IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

Methodologies:

- **Data Collection & Wrangling:**
  - Collected SpaceX launch data via APIs and web scraping techniques.
  - Cleaned, merged, and preprocessed the data using Python (e.g., pandas) to build a robust dataset for analysis.

- **Exploratory Data Analysis (EDA):**
  - Applied SQL queries and descriptive statistics to understand data trends and distributions (see jupyter-labs-eda-sql file).
  - Developed interactive visualizations and charts to reveal key patterns, such as the relationship between orbit types, flight numbers, and launch success (refer to edadataviz).

- **Interactive Visual Analytics:**
  - Created interactive maps with Folium to visualize launch site locations.
  - Built dashboards (using Plotly Dash) to provide a dynamic overview of geospatial and temporal insights.

- **Predictive Analysis:**
  - Developed machine learning classification models to predict launch outcomes, evaluating key performance metrics (detailed in the SpaceX_Machine Learning Prediction module).

Results:

- **Data Readiness:**
  - Successfully consolidated a clean and comprehensive dataset that enabled multifaceted analysis.

- **Key EDA Insights:**
  - Identified trends such as higher success rates associated with specific orbit types and flight numbers.
  - SQL-driven insights helped uncover outliers and deeper trends within the data.

- **Visual Analytics Impact:**
  - Interactive maps and dashboards provided clear, actionable geospatial insights on launch site performance.
  - Visualizations made complex data trends accessible to peer data scientists.

- • **Predictive Performance:**
  - The classification models demonstrated competitive performance in predicting launch outcomes, highlighting critical predictive variables.
  - These results suggest promising directions for refining operational strategies based on data-driven insights.

Introduction

- A brief overview of the project, emphasizing that it uses SpaceX launch data to drive insights into launch performance.
  • The data sources and methods used (API collection, web scraping, SQL-based analysis, and interactive visualizations).
  • The broader context of why analyzing launch performance is relevant—touch on operational efficiency, industry competitiveness, and data-driven decision-making in aerospace.
  • A clear list of research questions or problems, such as:
  – What are the key factors (e.g., orbit type, flight number) that influence launch success?
  – What patterns and trends emerge from the historical data?
  – How can interactive visual tools (like maps and dashboards) reveal geospatial and temporal trends?
  – Can predictive models effectively forecast launch outcomes based on the available data?
  • A statement on the overall objective: to uncover actionable insights and improve understanding of the variables driving SpaceX launch performance.

Section 1

Methodology

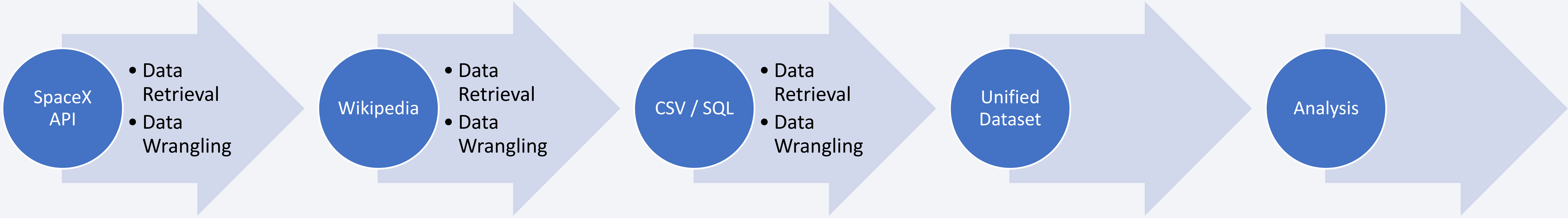
Methodology

Executive Summary

- Data collection methodology:
  - Used APIs and web scraping to gather SpaceX launch data from multiple sources
  - Incorporated additional references (e.g., Wikipedia) for supplementary information
- Perform data wrangling
  - Cleaned and merged data (removing duplicates, correcting data types)
  - Ensured consistency and integrity for further analysis
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Executed SQL queries to filter, group, and aggregate data (e.g., success rates by orbit type)
  - Created bar charts, scatter plots, and histograms to reveal key trends and distributions
- Perform interactive visual analytics using Folium and Plotly Dash
  - Mapped launch sites and distances using Folium for geospatial insights
  - Built interactive dashboards with Plotly Dash to dynamically explore variables and outcomes
- Perform predictive analysis using classification models
  - Selected features (e.g., orbit type, payload mass) to predict launch success
  - Applied classification algorithms (e.g., Logistic Regression, Decision Trees)
  - Tuned hyperparameters and evaluated performance (accuracy, precision, recall, F1-score)

Data Collection

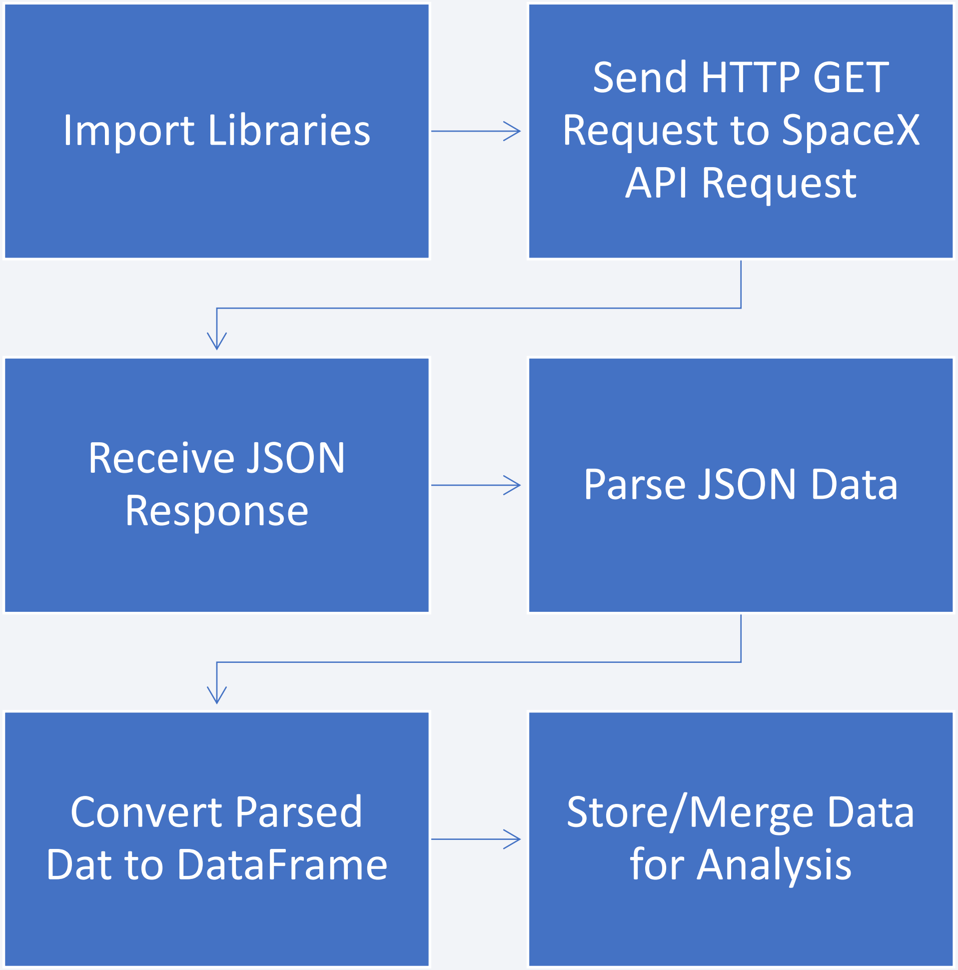
- **How Data Sets Were Collected**
  • **SpaceX API** – Retrieved flight numbers, rocket types, payload details, and launch site information.
  • **Web Scraping** – Used Requests and BeautifulSoup to gather supplementary data from Wikipedia pages.
  • **CSV/SQL Integration** – Merged local CSV files or SQL databases to consolidate historical records and ensure data completeness.



Data Collection – SpaceX API

Data Collection with SpaceX REST Calls

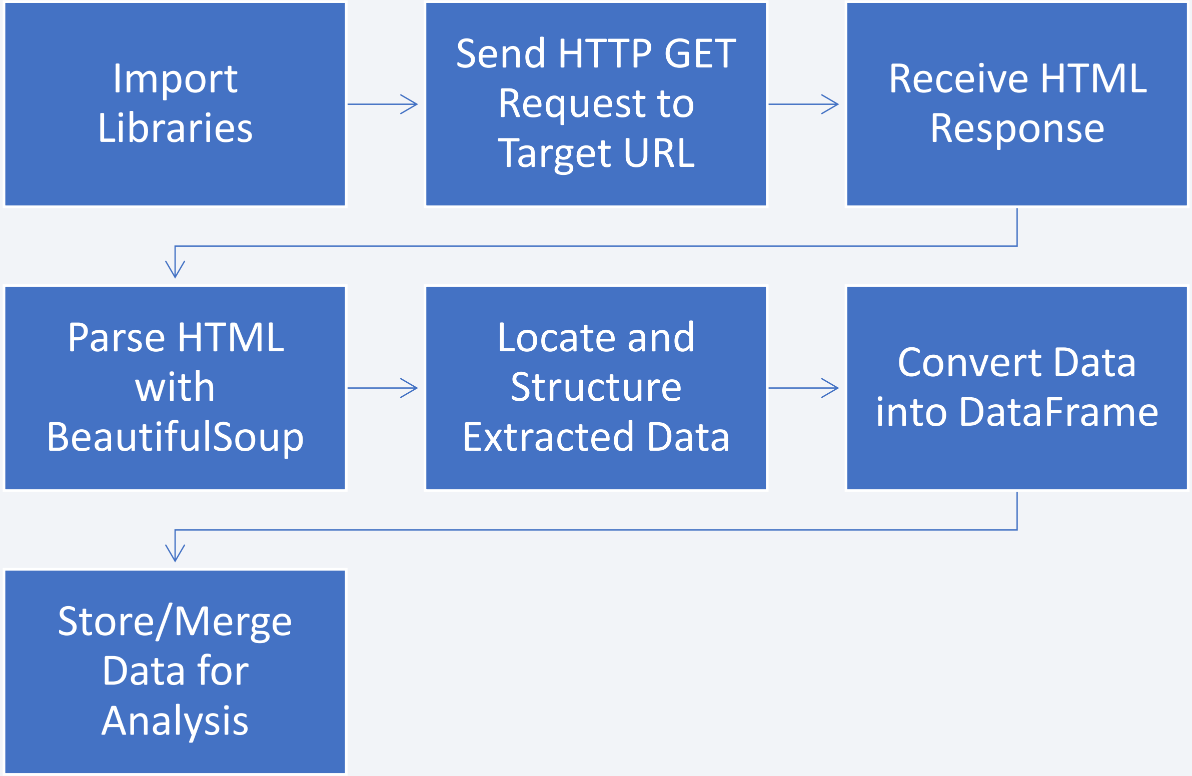
- **REST Endpoint** – Accessed SpaceX's public API to retrieve mission data (flight numbers, rocket names, payloads).
- **HTTP GET Request** – Used Python's requests library to call the endpoint and receive data in JSON format.
- **JSON Parsing** – Converted the JSON response into structured data (pandas DataFrame) for analysis.
- **Data Integration** – Merged API results with other sources (e.g., CSV files, Wikipedia data) to build a unified dataset.
- **Github Link:** https://github.com/rdtcamingawan/coursera/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

