IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Wambugu Muchemi
Sat 11 May, 2024

Outline

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

Executive Summary

- This project aims to predict the successful landing of the Falcon 9 first stage, a crucial factor in SpaceX's cost-effective rocket launch model. By accurately forecasting landing success, we can estimate launch costs and provide valuable insights for competitive bidding against SpaceX.

- To achieve this, we employed a multi-faceted approach encompassing data collection, wrangling, exploratory data analysis (EDA), and machine learning. Data was gathered through a combination of API calls to SpaceX's website and web scraping techniques, ensuring a comprehensive dataset.

- Data wrangling techniques were applied to clean, transform, and prepare the data for analysis. We then conducted EDA using SQL queries to uncover patterns and relationships within the data. Visualization techniques provided further insights, and Folium was used for geospatial analysis, mapping launch and landing locations.

- Interactive dashboards were created using Plotly, allowing for dynamic exploration of the data and key findings. Finally, we leveraged machine learning algorithms, specifically Support Vector Machines (SVM), Classification Trees, and Logistic Regression, to predict landing success.

- The dataset was split into training and test sets to optimize hyperparameters for each model. Rigorous testing on the test data identified the best-performing model, providing a reliable tool for predicting Falcon 9 first stage landing success.

Introduction

- The commercial space industry has witnessed a dramatic shift in recent years, with SpaceX emerging as a dominant player due to its innovative approach to reusable rockets. Central to this strategy is the Falcon 9, a two-stage rocket designed to recover and reuse its first stage, significantly reducing launch costs. SpaceX advertises Falcon 9 launches at a cost of 62 million dollars, significantly undercutting competitors who charge upwards of 165 million dollars per launch. This cost advantage stems largely from SpaceX's ability to recover and reuse the first stage.

- This project delves into the critical question: Can we accurately predict the successful landing of the Falcon 9 first stage? Answering this question holds significant implications for various stakeholders:

- Competitive Bidding: Understanding the factors influencing landing success allows other companies to estimate SpaceX's launch costs and formulate competitive bids.

- Risk Assessment: Predicting landing outcomes helps assess the risks associated with specific launch parameters and mission profiles.

- Technological Advancement: Analyzing the data can reveal insights into the engineering and operational factors contributing to successful landings, driving further innovation in reusable rocket technology.

- This project aims to develop a reliable model for predicting Falcon 9 first stage landing success, providing valuable information for competitive bidding, risk assessment, and technological advancement within the commercial space industry.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- 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

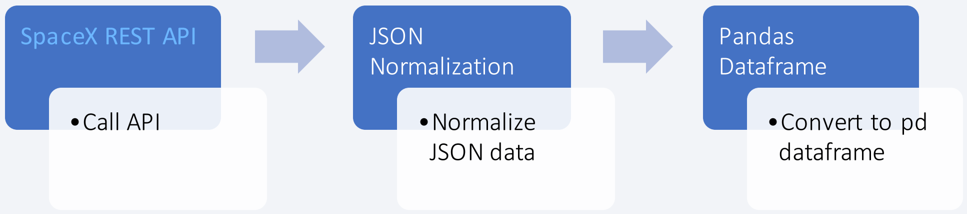
Data Collection

This project utilized a combination of API interactions and web scraping to gather a comprehensive dataset for predicting Falcon 9 first stage landing success. The data collection process involved the following steps:

Data Collection – SpaceX API

- . SpaceX REST API:
- Target Endpoint: The primary data source was the SpaceX REST API, endpoint. This endpoint provided detailed information on past Falcon 9 launches, including rocket specifications, payload details, launch parameters, and landing outcomes.
- GET Requests: We used the Python requests library to perform GET requests to the API endpoint, retrieving the data in JSON format.
- JSON Normalization: The retrieved JSON data, consisting of a list of JSON objects representing individual launches, was converted into a Pandas DataFrame using the json_normalize function. This process flattened the nested JSON structure into a tabular format suitable for analysis.

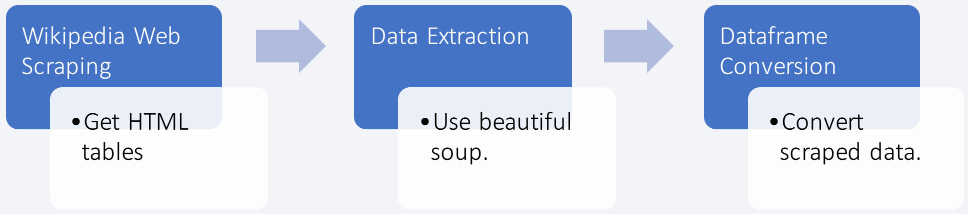
https://github.com/Wambugu-Muchemi/DSprojects/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- 2. Web Scraping:
- Target Websites: Supplementary data was collected by web scraping relevant Wikipedia pages containing Falcon 9 launch records.
- BeautifulSoup Library: The Python BeautifulSoup package was employed to parse the HTML structure of the target web pages and extract data from specific HTML tables.
- DataFrame Conversion: The scraped data was then transformed into a Pandas DataFrame for consistency and ease of analysis.

https://github.com/Wambugu-Muchemi/DSprojects/blob/main/jupyter-labs-webscraping.ipynb


Wikipedia Web Scraping
- Get HTML tables

Data Extraction
- Use beautiful soup.

Dataframe Conversion
- Convert scraped data.

Data Wrangling

3. Data Enrichment:

API Calls for Additional Data: The initial dataset contained identification numbers for various components like boosters, launchpads, payloads, and cores. To enrich the dataset with meaningful information, we performed additional API calls to specific endpoints for each component, retrieving detailed data based on their ID numbers.

Data Filtering: The launch data included records for both Falcon 1 and Falcon 9 boosters. We filtered the dataset to retain only Falcon 9 launches, ensuring relevance to our prediction task.

4. Data Cleaning:

Handling Null Values: The dataset contained null values in certain columns, particularly PayloadMass. To address this, we calculated the mean of the available PayloadMass data and replaced the null values with this mean, ensuring data completeness for analysis.

Preserving Informative Nulls: The LandingPad column contained null values when a landing pad was not used. These nulls were preserved as they provided valuable information for later analysis using one-hot encoding.

https://github.com/Wambugu-Muchemi/DSprojects/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

EDA with Data Visualization

To gain a deeper understanding of the factors influencing Falcon 9 first stage landing success, we employed data visualization techniques to explore relationships within the dataset. We utilized various chart types to highlight key trends and patterns:

1. Categorical Scatter Plot (sns.catplot):

- Variables: FlightNumber vs. PayloadMass, overlaid with landing outcome (success/failure).
- Purpose: To visualize the relationship between flight number, payload mass, and landing success.
- Insights:
    - Increased Success with Flight Number: The plot revealed a positive correlation between flight number and landing success, suggesti ng improvements in technology and operational procedures over time.
    - Payload Mass Impact: A negative correlation was observed between payload mass and landing success, indicating that heavier payl oads pose greater challenges for first stage recovery.

2. Bar Chart:

- Variables: Orbit type and success rate.
- Purpose: To examine the relationship between orbit type and landing success.
- Insights: The bar chart allowed for a direct comparison of success rates across different orbit types, revealing potential variations in landi ng success based on mission profiles.

3. Line Chart:

- Variables: Year and average success rate.
- Purpose: To analyze the trend of Falcon 9 landing success over time.
- Insights: The line chart depicted the overall trend of landing success, highlighting periods of improvement and potential plateaus or setba cks.

EDA with SQL

We utilized SQL queries to further explore the Falcon 9 launch dataset, uncovering patterns and relationships within the data. Here's a summary of the key queries performed:
Task 1: Unique Launch Sites
SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
This query identified all unique launch sites used for Falcon 9 missions.
Task 2: Launch Sites Starting with 'CCA'
SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
This query retrieved five records where the launch site name begins with "CCA."
Task 3: Total Payload Mass for NASA (CRS) Launches
SELECT SUM("Payload_Mass__kg_") FROM SPACEXTABLE WHERE "Customer" = 'NASA (CRS)';
This query calculated the total payload mass carried by Falcon 9 rockets launched for NASA's Commercial Resupply Services (CRS) missions.
Task 4: Average Payload Mass for F9 v1.1 Booster
SELECT AVG("Payload_Mass__kg_") FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
This query determined the average payload mass carried by the Falcon 9 v1.1 booster version.
Task 5: First Successful Ground Pad Landing Date
SELECT MIN(Date) FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';
This query identified the date of the first successful Falcon 9 landing on a ground pad.
Task 6: Boosters with Drone Ship Success and Specific Payload Mass Range
SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "Payload_Mass__kg_" BETWEEN 4000 AND 6000;
This query listed the booster versions that successfully landed on drone ships and carried payloads within a specific mass range (4000-6000 kg).
Task 7: Total Successful and Failed Missions
SELECT "Landing_Outcome", COUNT(*) FROM SPACEXTABLE GROUP BY "Landing_Outcome";
This query counted the total number of successful and failed Falcon 9 missions based on their landing outcomes.
Task 8: Booster Versions with Maximum Payload Mass
SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Payload_Mass__kg_" = (SELECT MAX("Payload_Mass__kg_") FROM SPACEXTABLE);
This query used a subquery to identify the booster versions that carried the maximum payload mass.
Task 9: Launch Details for Failed Drone Ship Landings in 2015
SELECT SUBSTR(Date, 6, 2) AS Month, "Booster_Version", "Launch_Site", "Landing_Outcome" FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE 'Failure (drone ship)' AND SUBSTR(Date, 0, 5) = '2015';
This query extracted launch details (month, booster version, launch site, landing outcome) for missions with failed drone ship landings in 2015.
Task 10: Ranked Landing Outcomes Between Specific Dates
SELECT "Landing_Outcome", COUNT(*) AS Count FROM SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY Count DESC;
This query ranked the count of different landing outcomes between specific dates in descending order.

https://github.com/Wambugu-Muchemi/DSprojects/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

We created an interactive map using the Folium library to visualize the geographical distribution of Falcon 9 launch sites. The map included the following objects:

Circles:

Purpose: To highlight the location of NASA Johnson Space Center (JSC).

Appearance: A blue circle with a radius of 1000 meters, centered on JSC's coordinates.

Interactivity: A popup label displaying "NASA Johnson Space Center" appears when clicking on the circle.

Markers:

Purpose: To provide a visual representation of NASA JSC on the map.

Appearance: A marker with a custom icon displaying "NASA JSC" in bold text and a distinct color.

Placement: Positioned at the same coordinates as the circle, providing a clear visual reference.

Rationale for Object Selection:

Circles: Effectively highlight areas of interest on the map, providing a visual cue for their location and size.

Markers: Offer a precise way to pinpoint specific locations and display associated information through custom icons and labels.

https://github.com/Wambugu-Muchemi/DSprojects/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

We built an interactive dashboard using Plotly and Dash to provide a dynamic and user-friendly way to explore the Falcon 9 launch dataset. The dashboard features two main visualizations and interactive components:

1. Pie Chart:

Purpose: To visualize the proportion of successful launches for all sites or a specific launch site.

Data:

All Sites: Displays the total count of successful and failed launches across all sites.

Specific Site: Shows the success vs. failure counts for the selected launch site.

