IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ju Yeon Eum
October 2024

Outline

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

Executive Summary

- **Project Overview**: This capstone project focuses on predicting the successful landing of SpaceX's Falcon 9 first stage. Accurate prediction can significantly reduce launch costs and improve competition in the aerospace industry.
- **Summary of Methodologies**:
  - **Data Collection**: Historical launch data was collected from the SpaceX API and web scraping Wikipedia.
  - **Data Wrangling**: The data was cleaned, and missing values were handled to ensure consistency.
  - **Exploratory Data Analysis (EDA)**: Various visualizations were created to explore relationships between key features such as payload mass, orbit type, and landing success.
  - **Predictive Modeling**: Machine learning models (Logistic Regression, SVM, Decision Tree) were trained and fine-tuned using GridSearchCV to find the best parameters for predicting landing success.
- **Summary of Results**:
  - **EDA Results**: A clear relationship between payload mass and landing success was identified, with heavier payloads generally correlating to lower success rates. Orbit type also played a significant role, with certain orbits (LEO, ISS) showing higher success rates.
  - **Predictive Model Performance**: Logistic Regression emerged as the best-performing model with a test accuracy of **83.33%**. Other models, such as SVM and Decision Trees, performed similarly but slightly lower.
  - **Visualization Tools**: Folium maps and Plotly Dash were used to create interactive visualizations that helped reveal geographical and operational patterns in the data.

Introduction

- **Project Background**:
  - SpaceX revolutionized the aerospace industry by significantly reducing launch costs, primarily due to the reuse of the Falcon 9 rocket's first stage. Understanding the factors that influence whether the first stage successfully lands can help further optimize these launches and offer valuable insights for competitors in the industry.
  - SpaceX advertises its Falcon 9 rocket launches at approximately **$62 million per launch**, whereas other providers charge over **$165 million**. The ability to predict successful landings can greatly reduce operational costs.
- **Problems to Find Answers**:
  - **What factors influence the success of Falcon 9's first stage landing?**
  - **How do variables such as payload mass, launch site, and orbit type affect the likelihood of a successful landing?**
  - **Can we build an accurate model to predict landing success, and which machine learning method will provide the best results?**

Section 1

Methodology

Methodology (1/2)

Executive Summary

The methodology summarizes the approach used to collect, process, analyze, and predict SpaceX Falcon 9 landing outcomes. We used various techniques including data wrangling, EDA, interactive analytics, and machine learning for predictive analysis.

Data collection methodology:

- Data was collected from two main sources:
  - **SpaceX API**: Real-time launch data for Falcon 9 was retrieved.
  - **Web Scraping**: Historical launch data was scraped from Wikipedia using BeautifulSoup.

Perform data wrangling:

- Data was cleaned and transformed. Missing values were handled, and the data was filtered to retain only Falcon 9 launches. Key information like payload mass, orbit types, and launch site data were included.

Describe how data was processed:

- The collected data was standardized and converted into a Pandas DataFrame. Categorical variables were converted into numerical format using techniques like OneHotEncoding, and missing payload values were filled using the mean.

Methodology (2/2)

Perform exploratory data analysis (EDA) using visualization and SQL:

- SQL queries were executed to gain insights such as the total number of successful launches per site, and various visualizations like histograms and scatter plots were used to examine relationships between flight numbers, payload mass, and launch outcomes.

Perform interactive visual analytics using Folium and Plotly Dash:

- **Folium**: Used to visualize the geographic location of SpaceX launch sites. Success and failure markers were plotted for each launch site.
- **Plotly Dash**: Created an interactive dashboard that allows filtering of launch data by site and payload mass to analyze success rates dynamically.

Perform predictive analysis using classification models:

- Supervised learning techniques were applied to predict the success of Falcon 9 landings. Logistic Regression, Support Vector Machines, Decision Trees, and K-Nearest Neighbors were employed, with the performance of each model assessed through accuracy scores.

How to build, tune, evaluate classification models:

- Models were built and tuned using **GridSearchCV** to find the best hyperparameters. Each model was evaluated using accuracy metrics and confusion matrices to compare their performance on the test dataset.

Data Collection

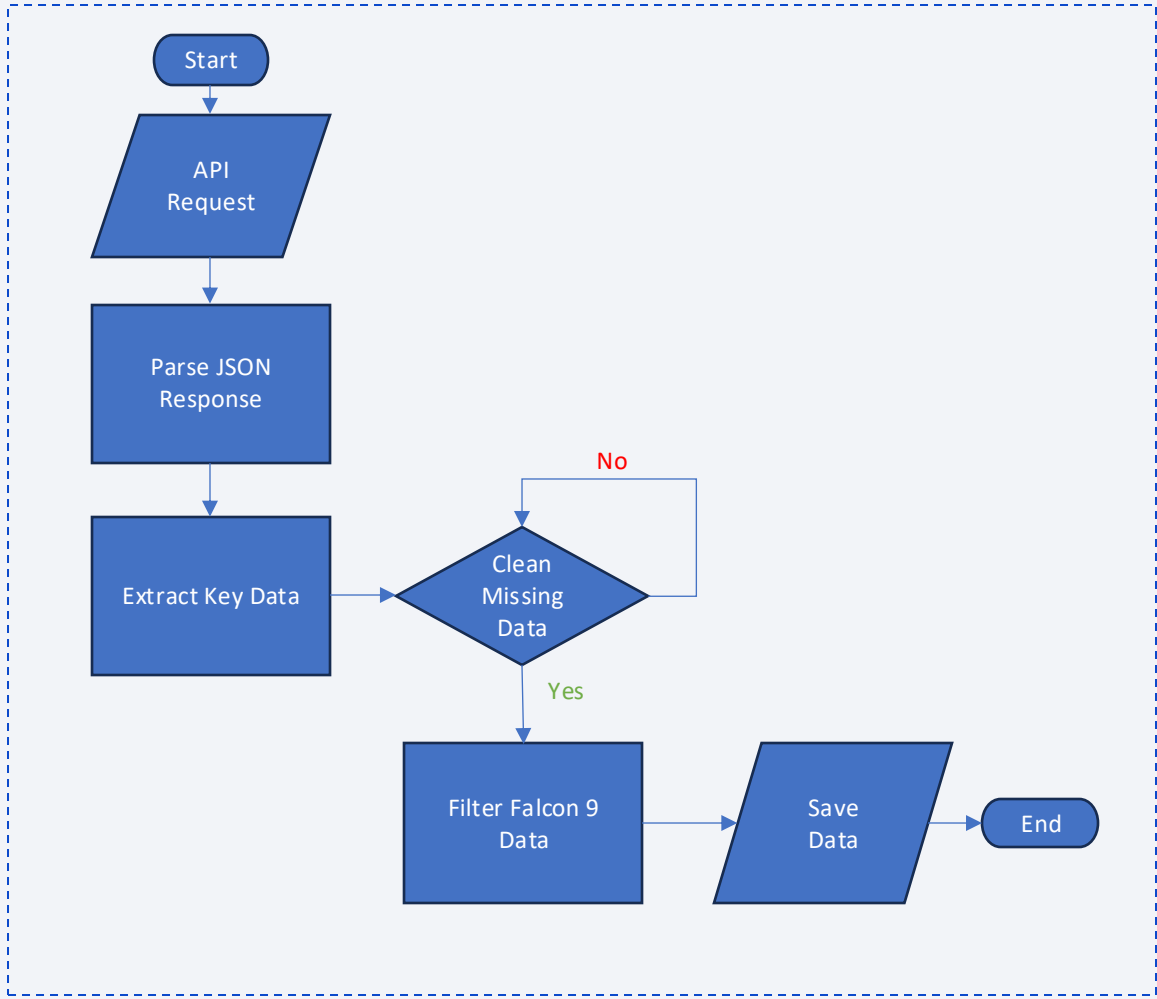
- How data sets were collected.
    - **SpaceX API**: We retrieved real-time data from the SpaceX API, including information such as the rocket, launchpad, payload mass, and core details. The API provided extensive data on Falcon 9 launches, which we used for building our predictive models.
    - **Web Scraping**: Historical data was gathered using BeautifulSoup to scrape tables from Wikipedia. This included details of previous Falcon 9 and Falcon Heavy launches, which supplemented the real-time API data for more comprehensive analysis.
- **Data Collection Process**:
    - The data collection process involved:
        - **API Requests**: Making GET requests to the SpaceX API for extracting data.
        - **Data Parsing**: Extracting relevant information such as rocket type, payload mass, launch site, and landing outcomes.
        - **Web Scraping**: Using BeautifulSoup to extract launch data tables from the Wikipedia page for historical launches.
        - **Data Integration**: Combining data from the SpaceX API and Wikipedia into a unified dataset for further analysis.

Data Collection – SpaceX API

1. **Identify API Endpoint:** Use the SpaceX API to request past launch data from the endpoint https://api.spacexdata.com/v4/launches/past.
2. **Make API Request:** Use the Python requests library to send a GET request to retrieve the historical data.
3. **Parse JSON Response:** Convert the response from JSON format into a pandas DataFrame using json_normalize.
4. **Extract Key Data:** Identify and extract relevant columns such as rocket, payload, launchpad, and cores to get booster version, payload mass, orbit type, and landing outcomes.
5. **Handle Missing Data:** Clean the data by filling missing values, particularly for fields like Payload Mass.
6. **Filter Falcon 9 Data:** Filter the data to only include Falcon 9 launches.
7. **Save Processed Data:** Store the cleaned and filtered data in a CSV file for further analysis.

GitHub URL of SpaceX API calls - notebook

GitHub URL of SpaceX API calls – HTML



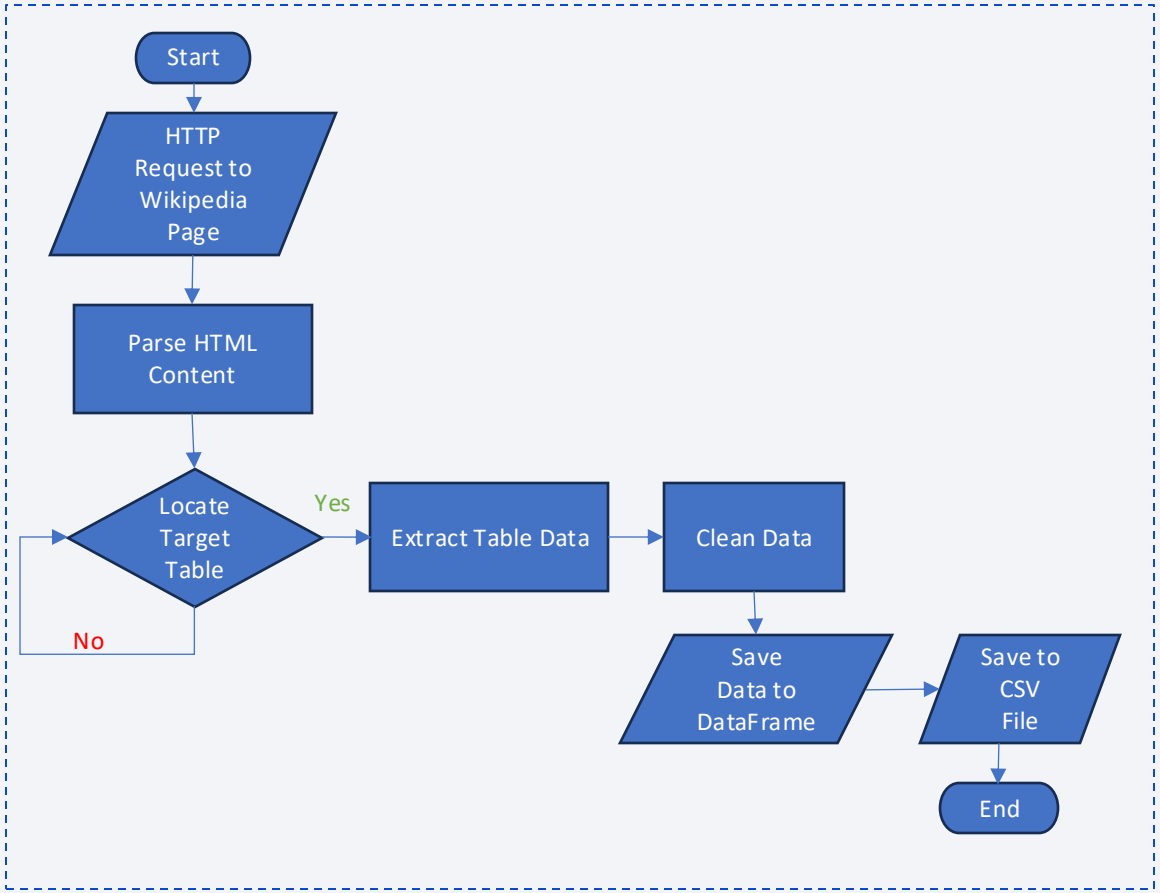
The interactive function may not display properly on GitHub, so an HTML file also has been provided

Data Collection - Scraping

1. **HTTP Request**: Send GET request to Wikipedia using the requests library.
2. **Parse HTML**: Use BeautifulSoup to parse the page's HTML content.
3. **Locate Table**: Use BeautifulSoup to find the launch table based on HTML structure.
4. **Extract Data**: Scrape the table rows to retrieve launch details.
5. **Clean Data**: Remove unnecessary rows and handle any missing or malformed data.
6. **Save Data**: Convert the cleaned table into a Pandas DataFrame and export it to CSV.

