IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dhruvesh Bawane

02/05/2024

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

The goal of this project is to train a machine learning model using public data from SpaceX. This model predicts whether SpaceX will successfully land and reuse the first stage of the Falcon 9 rocket based on launch information.

- **Summary of methodologies**

The techniques and methodologies used during this project include:

1. **Data Collection:** To Utilize both Rest API and Web Scraping methods to collect SpaceX launch data.
2. **Data Wrangling:** Normalized, filtered, and cleaned the collected data by replacing null values with averages.
3. **Exploratory Data Analysis (EDA):** Conducted EDA using visualization techniques and SQL queries to gain insights into the dataset.
4. **Interactive Visual Analytics:** Leveraged tools like Folium and Plotly Dash for interactive visual analytics.
5. **Predictive Analysis:** Applied classification models with standardized data and used GridSearch to optimize hyperparameters for accuracy verification.

Executive Summary

- **Summary of all results**

1. **Valuable Insights:** The data collected provided valuable information for analysis.
2. **Visualization Impact:** Data visualization techniques offered unique insights into SpaceX mission outcomes.
3. **Model Performance:** Employed machine learning models, including KNN, SVM, Logistic Regression, and Decision Trees, with varying performance.
4. **Predictive Accuracy:** Predictive analysis yielded accurate models, aligning with project objectives.

Introduction

- **Project Background and Context:**

  SpaceX is a cool company that launches rockets into space. They figured out a clever way to save money by reusing the first part of the rocket, which is super expensive. This makes them stand out from other space companies and saves them a ton of cash.

- **Problems You Want to Find Answers:**

  In this project, we want to dig into SpaceX's launch data to see how often they can reuse their rockets successfully. We'll use fancy math and computer stuff to predict if they can land the first part of the rocket again. This could help us start a new company called SpaceY!

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Utilizing SpaceX Rest API
  - Web Scraping SpaceX data from Wikipedia Site
- Perform data wrangling
  - Filtered and sorted data to extract relevant information for analysis.
  - Addressed them with average values or removing them from the dataset. missing values by either imputing
  - Created binary labels to indicate mission success or failure based on landing outcomes.
- Performed exploratory data analysis (EDA) using visualization and SQL, interactive visual analytics using Folium and Plotly Dash, And Performed predictive analysis using classification models By Build, train, and evaluate models using various algorithms

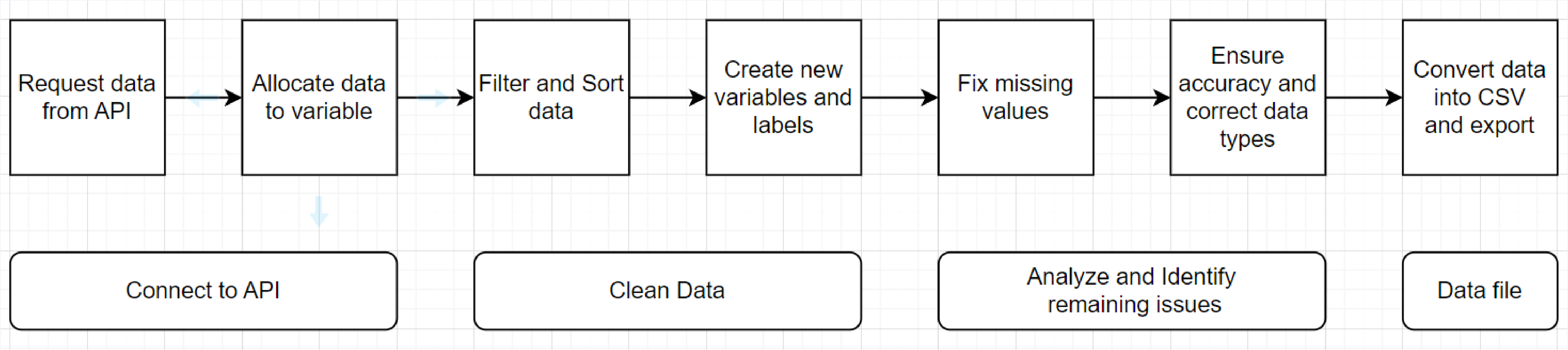
Data Collection

- Public Data related to SpaceX flights were collected via two methods
  - From SpaceX REST API
  - From Wikipedia Page of SpaceX Launch records via Web Scrapping

Data Collection – SpaceX API

- Data collected via SpaceX API Followed the process given in the Flowchart below.

SpaceX Rest API Process

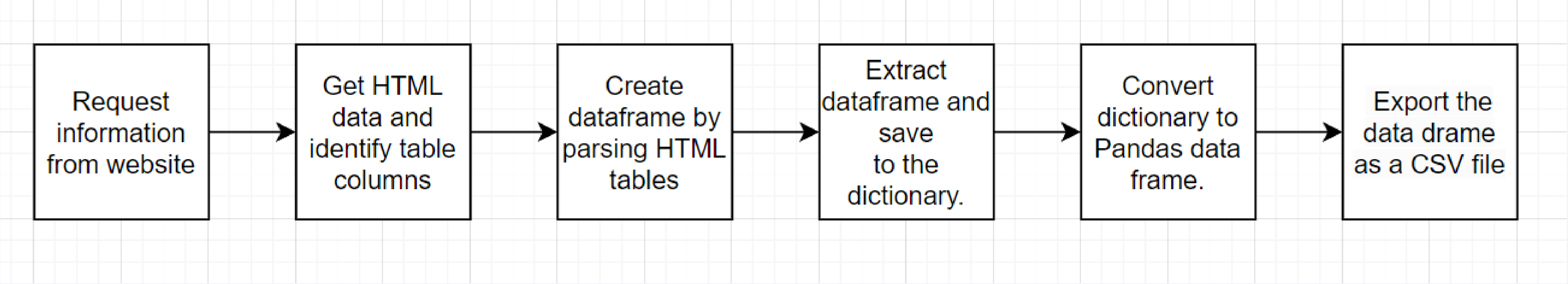

Request data from API
Allocate data to variable
Filter and Sort data
Create new variables and labels
Fix missing values
Ensure accuracy and correct data types
Convert data into CSV and export
Connect to API
Clean Data
Analyze and Identify remaining issues
Data file

Github: Python Notebook link

Data Collection - Scraping

- Data collected via Web Scraping followed the process given in the Flowchart below.
- Here, The Data is Downloaded from Wikipeda.

Web Scraping Process



Request information from website

Get HTML data and identify table columns

Create dataframe by parsing HTML tables

Extract dataframe and save to the dictionary.

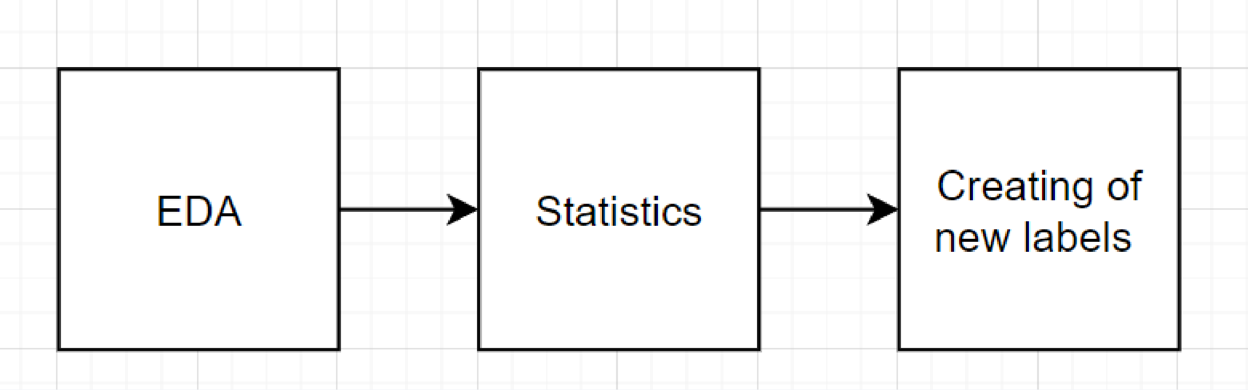
Convert dictionary to Pandas data frame.

Export the data drame as a CSV file

Github: Python Notebook Link

Data Wrangling

- Explanatory Data Analysis was performed on dataset. To create a number of Statistical outcomes
- The values for Launch Site, Orbit, and Outcome was determined, with outcome being attached to a new variable Landing Outcome.
- Landing Outcome was used to form Landing Class which was used to determine the success rate of a mission.


EDA → Statistics → Creating of new labels

Github: Python Notebook Link

EDA with SQL

Performed SQL queries:

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string ‘CCA'
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date when the first successful landing outcome in ground pad was achieved
- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

Github: Python Notebook Link

EDA with SQL

- Listing the total number of successful and failure mission outcomes
- Listing the names of the booster versions which have carried the maximum payload mass
- Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015
- Ranking 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: Python Notebook Link

EDA with Data Visualization

- Charts were plotted:

  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 and Success Rate Yearly Trend

- Scatter Plot : Show the relationship/correlation between two variables. Used to identify patterns.
- Bar Chart : Used to compare values among discrete categories. The bar chart created for this analysis illustrated success rate for each launch orbit type.
- Line Chart Typically used to show time series trends. The line chart created for this analysis illustrated annual success rate over time (from 2010 2020)

Github: Python Notebook Link

Build an Interactive Map with Folium

- Markers, circles, lines and marker clusters were created and added to a folium map
  - Markers: These are like map pins that point out important spots, like where rockets take off from (launch sites).
  - Circles: They're like rings around a spot on the map, drawing attention to specific areas, kind of like a highlight.
  - Marker Clusters: Imagine a bunch of markers all huddled together in one place on the map. That's what marker clusters show—they group together similar things happening in the same spot, like a bunch of rocket launches at one launch site.
  - Lines: These are like straight paths drawn between two points on the map. They're used to show how far apart different places are from each other, so you can see the distance between them.

Github: Python Notebook Link

Build a Dashboard with Plotly Dash

- The interactive dashboard built using Plotly Dash allows users to adjust parameters such as launch site selection and payload mass range, providing a dynamic and user-friendly experience for exploring and analyzing launch data.

- The percentage of launches by site helps us understand the distribution of launch activities across different sites. This allows us to identify which sites are the most active in terms of launches.

- The payload range slider enables users to filter launches based on the mass of the payload. This feature is valuable for analyzing how payload mass correlates with launch success rates and identifying any patterns or trends.