Interaction: A dropdown menu allows users to select a specific launch site or view data for all sites. The pie chart dynamically updates based on the selected site.

Build a Dashboard with Plotly Dash

2. Scatter Chart:

Purpose: To explore the correlation between payload mass and launch success, considering different booster versions.

Data: Plots payload mass on the x-axis and launch outcome (success/failure) on the y-axis. Each point represents a launch, colored by booster version category.

Interactions:

Launch Site Selection: The dropdown menu used for the pie chart also filters the scatter chart data by launch site.

Payload Range Slider: A range slider allows users to select a specific payload mass range, dynamically updating the scatter chart to display launches within that range.

Rationale for Plot and Interaction Choices:

Pie Chart: Effectively displays proportions and comparisons between categories (success/failure).

Scatter Chart: Ideal for visualizing correlations between two numerical variables (payload mass and launch outcome) and incorporating a categorical variable (booster version).

Dropdown Menu: Provides a user-friendly way to filter data by launch site, allowing for focused analysis.

Range Slider: Enables dynamic exploration of the data by selecting specific payload mass ranges, revealing potential trends within those ranges.

https://github.com/Wambugu-Muchemi/DSprojects/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

This project aimed to predict Falcon 9 first stage landing success using a robust classification model. We followed a structured process, starting with data preparation and culminating in the selection of the best-performing algorithm:

1. Data Loading and Preprocessing:

Data Acquisition: We loaded the Falcon 9 launch dataset from a specified URL.

Target Variable Extraction: We extracted the "Class" column, representing landing success (1) or failure (0), and converted it into a Pandas Series, assigning it to the variable Y.

Data Standardization: We standardized the numerical features in the dataset (X) using a provided transformation, ensuring consistent scaling for optimal model performance.

2. Data Splitting:

Training and Test Sets: We split the standardized data (X and Y) into training and test sets using the train_test_split function. We allocated 80% of the data for training and 20% for testing, setting random_state to 2 for reproducibility.

Predictive Analysis (Classification)

3. Model Building and Evaluation:

Model Selection: We evaluated four classification algorithms:

Logistic Regression: A linear model suitable for binary classification.

Support Vector Machines (SVM): A powerful algorithm capable of handling complex data relationships.

Decision Tree: A tree-based model that partitions data based on feature values.

K-Nearest Neighbors (KNN): A non-parametric method that classifies data points based on their proximity to labeled examples.

Hyperparameter Tuning: For each algorithm, we performed hyperparameter tuning using GridSearchCV with 10-fold cross-validation (cv=10). This involved systematically searching for the best combination of hyperparameters to maximize model performance on the training data.

Model Evaluation: We assessed each model's performance on the test data using the score method, which calculates accuracy. We also visualized the confusion matrix for each model to understand its classification performance in detail.

https://github.com/Wambugu-Muchemi/DSprojects/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

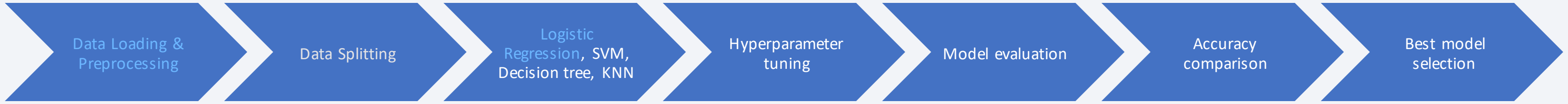
Predictive Analysis (Classification)

4. Best Model Selection:

Accuracy Comparison: We compared the accuracy scores of all four models on the test data.

Best Model Identification: We identified Logistic Regression as the best-performing model based on its highest accuracy score.

Flowchart Representation:



Data Loading & Preprocessing

Data Splitting

Logistic Regression, SVM, Decision tree, KNN

Hyperparameter tuning

Model evaluation

Accuracy comparison

Best model selection

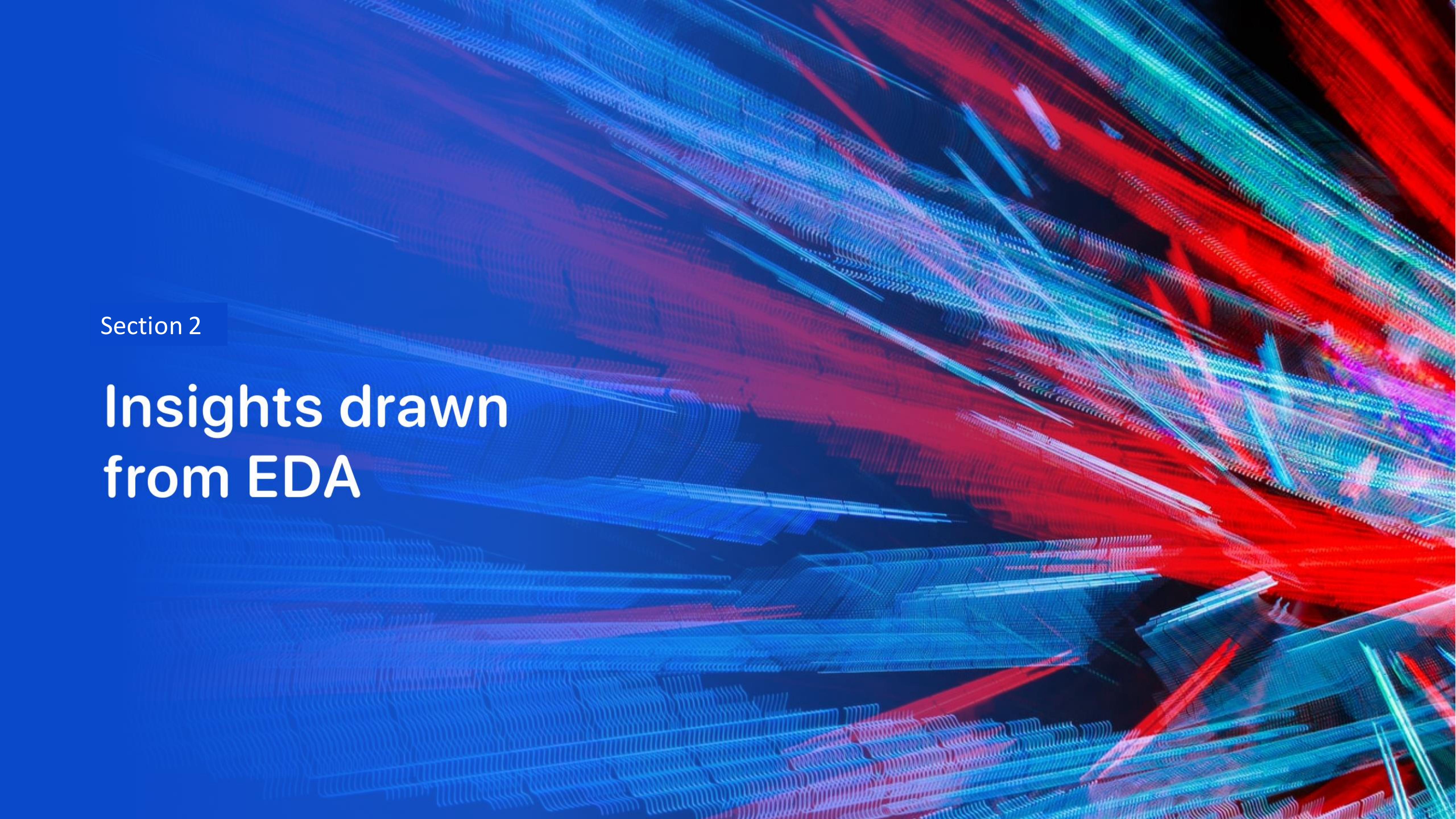
https://github.com/Wambugu-Muchemi/DSprojects/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

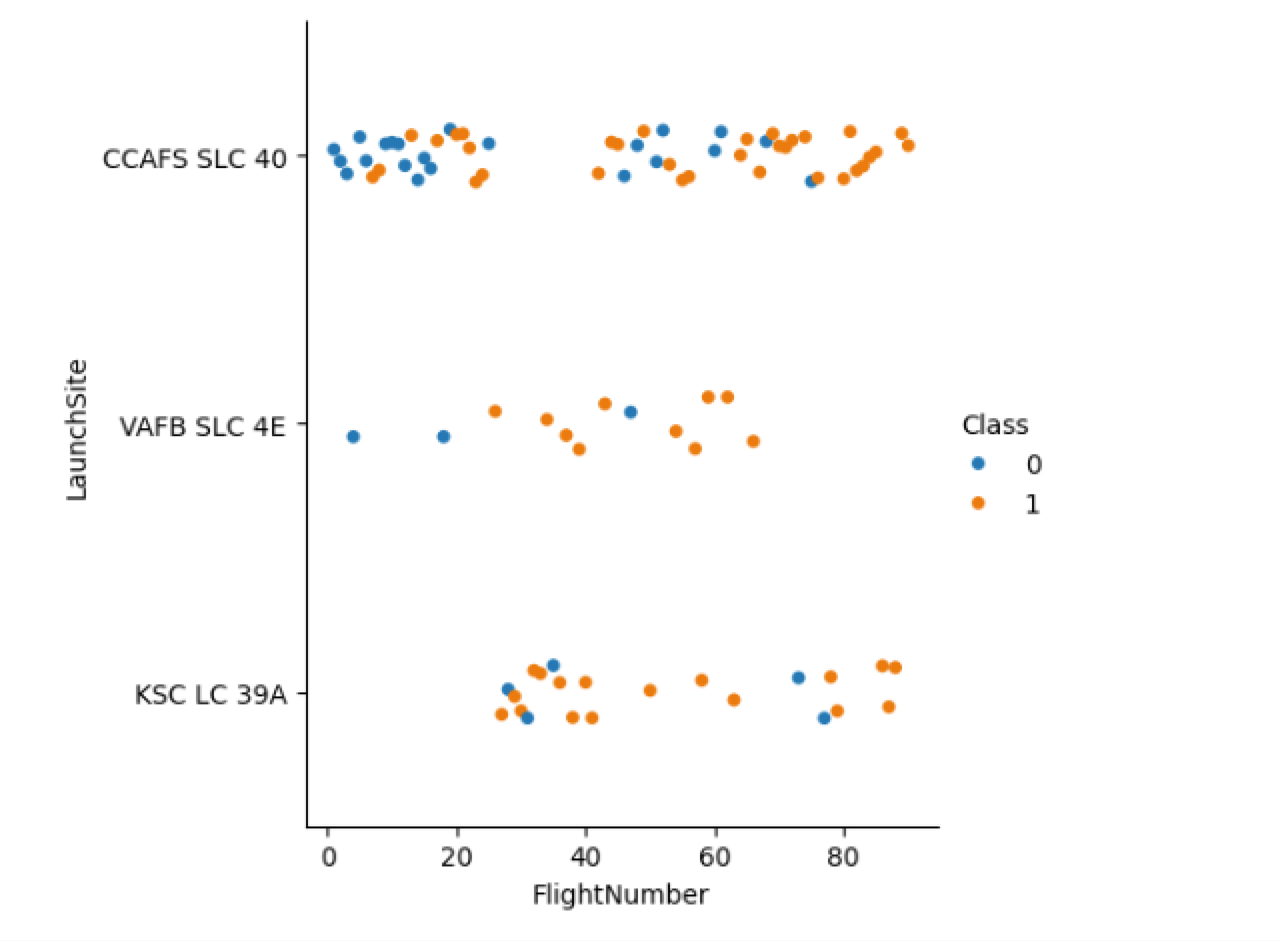
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

