IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Vahid Sahraei
August 2025

https://github.com/vahidsahraei/Applied-Data-Science-Capstone

# Outline

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

# Executive Summary

- In this project, data from the SpaceX API and Wikipedia was collected and processed to create a dataset with a classification label for successful landings. The dataset was prepared using SQL, visualizations, feature selection, one-hot encoding, and standardization. Model optimization was performed with GridSearchCV, and results were presented through dashboards and accuracy metrics.
- Four machine learning models, Logistic Regression, SVM, Decision Tree, and KNN were developed, each achieving approximately 83.3% accuracy.
- All models showed a tendency to over-predict successful landings.
- Additional data is needed to enhance model robustness, selection, and predictive accuracy.

# Introduction

- **Project Background and Context:**
- This capstone project focuses on predicting the successful landing of the Falcon 9 first stage. SpaceX offers launches at a significantly lower cost than other providers, primarily due to the reusability of its first-stage rockets. By accurately predicting landing success, this project aims to estimate launch costs and provide actionable insights for companies competing with SpaceX.
- **Problem:**

As data scientists at SpaceY, we are tasked with developing a machine learning model to predict the successful recovery of the Falcon 9 first stage

Section 1

# Methodology

# Methodology

- Data collection methodology:
  - Collected data from SpaceX public API and WikipediaPerformed data wrangling and classification of landings (successful vs. unsuccessful)
- Performed data wrangling and classification of landings (successful vs. unsuccessful)
- Data Analysis & Visualization
- Conducted exploratory data analysis (EDA) using SQL and visualizations
  - Created interactive visual analytics with Folium and Plotly Dash
- Predictive Modeling
  - Built classification models to predict landing success
  - Optimized model parameters using GridSearchCV

# Data Collection

## Step 1

### SpaceX API Request

- Initiated API request to fetch launch data
- Stored data locally

## Step 2

### Web Scraping (Wikipedia)

- Extracted launch table
- Parsed with BeautifulSoup
- Converted to DataFrame

## Step 3

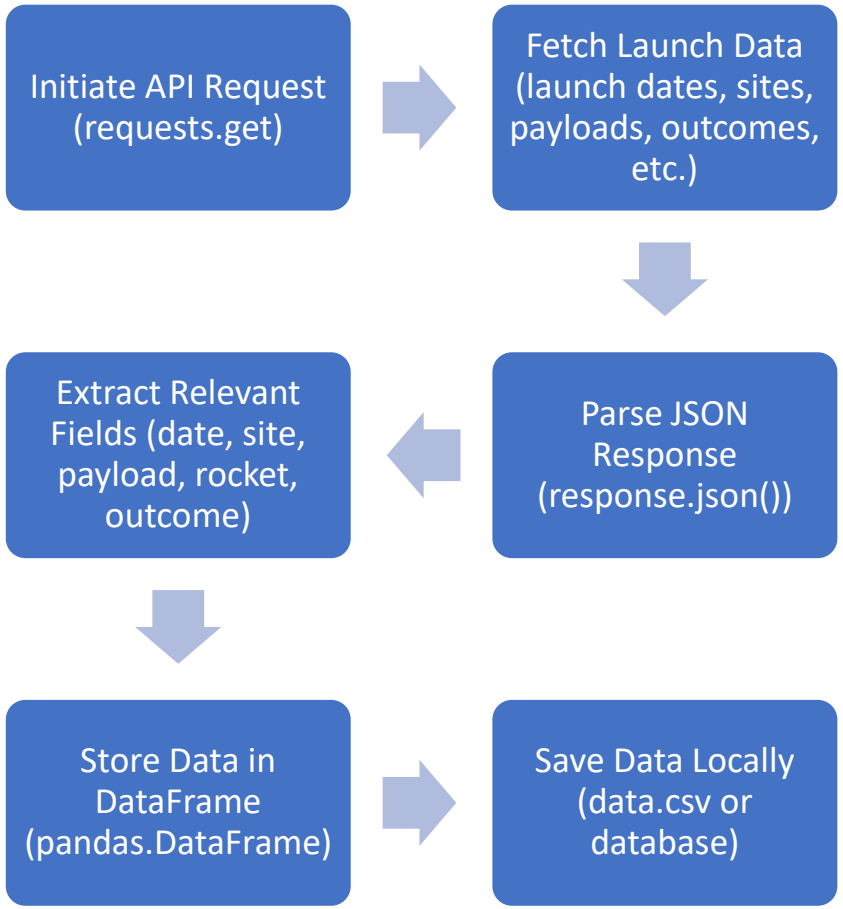
### Data Integration

- Combined SpaceX API data with Wikipedia data
- Merged into a final integrated dataset

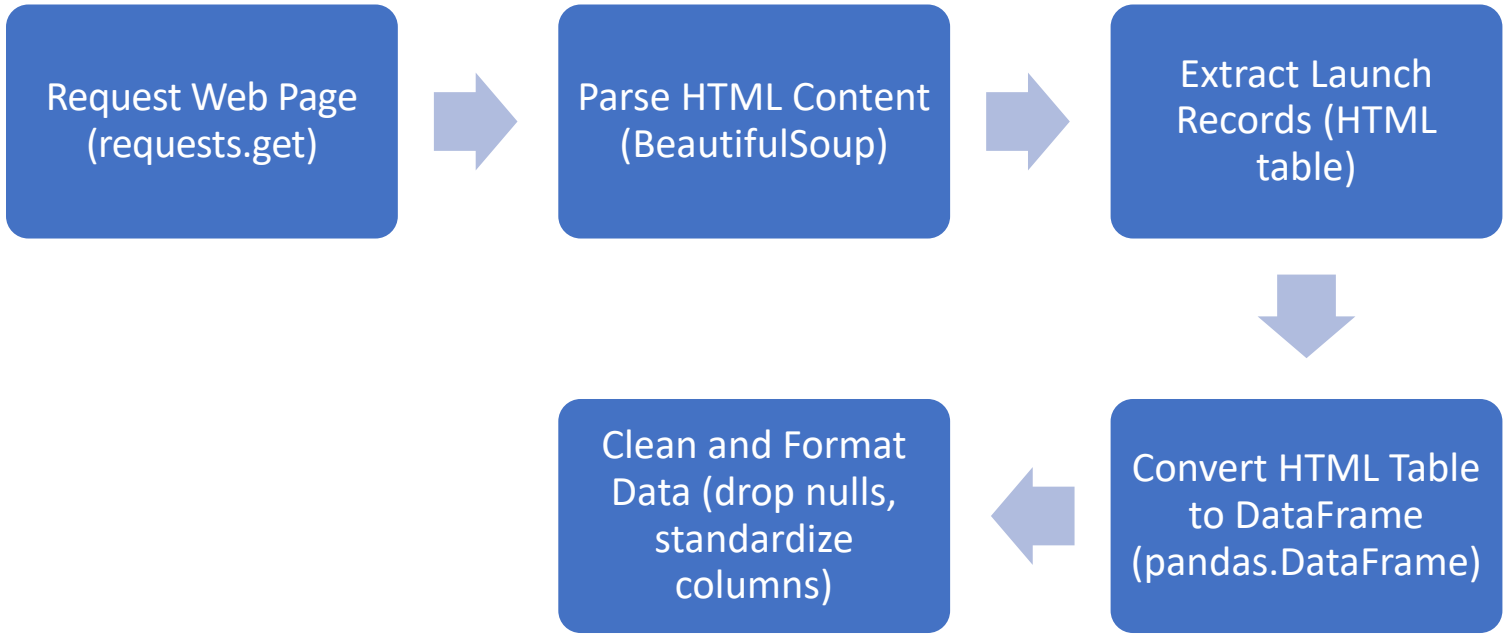
# Data Collection – SpaceX API

- GitHub URL:

https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/(1)%20jupyter-labs-spacex-data-collection-api.ipynb



# Data Collection - Scraping




- GitHub URL: https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/(2)%20jupyter-labs-webscraping.ipynb

# Data Wrangling

- **Step 1: Data Cleaning**
  - Identify and handle missing values (impute or remove).
  - Drop rows/columns with excessive missing data.
- **Step 2: Data Transformation**
  - Convert data types (e.g., dates, numeric).
  - Standardize text (lowercase, remove whitespace).
  - Create new features (e.g., extract year from date).
  - Normalize/scale numerical values.
- **Step 3: Data Integration**
  - Merge datasets from multiple sources (API, web scraping).
  - Ensure consistent column names and data formats.
- **Step 4: Data Validation**
  - Remove duplicate records.
  - Verify accuracy and consistency of entries.

