IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

RAJALAXMI PRADHAN
03-23-2025

Outline

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

Executive Summary

- Provide a high-level overview of the project, summarizing the purpose, methodology, key findings, and recommendations.
- Mention the goal of predicting the likelihood of a successful Falcon 9 rocket landing using machine learning models.
- Briefly state the data sources, key analytical steps, and model performance.

Introduction

- Describe the context of the project, focusing on the increasing importance of reusable rockets for reducing space travel costs.
- Introduce SpaceX's success with Falcon 9 and why a competitor would be interested in predicting landing success.
- State the problem statement: Predict whether the first stage of a Falcon 9 rocket will successfully land.

Section 1

Methodology

Methodology

This project followed a systematic data science approach to predict the success of the Falcon 9 first-stage landing. The key steps include:

1. Data Collection

- **Sources: Data was collected using the SpaceX API, which provided detailed historical information about launches.**
- **Additional Data: Relevant datasets on launch sites, payloads, and mission details were gathered.**
- **Tools Used: Python libraries like requests were used for API extraction, while Pandas managed data manipulation.**

2. Data Wrangling

- **Cleaning: The raw data was cleaned by removing irrelevant columns, handling missing values, and correcting data types.**
- **Formatting: Data was normalized and formatted for consistency using Pandas.**
- **Handling Nulls: Missing values were imputed using appropriate methods, such as mean or median imputation.**

3. Exploratory Data Analysis (EDA)

- Visualization: Data patterns and relationships were explored using libraries like Matplotlib and Seaborn.
- Correlation Analysis: Features with a high correlation to successful landings were identified.
- Insights: Visualizations such as scatter plots, bar charts, and correlation matrices helped uncover trends in launch success.

4. Feature Engineering

- Feature Creation: New features such as payload mass ranges, launch site categories, and booster versions were engineered.
- Encoding: Categorical data like launch sites and payload types were encoded using one-hot encoding.
- Normalization: Features were scaled using StandardScaler for uniform model input.

•5. Model Development

•Algorithms: Various models were applied, including:

  •Logistic Regression for binary classification.

  •Support Vector Machines (SVM) for complex decision boundaries.

  •Decision Trees for interpretable results.

•Hyperparameter Tuning: GridSearchCV was used to optimize model performance.

- **6. Model Evaluation**
- **Metrics:** The models were evaluated using key metrics, including:
- **Accuracy Score** to measure the overall performance.
- **Confusion Matrix** to visualize prediction results.
- **Precision, Recall, and F1 Score** for detailed performance analysis.
- **Best Model Selection:** The model with the highest accuracy and balanced performance was selected for final predictions.

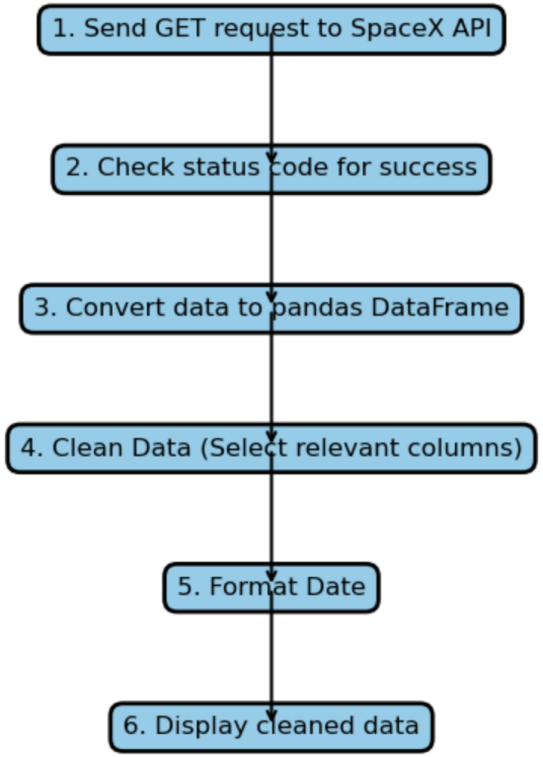
Data Collection

Key Phrases for Data Collection Process:

- **Identify Data Needs: Determine what information is required for analysis.**
- **Select Data Sources: Choose where to gather data from (e.g., APIs, web scraping).**
- **Use APIs: Access data through APIs like the SpaceX REST API to get launch data.**
- **Make Requests: Send requests to the API to retrieve data (GET requests).**
- **Receive Data: Get the data back in JSON format.**
- **Normalize Data: Convert JSON data into a structured format (like a table) using tools like** json_normalize**.**
- **Clean Data: Remove unnecessary data and handle missing values (e.g., replacing NULLs).**
- **Store Data: Save the cleaned data in a format suitable for analysis (e.g., Pandas DataFrame).**

Data Collection – SpaceX API

- **Request Data**: We send a GET request to the SpaceX API (https://api.spacexdata.com/v4/launches).
- **Convert JSON to DataFrame**: After successfully fetching the data, we convert it into a pandas DataFrame for easier manipulation.
- **Clean Data**: We select only the columns we care about (name, date_utc, rocket, success, and links).
- **Format Date**: We convert the date_utc column to a more readable format (YYYY-MM-DD HH:MM:SS).
- **Display Data**: We display the cleaned and formatted DataFrame.
- GitHub URL (Data Collection):



Data Collection - Scraping

- **Request Data: We send a GET request to the SpaceX API (https://api.spacexdata.com/v4/launches).**
- **Convert JSON to DataFrame: After successfully fetching the data, we convert it into a pandas DataFrame for easier manipulation.**
- **Clean Data: We select only the columns we care about (name, date_utc, rocket, success, and links).**
- **Format Date: We convert the date_utc column to a more readable format (YYYY-MM-DD HH:MM:SS).**
- **Display Data: We display the cleaned and formatted DataFrame.**
- **GitHub URL - Web Scraping**

```
Start
  |
  ▼
Web Scrape Page to Get HTML
  |
  ▼
Create BeautifulSoup Object
  |
  ▼
Select Tables
  |
  ▼
Extract Column Names from <th>
  |
  ▼
Parse Launch Table Data
  |
  ▼
Create Pandas DataFrame
  |
  ▼
End
```

Data Wrangling

- **Training Labels:**
- **1**: Successful landing (True Ocean, True RTLS, True ASDS).
- **0**: Unsuccessful landing (False Ocean, False RTLS, False ASDS).
- **EDA Steps:**
- **Load and Inspect Data**: Load the dataset and examine its structure (df.info(), df.head()).
- **Analyze "Landing Outcome"**: Identify patterns in the Landing Outcome column.
- **Convert to Binary Labels**: Map the landing outcomes to binary labels (1 for success, 0 for failure).
- **Check for Label Imbalance**: Check the distribution of labels (df['Class'].value_counts()).
- **Summarize Data**: Use summary statistics (df.describe()) and visualizations to understand data patterns.
- [GitHub URL - Data Wrangling]

```
Start
  |
  ▼
Load the Dataset (e.g., CSV)
  |
  ▼
Inspect Data Structure (df.info(), df.head())
  |
  ▼
Analyze "Landing Outcome" Column
(Identify patterns in landing outcomes)
  |
  ▼
Convert Landing Outcomes to Binary Labels
(1 for success, 0 for failure)
  |
  ▼
Check for Label Imbalance
(Examine label distribution with df['Class'].value_counts())
  |
  ▼
Summarize Data (df.describe(), Visualizations)
  |
  ▼
End
```

EDA with Data Visualization

- **Plotting Payload Mass vs. Flight Number: A catplot was used to show how the payload mass varies with the flight number, colored by the success (Class). This helps in observing any correlation between payload size and launch success.**
- **Plotting Payload Mass vs. Launch Site: Another catplot shows how payload mass relates to different launch sites. This visualizes any differences in success rates for different sites and how payload mass might influence those.**
- **Orbit Success Rate: A barplot shows the success rate of launches for each orbit type. This helps identify which orbit types are more successful and can help guide decision-making for future launches.**
- **Scatter Plots for Relationships:**
  - **Flight Number vs Orbit: A scatterplot was used to visualize the relationship between flight number and orbit type, colored by class (success).**
  - **Payload Mass vs Orbit: Another scatterplot visualizes how payload mass affects orbit type success.**
- **Launch Success Yearly Trend: A line plot was used to analyze how launch success has changed over the years. This helps track SpaceX's performance over time.**
- **GitHub URL - EDA with Data Visualization**

EDA with SQL

SQL queries were written to retrieve data related to the following aspects:

- **Launch Sites: Information about the different locations where the rockets were launched.**
- **Payload Masses: Data on the mass of the payloads carried by the rockets.**
- **Dates: The dates on which the launches took place.**
- **Booster Types: Details about the types of boosters used for the missions.**
- **Mission Outcomes: Results of the missions, indicating whether they were successful or failed.**

GitHub URL - EDA with SQL

Interactive Map with Folium

Summary of Map Objects Created:

- **Markers** were used to visualize the geographical locations of the launch sites. They allow us to see where the sites are located relative to each other, and give context to the rest of the analysis.
- **CircleMarkers** with different colors helped to visualize the success and failure rates of launches, enabling us to easily identify trends and outliers in launch outcomes based on site locations.
- **Polylines** provided a clear visual representation of distances between the launch sites and nearby infrastructures like coastlines, cities, railways, and highways. This allows us to investigate how proximity to key locations might impact the success rates of launches.
- **DivIcons** helped to add contextual information to the map (e.g., the distance to the coastline), which is crucial for deeper geographical analysis.
- GitHub URL - Folium Maps

Interactive Plotly Dash Dashboard

- The Plotly Dash dashboard featured a dropdown to filter data by a specific launch site or all sites.
- **Pie Chart**:
  - For one site: Showed success vs. failure of Falcon 9 first stage landings.
  - For all sites: Showed success distribution across sites.
- **Scatterplot**:
  - Filtered by payload mass using a slider.
  - Displayed landings by payload mass, mission outcome, and booster version.