Web Scraping Process

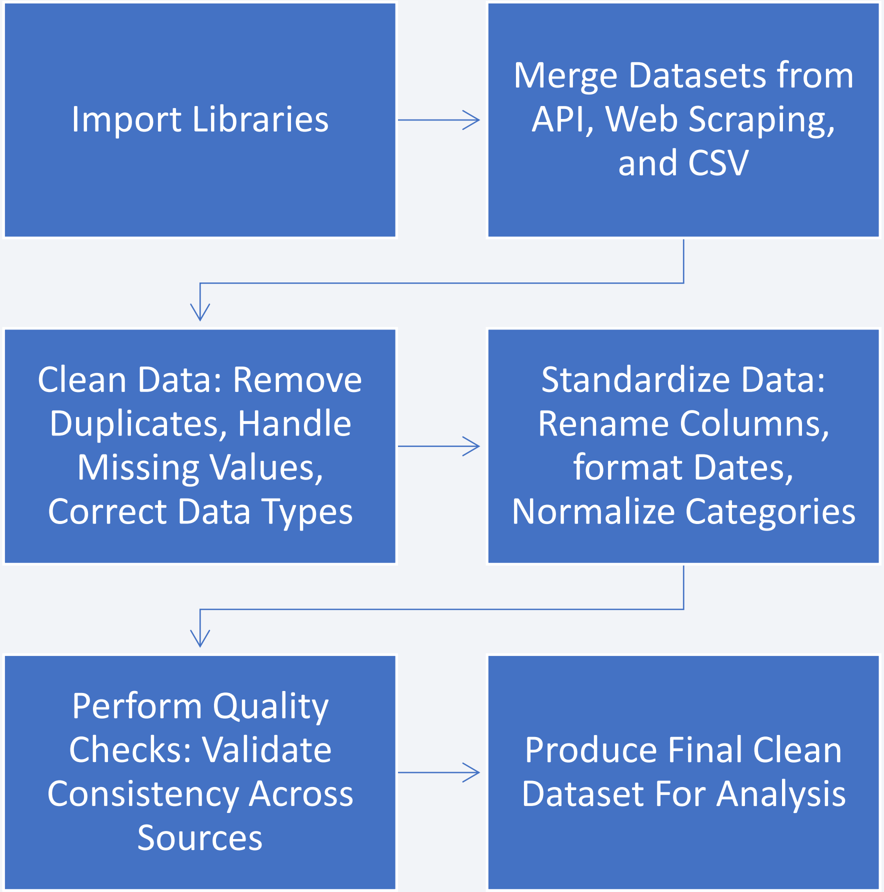
- **Requests Library** – Sent HTTP GET requests to Wikipedia pages containing SpaceX launch information.
- **HTML Parsing** – Used BeautifulSoup to parse the returned HTML structure (tags, attributes, text).
- **Data Extraction** – Selected relevant tables and text fields (e.g., launch dates, rocket names) by inspecting the HTML elements.
- **Data Cleaning** – Removed unnecessary formatting, handled missing values, and standardized column names.
- **Data Storage** – Converted the extracted data into a pandas DataFrame and stored or merged it with other data sources.
- **Github Link:** https://github.com/rdtcamingawan/coursera/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

Data Wrangling Process

- **Data Integration** – Merged datasets from SpaceX API, Wikipedia scraping, and local CSV files.
- **Data Cleaning** – Removed duplicates, handled missing values, and corrected data types (e.g., converting strings to numeric).
- **Data Standardization** – Renamed columns for consistency, unified date/time formats, and normalized categorical labels.
- **Quality Checks** – Verified data accuracy and consistency by cross-referencing multiple sources.
- **Final Dataset** – Produced a clean, structured DataFrame ready for EDA and predictive modeling.
- **Github Link:** https://github.com/rdtcamingawan/coursera/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

Charts and Their Purpose:

• **Bar Charts** – Compare categories (e.g., orbit types vs. success rates).
• **Scatter Plots** – Show relationships between numerical variables (e.g., flight number vs. payload mass).
• **Histograms** – Reveal the distribution of continuous data (e.g., payload mass).
• **Line Plots** – Track trends over time in launch performance.

- **Github Link: https://github.com/rdtcamingawan/coursera/blob/main/edadataviz.ipynb**

EDA with SQL

SQL Queries:

- Selected launch records filtered by success and failure.
- Aggregated data by orbit types and launch sites to compute counts and percentages.
- Joined multiple tables to combine launch details with auxiliary information.
- Grouped results by flight number to analyze trends over time.
- Applied sorting and limiting to identify top-performing categories.

Github Link: https://github.com/rdtcamingawan/coursera/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Map Objects Added

- **Marker**
  - Placed on each launch site to pinpoint its exact location.
  - Used to provide a clear, clickable reference for launch site information.
- **Circle/CircleMarker**
  - Overlaid around each launch site to visualize proximity or potential impact radius.
  - Helped highlight areas within a certain distance of the launch pad.
- **Line/Polyline**
  - Drawn to measure and display distances (e.g., from a reference point like the coastline or a city to the launch site)
  - Allowed for a visual representation of travel paths or direct connections.

Reasons for Adding These Objects

• **Markers** – Offer a simple, intuitive way to identify and access details about each launch site.

• **Circles/CircleMarkers** – Emphasize the geographic scope around a site, aiding in proximity-based analyses.

• **Lines/Polylines** – Illustrate distances or routes, enabling quick visual understanding of spatial relationships.

Github Link: https://github.com/rdtcamingawan/coursera/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Plots/Graphs & Interactions in the Dashboard

- **Scatter Plot (Success vs. Payload Mass)**
  - Interactively filter by launch site or orbit type.
  - Helps reveal how payload size correlates with launch success under different conditions.
- **Bar Chart (Launch Success Counts)**
  - Provides a dropdown or radio button to select specific launch sites.
  - Quickly compares success rates across multiple locations.
- **Interactive Callbacks**
  - Update plots dynamically based on user inputs (e.g., site selection, orbit category).
  - Enables real-time data exploration without manually re-running code.

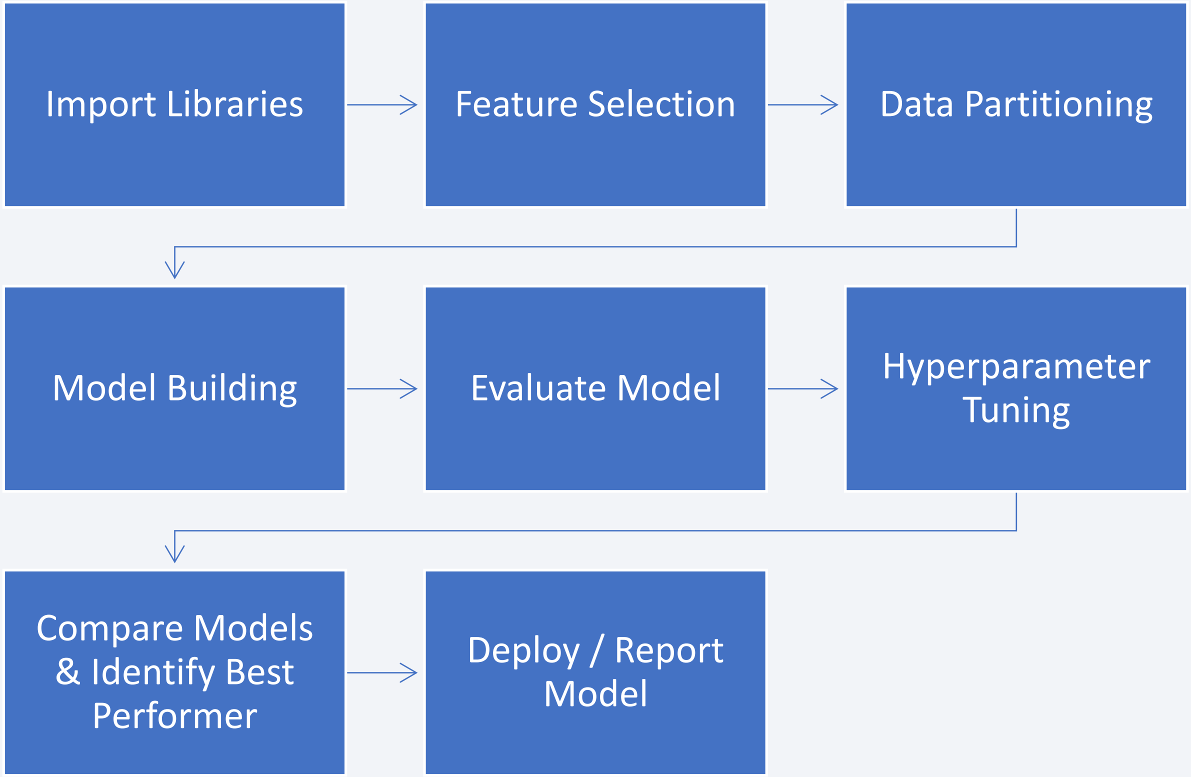
- **Why These Plots & Interactions Were Added**
- **Insight Discovery** – Users can isolate specific variables (payload range, orbit type, etc.) to see direct impacts on success rates.
- **User Engagement** – Interactive elements encourage deeper exploration, making the analysis more intuitive and revealing hidden patterns.
- **Immediate Feedback** – Real-time updates allow quick hypothesis testing (e.g., “Does a higher payload always reduce success?”) without leaving the dashboard.