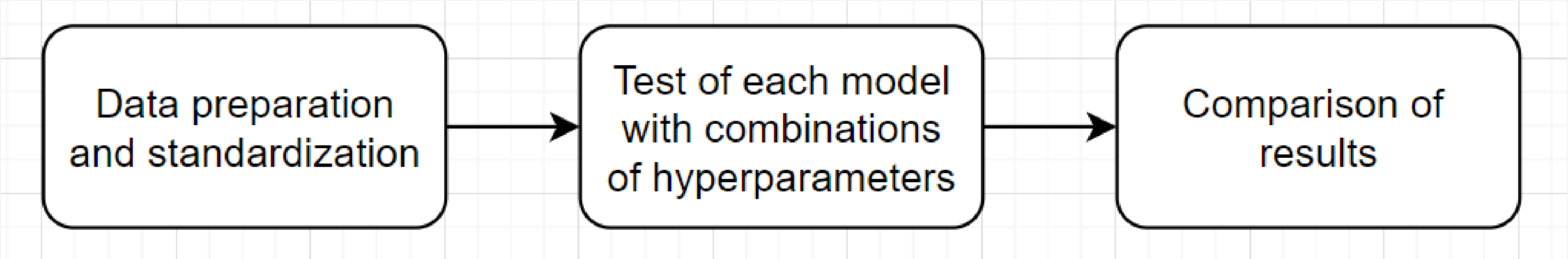
- The pie chart displaying success launches provides a clear visualization of the overall success rate across all launch sites. It also allows users to compare success and failure counts for specific launch sites, aiding in site selection for future missions.

- The scatter chart of payload mass vs. success rate for different booster versions offers insights into the relationship between payload mass and launch outcomes. By coloring points based on booster versions, users can identify which booster versions have the highest success rates, informing decision-making regarding booster selection for future missions.

Github: Python file Link

Predictive Analysis (Classification)

- During Predictive Analysis, four classification models logistic regression, support vector machine, decision tree and k nearest neighbors were compared.


Data preparation and standardization
Test of each model with combinations of hyperparameters
Comparison of results

Github: Python Notebook Link

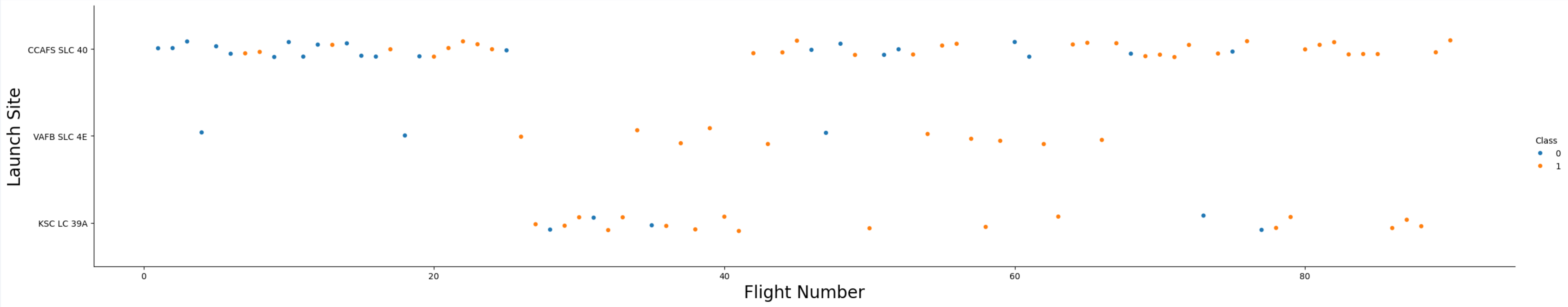
Results

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

Section 2

Insights drawn from EDA

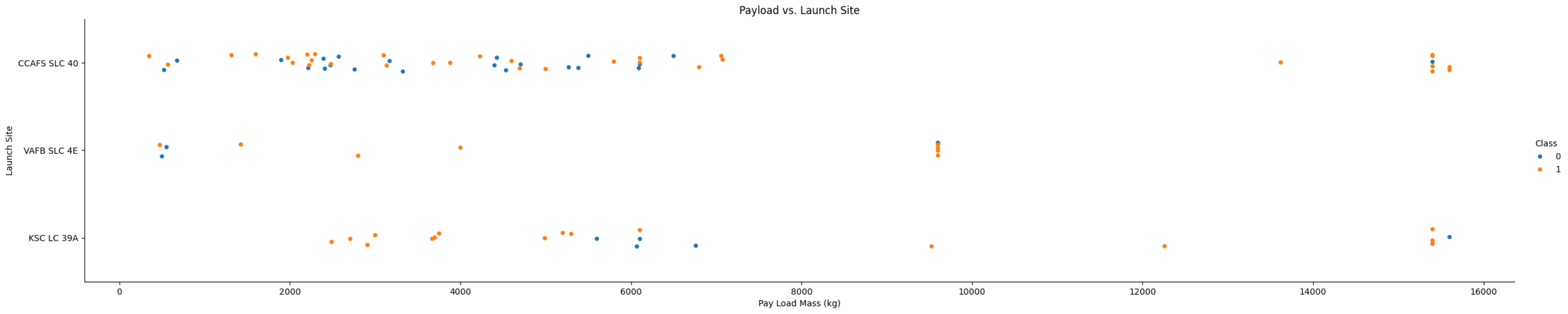
Flight Number vs. Launch Site



Scatter plot of Flight Number vs. Launch Site

- The scatter plot shows flight numbers on the horizontal axis and launch sites on the vertical axis.
- Blue dots mean the mission failed, while orange dots mean it succeeded.
- CCAFS SLC 40 had the most launches, including 18 of the first 20.
- Over time, success rates improved, especially after the 30th launch.
- Currently, CCAFS SLC 40 is the best site, with most recent launches successful.
- VAFB SLC 4E is second, and KSC LC 39A is third.
- The success rate has improved over time.

Payload vs. Launch Site



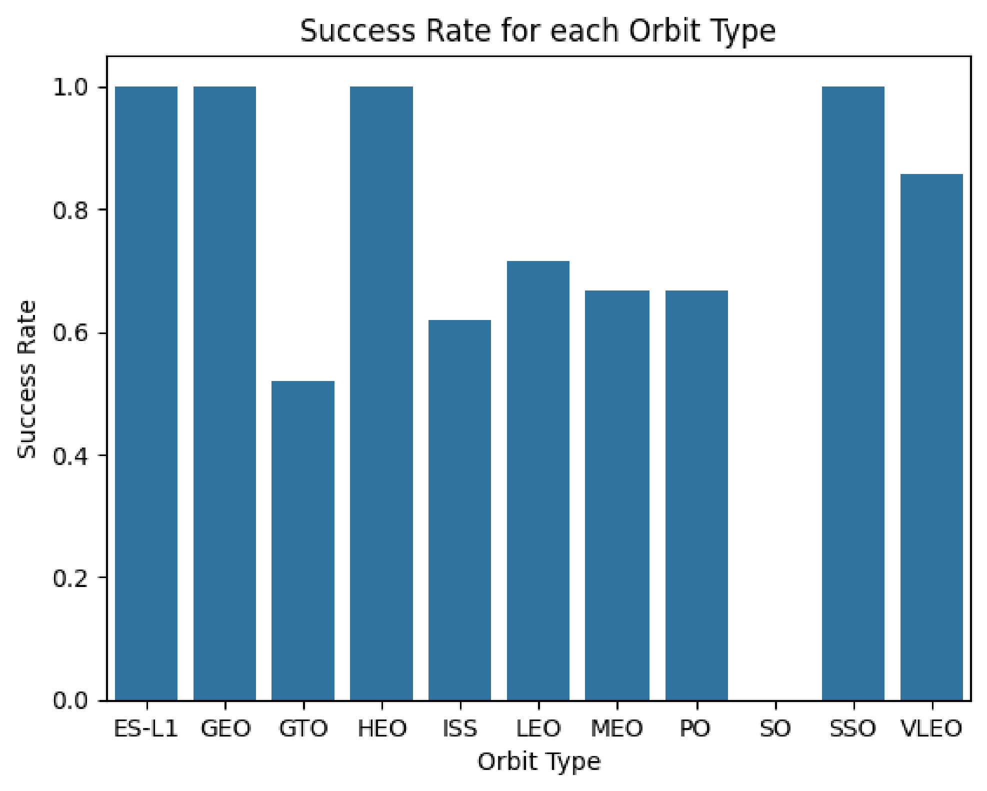
Scatter plot of Payload vs. Launch Site

The scatter plot shows Payload Mass (in kg) on the x-axis and Launch Site on the y-axis, with blue dots for failure and orange dots for success.

- Most launches had payloads under 7,000 kg.
- VAFB SLC 4E never launched a rocket with a payload exceeding 10,000 kg.
- High payload launches (over 8,000 kg) had a high success rate.
- Payloads over 9,000 kg had excellent success rates.
- Payloads over 12,000 kg were possible only at CCAFS SLC 40 and KSC LC 39A launch sites.

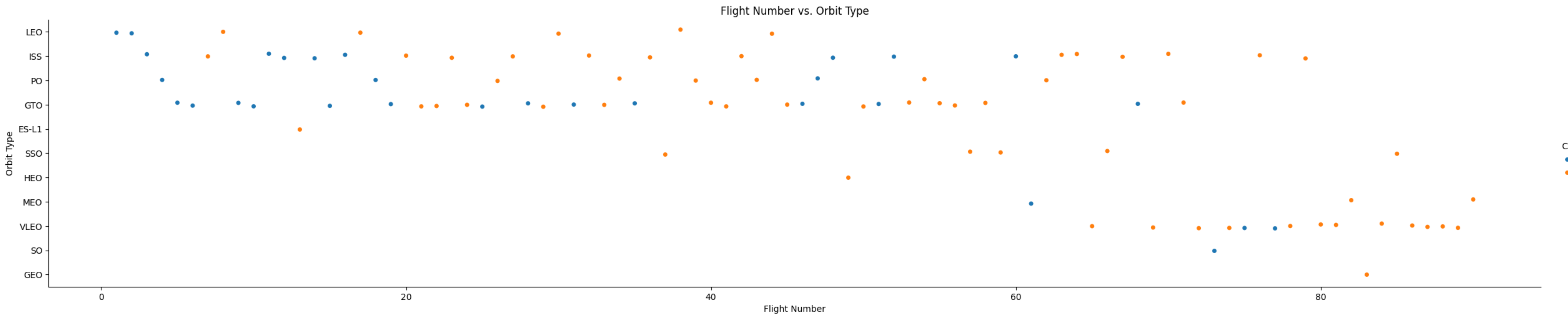
Success Rate vs. Orbit Type

- Here in the Bar Chart, Orbit type is the x-axis, success rate is on the y-axis.
- ES-L1, GEO, HEO, and SSO had the highest success rates at 100%.
- SO has the lowest Success Rate, at 0%