GitHub URL - Dashboard File

Predictive Analysis (Classification)

Model Development Process for Predicting Falcon 9 First Stage Landing

1 **EDA – Analyzed features, visualized data, handled missing values, and created the 'Class' column.**

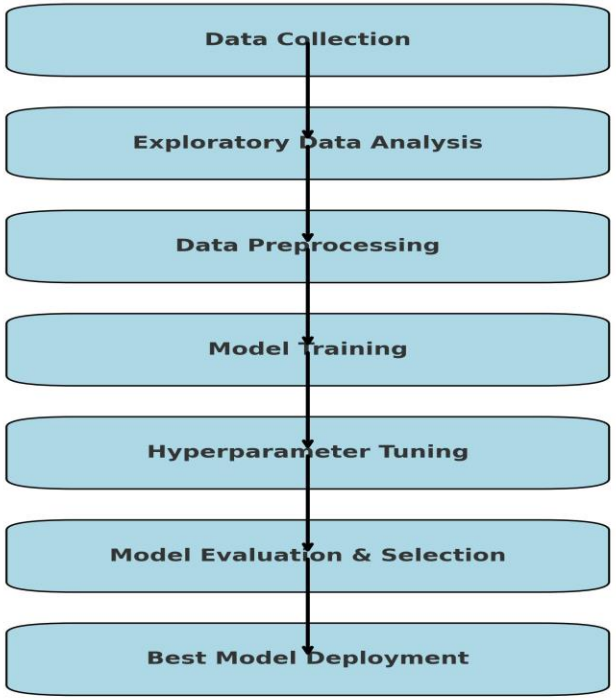
2 **Preprocessing – Standardized features, encoded categorical variables, and split data (80% train, 20% test).**

3 **Training & Tuning – Trained Logistic Regression, SVM, Decision Tree, and KNN models using GridSearchCV (cv=10).**

4 **Evaluation – Compared accuracy scores, plotted model performance, and analyzed confusion matrices.**

5 **Optimization – Fine-tuned the best model for deployment based on test accuracy and misclassification analysis.**

GitHub URL - Machine Learning



Results

1. **Exploratory Data Analysis (EDA)**

   Identified key features influencing Falcon 9 first stage landings.

   Visualized relationships using seaborn & matplotlib.

   Created a ‘Class’ column (1: Landed, 0: Not landed).

2. **Interactive Analytics (Screenshots)**

   Pie chart showing launch success rates per site.

   Scatterplot illustrating payload mass vs. success.

3. **Predictive Analysis Results**

   Trained Logistic Regression, SVM, Decision Tree, and KNN models.

   The best model was identified based on accuracy and confusion matrix analysis.

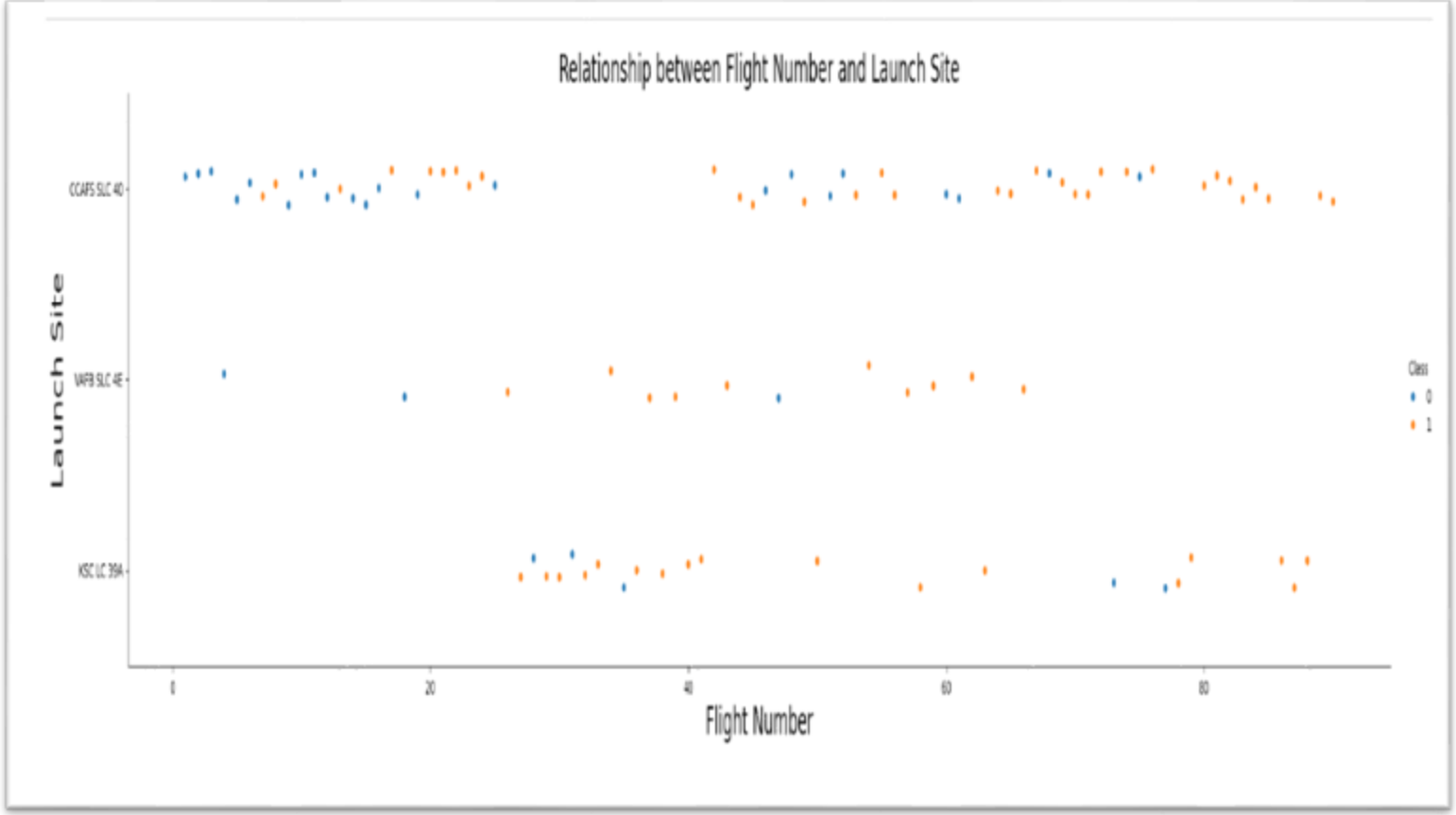
   The final optimized model was selected for deployment.

