';
```

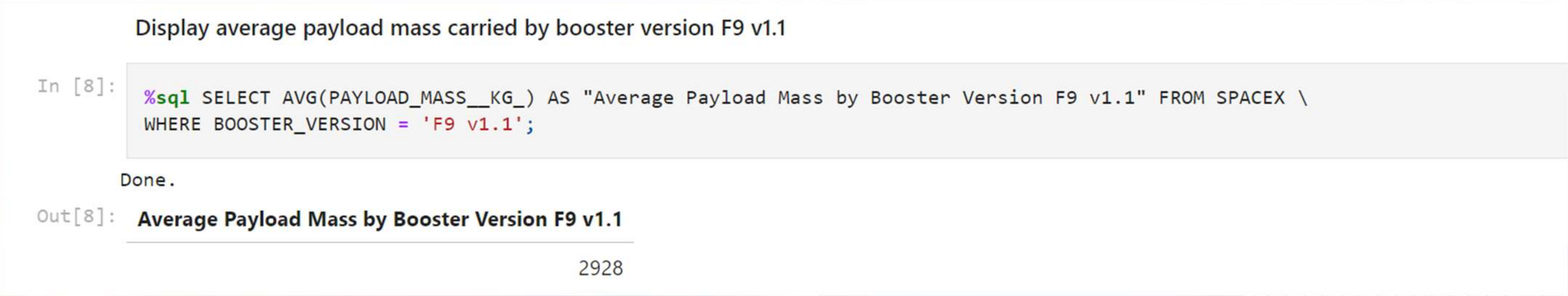
```
Done.
```

Out[7]:

Total Payload Mass by NASA (CRS)
45596

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 v1.1 was calculated to be 2928.4.



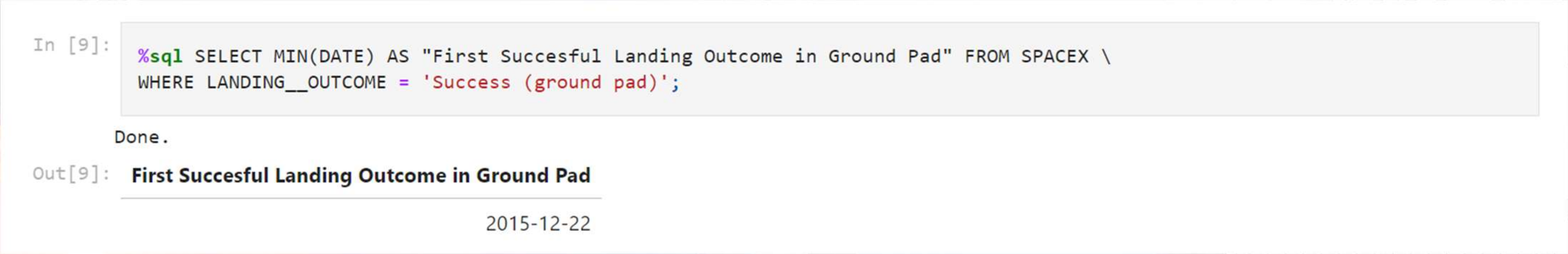
Display average payload mass carried by booster version F9 v1.1

```
In [8]:  %sql SELECT AVG(PAYLOAD_MASS__KG_) AS "Average Payload Mass by Booster Version F9 v1.1" FROM SPACEX \
         WHERE BOOSTER_VERSION = 'F9 v1.1';

         Done.
Out[8]:
```

Average Payload Mass by Booster Version F9 v1.1
2928

First Successful Ground Landing Date

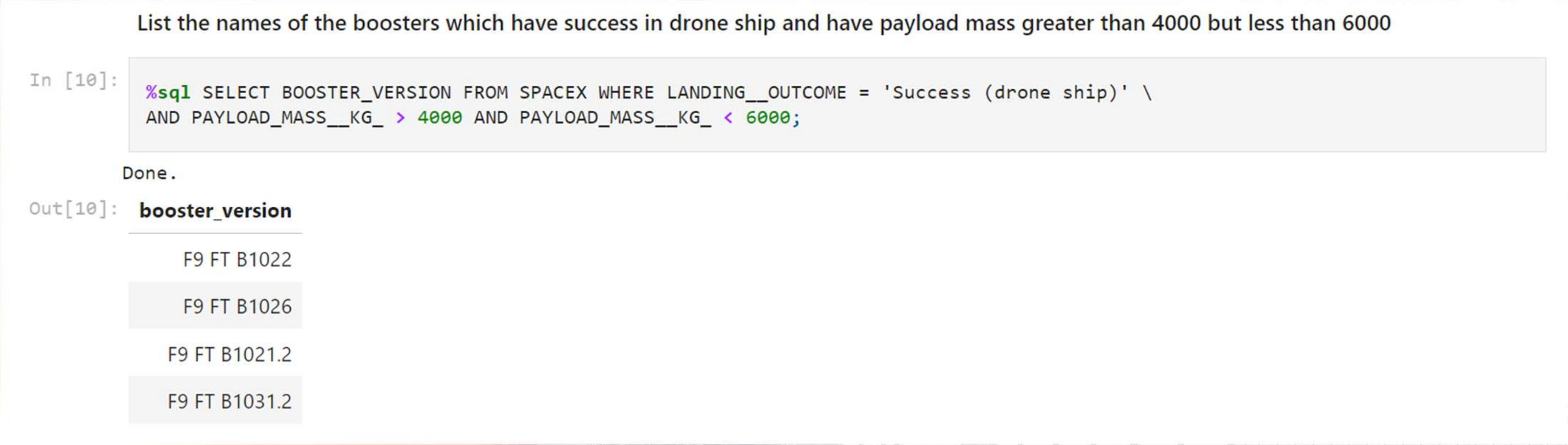
I applied the min() function to determine the result and found that the date of the first successful landing on a ground pad was December 22, 2015



```
In [9]: %sql SELECT MIN(DATE) AS "First Succesful Landing Outcome in Ground Pad" FROM SPACEX \
        WHERE LANDING__OUTCOME = 'Success (ground pad)';

        Done.

Out[9]: First Succesful Landing Outcome in Ground Pad
        ---------------------------------------------
                                           2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

I utilised the WHERE clause to filter for boosters that successfully landed on a drone ship, and applied the AND condition to identify those with a payload mass greater than 4000 but less than 6000.



List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [10]:

```
%sql SELECT BOOSTER_VERSION FROM SPACEX WHERE LANDING__OUTCOME = 'Success (drone ship)' \
AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000;
```