GitHub URL of the web scraping – notebook

GitHub URL of the web scraping – HTML



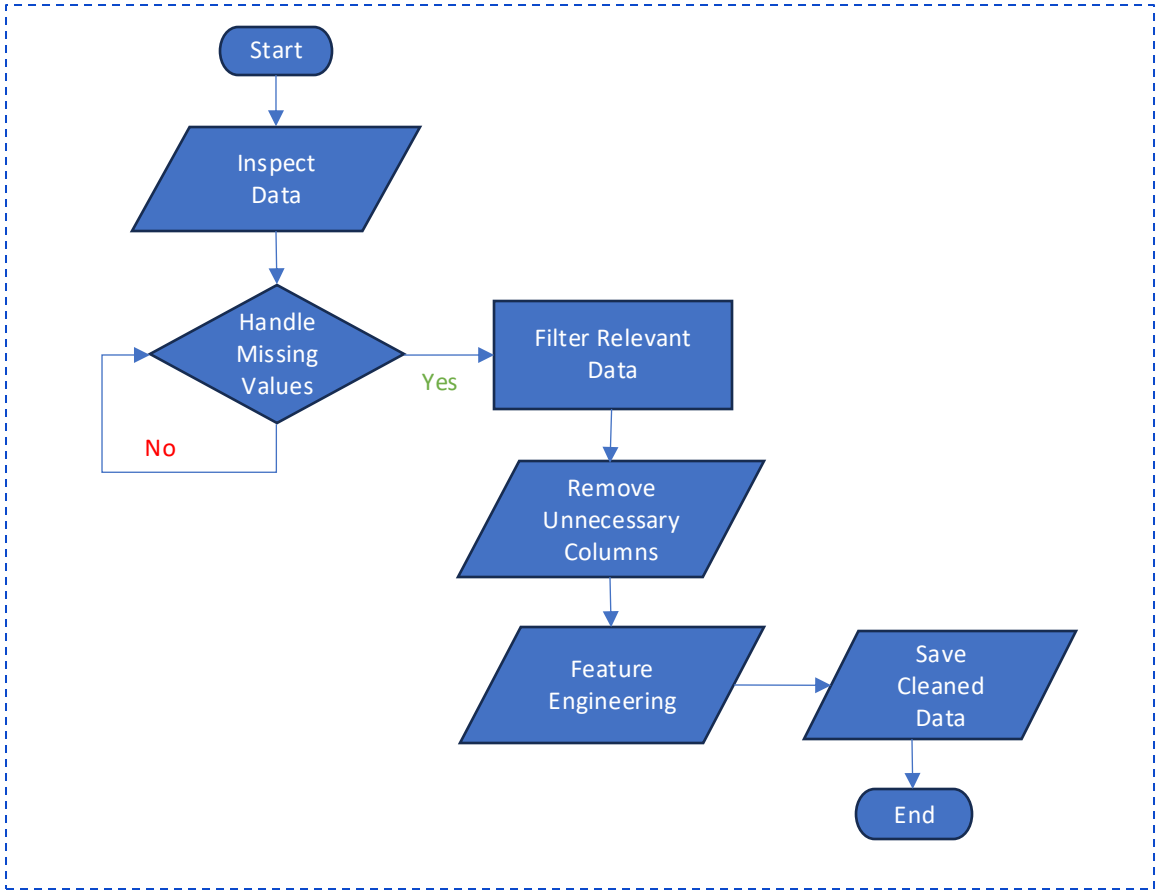
The interactive function may not display properly on GitHub, so an HTML file also has been provided

Data Wrangling

1. **Inspect Data** : Load the dataset and check for inconsistencies.
2. **Handle Missing Values** : Identify columns with missing values (e.g., Payload Mass).
3. **Filter Relevant Data** : Filter rows to include only Falcon 9 launches.
4. **Remove Unnecessary Columns** : Drop columns that are not relevant (e.g., Flight Number, if not needed).
5. **Feature Engineering** : Create new features or labels, e.g., convert landing outcome into binary class (1 for success, 0 for failure).
6. **Save Cleaned Data** : Save the cleaned and processed data for further analysis.

GitHub URL of completed data wrangling – notebook

GitHub URL of completed data wrangling – HTML



The interactive function may not display properly on GitHub, so an HTML file also has been provided

EDA with Data Visualization

- **Flight Number vs Launch Site (Scatter Plot)**
  - To visualize the relationship between the number of flights and the launch site.
  - A scatter plot helps observe trends between different launch sites and the frequency of launches. We can also detect if certain sites have a higher success rate with more launches.
- **Payload Mass vs Launch Site (Scatter Plot)**
  - To analyze if the payload mass influences launch outcomes at different sites.
  - It helps in understanding if heavier payloads are more successful at specific sites or if certain sites are better suited for heavier launches.
- **Success Rate by Orbit Type (Bar Chart)**
  - To compare the success rates of different orbit types.
  - Bar charts are great for comparing categorical variables like orbit type, helping to easily visualize which orbits have the highest success rates.
- **Flight Number vs Orbit Type (Scatter Plot)**
  - To explore the relationship between flight numbers and orbit types.
  - This scatter plot helps visualize whether the flight experience (number of flights) affects success based on the type of orbit.
- **Payload Mass vs Orbit Type (Scatter Plot)**
  - To analyze how payload mass correlates with different orbit types.
  - This chart helps to examine if certain orbit types can handle heavier payloads more successfully.
- **Launch Success Yearly Trend (Line Chart)**
  - To track the trend of launch success rates over the years.
  - Line charts are effective for visualizing trends over time. This chart helps identify if SpaceX has improved its success rate over the years.

GitHub URL of completed EDA with data visualization – notebook

GitHub URL of completed EDA with data visualization – HTML

The interactive function may not display properly on GitHub, so an HTML file also has been provided

EDA with SQL

- **Unique Launch Sites**: To identify the different launch sites used for Falcon 9 launches.
  - *Query*: SELECT DISTINCT launch_site FROM spacex_data;
- **Launch Sites Starting with 'CCA'**: To display records of launch sites beginning with "CCA" for detailed analysis.
  - *Query*: SELECT * FROM spacex_data WHERE launch_site LIKE 'CCA%';
- **Total Payload Mass by NASA (CRS) Missions**: To calculate the total payload mass carried by NASA's Commercial Resupply Services (CRS) missions.
  - *Query*: SELECT SUM(payload_mass) FROM spacex_data WHERE customer = 'NASA (CRS)';
- **Average Payload Mass for Booster Version F9 v1.1**: To determine the average payload mass for the booster version F9 v1.1.
  - *Query*: SELECT AVG(payload_mass) FROM spacex_data WHERE booster_version = 'F9 v1.1';
- **First Successful Landing Date on Ground Pad**: To find the date of the first successful landing on a ground pad.
  - *Query*: SELECT MIN(launch_date) FROM spacex_data WHERE landing_outcome = 'Success (ground pad)';
- **Boosters with Success on Drone Ship and Payload Between 4000-6000 kg**: To list boosters with successful landings on drone ships and payloads within a specific mass range.
  - *Query*: SELECT booster_version FROM spacex_data WHERE landing_outcome = 'Success (drone ship)' AND payload_mass BETWEEN 4000 AND 6000;
- **Total Number of Successful and Failed Landings**: To tally successful and failed landings across all missions.
  - *Query*: SELECT landing_outcome, COUNT(*) FROM spacex_data GROUP BY landing_outcome;
- **Booster Versions with Maximum Payload Mass**: To identify booster versions that carried the heaviest payloads.
  - *Query*: SELECT booster_version FROM spacex_data WHERE payload_mass = (SELECT MAX(payload_mass) FROM spacex_data);
- **Landing Outcomes in 2015 with Failure on Drone Ships**: To analyze failed drone ship landings in 2015, along with booster versions and launch sites.
  - *Query*: SELECT SUBSTR(launch_date, 6, 2) AS month, booster_version, launch_site FROM spacex_data WHERE SUBSTR(launch_date, 1, 4) = '2015' AND landing_outcome = 'Failure (drone ship)';
- **Ranked Landing Outcomes Between 2010 and 2017**: To rank the count of landing outcomes between 2010 and 2017 in descending order.
  - *Query*: SELECT landing_outcome, COUNT(*) FROM spacex_data WHERE launch_date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY landing_outcome ORDER BY COUNT(*) DESC;

GitHub URL of completed EDA with SQL– notebook

GitHub URL of completed EDA with SQL- HTML

The interactive function may not display properly on GitHub, so an HTML file also has been provided

Build an Interactive Map with Folium

- **Launch Site Markers & Circles**:
  - *Markers*: Placed at each SpaceX launch site to show their precise location (CCAFS LC-40, KSC LC-39A, etc.).
  - *Circles*: Added around launch sites to highlight their proximity to geographical landmarks such as coastlines, roads, and cities.
- **Success/Failure Markers**:
  - *Markers*: Green markers represent successful launches, while red markers show failed ones. These markers help visualize the success rate of launches at each site.
- **Clustered Markers**:
  - *Purpose*: To manage the overlapping of multiple markers for the same coordinates, a cluster feature is used. This groups launches from the same site, making the map clearer.
- **Distance Lines Between Sites and Proximity Points**:
  - *Lines*: Drawn between launch sites and nearby infrastructure (e.g., coastline, railways, highways, and cities) to analyze how launch site proximity affects operations.
- **MousePosition Feature**:
  - *Purpose*: Allows users to identify exact geographical coordinates of various points on the map by moving the cursor, ensuring precision when analyzing site proximity.

These objects were added to visually assess how launch site success correlates with infrastructure proximity, logistical factors, and site-specific outcomes.

GitHub URL of completed interactive map with Folium map – notebook

GitHub URL of completed interactive map with Folium map – HTML

The interactive function may not display properly on GitHub, so an HTML file also has been provided

Build a Dashboard with Plotly Dash

Summary of Dashboard Plots/Graphs and Interactions:

1. **Dropdown for Launch Site Selection**:
    - **Why**: Allows users to filter data by different launch sites to view specific insights per location, enhancing the interactivity and user control over the analysis.
2. **Pie Chart for Launch Success Rate**:
    - **Why**: Provides a clear visual representation of the success vs. failure rates for each launch site. This helps users quickly assess the performance of SpaceX at various locations.
3. **Payload Range Slider**:
    - **Why**: Allows users to explore the relationship between payload mass and launch success. It enables filtering data for specific payload mass ranges and observing how this factor affects the outcome of a launch.
4. **Scatter Plot for Payload Mass and Success Correlation**:
    - **Why**: Demonstrates the relationship between payload mass and the success of a launch. It helps to identify trends and patterns, such as whether higher payloads correlate with more or less successful landings.

These visualizations and interactions provide an intuitive way for users to explore the data and uncover trends related to SpaceX launches, including the impact of different factors on launch success.

GitHub URL of Plotly Dash - Python

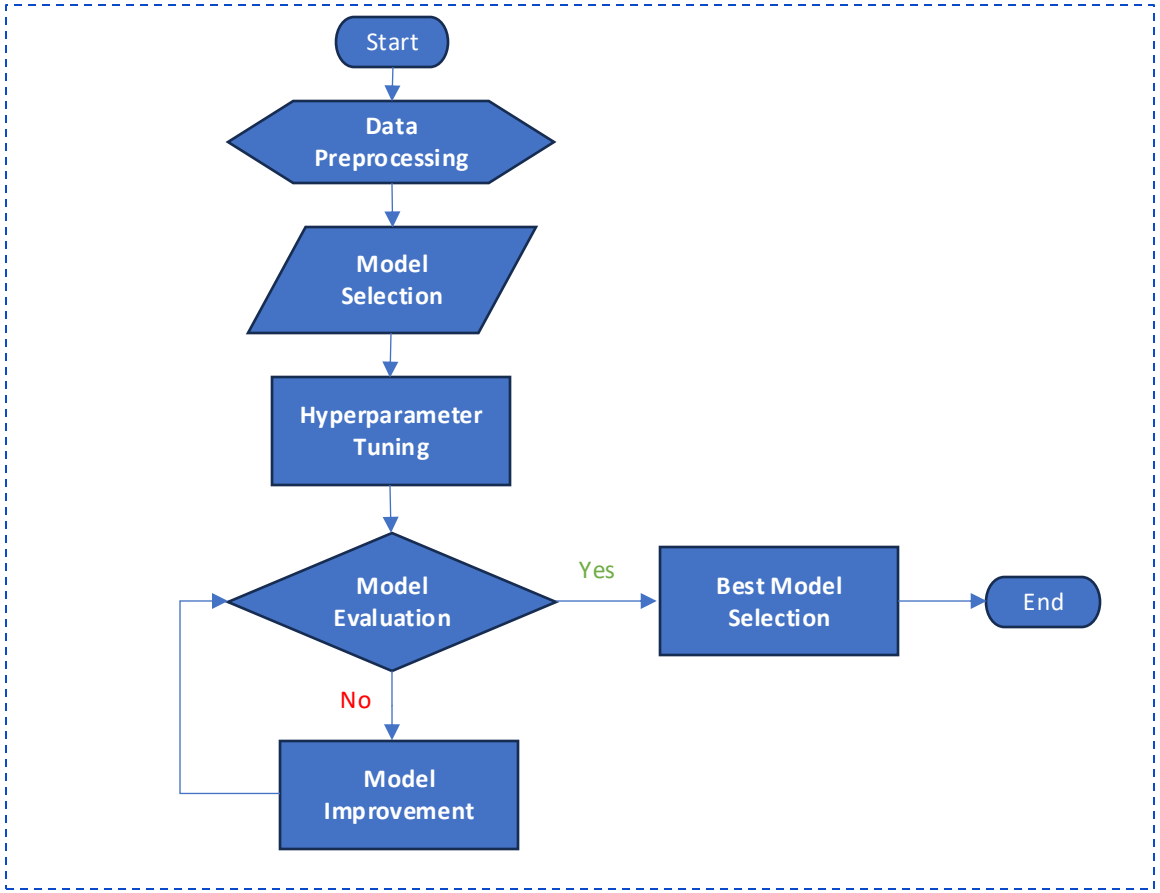
Predictive Analysis (Classification)