Section 2

Insights drawn from EDA

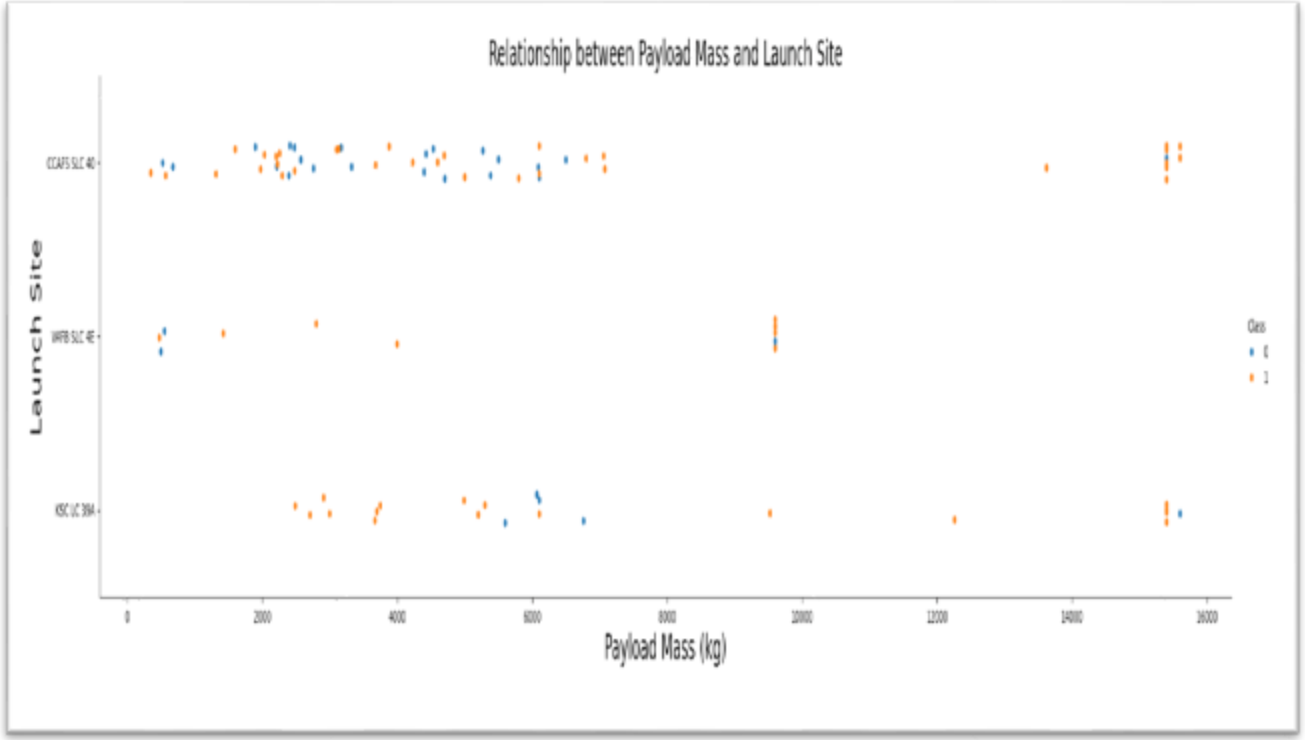
Insights from Flight Number vs. Launch Site Plot

- More launches occur at certain sites, showing operational preference.
- Early flights had more failures, but success rates improved over time.
- Some sites may have higher success rates, hinting at location advantages



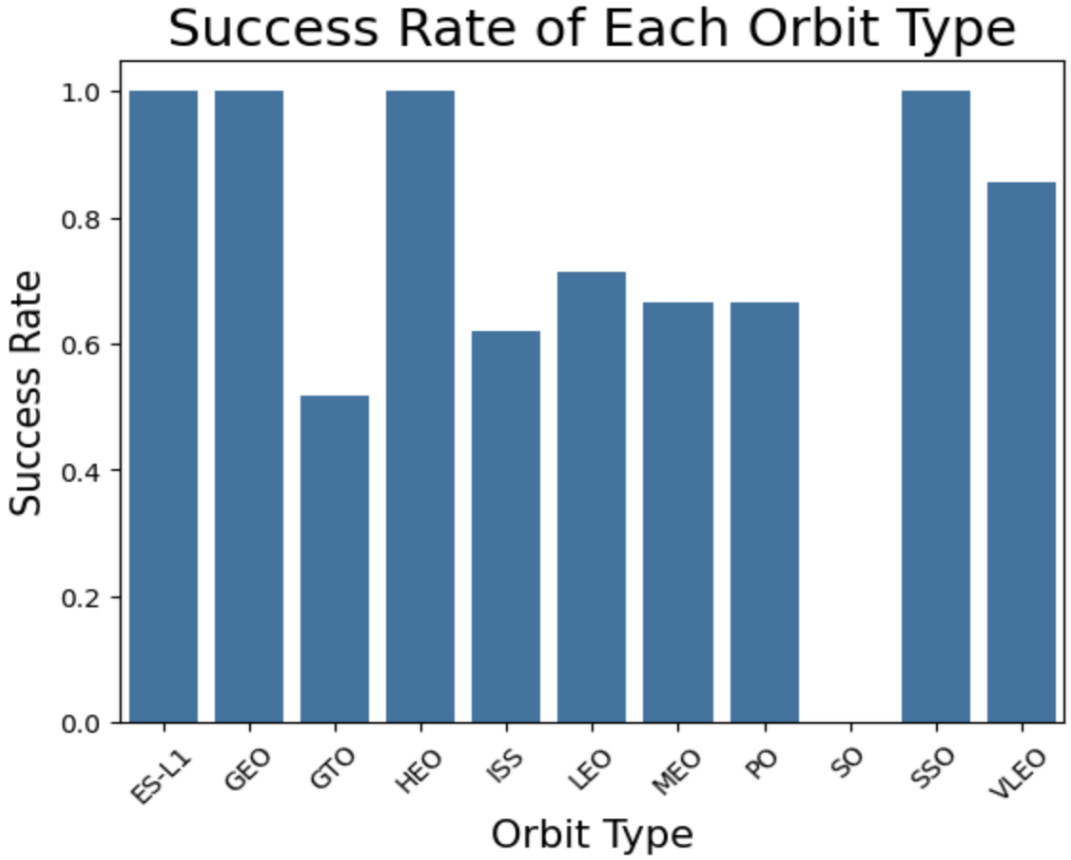
Insights from Payload vs. Launch Site

- VAFB-SLC has no launches for payloads >10,000 kg, suggesting site limitations or mission types.
- CCAFS & KSC handle heavier payloads, possibly due to better infrastructure.
- Higher payloads show mixed success rates, indicating weight impact on landing outcomes.



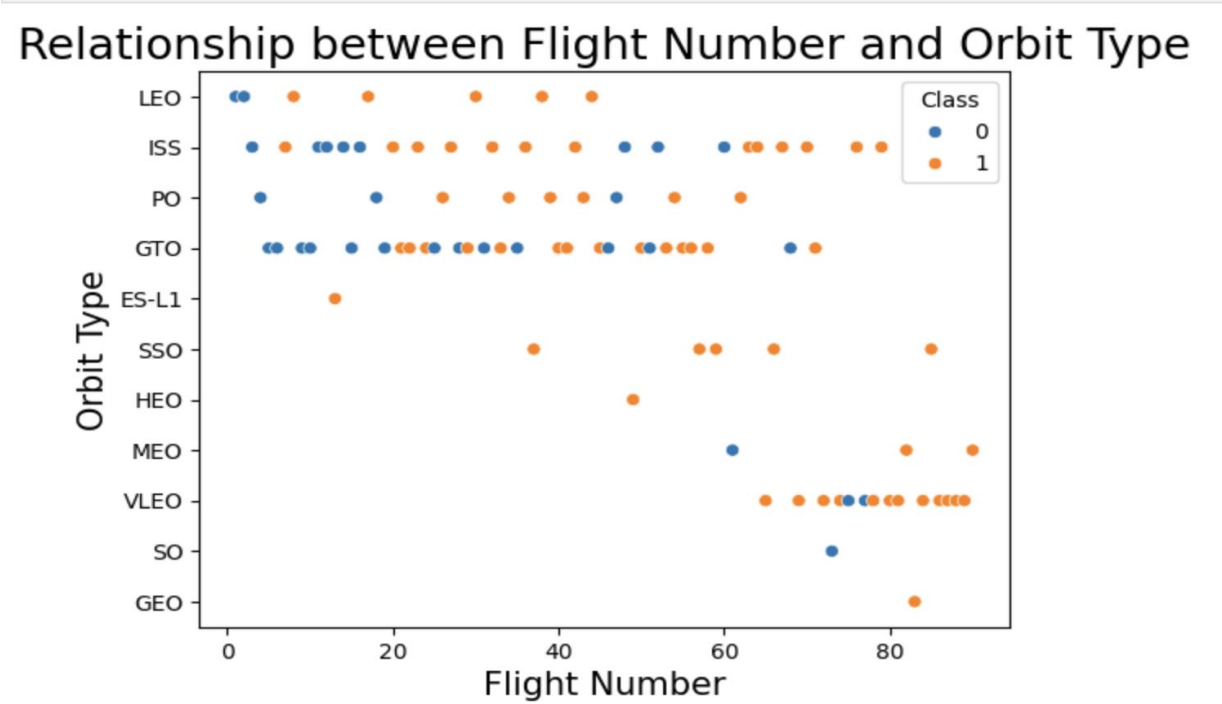
Insights from Orbit Type vs. Success Rate Chart

- **Orbits with 100% Success:**
- **ES-L1, GEO, SO, and SSO have the highest success rates, indicating they are more reliable for landings.**
- **Moderate Success Rates:**
- **ISS, MEO, PO, and LEO show success rates around 60-70%, meaning landings in these orbits are riskier.**
- **Lowest Success Rate:**
- **GTO (Geostationary Transfer Orbit) has the lowest success rate (~50%), likely due to higher energy demands and re-entry challenges.**
- **VLEO (Very Low Earth Orbit) also has a relatively high success rate, making it a promising target for future missions.**



Insights from Flight Number vs. Orbit Type

- **LEO Orbit**: There is a positive relationship between **Flight Number** and **Mission Success**, suggesting that with more flights, LEO missions tend to be more successful. This may be due to increased experience and simplified missions.
- **GTO Orbit**: No clear relationship between **Flight Number** and **Mission Success**, indicating that GTO missions are more complex, and success may depend on factors beyond the number of flights, such as technology or mission specifics.