- Bar chart for the success rate of each orbit type

Flight Number vs. Orbit Type

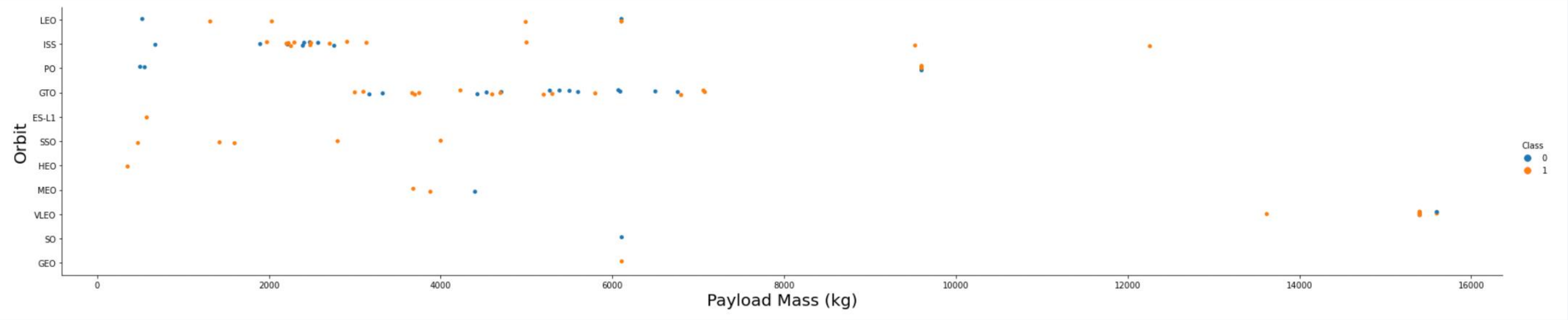


Scatter point of Flight number vs. Orbit type

- Here, Flight number is on the x-axis, orbit type is on the y-axis, with blue data points indicating mission failure and orange data points indicating mission success.

- 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

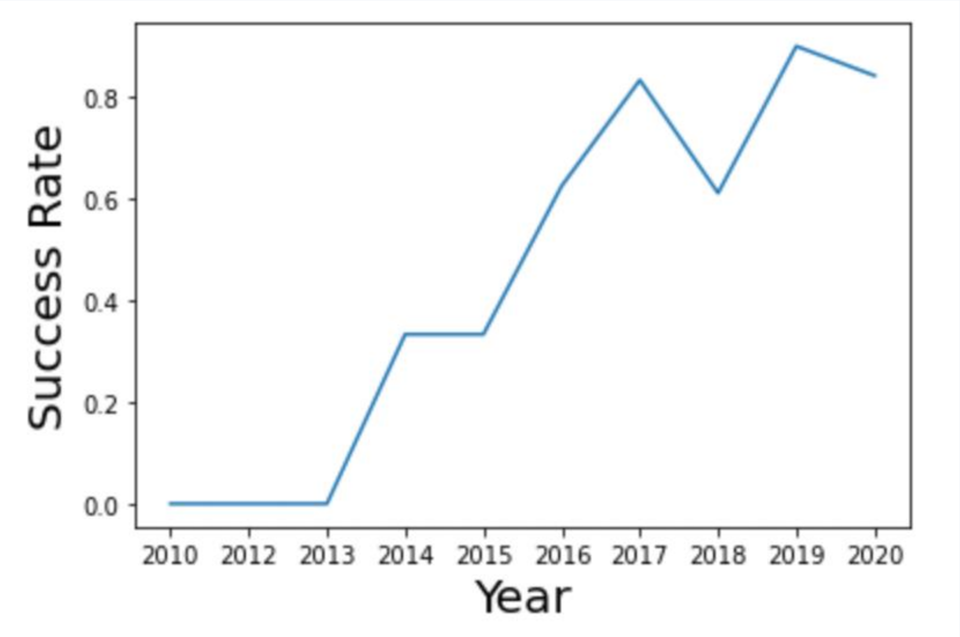


Scatter point of payload vs. orbit type

- Here, Payload Mass (in kg) is the x-axis, orbit type is the y-axis, with blue data points indicating mission failure and orange data points indicating success.

- And, We can see that heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.

Launch Success Yearly Trend

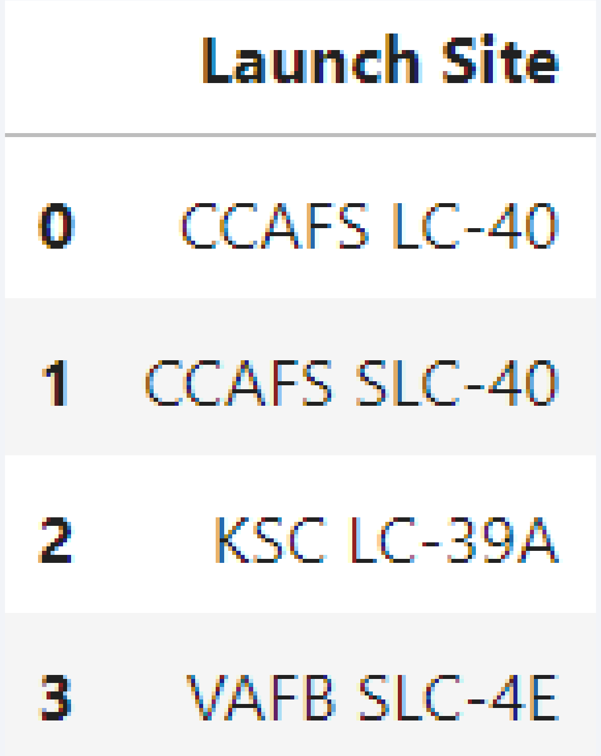


A line chart of yearly average success rate

- Here, Year is the x-axis, success rate is the y-axis.
- From the line chart we can conclude that the success rate since 2013 kept increasing till 2020

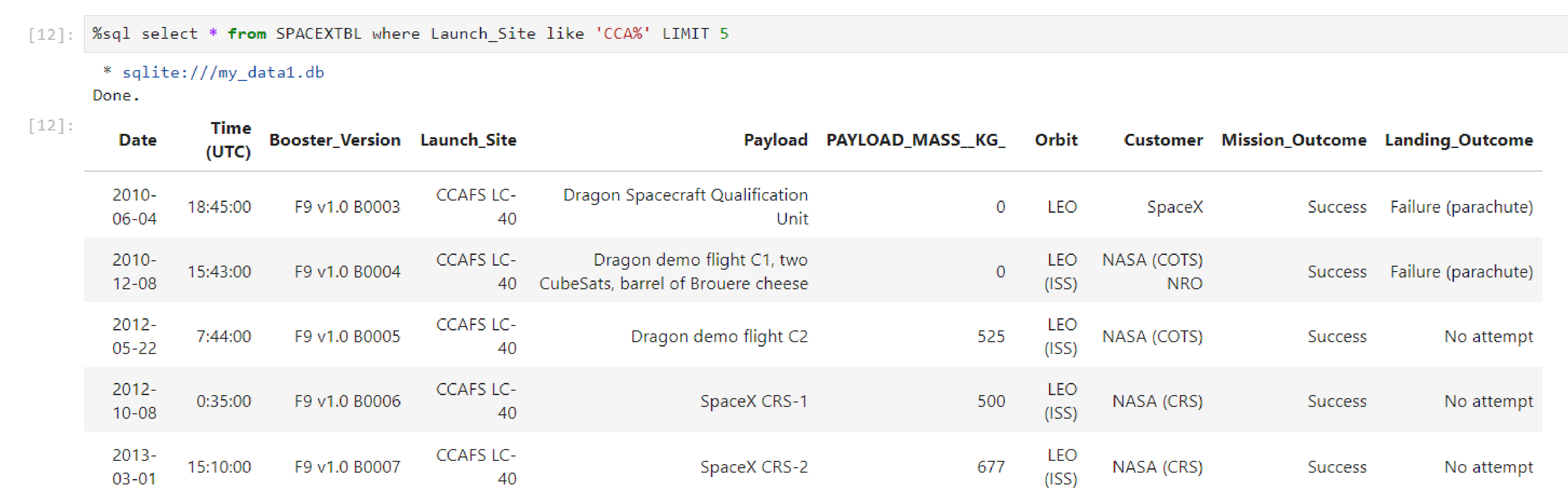
All Launch Site Names

- There were 4 launch sites used by SpaceX for flights contracted through NASA (CRS).



	Launch Site
0	CCAFS LC-40
1	CCAFS SLC-40
2	KSC LC-39A
3	VAFB SLC-4E

Launch Site Names Begin with 'CCA'



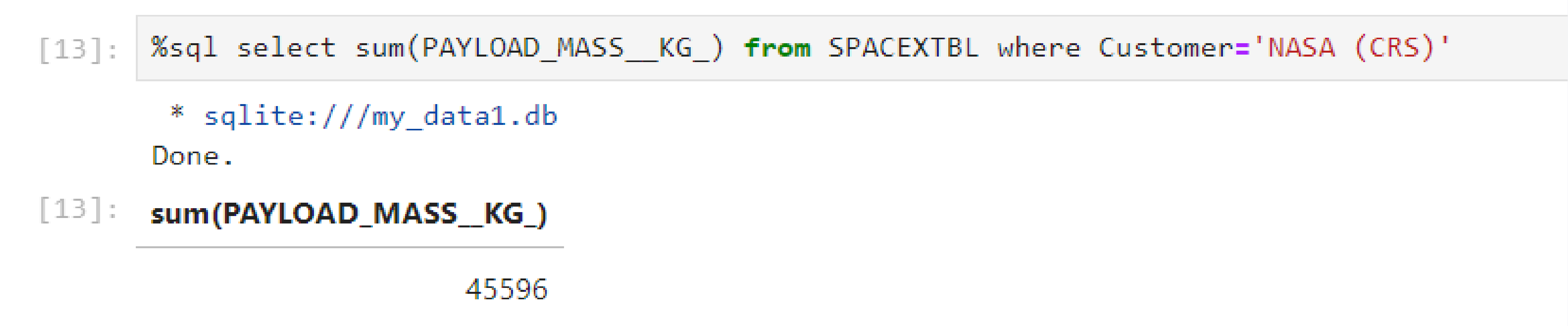
```
%sql select * from SPACEXTBL where Launch_Site like 'CCA%' LIMIT 5
```