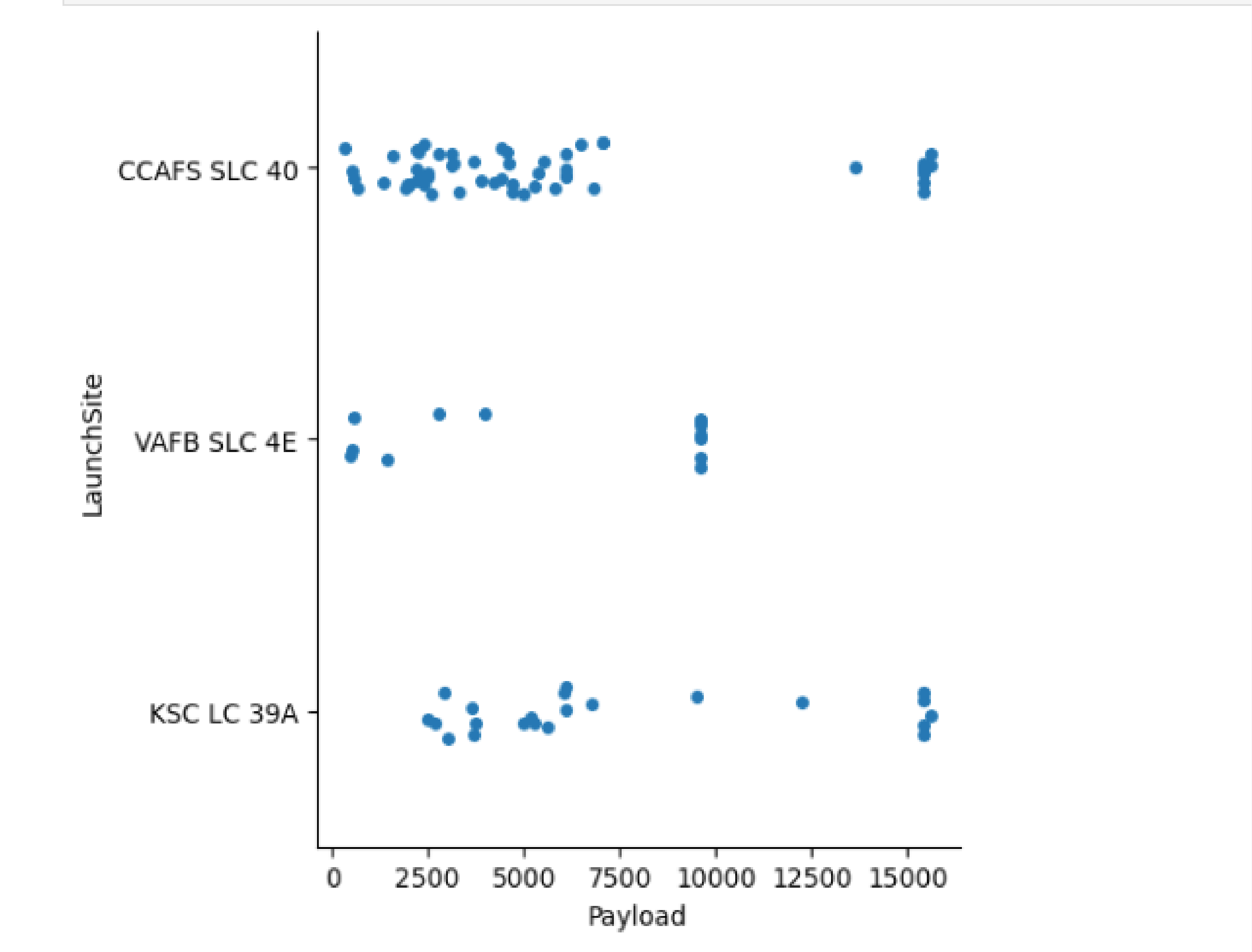
Insights drawn from EDA



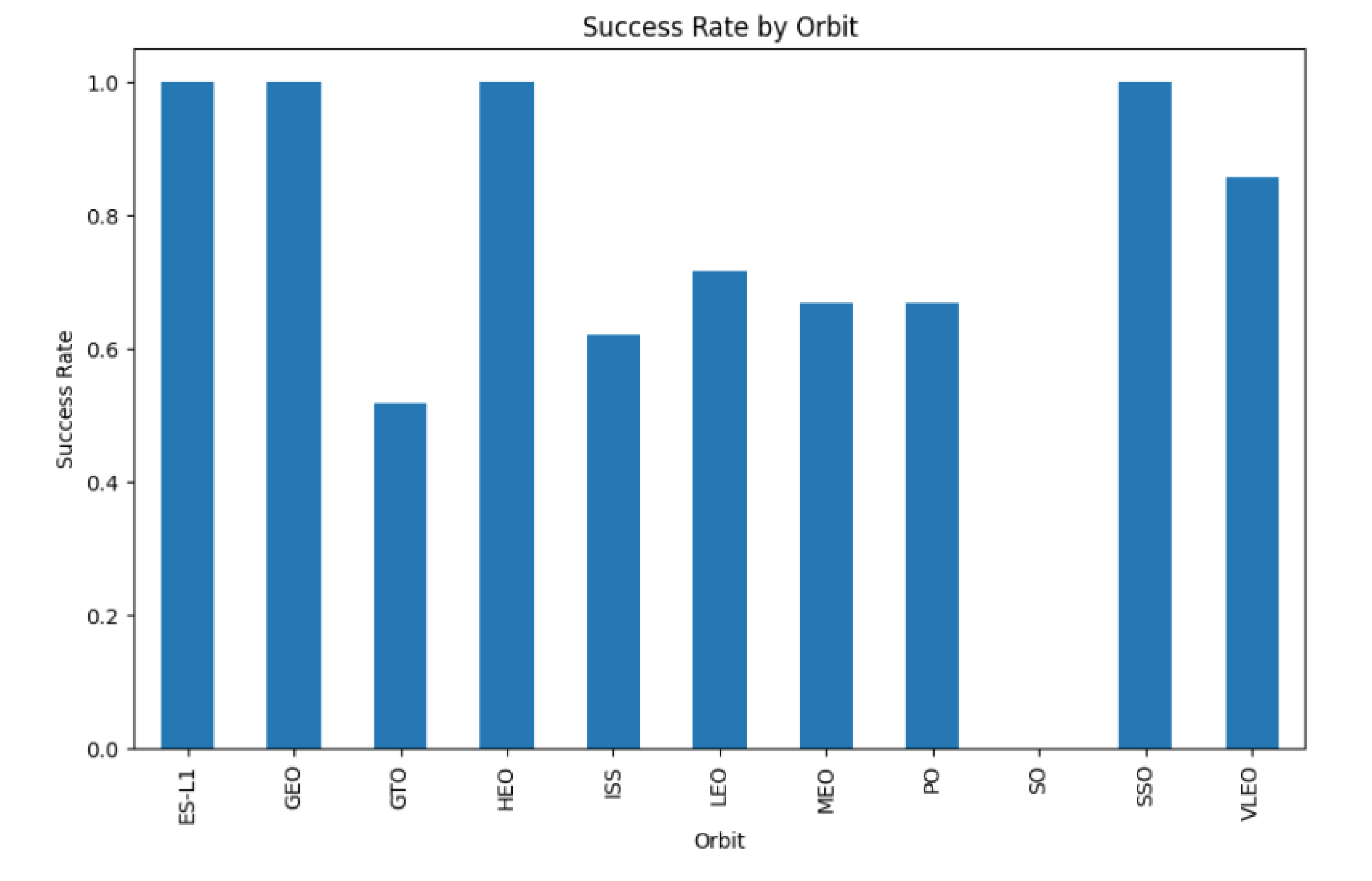
Flight Number vs. Launch Site



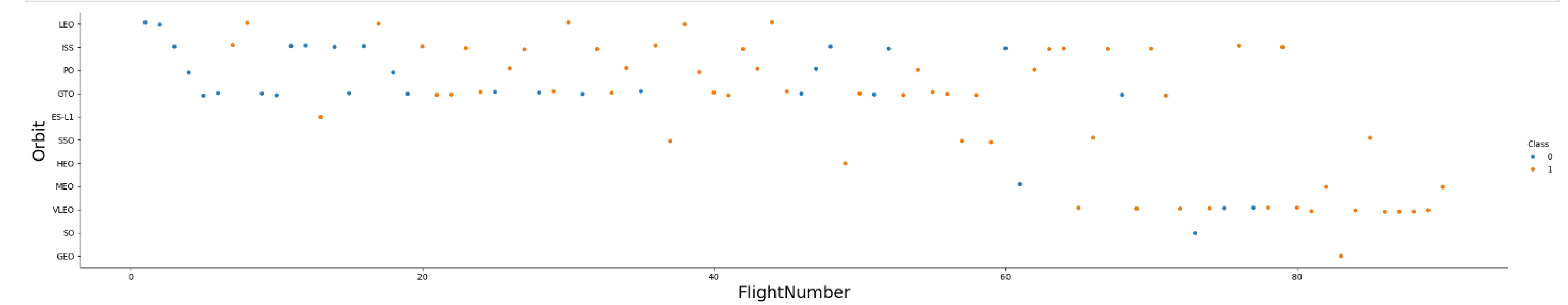
Payload vs. Launch Site



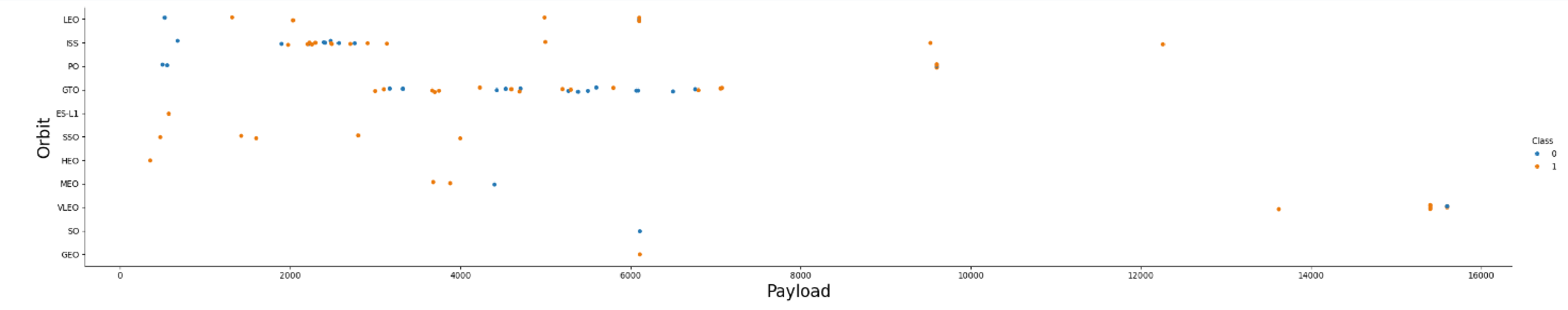
Success Rate vs. Orbit Type



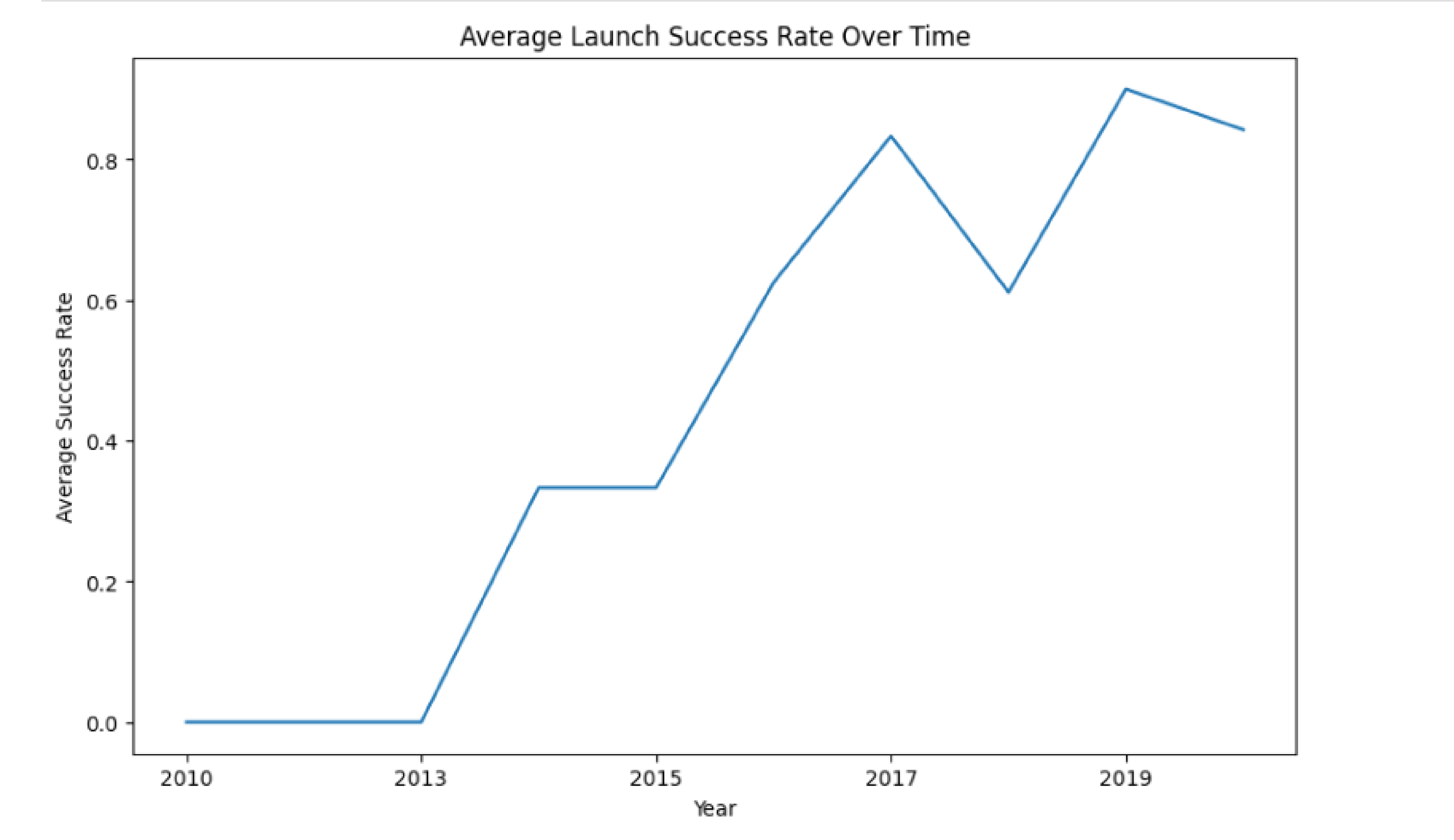
Flight Number vs. Orbit Type



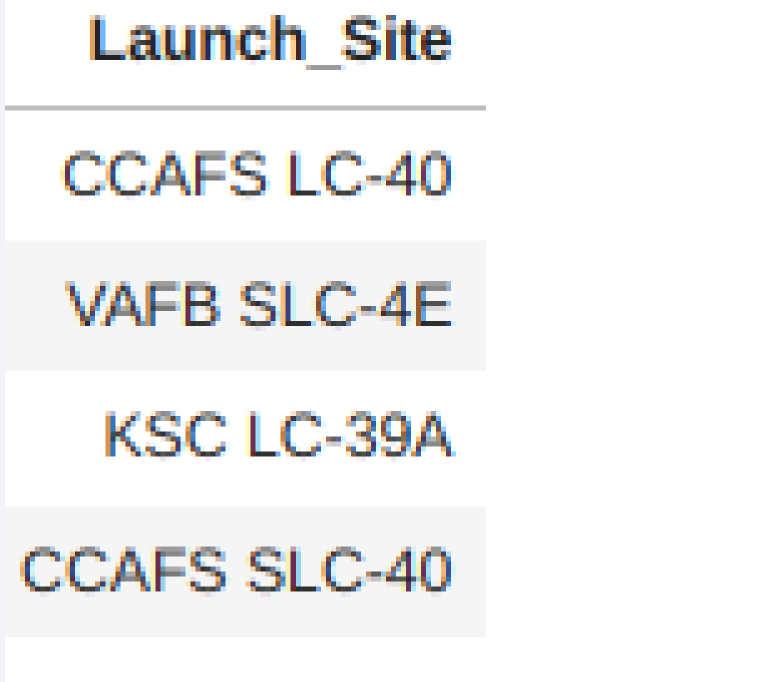
Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

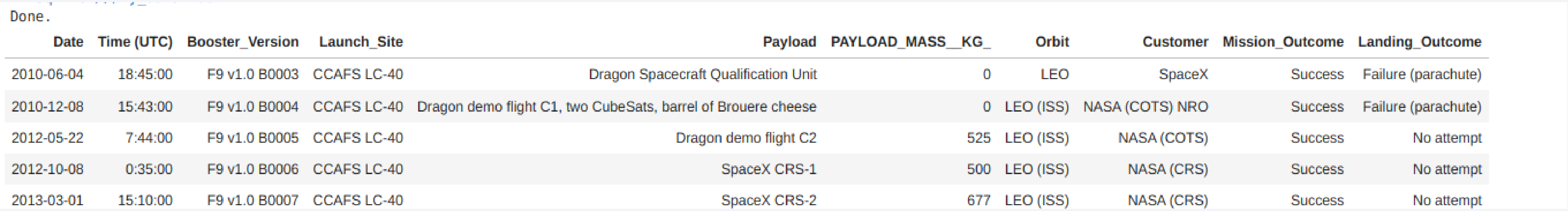


Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

This query identified all unique launch sites used for Falcon 9 missions.

Launch Site Names Begin with 'CCA'

This query retrieved five records where the launch site name be gins with "CCA."