Insights from Payload vs. Orbit Type

1. **Heavy Payloads and Success**:

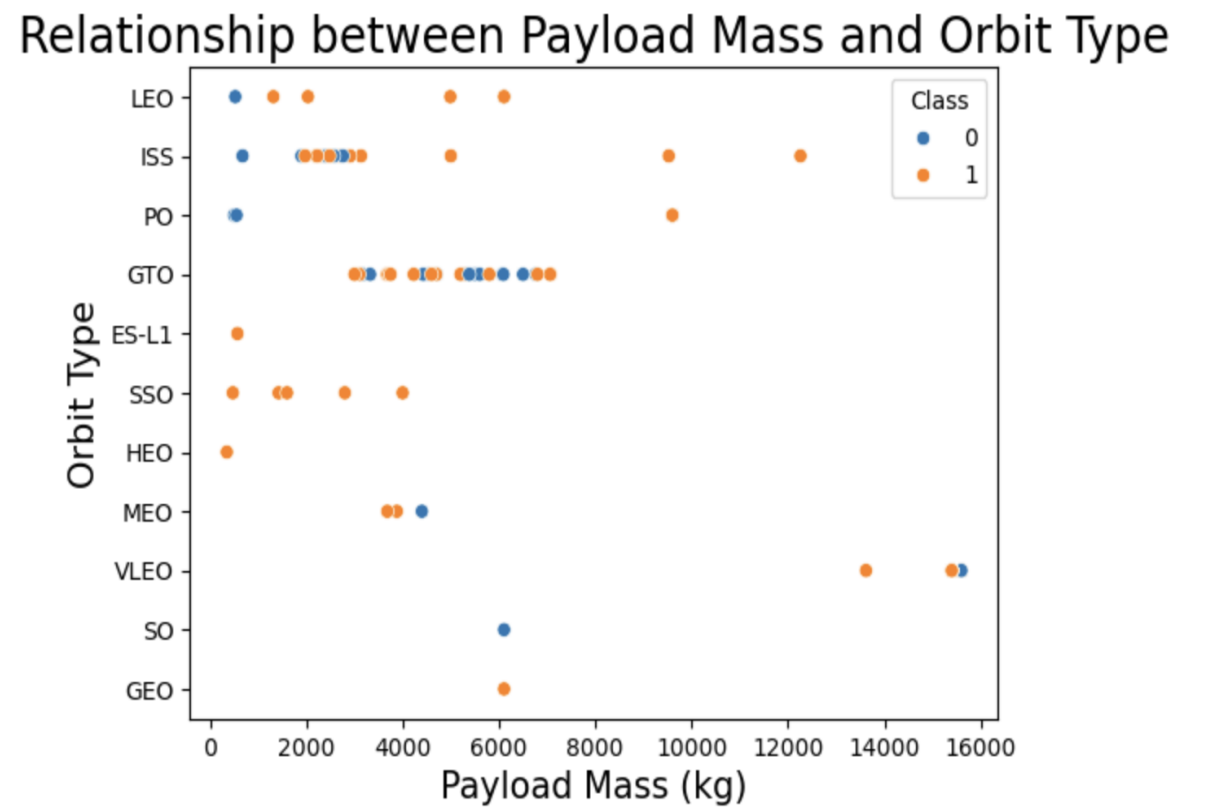
- **Polar, LEO, and ISS Orbits**: For missions with **heavier payloads**, there seems to be a higher success rate in **Polar**, **LEO**, and **ISS** orbits. These orbits show a clearer distinction between successful and unsuccessful missions, particularly as payload mass increases.

2. **GTO Orbit**:

- **No Clear Distinction**: In the **GTO** orbit, the plot shows a mix of successful and unsuccessful missions regardless of **Payload Mass**. This suggests that success or failure in GTO missions is more unpredictable, likely due to the higher complexity and challenges involved in these missions.

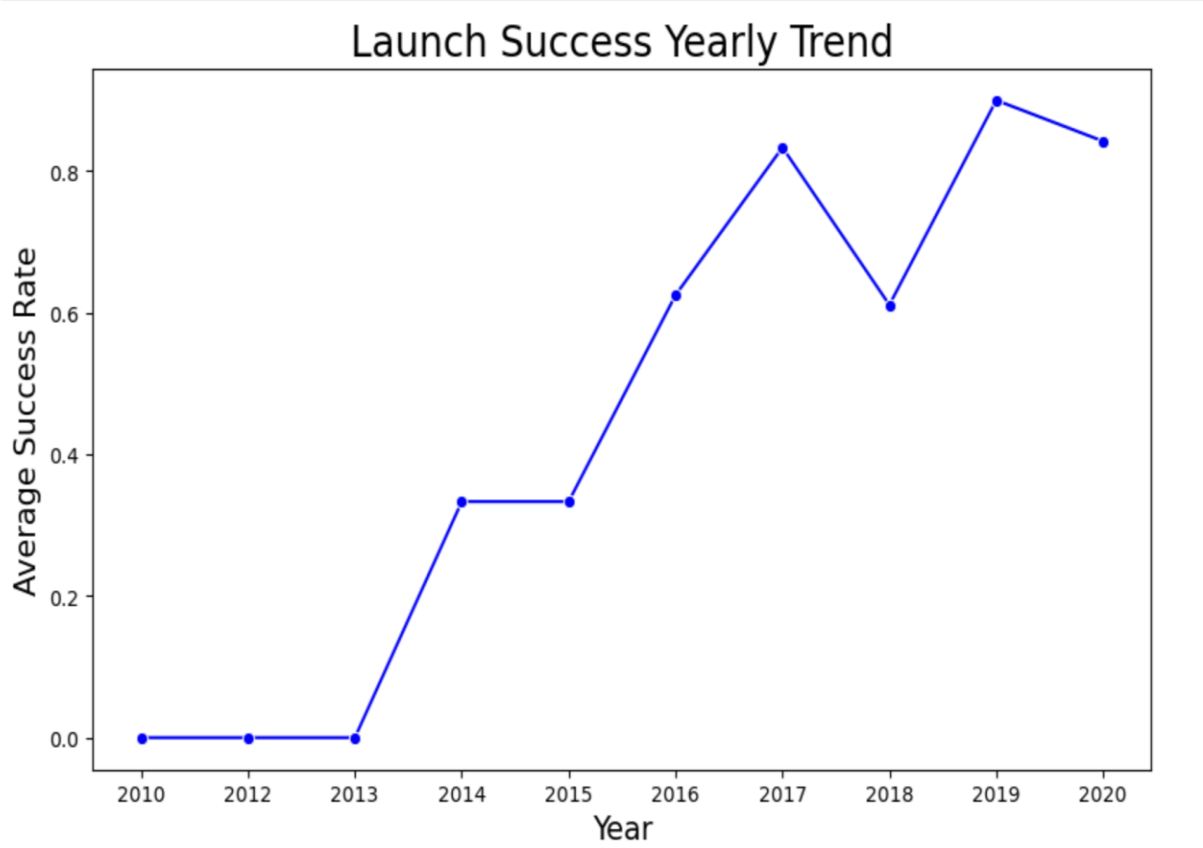
3. **General Observation**:

- **Payload Impact**: **Heavier payloads** seem to have a more predictable success rate in certain orbits like **Polar**, **LEO**, and **ISS**, while **GTO** missions may not show such clear patterns due to their complexity.



Insights from Launch Success Yearly Trend

- The line chart shows the average launch success rate over the years. Here are the key insights:
- Steady Increase: From 2013 to 2020, the success rate steadily increased, suggesting continuous improvement in technology, processes, and overall reliability of launches.
- Possible Reasons:
  - Technological Advancements: Ongoing advancements in space technology and innovation likely contributed to more successful missions.
  - Increased Experience: As companies gained more experience with space missions, their success rates improved.
- Launch Success Stability: The upward trend in success rates indicates growing confidence and consistency in space launches over time, especially after 2013.



