IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Tammy Wu>
<Jan 11, 2025>

Outline

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

Executive Summary

Summary of methodologies

This project follows these steps :

- Data Collection
- Data Wrangling
- Exploratory Data Analysis using SQL
- Exploratory Data Analysis for Data Visualization
- Interactive Visual Analytics with Folium Lab and Dashboards with Ploty Dash
- Predictive Analysis
- Summary of all results

This project produced the following outputs and visualizations

- **Exploratory Data Analysis (EDA)**
- **Interactive Analytics via Dashboard**
- **Predictive Analytics and Model Performance**
- **Key Insights and Visualizations**

Introduction

- **Project background and context**

- The goal of this project is to analyse SpaceX's Falcon 9 and Falcon Heavy launch records to determine factors influencing the success of first-stage landings. SpaceX aims to achieve full rocket reusability to significantly reduce launch costs. Predicting whether a rocket's first stage will successfully land is crucial for enhancing mission reliability and optimizing resource allocation.

- This project involves:

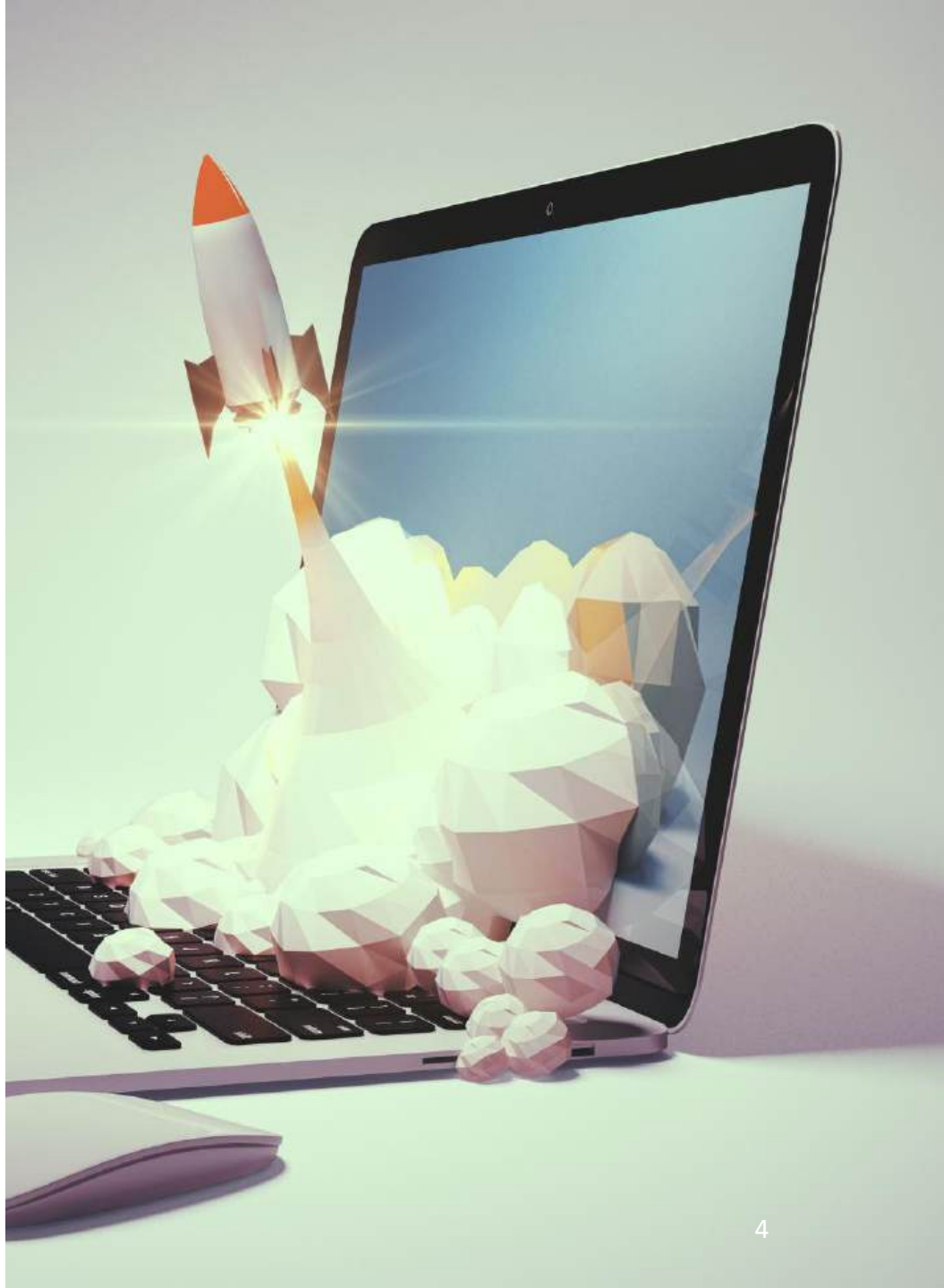
- **Data Collection and Exploration**: Gathering historical launch data from sources such as Wikipedia and conducting exploratory analysis to uncover patterns.

- **Interactive Dashboards**: Developing interactive visualizations to allow dynamic exploration of launch outcomes.

- **Predictive Modelling**: Applying machine learning techniques to build models that predict landing success based on factors like orbit type, payload mass, and booster version.

- **Problems you want to find answers**

- **What factors contribute to the success of Falcon 9 first-stage landings?**
- **Which launch site has the highest success rate?**
- **How does payload mass affect the likelihood of a successful landing?**
- **Which machine learning model performs best in predicting landing success?**
- **What is the correlation between orbit type and landing success?**



Section 1

Methodology

Methodology

Executive Summary

❖ **Data Collection Methodology**

- Data was gathered from Wikipedia using web scraping techniques (BeautifulSoup).
- Supplementary details were obtained via the SpaceX API.

❖ **Data Wrangling**

- Cleaning and formatting the dataset to ensure consistency and accuracy.
- Applied one-hot encoding to categorical features like Orbit, LaunchSite, and LandingPad.

❖ **Exploratory Data Analysis (EDA)**

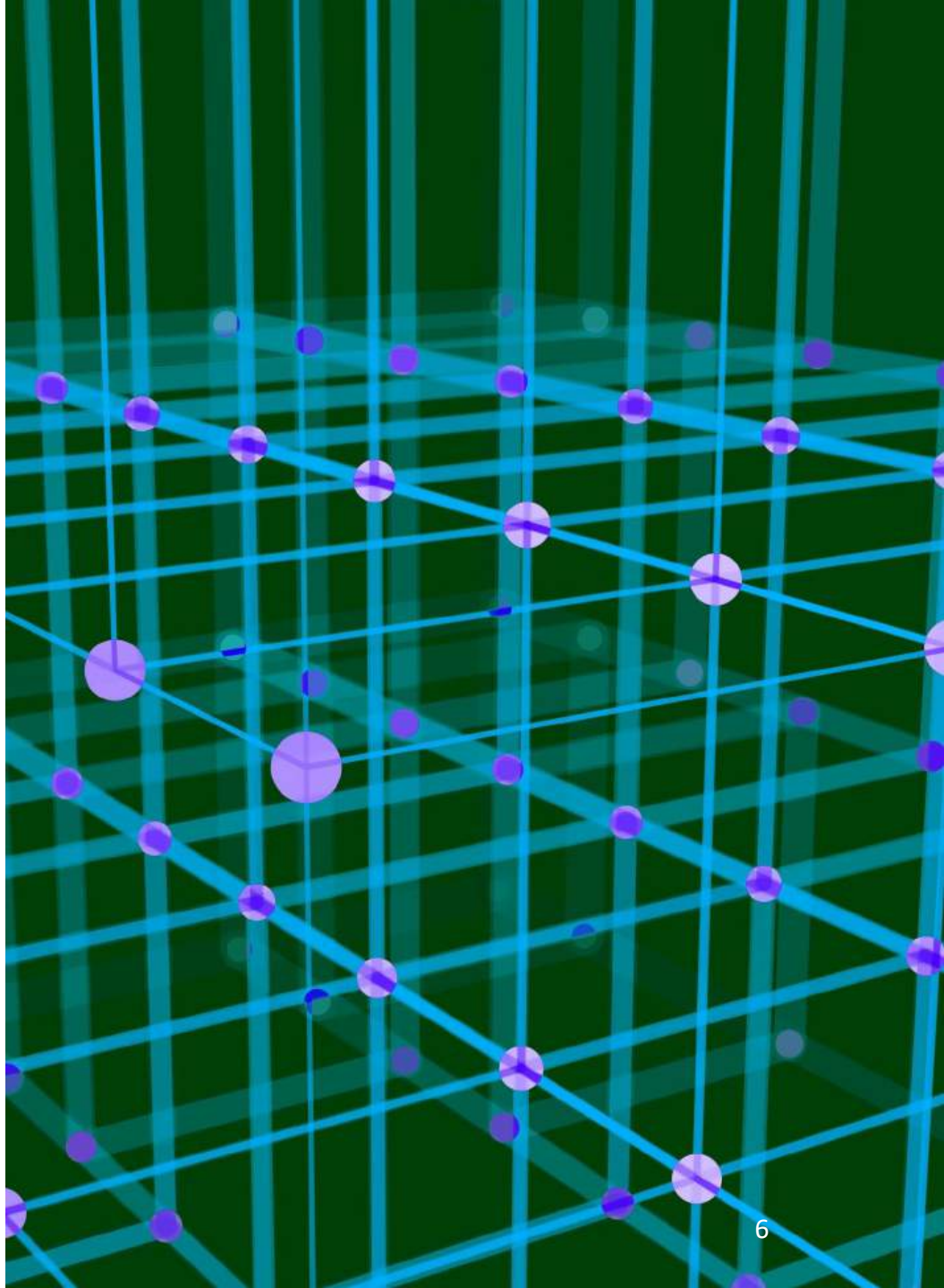
- Conducted visualizations using bar charts, scatter plots, and SQL queries.
- Identified key trends and correlations, such as the impact of Payload Mass on landing success.

❖ **Interactive Visual Analytics**

- Created an interactive dashboard using Plotly Dash and Folium to dynamically explore data.

❖ **Predictive Analytics**

- Built and tuned classification models, including Logistic Regression, SVM, Decision Tree, and KNN.
- **SVM** was the best-performing model with a test accuracy of 86%



Data Collection

Describe how data sets were collected.

Web Scraping:

Launch data for Falcon 9 and Falcon Heavy rockets was collected from Wikipedia using Python's BeautifulSoup library.

HTML tables containing mission details, launch dates, payloads, orbits, and landing outcomes were parsed and extracted.