Model Development Process

1. **Data Preprocessing:**
   - Standardized features
   - Split data (80% training, 20% testing)
2. **Model Selection:**
   - Logistic Regression, SVM, Decision Tree, KNN
3. **Hyperparameter Tuning:**
   - Used GridSearchCV (10-fold cross-validation)
4. **Model Evaluation:**
   - Accuracy Score
   - Confusion Matrix
5. **Model Improvement:**
   - Tuned parameters for the best performance
6. **Best Model Selection:**
   - Selected based on test accuracy and confusion matrix results

GitHub URL of predictive analysis lab – notebook

GitHub URL of predictive analysis lab – HTML



The interactive function may not display properly on GitHub, so an HTML file also has been provided

Results (1/3)

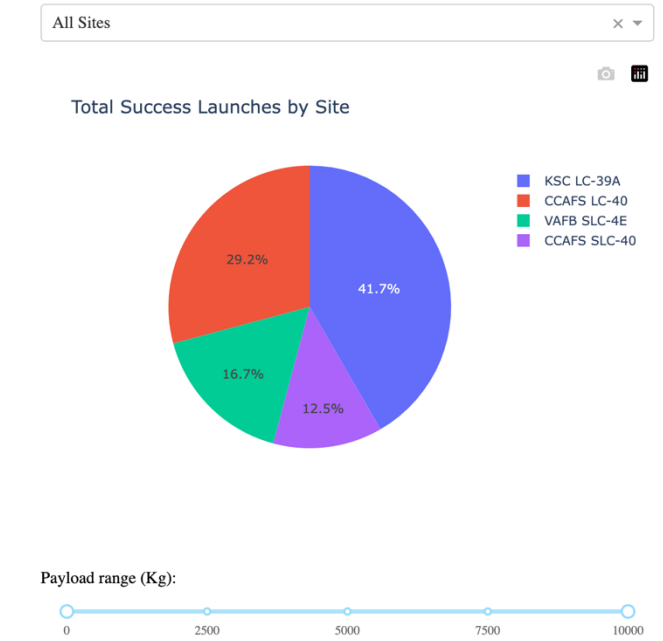
1. Exploratory Data Analysis Results

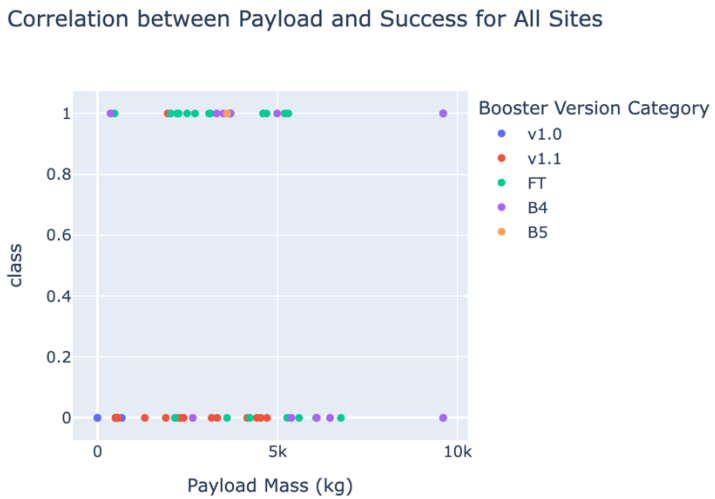
- **Payload Mass Distribution**: Visualized to understand how payload mass affects launch success.
- **Flight Number vs Launch Site**: Explored relationships between the number of flights and success rate by site.
- **Success Rate by Orbit Type**: Bar chart showing success rates across different orbits to determine which orbits are most reliable.
- **Yearly Success Trend**: Line chart visualizing the trend of successful launches over the years.

Results (2/3)

2. [Interactive Analytics Demo]

- **Folium Map**: Added markers for each launch site, color-coded based on success/failure.
- **Plotly Dash**:
  - **Launch Site Dropdown**: Allows users to filter data based on launch sites.
  - **Payload Slider**: Filters data based on payload mass to see the impact on launch success.
  - **Scatter Plot**: Shows correlation between payload mass and success rate, categorized by booster version.

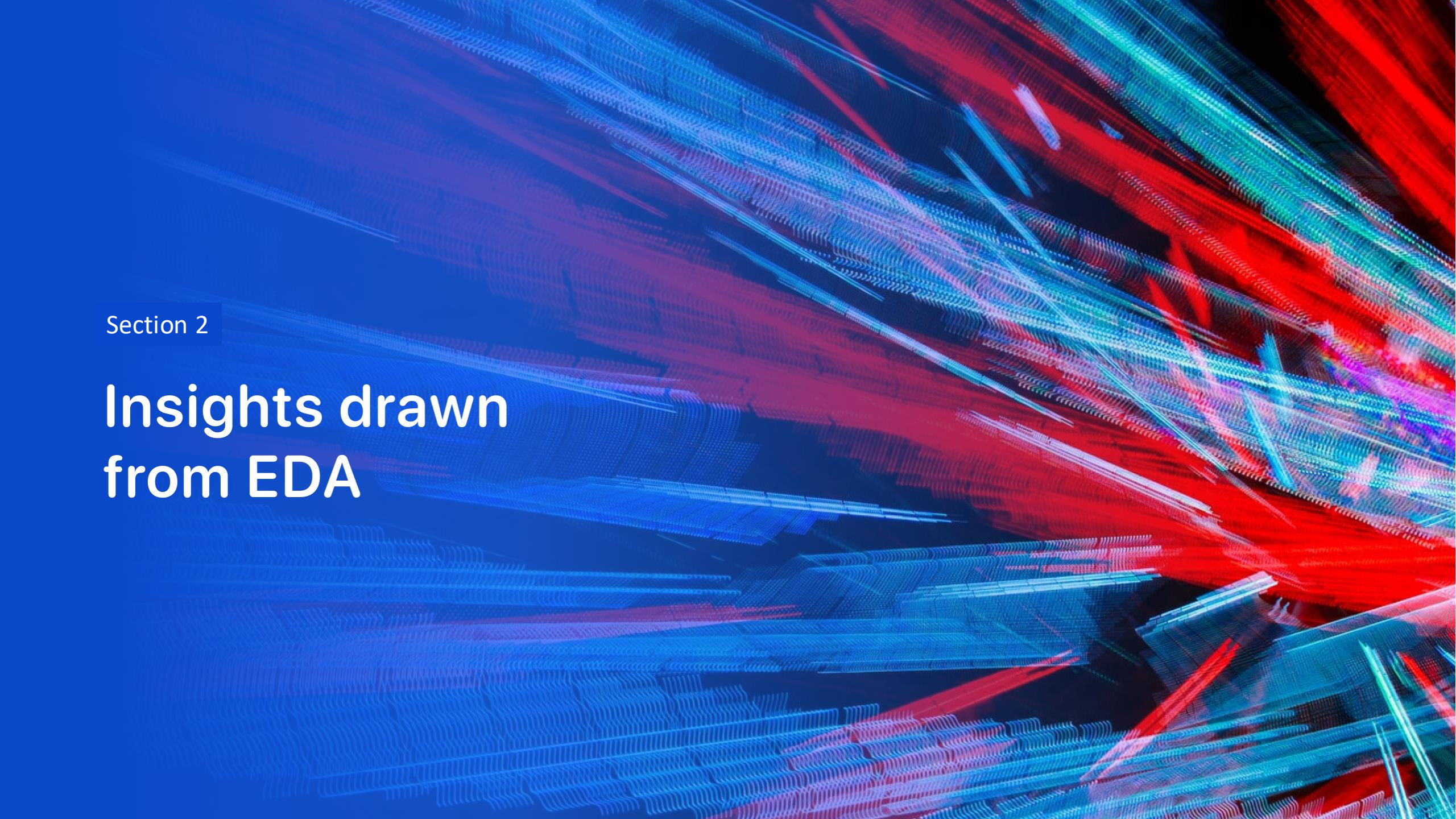




Results (3/3)

3. Predictive Analysis Results

- **Best Performing Model**: Logistic Regression achieved the highest accuracy after hyperparameter tuning, with a test accuracy of 83.33%.
- **Confusion Matrix**: Used to evaluate classification performance, with minimal false positives.
- **Model Improvement**: Hyperparameters were tuned for better accuracy and generalization, ensuring a robust predictive model.


Section 2
Insights drawn from EDA

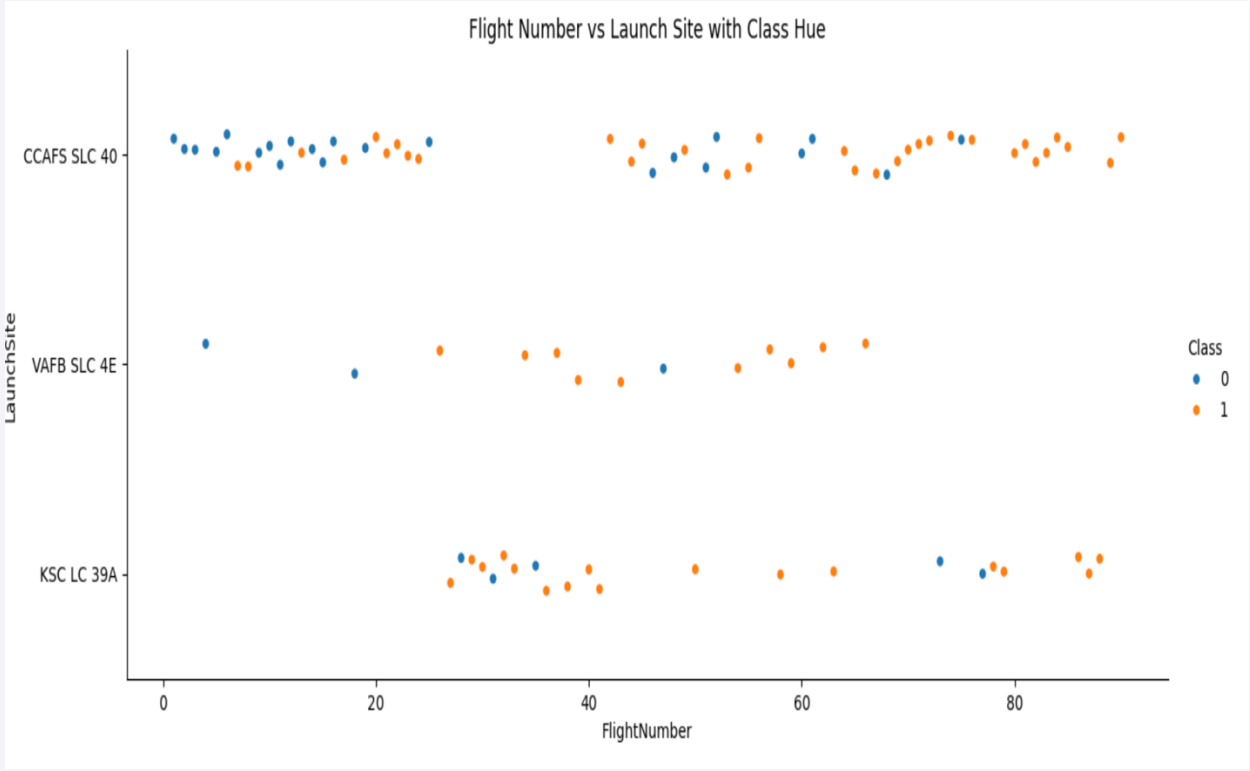
Flight Number vs. Launch Site

This scatter plot visualizes the **relationship between Flight Number and Launch Site** with success class indicated by color:

- **Blue (Class 0)**: Unsuccessful landings.
- **Orange (Class 1)**: Successful landings. The X-axis represents the flight number (indicating the number of attempts), and the Y-axis identifies the launch site for each launch.