All Launch Site Names

Unique launch sites:

SELECT DISTINCT Launch_Site FROM SPACEXTBL;

```
Unique Launch Sites:
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
```

There are four unique launch sites in the dataset, meaning that the dataset contains missions launched from four distinct locations. By extracting the unique values in the "Launch_Site" column, we can identify the different sites used for the missions.

Launch Site Names Begin with 'CCA'

- **Task**: **Find 5 records with launch sites that begin with `CCA`.**
- **Query**: **SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;**

```
('2010-06-04', '18:45:00', 'F9 v1.0  B0003', 'CCAFS LC-40', 'Dragon Spacecraft Qualification Unit', 0, 'LEO', 'Space
X', '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 c
heese', 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)', 'Succes
s', 'No attempt')
```

Explanation:**All these records are for launches that took place from the CCAFS LC-40 site, with various payloads to LEO (ISS) orbit. The common theme is that they are early missions for SpaceX involving Dragon spacecraft and various payloads, all with success and different causes of failure or no attempts in the secondary payload (parachute failure or no attempt).**

Total Payload Mass

Total payload carried by boosters from NASA.

```
SELECT SUM(PAYLOAD_MASS__KG_) as Total_Payload_Mass
    FROM SPACEXTBL
    WHERE Customer LIKE '%NASA (CRS)%';
```

	Total_Payload_Mass
0	48213

This output shows that between all the NASA CRS missions stored in the database, the total payload mass launched by SpaceX boosters is 48,213 kg.

Average Payload Mass by F9 v1.1

Average payload mass carried by booster version F9 v1.1

```
SELECT AVG(PAYLOAD_MASS__KG_) as Average_Payload_Mass
   FROM SPACEXTBL
   WHERE Booster_Version = 'F9 v1.1';
```

	Average_Payload_Mass
0	2928.4

The average payload mass carried by booster version F9 v1.1 is 2,928 kg.

First Successful Ground Landing Date

List the date when the first succesful landing outcome in ground pad was acheived.

```
SELECT MIN(Date) AS First_Successful_Landing
  FROM SPACEXTBL
  WHERE Landing_Outcome = 'Success (ground pad)';
```

	First_Successful_Landing
0	2015-12-22

The query successfully identifies December 22, 2015 as the date when the first successful landing on the ground pad by SpaceX occurred.

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.

```
SELECT Booster_Version
   FROM SPACEXTBL
   WHERE Landing_Outcome = 'Success (drone ship)'
   AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

	Booster_Version
0	F9 FT B1022
1	F9 FT B1026
2	F9 FT B1021.2
3	F9 FT B1031.2

The query successfully lists four booster versions that achieved a successful landing on a drone ship and carried a payload mass between 4000 kg and 6000 kg.

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failed mission outcomes

SELECT MISSION_OUTCOME, COUNT(*) as Total_Count FROM SPACEXTBL GROUP BY MISSION_OUTCOME;

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

The query helps summarize the total number of missions for each unique mission outcome, such as:

- **1 mission with failure during the flight.**
- **99 missions in total classified as successful (including those with unclear payload status).**

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass.

```
SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM
 SPACEXTBL);
```

	Booster_Version
0	F9 B5 B1048.4
1	F9 B5 B1049.4
2	F9 B5 B1051.3
3	F9 B5 B1056.4
4	F9 B5 B1048.5
5	F9 B5 B1051.4
6	F9 B5 B1049.5
7	F9 B5 B1060.2
8	F9 B5 B1058.3
9	F9 B5 B1051.6
10	F9 B5 B1060.3
11	F9 B5 B1049.7

This SQL query is used to retrieve the names of the booster versions that carried the maximum payload mass. The query uses a subquery to find the maximum payload mass and then selects the booster versions that correspond to this value.

2015 Launch Records

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015

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

- **Extract month names from the DATE column.**
- **Filter out records from 2015 where the landing outcome was a failure on a drone ship.**
- **Display the mission outcome, booster version, and launch site.**

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

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the dates 2010-06-04 and 2017-03-20, in descending order.