SpaceX API:

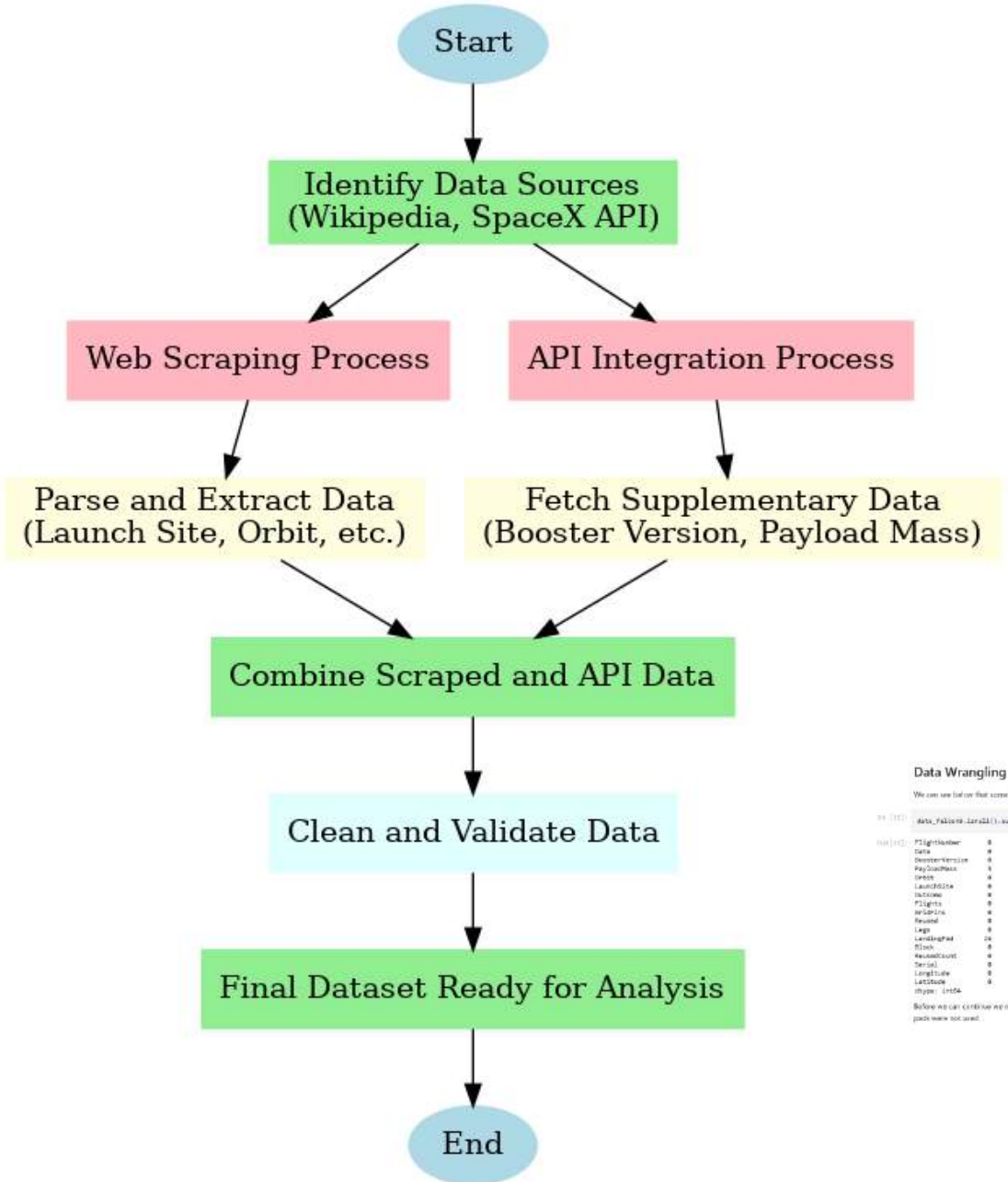
Additional information about launches, such as booster version and mission details, was obtained via the SpaceX API.

Data Consolidation:

Data from multiple sources was merged to form a comprehensive dataset for analysis.

Validation:

The collected data was cross-referenced with official SpaceX reports to ensure accuracy and reliability.



TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
# use requests.get() method with the provided static_url
response = requests.get(static_url)
response
```

```
<Response [200]>
```

Create a `BeautifulSoup` object from the HTML `response`

```
soup = BeautifulSoup(response.content, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
print(soup.title.string)
```

```
List of Falcon 9 and Falcon Heavy launches - Wikipedia
```

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

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

Check the content of the response

```
print(response.content)
```

we can apply the rest of the functions here:

```
# Call getLaunchSite
getLaunchSite(data)
```

```
# Call getPayloadData
getPayloadData(data)
```

```
# Call getCoreData
getCoreData(data)
```

Finally let's construct our dataset using the data we have obtained. We we combine the columns into a dictionary.

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

Data Wrangling

We can see below that some of the rows are missing values in our dataset.

```
data_falcon9.isnull().sum()
```

```
FlightNumber    0
Date            0
BoosterVersion  0
PayloadMass     5
Orbit           0
LaunchSite      0
Outcome         0
Flights         0
GridFins        0
Reused          0
Legs            0
LandingPad     26
Block           0
ReusedCount     0
Serial          0
Longitude       0
Latitude        0
dtype: int64
```

Before we can continue we must deal with these missing values. The LandingPad column will retain None values to represent when landing pads were not used.

Task 3: Dealing with Missing Values

Calculate below the mean for the PayloadMass using the .mean(). Then use the mean and the .replace() function to replace np.nan values in the data with the mean you calculated.

```
# Calculate the mean value of PayloadMass column
payload_mass_mean = data_falcon9['PayloadMass'].mean()
# Replace the np.nan values with its mean value
data_falcon9.loc[:, 'PayloadMass'] = data_falcon9['PayloadMass'].fillna(payload_mass_mean)
```

Data Collection – SpaceX API

github-API

Initiate REST API Call:
Send a GET request to SpaceX API

Receive JSON Response:

Parse the JSON data from the response.

Normalize Data:
Use pandas.join_normalize() to flatten nested fields.

```
# Use json_normalize meethod to convert the json result into a dataframe
response.json()
data=pd.json_normalize(response.json())
```

Filter and Clean Data:
Extract relevant fields and handle missing values

```
# Calculate the mean value of PayloadMass column
payload_mass_mean = data_falcon9['PayloadMass'].mean()
# Replace the np.nan values with its mean value
data_falcon9.loc[:, 'PayloadMass'] = data_falcon9['PayloadMass'].fillna(payload_mass_mean)
```

```
print(data_falcon9)
```

Store as DataFrame:
Save the results as a csv file or excel for analysis

```
# Export to Excel
df = pd.DataFrame(launch_dict)
df.to_excel('spacex_launch_data.xlsx', index=False)
```

Data Collection - Scraping

Github - Scraping

HTTP Request and HTML Parsing

Fetch raw HTML from the URL **and** Use BeautifulSoup to parse and extract meaningful data.

Extract all column/variable names

Retrieve Column headers using the table <th> elements to extract all variable names for the Data Frame.
Look through the rows of the table (<tr>) and extract data from the cell (<td>).

Create a DataFrame by parsing the launch HTML tables:

Structure the extracted data into a Pandas DataFrame using the variable names.

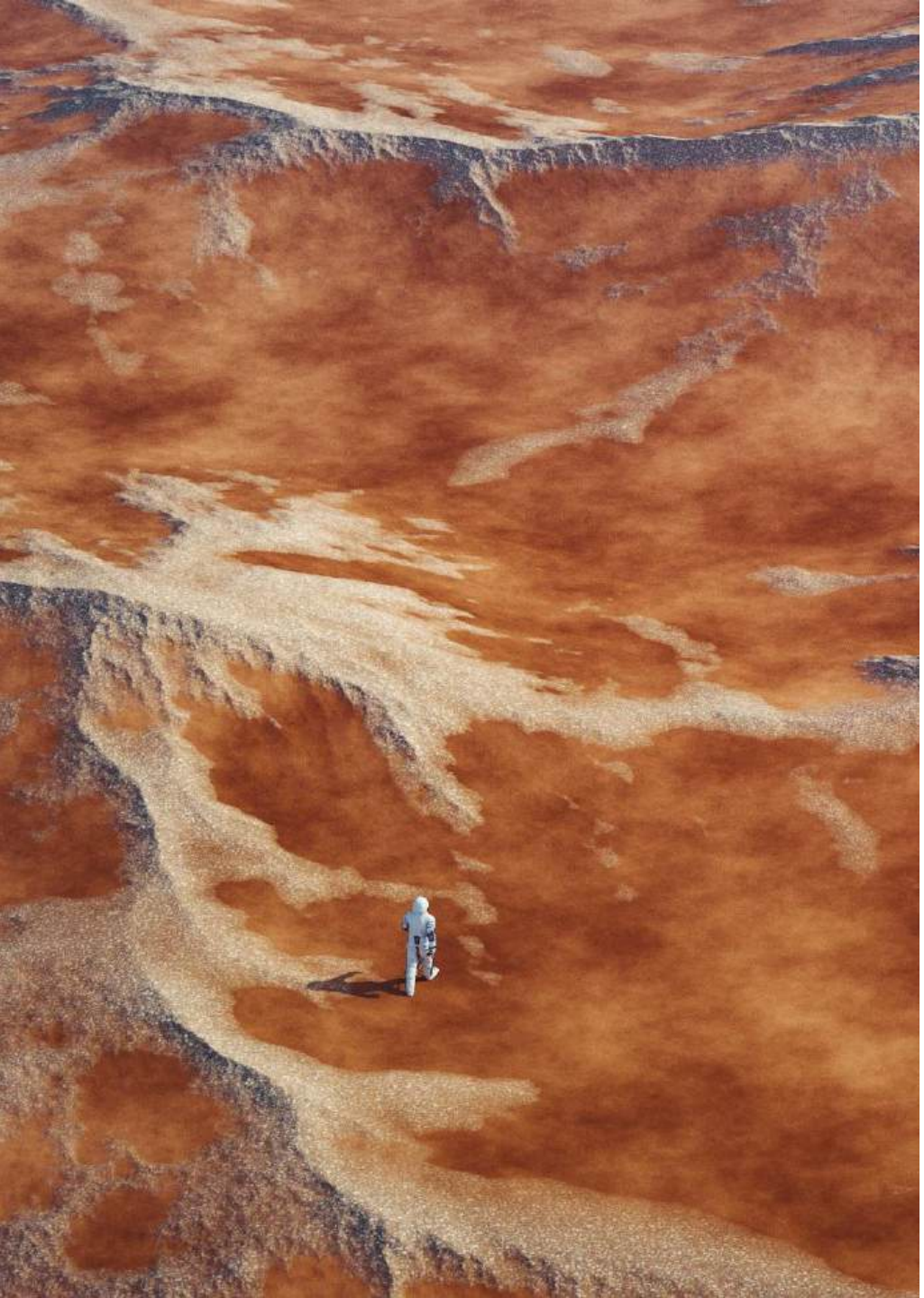
Data Cleaning and Structuring

Using booster_version() for directly from cell, adding to to launch_dict

Storage

Export the DataFrame to csv for Excel format for futher analysis

