IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

The objective of this project is to predict the success or failure of landing the first-stage booster, a core factor in SpaceX's launch costs. To achieve this objective, we systematically implemented the following multifaceted data science methodologies.

Methodologies

- Data Collection
- Data Wrangling
- Exploratory Data Analysis(EDA)
- Interactive Visual Analytics and Dashboard
- Predictive Analysis (Classification)

Results

- Identifying the key factors that determine success.

Introduction

To develop a pricing strategy capable of competing with rival SpaceX, this project analyzes factors from past publicly available launch data to determine whether the landing of their “first-stage booster”—the key to their low costs—will succeed or fail, and then uses machine learning models to predict the outcome.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - The data was obtained via a Python library from SpaceX's REST API and via web scraping pages.
- Perform data wrangling
  - Convert data obtained in JSON or HTML format into a Pandas DataFrame and perform tasks such as imputing missing values.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Using machine learning, determine whether the Falcon 9 first stage will successfully land.

Data Collection

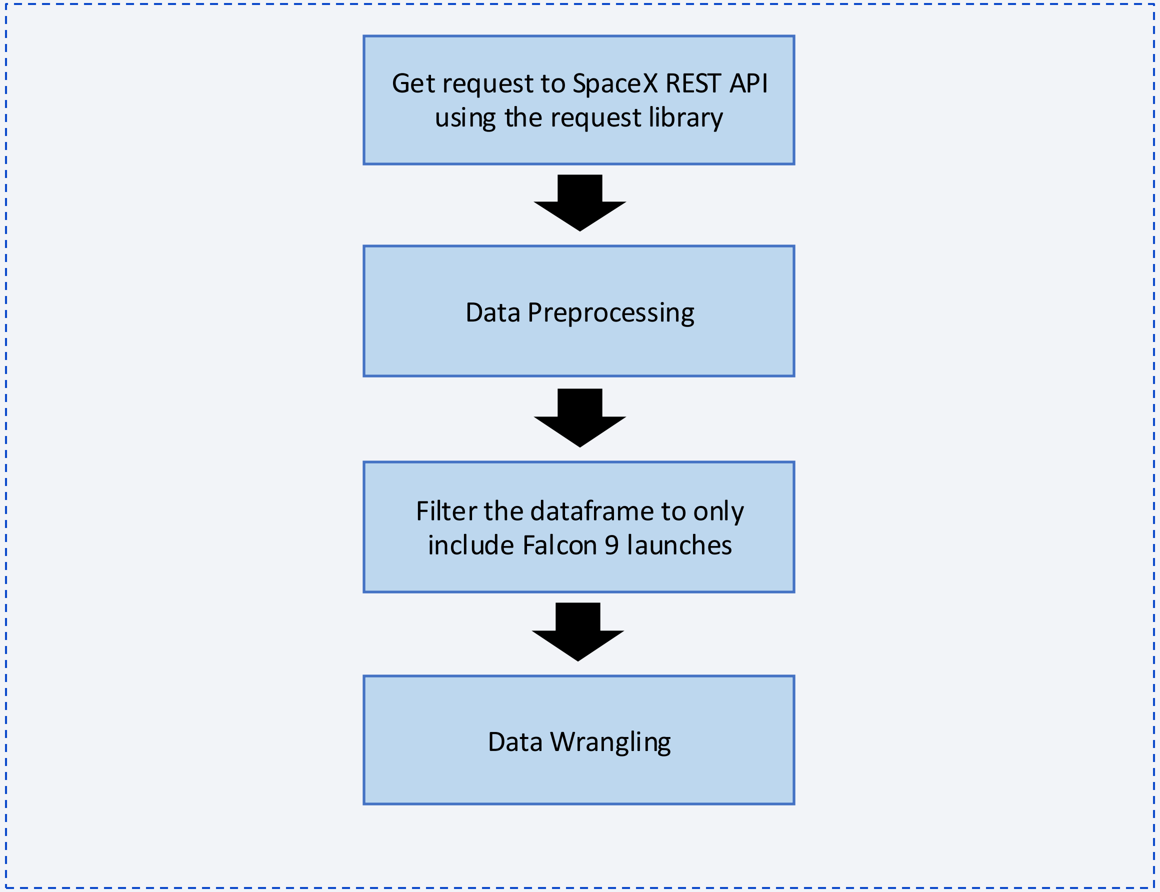
- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose

- The GitHub URL of the completed SpaceX API calls notebook: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Get request to SpaceX REST API using the request library

Data Preprocessing

Filter the dataframe to only include Falcon 9 launches

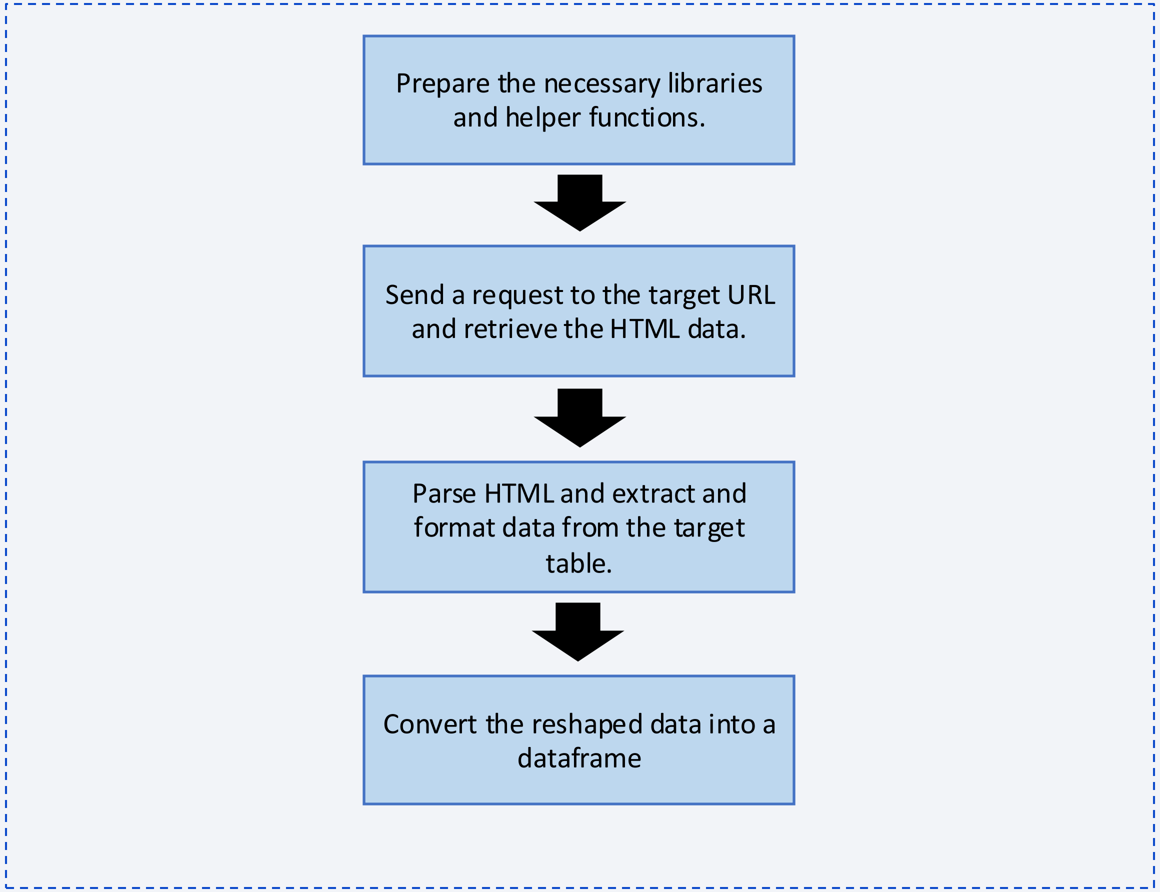
Data Wrangling

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

- The GitHub URL of the completed web scraping notebook: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/jupyter-labs-webscraping.ipynb



Prepare the necessary libraries and helper functions.

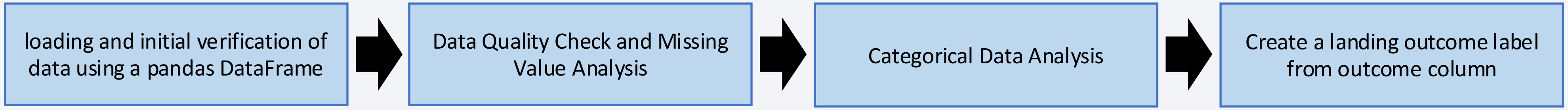
Send a request to the target URL and retrieve the HTML data.

Parse HTML and extract and format data from the target table.

Convert the reshaped data into a dataframe

Data Wrangling

- Convert data obtained in JSON or HTML format into a Pandas DataFrame and perform tasks such as imputing missing values.



loading and initial verification of data using a pandas DataFrame

Data Quality Check and Missing Value Analysis

Categorical Data Analysis

Create a landing outcome label from outcome column

- The GitHub URL of the completed Data Wrangling notebook: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- **Catplot 1:** Flight Number vs Payload Mass to identify landing success patterns across differentpayload weights
- **Catplot 2:** Flight Number vs Launch Site to analyze success rates across different launchlocations
- **Catplot 3:** Payload Mass vs Launch Site to examine payload capacity and success relationships by location
- **Barchart:** Success Rate by Orbit Type to compare landing success across different orbital paths
- **Catplot 4:** Flight Number vs Orbit Type to understand experience-based success patterns by orbit
- **Catplot 5:** Payload Mass vs Orbit Type to analyze payload effects on success rates by orbit
- **Line chart:** Launch Success Yearly Trend to show temporal progression of landing success rates

The GitHub URL of the completed EDA with data visualization notebook: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/edadataviz.ipynb

EDA with SQL

- SQL queries performed:
- Query: SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL
    - Perform: Display unique launch sites
- Query: SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
    - Perform: Display 5 records where launch sites begin with 'CCA'
- Query: SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
    - Perform: Calculate total payload mass for NASA (CRS) missions
- Query: SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION='F9 v1.1'
    - Perform: Calculate average payload mass for F9 v1.1 booster version
- Query: SELECT MIN(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME='Success (ground pad)'
    - Perform: Find first successful ground pad landing date

EDA with SQL

- SQL queries performed:
- Query: SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND LANDING_OUTCOME='Success (drone ship)'
  - Perform: List boosters with successful drone ship landings and payload mass between 4000-6000 kg
- Query: SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
  - Perform: Count successful and failure mission outcomes
- Query: SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
  - Perform: List booster versions with maximum payload mass using subquery

EDA with SQL

- SQL queries performed:

- Query: SELECT strftime('%m', DATE) AS MONTH_NUMBER, \ CASE strftime('%m', DATE) \ WHEN '01' THEN 'January' \ WHEN '02' THEN 'February' \ WHEN '03' THEN 'March' \ WHEN '04' THEN 'April' \ WHEN '05' THEN 'May' \ WHEN '06' THEN 'June' \ WHEN '07' THEN 'July' \ WHEN '08' THEN 'August' \ WHEN '09' THEN 'September' \ WHEN '10' THEN 'October' \ WHEN '11' THEN 'November' \ WHEN '12' THEN 'December' \ END AS MONTH_NAME,LANDING_OUTCOME, BOOSTER_VERSION, LAUNCH_SITE, DATE FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Failure (drone ship)' AND DATE LIKE '%2015%'
    - Perform: Display month names, failure landing outcomes, booster versions, and launch sites for 2015
- Query: SELECT "LANDING_OUTCOME", COUNT(*) AS "COUNT" FROM SPACEXTBL WHERE "DATE" BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "LANDING_OUTCOME" ORDER BY "COUNT" DESC;
    - Perform: Rank landing outcomes count between 2010-06-04 and 2017-03-20 in descending order

The GitHub URL of the completed EDA with SQL notebook: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map

- Markers: NASA Johnson Space Center, each launch site, launch success or failure, and markers indicating nearby facilities (coastline, cities, etc.) have been installed.
- Circles: A circle indicating a specific radius was drawn around NASA Johnson Space Center and each launch site.
- Lines: A blue line indicating the distance between the launch site and adjacent facilities, and a polyline showing the straight-line connection between locations, have been added.
- Marker clusters: When multiple launch result markers exist at the same coordinates, they were grouped into a single cluster.