Predictive Analysis (Classification)

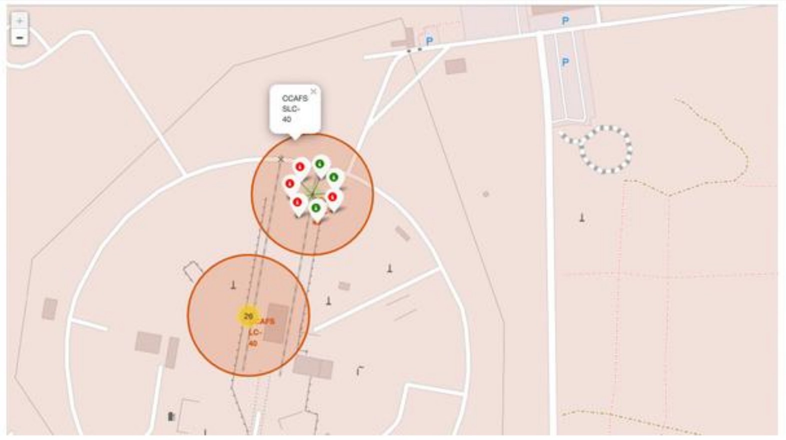
Model Development Process

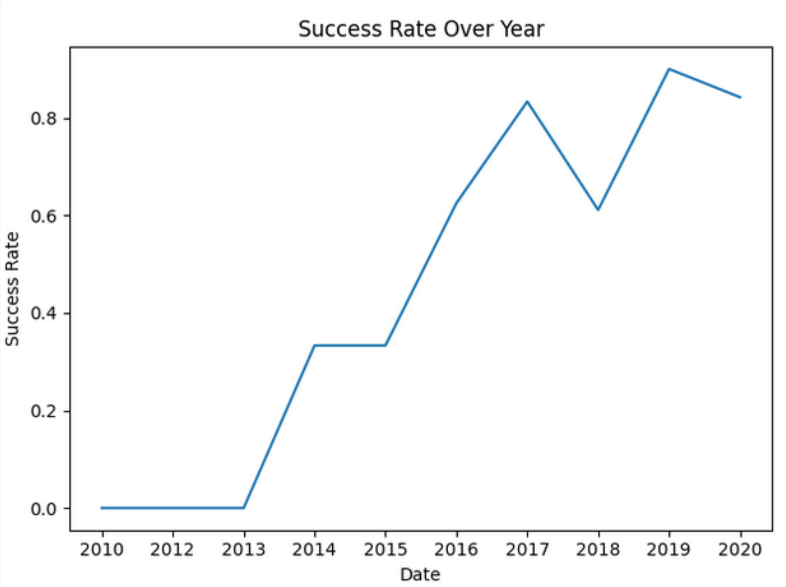
- **Feature Selection** – Identified critical variables (e.g., orbit type, payload mass) that influence launch success.
- **Data Partitioning** – Split data into training and testing sets to ensure unbiased evaluation.
- **Model Building** – Trained multiple classification algorithms (e.g., Logistic Regression, Decision Tree).
- **Evaluation Metrics** – Used accuracy, precision, recall, and F1-score to gauge model performance.
- **Hyperparameter Tuning** – Employed grid or random search methods to refine parameters (e.g., max depth for Decision Trees).
- **Best Model Identification** – Compared evaluation metrics across models and selected the highest-performing one.

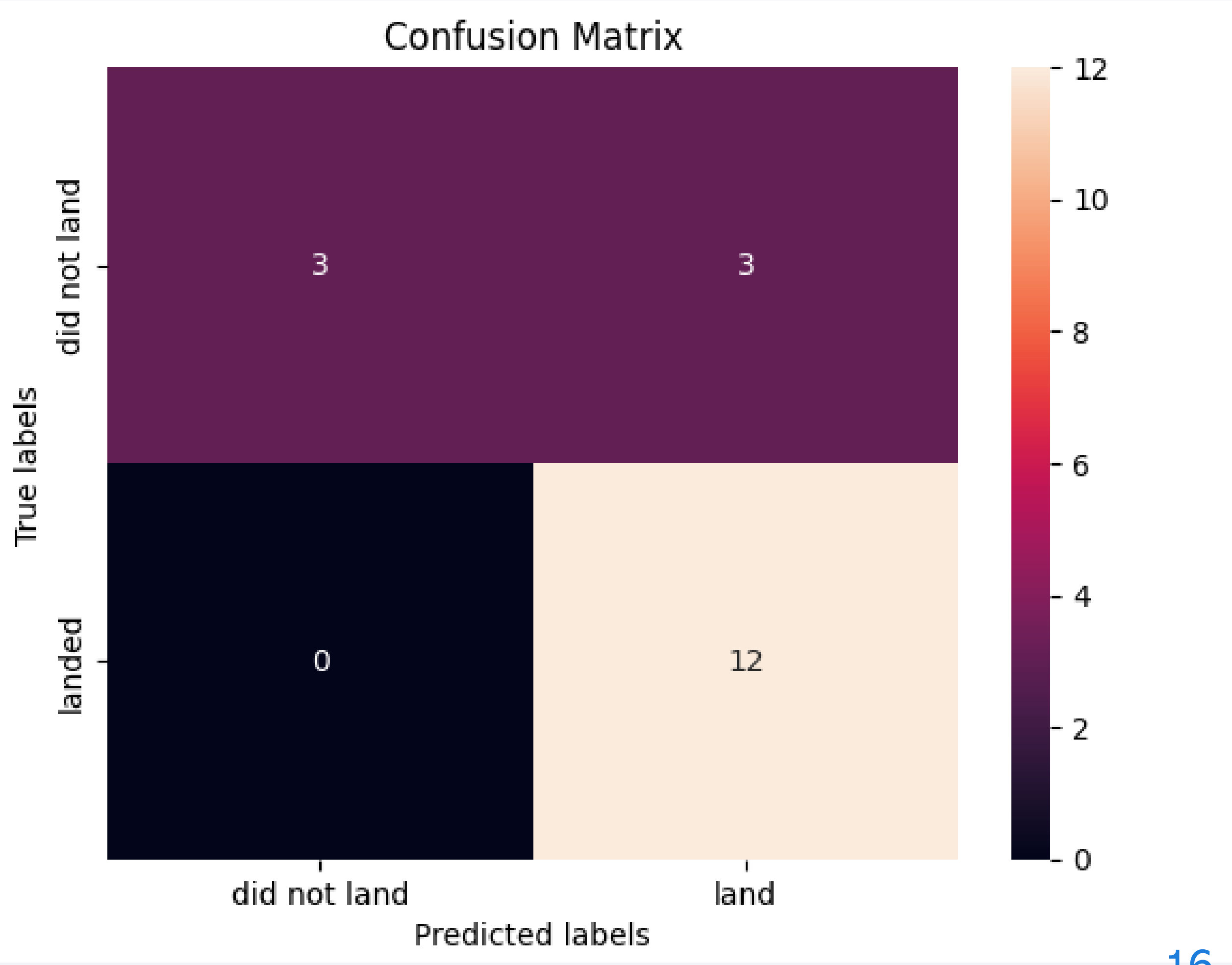
Github Link: https://github.com/rdtcamingawan/coursera/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



Results




Success Rate Over Year
Success Rate
0.0
0.2
0.4
0.6
0.8
2010
2012
2013
2014
2015
2016
2017
2018
2019
2020
Date


Confusion Matrix
True labels
did not land
landed
3
3
0
12
did not land
land
Predicted labels
0
2
4
6
8
10
12

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

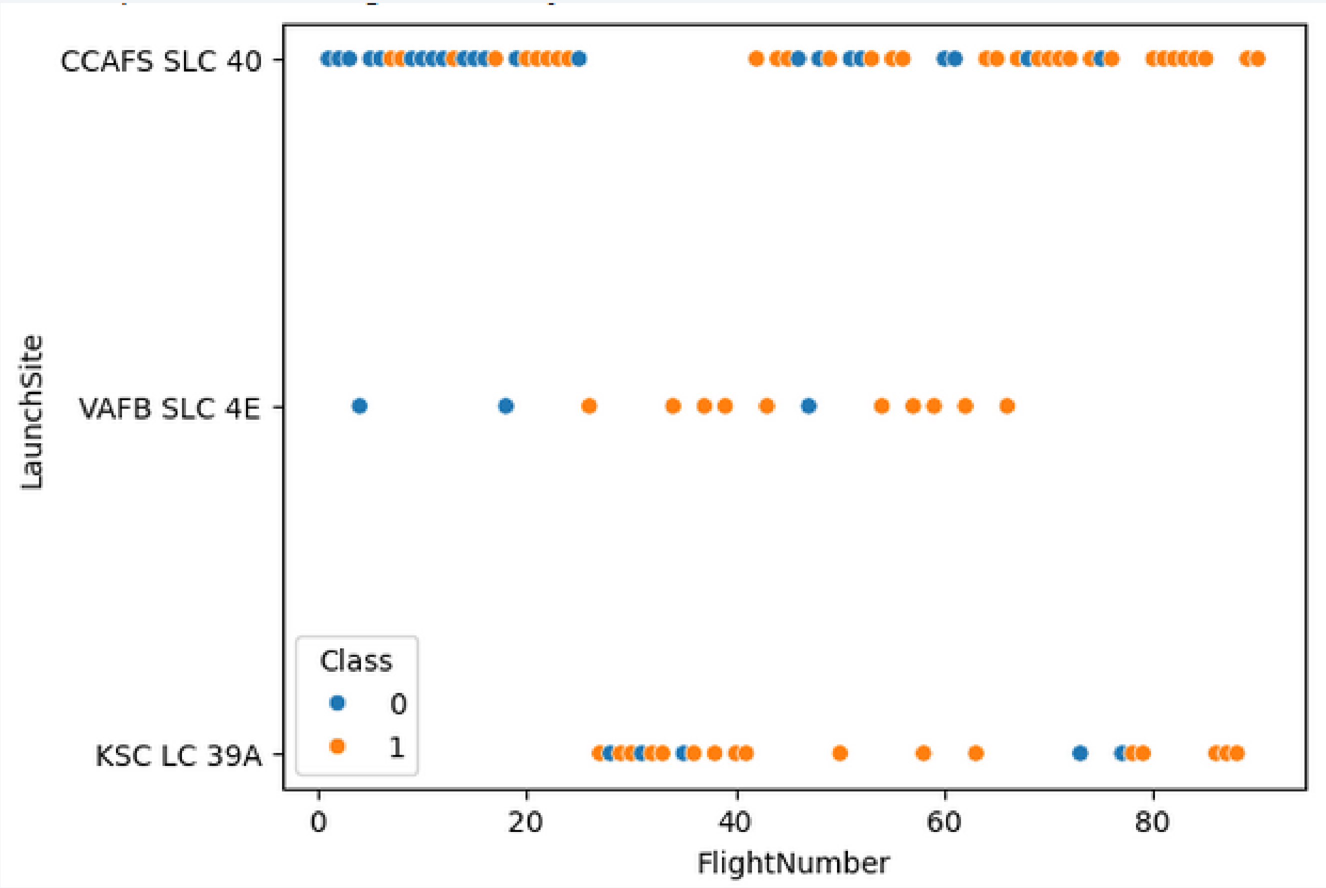
The scatter plot of flight number versus launch site displays each flight as a point, with the flight number on one axis (typically the x-axis) and the corresponding launch site on the other (usually the y-axis). This visualization helps reveal:

• **Distribution Patterns:** It shows how flight numbers are distributed across different launch sites, indicating which sites are used more frequently.