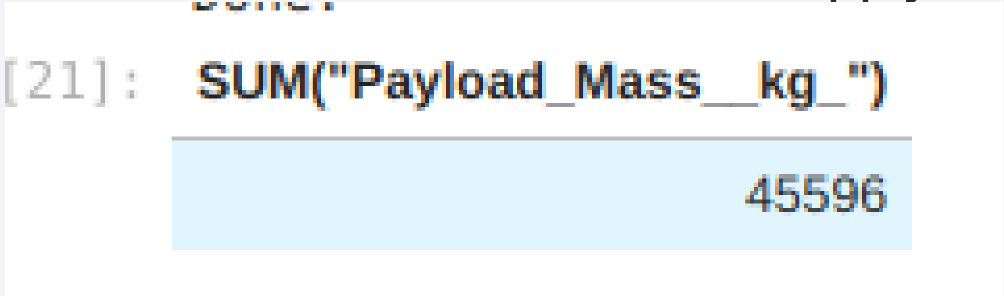


```
Done.
```

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

Total Payload Mass

This query calculated the total payload mass carried by Falcon 9 rockets launched for NASA's Commercial Resupply Services (CRS) missions.

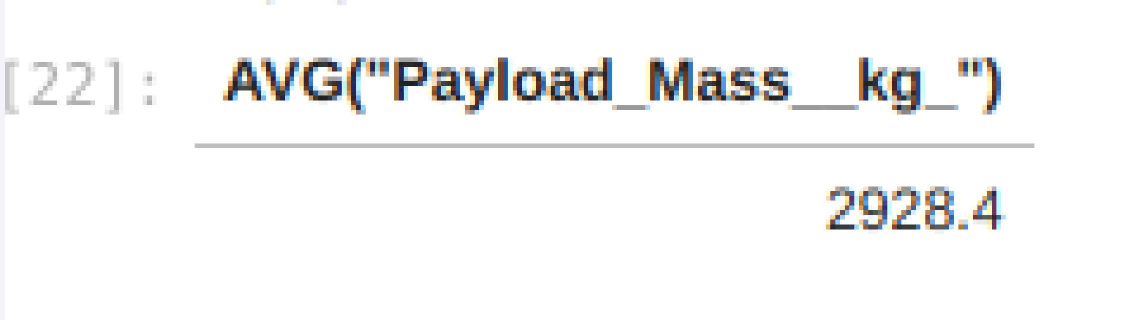


[21]:

SUM("Payload_Mass__kg_")
45596

Average Payload Mass by F9 v1.1

This SQL query will select the average of the values in the "Payload_Mass_kg_" column from the "SPACEXTABLE" table where the "Booster_Version" column is equal to "F9 v1.1".



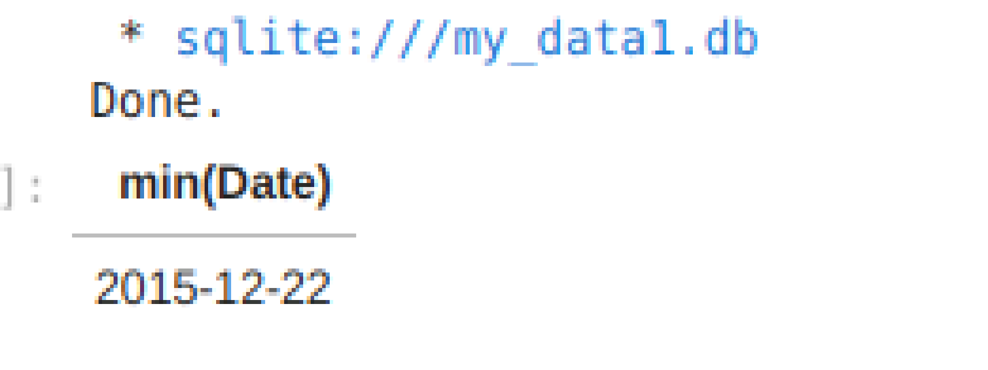
[22]:

AVG("Payload_Mass__kg_")
2928.4

First Successful Ground Landing Date

- This query identified the date of the first successful Falcon 9 landing on a ground pad.