GitHub URL: https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/(3)%20labs-jupyter-spacex-Data%20wrangling.ipynb

# EDA with Data Visualization

- **Key variables: Flight Number, Payload Mass, Launch Site, Orbit, Class, Year**
- **Visualizations: Scatter, Line, and Bar Charts**
- **Relationships examined:**
  - Flight Number vs. Payload & Launch Site
  - Payload vs. Launch Site & Orbit
  - Orbit vs. Success Rate
  - Yearly Success Trend

**Goal: Identify patterns to support machine learning model training**

GitHub URL: https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/(5)%20EDA%20Data%20Visualisation.ipynb

# EDA with SQL

- **Loaded dataset into IBM DB2 database.Visualizations: Scatter, Line, and Bar Charts**

- **Used SQL via Python integration for querying.**

**Explored dataset to understand:**

- Launch site names
- Mission outcomes
- Payload sizes and customer info
- Booster versions
- Landing outcomes

GitHub URL: https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/(4)%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

# Build an Interactive Map with Folium

- **Interactive Map with Folium show:**
  - Launch Sites
  - Successful vs. Unsuccessful Landings
  - Proximity to key locations: Railway, Highway, Coast, City

**Purpose:**

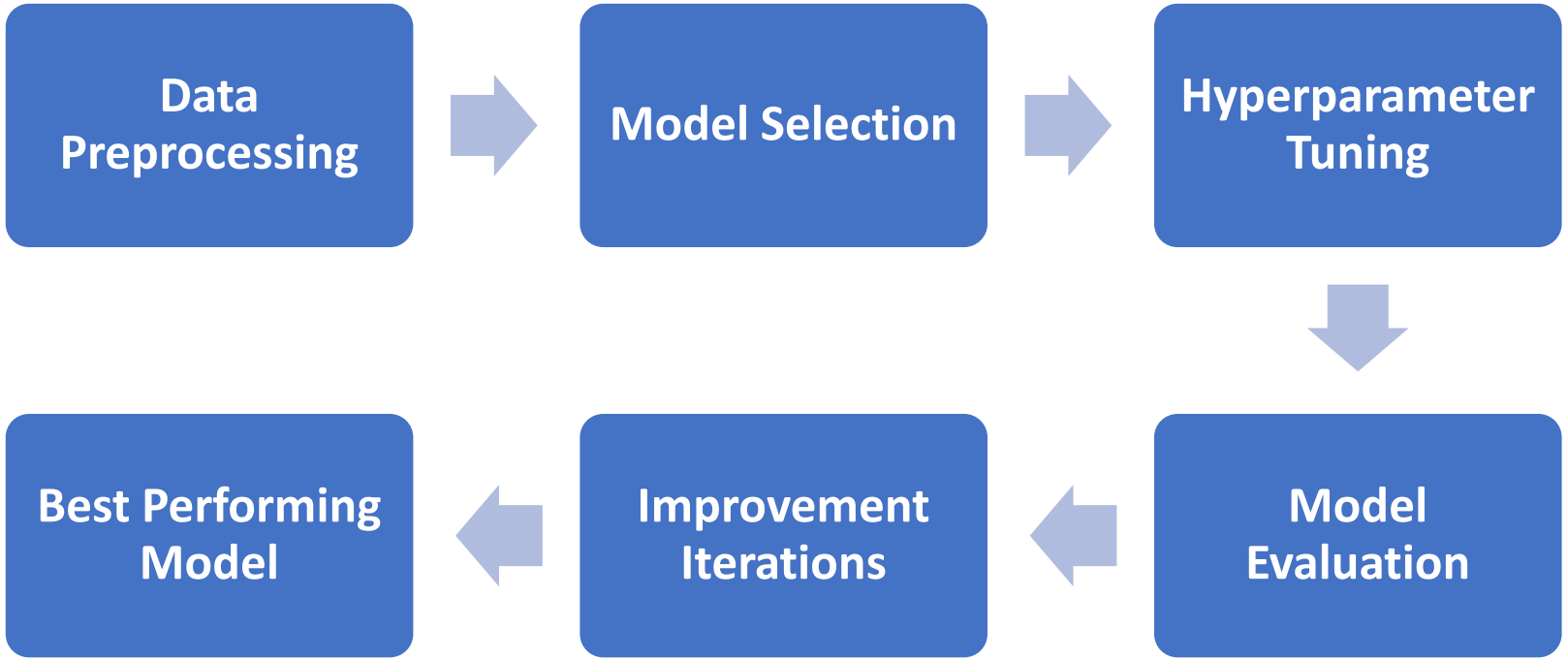
  - Understand launch site placement
  - Visualize landing success relative to location

GitHub URL: https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/(6)%20Visualisation_maps_launch_site_location.ipynb

# Build a Dashboard with Plotly Dash

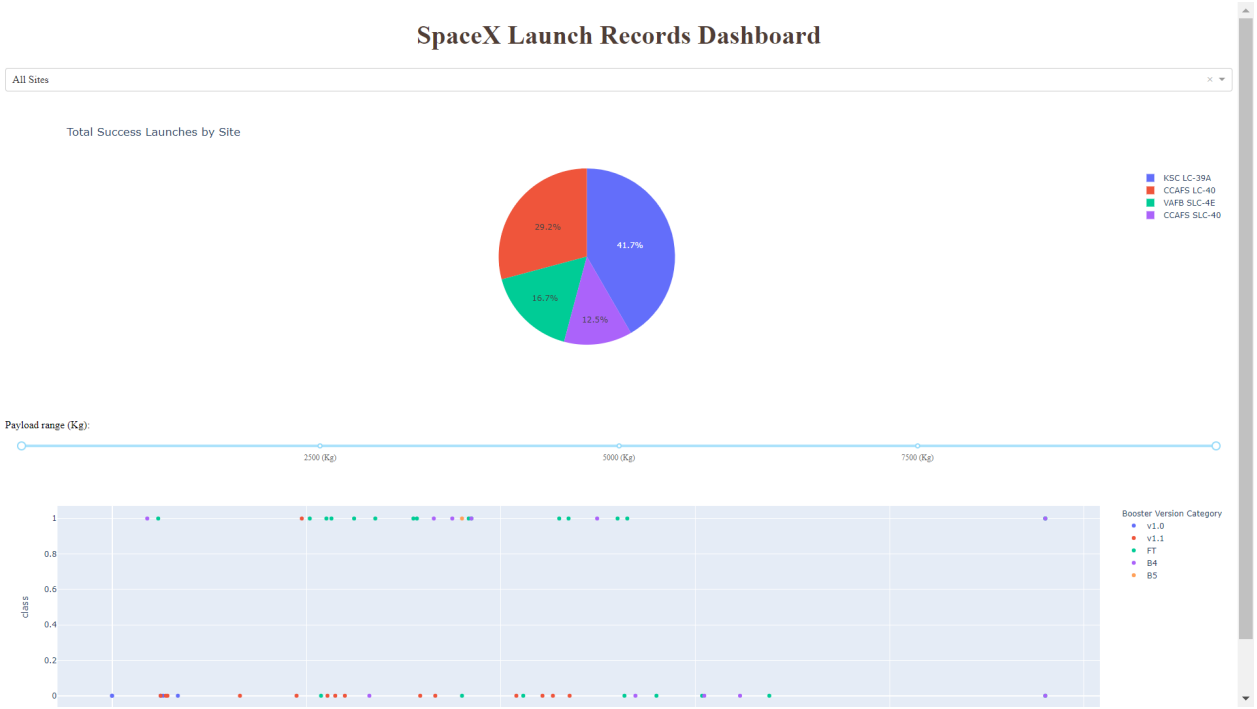
- **Visuals: Pie Chart & Scatter Plot**
- **Pie Chart:**
  - Shows overall distribution of successful landings
  - Can display individual launch site success rates
- **Scatter Plot:**
  - Inputs: All sites or selected site, Payload Mass (0–10,000 kg)
  - Shows how success varies by launch site, payload, and booster version
- **Purpose: Interactively explore relationships and trends in landing success**
- GitHub URL: https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/spacex-dash-app.py