• **Usage Trends:** You can observe trends or clusters, such as a site having a series of consecutive flight numbers, suggesting consistent usage.

• **Outliers or Anomalies:** Any irregular spacing or unexpected flight numbers at a particular site become apparent, which may indicate unique events or operational changes.

Overall, the plot provides insight into the operational cadence and allocation of flights among various launch sites.

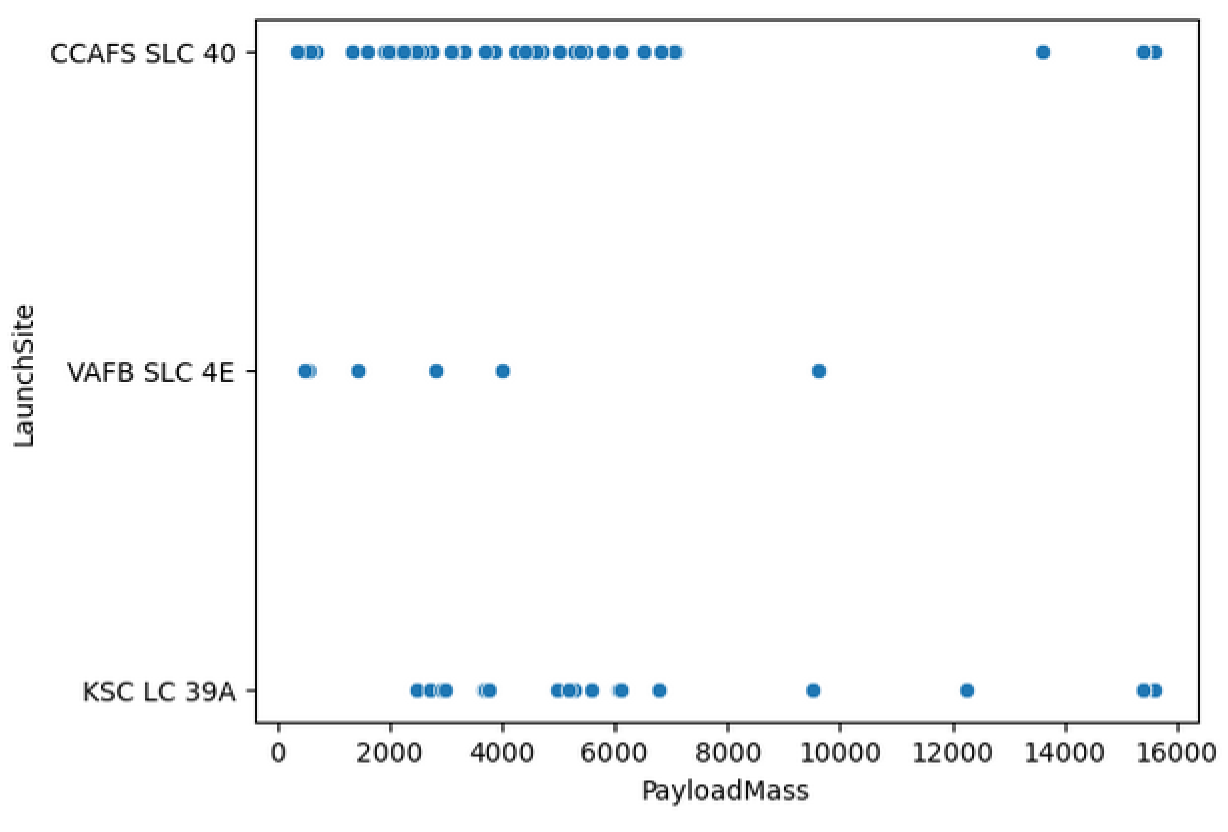


Payload vs. Launch Site

The plot of payload versus launch site typically displays payload masses on the y-axis against different launch site categories on the x-axis. This visualization helps reveal:

- **Payload Distribution:** How payload masses vary across different launch sites.
- **Site-Specific Trends:** Whether certain launch sites consistently handle heavier or lighter payloads.
- **Outliers:** Any unusually high or low payload values that might indicate special missions or data anomalies.

Overall, it provides insight into the operational capabilities and characteristics of each launch site



Success Rate vs. Orbit Type

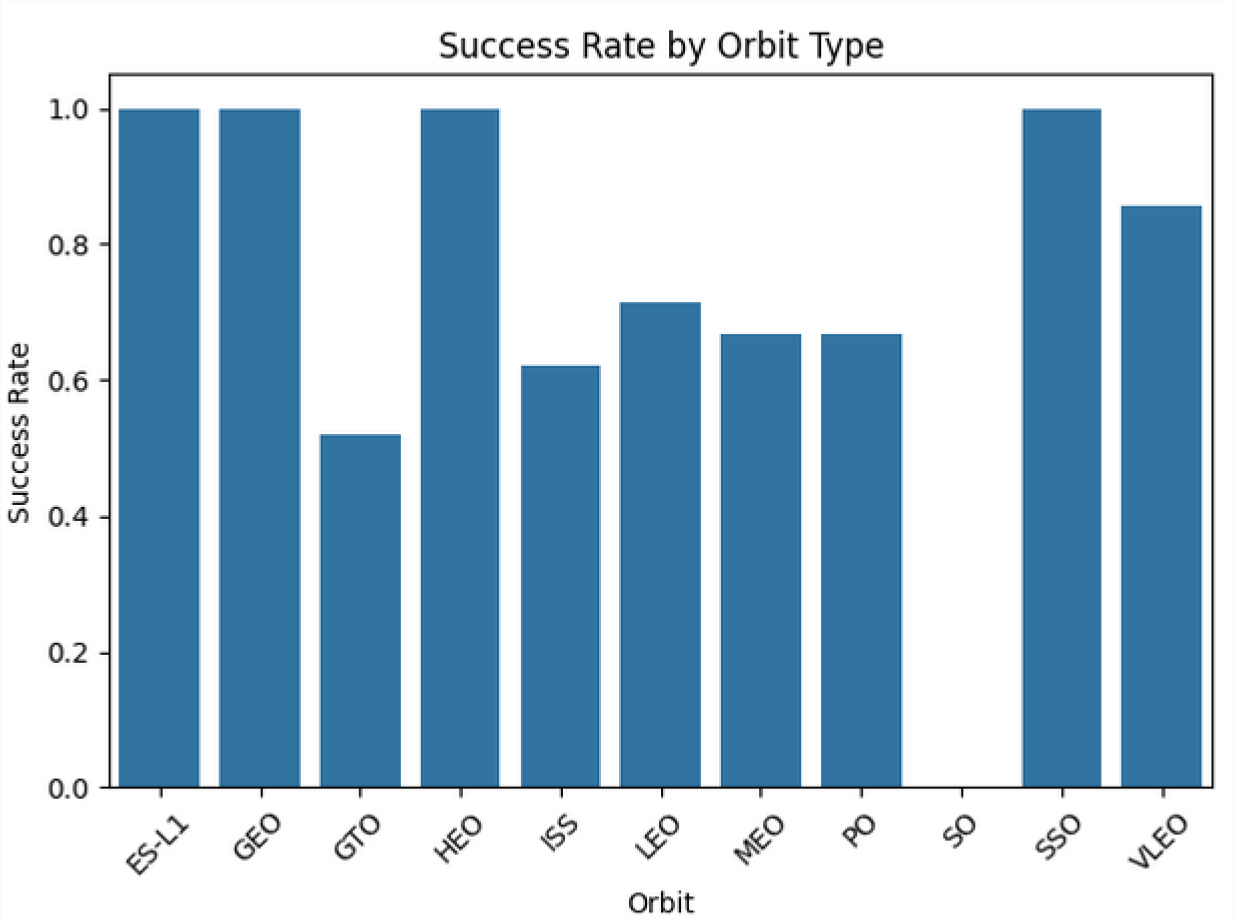
The bar chart of success rate versus orbit type reveals several key insights:

• **SO Orbit:** It appears that the SO category shows no recorded success, which might indicate either a lack of data or consistently unsuccessful missions for that orbit.

• **GTO Orbit:** The chart indicates that GTO has the lowest success rate among the orbit types, suggesting potential challenges or higher risk associated with these missions.