Build an Interactive Map with Folium

Explain why you added those objects

- These objects were added to create an interactive map that visualizes SpaceX launch site geographic patterns, analyzes launch success rates by location, measures distances to key proximities (coastline, cities, infrastructure), and enables users to explore spatial relationships for optimal launch site selection.

The GitHub URL of the completed build an interactive map with folium notebook: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Summarize what plots/graphs and interactions you have added to a dashboard

- **Launch Site Selection Dropdown:** Added a dropdown menu that allows users to select either all sites or specific launch sites (CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40)
- **Success/Failure Pie Chart**: Added a pie chart that displays the count of successful and failed launches for the selected site
- **Payload Range Slider:** Added a slider that allows users to select payload mass range from 0kg to 10,000kg
- **Payload vs Success Correlation Scatter Plot:** Added a scatter plot that shows the relationship between payload mass and launch success, color-coded by booster version category
- **Dynamic Site Filtering:** Added functionality where the pie chart and scatter plot content dynamically change based on the site selected in the dropdown
- **Dynamic Payload Filtering:** Added functionality where the scatter plot data is dynamically filtered based on the payload range selected with the slider

Build a Dashboard with Plotly Dash

Explain why you added those plots and interactions

- The purpose of this dashboard is to provide users with a tool that allows them to interactively explore data and discover insights on their own.

The GitHub URL of your completed Plotly Dash lab: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/spacex_launch_dash/spacex-dash-app.py

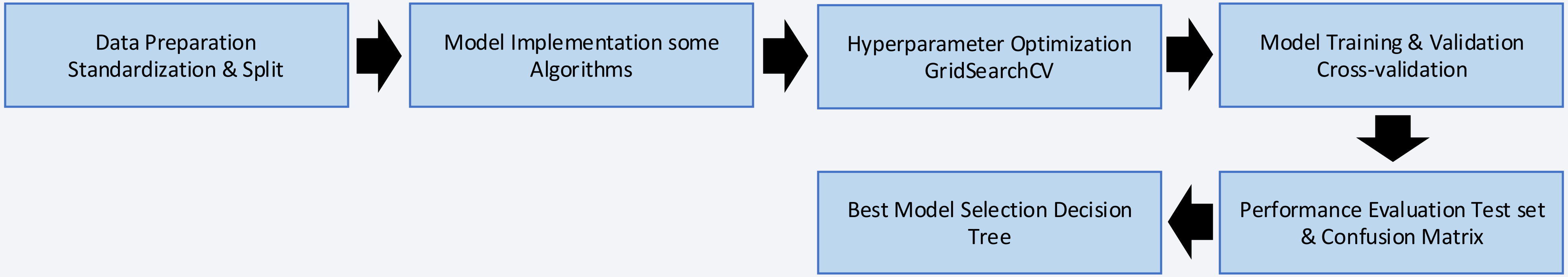
Predictive Analysis (Classification)

Summarize how you built, evaluated, improved, and found the best performing classification model

- Model Building: Standardized data was split into training and test sets at an 8:2 ratio, and four classification models (logistic regression, SVM, decision tree, KNN) were implemented.
- Model Evaluation: Evaluated the performance of each model using 10-fold cross-validation and confusion matrices to measure their generalization ability on the test data.
- Model Improvement: GridSearchCV was executed to comprehensively explore the hyperparameters of each algorithm and optimize their respective prediction accuracy.
- Best Performing Model Identification: Based on the comparative evaluation results, the decision tree achieved the highest accuracy rate (87.5%), outperforming other models, and was therefore selected as the optimal prediction model.

Predictive Analysis (Classification)

You need present your model development process using key phrases and flowchart



Data Preparation Standardization & Split

Model Implementation some Algorithms

Hyperparameter Optimization GridSearchCV

Model Training & Validation Cross-validation

Performance Evaluation Test set & Confusion Matrix

Best Model Selection Decision Tree

The GitHub URL of your completed predictive analysis lab: https://github.com/ryoosuke/applied-data-science-capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

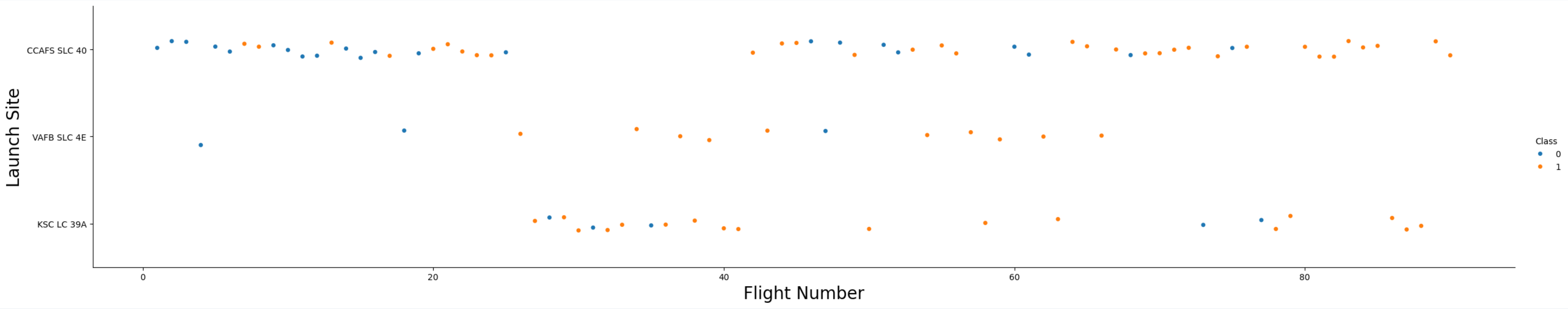
Results

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

Section 2

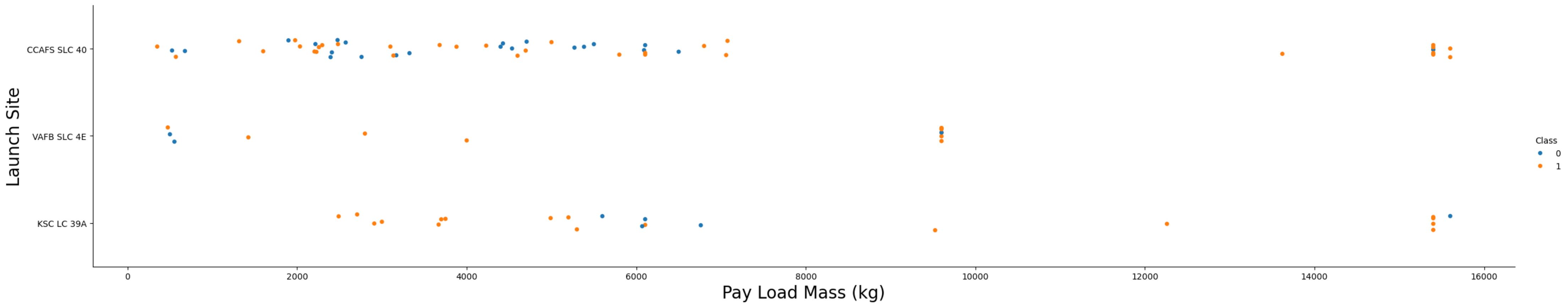
Insights drawn from EDA

Flight Number vs. Launch Site



- This scatter plot visualizes the relationship between Flight Number and Launch Site.
- As the number of flights increases, the first-stage landing success rate tends to improve across the entire launch site.

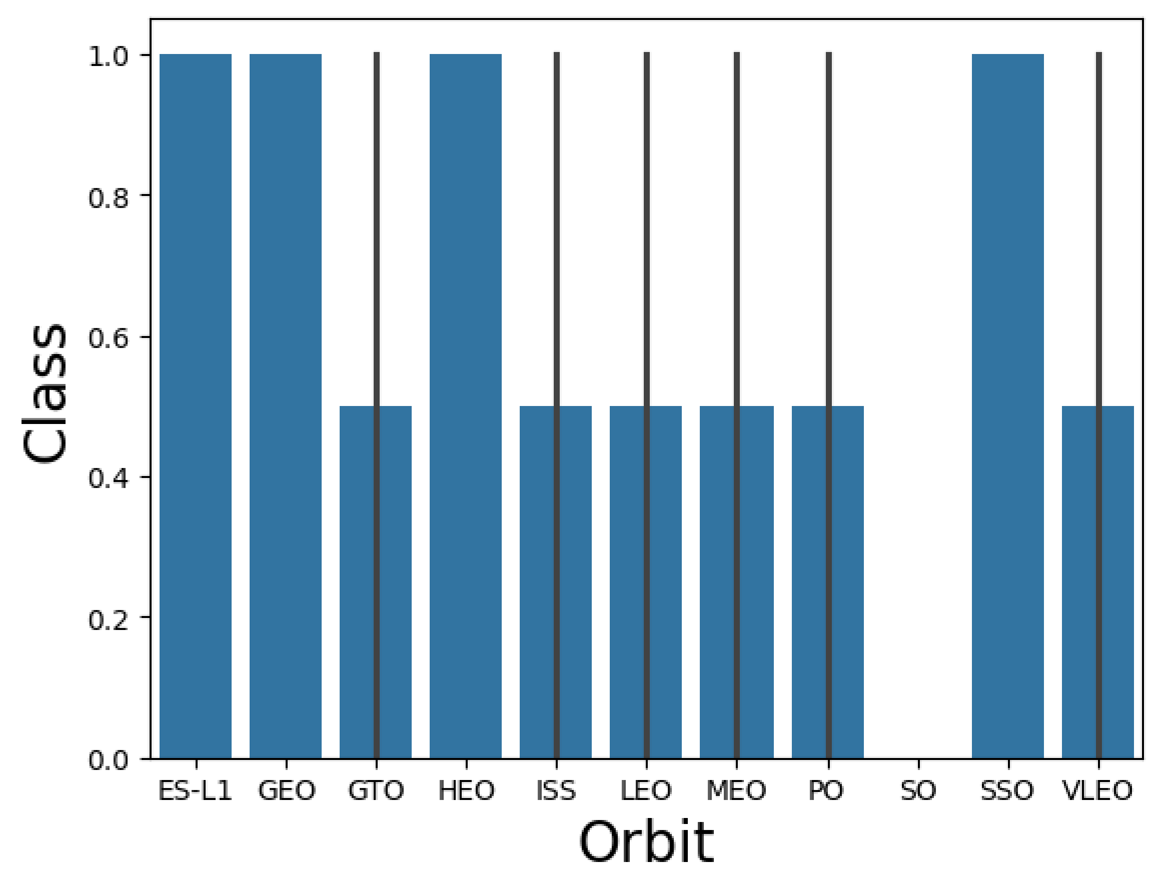
Payload vs. Launch Site



- This scatter plot visualizes the relationship between Payload and Launch Site.
- An overall trend shows that launch success rates increase with greater payload mass, and it is evident that the CCAFS SLC-40 launch site has conducted the most launches across a diverse range of payload weights.