# Predictive Analysis (Classification)

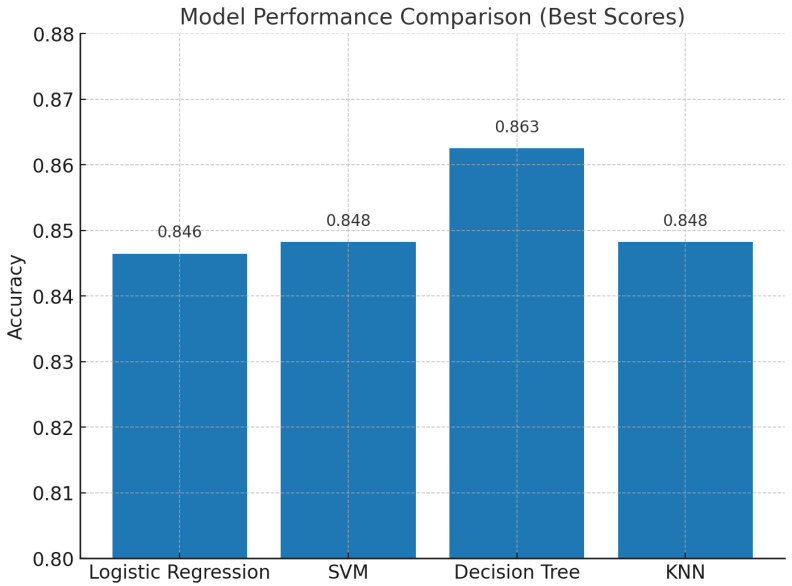



https://github.com/vahidsahraei/Applied-Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

# Results




Plotly dashboard



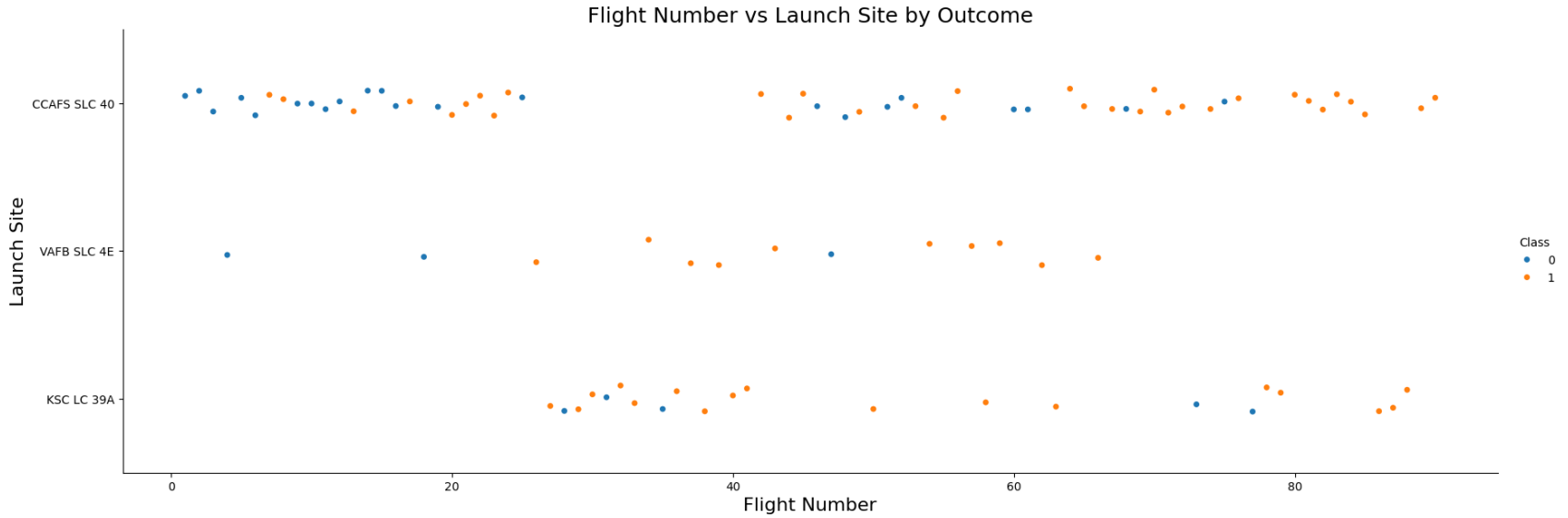

Predictive analysis results

Section 2

# Insights drawn from EDA

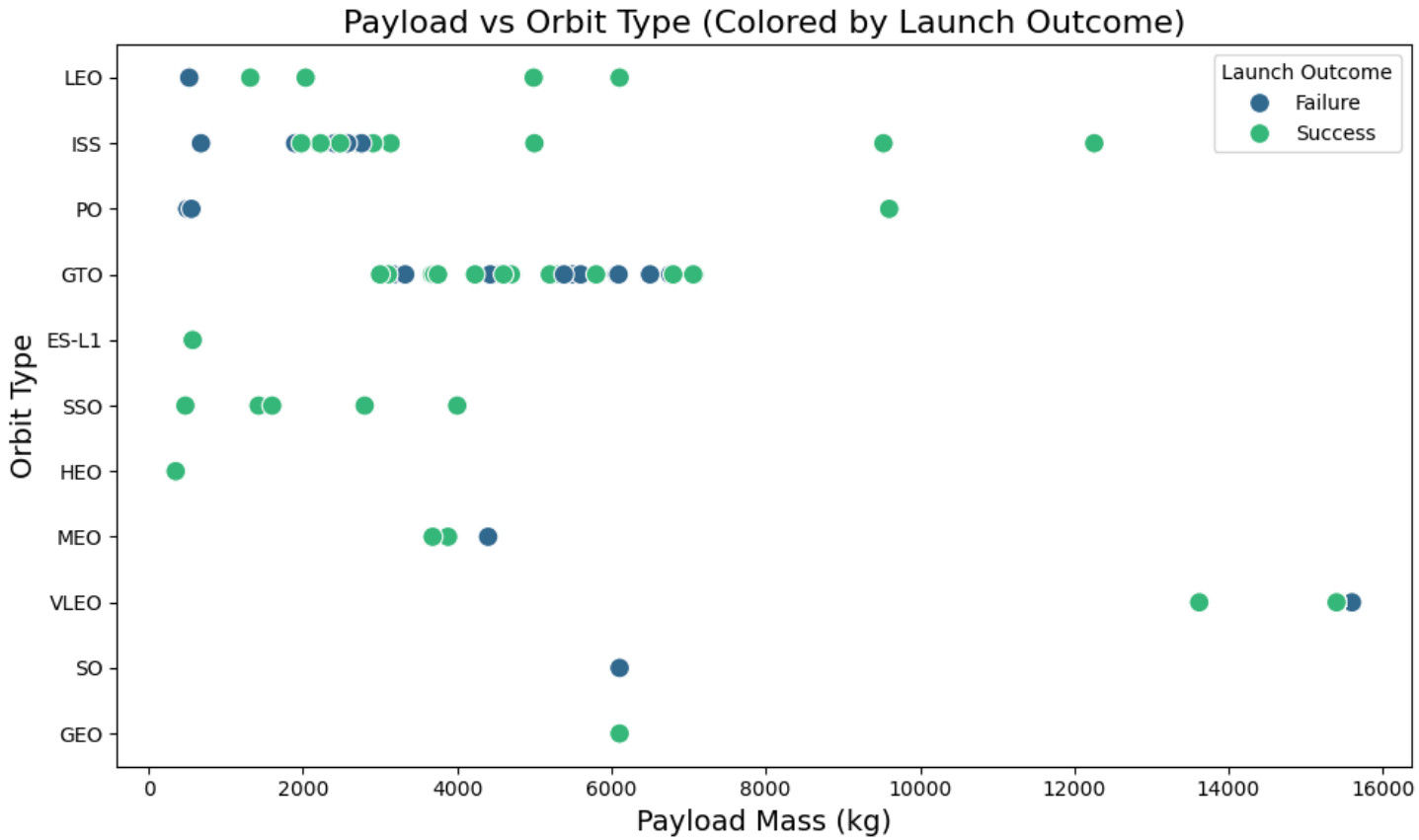
# Flight Number vs. Launch Site

- **Success on the Rise:** Success rate climbs steadily with flight number — major breakthrough after Flight 20.
- **CCAFS Leads the Way:** it dominates with the highest number of launches.
- **Mixed Results at Key Sites:** Both *CCAFS SLC-40* and *KSC LC-39A* show successes and failures — location alone isn't the driver.