```
df.to_csv('spacex_web_scraped.csv', index=False)
```



Data Wrangling

In the dataset, various scenarios describe unsuccessful booster landings, For example

- **True Ocean**: Successful landing in a specific ocean region.
- **False Ocean**: Unsuccessful landing attempt in a specific ocean region.
- **True RTLS**: Successful landing on a ground pad.
- **False RTLS**: Failed landing on a ground pad.
- **True ASDS**: Successful landing on a drone ship.
- **False ASDS**: Failed landing on a drone ship.

For training purposes, these outcomes are converted into labels:
•1: Successful booster landing.
•0: Unsuccessful booster landing.

Data Wrangling Process Key Phrases for Flowchart

1. Load Dataset
Import the dataset using pandas.read_csv().
Inspect data using df.head() and df.info().

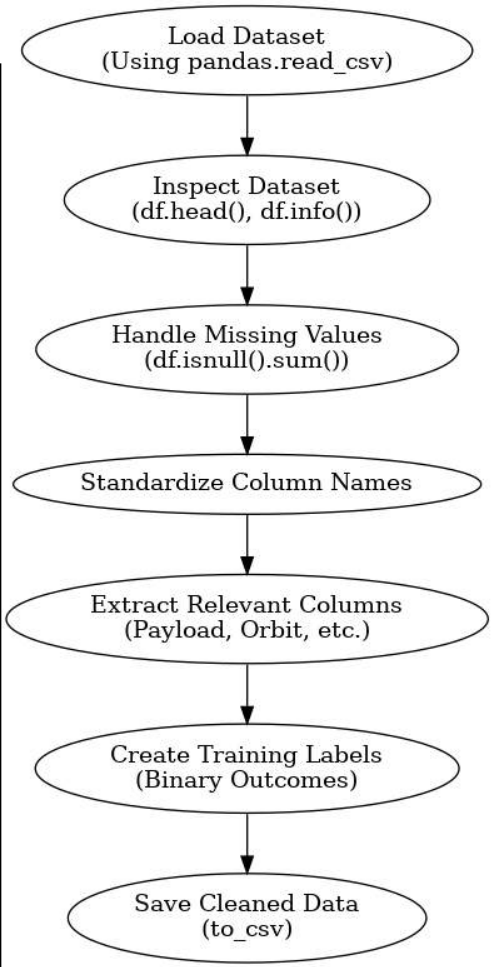
2. Handle Missing Values
Identify missing data using df.isnull().sum().
Fill or drop missing values as necessary.

3. Standardize Column Names
Ensure consistent and meaningful column names.

4. Extract Relevant Columns
Select key features such as Payload Mass, Orbit, Booster Version, and Launch Outcome.

5. Create Training Labels
Convert mission outcomes (e.g., True/False ASDS, Ocean, RTLS) into binary labels (1 for success, 0 for failure).

6. Save Cleaned Data
Export the processed dataset as a CSV file using to_csv() for further analysis.



Github - Data Wrangling

EDA with Data Visualization

Bar Chart: Success Rate by Orbit Type	**Purpose:** To visualize and compare the success rates of launches across different orbit types.	This helps identify which orbit type has the highest probability of a successful launch.
Scatter Plot: Correlation Between Payload and Launch Outcome	**Purpose:** To explore the relationship between payload mass and the success of launches.	This chart helps in identifying trends or limitations in payload capacity for successful missions.
Pie Chart: Distribution of Launch Success Across Sites	**Purpose:** To provide an overview of the percentage of successful launches per launch site.	This helps identify the best-performing launch sites.
Line Plot: Launch Success Over Time	**Purpose:** To track changes in success rates over the years.	This visualization highlights trends or improvements in launch success over time.

EDA with SQL

Github - EDA with SQL

Summary of SQL Queries Performed:

Query 1: Retrieve Unique Launch Sites	Extracted distinct launch sites from the dataset to analyze their distribution and performance.
Query 2: Count Total Launches Per Site	Counted the number of launches at each site to determine their frequency and importance.
Query 3: Calculate Success Rate by Site	Computed the success rate for each launch site to identify which site had the highest performance.
Query 4: Filter Launches Based on Payload Mass	Retrieved launches within a specific payload mass range to explore their outcomes and correlations.
Query 5: Group Launches by Orbit Type	Analyzed the number of launches and their success rates for different orbit types.
Query 6: Yearly Success Analysis	Counted successful launches by year to identify trends over time.

Build an Interactive Map with Folium

- **Markers**: Added to mark key launch sites, providing geographical information for each location.

- **Circles**: Represented regions of interest or launch areas with a specific radius, highlighting zones around launch sites.

- **Lines/Polylines**: Used to connect launch sites with landing zones or to illustrate trajectories visually.

Github - Folium

Build a Dashboard with Plotly Dash

Summary of Plots/Graphs and Interactions in the Dashboard

1. Dropdown for Launch Site Selection

- Added a dropdown menu to select a specific launch site or view data for all sites.
- **Reason**: Allows users to filter and explore SpaceX launch data by site.

2. Pie Chart: Success Rates by Launch Site

- Displays either total successful launches across all sites or success vs. failure for a specific site.
- **Reason**: Visualizes success distribution, helping identify the most reliable launch sites.

3. Range Slider: Payload Mass Selection

- Enables users to select a payload range dynamically.
- **Reason**: Allows exploration of the relationship between payload mass and launch success.

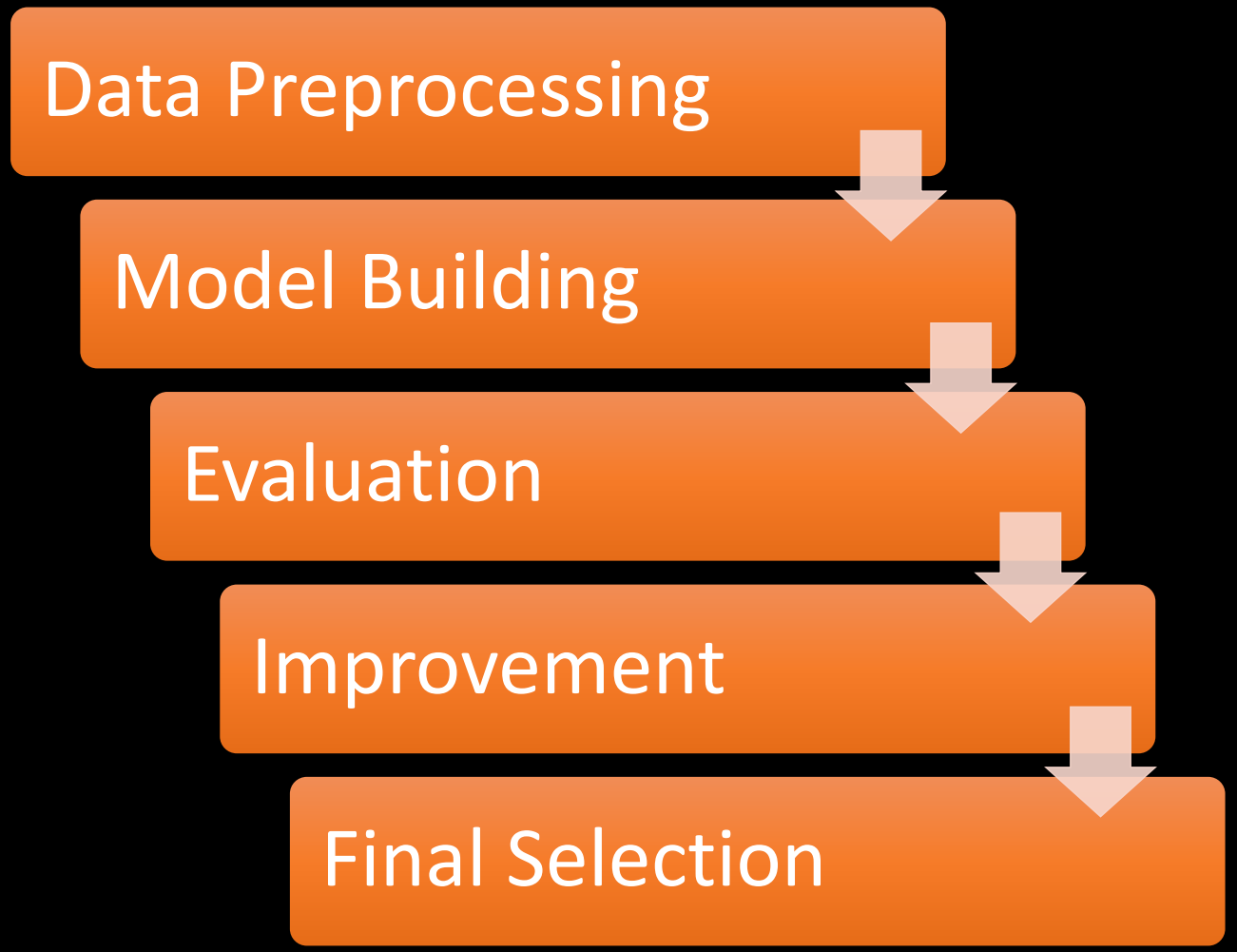
4. Scatter Plot: Payload vs. Success

- Illustrates the correlation between payload mass and launch success for selected sites and payload ranges.
- **Reason**: Provides insight into how payload size impacts success rates, helping identify patterns.

Github - Dashboard with Ploty Dash

PREDICTIVE ANALYSIS (CLASSIFICATION)

Github - Predictive Analysis



The notebook contains extensive details about model development, including data preprocessing, model selection, training, and evaluation. Based on the extracted content, I will outline the key phrases and create a flowchart summarizing the process. Here's the breakdown:

1.Data Preprocessing:

- Importing necessary libraries (Pandas, NumPy, Matplotlib, Seaborn).
- Standardizing data using sklearn.preprocessing.
- Splitting data into training and testing sets using train_test_split.

2.Model Building:

- Selection of algorithms (e.g., Logistic Regression, Decision Tree, etc.).
- Training models on the training dataset.

3.Evaluation:

- Metrics like accuracy, precision, recall, and F1-score.
- Using cross-validation to assess model performance.

4.Improvement:

- Hyperparameter tuning using GridSearchCV.
- Feature selection to enhance model performance.

5.Final Selection:

- Comparing models and selecting the best-performing one.

RESULTS

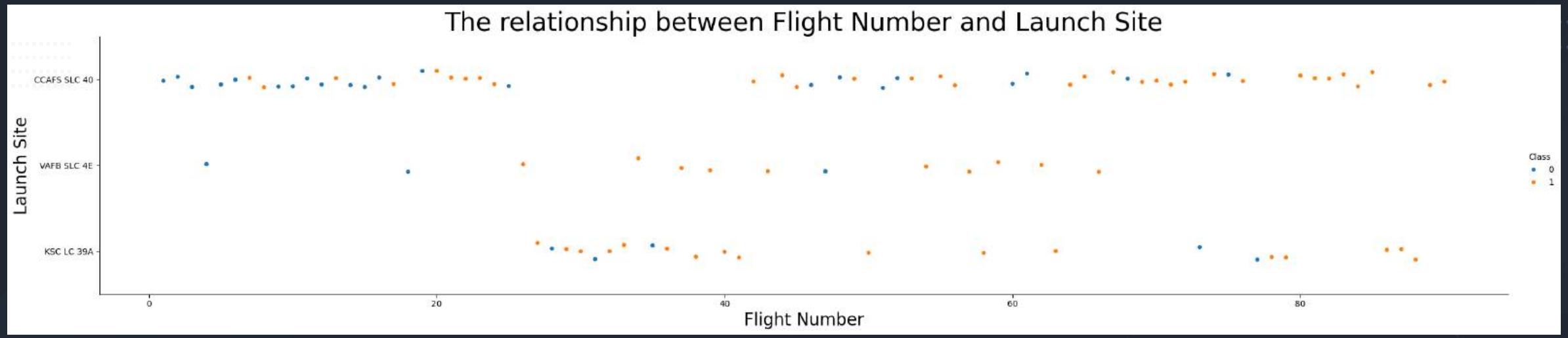
Exploratory data analysis results

Interactive analytics demo in screenshots

Predictive analysis results

Section 2

Insights drawn from EDA