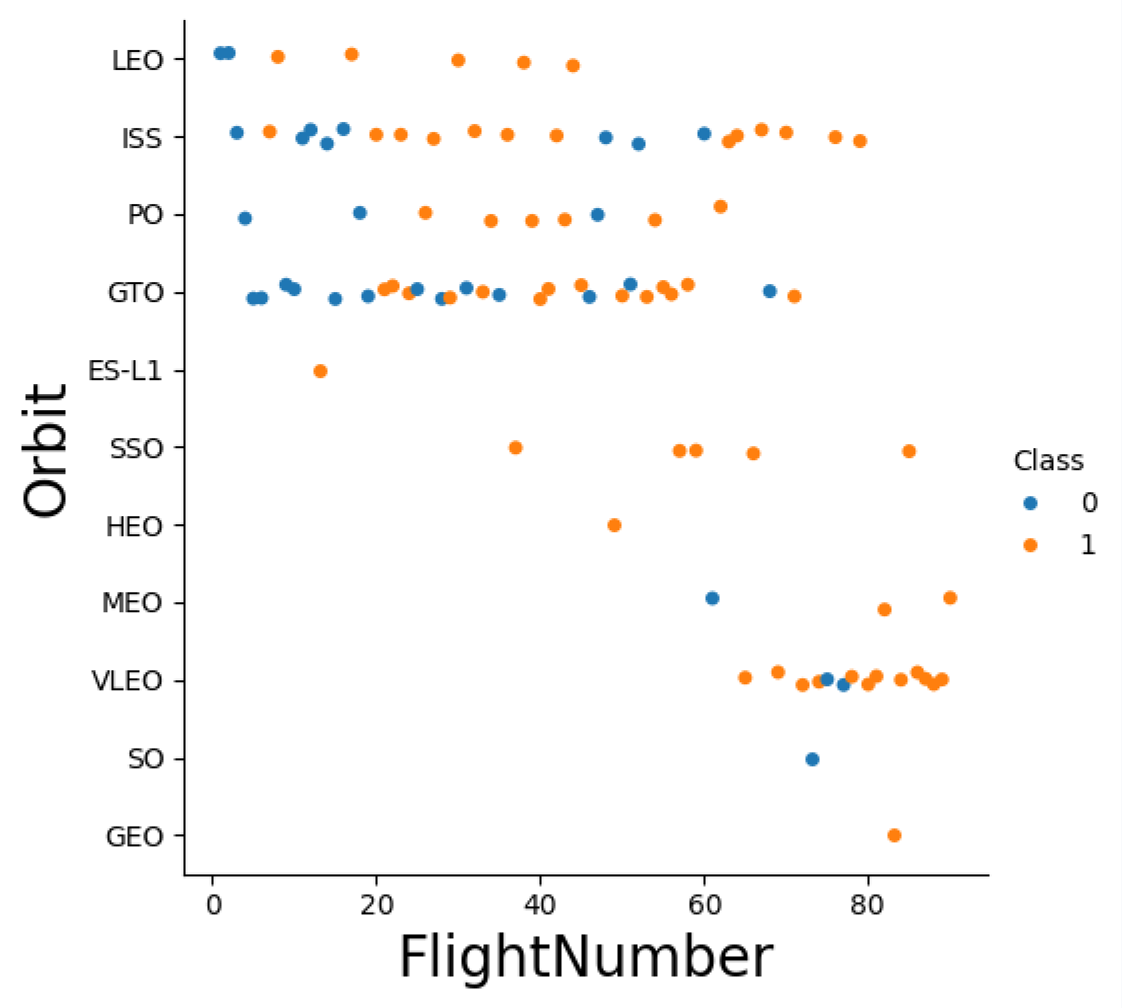
Success Rate vs. Orbit Type

- This bar chart shows the relationship between Success Rate and Orbit Type.
- Launches to specific orbits such as ES-L1, GEO, HEO, and SSO have a 100% success rate, while success rates for other orbits like GTO and the ISS remain at 50%. This strongly suggests that the target orbit is critically linked to mission success.



Flight Number vs. Orbit Type

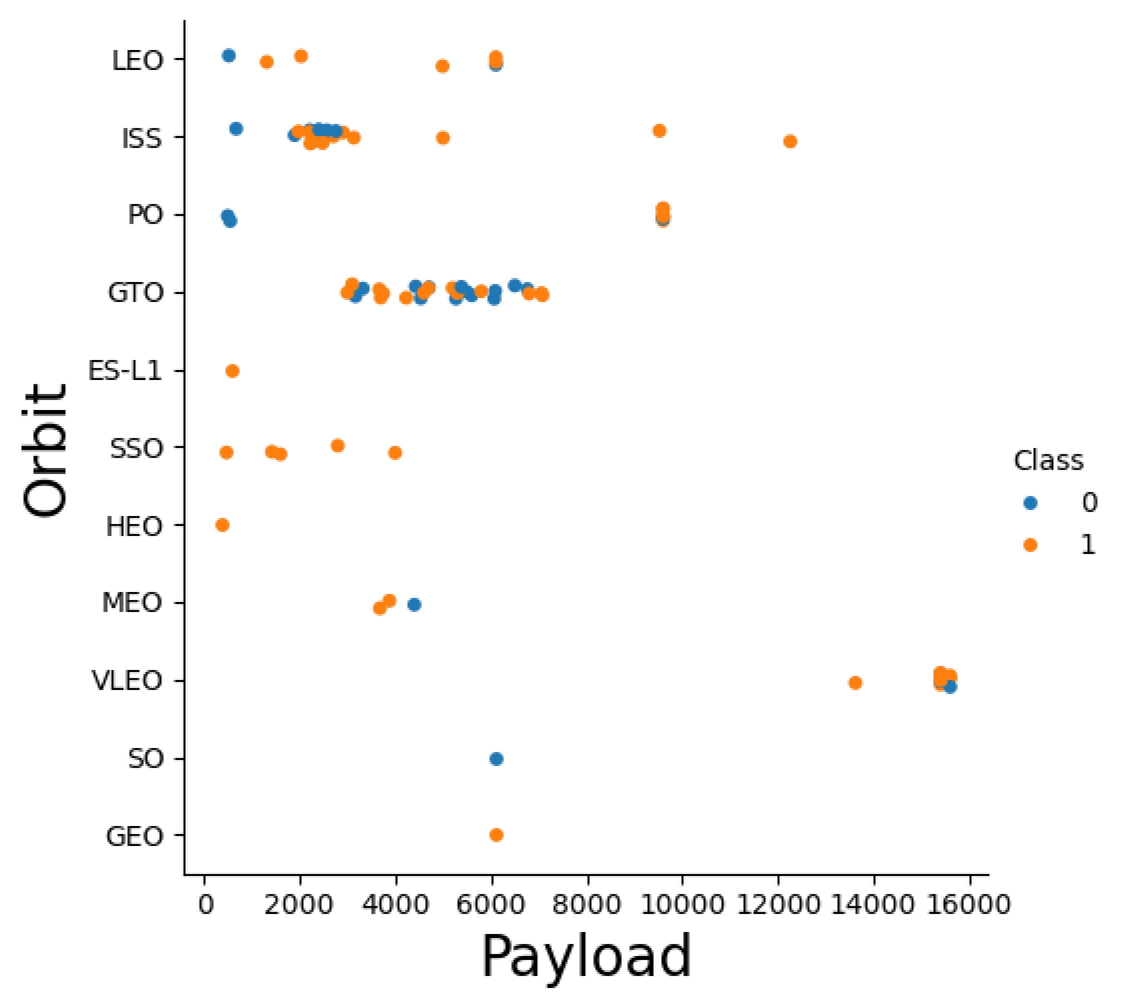
- This scatter plot shows the relationship between Flight Number and Orbit Type.
- As the number of flights increases, the proportion of successful missions rises, and it is evident that success rates improve with experience, particularly for specific orbits such as GTO and VLEO.



Payload vs. Orbit Type

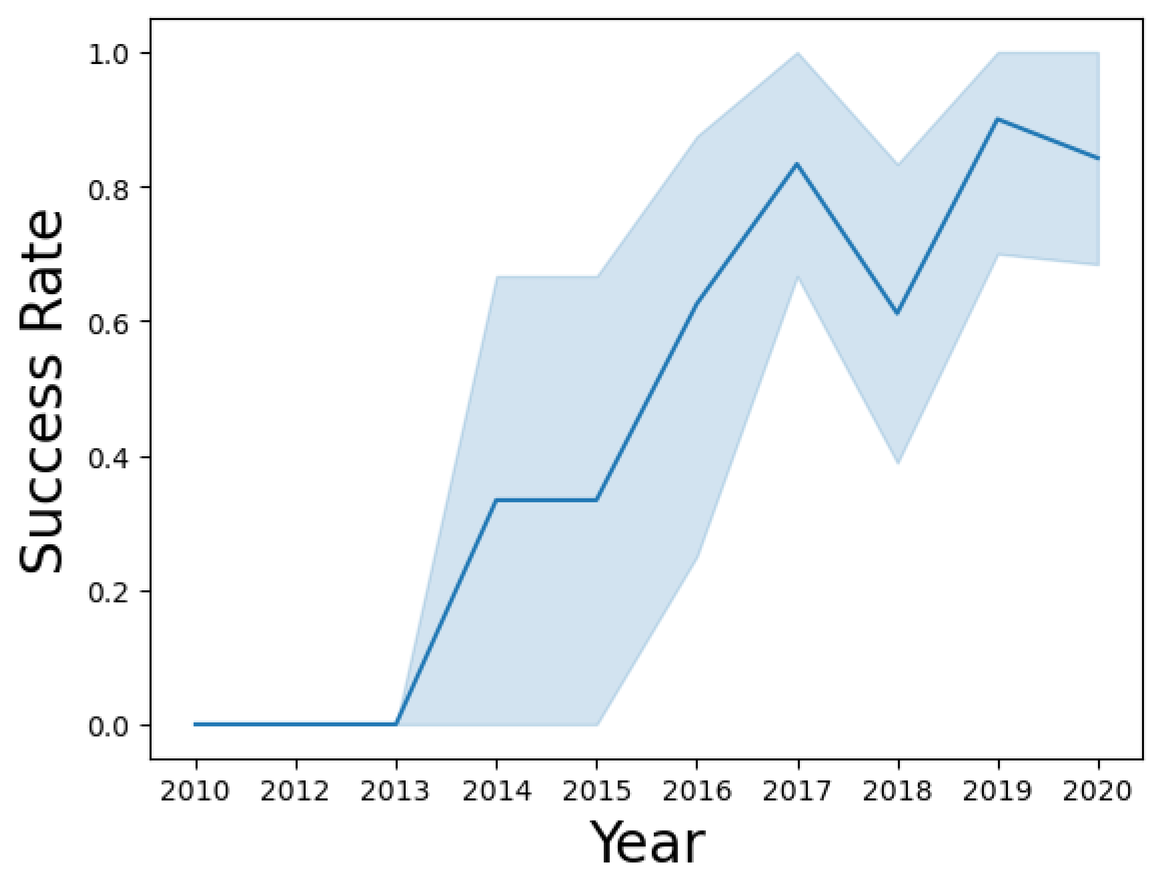
- This scatter plot shows the relationship between Payload and Orbit Type.