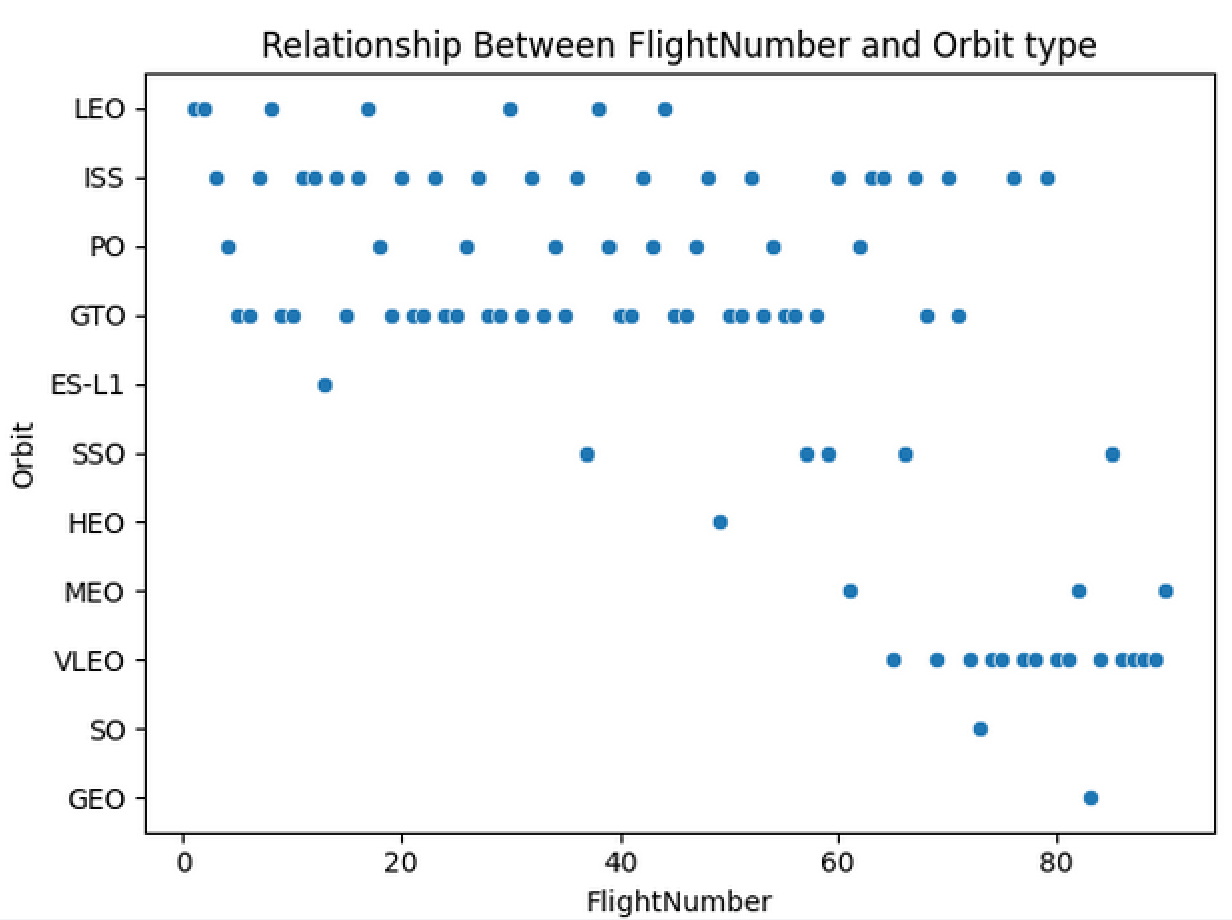
• **High Success Orbits:** ES-L1, GEO, HEO, and SSO display the highest success rates, highlighting these orbits as more reliable or perhaps more frequently executed under optimal conditions.

These observations help in understanding how different orbital targets correlate with mission outcomes and can guide further investigation into the operational nuances behind these success rates.



Flight Number vs. Orbit Type

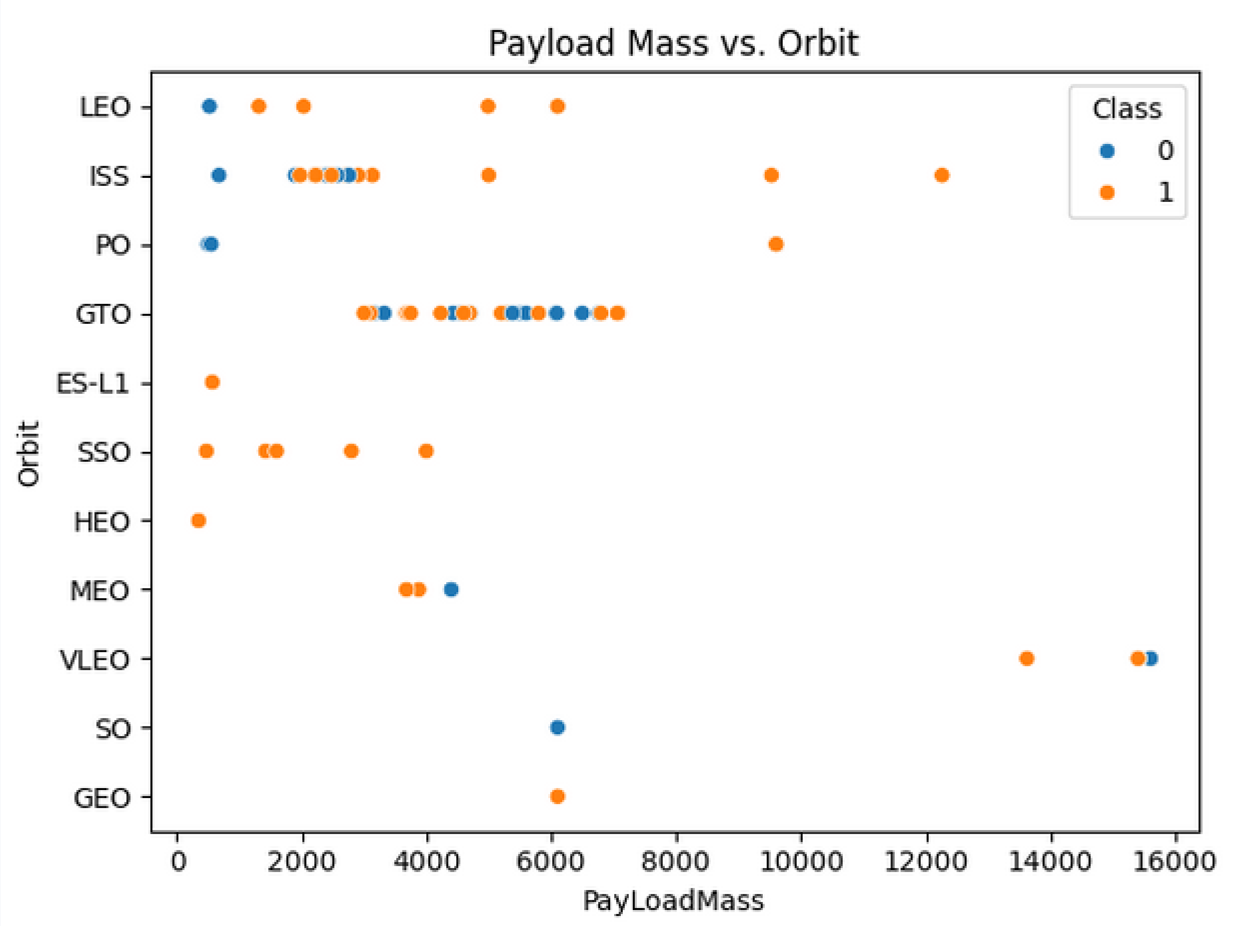
The chart shows that ES-L1 has only one recorded flight, indicating it's a rare or specialized mission target, while orbit types like LEO, ISS, PO, GTO, and VLEO have many flight numbers, reflecting their more frequent use in launch operations.



Payload vs. Orbit Type

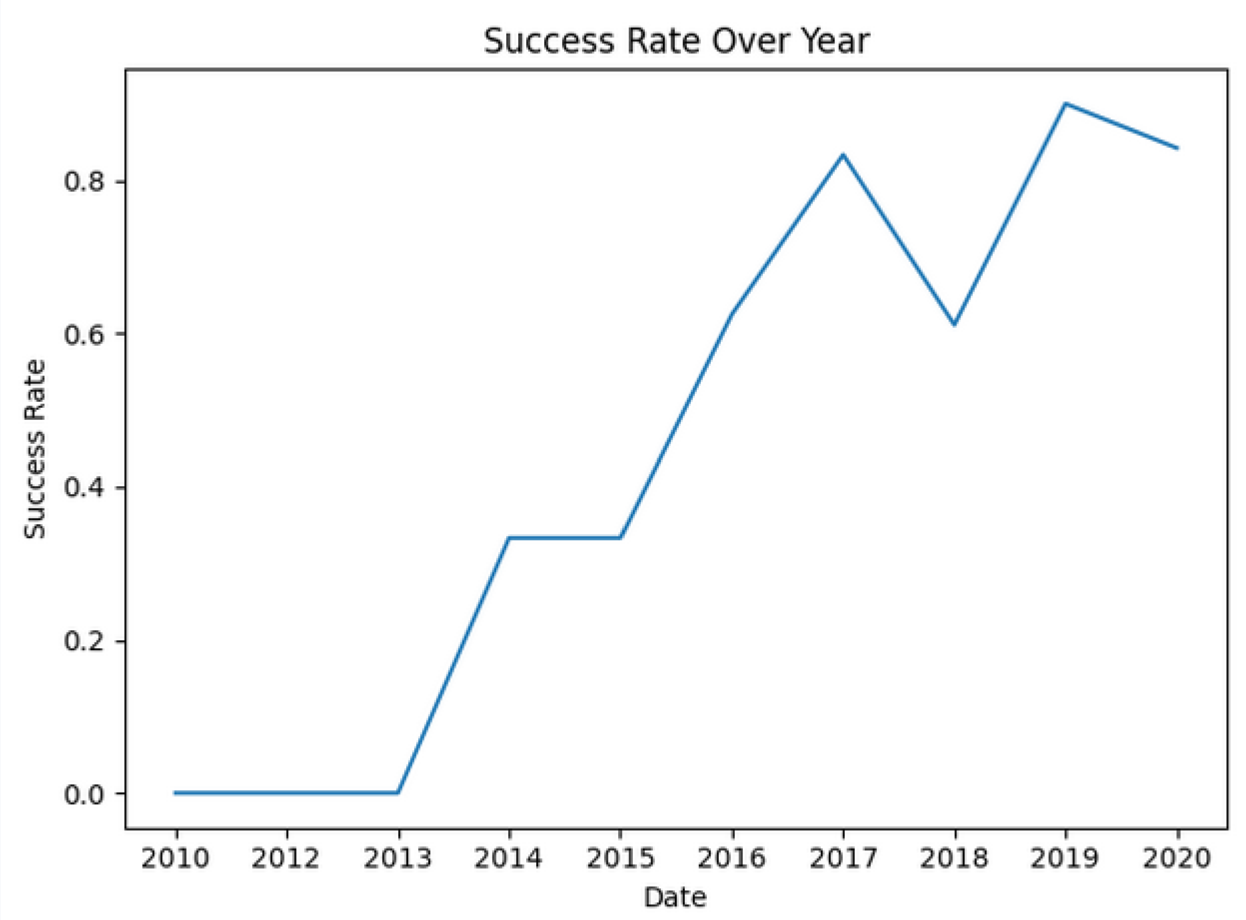
With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

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



Launch Success Yearly Trend

The line plot of the launch success yearly trend shows that since 2013, the success rate has been steadily increasing, reaching its peak around 2019. This upward trend indicates consistent improvements in mission reliability over time, reflecting advancements in launch technology and operational procedures.



All Launch Site Names

Query Result
- CCAFS LC-40
- VAFB SLC-4E
- KSC LC-39A
- CCAFS SLC-40

Short Explanation
These four entries represent the distinct launch sites recorded in the dataset. Two are located at Cape Canaveral Air Force Station (CCAFS), one at Vandenberg Air Force Base (VAFB), and one at Kennedy Space Center (KSC).