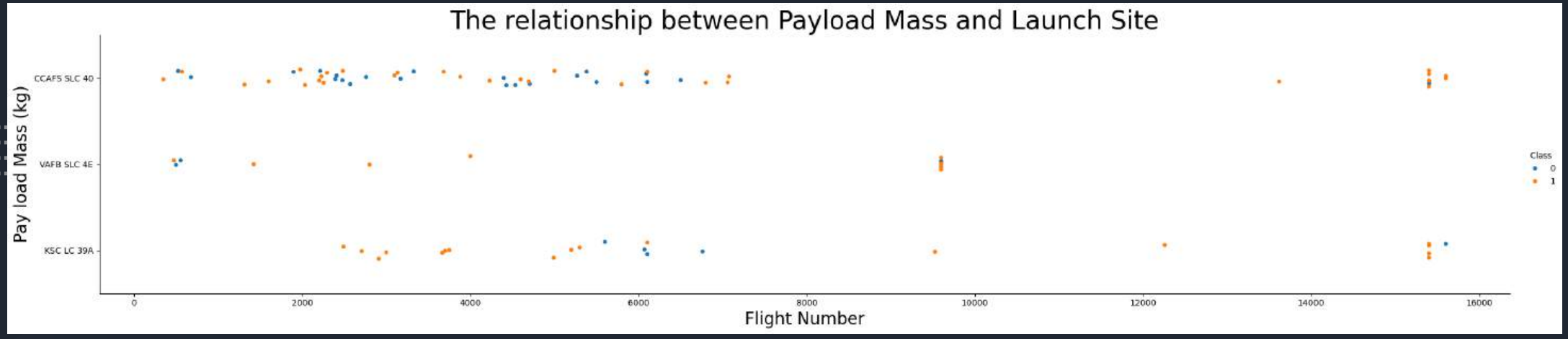


FLIGHT NUMBER VS. LAUNCH SITE

Key Observations:

- **X-axis (Flight Number)**: Represents the sequential numbering of SpaceX flight missions.
- **Y-axis (Launch Site)**: Specifies the launch sites, including:
  - CCAFS SLC 40
  - VAFB SLC 4E
  - KSC LC 39A
- **Data Points (Classes)**:
  - Blue dots represent Class 0 (failed missions).
  - Orange dots represent Class 1 (successful missions).
- **Insights**:
  - **CCAFS SLC 40** and **KSC LC 39A** show a mix of successful and failed missions, with increasing success rates as flight numbers increase.
  - **VAFB SLC 4E** has relatively fewer flights, predominantly failures.

This visual suggests a trend of improved success rates with higher flight numbers, possibly indicating advancements in technology or better operational practices over time.



Payload vs. Launch Site

Key Observations:

1. X-axis (Flight Number):
- Represents the sequence of missions, showing the progression of launches over time.

2. Y-axis (Payload Mass in kg):
- Indicates the payload weight carried during the missions, ranging from small to very heavy payloads.

3. Launch Sites:
- Missions are grouped by the launch sites:
  - CCAFS SLC 40
  - VAFB SLC 4E
  - KSC LC 39A

4. Data Points:
- Blue dots (Class 0): Represent failed missions.
- Orange dots (Class 1): Represent successful missions.

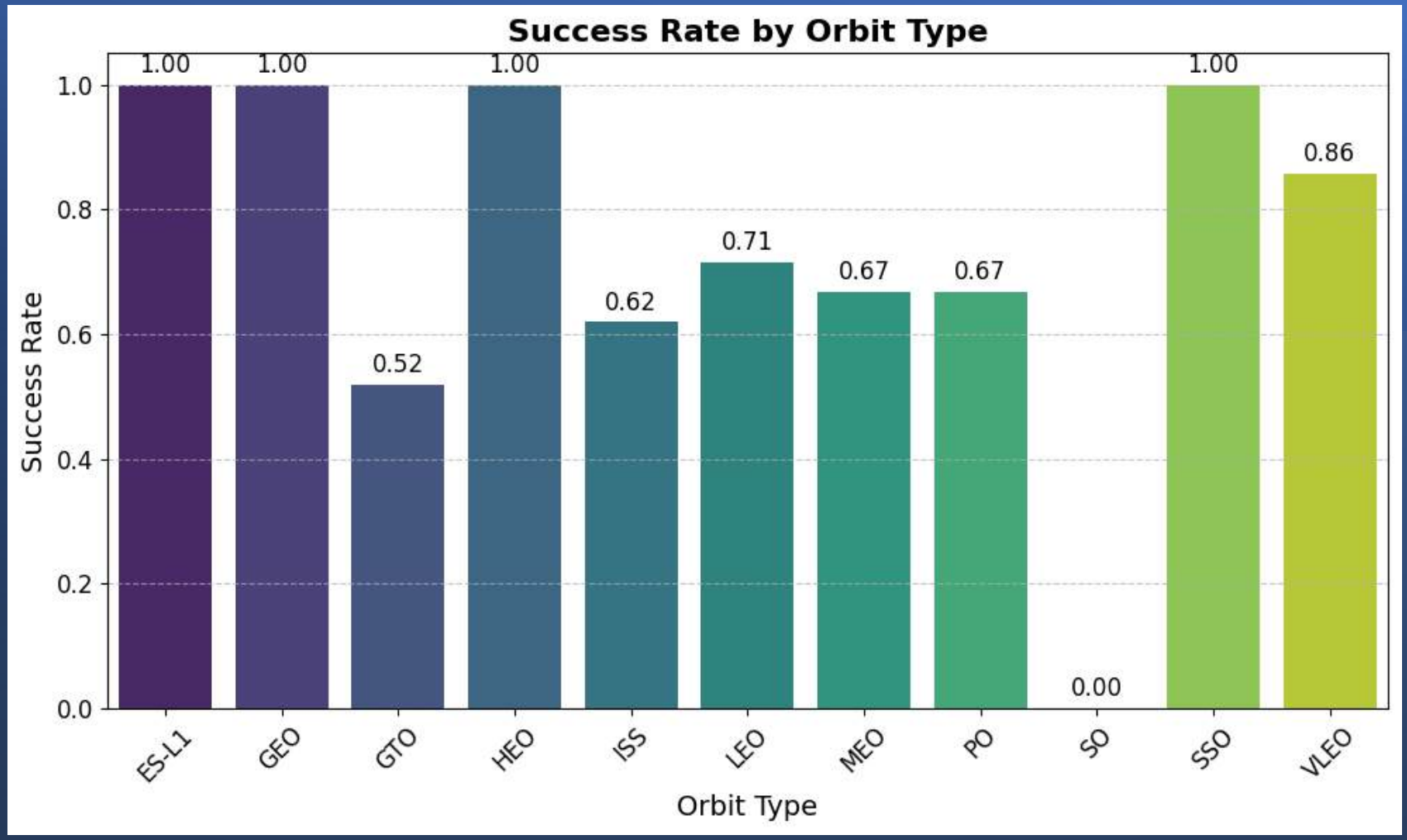
Insights:

✓ Higher payloads and success rate:
- For CCAFS SLC 40 and KSC LC 39A, as the payload mass increases, a significant proportion of missions are successful (Class 1).

✓ VAFB SLC 4E:Only a few missions are visible, mostly carrying lighter payloads with mixed outcomes.

✓ Flight Number Trends:
- Earlier flights carried smaller payloads with more failures.
- Later flights appear to handle heavier payloads with improved success rates, likely due to technological advancements.

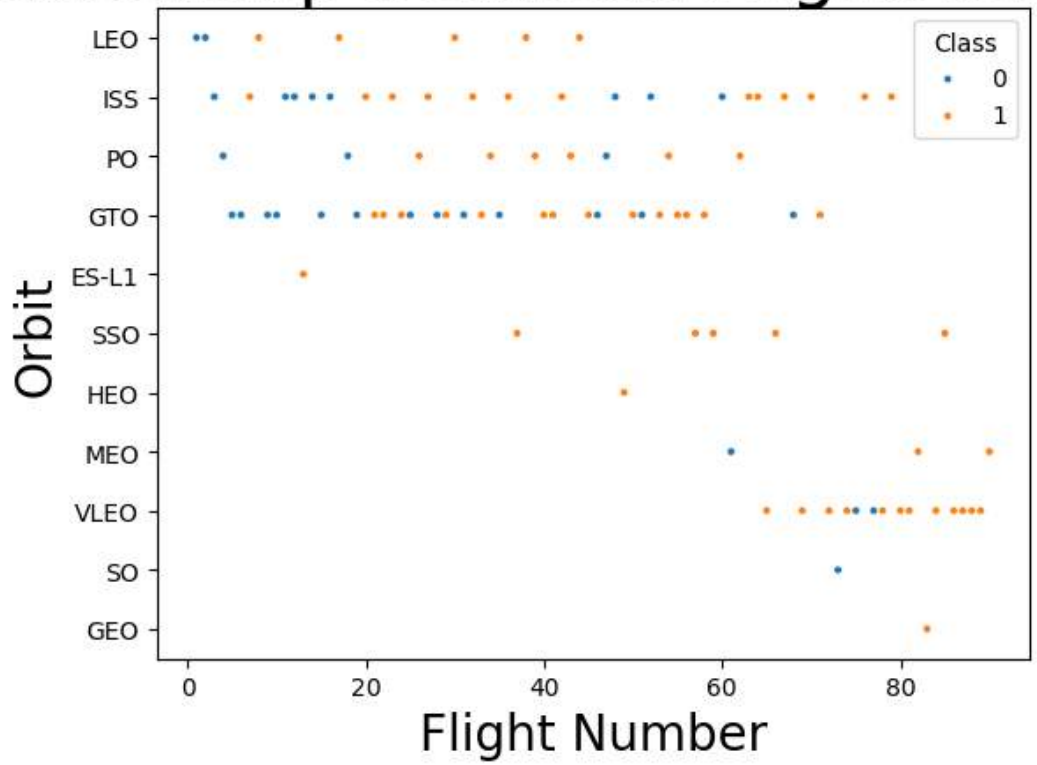
This analysis suggests a positive correlation between experience (higher flight numbers) and the capability to handle heavier payloads successfully.



Analyze the plotted bar chart to identify which orbits have the highest success rates.

ES-L1, GEO, SSO, and VLEO have success rates close to 1.0, indicating nearly perfect success for launches to these orbits. These orbits are likely well-established with proven methodologies for successful launches.

Visualize the relationship between FlightNumber and Orbit type



You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

Flight Number vs. Orbit Type

Key Features:

1.X-axis (Flight Number):
- •Represents the sequence of SpaceX missions.

2.Y-axis (Orbit Type):
- •Displays different types of orbits, including:
  - •**LEO** (Low Earth Orbit)
  - •**ISS** (International Space Station)
  - •**PO** (Polar Orbit)
  - •**GTO** (Geostationary Transfer Orbit)
  - •**SSO** (Sun-Synchronous Orbit)
  - •And others like GEO, VLEO, and MEO.

3.Data Points:
- •**Blue dots**: Represent Class 0 (failed missions).
- •**Orange dots**: Represent Class 1 (successful missions).

.

Observations:

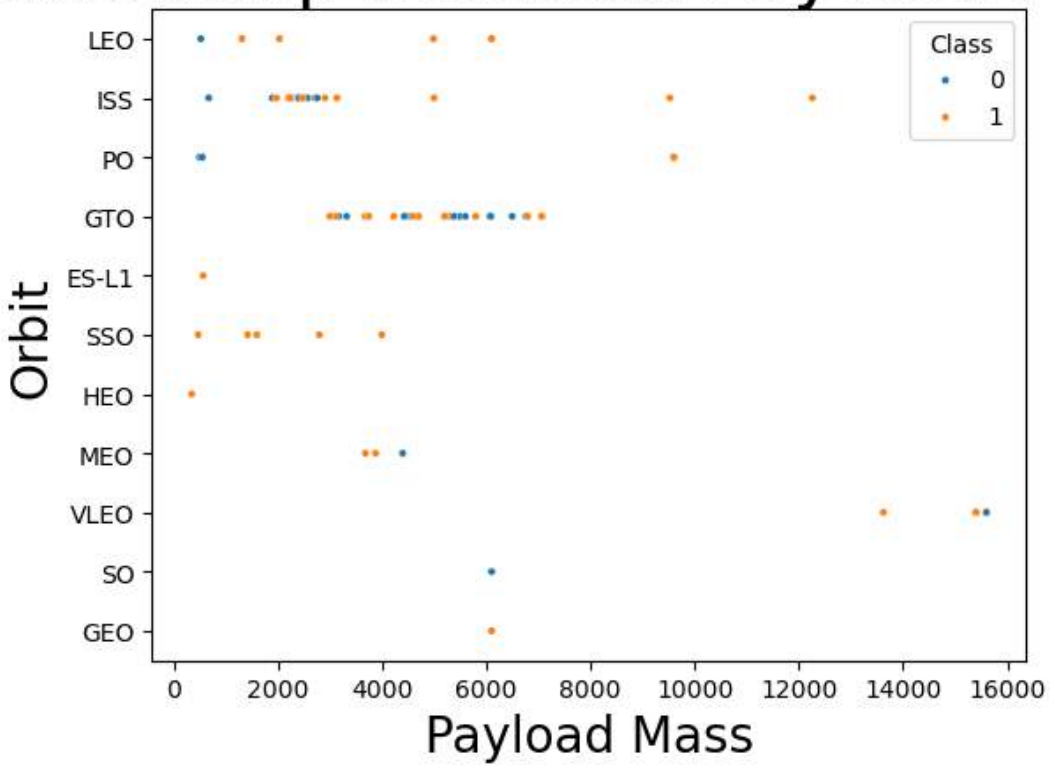
- •**LEO and ISS orbits**:
  - •These orbits dominate the early flights, showing a mix of successes and failures.
  - •Over time, the success rate improves, as shown by more orange dots in later flights.
- •**GTO and SSO orbits**:
  - •These orbits show relatively fewer failures, with a higher success rate even in earlier flight numbers.
- •**Higher flight numbers and orbit diversity**:
  - •Later flights show missions targeting more diverse orbits, such as GEO and MEO, with higher success rates.

Insights:

- •**Learning curve**: As flight numbers increase, SpaceX likely improved technology and processes, leading to higher success rates across all orbit types.
- •**Orbit-specific trends**: Some orbits, like GTO and SSO, appear to have inherently higher success rates compared to LEO and ISS, possibly due to mission-specific considerations.

This analysis shows that both orbit type and flight experience influence the success of missions

Visualize the relationship between Payload Mass and Orbit type



With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

Payload vs. Orbit Type

Key Features:

1.X-axis (Payload Mass):
- **Represents the weight of the payload carried, ranging from small to very heavy loads (up to ~16,000 kg).**

2.Y-axis (Orbit Type):
- **Displays different orbit types, such as:**
- **LEO (Low Earth Orbit)**
- **ISS (International Space Station)**
- **GTO (Geostationary Transfer Orbit)**
- **SSO (Sun-Synchronous Orbit)**
- **And others like GEO, MEO, and VLEO.**

3.Data Points:
Blue dots: Represent Class 0 (failed missions).
Orange dots: Represent Class 1 (successful missions).

.

Observations:

•**LEO and ISS orbits**:
- These orbits feature a wide range of payload masses, with a mix of successes and failures.
- Success rates improve with moderate payloads (around 4,000 to 6,000 kg).

•**GTO and SSO orbits**:
- These orbits show more consistent success (Class 1) across a wide payload range, indicating reliability for these mission types.

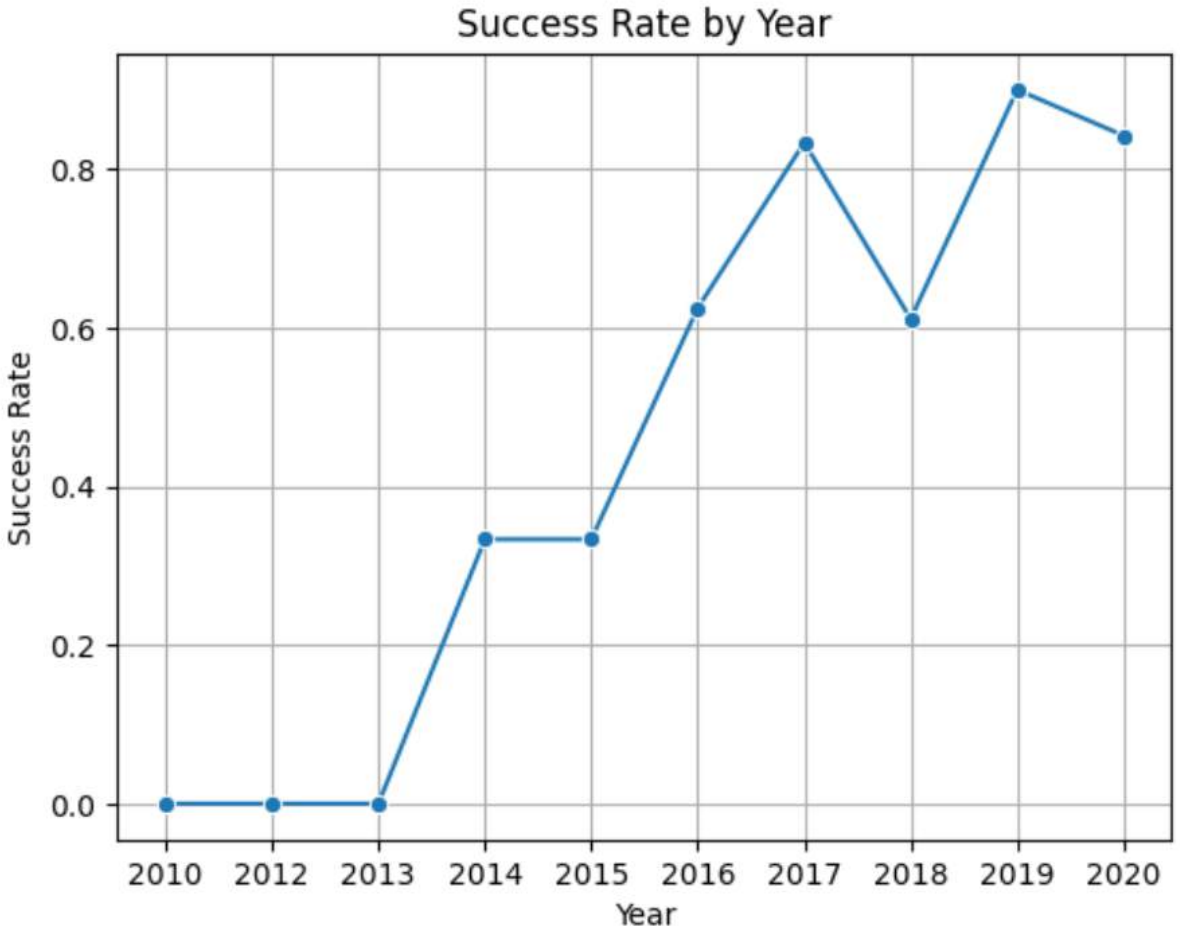
•**Heavier payloads (over 10,000 kg)**::
- Limited to specific orbits like GTO and GEO, with a high success rate.

Insights:

•**Payload and orbit-specific success rates**:
- Heavier payloads are typically successful for certain orbits, suggesting that SpaceX has optimized its technology for high-payload missions.
- Some orbits (e.g., GTO and SSO) exhibit consistently high success rates regardless of payload.

This analysis suggests that both payload mass and orbit type significantly impact mission success.

Launch Success Yearly Trend



you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

Display the names of the unique launch sites in the space mission