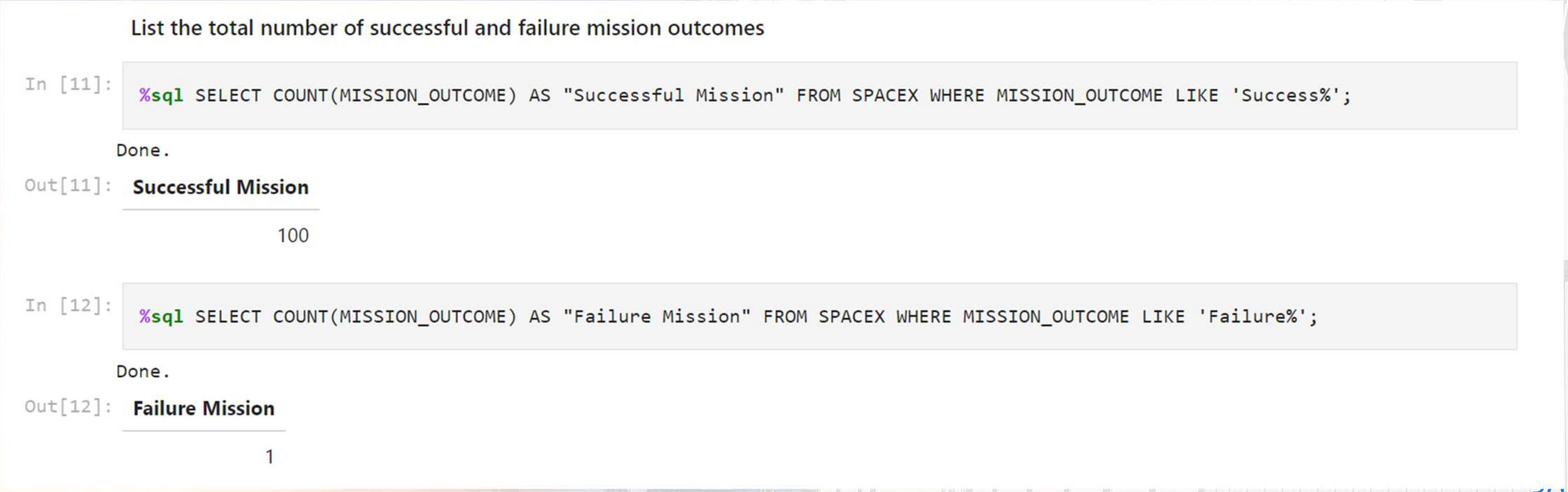
```
Done.
```

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

Wildcard like ‘%’ was utilised to filter for WHERE MissionOutcome was a success or a failure.

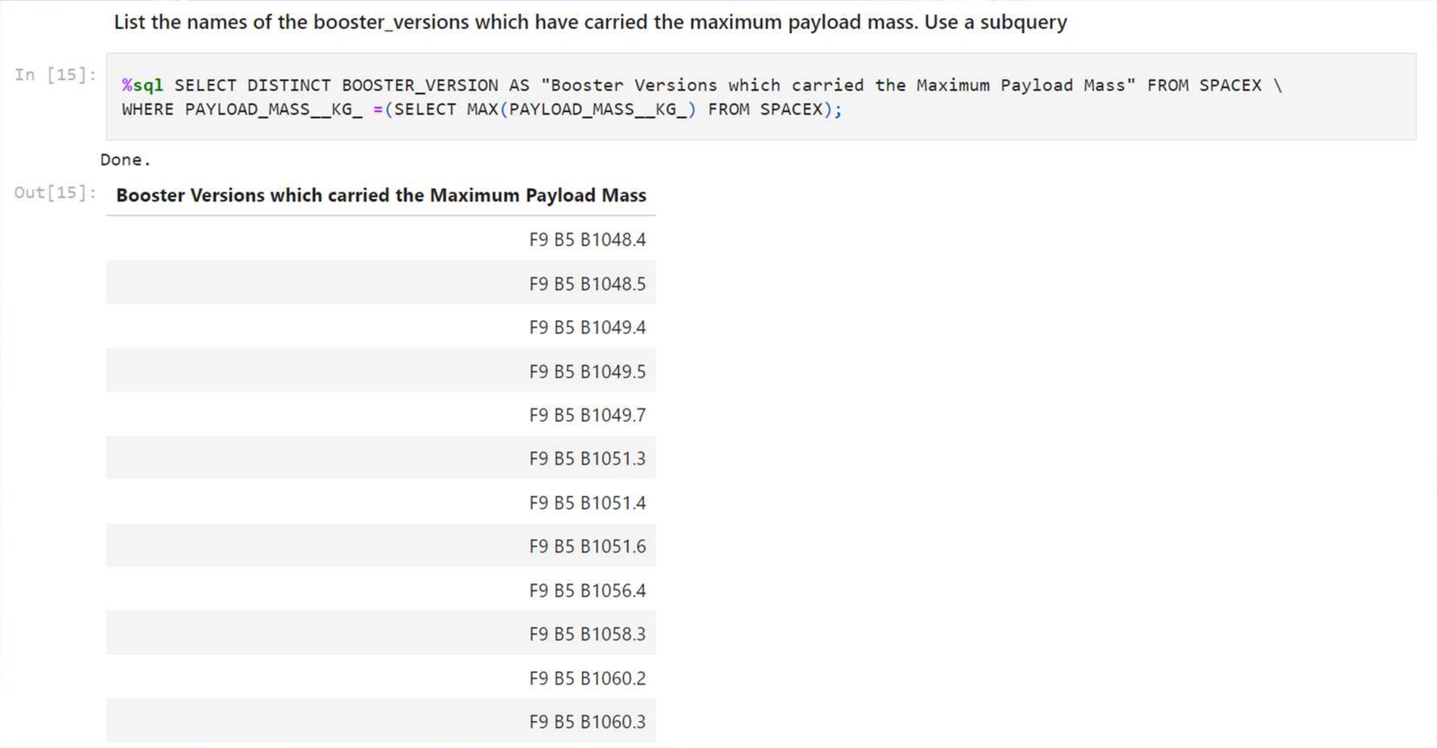


List the total number of successful and failure mission outcomes

```
In [11]:  %sql SELECT COUNT(MISSION_OUTCOME) AS "Successful Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Success%';

          Done.
Out[11]:  Successful Mission
                         100

In [12]:  %sql SELECT COUNT(MISSION_OUTCOME) AS "Failure Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Failure%';

          Done.
Out[12]:  Failure Mission
                        1
```

Boosters Carried Maximum Payload

I determined the booster that has carried the maximum payload by utilising subquery in the WHERE clause and the MAX() function.



List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
In [15]:  %sql SELECT DISTINCT BOOSTER_VERSION AS "Booster Versions which carried the Maximum Payload Mass" FROM SPACEX \
          WHERE PAYLOAD_MASS__KG_ =(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEX);
```

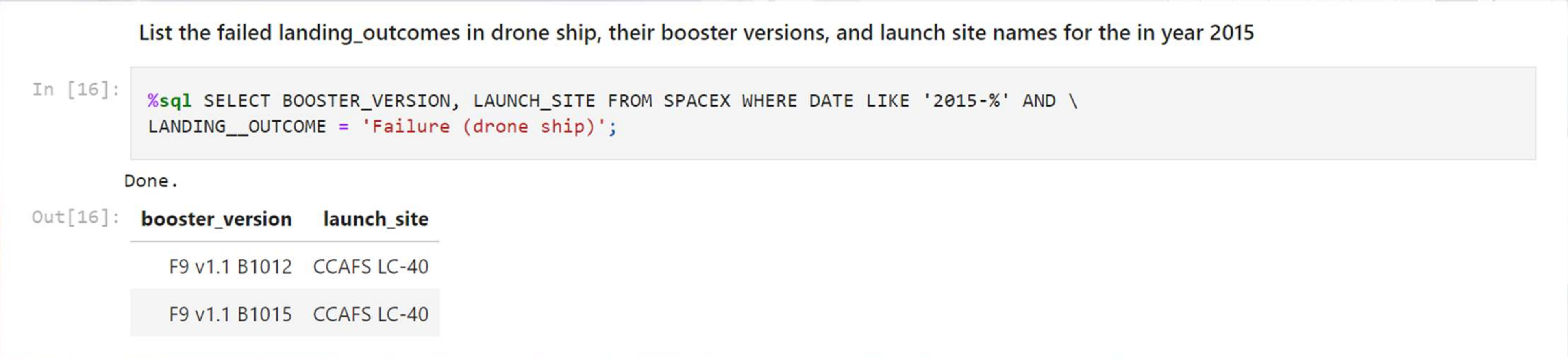
```
Done.
```

Out[15]:

Booster Versions which carried the Maximum Payload Mass
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2015 Launch Records

I utilised a combination of the WHERE clause, LIKE, AND, and BETWEEN conditions to filter for failed landing outcomes on drone ships, along with their booster versions and launch site names for the year 2015.



List the failed landing_outcomes in drone ship, their booster versions, and launch site names for the in year 2015