Key Insights:

- **CCAFS SLC 40**: This site shows a **consistent improvement** in success rates over time (flight number). As flight numbers increase, more successful landings (orange) are observed, indicating better performance with experience.
- **VAFB SLC 4E**: There are **fewer launches from this site**, and most of the launches show a mixed pattern of success and failure. No clear trend of improvement is visible based on flight numbers.
- **KSC LC 39A**: This site also shows a **mixed performance** across flight numbers. There are successful landings, but failures are also present even with higher flight numbers, indicating variability in performance across different launches.



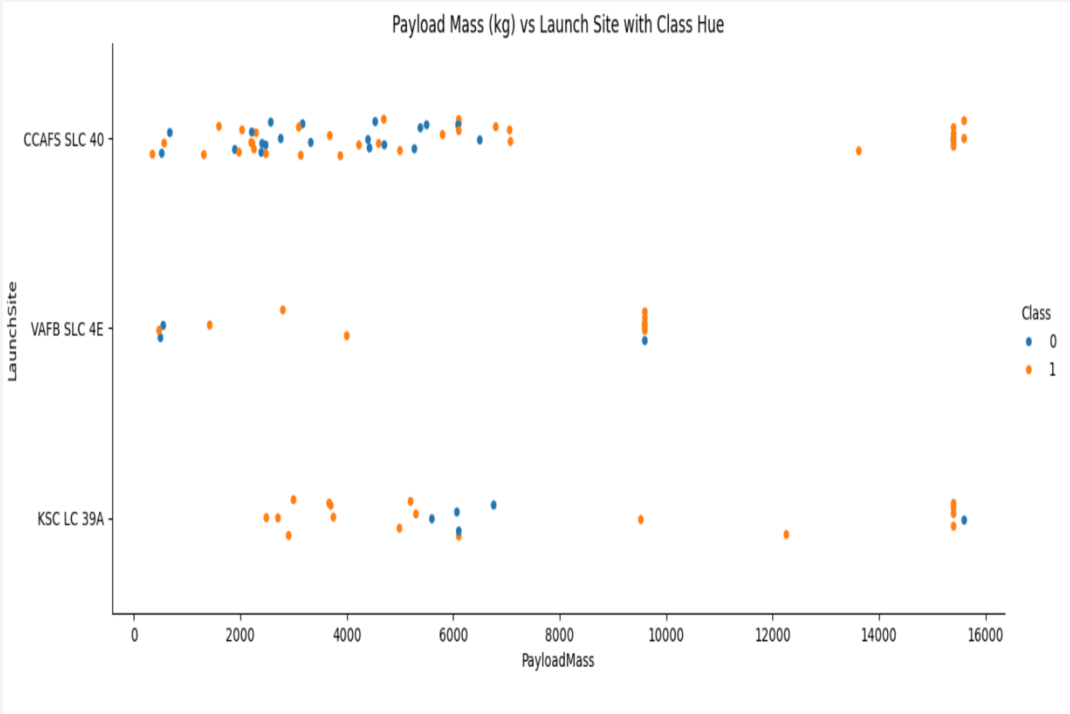
Payload vs. Launch Site

This scatter plot illustrates the **relationship between Payload Mass (in kilograms) and Launch Sites** with the success class as the color hue. Each point represents a SpaceX launch, where:

- **Blue (Class 0)**: Unsuccessful landings.
- **Orange (Class 1)**: Successful landings. The X-axis indicates the payload mass, while the Y-axis identifies the launch site. There are three launch sites represented: **CCAFS SLC-40**, **VAFB SLC 4E**, and **KSC LC 39A**.

Key Insights:

- **CCAFS SLC 40**: This site shows a **wider range of payloads**, with successful landings (orange) clustering mostly around the mid to high payload range. However, we also observe failures across various payload sizes, making it a launch site with mixed results.
- **VAFB SLC 4E**: Launches from this site tend to have **lower payload masses** compared to others. The majority of launches here have been successful, with only a few failures for small payloads.
- **KSC LC 39A**:This site has a **diverse payload range**, including many heavier payloads.The successful launches are distributed across the payload spectrum, although failures occur at the lower and mid-range payload masses.
- **Payload Mass Influence**: There is **no clear linear relationship** between payload mass and success rate at any specific launch site. Both successful and failed launches occur across a range of payload masses, suggesting that other factors (like launch conditions or orbit types) may also play a role in determining success.



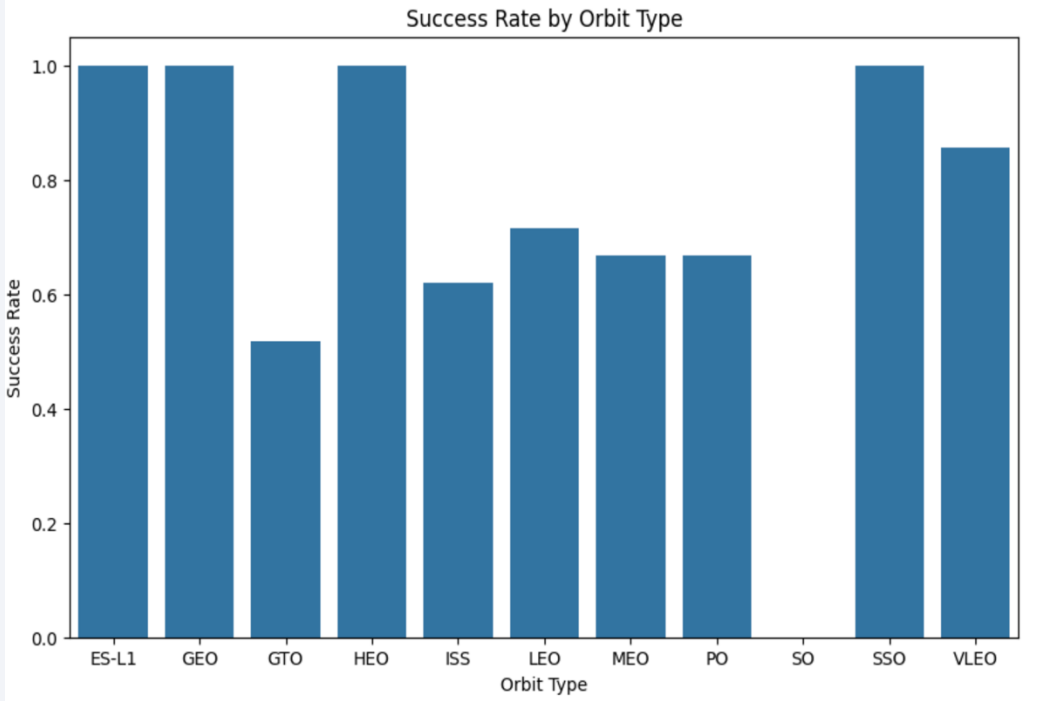
Success Rate vs. Orbit Type

This bar chart shows the **success rate** for different orbit types that SpaceX has launched rockets into. Each bar represents a specific orbit type (e.g., LEO, GTO, SSO), and the height of the bar reflects the percentage of successful landings for that orbit type.

- **X-axis**: Orbit Type (e.g., LEO, GTO, SSO)
- **Y-axis**: Success Rate (percentage of successful landings)

Key Insights:

- **ES-L1**, **GEO**, **HEO**, and **SSO** orbits all share the highest success rate, suggesting that launches to these orbits have been highly reliable.
- **LEO**, **VLEO**, and **PO** show strong success rates as well, though they are slightly lower compared to the perfect success rate of the top group.
- **GTO** has the lowest success rate, indicating that launches to this orbit are the most challenging and prone to failure.



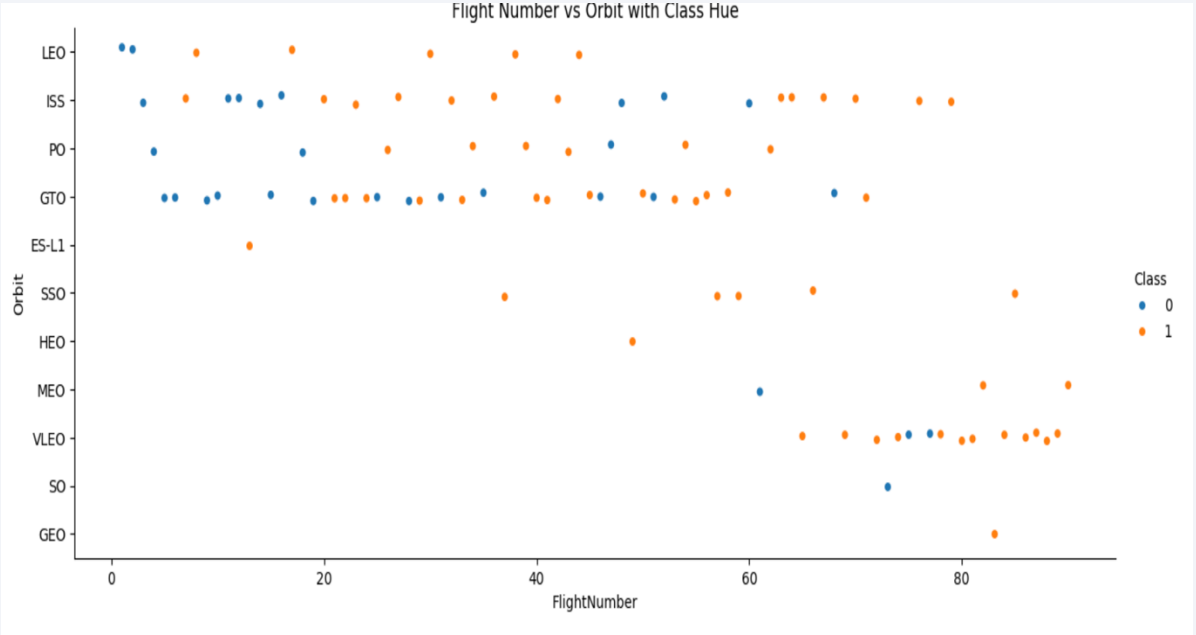
Flight Number vs. Orbit Type

This scatter plot illustrates the **relationship between Flight Number and Orbit type** with the class of the flight indicated by color:

- **Blue (Class 0)**: Unsuccessful landings.
- **Orange (Class 1)**: Successful landings. The X-axis represents the flight number, and the Y-axis categorizes different orbit types, such as LEO, ISS, and GTO.

Key Insights:

- **LEO (Low Earth Orbit)**: There is a **clear pattern of improvement** in landing success with increasing flight numbers, indicating that as more launches to LEO were made, SpaceX was able to improve the success rate.
- **GTO (Geostationary Transfer Orbit):** The success rate remains **mixed across flight numbers**, indicating that flight number does not significantly affect success for this orbit. Many flights in GTO were unsuccessful.
- **VLEO and ISS**: These orbits show an **increasing success rate** with higher flight numbers, with more successful landings in later flights. This suggests that operational experience positively impacted these orbits' outcomes.
- **Other Orbits**: Some orbits like ES-L1, SSO, and GEO have a **limited number of flights**, so trends are less clear, but there are signs of improvement with more flights for certain orbits like SSO.



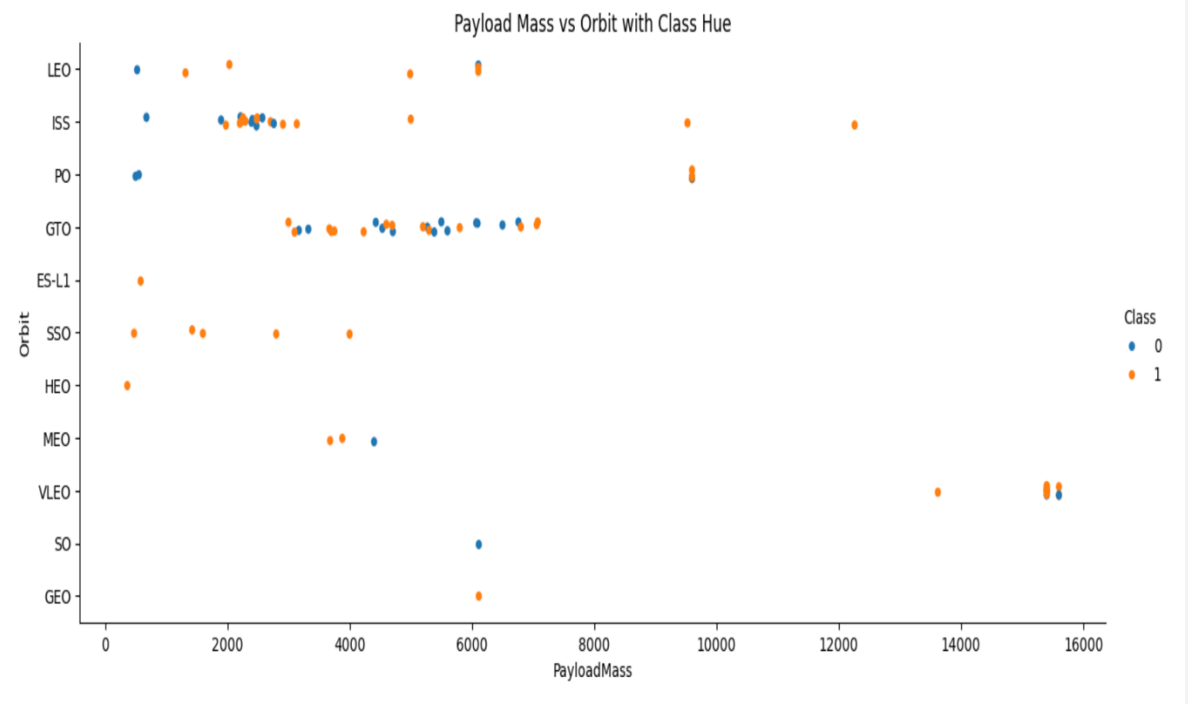
Payload vs. Orbit Type

In this scatter plot, the relationship between **Payload Mass and Orbit type** is examined, specifically focusing on the success (Class) of each launch.