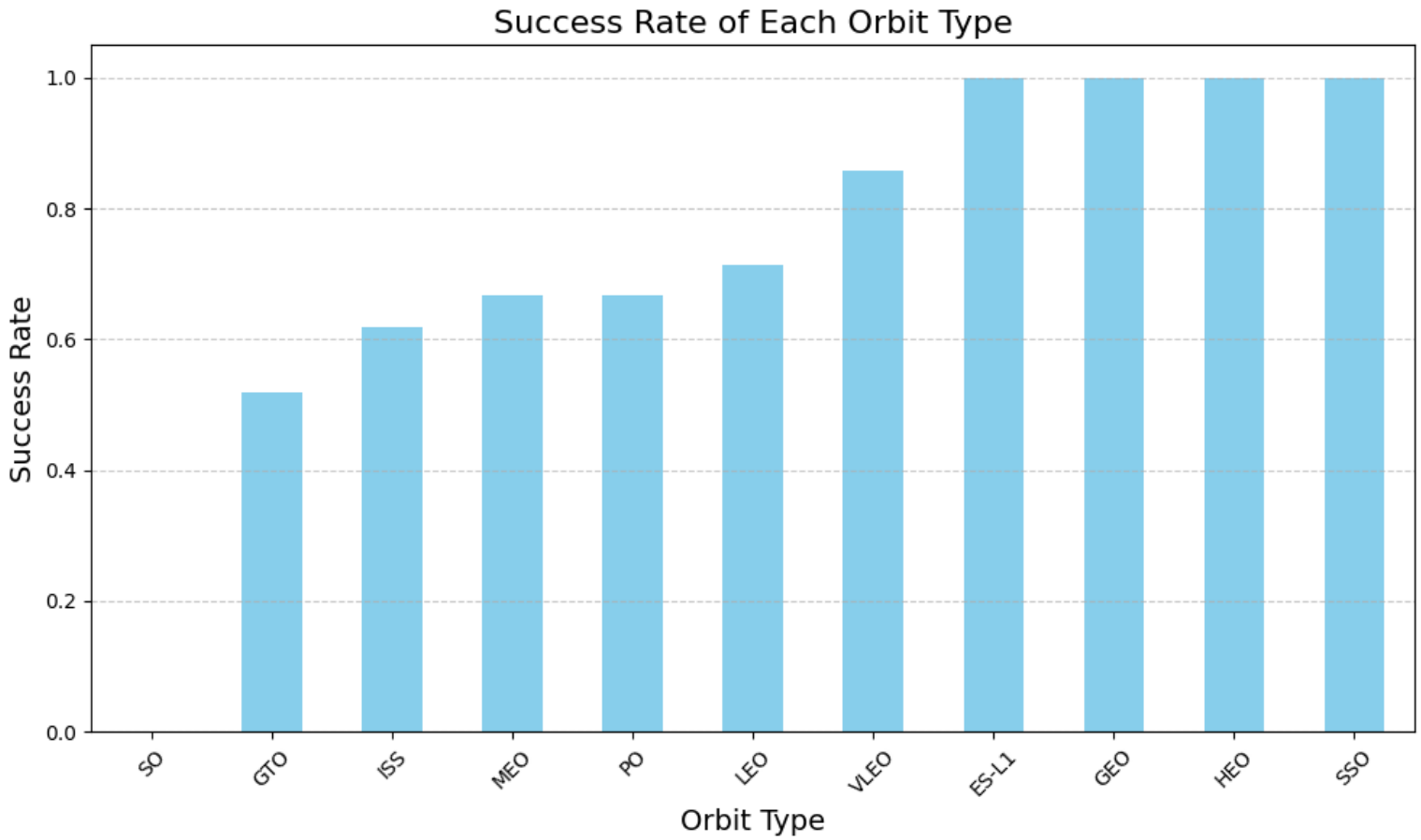
# Payload vs. Launch Site

- **Payload Distribution:** CCAFS SLC-40 mainly launches <10,000 kg, while VAFB SLC-4E and KSC LC-39A support a wider range
- **High-Capacity Launches:** KSC LC-39A stands out for handling heavier payloads, with several missions exceeding 15,000 kg, highlighting its role in high-capacity.



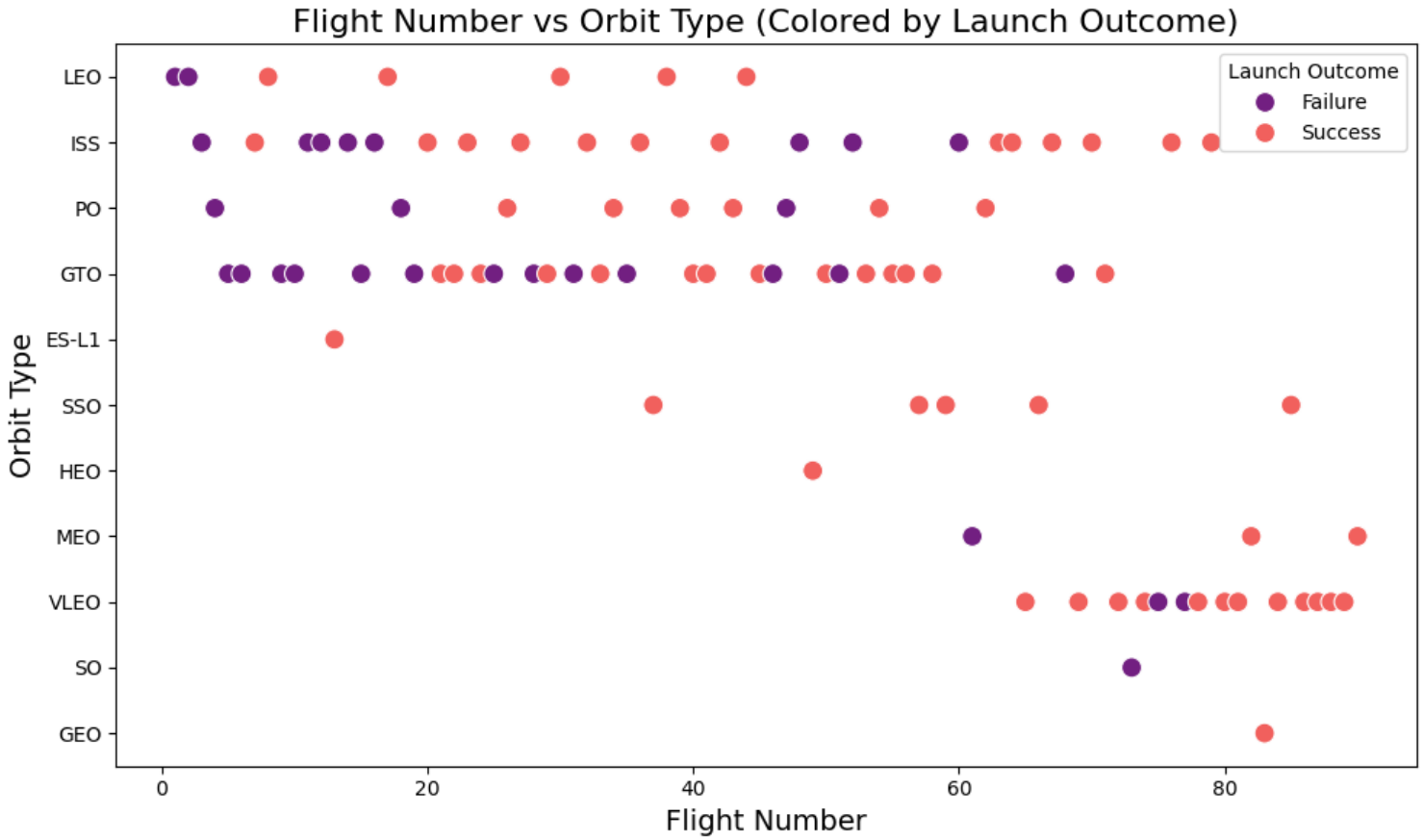
# Success Rate vs. Orbit Type

- **100% Success:** ES-L1 (1), GEO (1), HEO (1), SSO (5)
- **Strong Performance:** VLEO (14) – good success & attempts
- **Low Success:** SO (1) – 0% success
- **Mixed Results:** GTO (27) – ~50% success, largest sample