```
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;
```

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

This query effectively helps analyze landing success rates and failure trends over time, showing that drone ship landings had mixed results, while ground pad landings had fewer but more successful outcomes.

Section 3

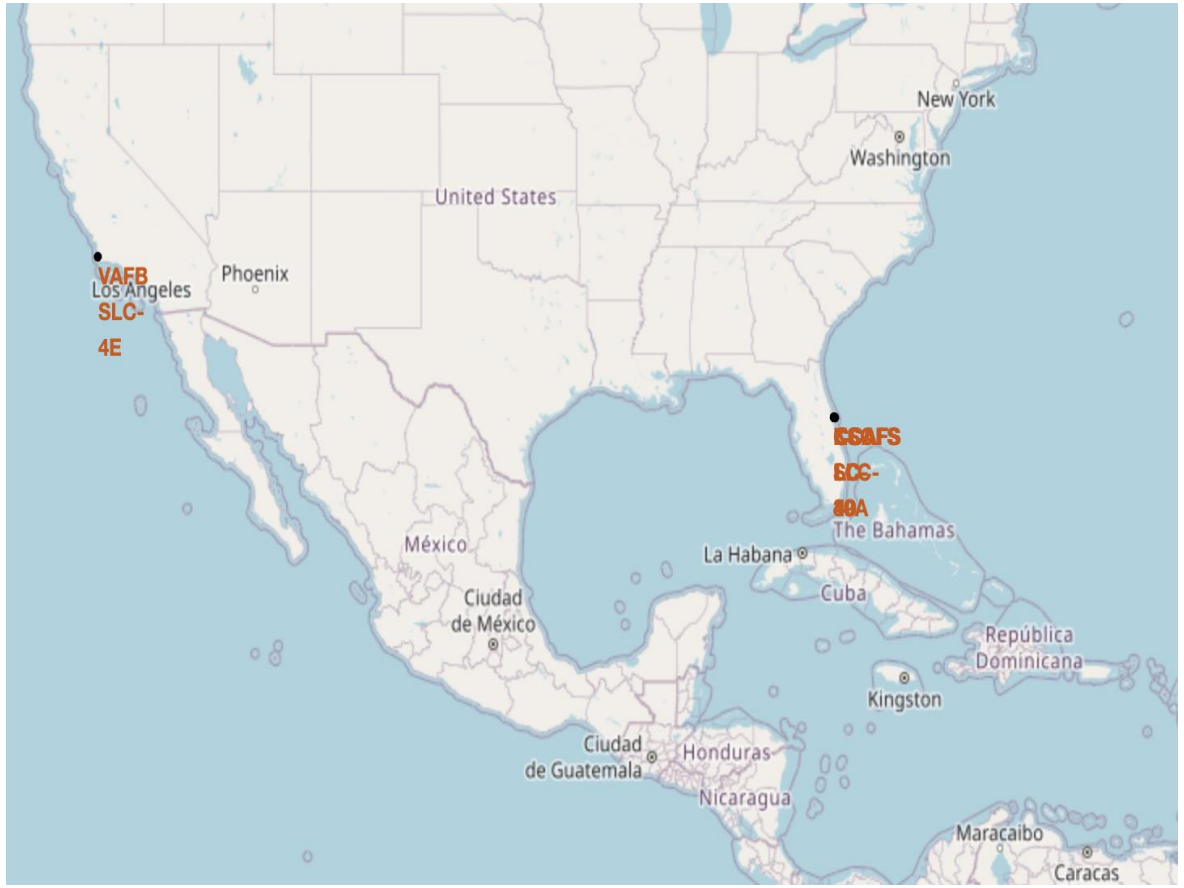
Launch Sites Proximities Analysis

Marking All Launch Sites on the Map

Most launch sites are located near coastlines for safety,allowing launches over water to minimize risk to populated areas

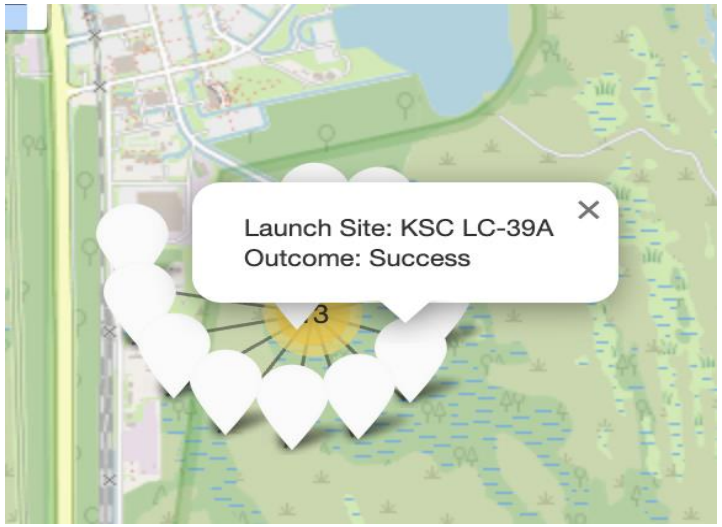
- **KSC LC-39A** and **CCAFS LC-40** are on Florida's east coast, near the Atlantic Ocean.
- **VAFB SLC-4E** is on California's west coast, near the Pacific Ocean.
- **Tanegashima Space Center** is on an island off Japan's southern coast, close to the Pacific Ocean.
- **Baikonur Cosmodrome** is inland in Kazakhstan, in a remote area far from populated regions.

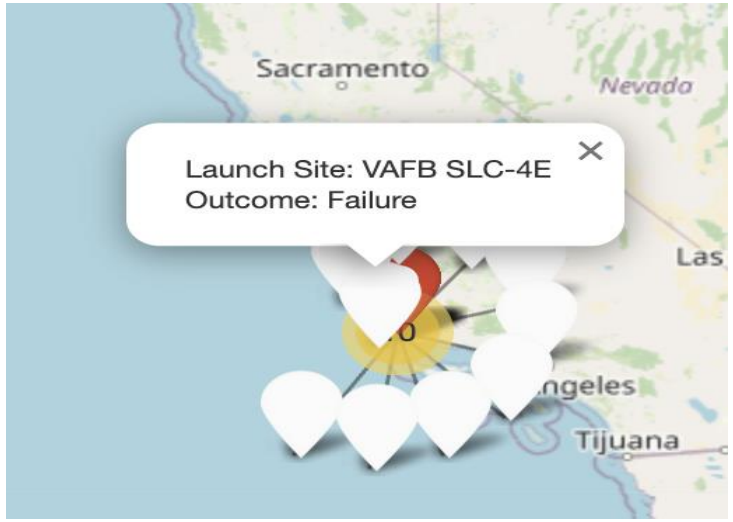
These sites are strategically chosen for safety and orbital considerations, not for proximity to the Equator.



Marking Success/Failed Launches for Each Site on the Map

1. Loading the dataset: We use the SpaceX launch data to create a map centered around the average coordinates of all launch sites.
2. Color coding: A function is created to assign a green marker for successful launches (class = 1) and a red marker for failed launches (class = 0).
3. Marker clustering: Given that multiple launches occur at the same site, we use MarkerCluster to group markers with identical coordinates, simplifying the map and reducing visual clutter.
4. Pop-up details: Each marker shows the launch site name and the outcome (success or failure) when clicked.



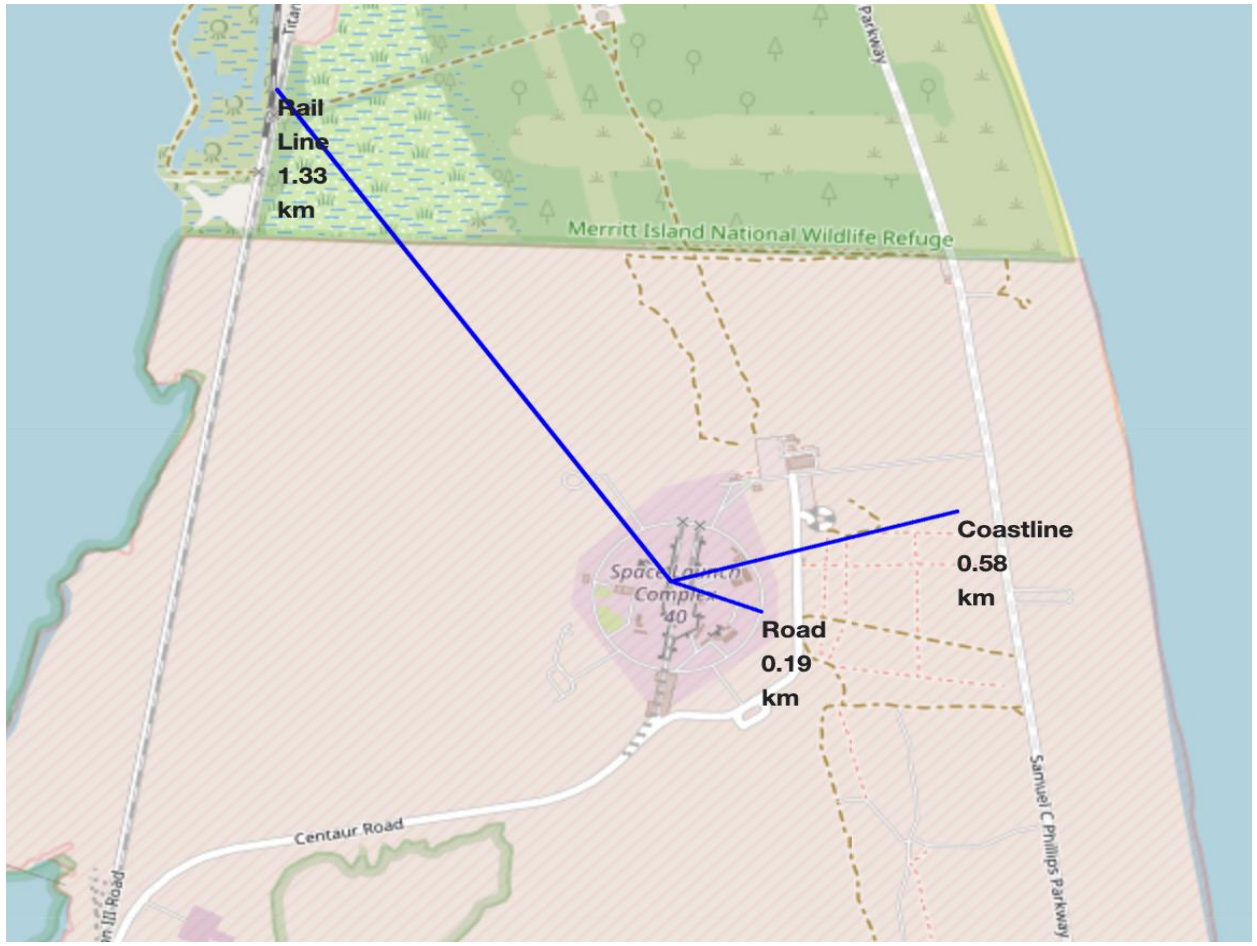


Calculate the distances between a launch site to its proximities

Analysis of Launch Site Proximities:

After calculating distances from the CCAFS LC-40 launch site, I found:

- **Coastline: 0.92 km away**
- **Railway Line: 1.33 km away**