- **X-axis**: Payload Mass (kg)
- **Y-axis**: Orbit Type
- **Class 1 (Orange)**: Successful landings
- **Class 0 (Blue)**: Unsuccessful landings

Key Insights:

- **Polar, LEO, and ISS Orbits**: Heavier payloads in these orbits (Payload Mass > 10,000 kg) have a relatively higher rate of successful landings. The orange dots representing successful missions dominate these orbits, indicating a positive trend for handling larger payloads.
- **GTO Orbit**: The results in GTO (Geostationary Transfer Orbit) show both successful and unsuccessful landings with a wide range of payload masses. This suggests **greater variability** in landing outcomes, making it harder to clearly distinguish whether heavier payloads affect success in GTO.



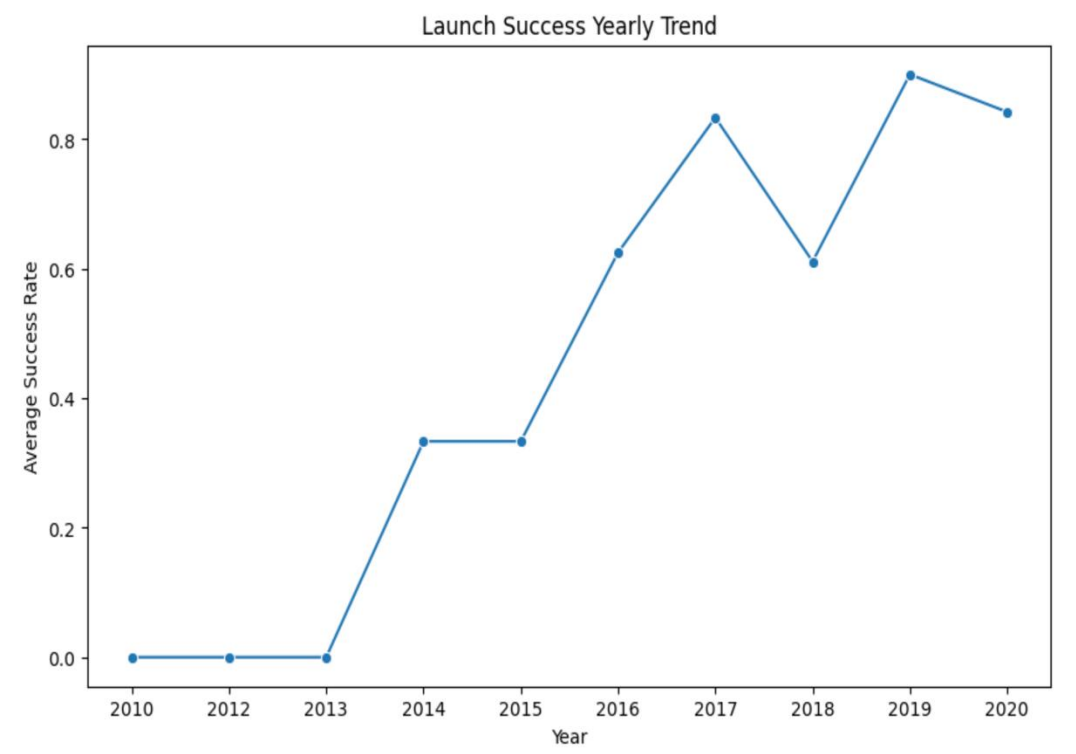
Launch Success Yearly Trend

This line chart depicts the yearly trend of the average **launch success rate** for SpaceX from 2010 to 2020.

- **X-axis**: Year (2010-2020)
- **Y-axis**: Average Success Rate (ranging from 0.0 to 1.0)

Key Insight:

- From **2013 onwards**, the success rate began to climb steadily.
- By **2015**, the success rate saw a significant jump, with each subsequent year showing consistent improvement.
- The peak success rate occurred in **2019**, reaching close to **90%**, before a slight dip in **2020**.



All Launch Site Names

The query retrieves the **unique names** of all **launch sites** where SpaceX launches occurred. There are three distinct launch sites in the dataset: **CCAFS SLC 40** (Cape Canaveral), **VAFB SLC 4E** (Vandenberg Air Force Base), and **KSC LC 39A** (Kennedy Space Center). Each site represents a different geographic location used for launching SpaceX rockets.

```
%%sql
SELECT DISTINCT Launch_Site FROM SPACEXTABLE;
```

```
 * 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'

The query selects the **first 5 records** where the **launch site name** begins with 'CCA', which corresponds to **CCAFS SLC 40** (Cape Canaveral). This site is one of the primary launch sites for SpaceX, and the records show different missions, payloads, and their success or failure outcomes. The LIKE 'CCA%' condition ensures only launch sites that start with 'CCA' are retrieved.

```
%%sql
SELECT * FROM SPACEXTABLE
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_
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (p
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 (p
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	N
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	N
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	N

Total Payload Mass

The query calculates the **total payload mass** carried by boosters that were launched by **NASA**. The SUM(PayloadMass) function adds up the payloads for all NASA-related launches. In this case, the total payload is **48,213 kg** across all missions launched for NASA.

```
%%sql
SELECT SUM("PAYLOAD_MASS__KG_")
FROM SPACEXTABLE
WHERE "Customer" LIKE '%NASA (CRS)%';
```

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

SUM("PAYLOAD_MASS__KG_")
48213

Average Payload Mass by F9 v1.1

The query calculates the **average payload mass** carried by the **F9 v1.1** booster version. The AVG(PayloadMass) function computes the average of the payloads for all missions launched using the **F9 v1.1** booster. In this case, the average payload mass is **2,928.4 kg**.

```
%%sql
SELECT AVG("PAYLOAD_MASS__KG_")
FROM SPACEXTABLE
WHERE "Booster_Version" = 'F9 v1.1';
```

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

AVG("PAYLOAD_MASS__KG_")
2928.4

First Successful Ground Landing Date

The query finds the **earliest date** (using MIN(Date)) when a successful landing was achieved on a **ground pad**. The condition Landing_Outcome = 'Success' filters only the successful landings, and the condition Landing_Pad = 'Ground Pad' ensures that only landings on ground pads are considered. The result indicates that the **first successful landing on a ground pad** occurred on **December 22, 2015**.

```
%%sql
SELECT MIN("Date")
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

MIN("Date")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

The query retrieves the **names of booster versions** that have successfully landed on a **drone ship** and had a **payload mass** between **4000 and 6000 kg**. The conditions ensure that only those missions with a successful landing on a drone ship and within the specified payload range are listed.

```
%%sql
SELECT "Booster_Version"
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (drone ship)'
AND "PAYLOAD_MASS__KG_" > 4000
AND "PAYLOAD_MASS__KG_" < 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 calculates the **total number of successful and failed mission outcomes** by grouping the data based on the **Landing_Outcome**. It counts the occurrences of each outcome (either 'Success' or 'Failure') using the COUNT(*) function.

```
%%sql
SELECT "Mission_Outcome", COUNT(*)
FROM SPACEXTABLE
GROUP BY "Mission_Outcome";
```

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

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

Boosters Carried Maximum Payload

This query retrieves the **booster version** that carried the **maximum payload mass** from the SPACEXTABLE. The subquery first finds the maximum value of PAYLOAD_MASS__KG_ using the MAX() function. The outer query then selects the Booster_Version where the payload mass matches this maximum value.

```
%%sql
SELECT "Booster_Version"
FROM SPACEXTABLE
WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE);
```

```
 * 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

2015 Launch Records

This query extracts the **month**, **landing outcome**, **booster version**, and **launch site** for all records where the **landing outcome** was "Failure (drone ship)" in the year **2015**. The SUBSTR() function is used to filter the **year 2015** from the "Date" field and also to extract the **month** from the date. The result shows two failed drone ship landings for **F9 v1.1** at the **CCAFS** sites during the months of **January** and **April** of 2015.

```
%%sql
SELECT SUBSTR("Date", 6, 2) AS Month, "Landing_Outcome", "Booster_Version", "Launch_Site"
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Failure (drone ship)'
AND SUBSTR("Date", 0, 5) = '2015';
```

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

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

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

This query ranks the landing outcomes between the date range of June 4, 2010, and March 20, 2017, in descending order of frequency. It groups the results by the "Landing_Outcome" field and counts the occurrences of each outcome. The results allow you to see which landing outcomes, such as 'Failure (drone ship)' or 'Success (ground pad)', occurred most frequently during this period.