In [12]:

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

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

Out[12]:

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

Launch Site Names Begin with 'CCA'

Query Result
Five records with Launch_Site values starting with "CCA" (e.g., "CCAFS LC-40" or "CCAFS SLC-40").

Short Explanation
These entries correspond to launches originating from Cape Canaveral Air Force Station, demonstrating how a simple pattern match (LIKE 'CCA%') filters the dataset to include only those sites

In [16]:

```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

Out[16]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO
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)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)

Total Payload Mass

Query Result
The total payload mass for boosters from NASA (CRS) is **45,596 kg**.

Short Explanation
By filtering on rows where the Customer is 'NASA (CRS)' and summing the payload mass, we obtain the total weight of all payloads launched under NASA's Commercial Resupply Services

```
In [23]:
# df.Customer.unique()
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS total_payload_mass FROM SPACEXTABLE WHERE "Cu

 * sqlite:///my_data1.db
Done.
Out[23]:
```

total_payload_mass
45596

Average Payload Mass by F9 v1.1

Query Result

The average payload mass for the F9 v1.1 booster version is **2,928.4 kg**.

Short Explanation

By selecting only rows where the booster version “F9 v1.1” and averaging the payload mass, we get the typical payload weight carried by this specific booster configuration.

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

```
In [24]:
%sql SELECT AVG("PAYLOAD_MASS__KG_") AS AVERAGE_PAYLOAD_MASS FROM SPACEXTABLE WHERE Bo
```

```
 * sqlite:///my_data1.db
Done.
Out[24]:
```

AVERAGE_PAYLOAD_MASS
2928.4

First Successful Ground Landing Date

Query Result

The first successful landing on a ground pad took place on **2015-12-22**.

Short Explanation

Using the MIN() function on the Date column where Landing_Outcome equals "Success (ground pad)" identifies the earliest recorded ground pad success date in the dataset.

```
In [27]:
ql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)'

 * sqlite:///my_data1.db
Done.
Out[27]:
```

MIN("Date")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Query Result

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

Short Explanation

These boosters successfully landed on a drone ship while carrying payloads between 4,000 and 6,000 kg, highlighting their capacity to achieve both significant payload delivery and recovery success.

In [37]:

```
%%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.
```

Out[37]:

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

Total Number of Successful and Failure Mission Outcomes

Query Result
•**Successful Outcomes:** 61
•**Failure Outcomes:** The remainder of missions not matching “Success%” in the Landing_Outcome field.

Short Explanation
The query counts all landing outcomes beginning with the word “Success,” yielding a total of 61 successful outcomes. A similar approach (e.g., “WHERE Landing_Outcome LIKE 'Failure%'”) can be used to determine the number of failures.

List the total number of successful and failure mission outcomes

In [41]:

```
%%sql SELECT COUNT("Landing_Outcome") AS Succes_Count FROM SPACEXTABLE
WHERE "Landing_Outcome" LIKE 'Success%'
```

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

Out[41]:

Succes_Count
61

Boosters Carried Maximum Payload

Query Result
- 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

Short Explanation

All these boosters share the same maximum payload mass value, indicating they each carried the heaviest payload recorded in the dataset.

In [42]:

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

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

Out[42]:

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

Query Result

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

2.Month: 04 | Landing_Outcome: Failure (drone ship) | Booster_Version: F9 v1.1 B1015 | Launch_Site: CCAFS LC-40

Short Explanation

These two flights in 2015 used F9 v1.1 boosters, launched from CCAFS LC-40, and both failed to land successfully on the drone ship.

In [43]:

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

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

Out[43]:

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

Query Result

1.No attempt: 10
2.Success (drone ship): 5
3.Failure (drone ship): 5
4.Success (ground pad): 3
5.Controlled (ocean): 3
6.Uncontrolled (ocean): 2
7.Failure (parachute): 2
8.Precluded (drone ship): 1

Short Explanation

This ranking shows how often each landing outcome occurred between June 4, 2010, and March 20, 2017. "No attempt" is the most frequent, followed by drone ship successes and failures, while outcomes like "Precluded (drone ship)" appear less frequently.

In [44]:

```
%%sql
SELECT "Landing_Outcome", COUNT(*) AS outcome_count
FROM SPACEXTABLE
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[44]:

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

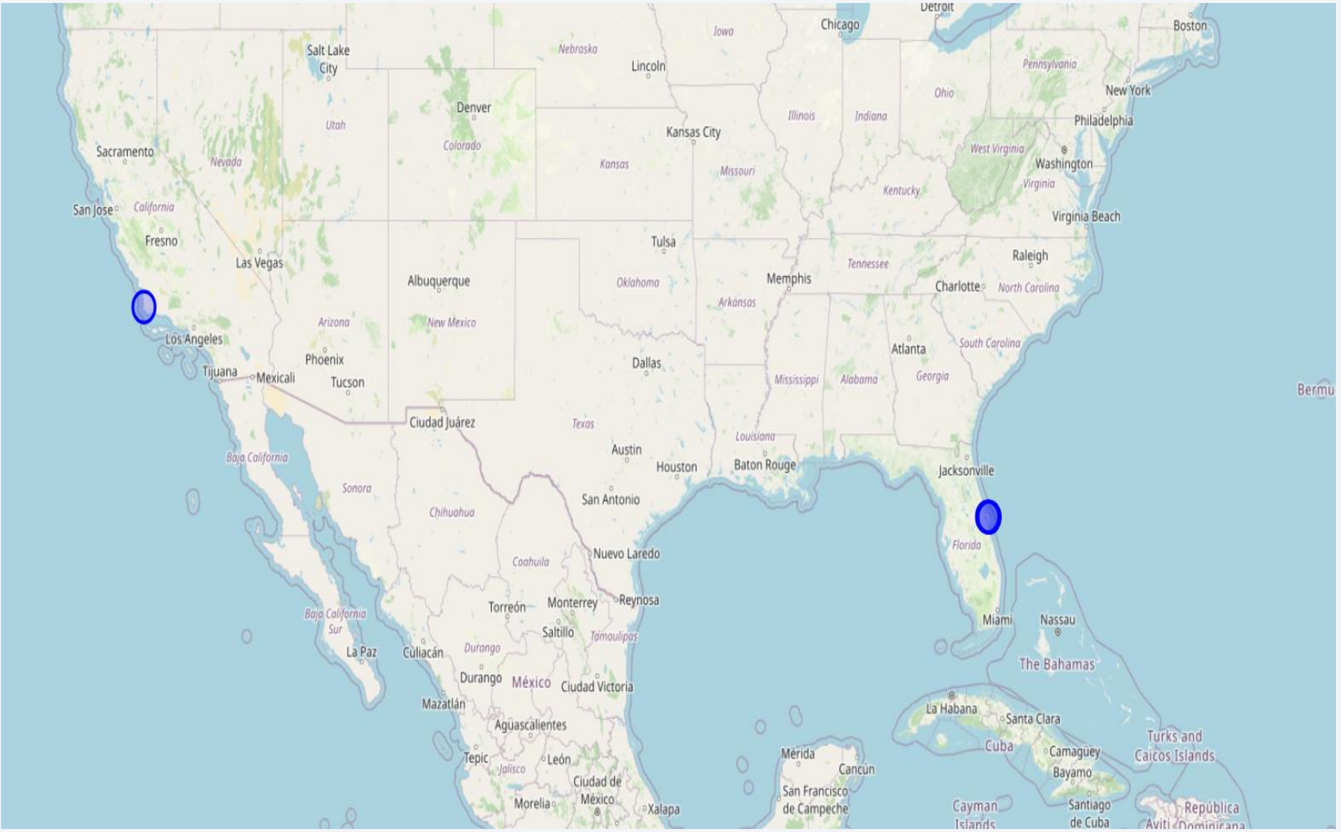
Section 3

Launch Sites Proximities Analysis

Launch Site Locations

Important Elements and Findings in the Folium Map Screenshot

- **Global Perspective:** Shows how the launch sites are spread geographically, allowing you to see relative distances and positions.
- **Markers for Each Launch Site:** Clicking on a marker typically reveals additional details (e.g., site name, coordinate information).
- **Zoom & Pan Controls:** Allow exploration of various regions and the distance between sites.
- **Immediate Insight:** You can see that most of the SpaceX launch sites are concentrated along the U.S. coastline (Florida and California), highlighting strategic geographic considerations like proximity to the equator for efficient launches.



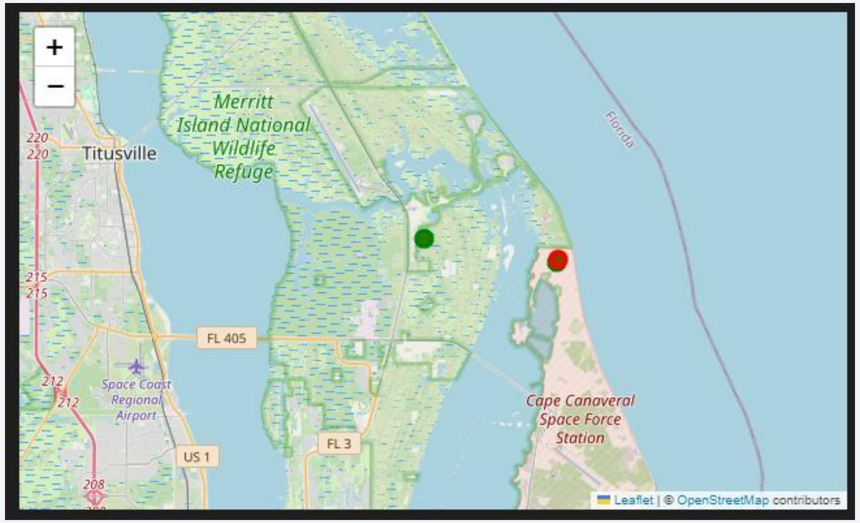
Launch Site Success / Failure Sites

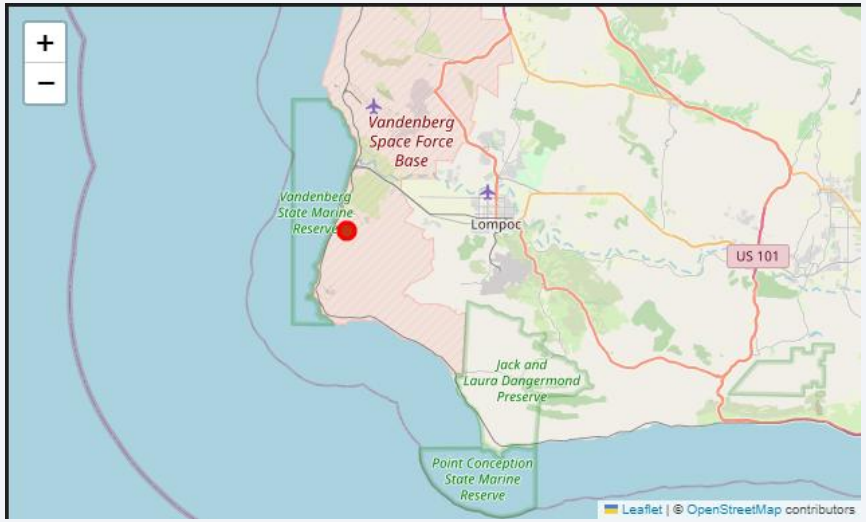
Important Elements and Findings on the Color-Labeled Folium Map

• **Success/Failure Markers** – Each circle's color (green or red) corresponds to a successful or failed launch, providing immediate visual feedback on performance at each location.