- This query identified the date of the first successful Falcon 9 landing on a ground pad.

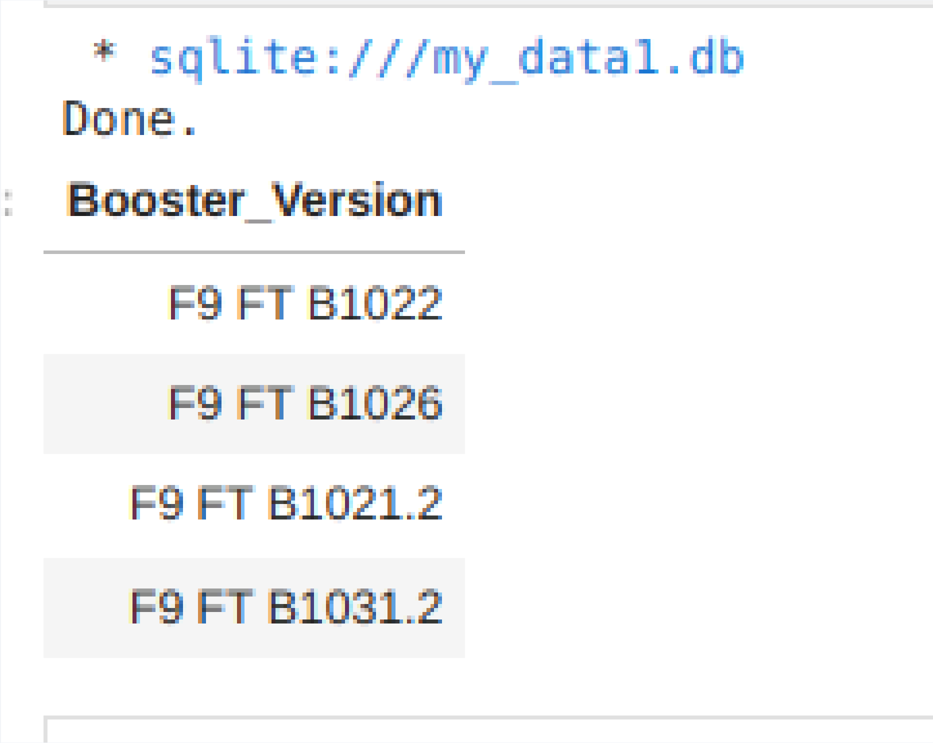


```
 * sqlite:///my_data1.db
Done.
```

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

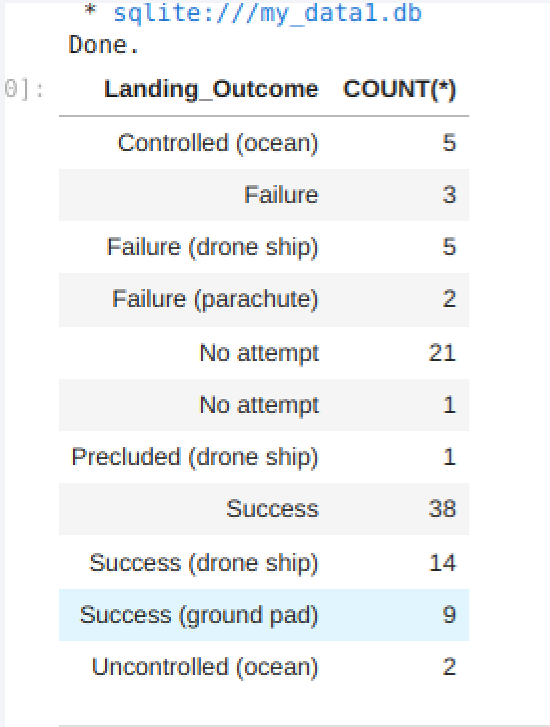


```
 * sqlite:///my_data1.db
Done.
```

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 counted the total number of successful and failed Falcon 9 missions based on their landing outcomes.

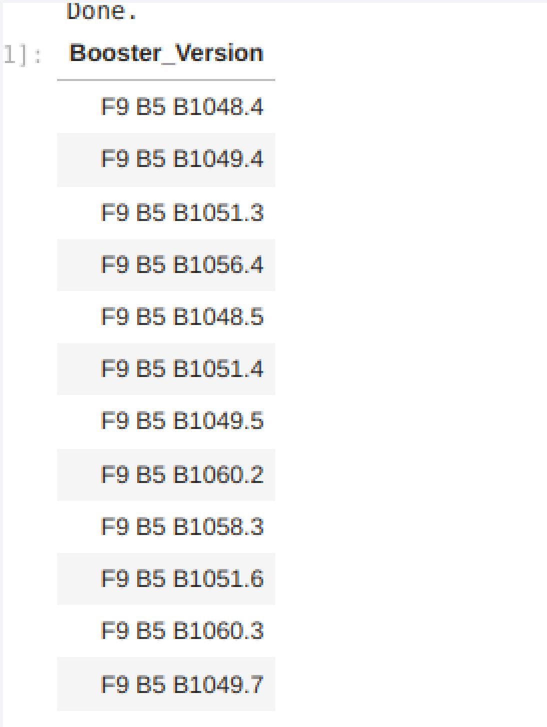


```
 * sqlite:///my_data1.db
Done.
0]:
```

Landing_Outcome	COUNT(*)
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

This query used a subquery to identify the booster versions that carried the m aximum payload mass.



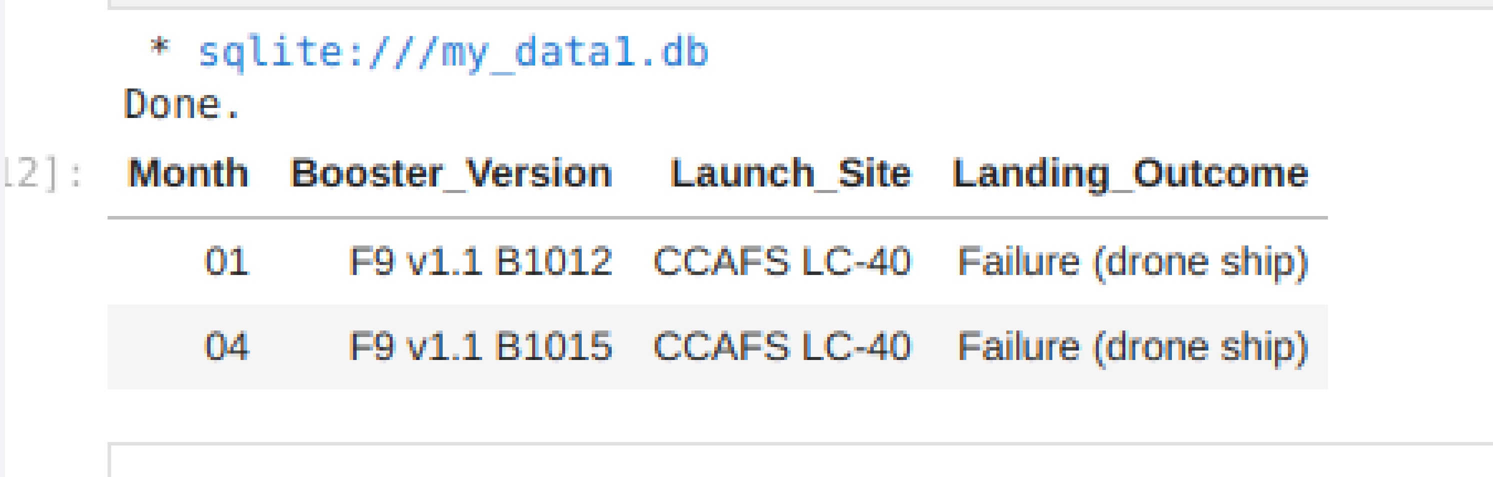
```
Done.
```

1]:

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

This query extracted launch details (month, booster version, launch site, landing outcome) for missions with failed drone ship landings in 2015.

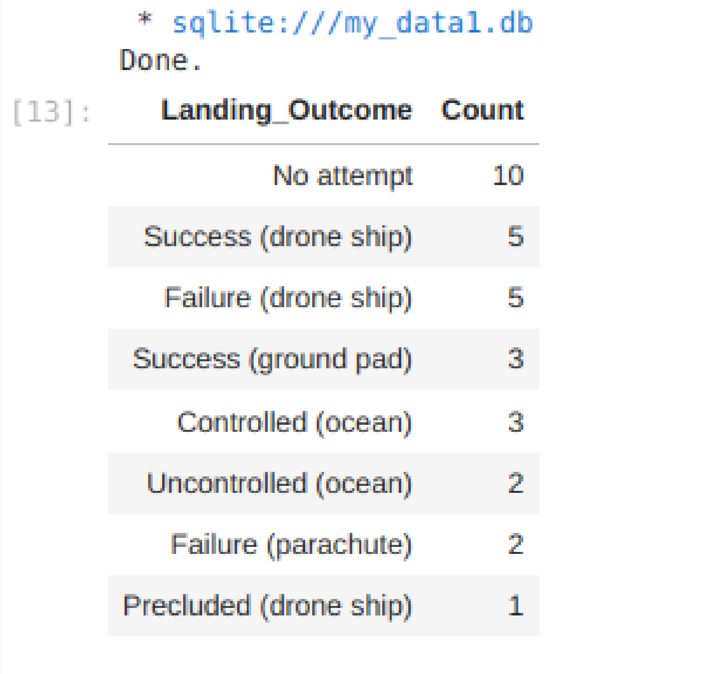


```
 * sqlite:///my_data1.db
Done.
```

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

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

This query ranked the count of different landing outcomes between specific dates in descending order.



```
 * sqlite:///my_data1.db
Done.
[13]:
```