- Although no clear correlation exists between payload mass and launch success, it is evident that launches with diverse payload masses are conducted for specific orbits such as the ISS and GTO.



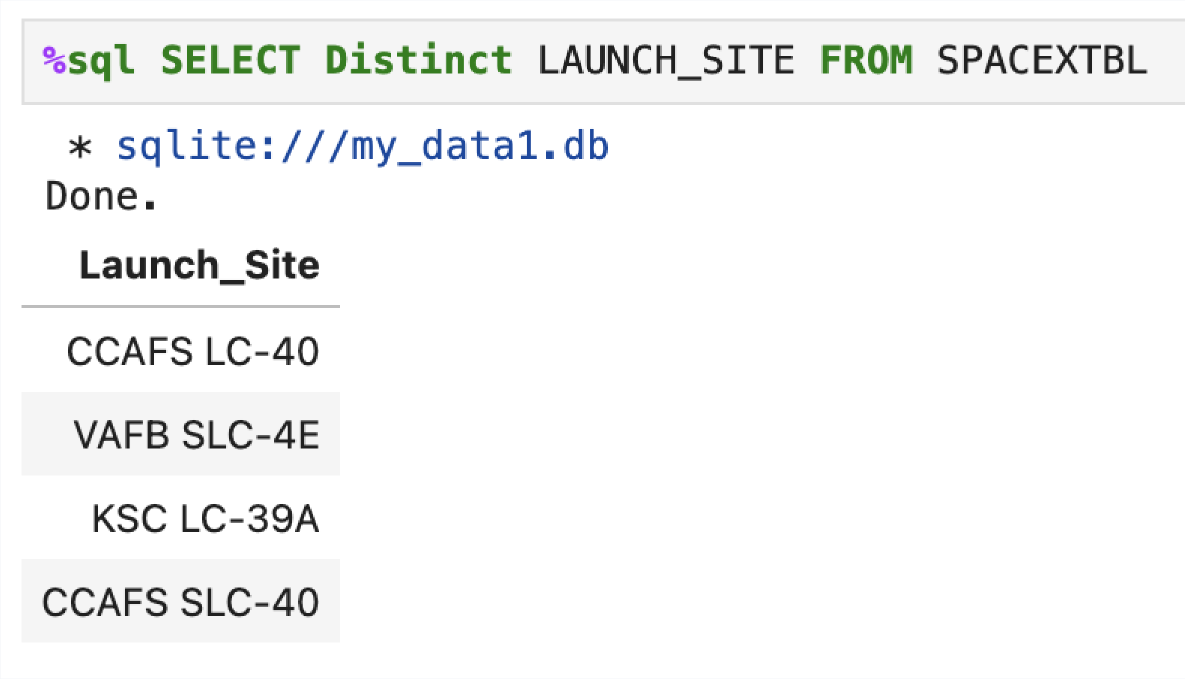
Launch Success Yearly Trend

- This line chart shows yearly average success rate
- Since 2013, the launch success rate has steadily improved year by year, suggesting that technological maturity and accumulated experience have directly contributed to this increase.



All Launch Site Names

- Find the names of the unique launch sites
- DISTINCT displays unique launch sites.
- This result found four launch sites.

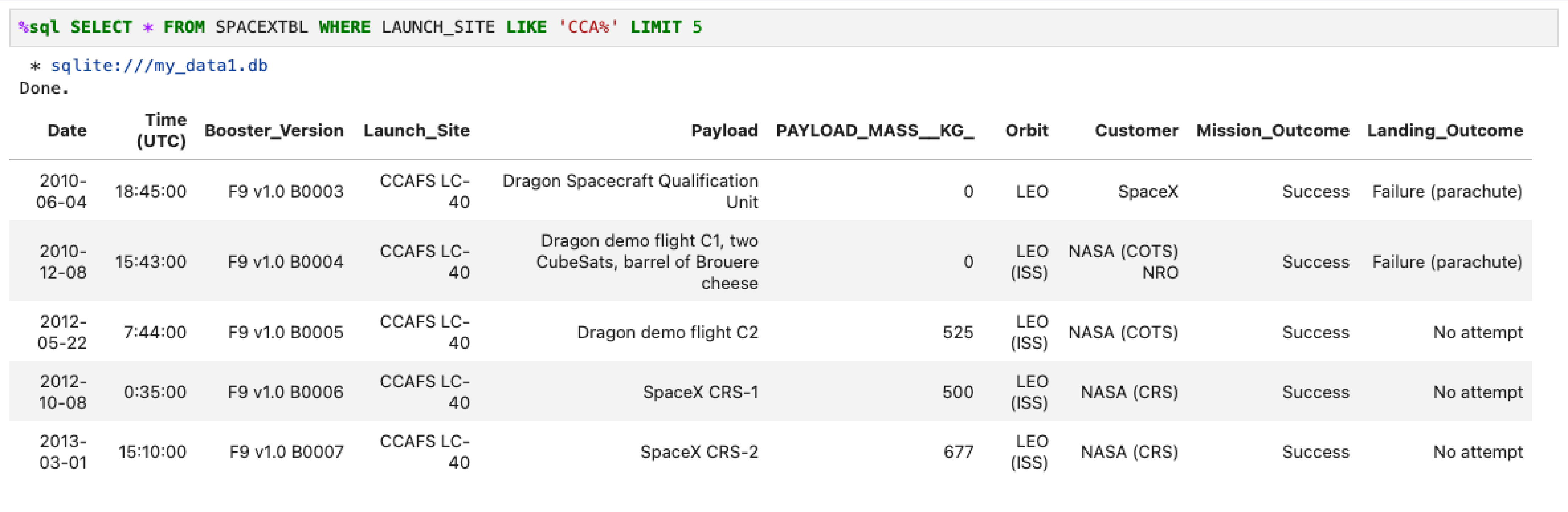


```
%sql SELECT Distinct LAUNCH_SITE FROM SPACEXTBL
```

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

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

Launch Site Names Begin with 'CCA'



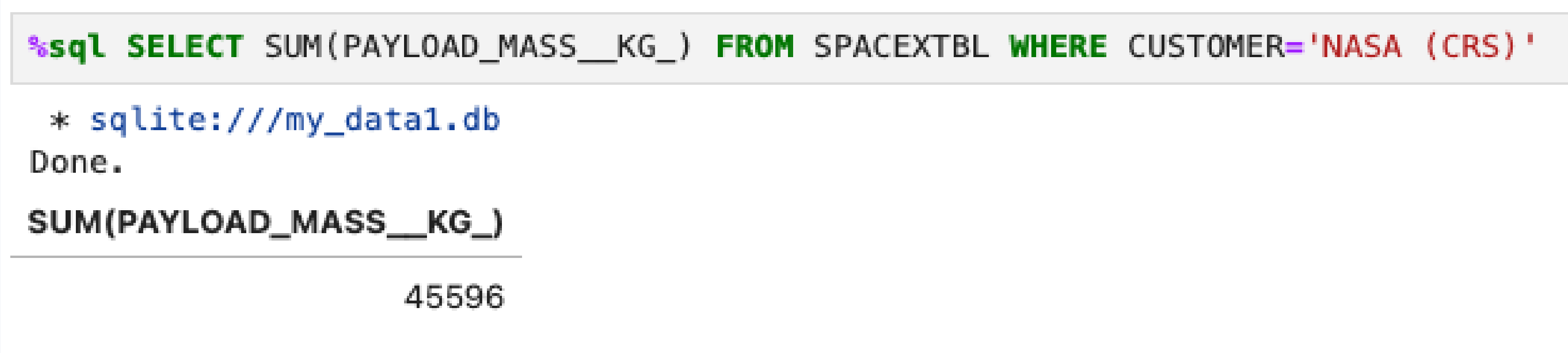
```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

```
 * 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

- This query displays five records for Launch Sites whose names begin with “CCA”.

Total Payload Mass



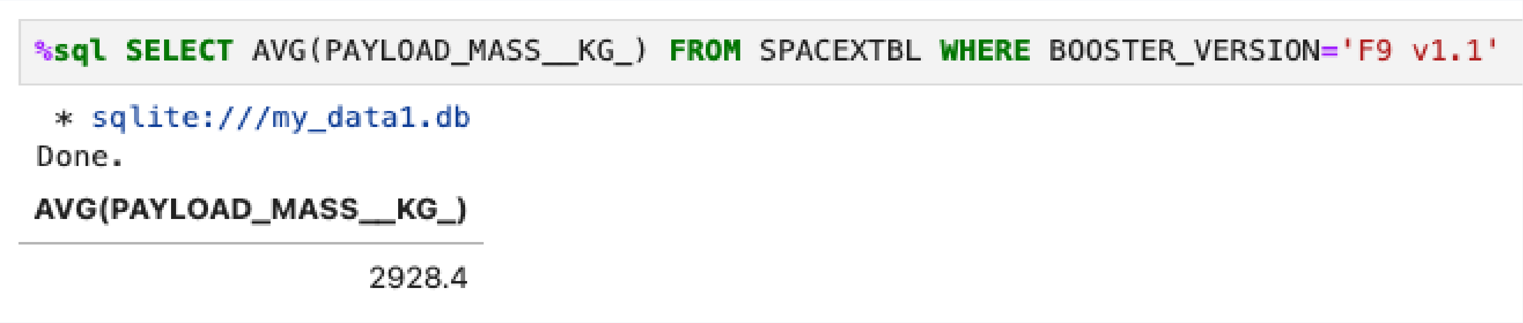
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
```

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

SUM(PAYLOAD_MASS__KG_)
45596

- This query extracts all launch missions where CUSTOMER is 'NASA (CRS)' and sums the total payload mass carried by the boosters launched.

Average Payload Mass by F9 v1.1



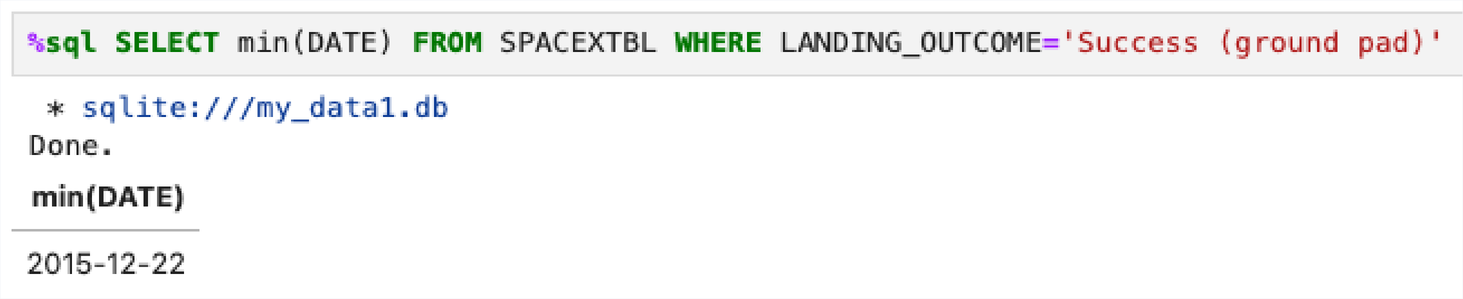
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION='F9 v1.1'
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

- This query extracts all launch missions where BOOSTER_VERSION is 'F9 v1.1' and averages the total payload mass carried by the boosters launched.

First Successful Ground Landing Date



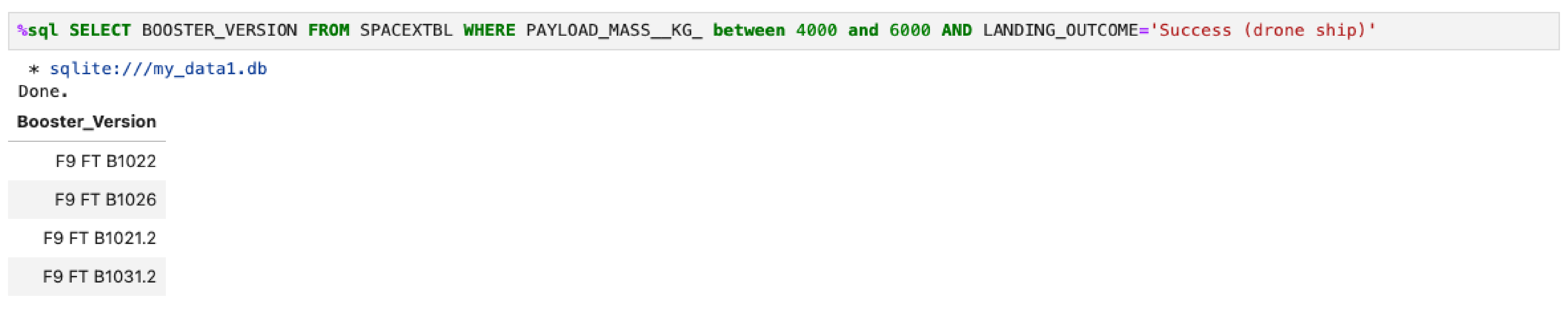
```
%sql SELECT min(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME='Success (ground pad)'