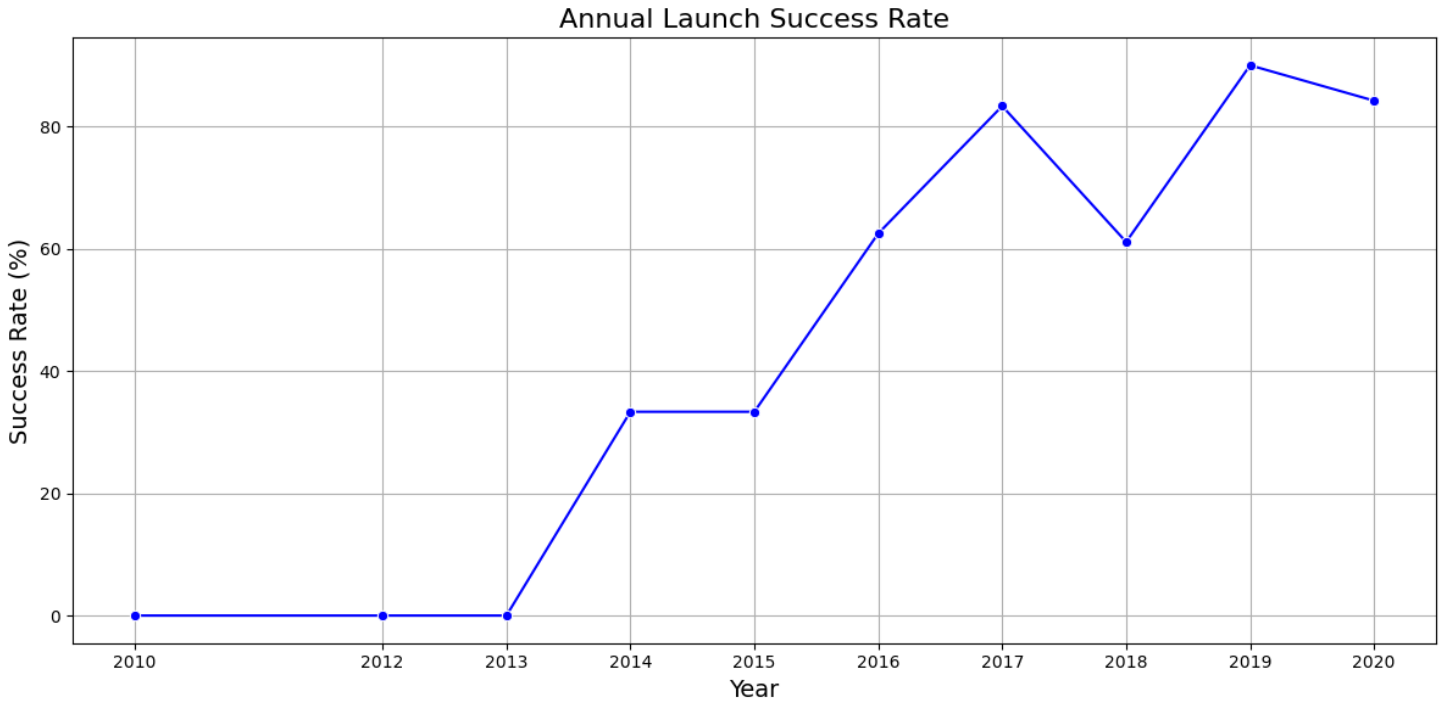
# Flight Number vs. Orbit Type

- **Success increased Over Time:** Falcon 9 launch success rises with higher flight numbers, showing the impact of experience and iterative improvements. This trend is clear in LEO, where more flights correlate with higher success.
- **Orbit-Specific Performance:**
- For GTO, success shows no clear link to flight numbers. Early GTO and ISS missions had mixed results, but recent ones demonstrate improved outcomes, highlighting advances in planning and execution.
- Stronger results in lower and Sun-synchronous orbits.



# Launch Success Yearly Trend

- **Steady Improvement:** Since 2013, launch success rates climbed, topping 80% by 2020.
- **Resilient Growth:** Aside from a 2018 dip, Falcon 9 has become increasingly reliable.



# All Launch Site Names

- This query returns a list of unique launch sites stored in the SPACEXTABLE.
- The SELECT statement retrieves data from a table.The keyword DISTINCT ensures that only unique values are returned, removing duplicates.

### Task 1

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

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'

- This sql query retrieves the first five records from the database where the Launch Site name starts with ‘CCA’.”

## Task 2

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

In [14]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

Out[14]:

| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_Outcome |
|---|---|---|---|---|---|---|---|---|---|
| 2010-06-04 | 18:45:00 | F9 v1.0 B0003 | CCAFS LC-40 | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success | Failure (parachute) |
| 2010-12-08 | 15:43:00 | F9 v1.0 B0004 | CCAFS LC-40 | Dragon demo flight C1, two CubeSats, barrel of Brouere cheese | 0 | LEO (ISS) | NASA (COTS) NRO | Success | Failure (parachute) |
| 2012-05-22 | 7:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 2012-10-08 | 0:35:00 | F9 v1.0 B0006 | CCAFS LC-40 | SpaceX CRS-1 | 500 | LEO (ISS) | NASA (CRS) | Success | No attempt |
| 2013-03-01 | 15:10:00 | F9 v1.0 B0007 | CCAFS LC-40 | SpaceX CRS-2 | 677 | LEO (ISS) | NASA (CRS) | Success | No attempt |

# Total Payload Mass

- This query sums the total payload  mass in where NASA was the  customer.

### Task 3

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

In [17]:
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)';
```

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

Out[17]:

| SUM(PAYLOAD_MASS__KG_) |
|---|
| 45596 |

# Average Payload Mass by F9 v1.1

## Task 4

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

In [18]:
```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

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

Out[18]:

| AVG("PAYLOAD_MASS__KG_") |
|---|
| 2928.4 |

# First Successful Ground Landing Date

### Task 5

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

*Hint:Use min function*

In [19]:
```
%sql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

Out[19]:

| MIN("Date") |
|---|
| 2015-12-22 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

## Task 6

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

In [20]:
```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS_
```

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

Out[20]:

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

# Total Number of Successful and Failure Mission Outcomes

## Task 7

**List the total number of successful and failure mission outcomes**

In [21]:
```
%sql SELECT "Mission_Outcome", COUNT(*) AS "Total" FROM SPACEXTABLE WHERE "Mission_Outcome" IN ('Success', 'Failure') GROUP
```

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

Out[21]:

| Mission_Outcome | Total |
|---|---|
| Success | 98 |

In [32]:
```
%sql SELECT "Mission_Outcome", COUNT(*) AS "Total" FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE 'Success' OR "Mission_Outcc
```

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

Out[32]:

| Mission_Outcome | Total |
|---|---|
| Success | 98 |

# Boosters Carried Maximum Payload

### Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

In [33]:
```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SP
```

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

Out[33]:

| 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

## Task 9

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.

**Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.**

In [35]:

```
%%sql

SELECT
    CASE
        WHEN substr("Date", 6, 2) = '01' THEN 'January'
        WHEN substr("Date", 6, 2) = '02' THEN 'February'
        WHEN substr("Date", 6, 2) = '03' THEN 'March'
        WHEN substr("Date", 6, 2) = '04' THEN 'April'
        WHEN substr("Date", 6, 2) = '05' THEN 'May'
        WHEN substr("Date", 6, 2) = '06' THEN 'June'
        WHEN substr("Date", 6, 2) = '07' THEN 'July'
        WHEN substr("Date", 6, 2) = '08' THEN 'August'
        WHEN substr("Date", 6, 2) = '09' THEN 'September'
        WHEN substr("Date", 6, 2) = '10' THEN 'October'
        WHEN substr("Date", 6, 2) = '11' THEN 'November'
        WHEN substr("Date", 6, 2) = '12' THEN 'December'
        ELSE 'Unknown'
    END AS "Month_Name",
    "Mission_Outcome",
    "Booster_Version",
    "Launch_Site"
FROM
    SPACEXTABLE
WHERE
    substr("Date", 0, 5) = '2015';
```

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

Out[35]:

| Month_Name | Mission_Outcome | Booster_Version | Launch_Site |
|---|---|---|---|
| January | Success | F9 v1.1 B1012 | CCAFS LC-40 |
| February | Success | F9 v1.1 B1013 | CCAFS LC-40 |
| March | Success | F9 v1.1 B1014 | CCAFS LC-40 |
| April | Success | F9 v1.1 B1015 | CCAFS LC-40 |
| April | Success | F9 v1.1 B1016 | CCAFS LC-40 |
| June | Failure (in flight) | F9 v1.1 B1018 | CCAFS LC-40 |
| December | Success | F9 FT B1019 | CCAFS LC-40 |

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

### Task 10

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

In [36]:

```
%%sql

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

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

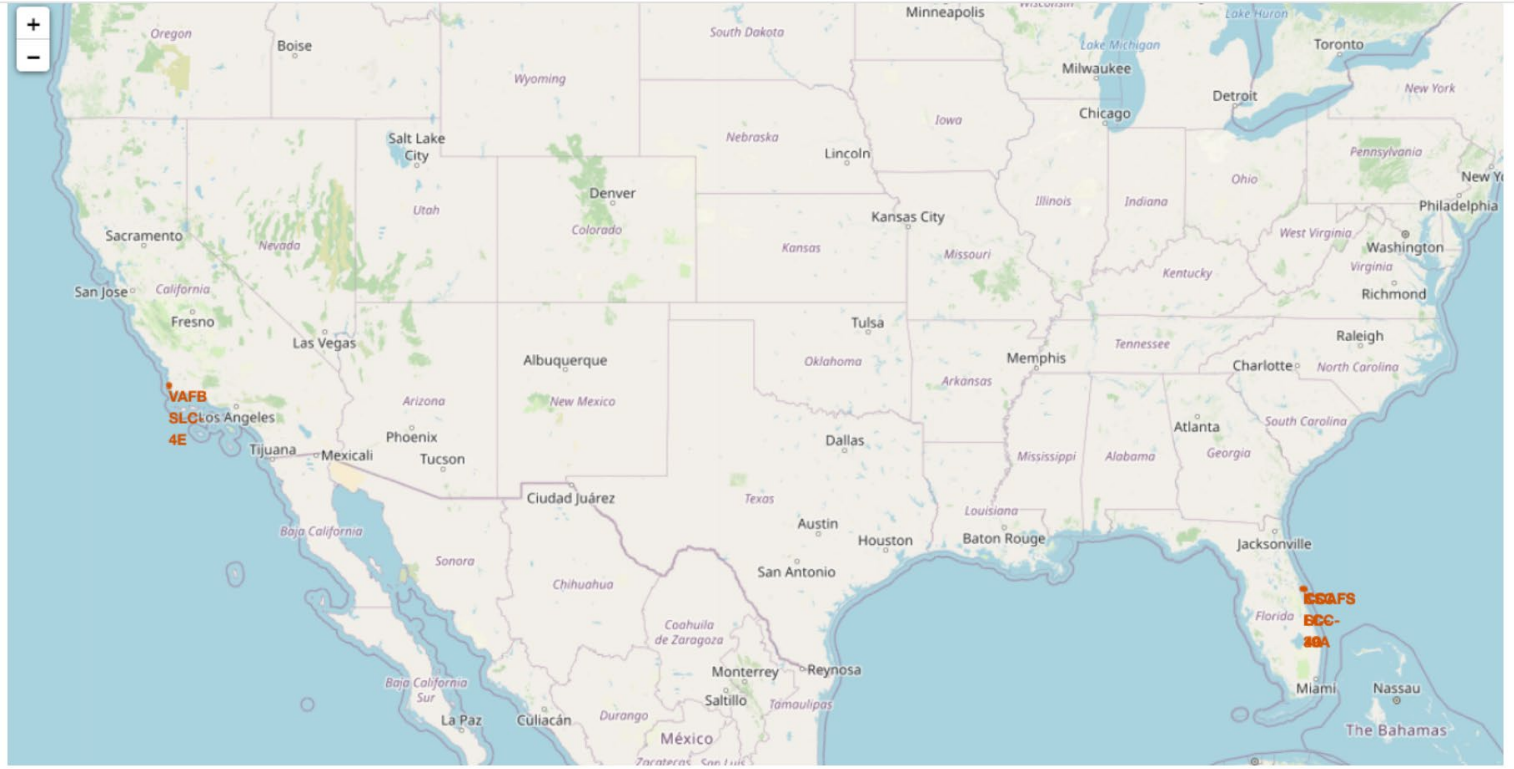
Out[36]:

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


Section 3
Launch Sites
Proximities Analysis

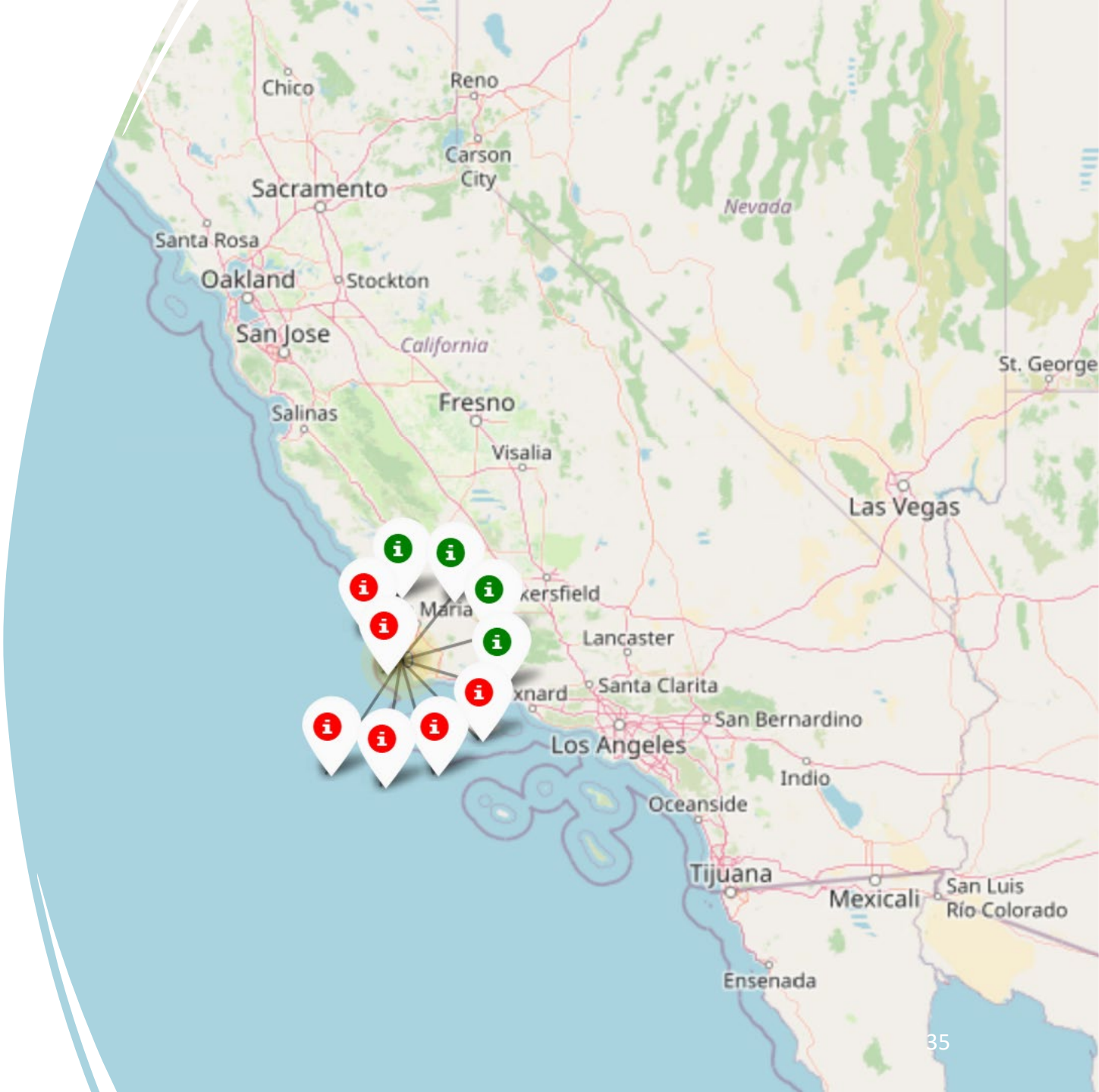
# all launch sites on the map

The map shows all launch sites on US map.