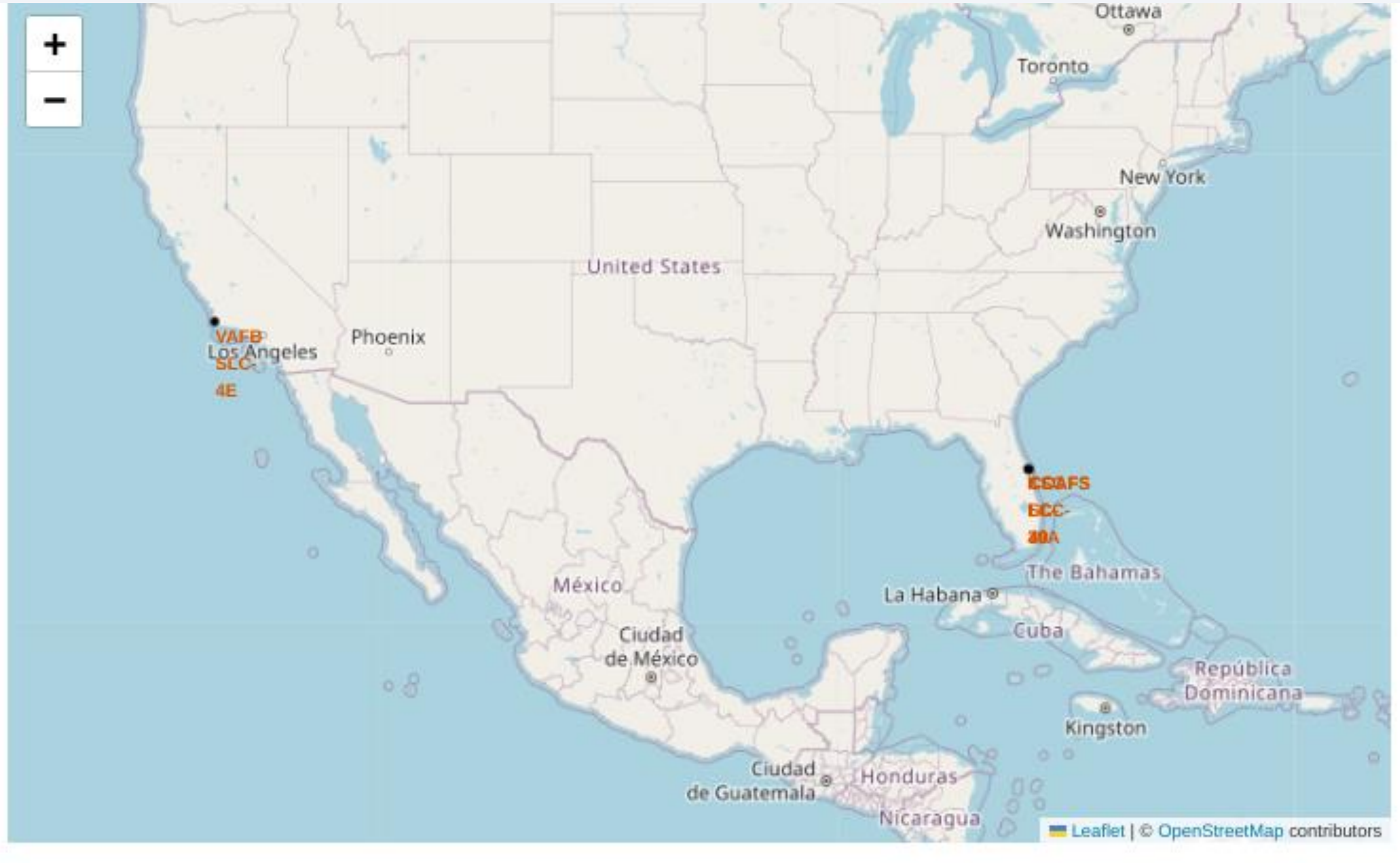
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

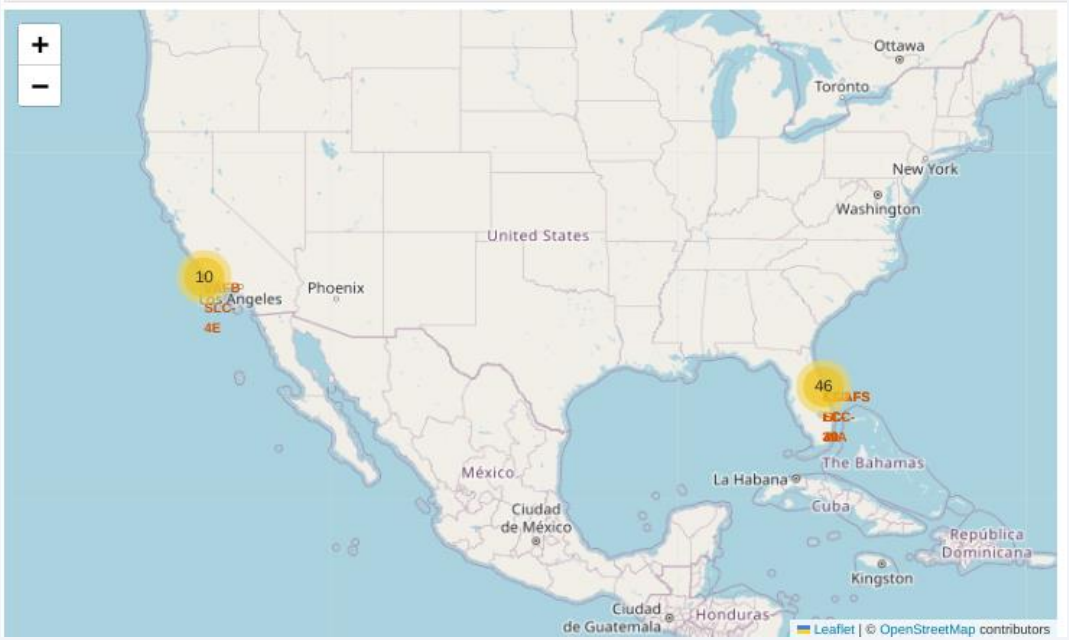
All Launch Sites on Folium Map

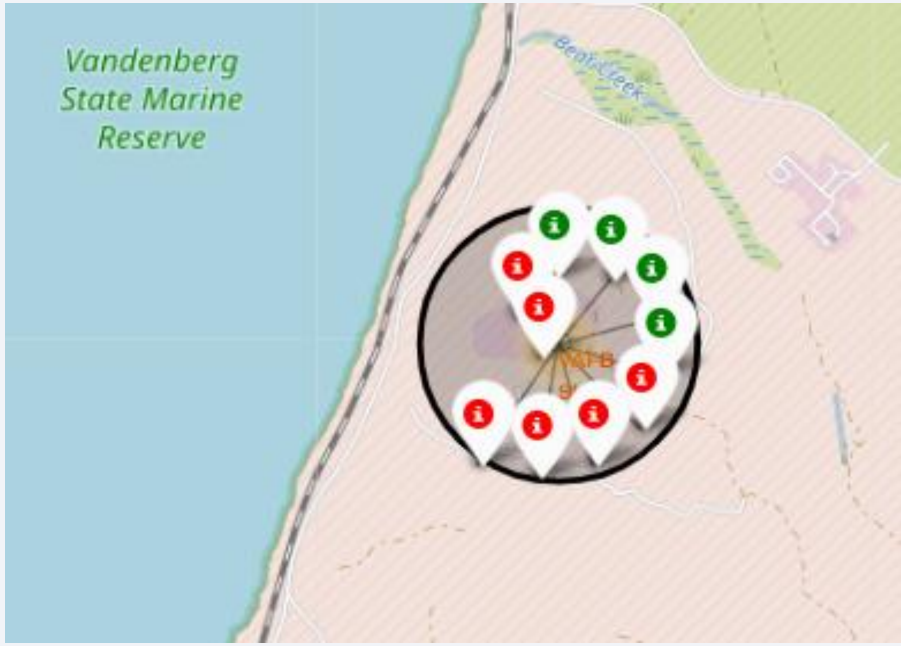
All launch sites on a map repres3ented by folium.Circle and folium.Marker



Classified Launch Sites Using Colored Markers.

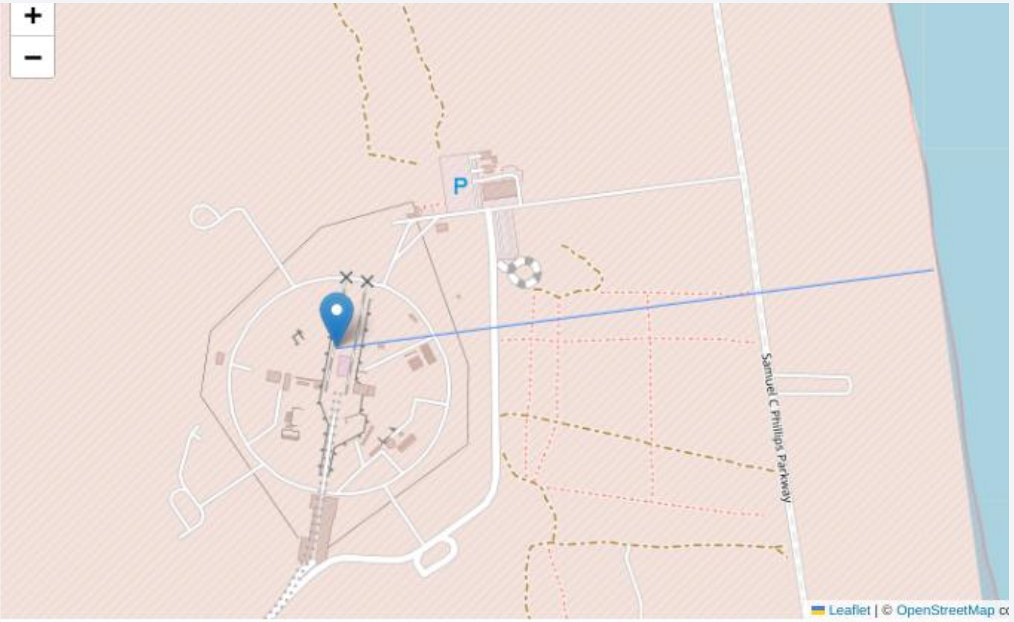
Folium map showing the color labelled version of the map locations of the launch sites. If a launch was successful (class=1), then we use a green marker and if a launch was failed, we use a red marker (class=0)

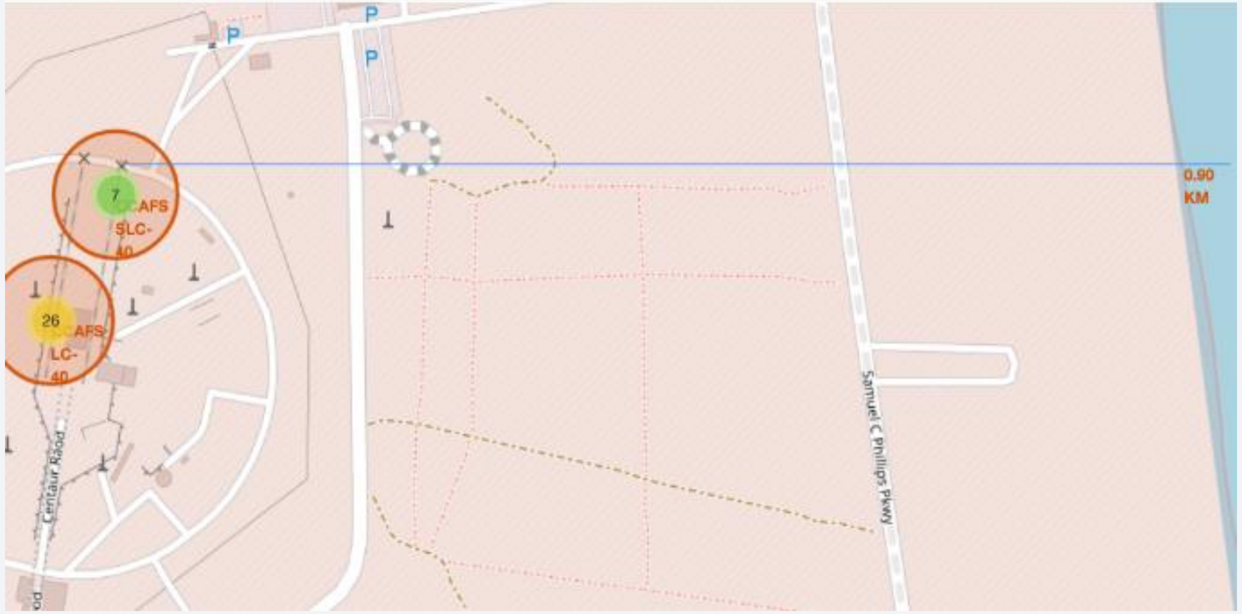




Calculating proximities from Launch Site to Coastline.