```
%%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

Explanation:

- SELECT DISTINCT: This command retrieves unique values from the specified column.
- Launch_Site: The column that contains the names of the launch sites in the SPACEXTBL table.
- The query ensures that any duplicate entries are removed, returning only unique launch site names.

Results: The query outputs the following unique launch sites:

1.CCAFS LC-40

2.VAFB SLC-4E

3.SC LC-39A

Insight:

These are the three unique launch sites used for SpaceX missions, with CCAFS LC-40 appearing twice in the table (but only once in the query result due to DISTINCT).

Launch Site Names Begin with 'CCA'

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

In [13]:
```
%sql SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
```

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

Out[13]:

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

Insights:

- All missions in this query were launched from CCAFS LC-40.
- Payloads varied in type and weight, with orbits primarily targeting LEO (ISS).
- While mission outcomes were successful, landing outcomes showed room for improvement (e.g., failures or no attempts).

Total Payload Mass

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
```

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

TOTAL_PAYLOAD_MASS__KG_
45596

Insight:
This shows the cumulative payload mass successfully delivered by SpaceX for NASA's CRS program, highlighting the scale of their collaboration for resupplying the International Space Station (ISS).

Average Payload Mass by F9 v1.1

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

In [14]:
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE Booster_Version =  'F9 v1.1';
```

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

Out[14]:

AVERAGE_PAYLOAD_MASS__KG_
2928.4

Insight:
This value reflects the typical payload capacity of the **F9 v1.1** booster, showcasing its performance for medium-sized payload missions.

First Successful Ground Landing Date

In [16]:
```
%sql SELECT MIN(Date) AS First_Successful_Landing FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)';
```

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

Out[16]:

First_Successful_Landing
2015-12-22

Insight:
This date marks a pivotal milestone in SpaceX's history, showcasing its progress in achieving reusable rocket technology with ground pad landings.

Successful Drone Ship Landing with Payload between 4000 and 6000

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

In [17]:
```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = 'Success (drone ship)'AND PAYLOAD_MASS__KG_ > 4000 AND PA
```

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

Out[17]:

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

Insight:
These boosters successfully carried payloads between 4000 and 6000 kg and achieved successful landings on drone ships, demonstrating their reliability for medium payload missions.