```
%%sql
SELECT "Landing_Outcome", COUNT(*) AS OutcomeCount
FROM SPACEXTABLE
WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY OutcomeCount DESC;
```

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

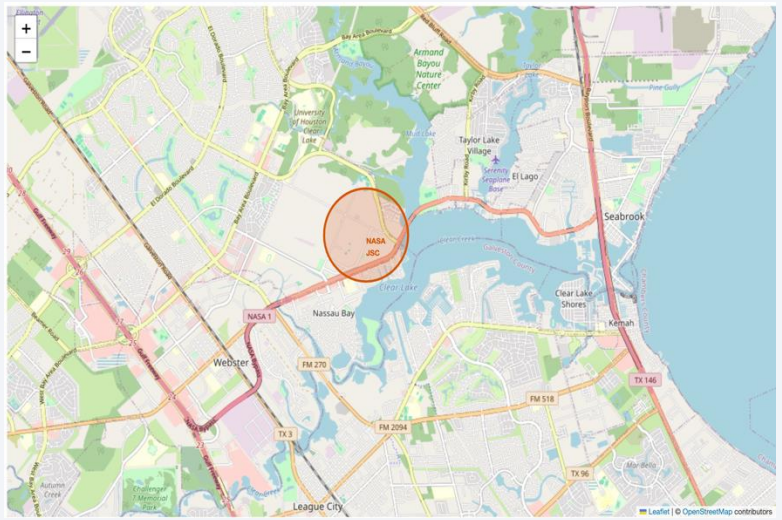
Landing_Outcome	OutcomeCount
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

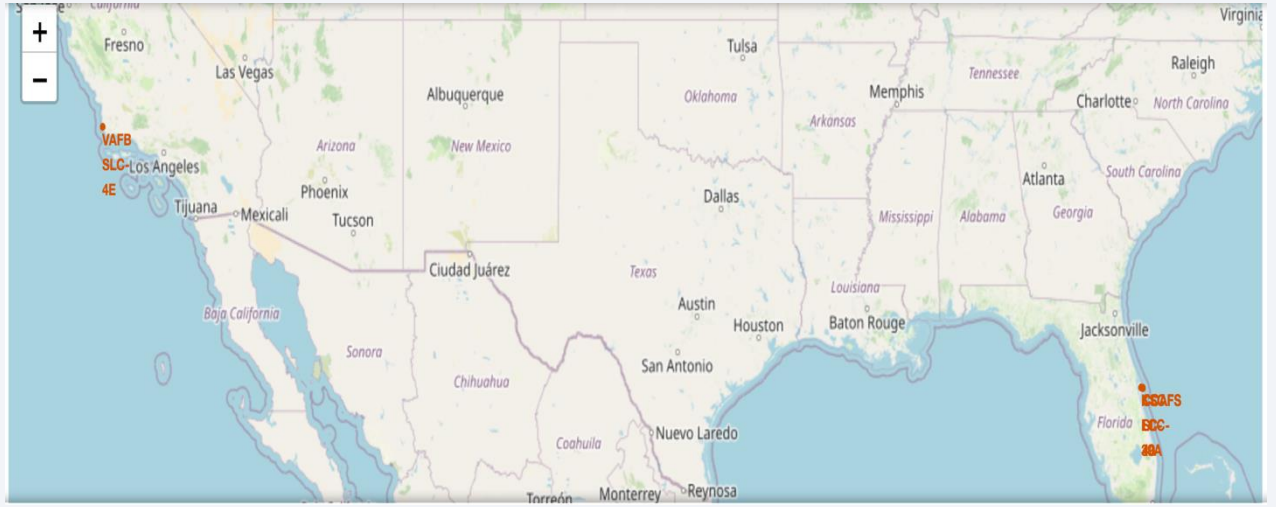
Section 3

Launch Sites Proximities Analysis

Mark all launch sites on a map

- Replace <Folium map screenshot 1> title with an appropriate title
- Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map
- Explain the important elements and findings on the screenshot

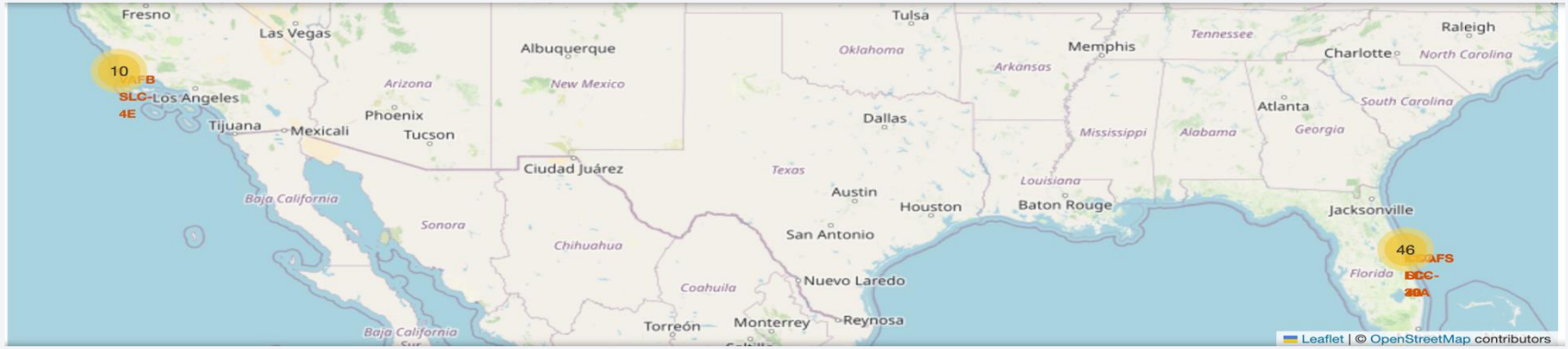


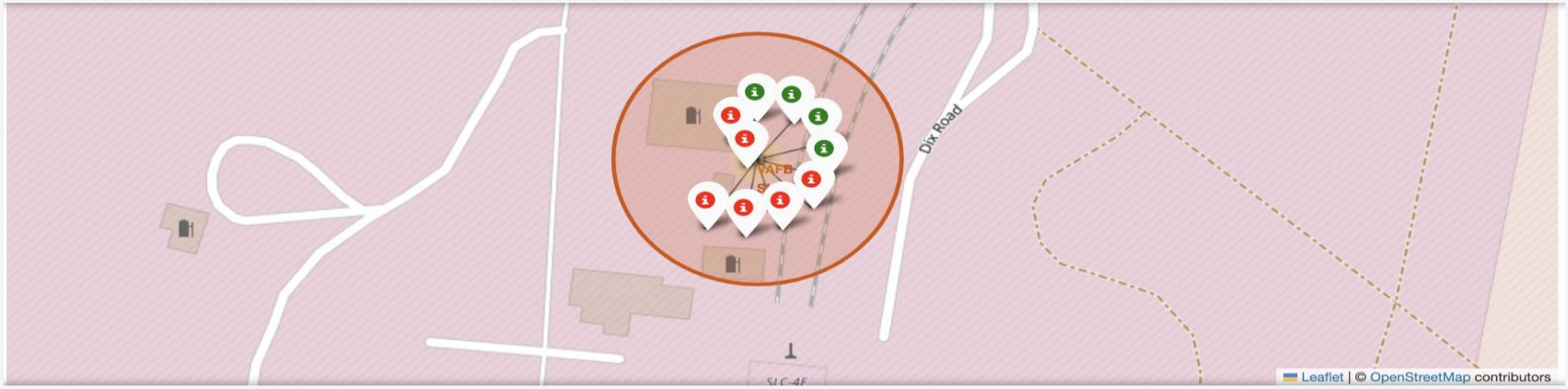


Mark the success/failed launches for each site on the map

- Replace <Folium map screenshot 2> title with an appropriate title

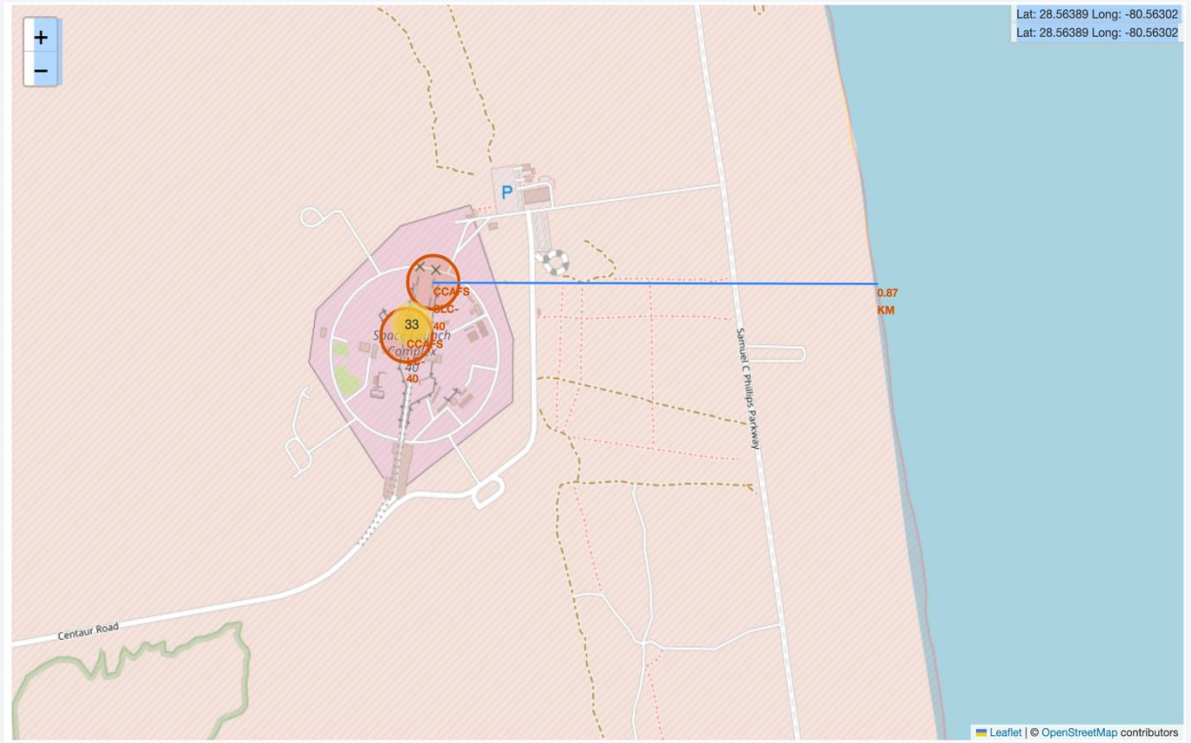
- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map

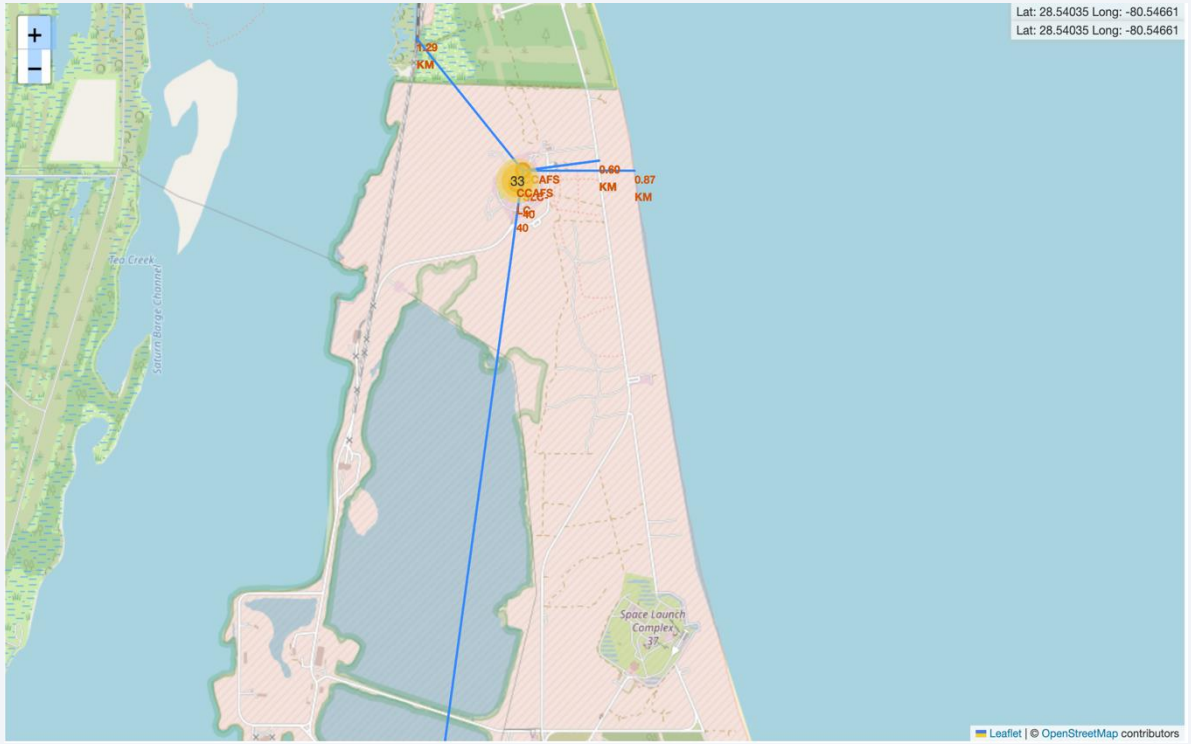


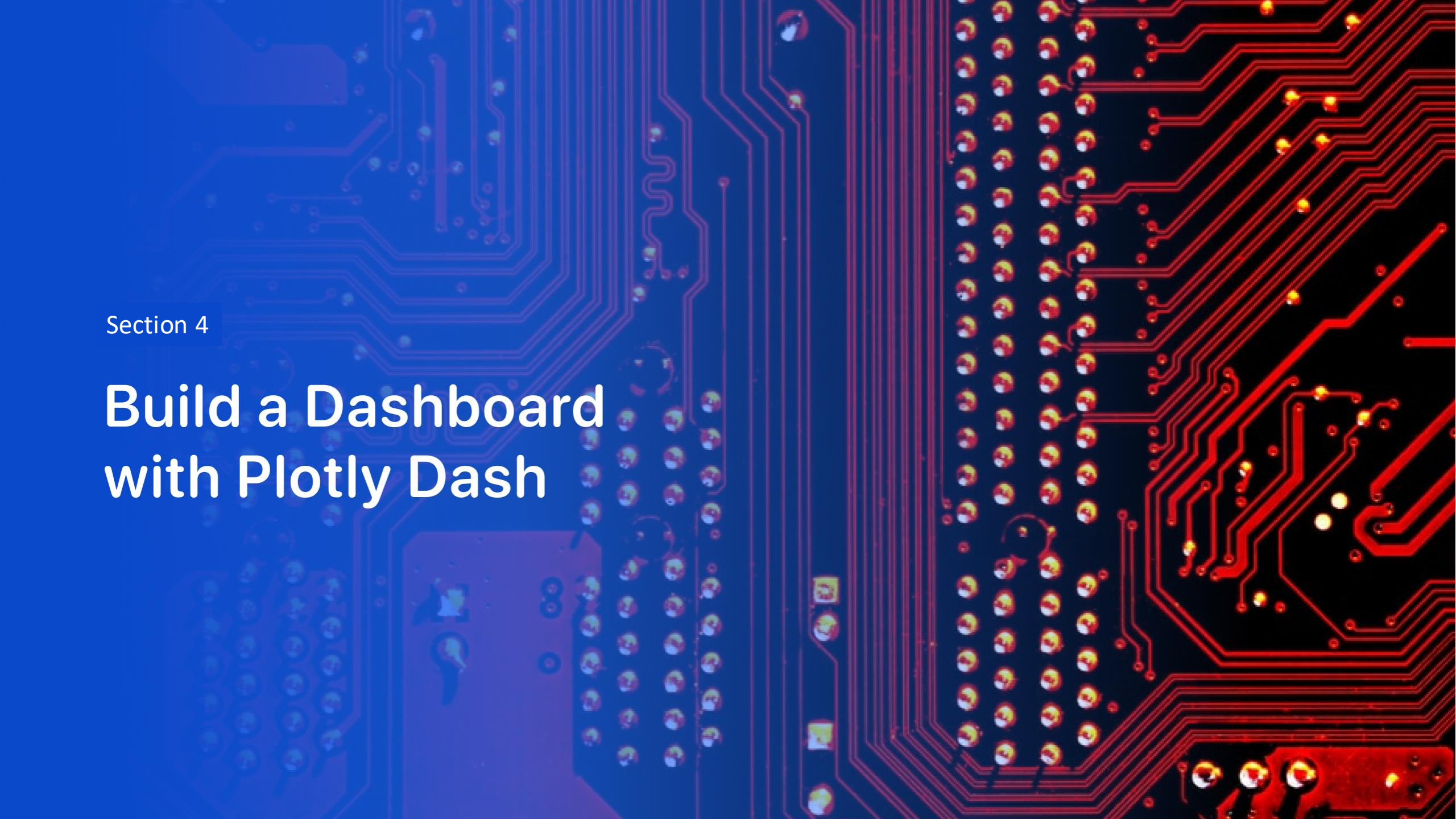


Calculate the distances between a launch site to its proximities

- Replace <Folium map screenshot 3> title with an appropriate title
- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed
- Explain the important elements and findings on the screenshot





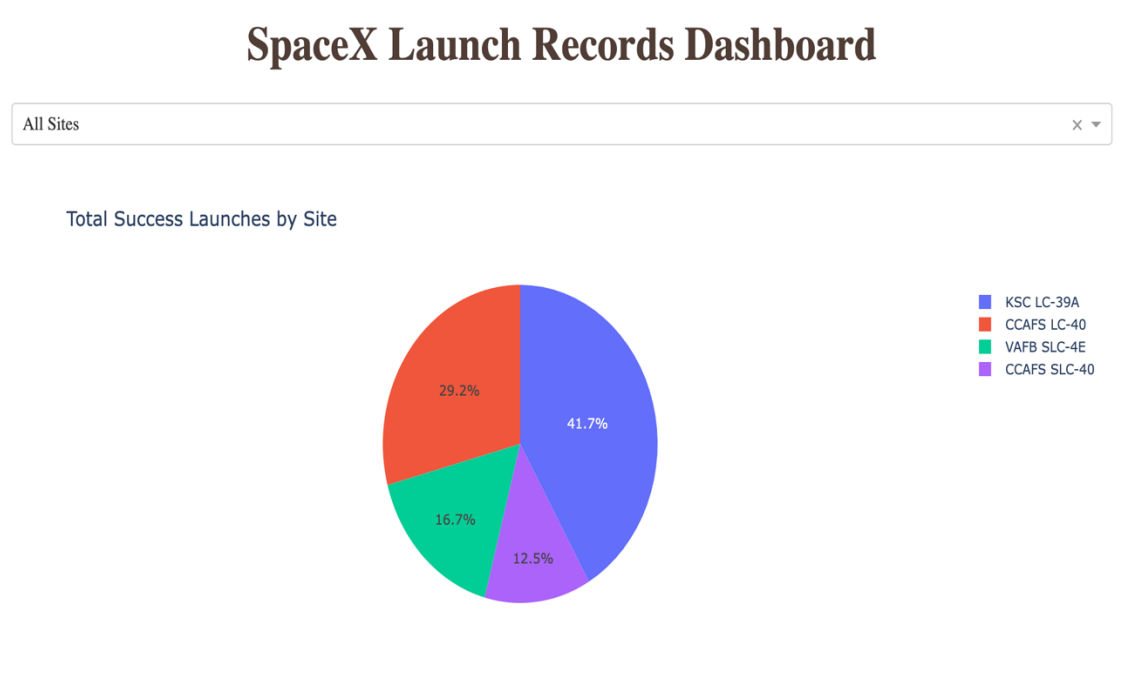

Section 4
Build a Dashboard with Plotly Dash

launch success count for all sites

Explanation of Important Elements and Findings:

- **Dropdown Menu (All Sites)**
  This dropdown allows users to filter the results by specific launch sites, or view data for all sites combined. In this screenshot, "All Sites" is selected, showing the total success launches across all SpaceX launch sites.
- **Pie Chart: Total Success Launches by Site**
  The pie chart visualizes the distribution of successful launches across four key SpaceX launch sites:
  - **KSC LC-39A**: Represented by the blue segment, this site accounts for 41.7% of total successful launches.
  - **CCAFS LC-40**: Represented by the red segment, this site accounts for 29.2% of total successful launches.
  - **VAFB SLC-4E**: Represented by the green segment, this site accounts for 16.7% of total successful launches.
  - **CCAFS SLC-40**: Represented by the purple segment, this site accounts for 12.5% of total successful launches.

Key Insight:
The majority of successful launches were conducted from the **KSC LC-39A** site, which represents over 40% of the total successful launches. Other significant sites include **CCAFS LC-40** and **VAFB SLC-4E**, but **KSC LC-39A** appears to be the most frequently successful launch site.



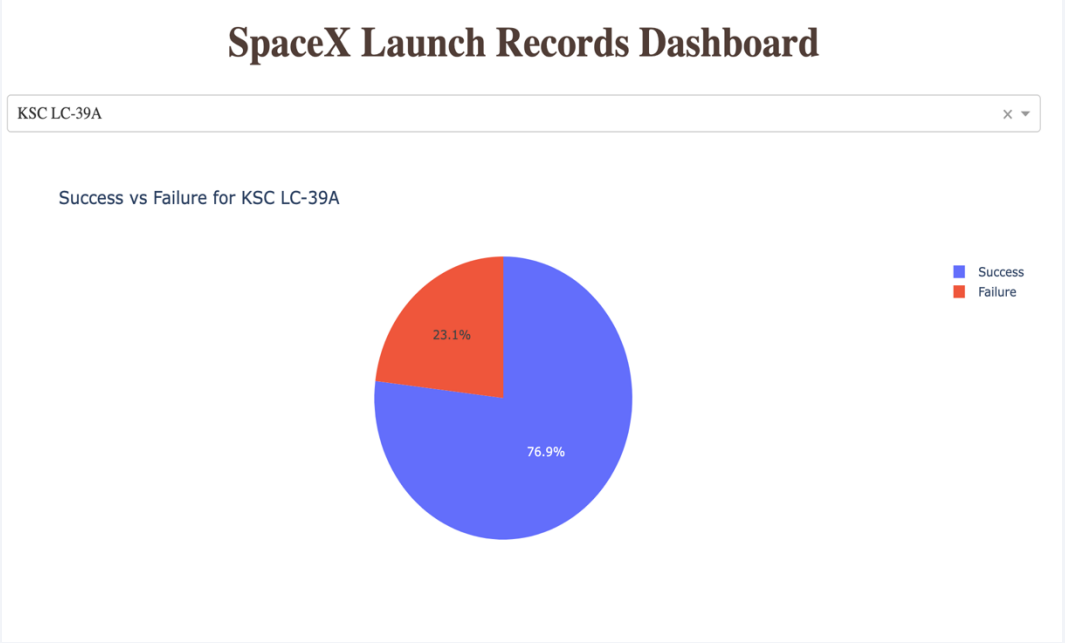
the launch site with highest launch success ratio

Explanation of Important Elements and Findings:

- **Dropdown Menu (KSC LC-39A)**
  The dropdown is set to "KSC LC-39A," which filters the results specifically to show data for launches from the **Kennedy Space Center Launch Complex 39A**.
- **Pie Chart: Success vs. Failure for KSC LC-39A**
  This pie chart visualizes the proportion of successful versus failed launches for **KSC LC-39A**. The chart uses two colors:
  - **Blue (Success)**: Represents 76.9% of launches from KSC LC-39A that were successful.
  - **Red (Failure)**: Represents 23.1% of launches from KSC LC-39A that failed.

Key Insight:

- The majority of launches from **KSC LC-39A** were successful, with nearly 77% success rate. Only around 23% of launches failed, indicating a high success rate at this launch site.



Payload vs. Launch Outcome scatter plot for all sites(1/4)

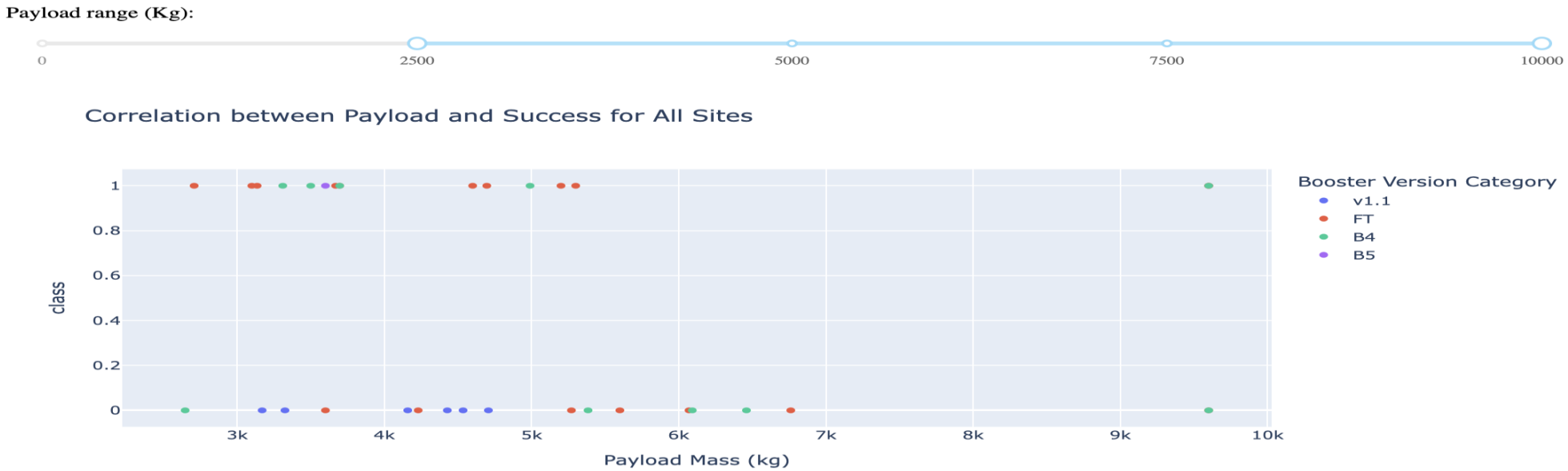
Low Payload Range (0 to 2500 kg):

- **Observation:** In the low payload range (0 to 2500 kg), a high success rate is seen across multiple booster versions such as v1.1 and FT.
- **Key Insight:** Smaller payloads appear to be well-handled by most booster versions, showing almost consistent success, particularly for versions FT and B4.