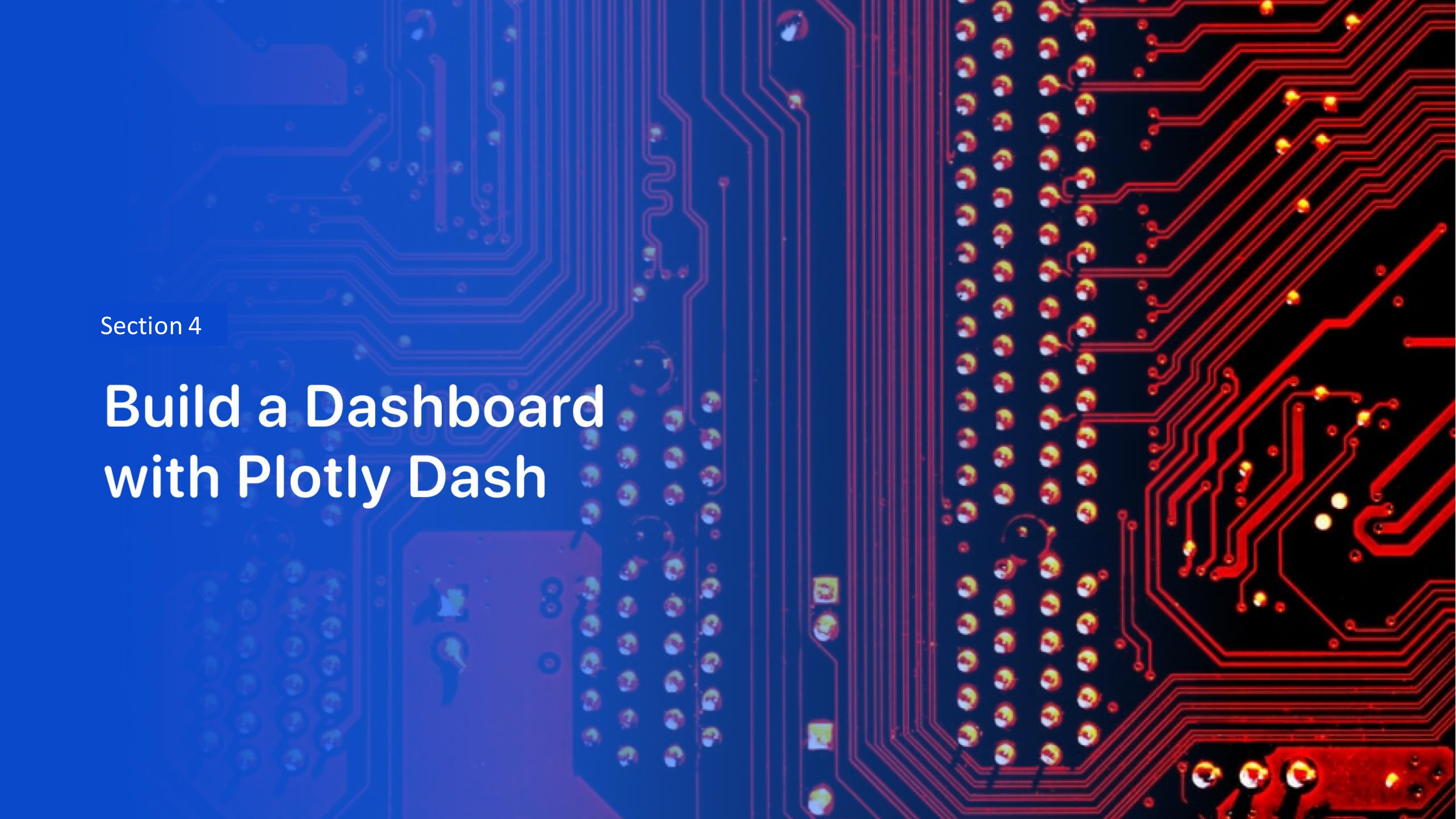
Launch site to its proximities in this case to the coastline. This will help identify shortest distances to resources such as to railway etc.





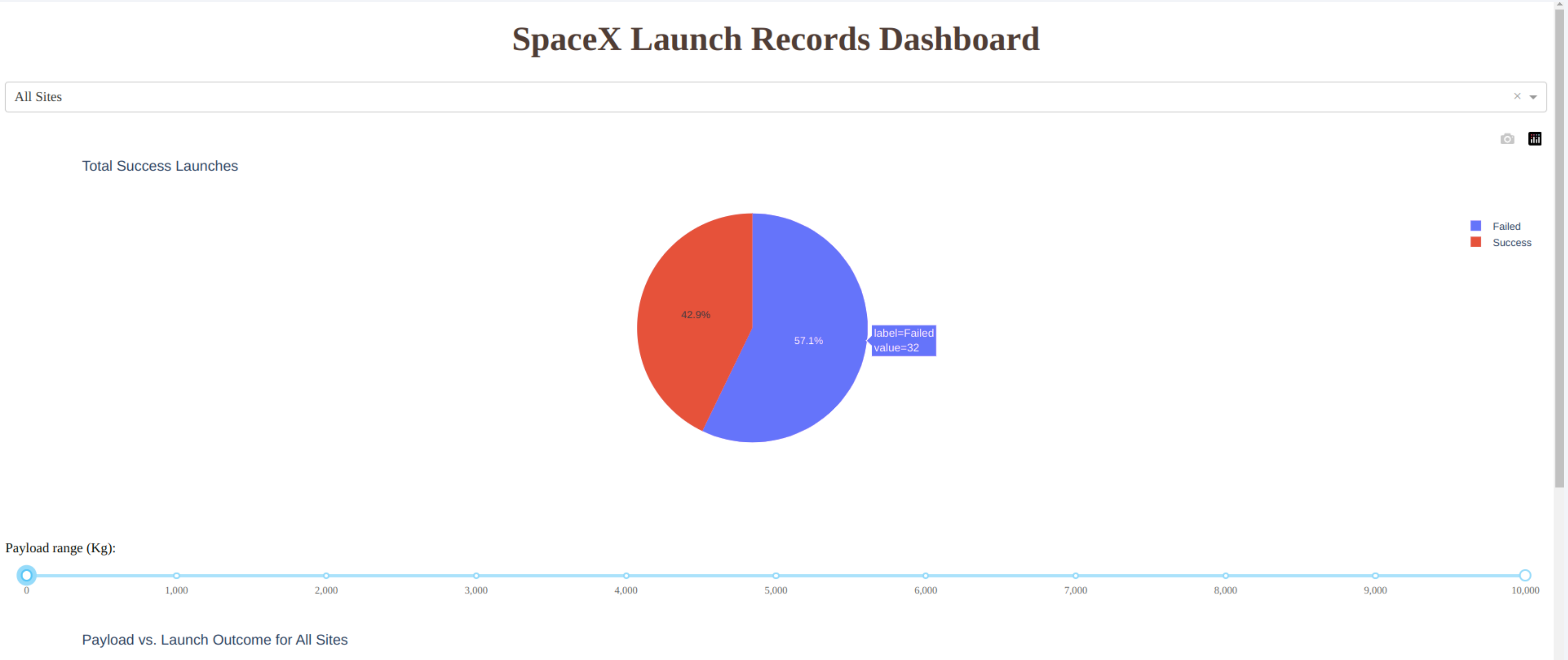
Section 4

Build a Dashboard with Plotly Dash



Dash displaying pie chart showing launch success count.

Failed lauches had a higher percentage with 57.1 %



SpaceX Launch Records Dashboard

All Sites

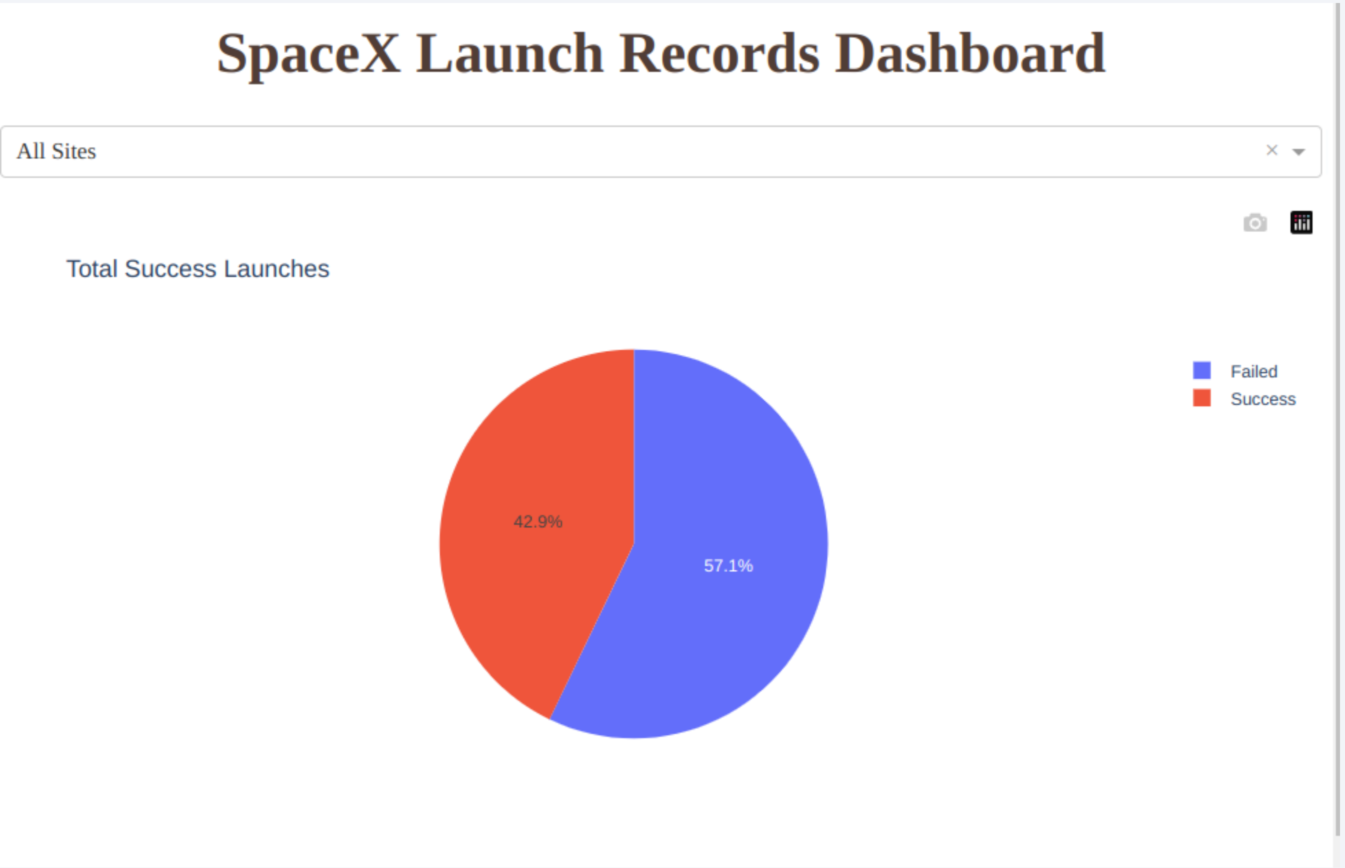
Total Success Launches

Payload range (Kg):

Payload vs. Launch Outcome for All Sites

Launch site with the highest launch success ratio.