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%sql SELECT Mission_Outcome, COUNT(*) AS Total_Count FROM SPACEXTBL GROUP BY Mission_Outcome;
```

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

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

Insight:
This result highlights SpaceX's high success rate, with **98 successful missions** out of the total recorded. There was one in-flight failure and one mission where the payload status was unclear, indicating rare instances of mission issues.

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

Insight:
These booster versions successfully carried the heaviest payloads, showcasing their advanced capabilities and reliability for high-demand missions.

2015 Launch Records

```
%%sql
SELECT
    CASE substr(Date, 6, 2)
        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,
    Booster_Version,
    Launch_Site,
    Landing_Outcome
FROM SPACEXTBL
WHERE
    substr(Date, 0, 5) = '2015'
    AND Landing_Outcome LIKE 'Failure (drone ship)';
```

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

Month_Name	Booster_Version	Launch_Site	Landing_Outcome
January	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
April	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

Insight:
These failed attempts on a drone ship in 2015 reflect SpaceX's early challenges with booster recovery, marking critical learning points for future success.

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

In [21]:
```
%%sql SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTBL WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Outcome_Count DESC;
```

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

Out[21]:

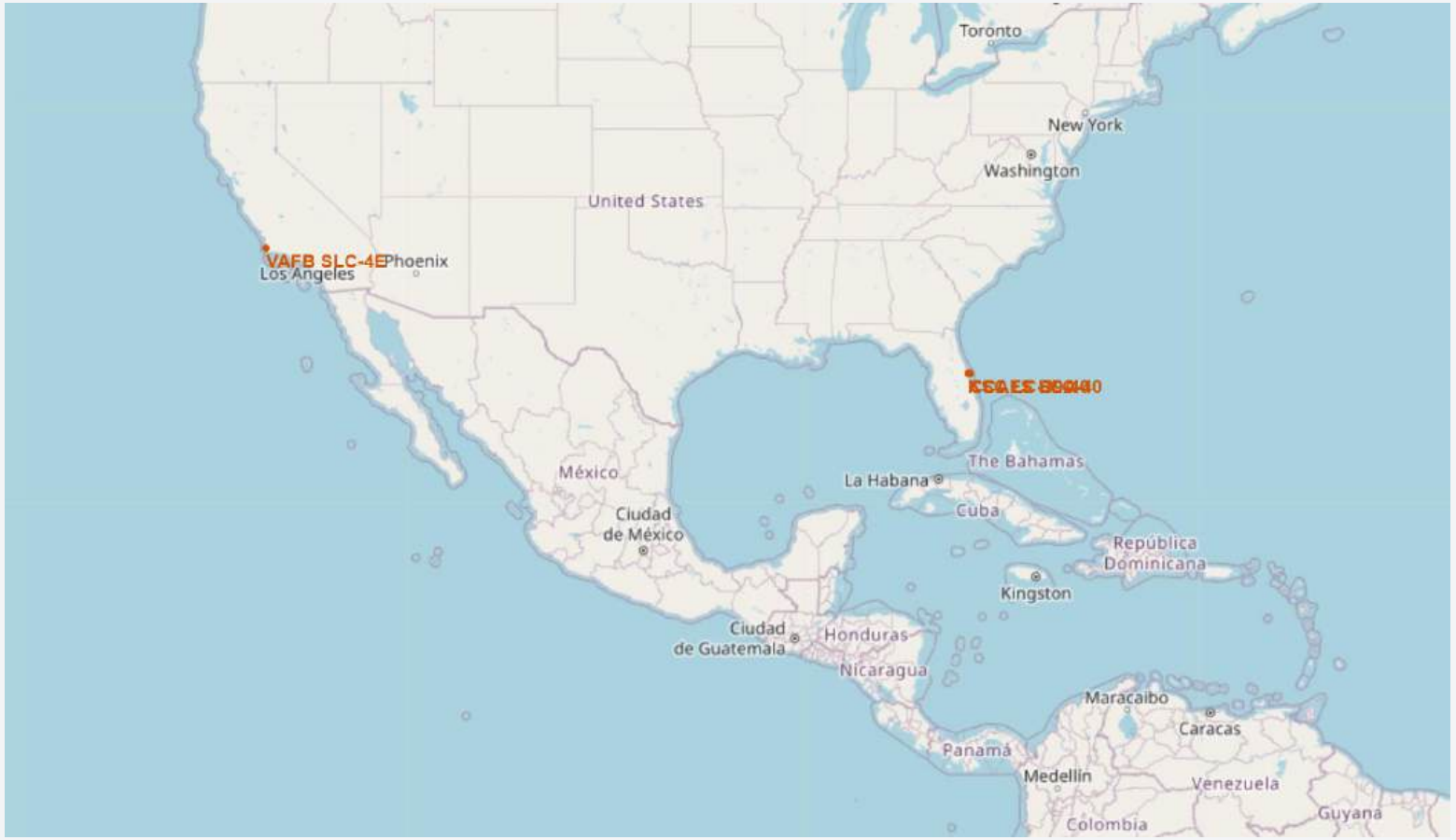
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Insight:
The most common outcome was **No attempt** (10 instances), followed by **Success (drone ship)** and **Failure (drone ship)** (5 each). This data highlights SpaceX's early challenges and their progress toward successful landings.

Section 3

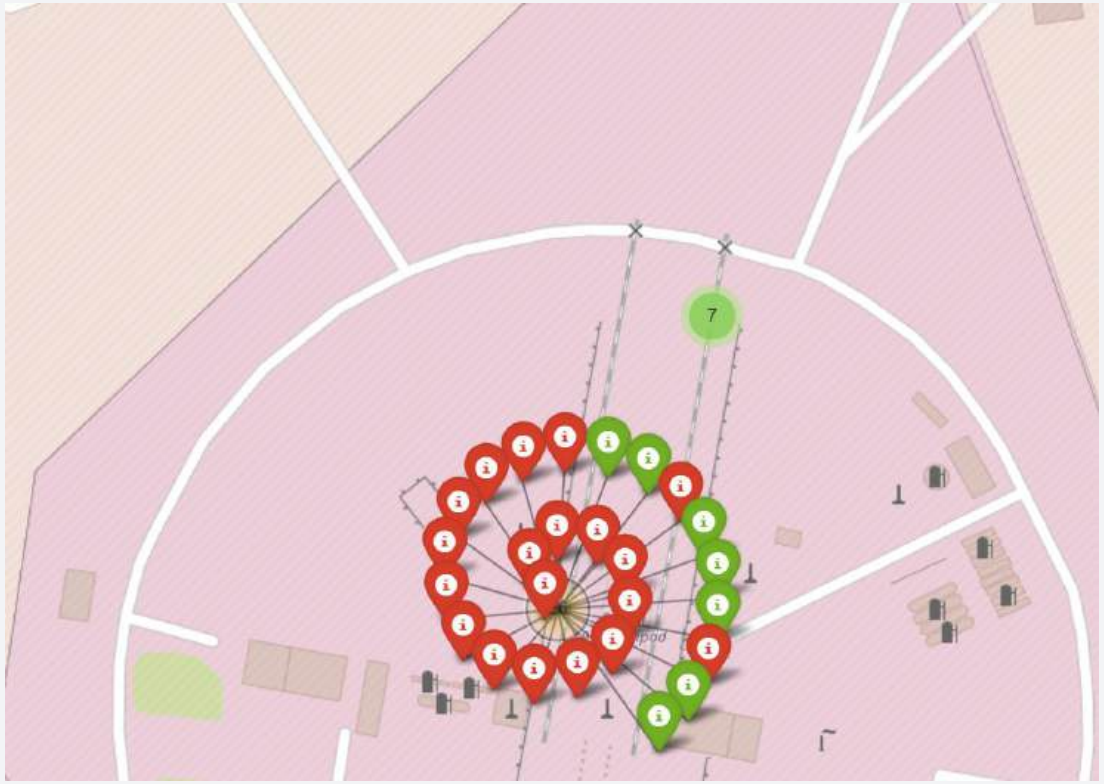
Launch Sites Proximities Analysis

Global Launch Sites Map



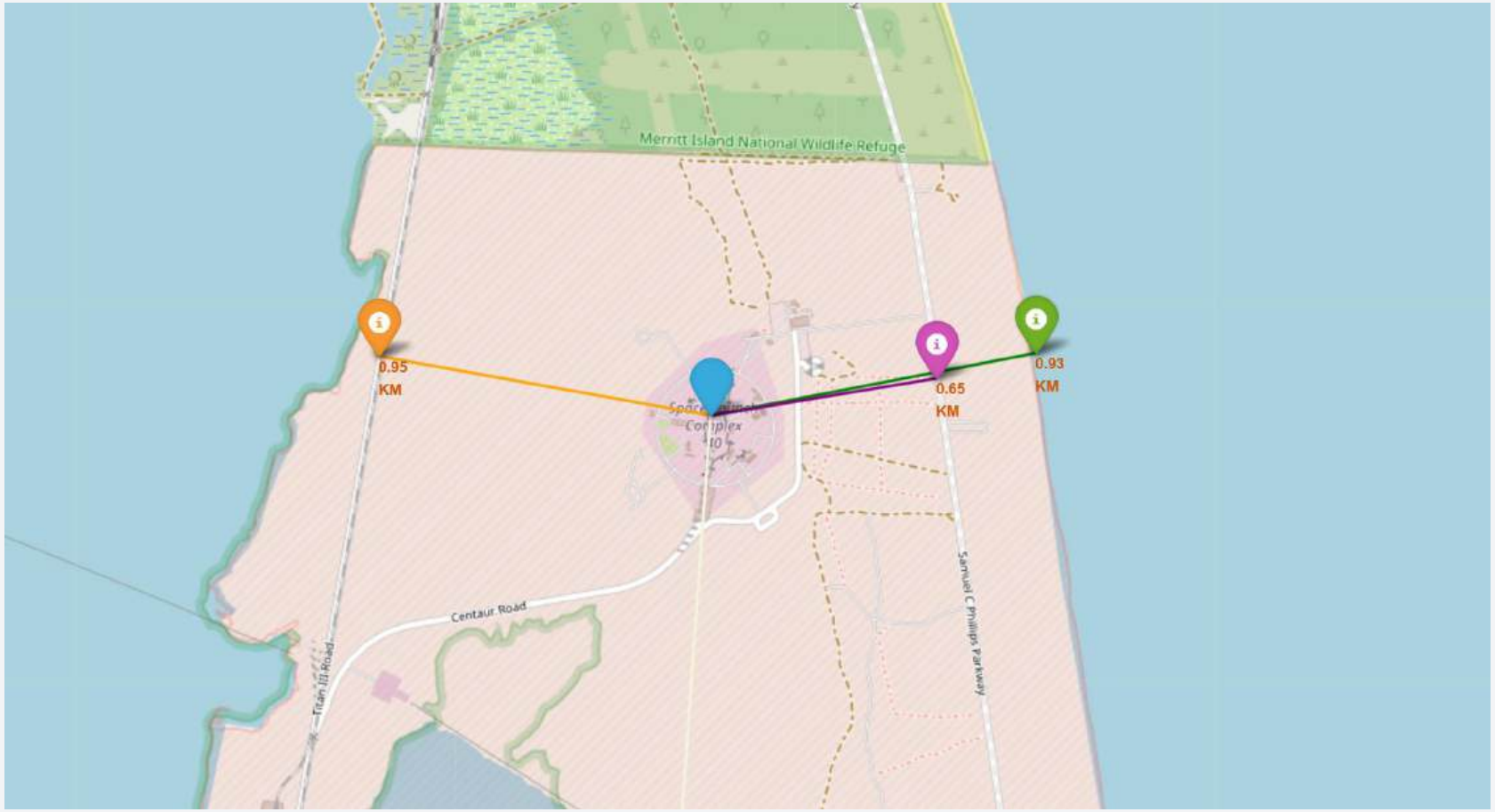
This map shows the geographic locations of all SpaceX launch sites globally.

Launch Outcomes with Color-Coded Markers



This map displays launch sites with markers indicating the outcomes of launches (e.g., green for success, red for failure)

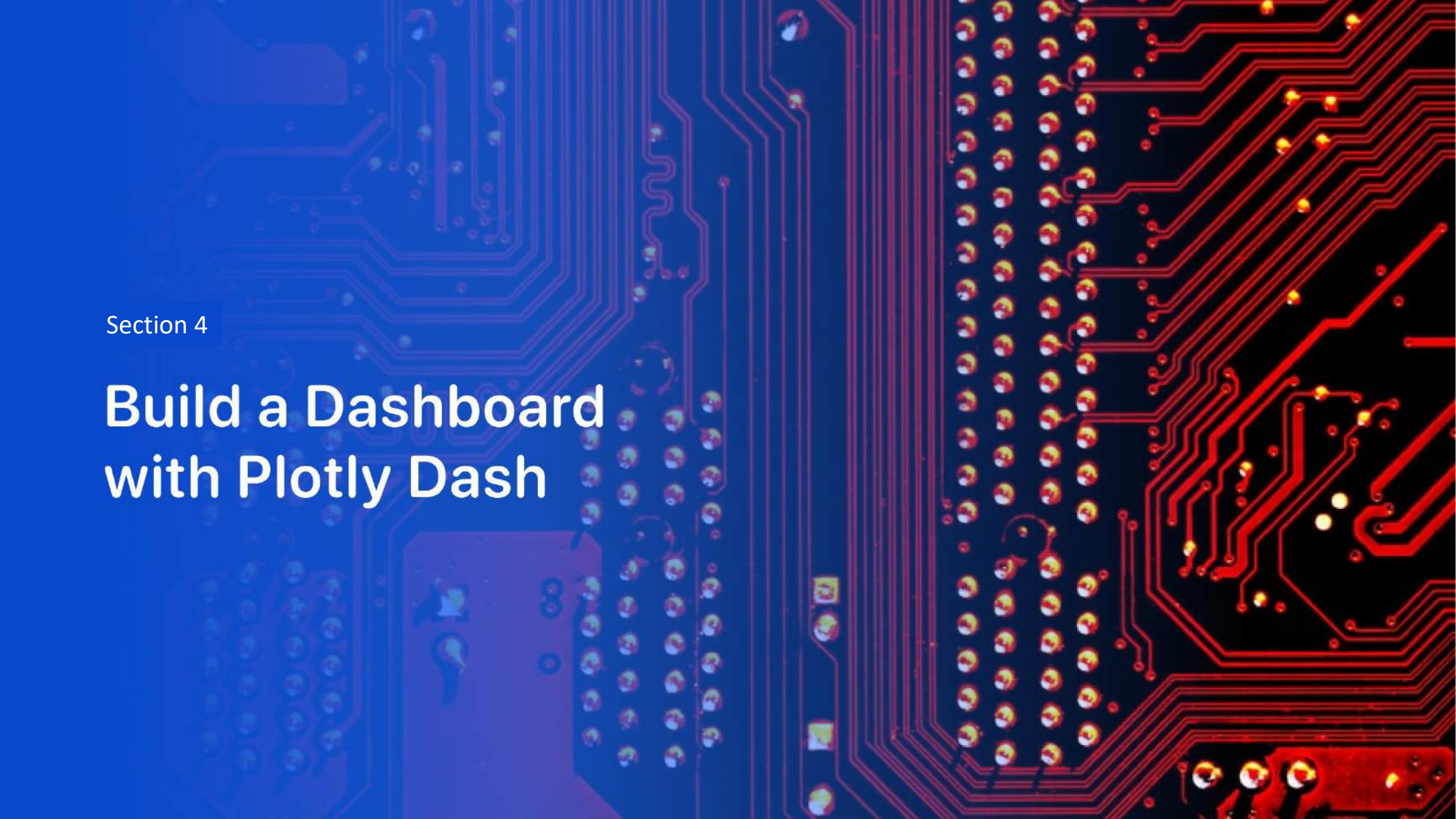
Proximity Analysis of Launch Sites