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

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

Displaying 5 records where launch sites begin with `CCA`

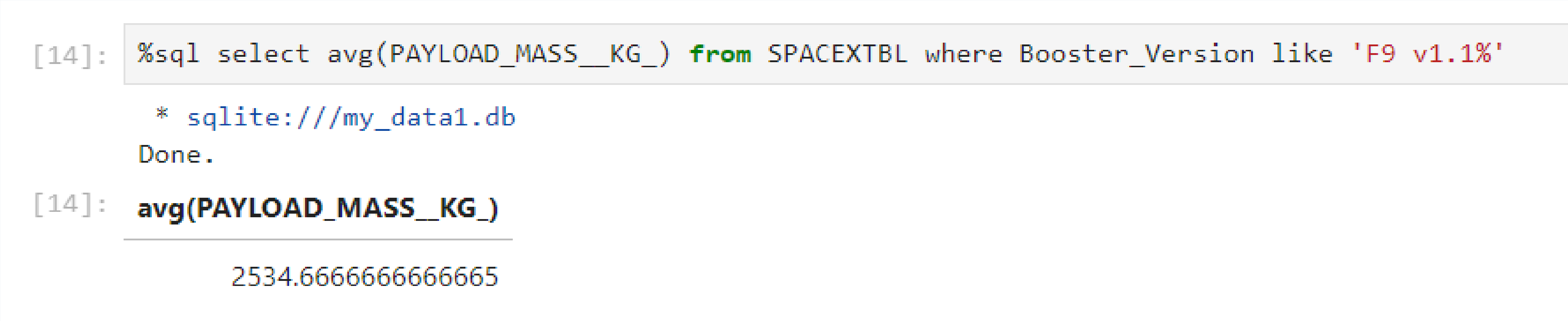
Total Payload Mass



```
[13]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where Customer='NASA (CRS)'

       * sqlite:///my_data1.db
      Done.

[13]: sum(PAYLOAD_MASS__KG_)
      ----------------------
                       45596
```

Displaying the total payload carried by boosters from NASA

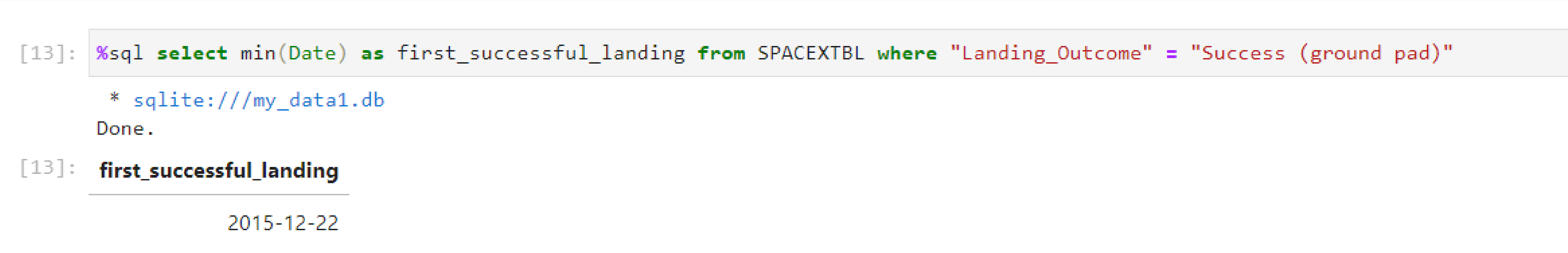
Average Payload Mass by F9 v1.1



```
[14]: %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version like 'F9 v1.1%'

       * sqlite:///my_data1.db
      Done.

[14]: avg(PAYLOAD_MASS__KG_)

                2534.6666666666665
```

Displaying the average payload mass carried by booster version F9 v1.1

First Successful Ground Landing Date



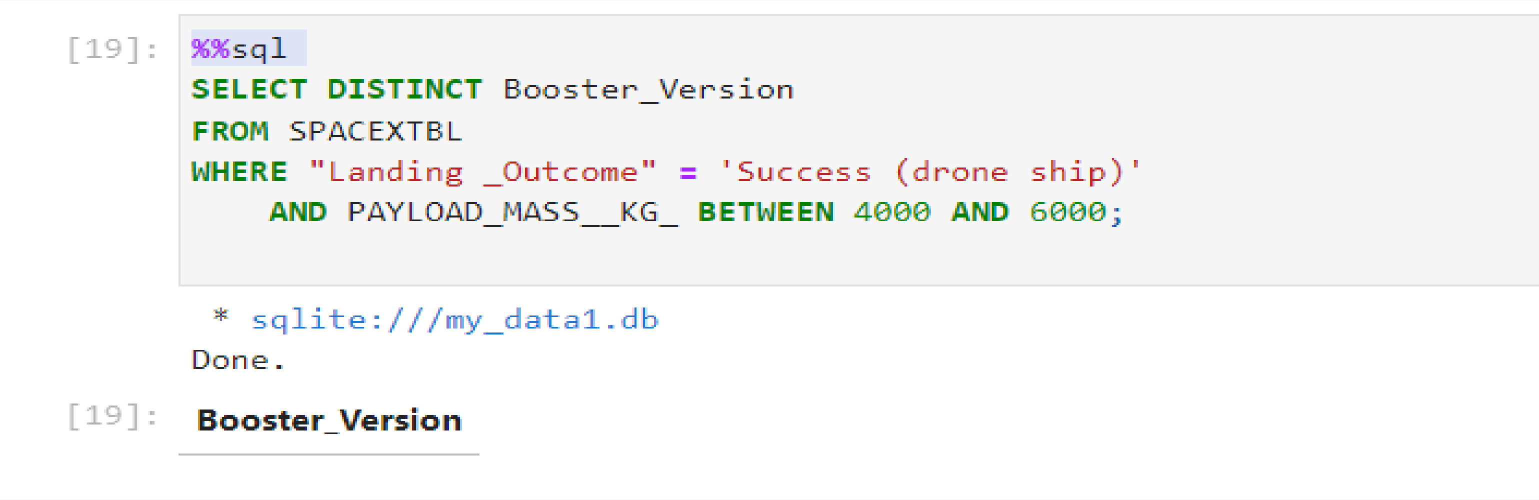
```
[13]: %sql select min(Date) as first_successful_landing from SPACEXTBL where "Landing_Outcome" = "Success (ground pad)"

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

[13]:

first_successful_landing
2015-12-22

Here, we are finding the dates of the first successful landing outcome on ground pad

Successful Drone Ship Landing with Payload between 4000 and 6000

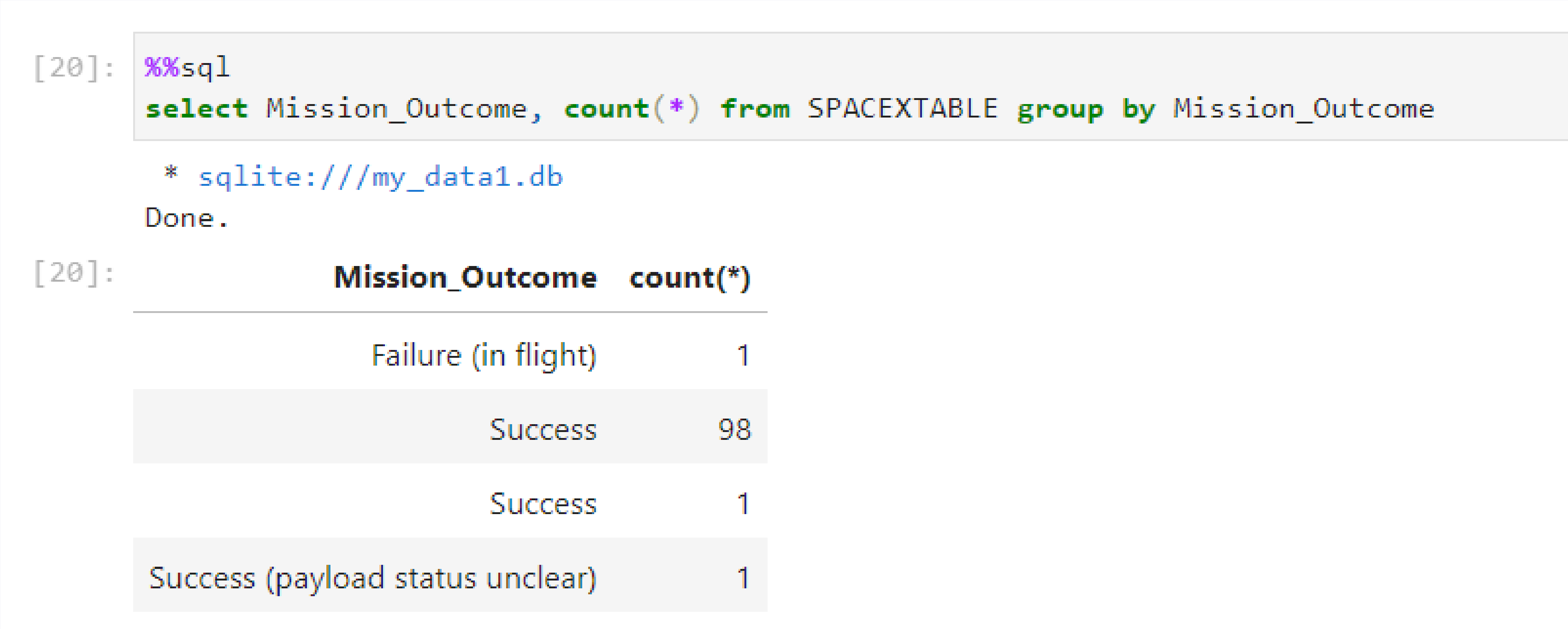


```
[19]: %%sql
      SELECT DISTINCT Booster_Version
      FROM SPACEXTBL
      WHERE "Landing _Outcome" = 'Success (drone ship)'
          AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;

       * sqlite:///my_data1.db
      Done.