```
In [16]:  %sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEX WHERE DATE LIKE '2015-%' AND \
          LANDING__OUTCOME = 'Failure (drone ship)';
```

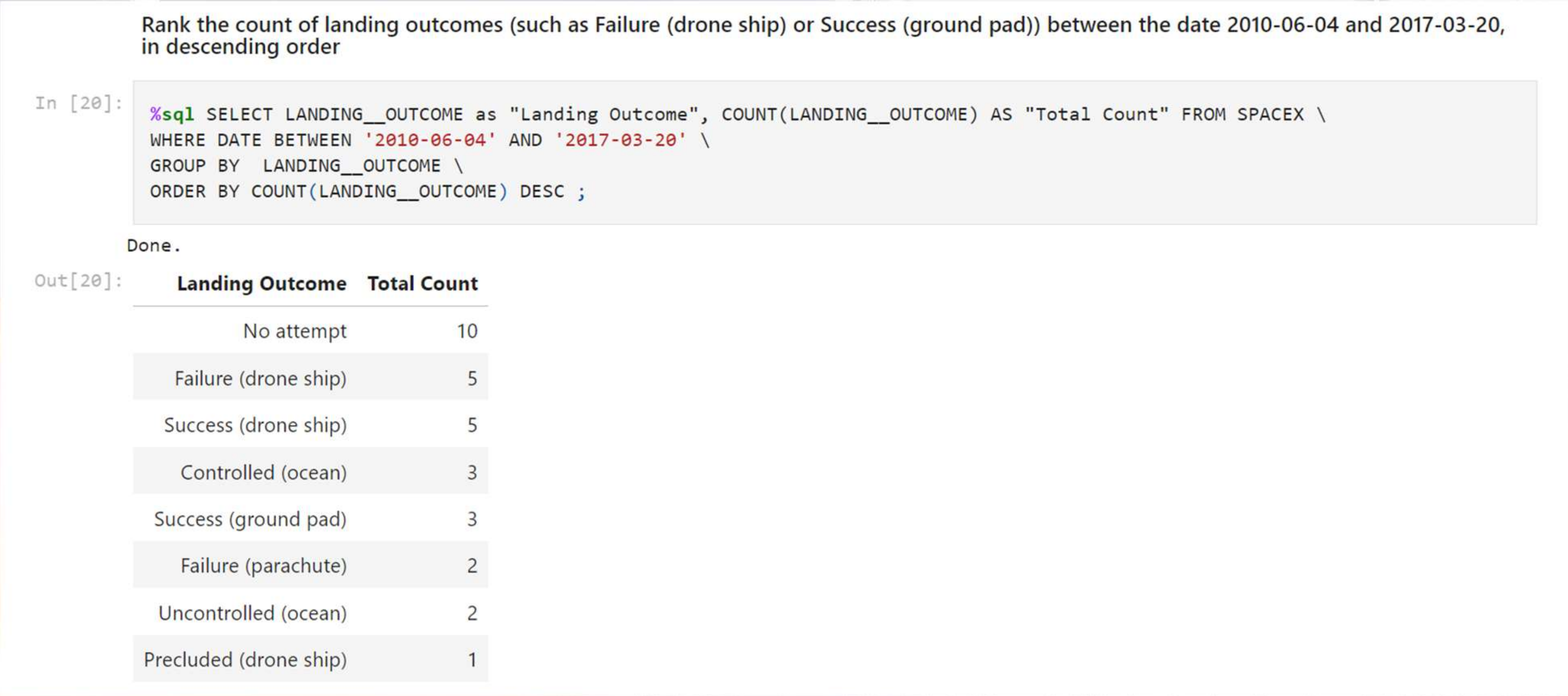
```
Done.
```

Out[16]:

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

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

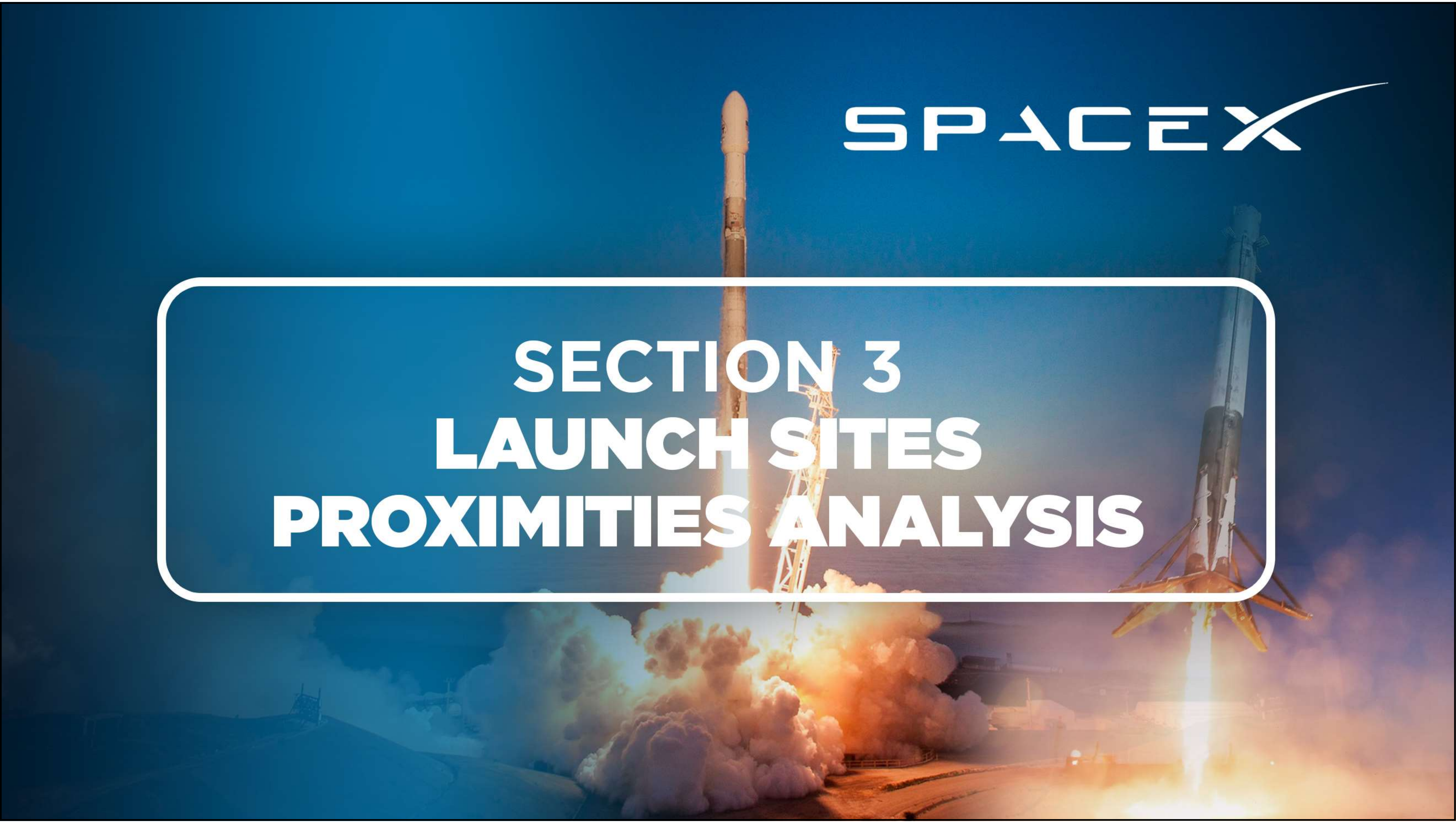
I extracted the landing outcomes and their counts from the data, using the WHERE clause to filter for outcomes between 2010-06-04 and 2010-03-20. We