• **Pop-up Information** – Clicking a marker reveals the launch site name and its outcome ("Success" or "Failure"), helping you quickly understand site-specific results.

• **Geographical Distribution** – Seeing multiple markers together highlights clusters of successful launches at certain sites, suggesting operational efficiencies or optimal conditions in those areas. Meanwhile, any concentration of red markers may indicate potential issues or challenges at specific locations.





Launch Site Proximity to Coastline

Important Elements and Findings on the Screenshot

• **Highlighted Launch Site:** The map focuses on a selected launch site with clear markers.

• **Proximity Line:** A line is drawn from the launch site to the nearby coastline, visually representing the distance.

• **Distance Display:** The calculated distance is shown as 0.9 km, providing a precise measure of proximity.

• **Contextual Insight:** This close proximity to the coastline may have implications for logistics, environmental factors, and operational planning, offering a practical perspective on site selection.

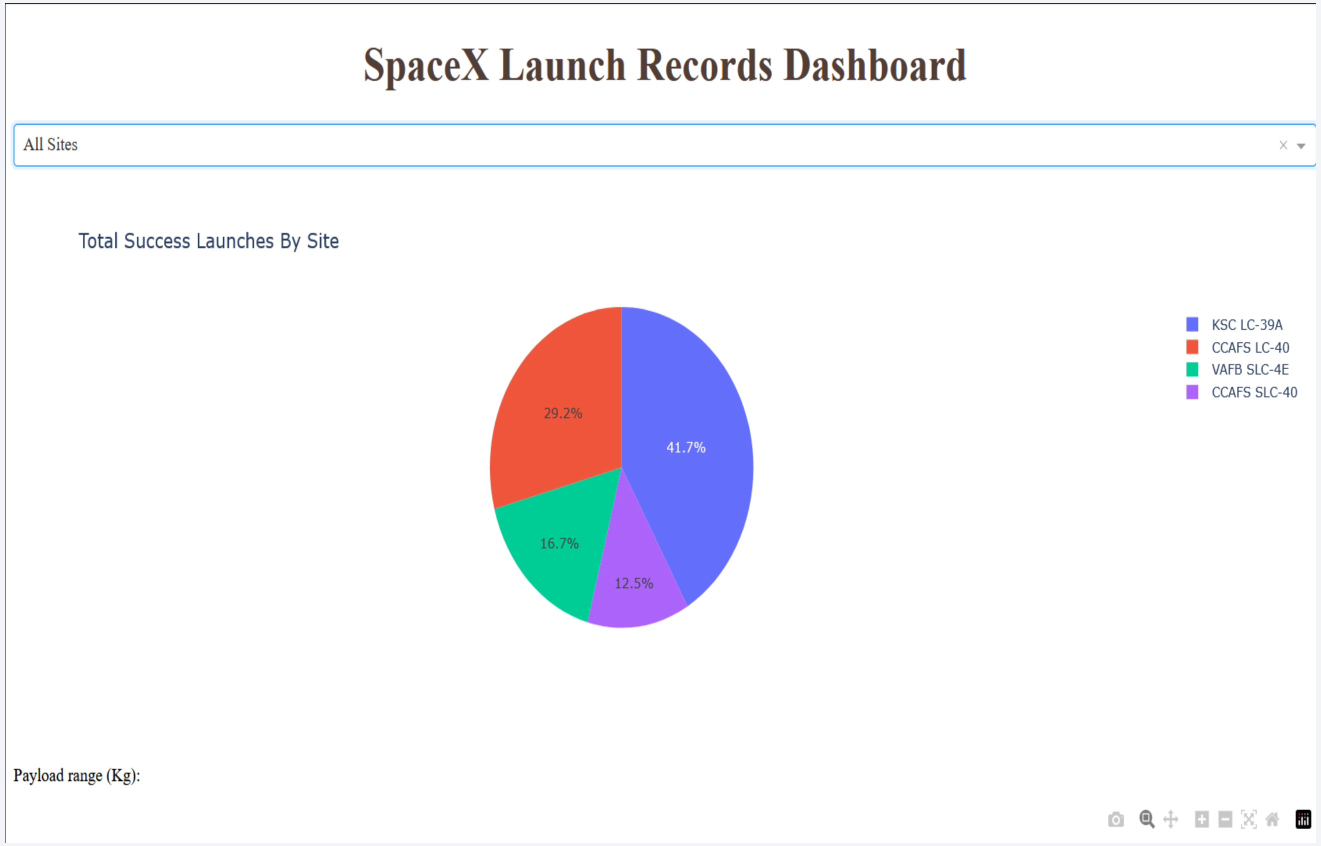
Section 4

Build a Dashboard with Plotly Dash

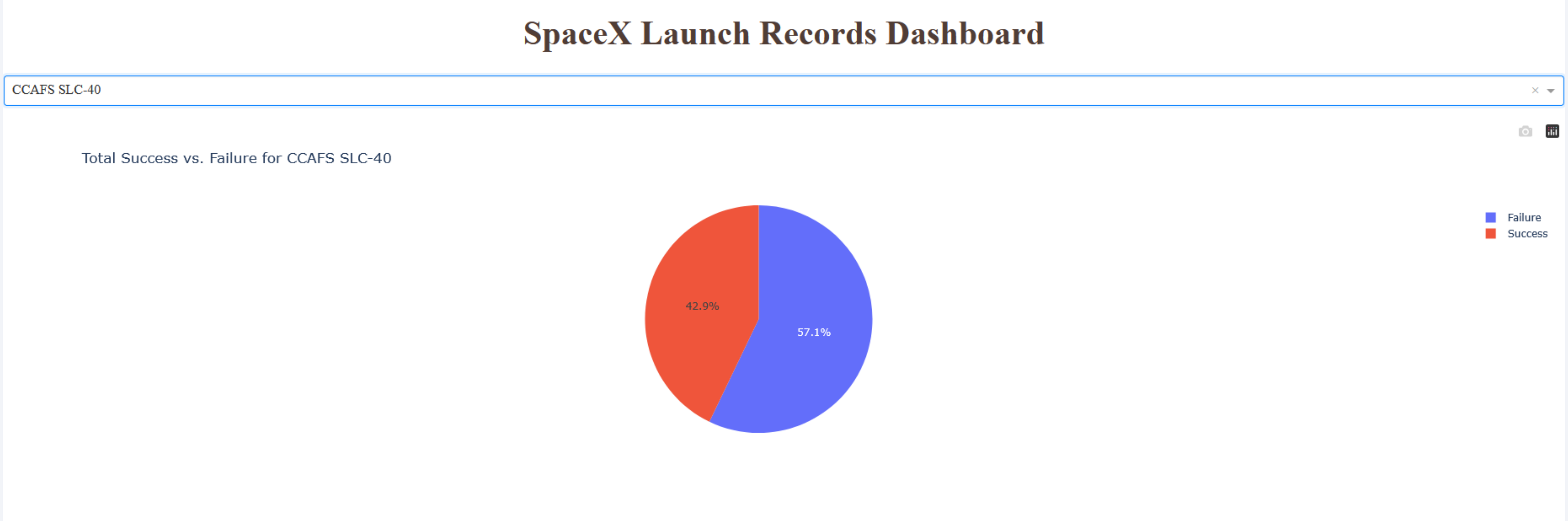
Launch Success Distribution by Site

Important Elements and Findings:

- **Dominant Site:** KSC LC-39A holds the highest success percentage at 41.7%.
- **Relative Shares:** CCAFS LC-40 follows with 29.2%, while VAFB SLC-4E and CCAFS SLC-40 show smaller but still notable shares.
- **Overall Insight:** The pie chart clearly indicates which sites account for the largest proportion of successful launches, helping focus operational or resource decisions where success is highest.



Launch Site with Highest Success Ratio (CCAFS SLC-40)



Important Elements and Findings:

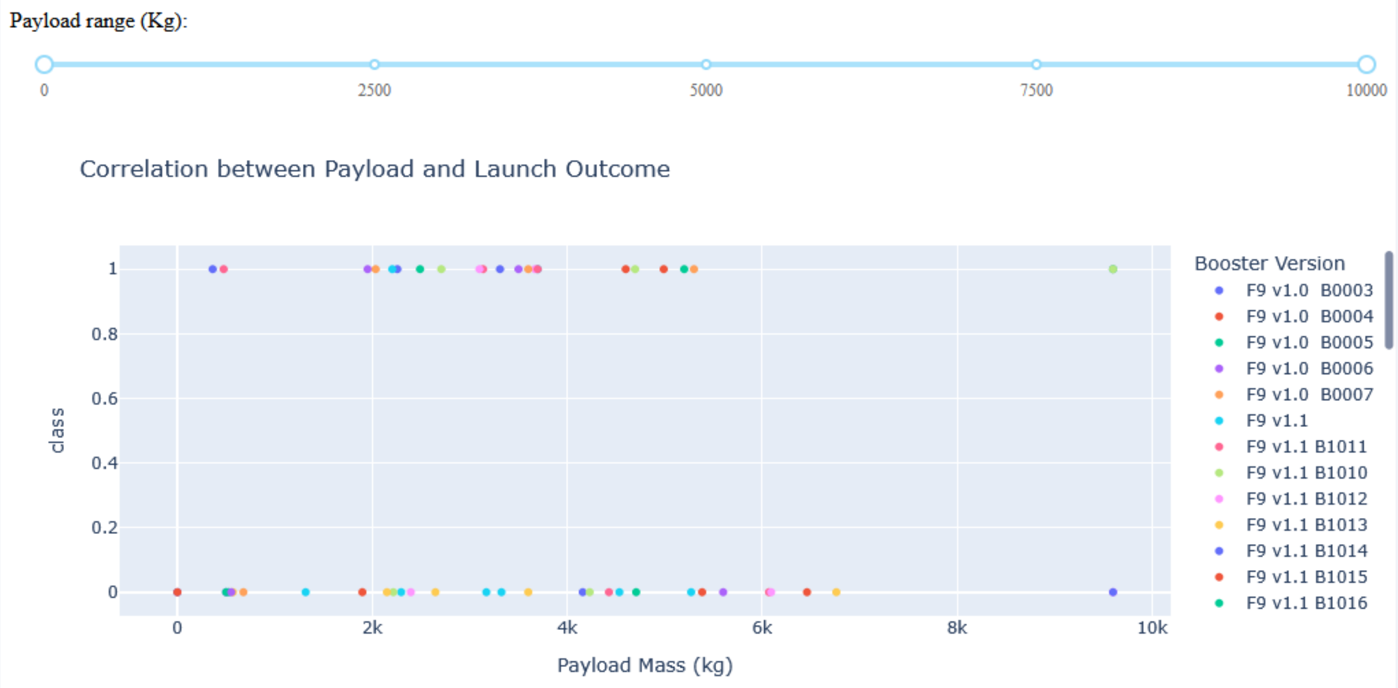
• The pie chart segments illustrate the proportion of successful launches versus failures at this site.