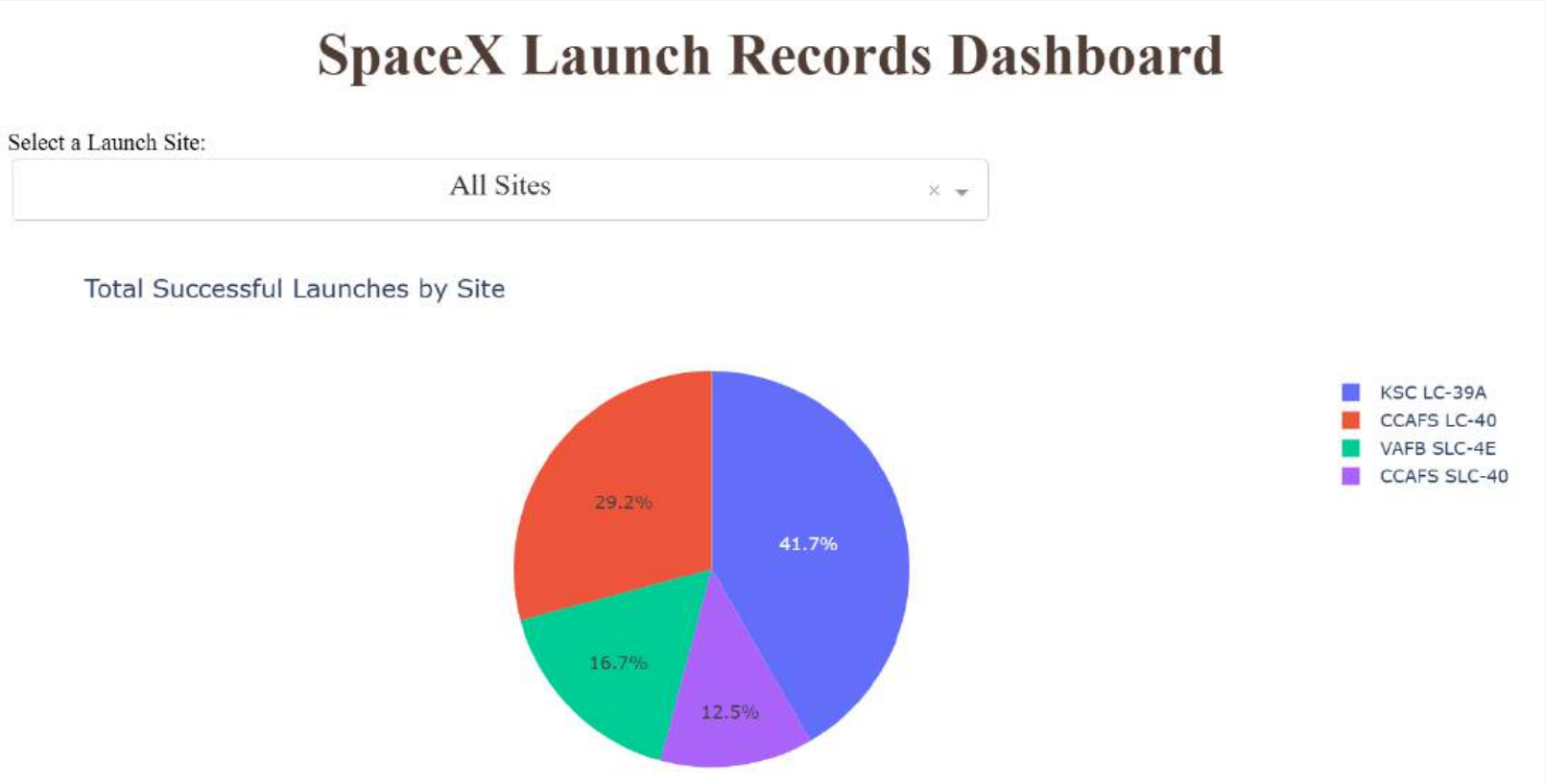
This map highlights the proximities of launch sites to key infrastructure, such as railways, highways, and coastlines, with distance measurements.

Section 4

Build a Dashboard with Plotly Dash



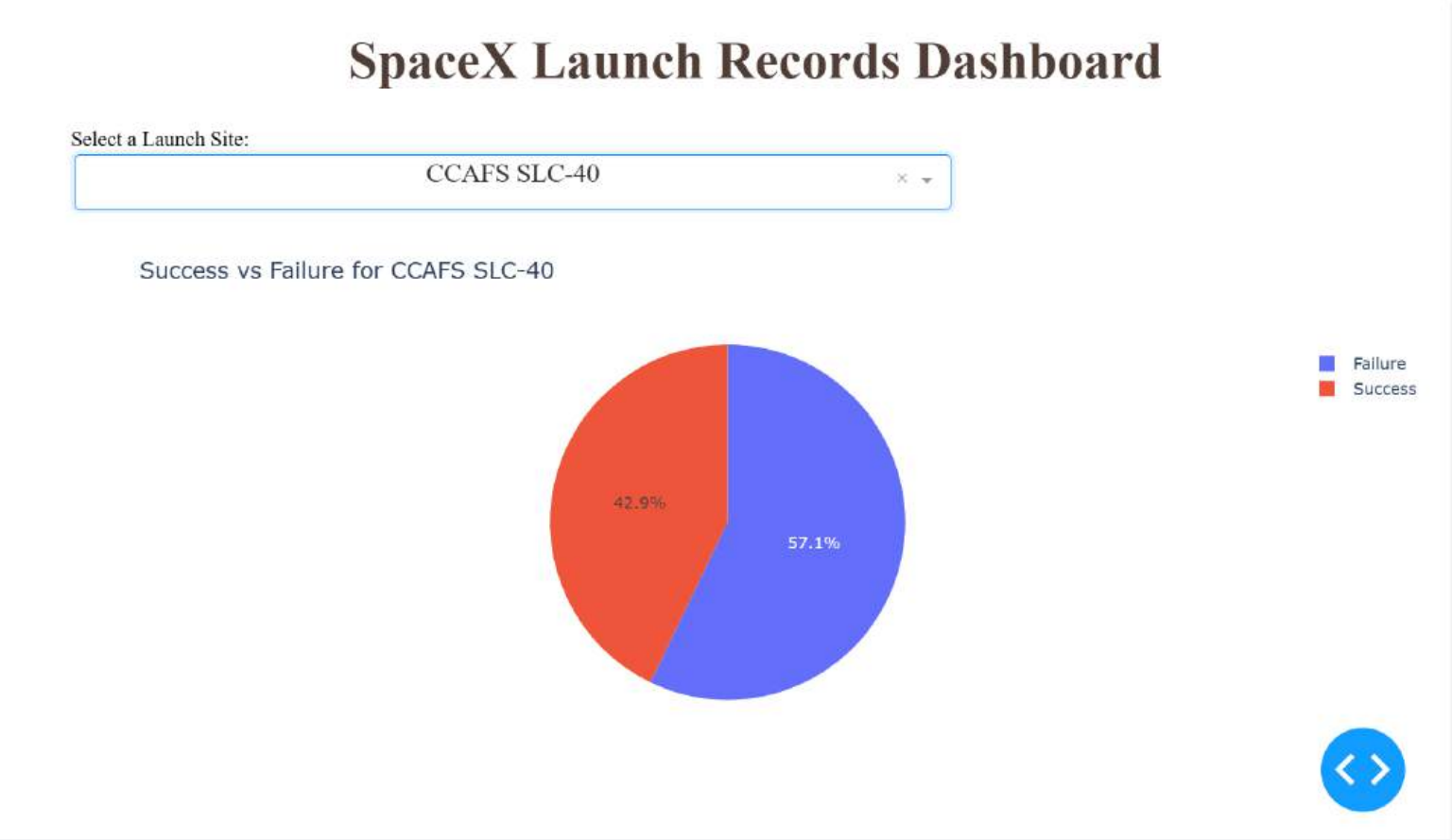
SpaceX Launch Records Dashboard



Key Insights:

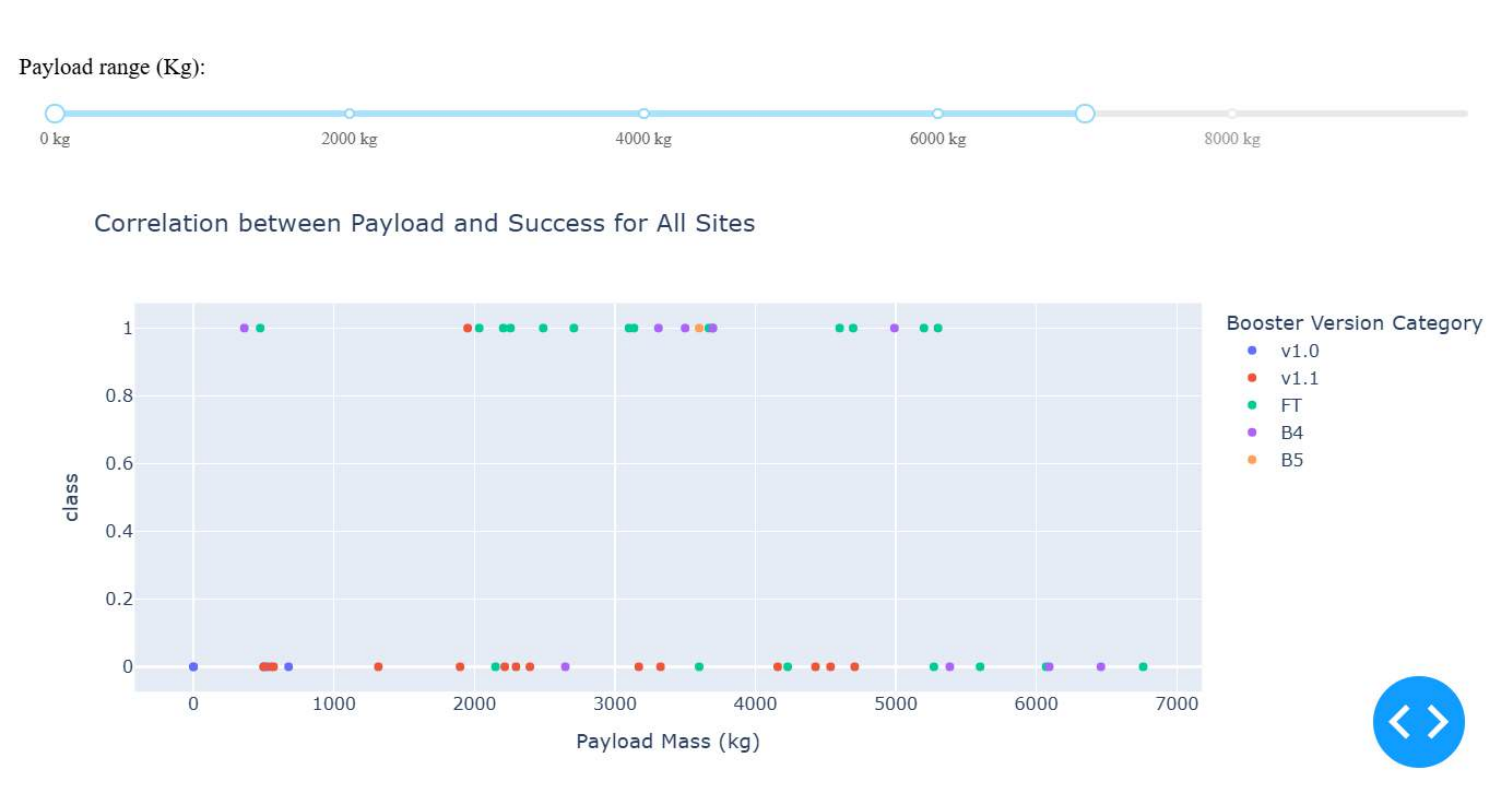
- **KSC LC-39A** has the highest success rate, contributing to nearly half of all successful launches.
- **CCAFS LC-40** is the second most successful site, with a significant share of 29.2%.
- **VAFB SLC-4E** and **CCAFS SLC-40** have comparatively lower success rates, highlighting potential differences in their usage or operational challenges.

Success vs Failure for Launch Site CCAFS SLC-40



This visual clearly shows that the failure rate for this site is higher than the success rate, emphasizing the need for improvement at this specific launch location.

<orrelation Between Payload and Launch Success for All Sites>



Key Findings:

•**FT** and **B5** booster versions exhibit the highest success rates, especially in the **2000–6000 kg** payload range.

•Older booster versions (**v1.0** and **v1.1**) are more prone to failures, particularly with lighter payloads (<2000 kg).

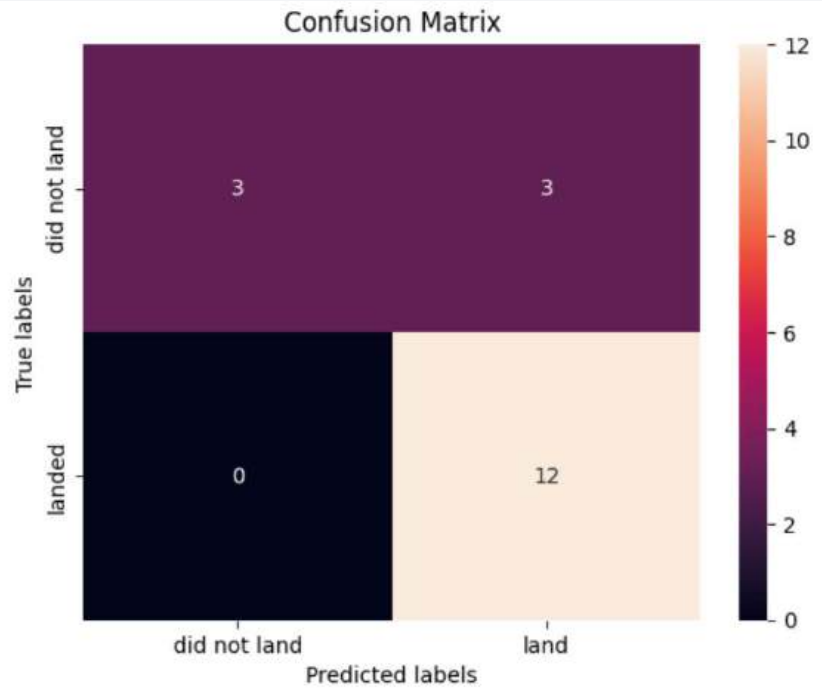
Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart
- Find which model has 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 problem is false positives.

Overview:

True Postive - 12 (True label is landed, Predicted label is also landed)

False Postive - 3 (True label is not landed, Predicted label is landed)

Insights:

•**High TP (12)**: Indicates strong predictive accuracy for landings.

•**Moderate FP (3)**: Shows the model occasionally predicts a landing when it didn't actually happen.

•**No FN (0)**: Suggests excellent sensitivity, as all actual landings were correctly predicted.

•**Model Strengths**:

- Good balance between precision and recall.
- Indicates that logistic regression is effective in distinguishing between "landed" and "did not land" classes.

This confusion matrix demonstrates the reliability of the model, with only minor misclassifications.

Conclusions

Point 1:
KSC LC-39A has the highest success rate, contributing to nearly half of all successful launches. This highlights its significance as SpaceX's most reliable launch site.

Point 2:
CCAFS LC-40, the second most successful site, accounts for 29.2% of the total launches, showcasing its role as a key site, though with room for improvements compared to KSC LC-39A.