Launch site with highest launch success ratio

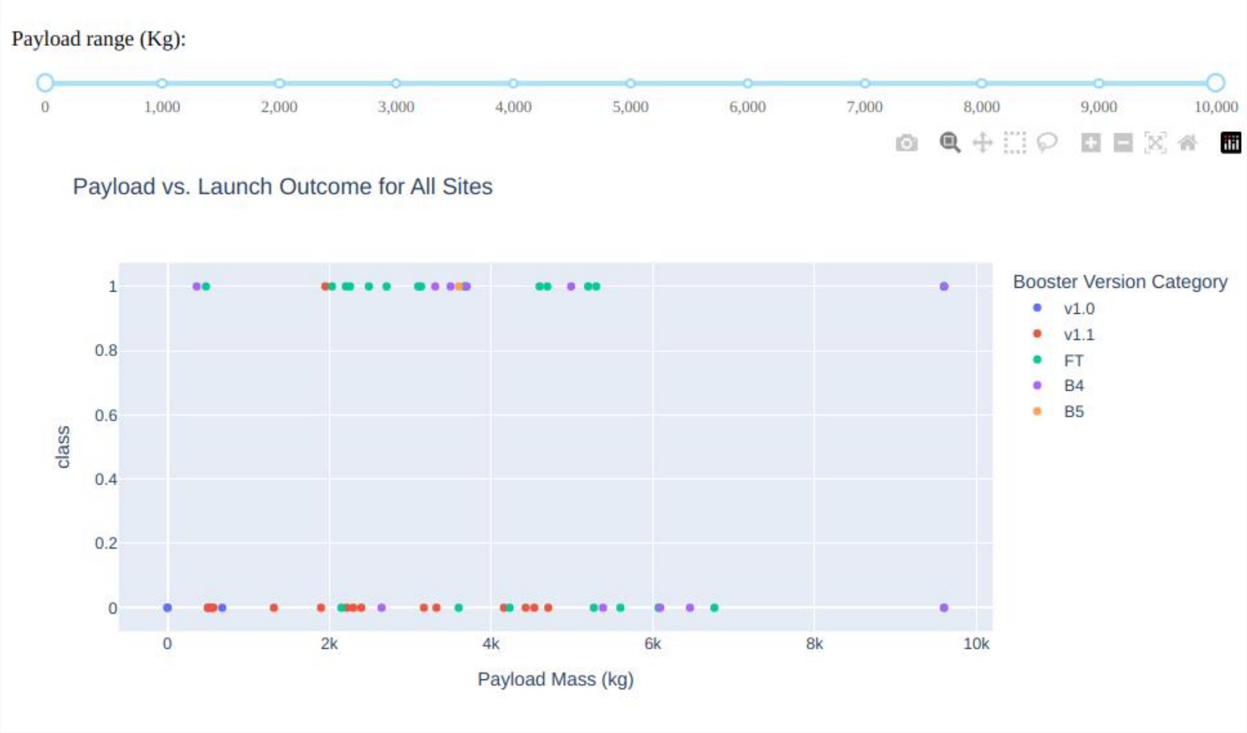


SpaceX Launch Records Dashboard

All Sites

Payload vs. Launch Outcome

Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider. From the scatter plot the B4 booster vesion managed a successful launch with a high payload mass.

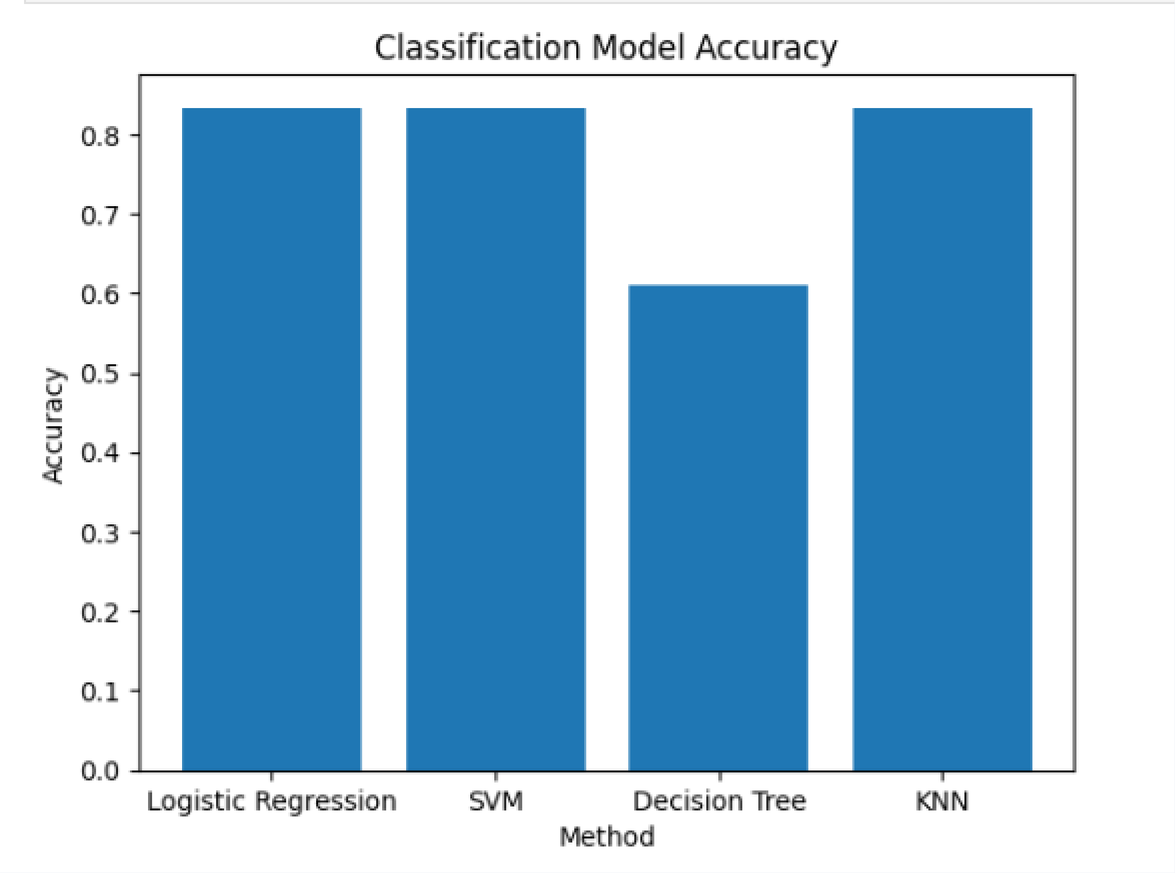


Section 5

Predictive Analysis (Classification)

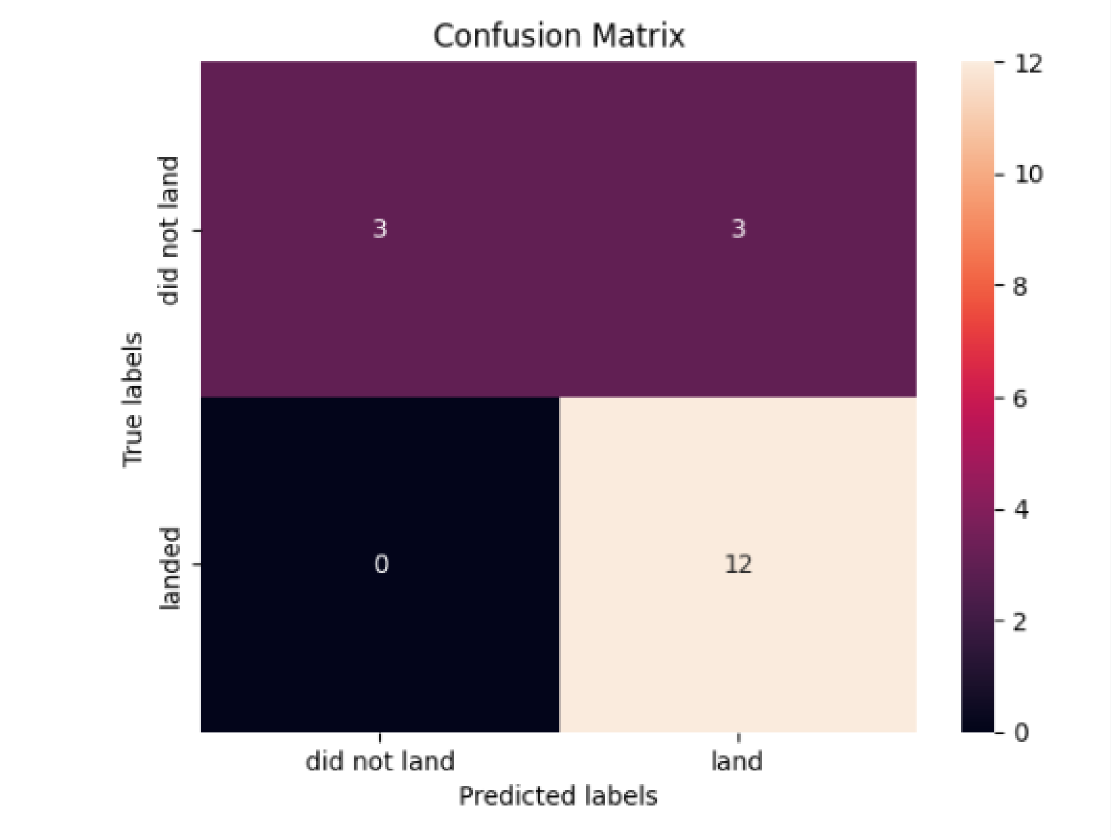
Classification Accuracy

- Logistic regression was the model that had the highest classification accuracy



Confusion Matrix

- Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the major problem is false positives.



Conclusions

1. Developed a reliable machine learning model (logistic regression) to accurately predict landing success of SpaceX's Falcon 9 first stage
2. Logistic regression achieved high accuracy in classifying landing outcomes on test data
3. Interactive visualizations and SQL queries revealed valuable insights:

A) Relationships between payload mass, flight number, booster version, and landing success rates

B) Folium map for geospatial analysis of launch sites

C) Plotly Dash dashboard for dynamic data exploration

Conclusions

4. Findings enable SpaceX competitors to:

- Estimate Falcon 9 launch costs for competitive bidding
- Assess risks associated with specific mission parameters
- Make informed investment decisions in reusable rocket technology

5. Need to continuously update predictive model with new SpaceX rocket versions and operational changes

6. Exemplifies power of data-driven approaches for:

- Understanding complex systems
- Making accurate predictions
- Gaining competitive edge in commercial space industry

7. Ongoing data collection and analysis crucial for staying ahead in rapidly evolving space sector

Appendix

Python code snippets, SQL queries, charts, Notebook outputs, data sets that I created during this project can be found on this github repo.

https://github.com/Wambugu-Muchemi/DSprojects

Thank you!