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

min(DATE)
2015-12-22

- This query identifies the oldest date from all missions where the LANDING_OUTCOME was 'Success (ground pad)'

Successful Drone Ship Landing with Payload between 4000 and 6000



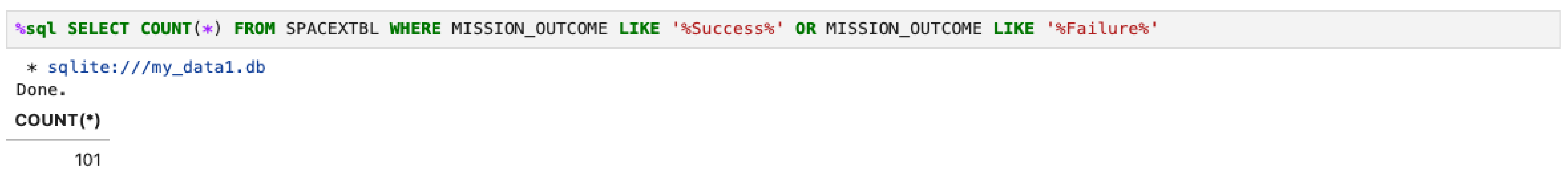
```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 AND LANDING_OUTCOME='Success (drone ship)'
```

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

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

- This query lists the BOOSTER_VERSION used in missions where the payload mass was between 4000kg and 6000kg and the landing on the drone ship was successful.

Total Number of Successful and Failure Mission Outcomes

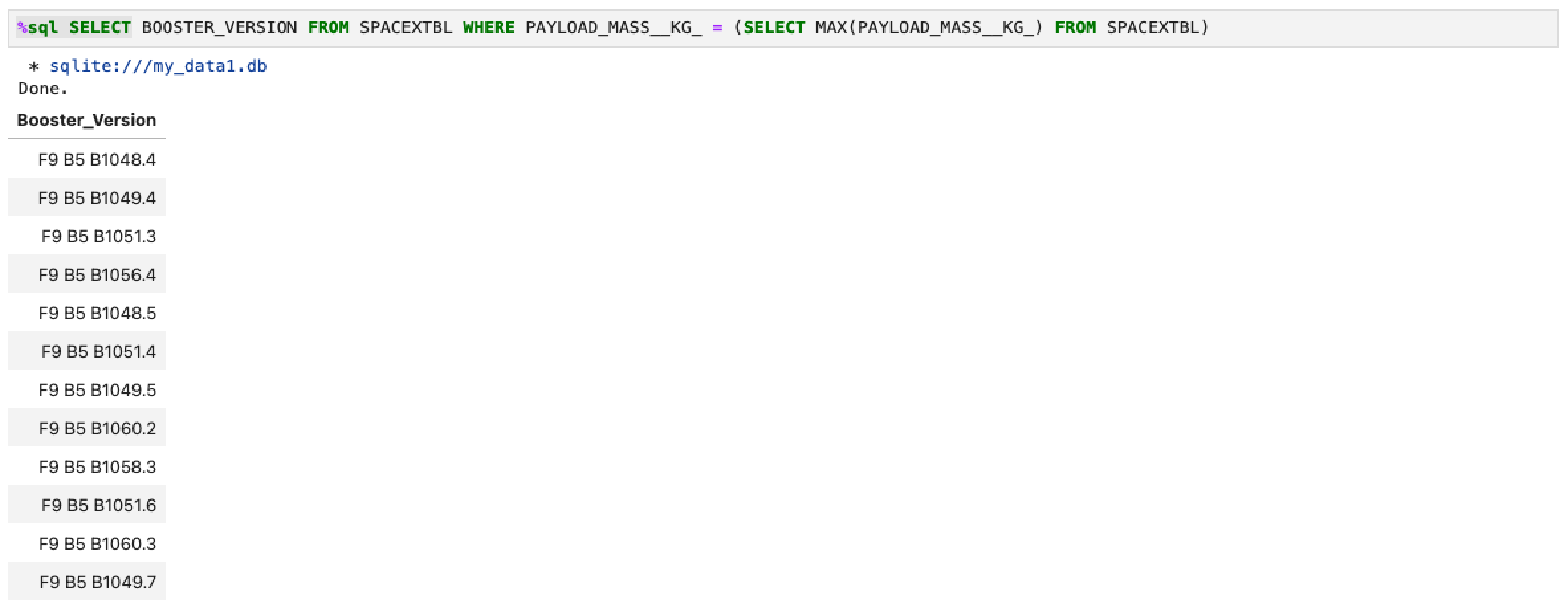


```
%sql SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
```

```
 * sqlite:///my_data1.db
Done.
COUNT(*)
      101
```

- This query counts the total number of records where the MISSION_OUTCOME field contains the strings 'Success' or 'Failure'.

Boosters Carried Maximum Payload



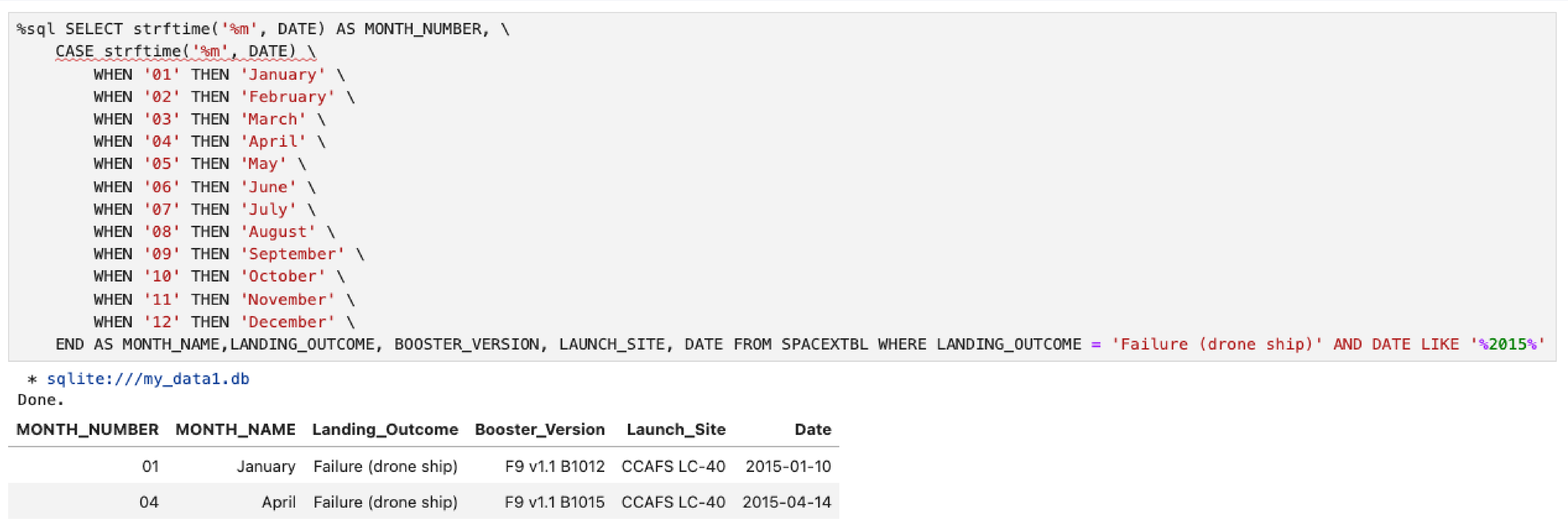
```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

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

- This query uses a subquery to identify the heaviest payload mass across the entire dataset. It then lists all booster versions used in missions that carried payloads of the same mass as this maximum value.

2015 Launch Records



```
%sql SELECT strftime('%m', DATE) AS MONTH_NUMBER, \
    CASE strftime('%m', DATE) \
        WHEN '01' THEN 'January' \
        WHEN '02' THEN 'February' \
        WHEN '03' THEN 'March' \
        WHEN '04' THEN 'April' \
        WHEN '05' THEN 'May' \
        WHEN '06' THEN 'June' \
        WHEN '07' THEN 'July' \
        WHEN '08' THEN 'August' \
        WHEN '09' THEN 'September' \
        WHEN '10' THEN 'October' \
        WHEN '11' THEN 'November' \
        WHEN '12' THEN 'December' \
    END AS MONTH_NAME,LANDING_OUTCOME, BOOSTER_VERSION, LAUNCH_SITE, DATE FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Failure (drone ship)' AND DATE LIKE '%2015%'
```

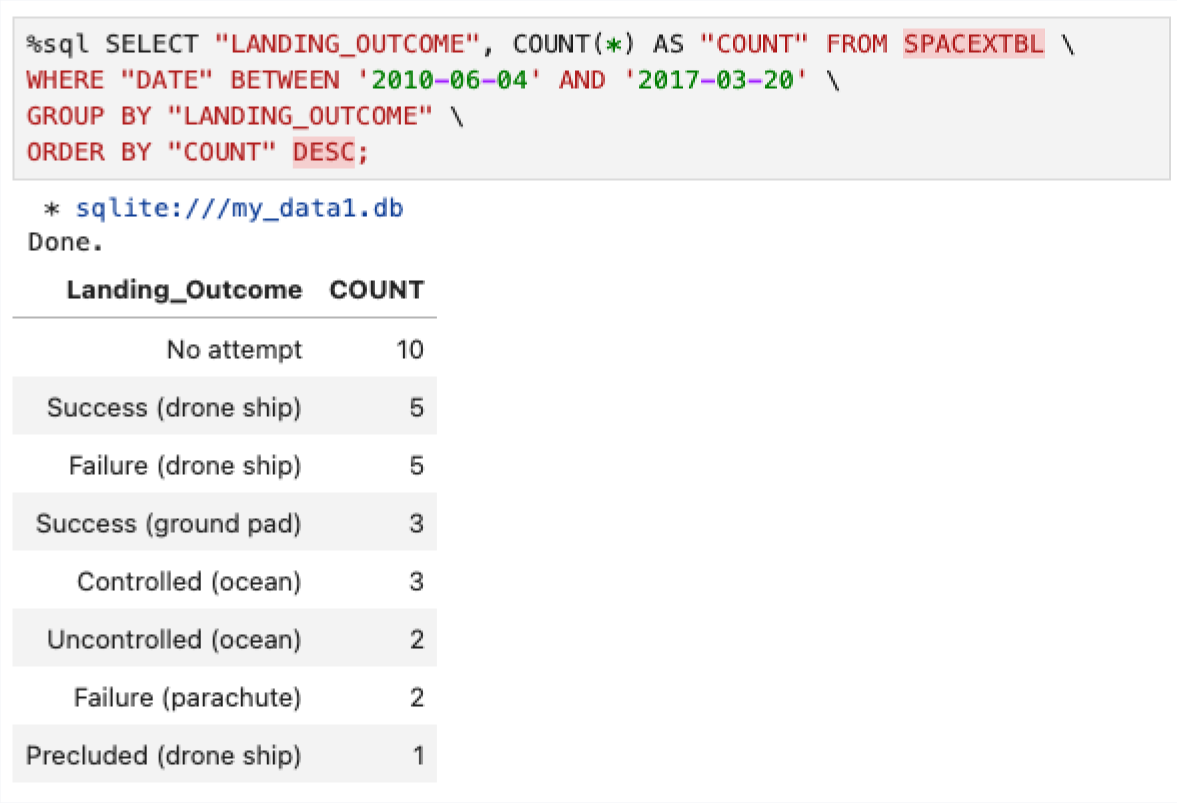
```
 * sqlite:///my_data1.db
Done.
```

MONTH_NUMBER	MONTH_NAME	Landing_Outcome	Booster_Version	Launch_Site	Date
01	January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015-01-10
04	April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015-04-14

- This query filters records from missions that occurred in 2015 where the LANDING_OUTCOME was 'Failure (drone ship)'. It uses a CASE statement to generate the MONTH_NAME from the date and displays it alongside related information such as the month number and booster version.

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

- This query aggregates the number of records for each LANDING_OUTCOME type for missions occurring between June 4, 2010, and March 20, 2017, then displays them sorted in descending order using DESC.