then applied the GROUP BY clause to categorise the landing outcomes, and the ORDER BY clause to sort these grouped outcomes in descending order.



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

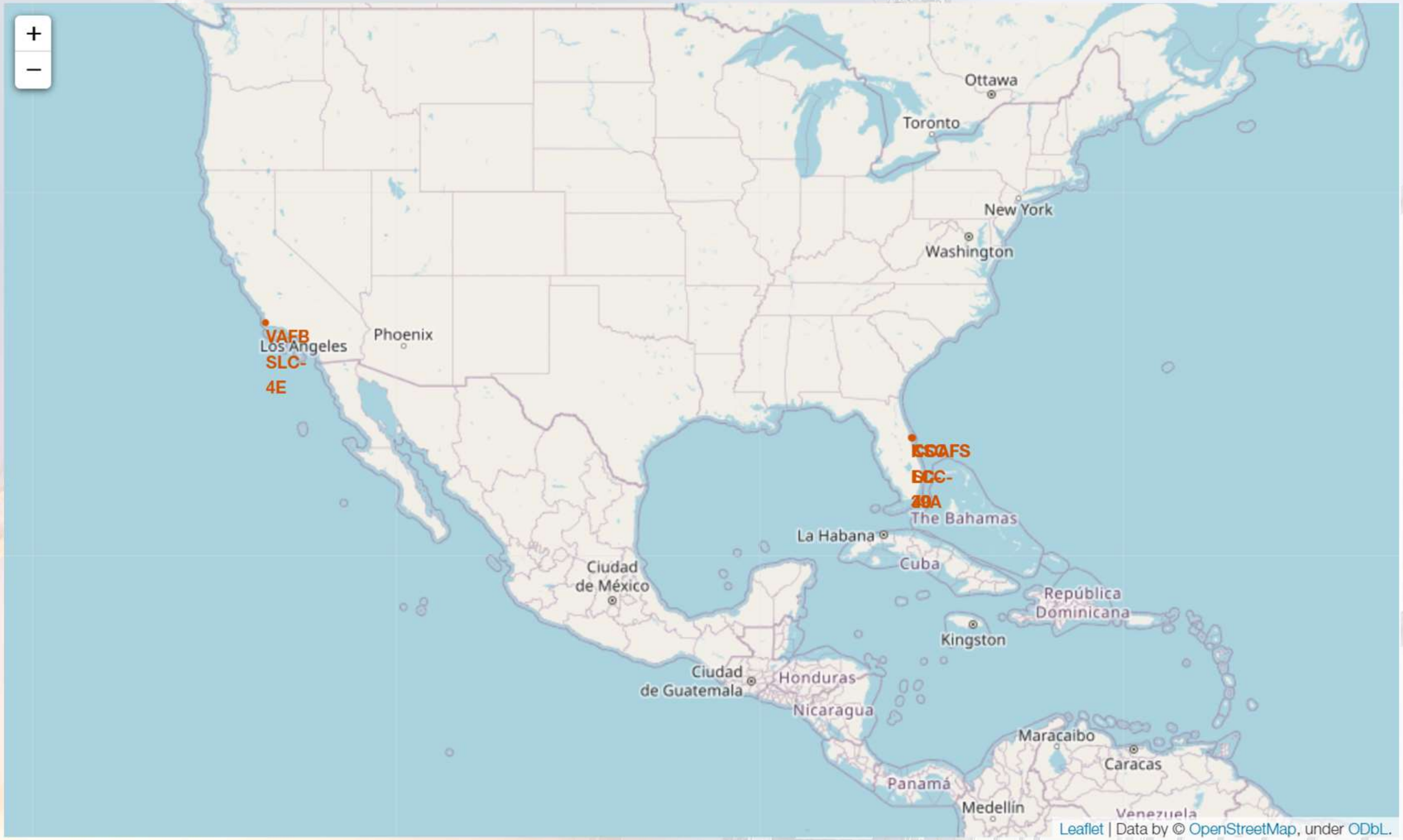
```
In [20]: %sql SELECT LANDING__OUTCOME as "Landing Outcome", COUNT(LANDING__OUTCOME) AS "Total Count" FROM SPACEX \
         WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
         GROUP BY  LANDING__OUTCOME \
         ORDER BY COUNT(LANDING__OUTCOME) DESC ;