- **Perimeter Road: 0.19 km away**

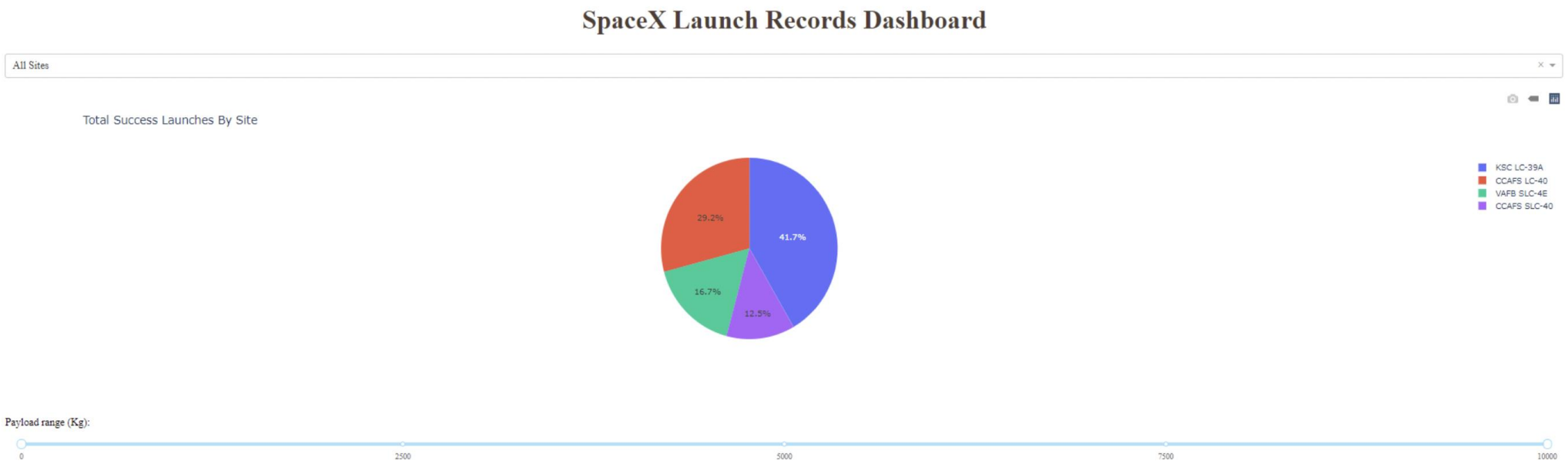


Section 4

Build a Dashboard with Plotly Dash

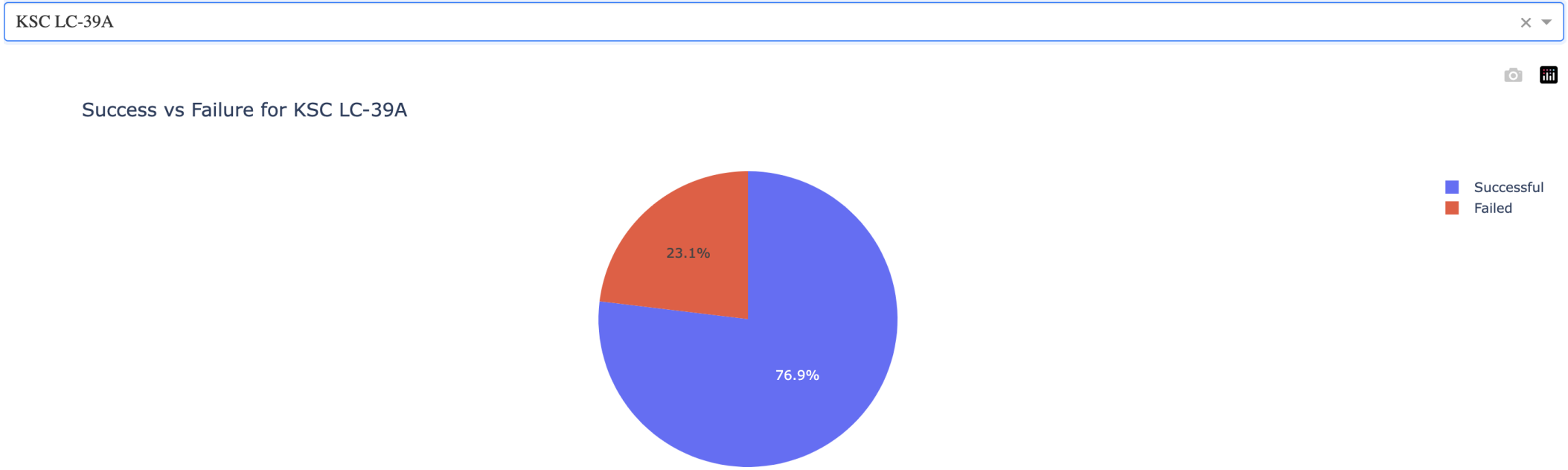
Launch Success Count for All Sites

- The dropdown menu enabled the selection of a specific launch site or all launch sites.
- When all launch sites were selected, the pie chart showed the distribution of successful Falcon 9 first-stage landing outcomes across the various launch sites.



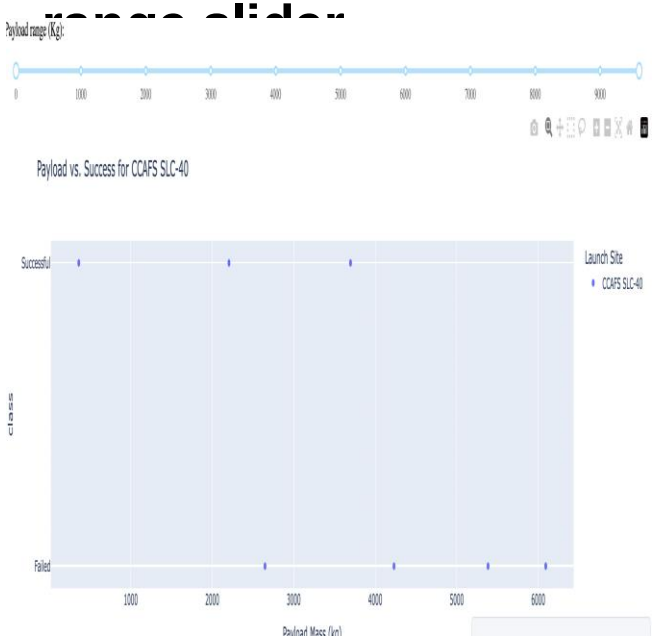
Launch Site with Highest Launch Success Ratio

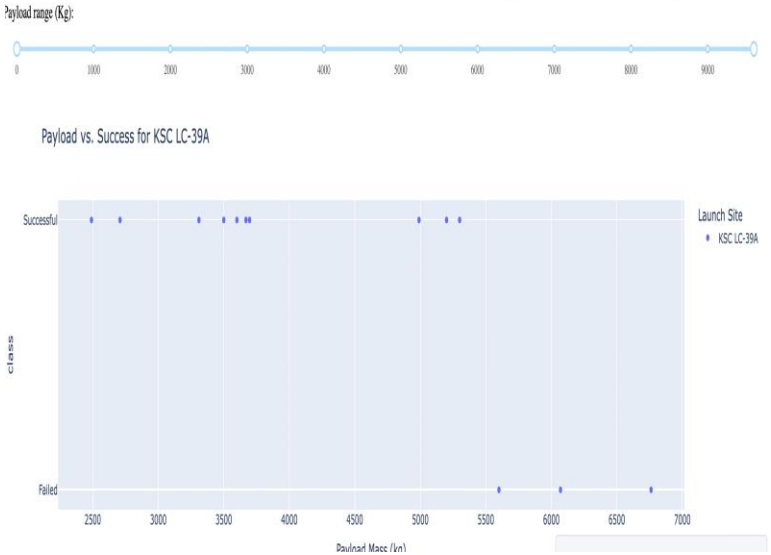
The pie chart shows the success and failure rates of launches from the KSC LC-39A launch site. 76.9% of the launches were successful, while 23.1% failed. The chart provides a clear visual representation of the launch outcomes.

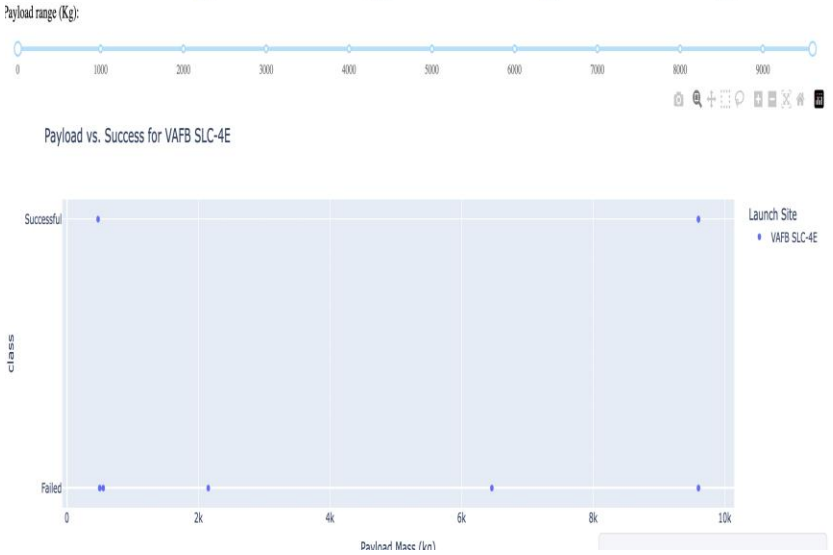


Payload vs. Launch Outcome

These screenshots are of the Payload vs. Launch Outcome scatter plots for all sites, with different payload selected in the range slider







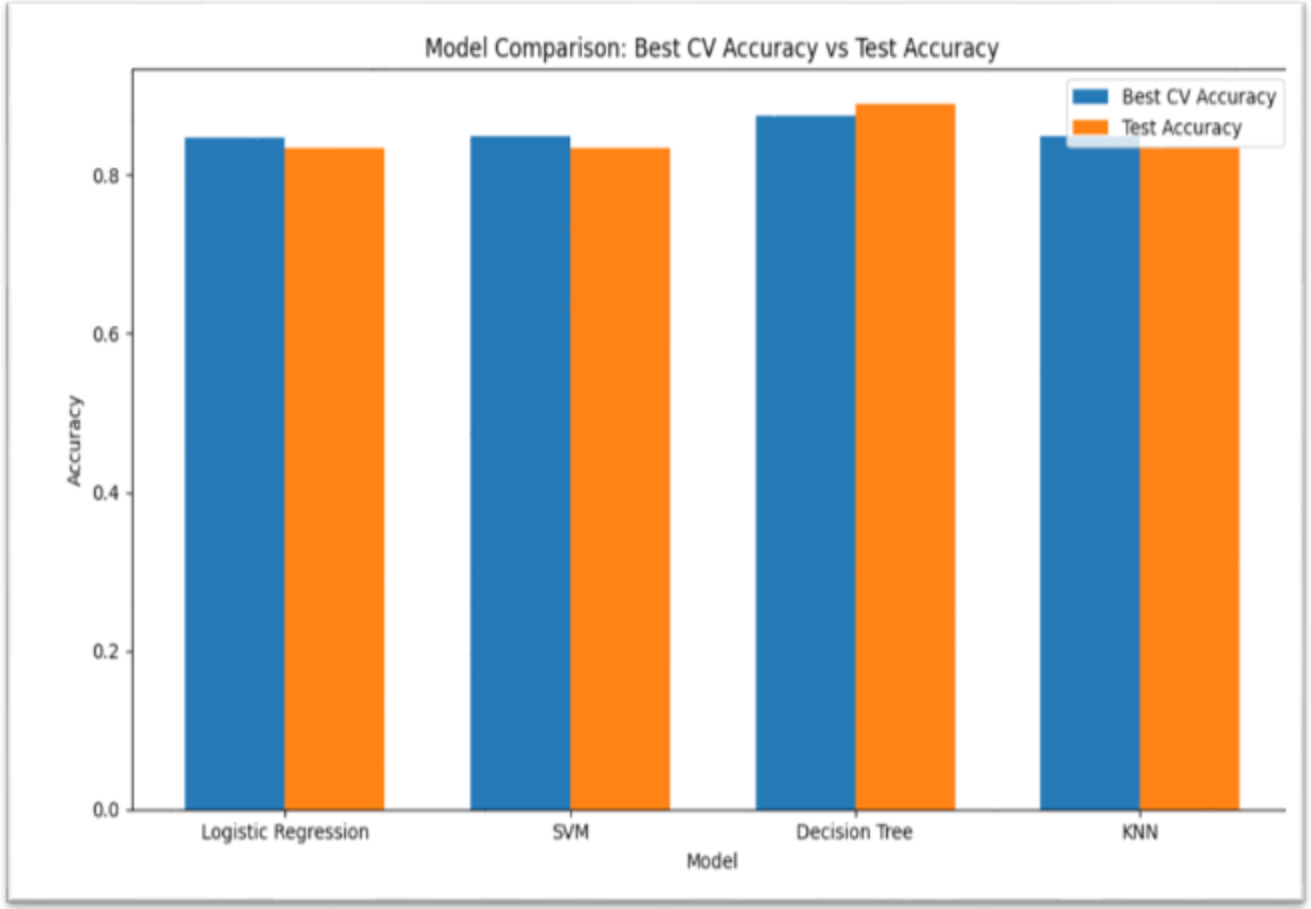
Section 5

Predictive Analysis (Classification)

Classification Accuracy

The performance of different models (Logistic Regression, SVM, Decision Tree, and KNN) was calculated by models (Logistic Regression, SVM, Decision Tree, and KNN) by plotting their best cross-validation (CV) accuracy and test accuracy on a bar chart.

- Bar 1: Represents the best accuracy achieved during cross-validation for each model.
- Bar 2: Represents the test accuracy (performance on unseen data) for each model.
- The chart helps to visually compare how well each model generalizes based on cross-validation versus its test set performance.

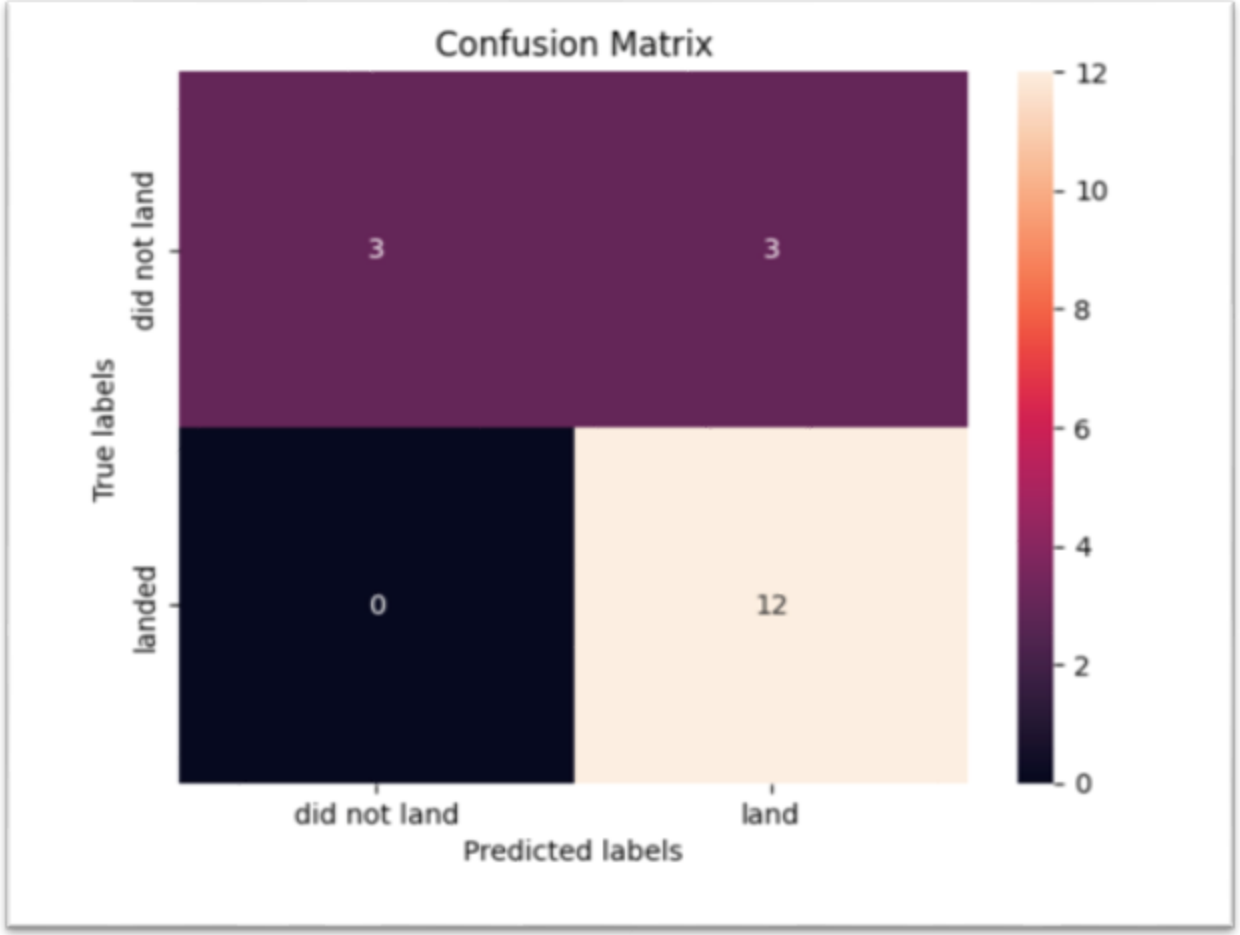


Confusion Matrix

This confusion matrix shows the model's performance in predicting rocket landings.

- True Positives (landed correctly predicted as landed): 12
- True Negatives (did not land correctly predicted as did not land): 3
- False Positives (did not land incorrectly predicted as landed): 3