```
%sql SELECT "LANDING_OUTCOME", COUNT(*) AS "COUNT" FROM SPACEXTBL \
WHERE "DATE" BETWEEN '2010-06-04' AND '2017-03-20' \
GROUP BY "LANDING_OUTCOME" \
ORDER BY "COUNT" DESC;
```

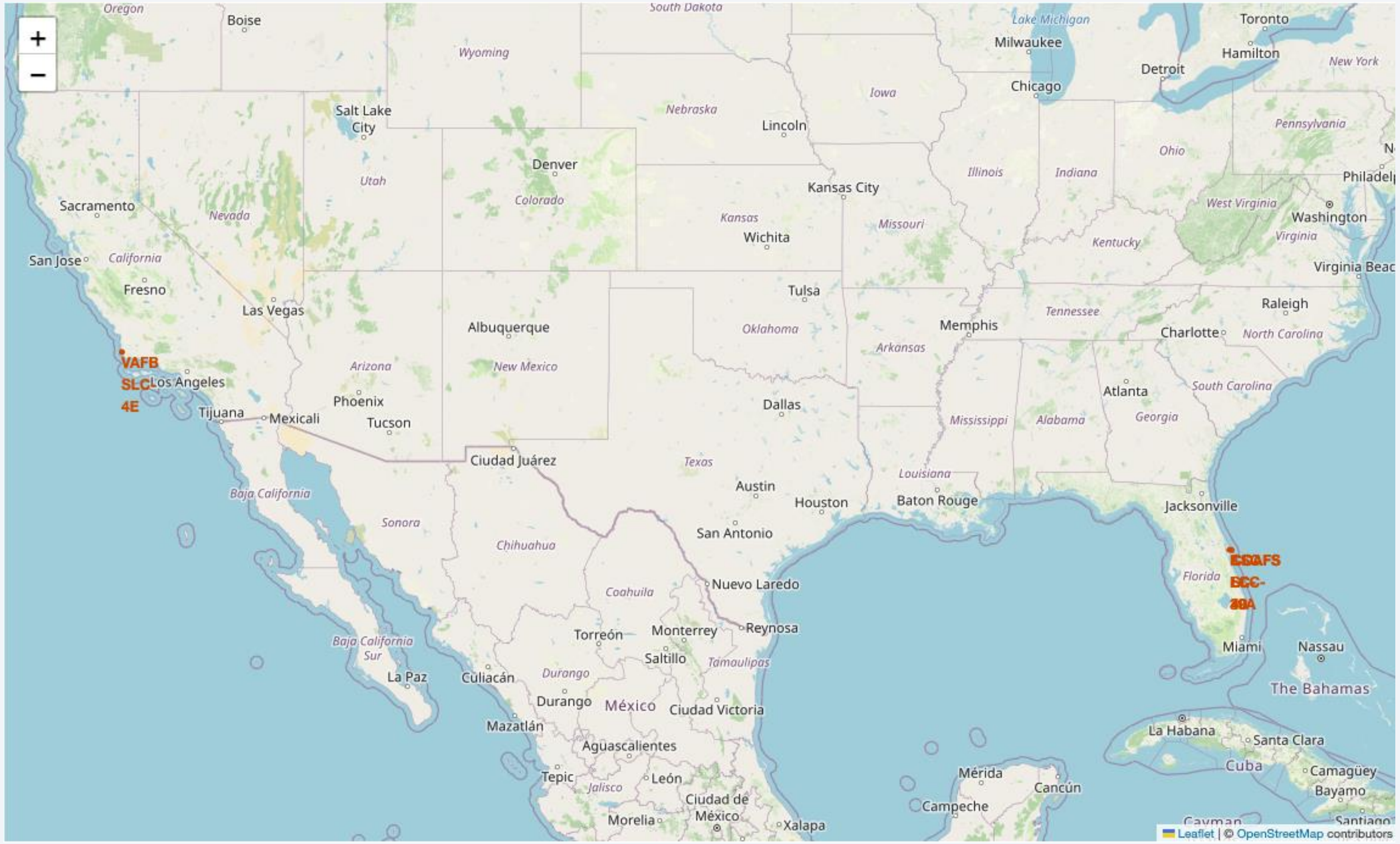
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	COUNT
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

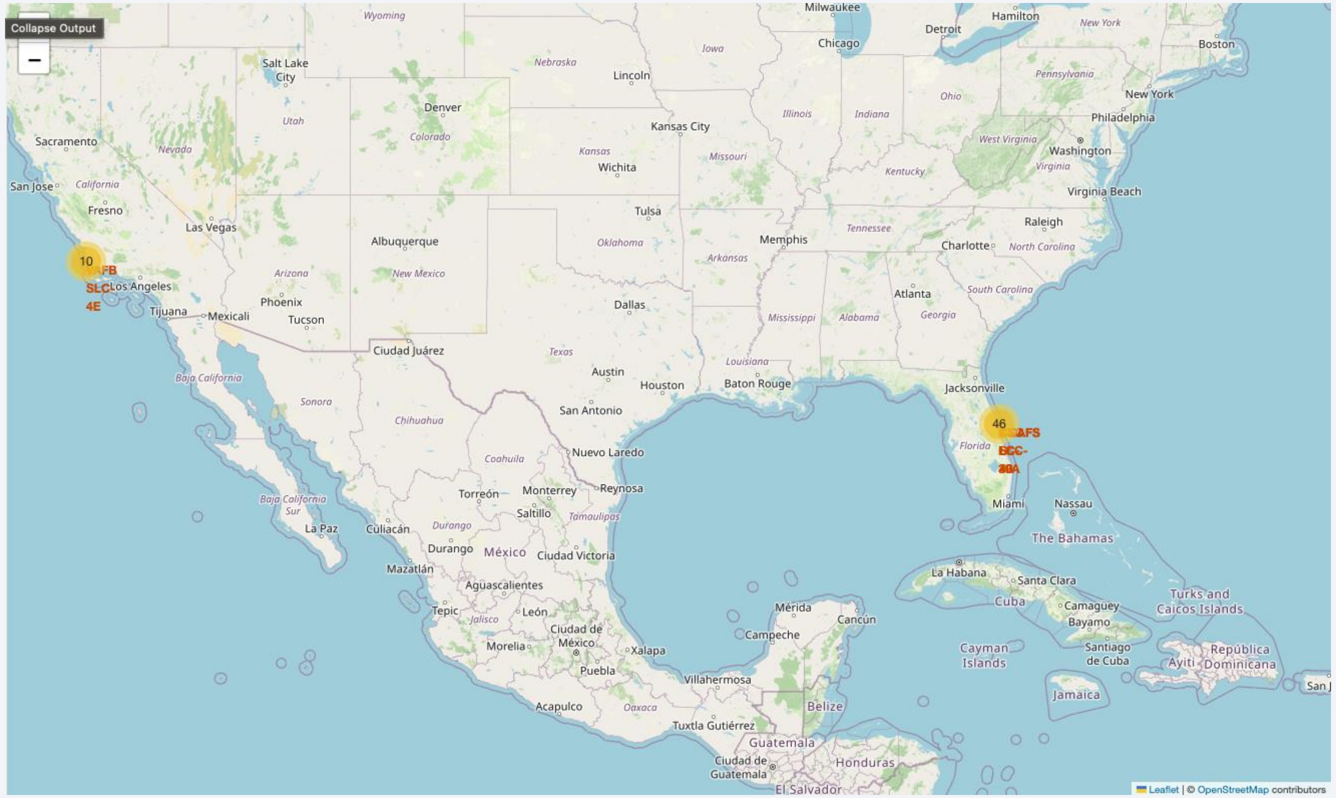
Launch Sites Proximities Analysis

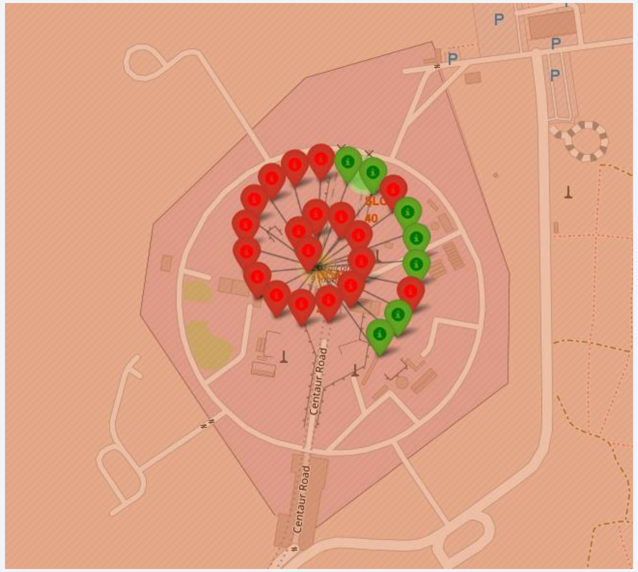
All Launch Sites on a map



- All launch sites are located along the coast.

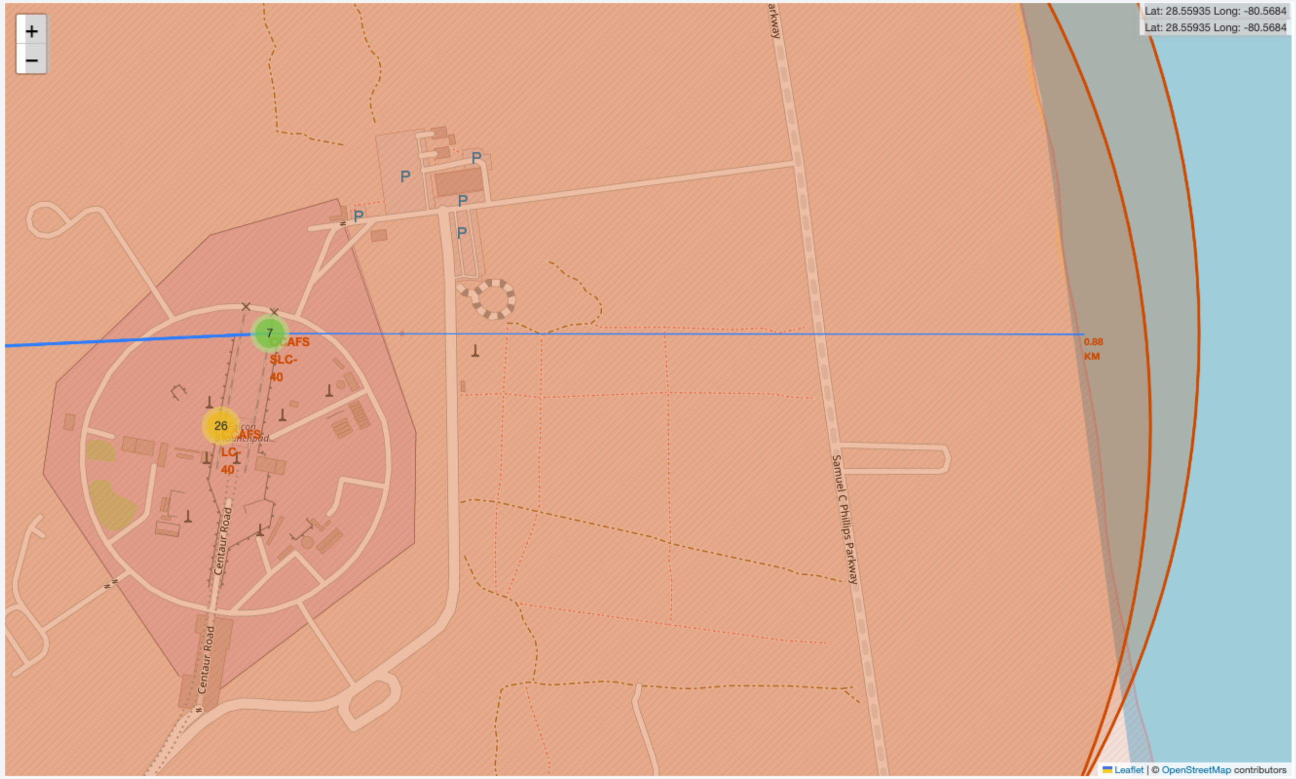
Success/Failed Lauches for Each Site



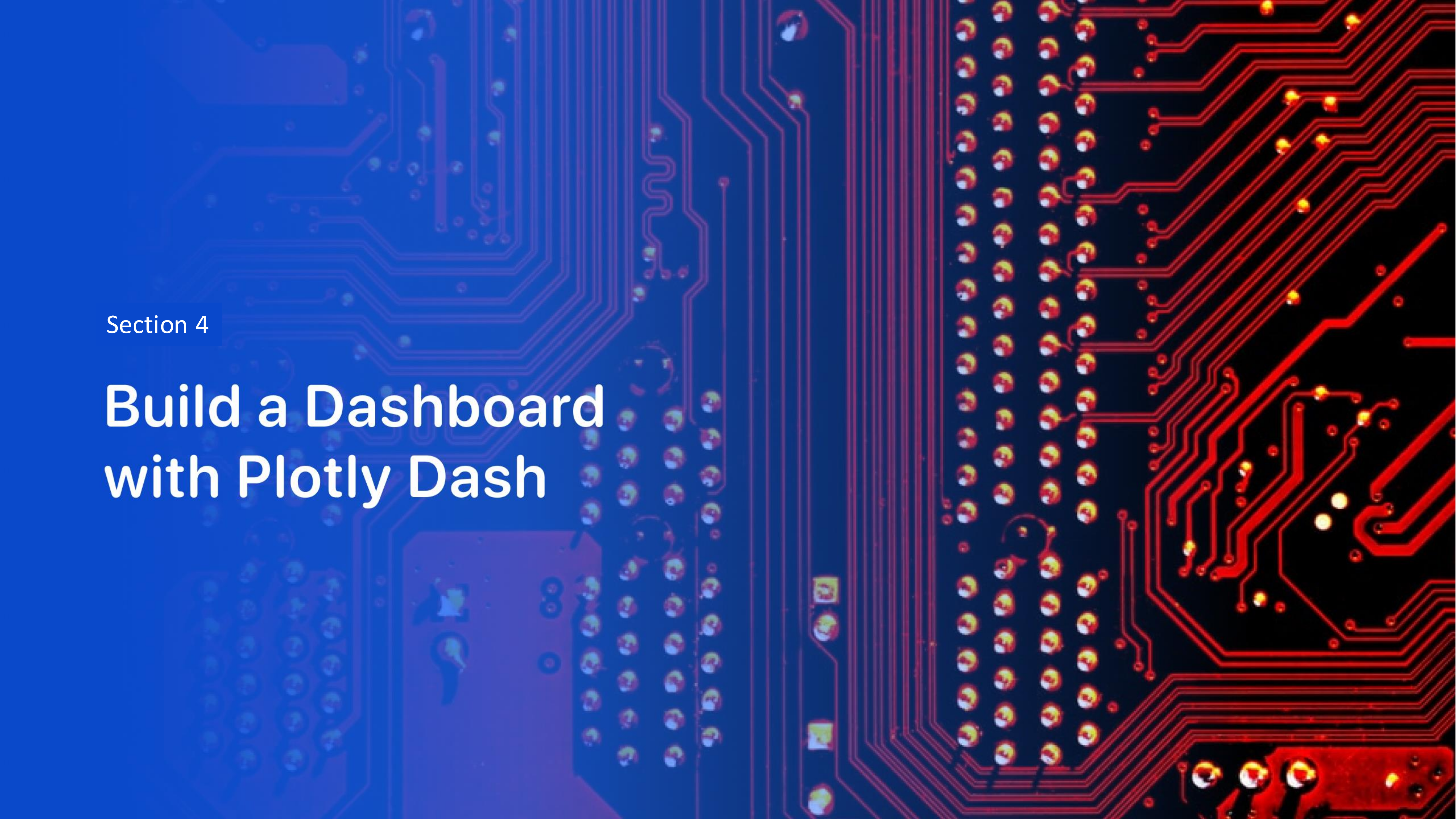


- The map on the left uses MarkerCluster to display multiple launch results grouped together as clusters.
- The map on the left is a detailed enlargement of a specific launch site.