Payload vs. Launch Outcome scatter plot for all sites(2/4)

Mid Payload Range (2500 kg to 5000 kg):

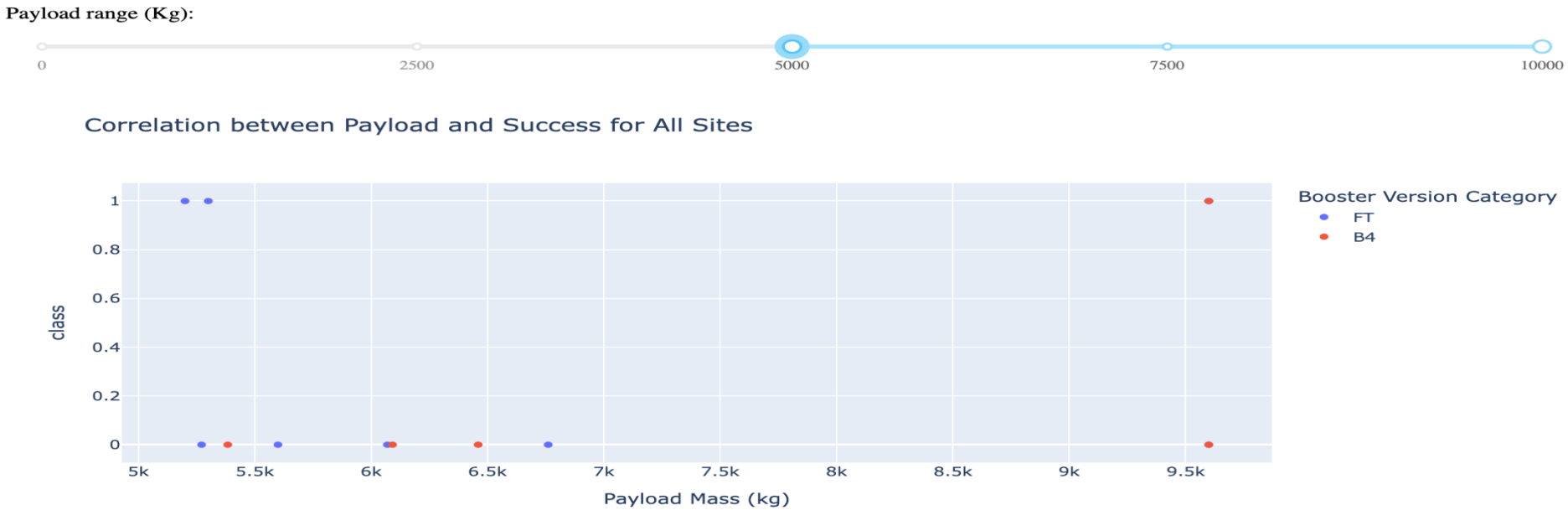
- **Observation:** For payloads between 2500 kg to 5000 kg, booster versions such as FT and B4 continue to show a high success rate. Some failures are still observed in this range, but the overall trend is that of high success.
- **Key Insight:** Payloads within this mid-range demonstrate reliable performance for newer booster versions (e.g., B4), indicating technological improvements for mid-weight payloads.



Payload vs. Launch Outcome scatter plot for all sites(3/4)

Higher Payload Range (5000 kg to 7500 kg):

- **Observation:** As the payload weight increases, fewer data points are present. The success rate is still high for certain booster versions, such as FT and B4, though failures are seen for heavier payloads.
- **Key Insight:** As payload mass increases, the success rate still remains robust for modern booster versions, particularly for payloads handled by FT and B4 versions. However, there are some failures, indicating that very high payloads may introduce challenges.



Payload vs. Launch Outcome scatter plot for all sites(4/4)

Very High Payload Range (7500 kg to 10,000 kg):

- **Observation:** In the highest payload range, we see mostly newer booster versions such as B4 and B5 handling the launches, with a high success rate. The data points for this range are fewer, but it's clear that the larger payloads are successfully managed by the latest technologies.
- **Key Insight:** The highest payloads are exclusively handled by newer booster versions like B4 and B5, showing the technological advancements in handling larger payloads with high success.

Section 5

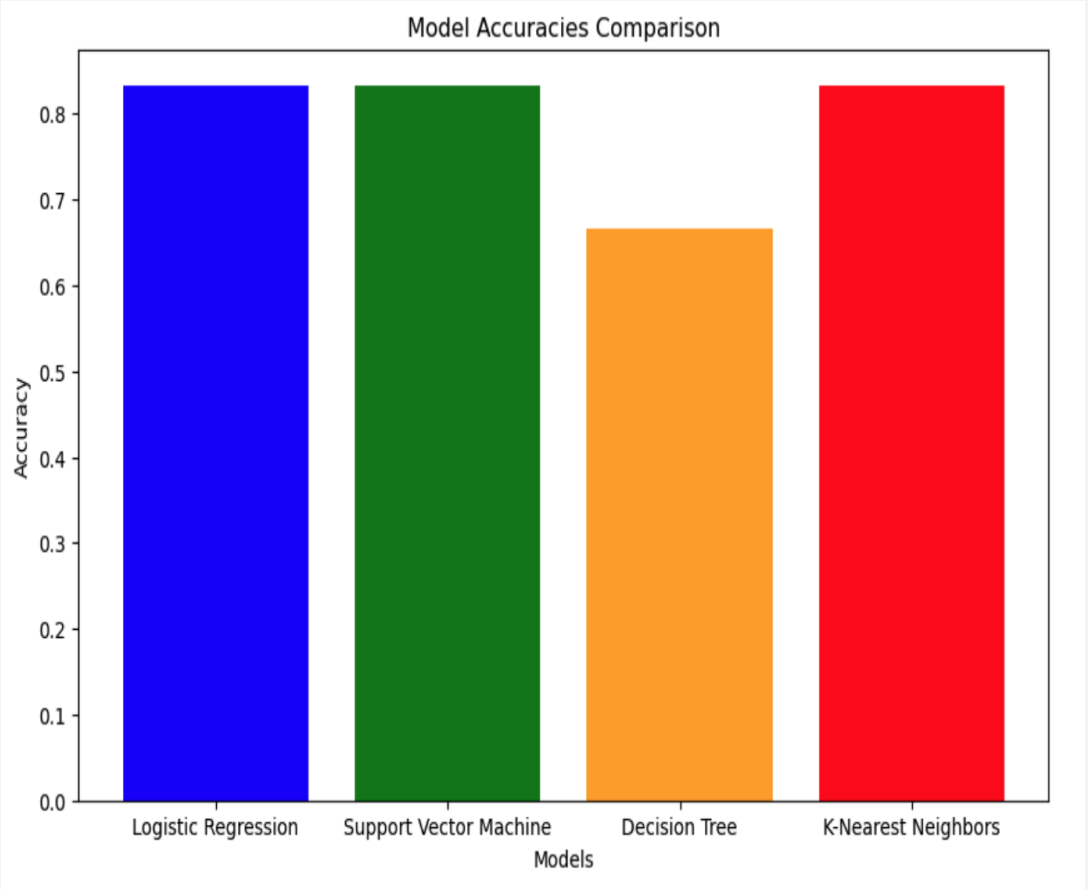
Predictive Analysis (Classification)

Classification Accuracy

The classification accuracy comparison shows the performance of different models: Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbors. Based on the graph:

- **Logistic Regression**, **Support Vector Machine**, and **K-Nearest Neighbors** all have the same highest accuracy at approximately 0.83.
- The **Decision Tree** model has the lowest accuracy at approximately 0.67.

The explanation here is that Logistic Regression, Support Vector Machine, and K-Nearest Neighbors all performed similarly, and any of these models can be considered the best in terms of accuracy for this particular dataset. The Decision Tree model underperformed compared to the other three models.

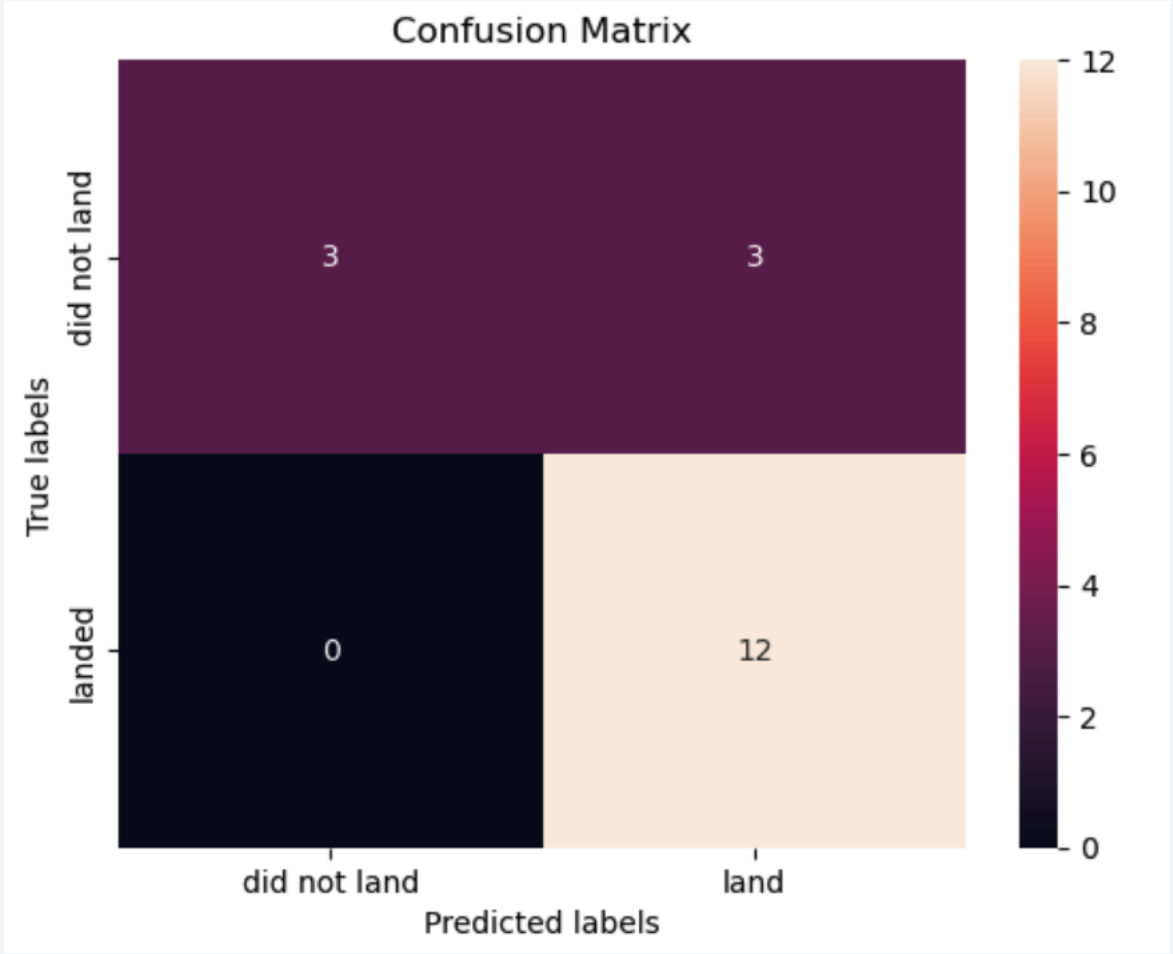


Confusion Matrix

- **True Positives (TP)**: These are cases where the true label is "landed" and the model also predicted "landed" correctly. In your case, there are 12 true positives. This means the model made 12 correct predictions where a successful landing was correctly identified.
- **False Positives (FP)**: These are cases where the true label is "not landed," but the model incorrectly predicted "landed." In your case, there are 3 false positives. This indicates that the model predicted a successful landing when, in reality, it was not successful.

Insights:

- The issue with **False Positives** could be problematic in situations where predicting a successful landing is critical (e.g., financial implications or safety measures).
- The confusion matrix shows that logistic regression performs reasonably well in identifying true positives but still has a small number of false positives.



Conclusions

- **Logistic Regression**, **Support Vector Machine**, and **K-Nearest Neighbors** models performed equally well, each achieving an accuracy of **83.33%**.
- **Decision Tree** had the lowest performance, with an accuracy of **66.67%**.
- The **best performing models** are **Logistic Regression**, **Support Vector Machine**, and **K-Nearest Neighbors**, all with identical accuracy.