Done.
Out[20]:
```

Landing Outcome	Total Count
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1



SECTION 3
LAUNCH SITES PROXIMITIES ANALYSIS

Map Of All Launch Site Locations



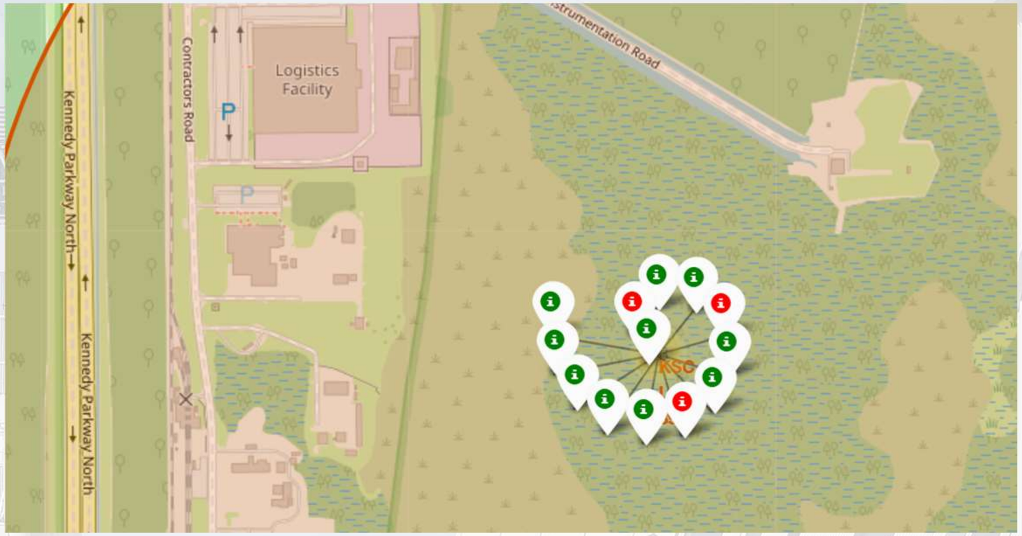
To the left we have a map of all launch sites locations (all located within the USA).

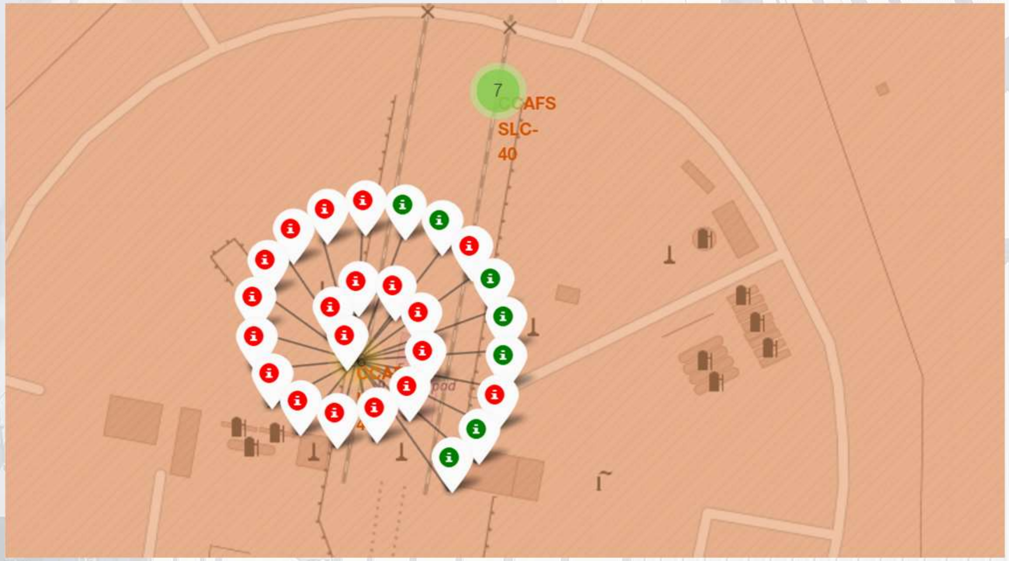
Launch Sites With Labels

The color-coded markers allow us to easily identify launch sites with relatively high success rates.

Green Markers: Successful Launch

Red Markers: Failed Launch



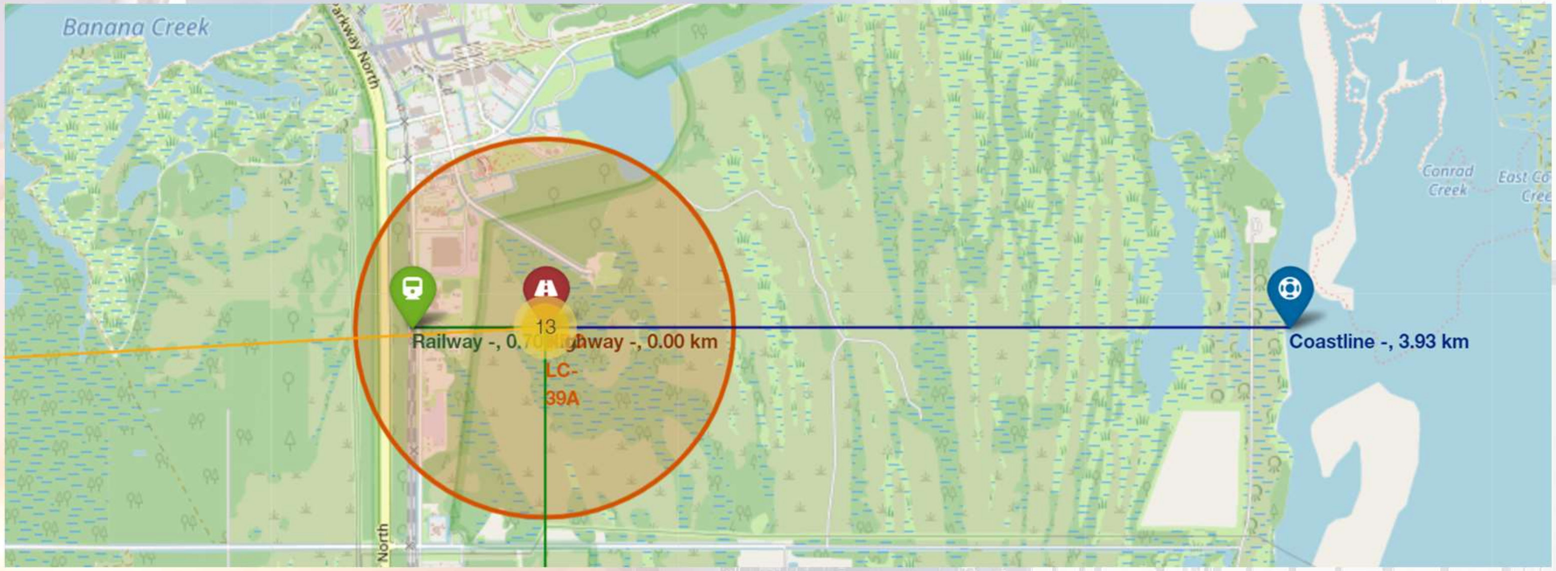


Distance To Landmarks From Launch Sites

From the visual analysis of launch site KSC LC-39A, it is evident that it is:

- Very close to a railway (0.70 km)
- Very close to a highway (0 km)
- Fairly close to the coastline (3.93 km)

A failed rocket traveling at high speeds can cover vast distances in minimal times, with the very real possibility of serious damage to built-up areas.

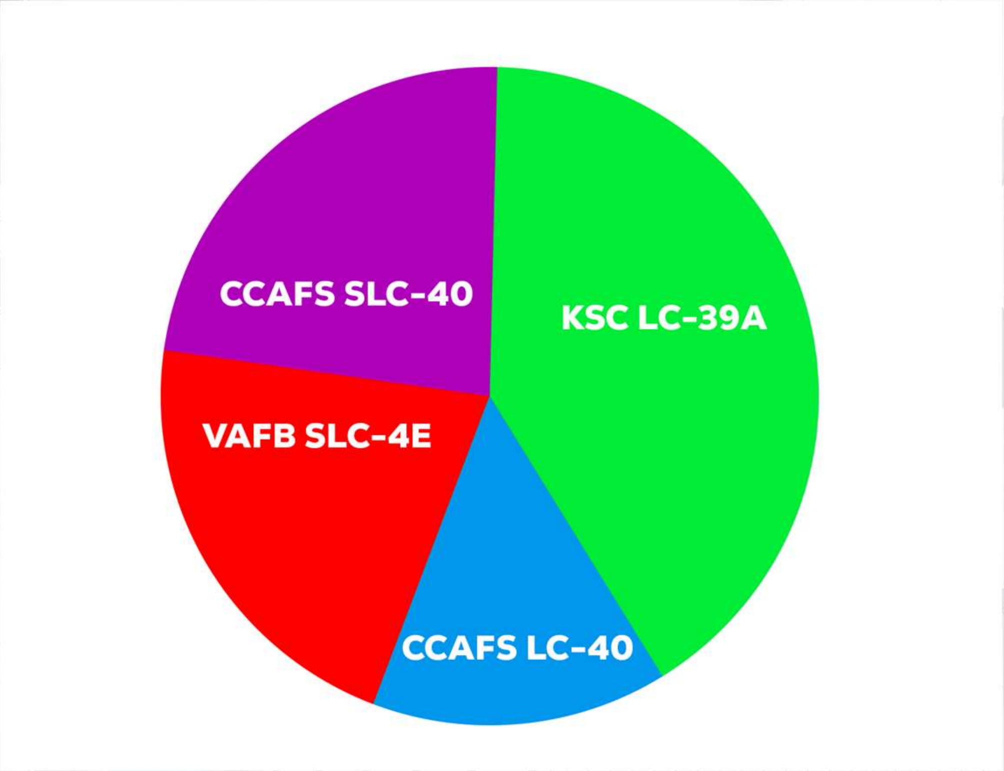


SPACEX

SECTION 4
BUILD A DASHBOARD WITH PLOTY DASH

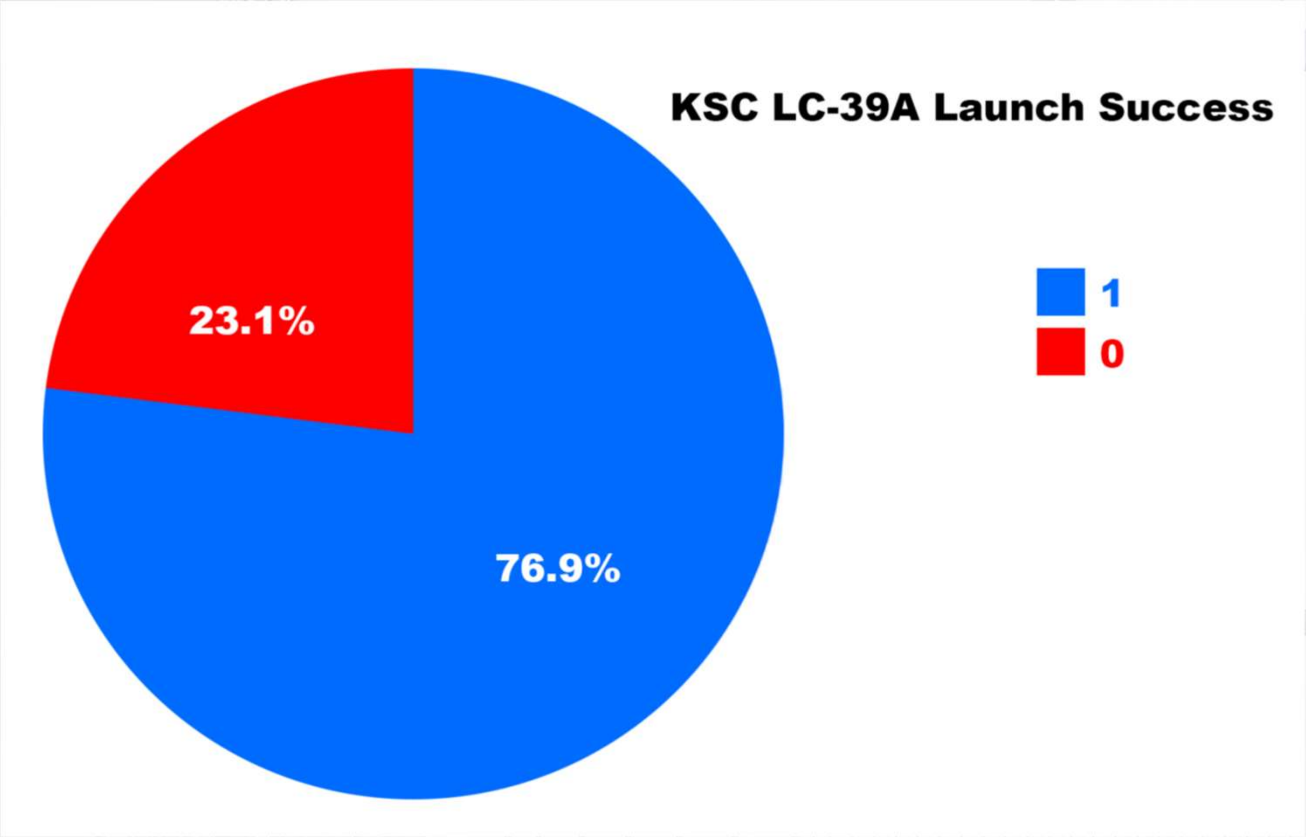
Representing Launch Success

The pie chart below demonstrates that KSC LC-39A has experience the most successful launches out of all sites.



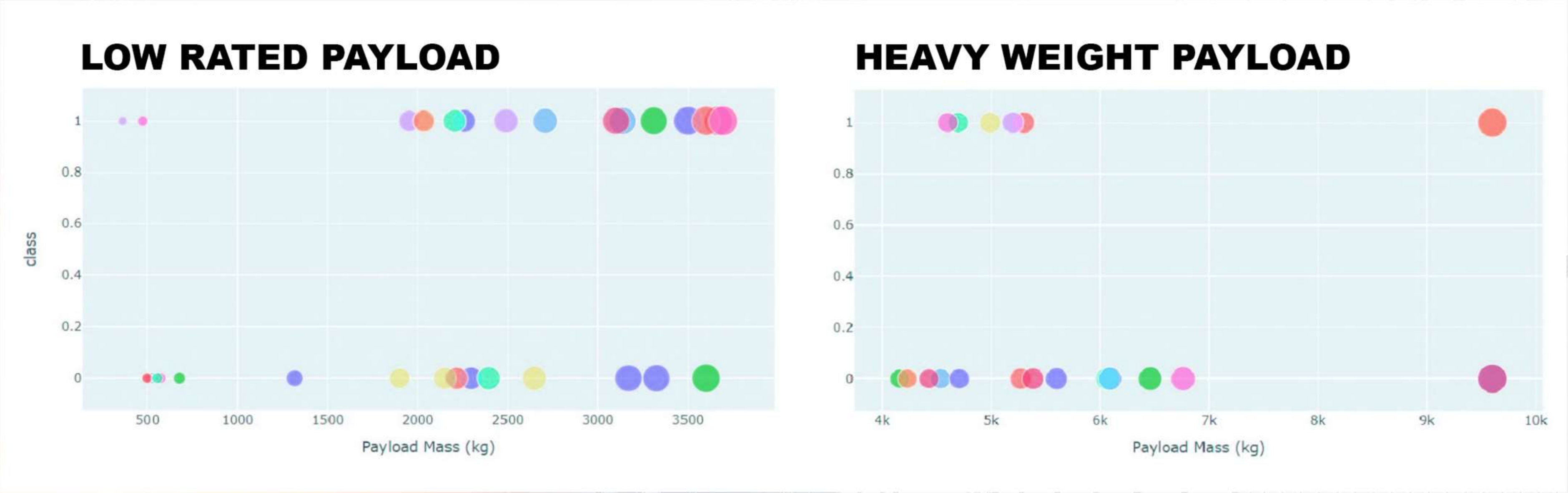
The Launch Site With The Highest Launch Success Ratio

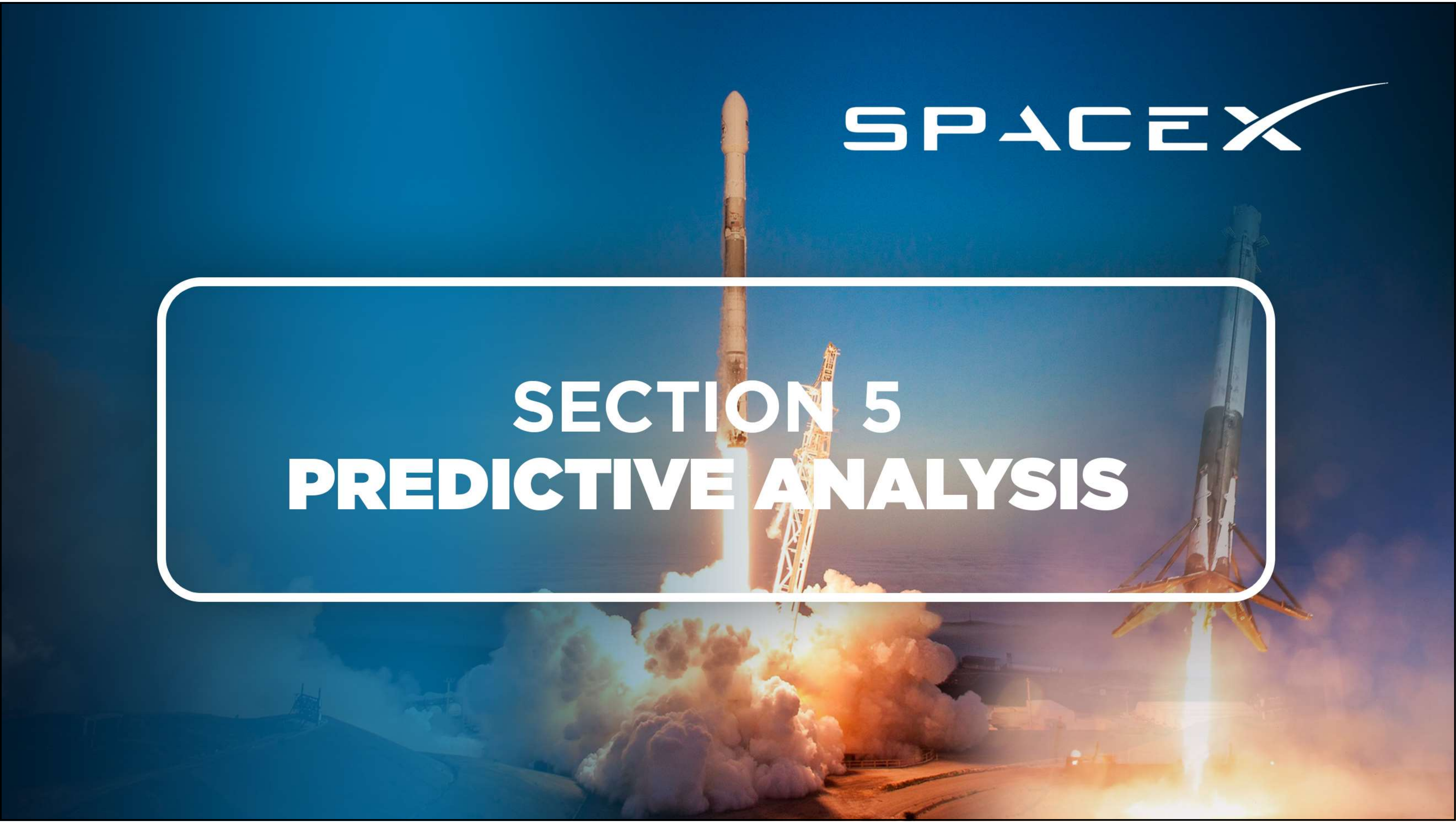