- False Negatives (landed incorrectly predicted as did not land): 0

The model has a good performance, especially in predicting successful landings.



Conclusions

- **SpaceX's Falcon 9 first stage landing outcomes have shown significant improvement.**
- **A clear trend indicates increased success rates with more launches.**
- **Machine learning models offer reliable predictions for future Falcon 9 landing outcomes.**

Appendix

- Sources of Data:
- **Wikipedia (Webpage)-** https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- **SpaceX :-** https://api.spacexdata.com/v4/launches/past
- **CSV files(Github URL)-**
- https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/SPACEXTABLE.csv
- https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/Spacex.csv
- https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/data.csv
- https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/spacex_launch_dash%20.csv

Jupyter Notebook And Plotly Dashboard File

Github URL(DataCollection) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/Data%20Collection%20of%20SpaceX%20Falcon%209%20first%20stage%20Landing%20Prediction.ipynb

Github URL(Web Scraping) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/Web%20scraping%20Falcon%209%20and%20Falcon%20Heavy%20Launches.ipynb

Gihub URL (Data Wrangling) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/Data%20Wrangling%20Space%20X%20Falcon%209%20First%20Stage%20Landing%20Prediction%20.ipynb

Github URL(EDA with SQL) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/EDA%20with%20SQL.ipynb

Github URL(EDA with Data Visualisation) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/EDA%20with%20Visualization.ipynb

Github URL(Folium Maps)- https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/Visual%20Analytics%20with%20Folium.ipynb

Github URL(Dashboard File) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/spacex-dash-app%20(1).py

Github URL(Machine Learning) - https://github.com/Rajalaxmipradhan83/Applied-Data-Science-Capstone/blob/main/Machine%20Learning%20Prediction.ipynb

Thank you!