The distances between a launch site to its proximities


+
−
Lat: 28.55935 Long: -80.5684
Lat: 28.55935 Long: -80.5684
P
P
P
P
P
7
CCAFS SLC-40
26
CCAFS LC-40
0.88 KM
Centaur Road
Centaur Road
Samuel C Phillips Parkway
Parkway
Leaflet | © OpenStreetMap contributors

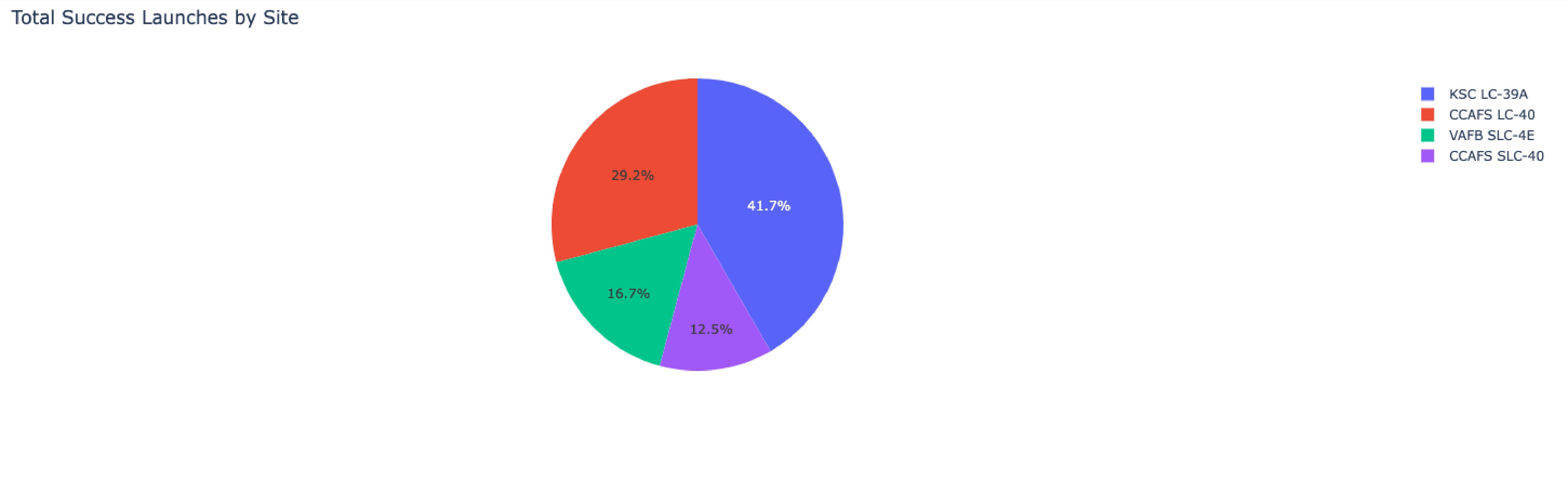
- This map shows the geographical relationship and distance between CCAFS SLC-40 and the nearest coastline.



Section 4

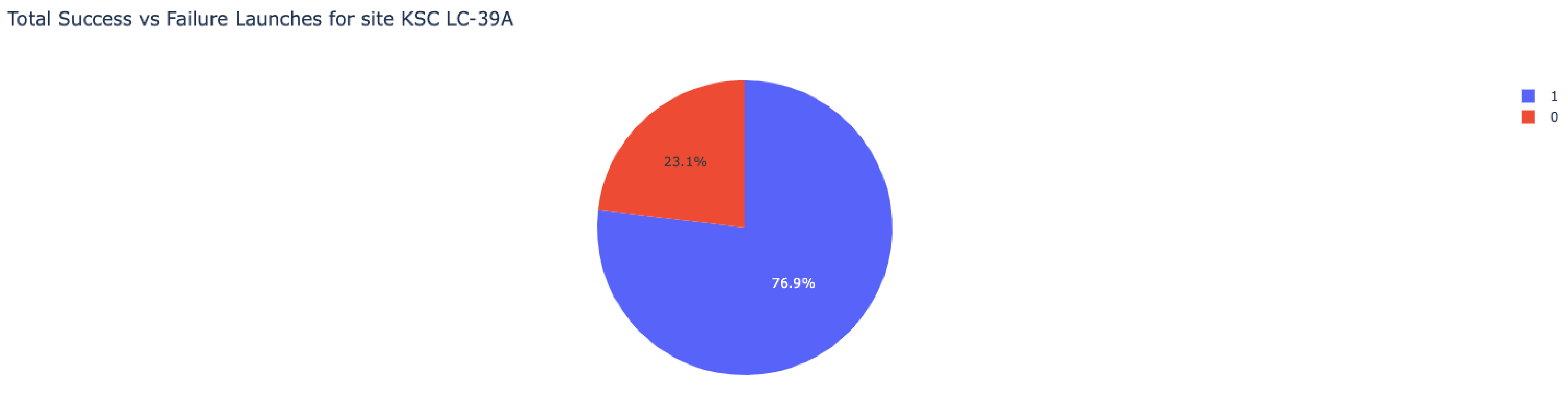
Build a Dashboard with Plotly Dash

Total Success Launches by Site



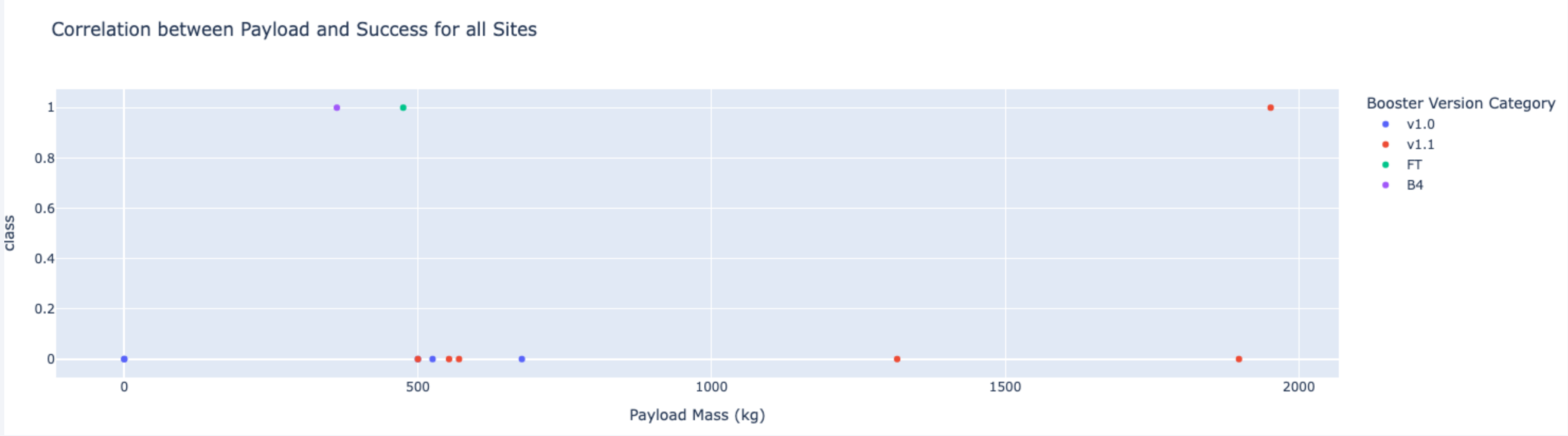
- This pie chart visually represents the percentage of successful launches per launch site across all launch sites.

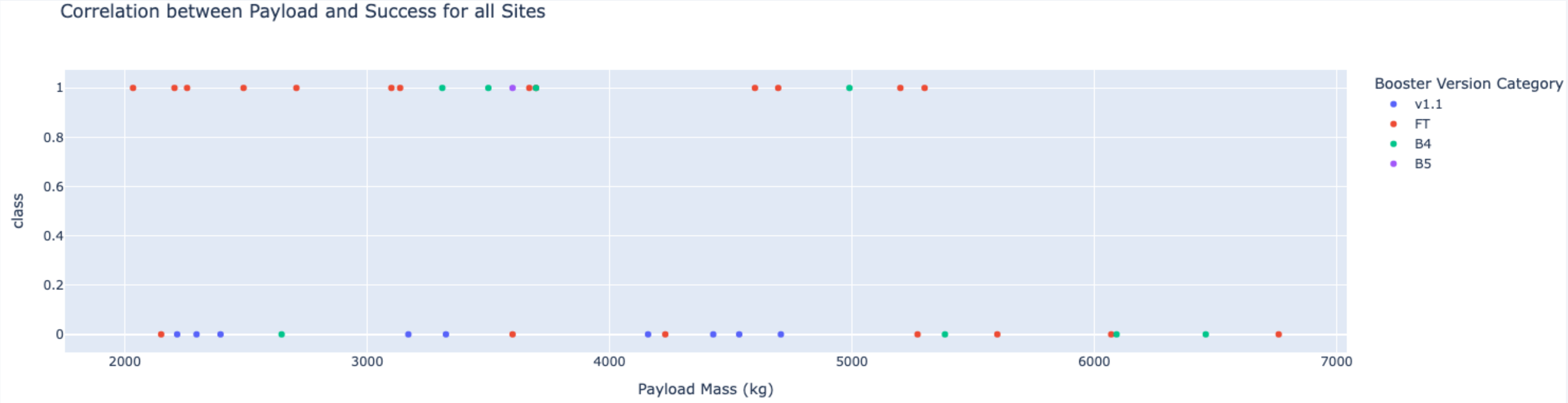
Total Success vs Failure Launches for site KSC LC-39A



- This pie chart visually shows the total success and failure rates for KSC LC-39A.