Point 3:
The failure rate for CCAFS SLC-40 is higher than the success rate, emphasizing the need for further improvements in operational and technical processes for this specific location.

Point 4:
Analysis of payload vs. success shows that medium-range payloads (2000–6000 kg) tend to have higher success rates. This suggests an optimal payload range for successful missions.

Point 5:
Booster versions FT and B5 exhibit the highest success rates, demonstrating the advancements in technology with these newer models compared to older versions like v1.0 and v1.1.

Point 6:
The confusion matrix for the logistic regression model shows excellent sensitivity, with no false negatives and high true positives. This indicates the model's strong predictive accuracy for successful landings.

Point 7:
Interactive dashboards and data visualizations reveal clear patterns, such as the positive correlation between higher payload masses and increased success rates for certain orbits like GTO and SSO.

Point 8:
Early missions, particularly at VAFB SLC-4E, experienced a higher proportion of failures, but recent trends show improved success rates across all sites and orbits, reflecting learning curves and technological enhancements.

Point 9:
The proximity analysis of launch sites to infrastructure (e.g., railways, highways, coastlines) highlights strategic placement and the logistical advantages that influence mission outcomes.

Point 10:
Overall, SpaceX has demonstrated a steady increase in success rates, emphasizing the importance of continuous improvements in booster design, site operations, and mission planning.

Appendix

Code Snippets

- *Python code for data preprocessing, visualization, and machine learning model training.*
- *SQL queries for data extraction and analysis, including success rate calculations and payload trends.*

Charts and Visualizations

- *Scatter plots visualizing the relationship between Flight Number, Launch Site, and Payload Mass.*
- *Pie charts depicting launch success rates by site and overall mission outcomes.*
- *Correlation scatter plots between payload and success rates.*

Notebook Outputs

- *Interactive dashboards displaying success/failure by site and payload ranges.*
- *Confusion matrix showcasing the accuracy of the best-performing model.*

Data Sets

- *Cleaned and processed SpaceX mission data used for exploratory analysis and machine learning.*
- *Supplementary data on infrastructure proximity for launch site analysis*.

Thank you!