# Visualizing Successful and Failed Landings

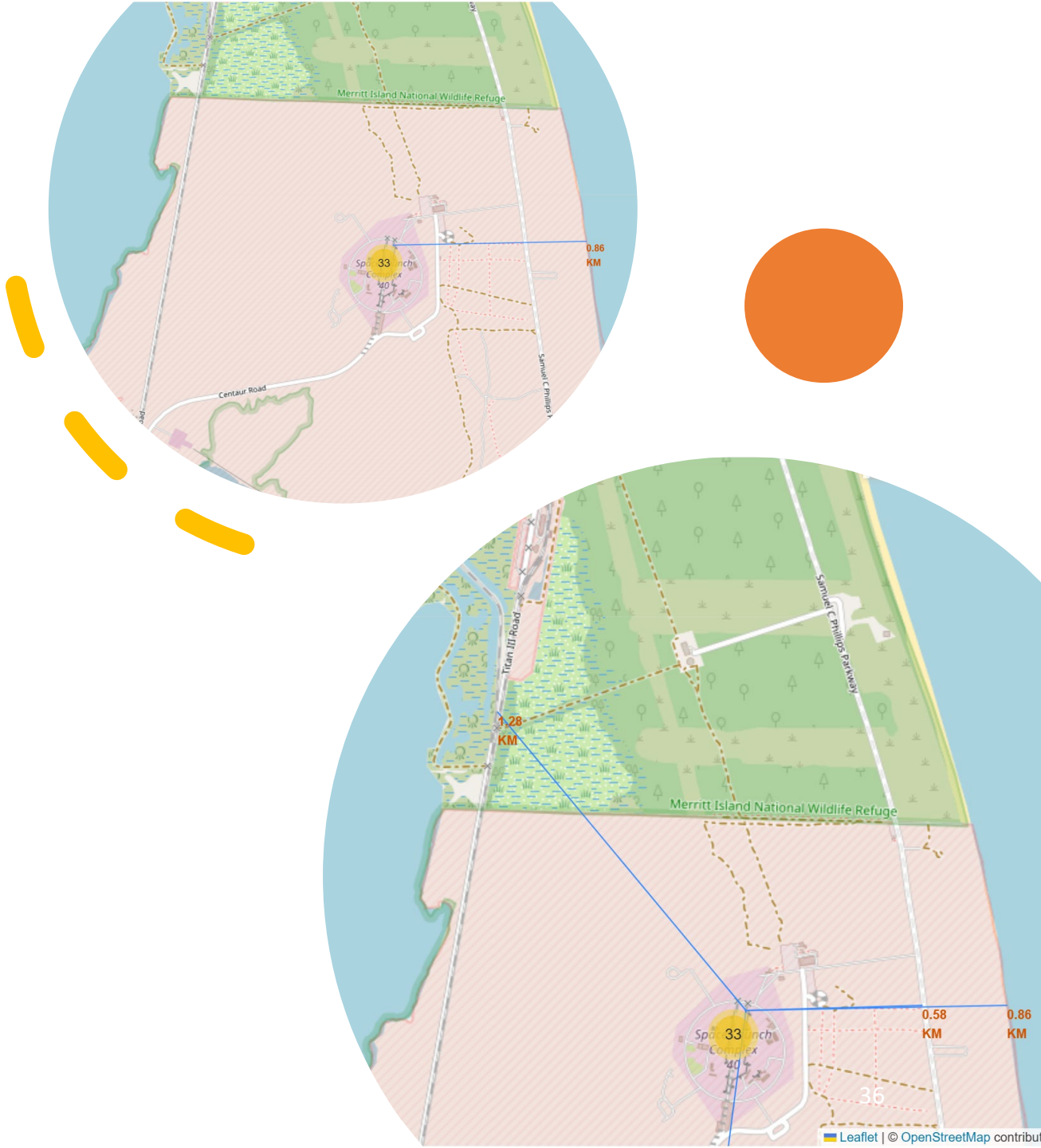
- On the Folium map, clusters can be clicked to reveal individual landings: green icons represent successful landings, and red icons indicate failures. For example, at VAFB SLC-4E, there are 4 successful landings and 6 failed landings

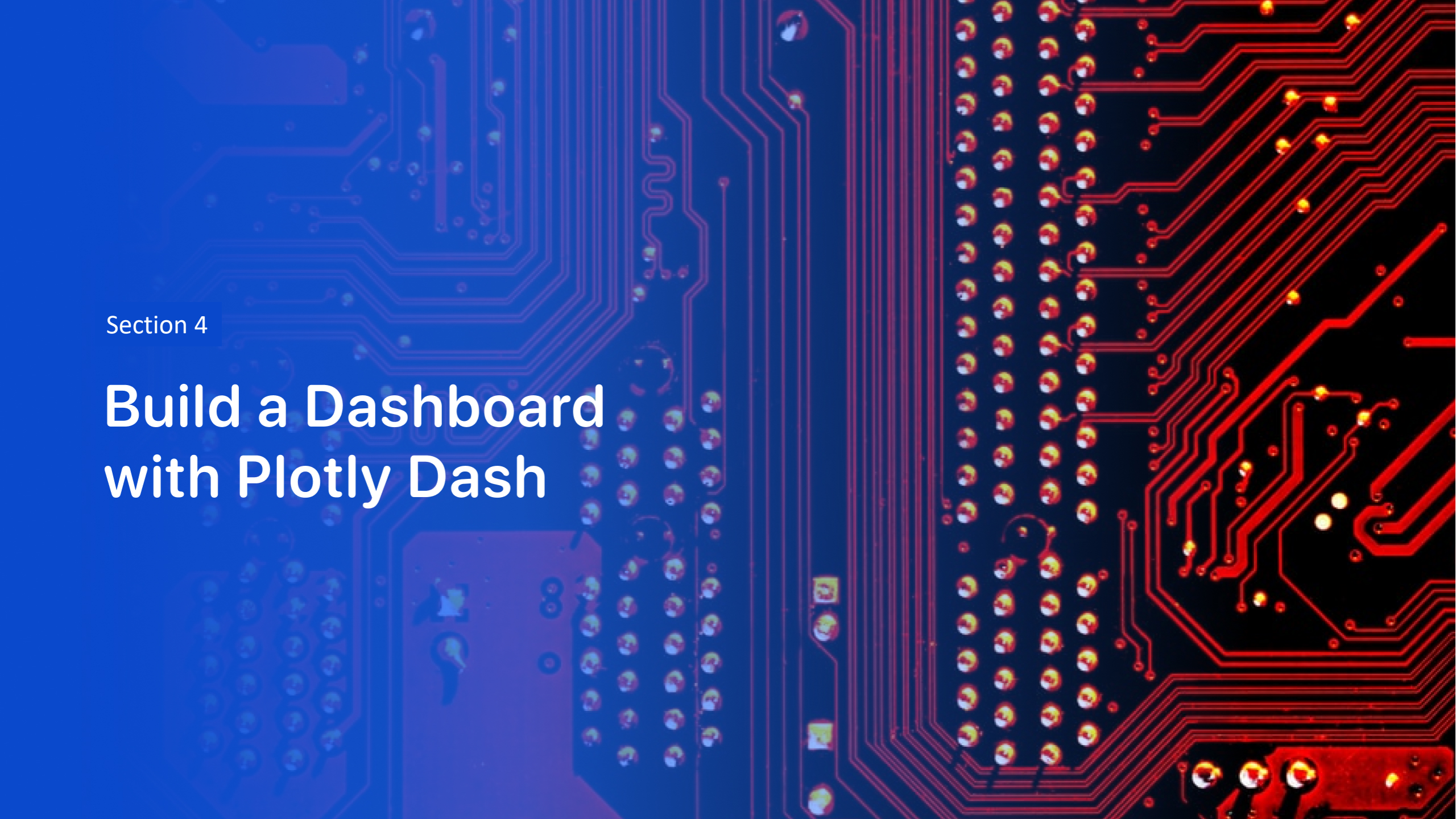


# Launch Site Proximity to Infrastructure and Coastline

- **KSC LC-39A: Key Spatial Features**
- **Railways Nearby:** Supports transport of heavy supplies.
- **Highways Accessible:** Facilitates personnel and logistics movement.
- **Coastal Proximity:** Launches occur over water, reducing risk from debris in case of failure.
- **Far from Cities:** Minimizes hazards to populated areas.
- **Key Insight:**