[19]:  Booster_Version
```

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

Note: There seems to be and problem with the SQL, That's why the booster versions are not showing.

Total Number of Successful and Failure Mission Outcomes



```
[20]: %%sql
      select Mission_Outcome, count(*) from SPACEXTABLE group by Mission_Outcome
```

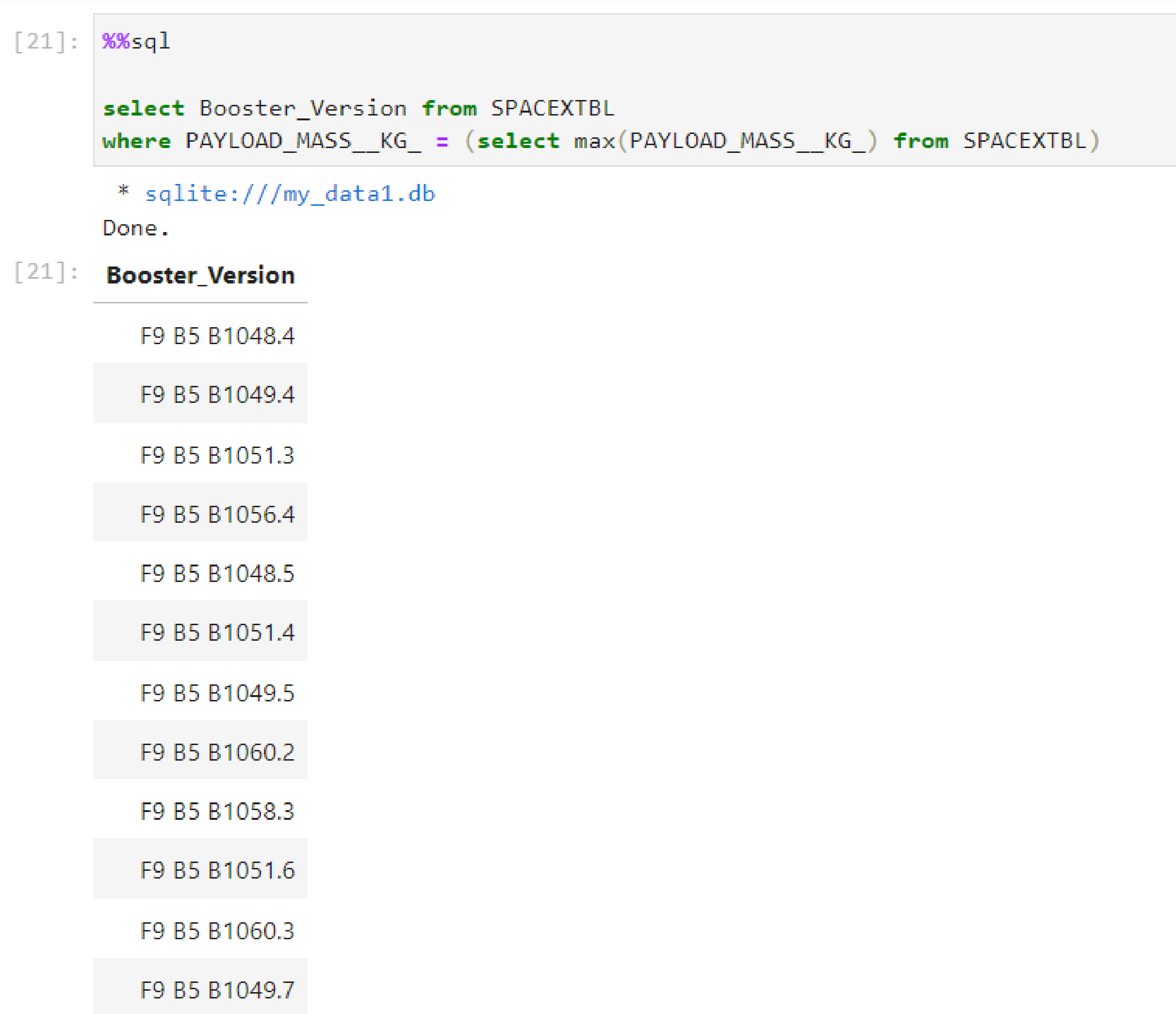
```
 * sqlite:///my_data1.db
Done.
```

[20]:

Mission_Outcome	count(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Displaying the Calculation of the total number of successful and failure mission outcomes.

Boosters Carried Maximum Payload



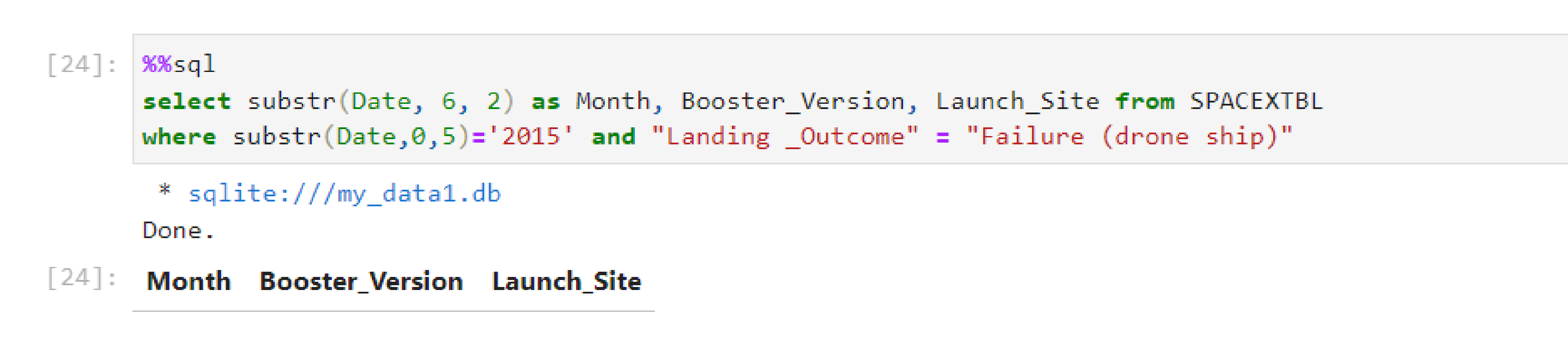
```
[21]: %%sql

      select Booster_Version from SPACEXTBL
      where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)

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

[21]:

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

The following display, the List of names of the booster which have carried the maximum payload mass

2015 Launch Records

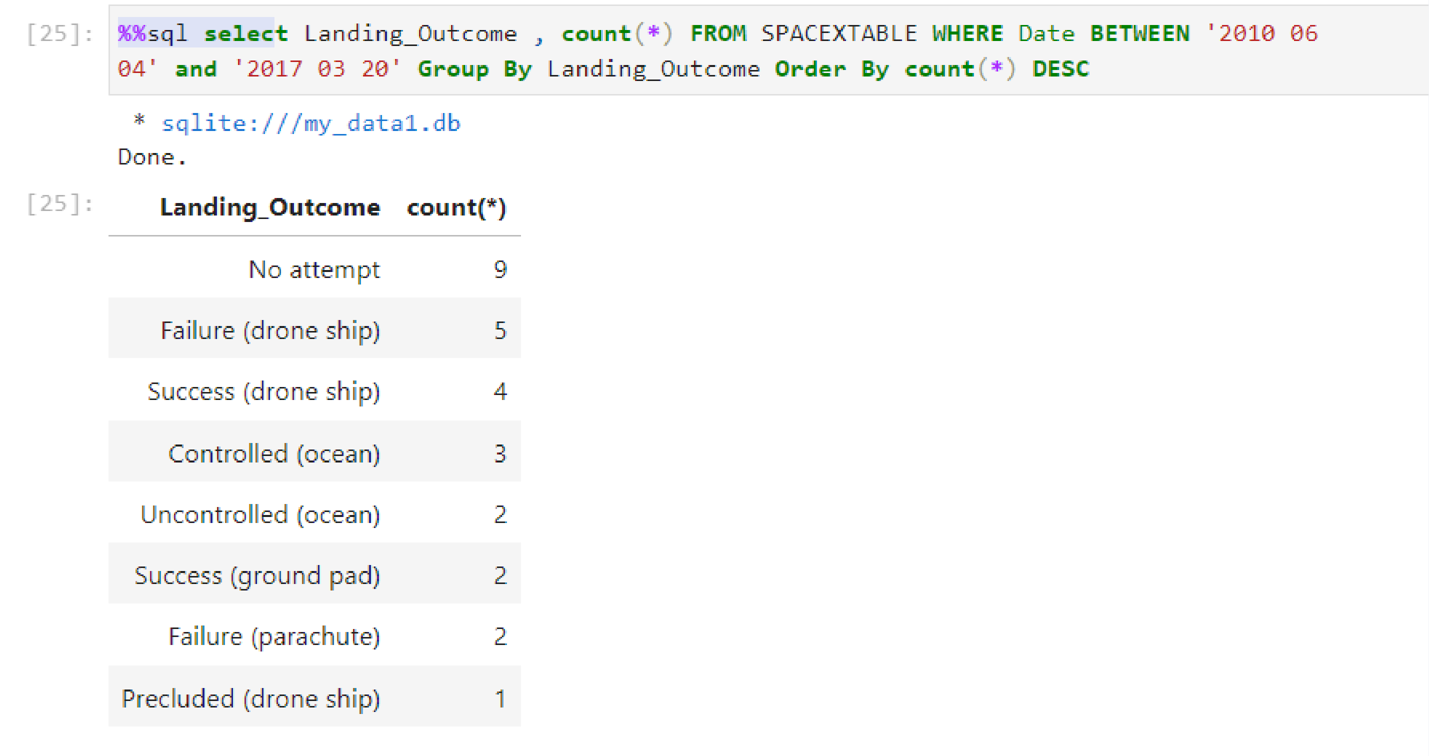


```
[24]: %%sql
      select substr(Date, 6, 2) as Month, Booster_Version, Launch_Site from SPACEXTBL
      where substr(Date,0,5)='2015' and "Landing _Outcome" = "Failure (drone ship)"

       * sqlite:///my_data1.db
      Done.

[24]: Month   Booster_Version   Launch_Site
```

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

Note: There seems to be and problem with the SQL, That’s why the booster versions are not showing.

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



```
[25]: %%sql select Landing_Outcome , count(*) FROM SPACEXTABLE WHERE Date BETWEEN '2010 06
      04' and '2017 03 20' Group By Landing_Outcome Order By count(*) DESC
```

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

[25]:

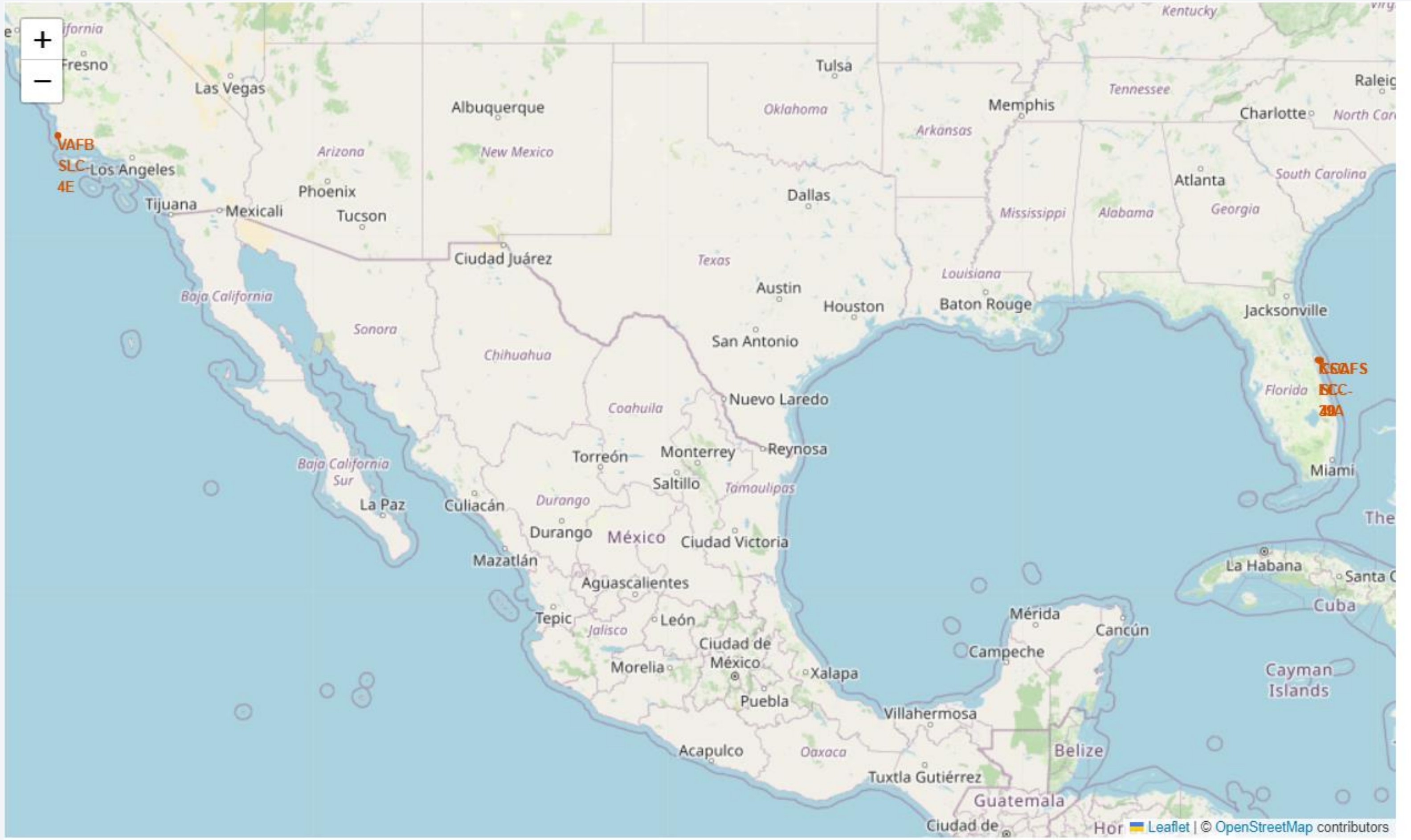
Landing_Outcome	count(*)
No attempt	9
Failure (drone ship)	5
Success (drone ship)	4
Controlled (ocean)	3
Uncontrolled (ocean)	2
Success (ground pad)	2
Failure (parachute)	2
Precluded (drone ship)	1

- The Rank 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.

Section 3

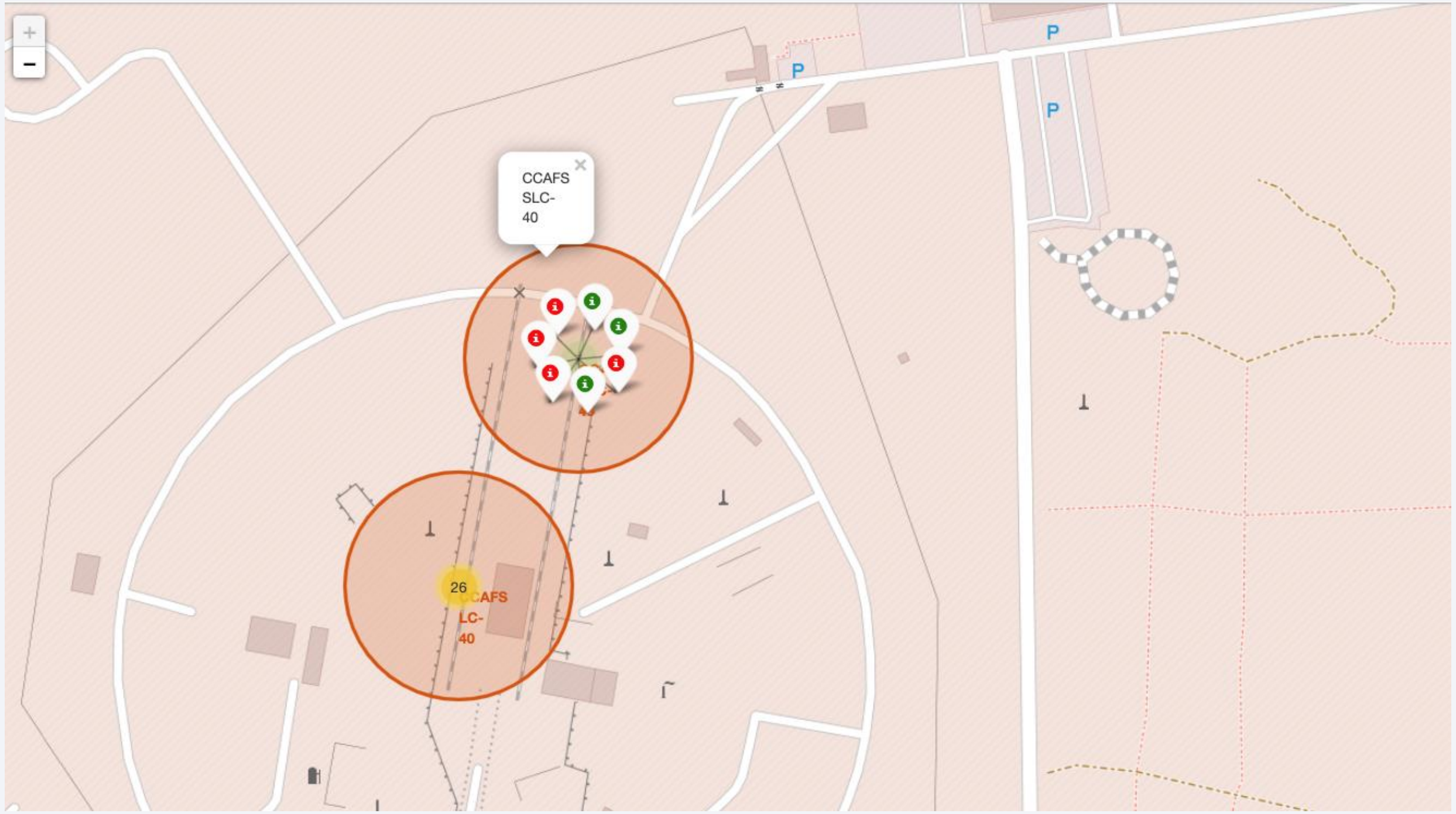
Launch Sites Proximities Analysis

Map of All launch Sites



- This map shows the location of the launch sites, which are near sea, by safety but not too far from roads and railroads.

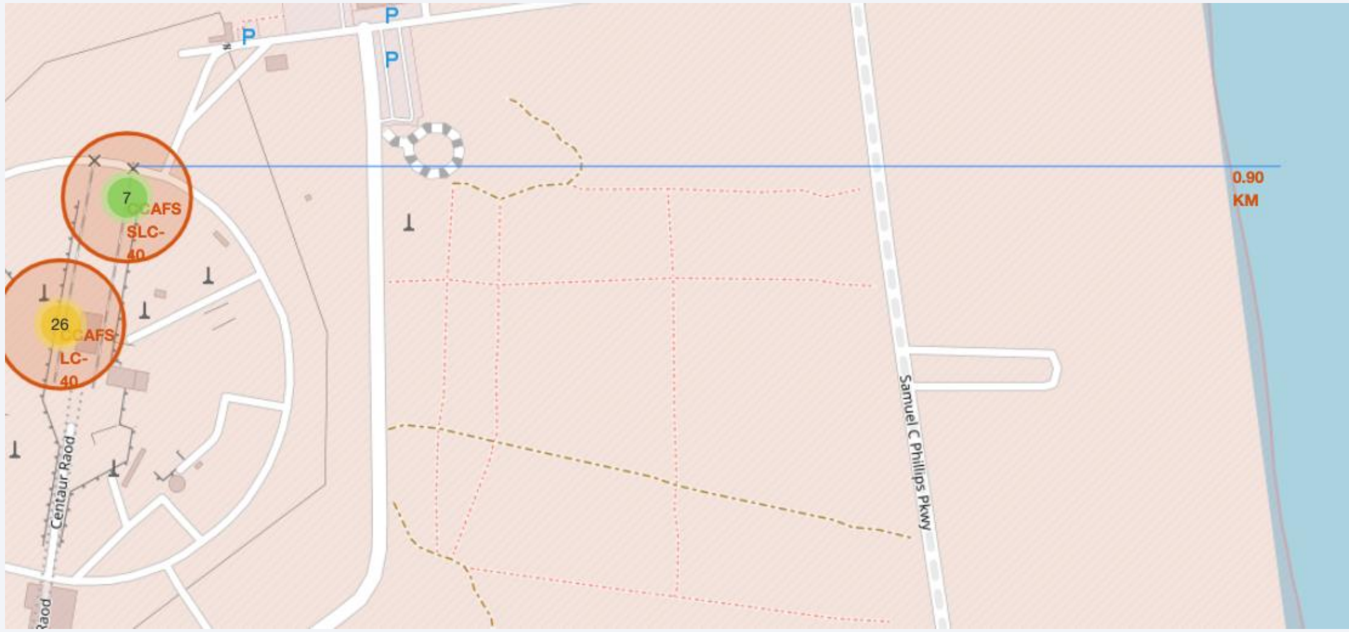
Launch Outcomes by Site



- Here, from the color-labeled markers in marker clusters, you should be able to easily identify which launch sites have relatively high success rates.
- Green Marker = Successful Launch

  Red Marker = Failed Launch

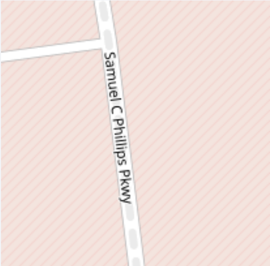
Mapping Launch Site Proximity: Exploring Points of Interest

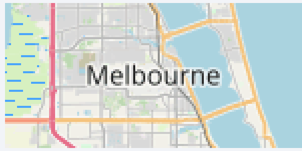


- Discovering how close launch sites are to places like railways, highways, beaches, and cities by using a fancy interactive map.