• At 42.9% success, CCAFS SLC-40 outperforms other sites, indicating potentially optimal conditions or operational efficiency.

Payload vs. Launch Outcome (All Sites) with Varying Payload Ranges

Screenshot Explanation

- **Range Slider Usage:** By adjusting the payload range slider, the scatter plot updates to show only launches within that specified payload interval, helping you identify which payload masses correlate with successful or failed outcomes.
- **Booster Version Highlights:** Certain boosters (e.g., F9 B4 B1045.1, F9 FT B1038.1, F9 FT B1021.1, etc.) appear consistently in the successful range, indicating a strong track record for those specific versions.
- **Payload-Outcome Relationship:** You can observe how heavier payloads sometimes cluster around fewer successes, while mid-range payloads often see higher success rates, depending on the booster version.
- **Overall Insight:** This interactive view provides a clear picture of which booster versions perform best across different payload masses, guiding decisions about matching boosters to payload sizes for optimal mission success.

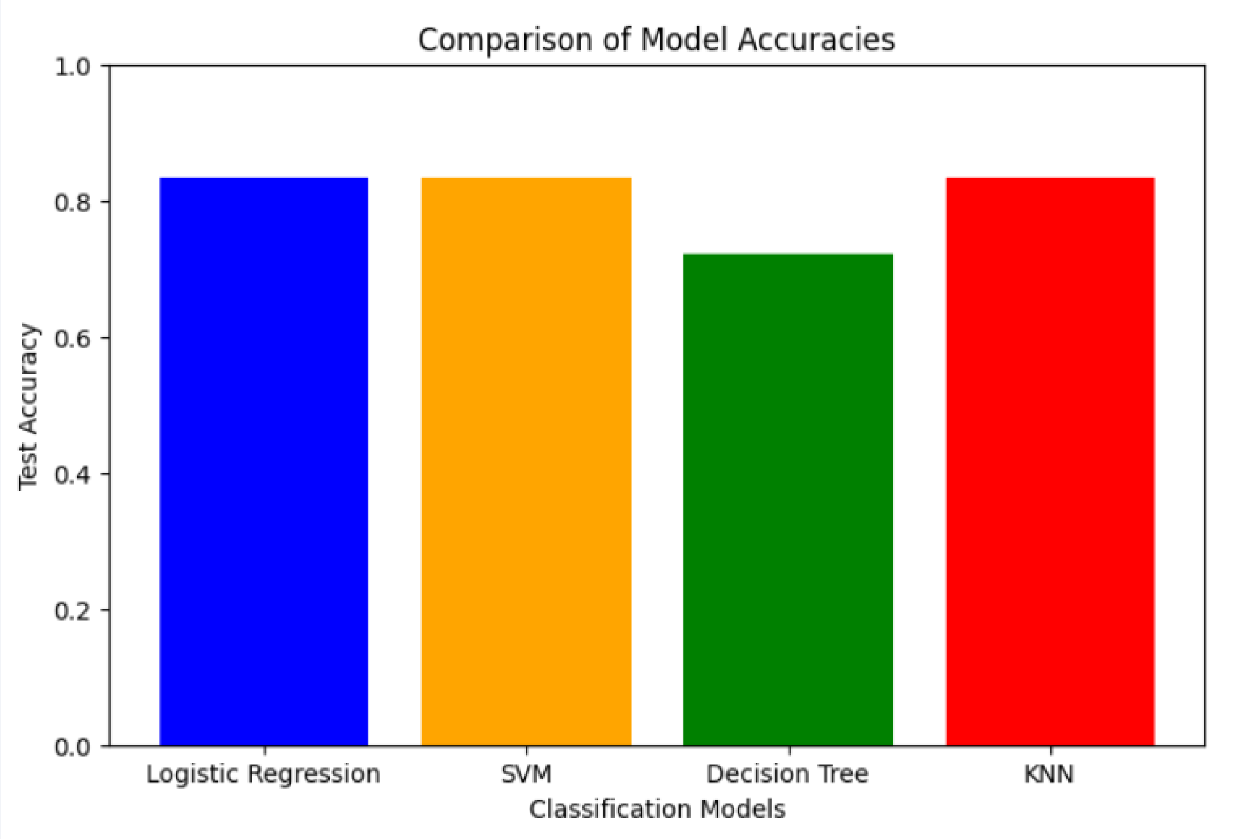


Section 5

Predictive Analysis (Classification)

Classification Accuracy

Logistic Regression, SVM, and KNN are tied for the highest test accuracy at 0.8333, while the Decision Tree model has a lower accuracy at 0.7222.



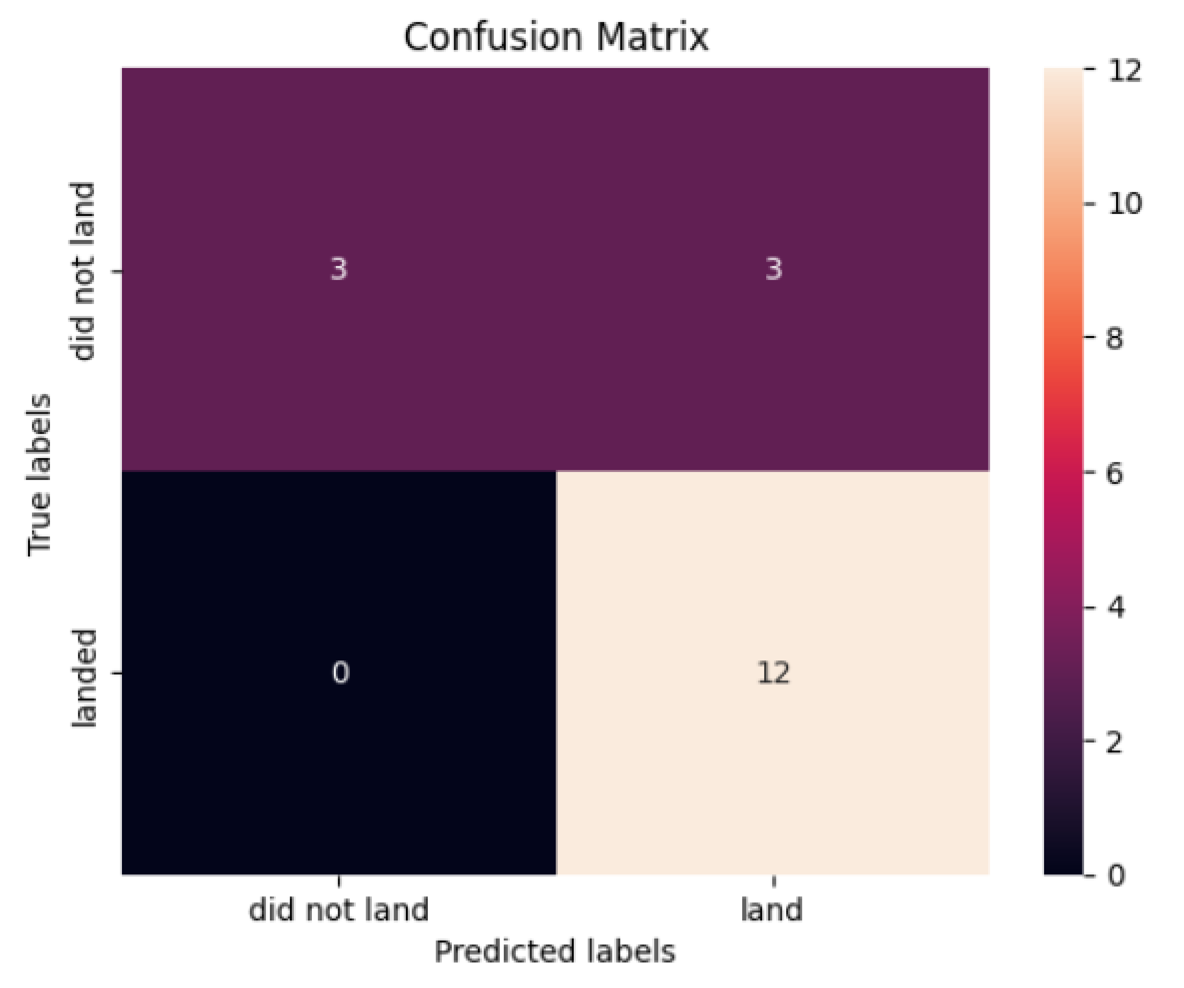
Confusion Matrix – Logistic Regression Model

Logistic Regression Confusion Matrix Explanation

•**True Positives (TP):** The model correctly predicted that the first stage would land (e.g., 12 cases).

•**False Positives (FP):** The model incorrectly predicted a landing when it actually failed (e.g., 3 cases).

•**True Negatives & False Negatives:** These values further detail how the model performs on non-landing predictions.



Conclusions

1. Logistic Regression, SVM, and KNN all achieved similar test accuracies (≈83.33%), while the Decision Tree model lagged behind at 72.22%.

2. The confusion matrix for logistic regression reveals effective differentiation between successful and failed landings, though some false positives persist.

3. Logistic Regression offers an interpretable and cost-effective solution, making it a strong candidate when models have similar predictive performance.

4. Overall, the ensemble of models shows robust predictive power for launch outcomes, with model selection balancing accuracy, interpretability, and operational risk.

Thank you!