KSC LC-39A proved the highest launch success rate at 76.9%, with 10 successful landings and only 3 failures.



Scatter Plot – Payload vs Launch Outcome

We can see from the scatter graphs below that the success rate for low-weighted-payload is higher vs the heavy-weighted-payload.

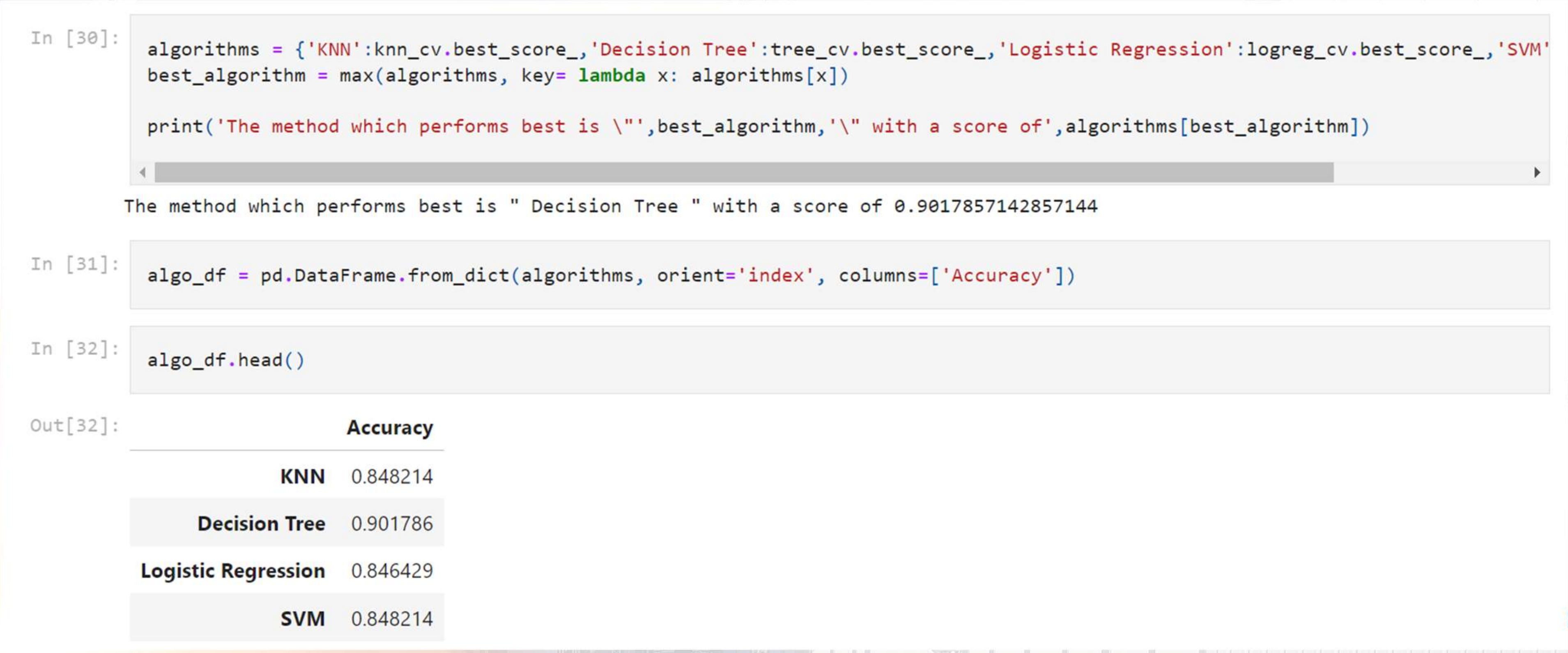




SECTION 5
PREDICTIVE ANALYSIS

Classification Accuracy

As demonstrated by the code below, I identified the “Decision Tree” Algorithm as the best algorithm due to its highest classification accuracy.



In [30]:
```
algorithms = {'KNN':knn_cv.best_score_,'Decision Tree':tree_cv.best_score_,'Logistic Regression':logreg_cv.best_score_,'SVM'
best_algorithm = max(algorithms, key= lambda x: algorithms[x])