- The symbols of railway, highway and city looks like this:

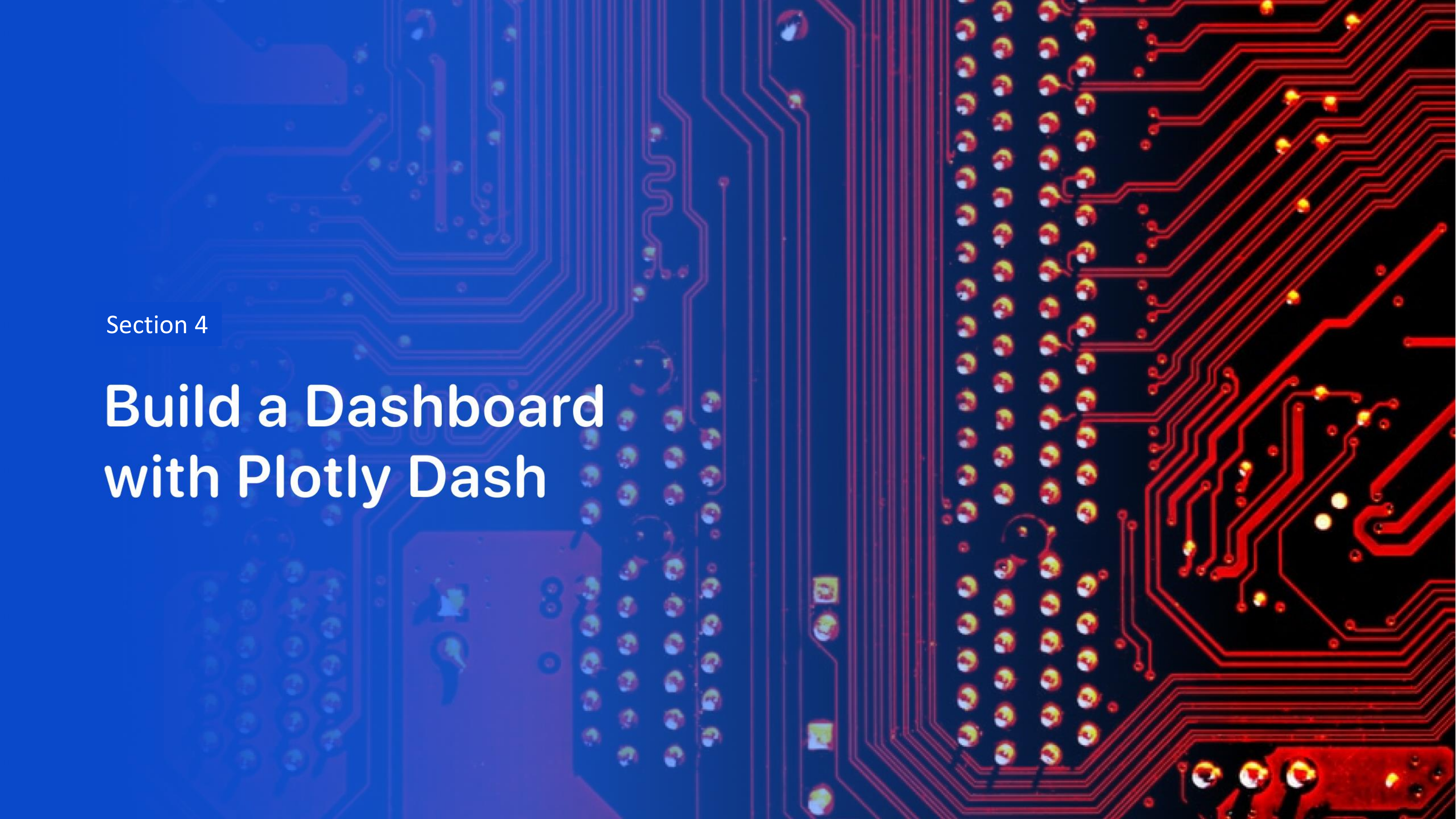
NASA Railroad



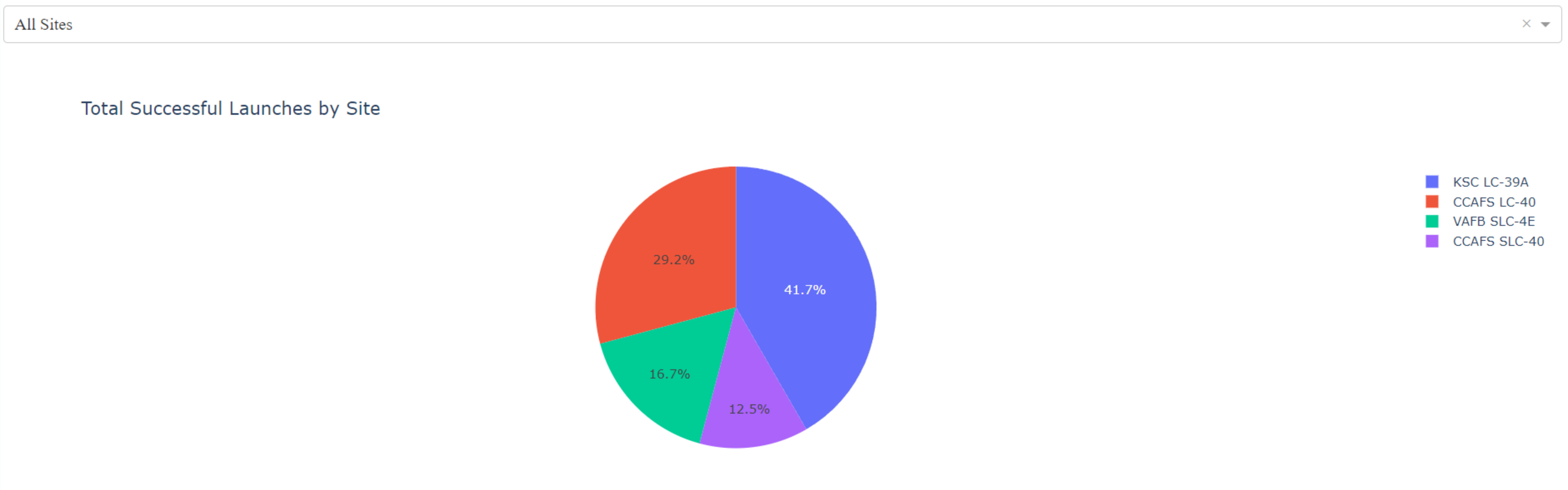


Section 4

Build a Dashboard with Plotly Dash



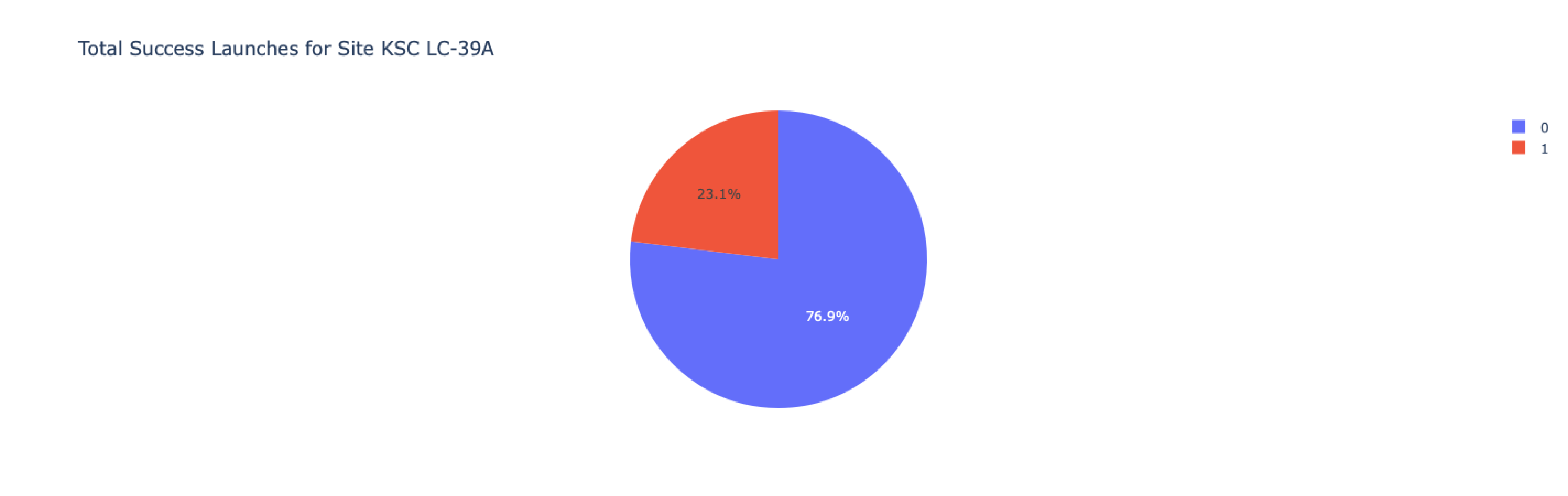
Successful Launches, Site wise



All Sites

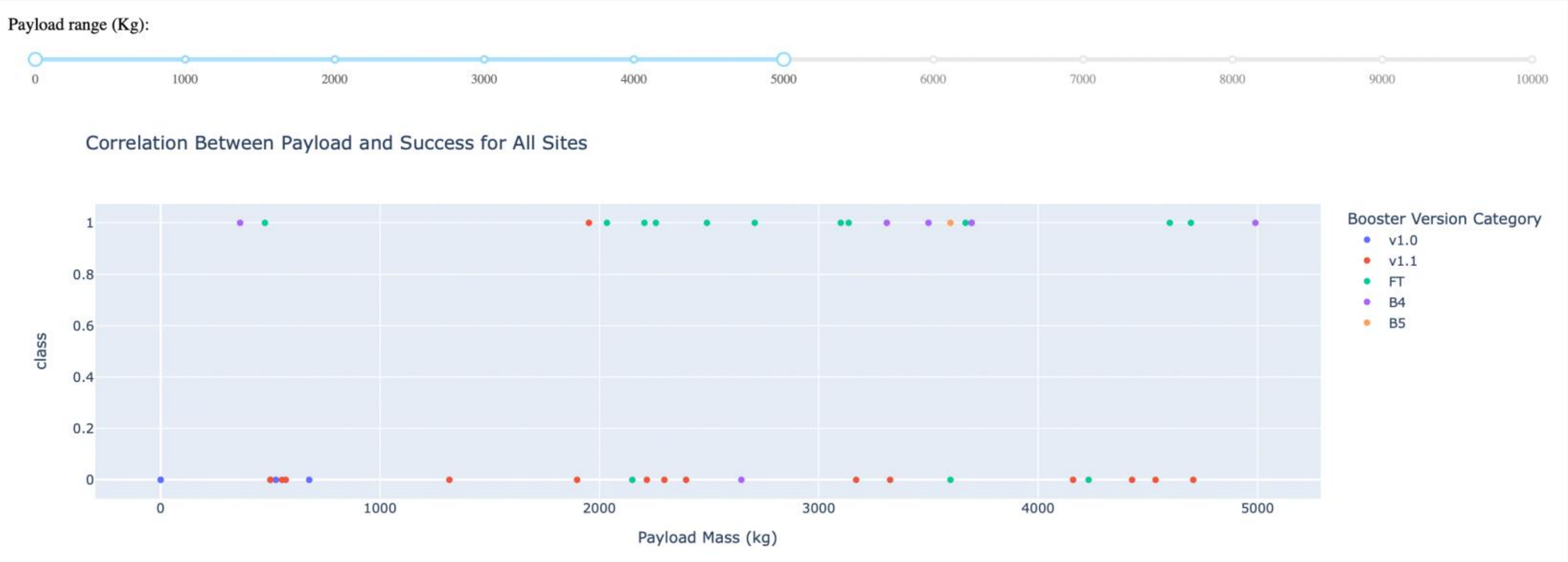
- A pie chart displaying the overall success rates of launches across all sites.