Correlation between Payload and Success for all Sites



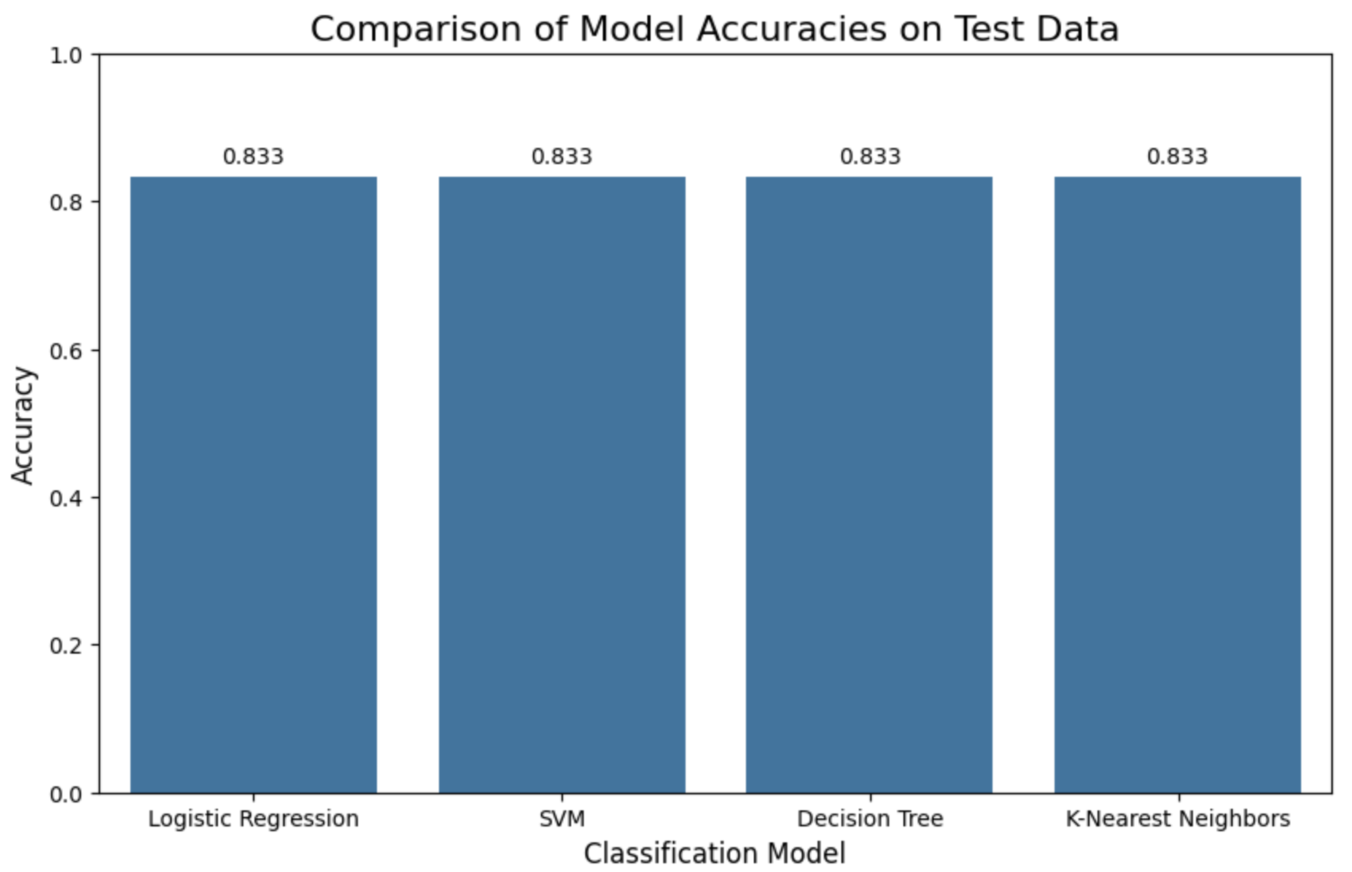


- This scatter plot shows the success and failure of each launch across the 0-2500kg and 2500kg-7000kg ranges for all sites.

Section 5

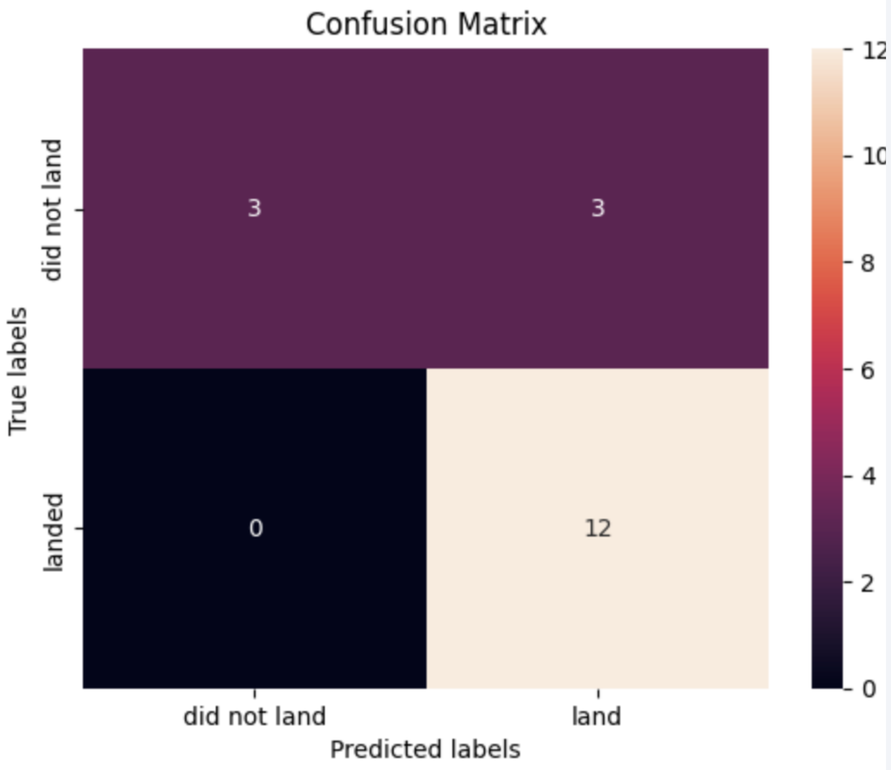
Predictive Analysis (Classification)

Classification Accuracy



- Accuracy is the same across all models.

Confusion Matrix



- The confusion matrix results were similar across all models.

Conclusions

- In this project, we constructed four machine learning classification models to predict the success or failure of SpaceX's booster landings, and rigorously compared and evaluated their performance.

Appendix

- GitHub links for notebooks, datasets, etc:

  https://github.com/ryoosuke/applied-data-science-capstone/tree/main

Thank you!