print('The method which performs best is \"',best_algorithm,'\" with a score of',algorithms[best_algorithm])
```

```
The method which performs best is " Decision Tree " with a score of 0.9017857142857144
```

In [31]:
```
algo_df = pd.DataFrame.from_dict(algorithms, orient='index', columns=['Accuracy'])
```

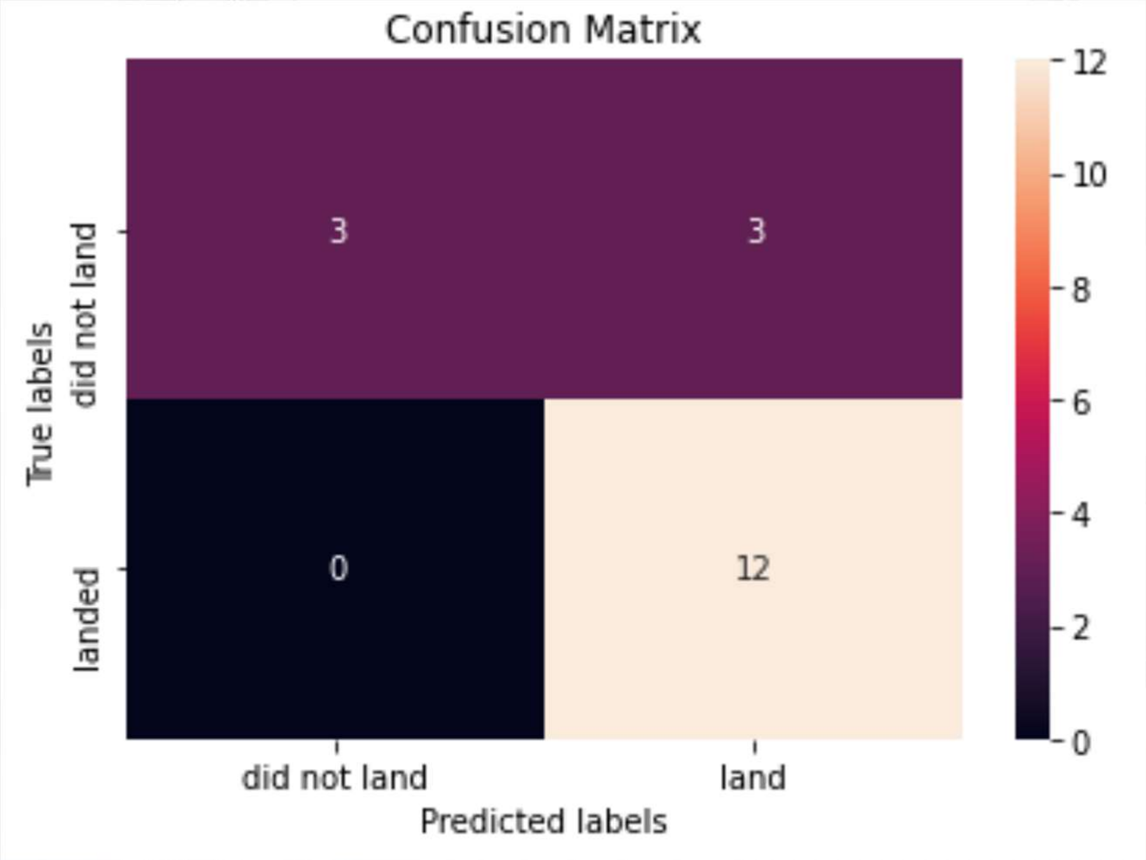
In [32]:
```
algo_df.head()
```

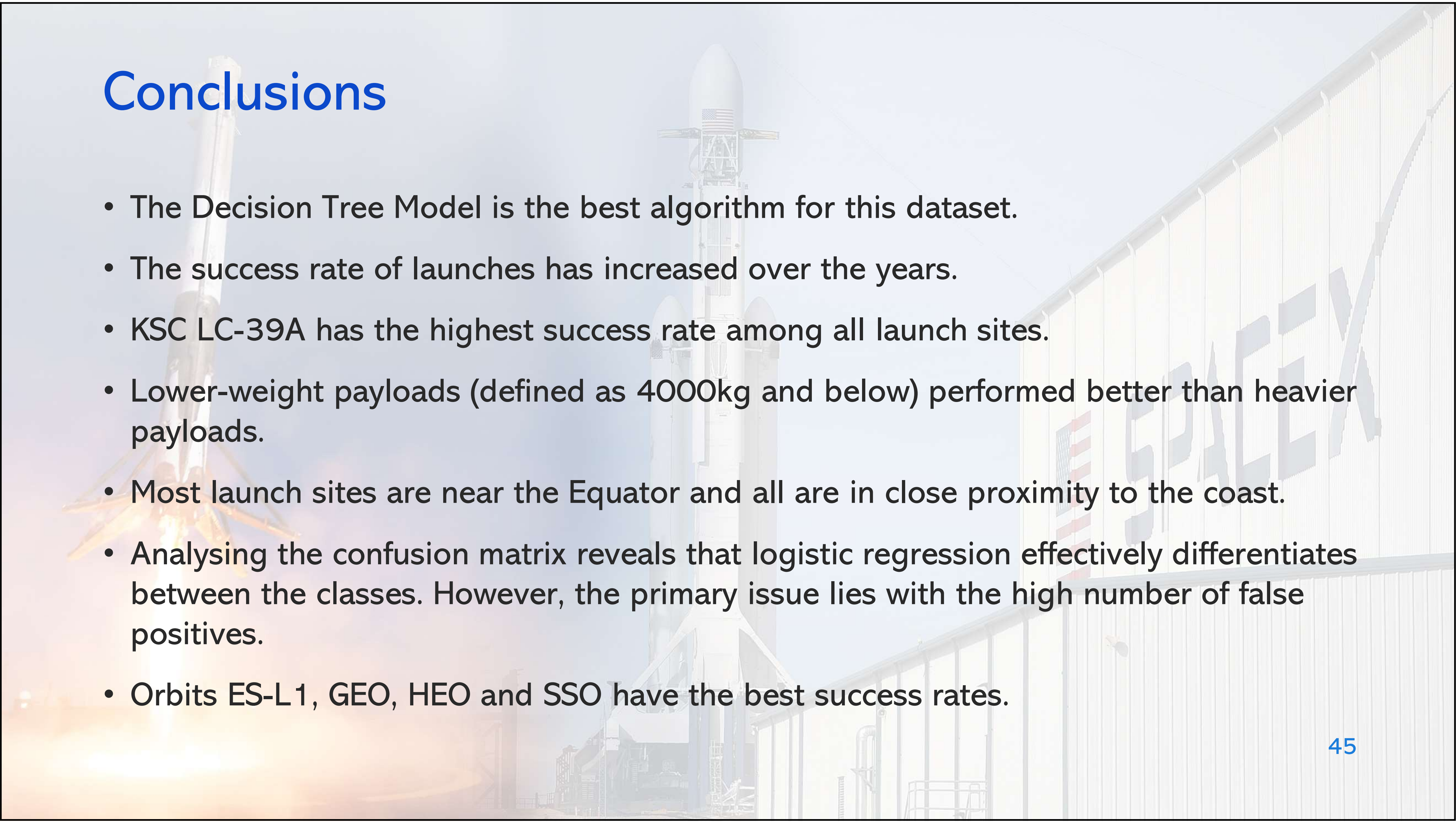
Out[32]:

	Accuracy
KNN	0.848214
Decision Tree	0.901786
Logistic Regression	0.846429
SVM	0.848214

Confusion Matrix

Analysing the confusion matrix reveals that logistic regression effectively differentiates between the classes. However, the primary issue lies with the high number of false positives.





Conclusions

- The Decision Tree Model is the best algorithm for this dataset.
- The success rate of launches has increased over the years.
- KSC LC-39A has the highest success rate among all launch sites.
- Lower-weight payloads (defined as 4000kg and below) performed better than heavier payloads.
- Most launch sites are near the Equator and all are in close proximity to the coast.
- Analysing the confusion matrix reveals that logistic regression effectively differentiates between the classes. However, the primary issue lies with the high number of false positives.
- Orbits ES-L1, GEO, HEO and SSO have the best success rates.

THANK YOU!