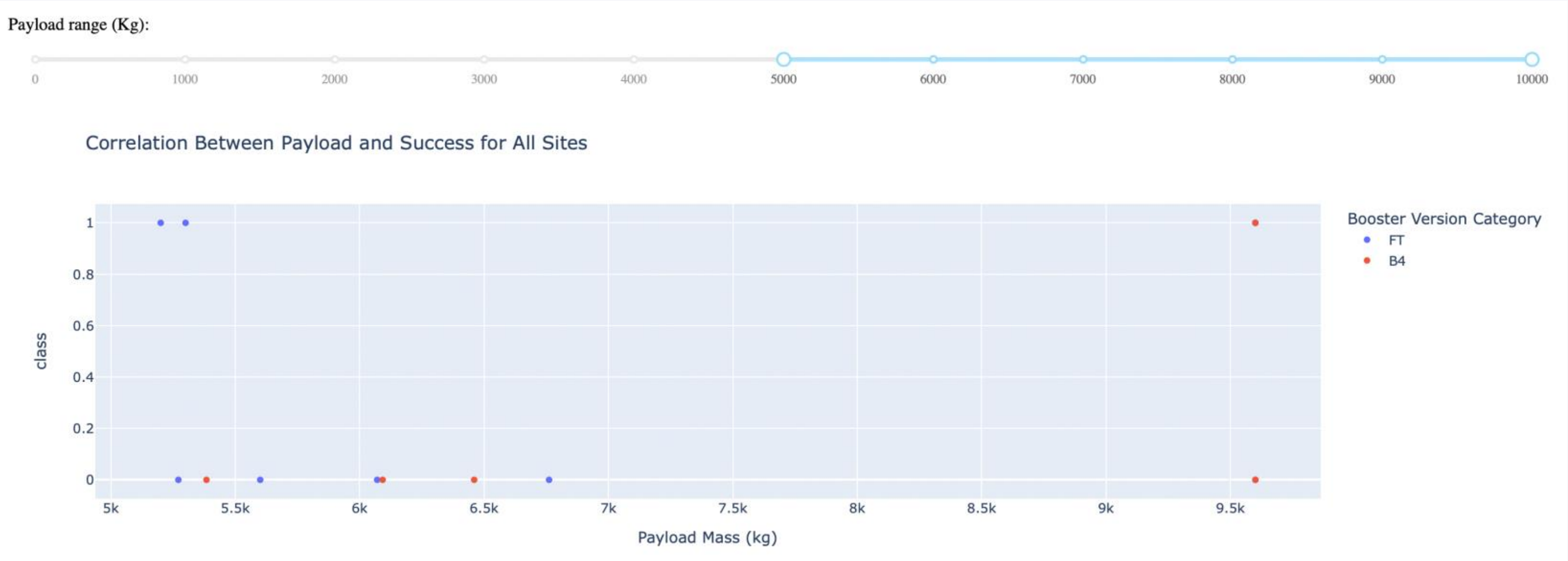
Success Launch Ratio of LC-39A



- KSC LC-39A has the highest launch success rate of 76.9%, with 10 successes and 3 failures, resulting in a failure rate of 23.1%.

Payload vs Success Rate for All





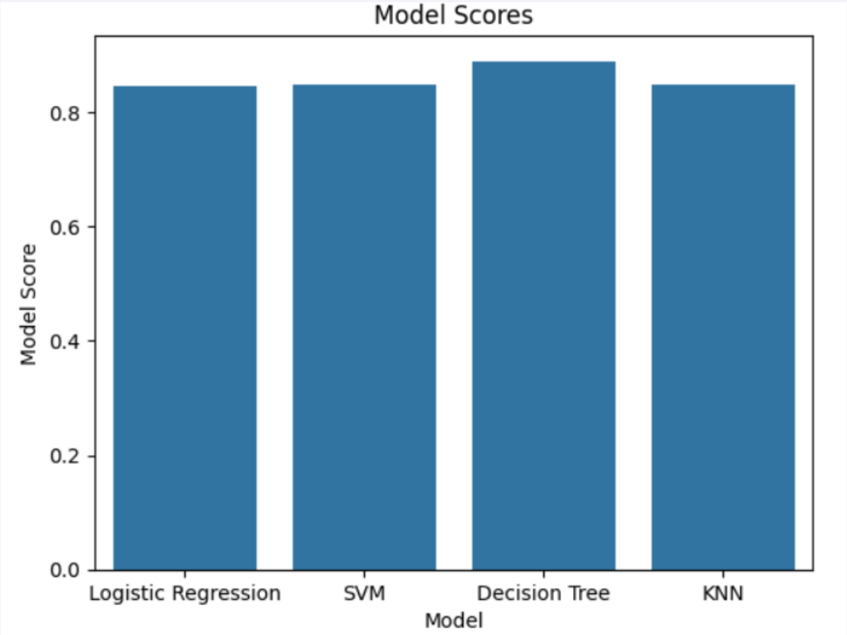
The data indicates that payloads ranging from 2000 to 5500 kg have the highest success rate.

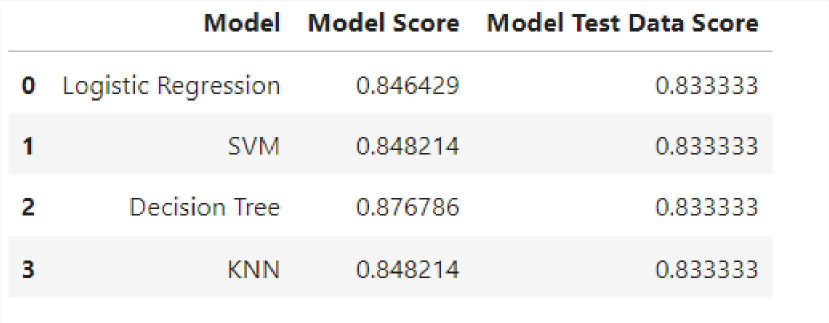
Section 5

Predictive Analysis (Classification)

Classification Accuracy

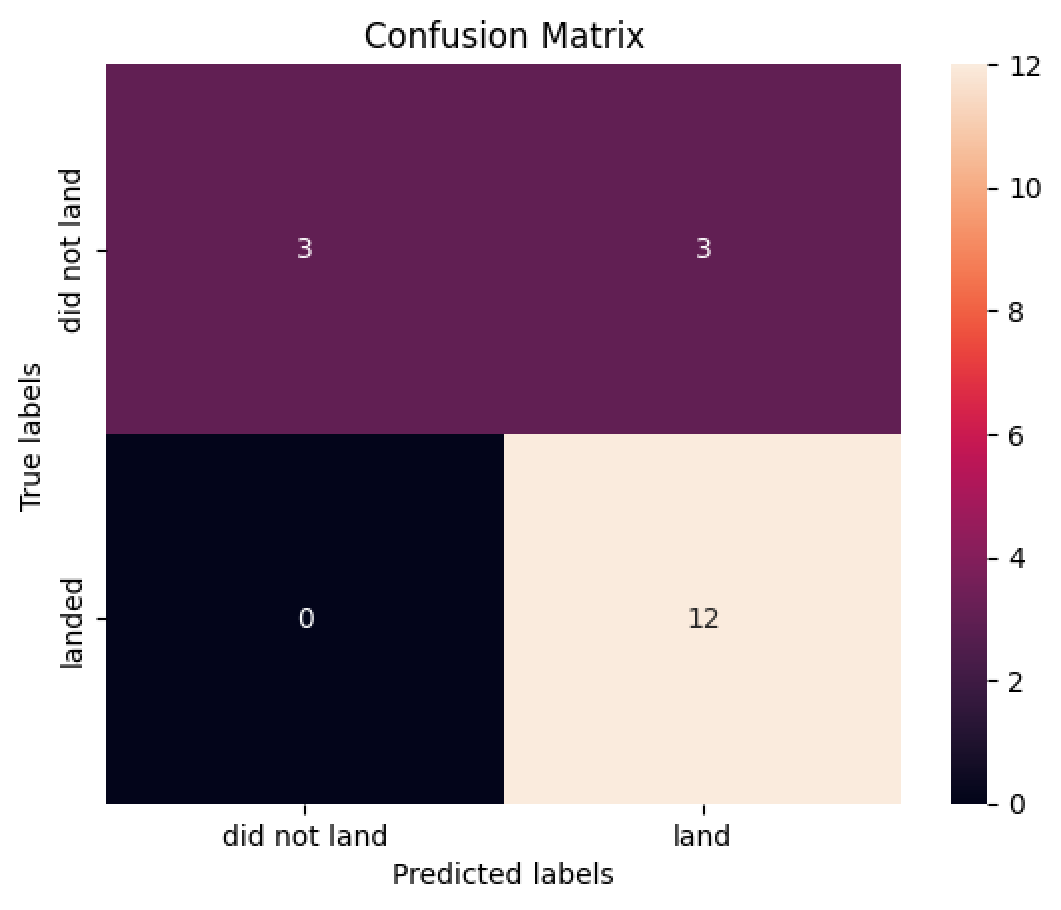
- The Decision Tree Classification Model performed the best among the four models, with an accuracy exceeding 87%.
- Despite similar classification scores(0.83333), the Decision Tree model stood out.





	Model	Model Score	Model Test Data Score
0	Logistic Regression	0.846429	0.833333
1	SVM	0.848214	0.833333
2	Decision Tree	0.876786	0.833333
3	KNN	0.848214	0.833333

Confusion Matrix



- All the confusion matrices showed the same results, predicting the outcomes of 18 launches.
- Among these, 15 were predicted correctly, giving us an accuracy rate of 83.3%. However, 3 predicted successes actually failed, which is 16.7% Type 1 Errors.
- These errors are less ideal than Type 2 Errors and can lead to underestimating launch costs, as fewer rockets may be successfully reused than expected.

Conclusions

- The Decision Tree Model is the best algorithm for this dataset.
- Launches with a payload mass below 7,000 kg tend to have better success rates compared to launches with larger payloads.
- KSC LC-39A stands out with the highest success rate among all launch sites.
- Orbits ES-L1, GEO, HEO, and SSO show a 100% success rate, indicating their reliability.
- Most successful launches had a payload mass between 2,000 kg and 5,500 kg, indicating an optimal payload range for successful missions.
- Success rates improved over the years, with launches from 2013 to 2020 showing an increase from 0% to approximately 80%, reflecting advancements in technology and processes.

Thank you!