- The **Decision Tree** model underperformed compared to the other models, indicating it might not be the best choice for this particular problem without further tuning or feature engineering.
- **Further steps** could involve deeper hyperparameter tuning or exploring ensemble methods to improve the accuracy across models or find a more distinct leader.

Appendix

contents	Sources
Data Collection – SpaceX API	https://github.com/juyeon444/data-science-capstone/blob/main/Module 1 - 1. Lab 1 Collecting the data - jupyter-labs-spacex-data-collection-api.ipynb
Data Collection – SpaceX API	https://github.com/juyeon444/data-science-capstone/blob/main/Module 1 - 1. Lab 1 Collecting the data - dataset_part_1.csv
Data Collection – SpaceX API	https://juyeon444.github.io/data-science-capstone/Module 1 - 1. Lab 1 Collecting the data - jupyter-labs-spacex-data-collection-api.html
Data Collection - Scraping	https://juyeon444.github.io/data-science-capstone/Module 1 - 2. Space X Falcon 9 First Stage Landing Prediction - jupyter-labs-webscraping.html
Data Collection - Scraping	https://github.com/juyeon444/data-science-capstone/blob/main/Module 1 - 2. Space X Falcon 9 First Stage Landing Prediction - jupyter-labs-webscraping.ipynb
Data Collection - Scraping	https://github.com/juyeon444/data-science-capstone/blob/main/Module 1 - 2. Space X Falcon 9 First Stage Landing Prediction - spacex_web_scraped.csv
Data Wrangling	https://github.com/juyeon444/data-science-capstone/blob/main/Module 1 - 3. Lab 2 Data wrangling - dataset_part_2.csv
Data Wrangling	https://juyeon444.github.io/data-science-capstone/Module 1 - 3. Lab 2 Data wrangling - labs-jupyter-spacex-Data wrangling.html
Data Wrangling	https://github.com/juyeon444/data-science-capstone/blob/main/Module 1 - 3. Lab 2 Data wrangling - labs-jupyter-spacex-Data wrangling.ipynb
EDA with SQL	https://juyeon444.github.io/data-science-capstone/Module 2 - 1. Assignment SQL Notebook for Peer Assignment - jupyter-labs-eda-sql-coursera_sqllite.html
EDA with SQL	https://github.com/juyeon444/data-science-capstone/blob/main/Module 2 - 1. Assignment SQL Notebook for Peer Assignment - jupyter-labs-eda-sql-coursera_sqllite.ipynb

Appendix

Contents	Sources
EDA with Data Visualization	https://github.com/juyeon444/data-science-capstone/blob/main/Module 2 - 2. SpaceX Falcon 9 First Stage Landing Prediction - dataset_part_3.csv
EDA with Data Visualization	https://juyeon444.github.io/data-science-capstone/Module 2 - 2. SpaceX Falcon 9 First Stage Landing Prediction - edadataviz.html
EDA with Data Visualization	https://github.com/juyeon444/data-science-capstone/blob/main/Module 2 - 2. SpaceX Falcon 9 First Stage Landing Prediction - edadataviz.ipynb
Build an Interactive Map with Folium	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 1. Launch Sites Locations Analysis with Folium - lab_jupyter_launch_site_location.ipynb
Build an Interactive Map with Folium	https://juyeon444.github.io/data-science-capstone/Module 3 - 1. Launch Sites Locations Analysis with Folium - lab_jupyter_launch_site_location.html
Build a Dashboard with Plotly Dash	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 2. spacex_dash_app.py
Build a Dashboard with Plotly Dash	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 2. All Sites.png
Build a Dashboard with Plotly Dash	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 2. CCAFS LC-40.png
Build a Dashboard with Plotly Dash	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 2. CCAFS SLC-40.png
Build a Dashboard with Plotly Dash	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 2. KSC LC-39A.png
Build a Dashboard with Plotly Dash	https://github.com/juyeon444/data-science-capstone/blob/main/Module 3 - 2. VAFB SLC-4E.png
Predictive Analysis (Classification)	https://juyeon444.github.io/data-science-capstone/Module 4 - 1. SpaceX_Machine Learning Prediction_Part_5.html
Predictive Analysis (Classification)	https://github.com/juyeon444/data-science-capstone/blob/main/Module 4 - 1. SpaceX_Machine Learning Prediction_Part_5.ipynb

Thank you!