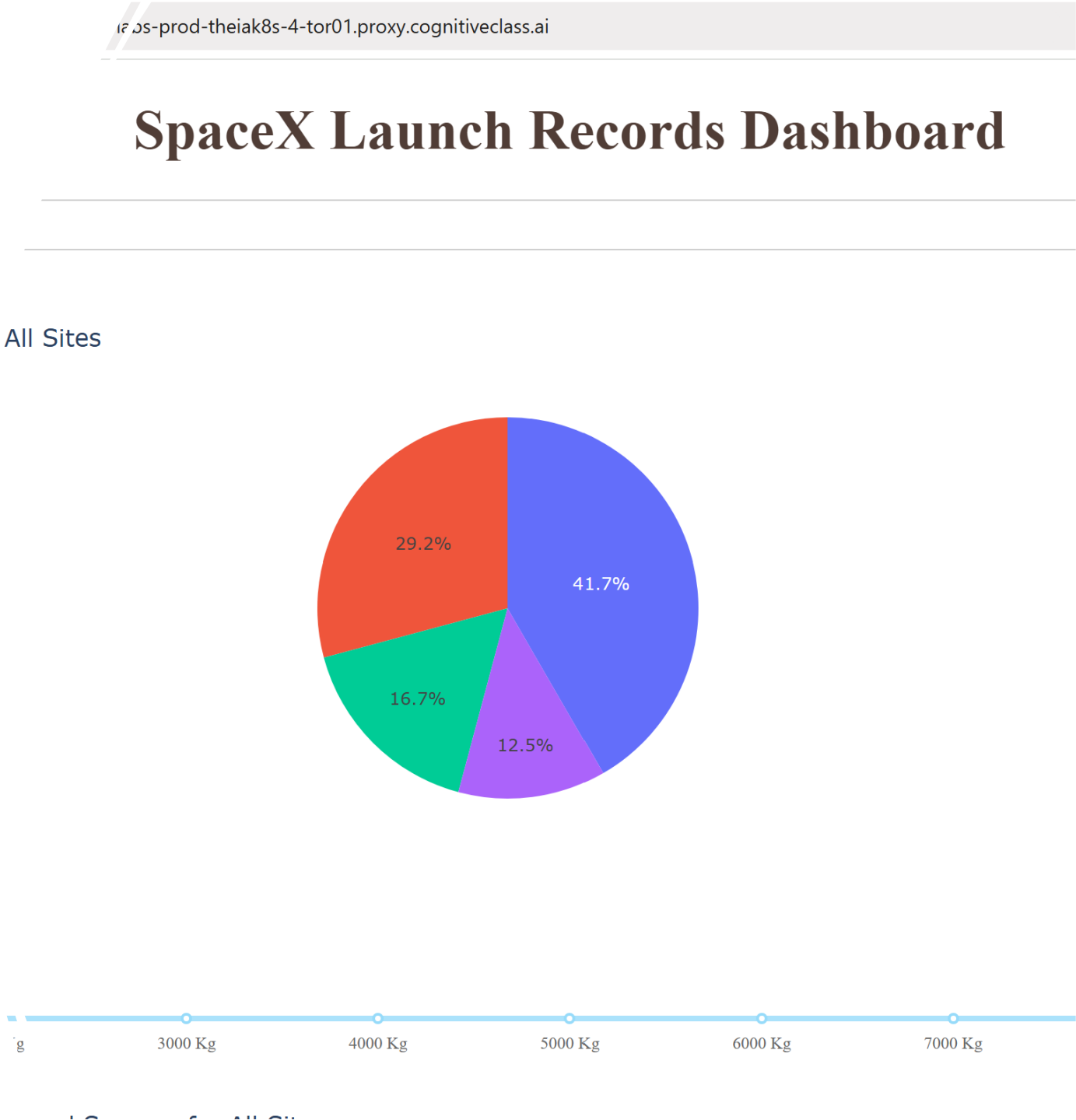
Launch sites are strategically positioned to optimize **safety, operational efficiency, and logistics**, with transport links and coastal access as essential factors in risk mitigatio




Section 4
Build a Dashboard with Plotly Dash

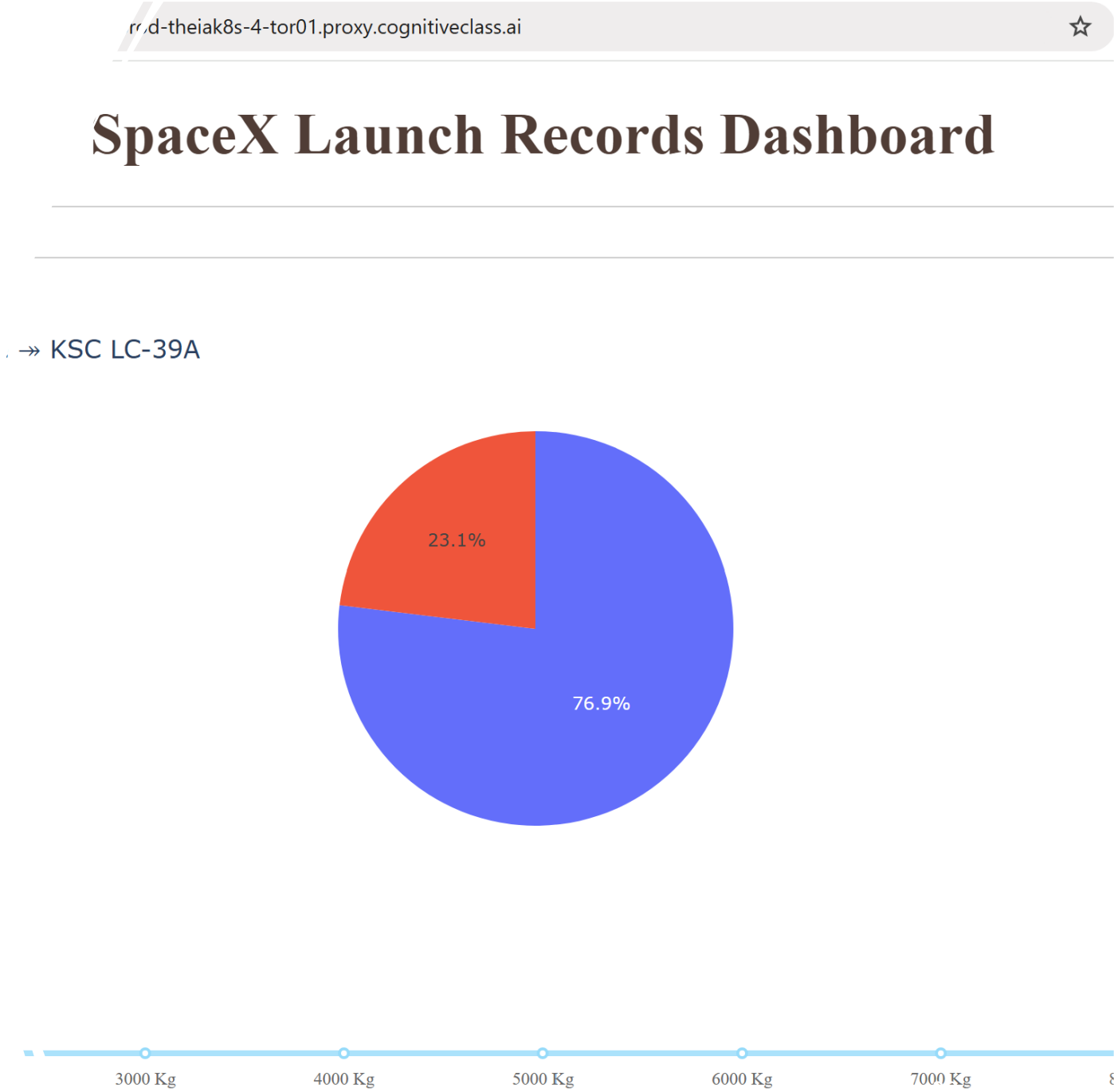
# Launch Success Distribution Across Sites

- **KSC LC-39A** leads with **41.7% of all successful launches**, highlighting its reliability as SpaceX's most successful site.
- **CCAFS LC-40** accounts for **29.2%**, making it the second most successful site.
- **VAFB SLC-4E** contributes **16.7%** to the total successes.
- **CCAFS SLC-40** has the smallest share, with **12.5%** of successful launches.



# Launch Site with the Highest Success Ratio

- The pie chart illustrates the launch outcomes at **KSC LC-39A**, the site with the **highest number of successful launches**.
- **Class 1 (Successful Launches)**: **76.9%** – represented by the dominant blue segment, highlighting the site's **high reliability and effectiveness**.
- **Class 0 (Unsuccessful Launches)**: **23.1%** – shown as the smaller red segment, indicating that while failures exist, they are relatively limited.



# Payload Mass vs. Success Across Booster Versions

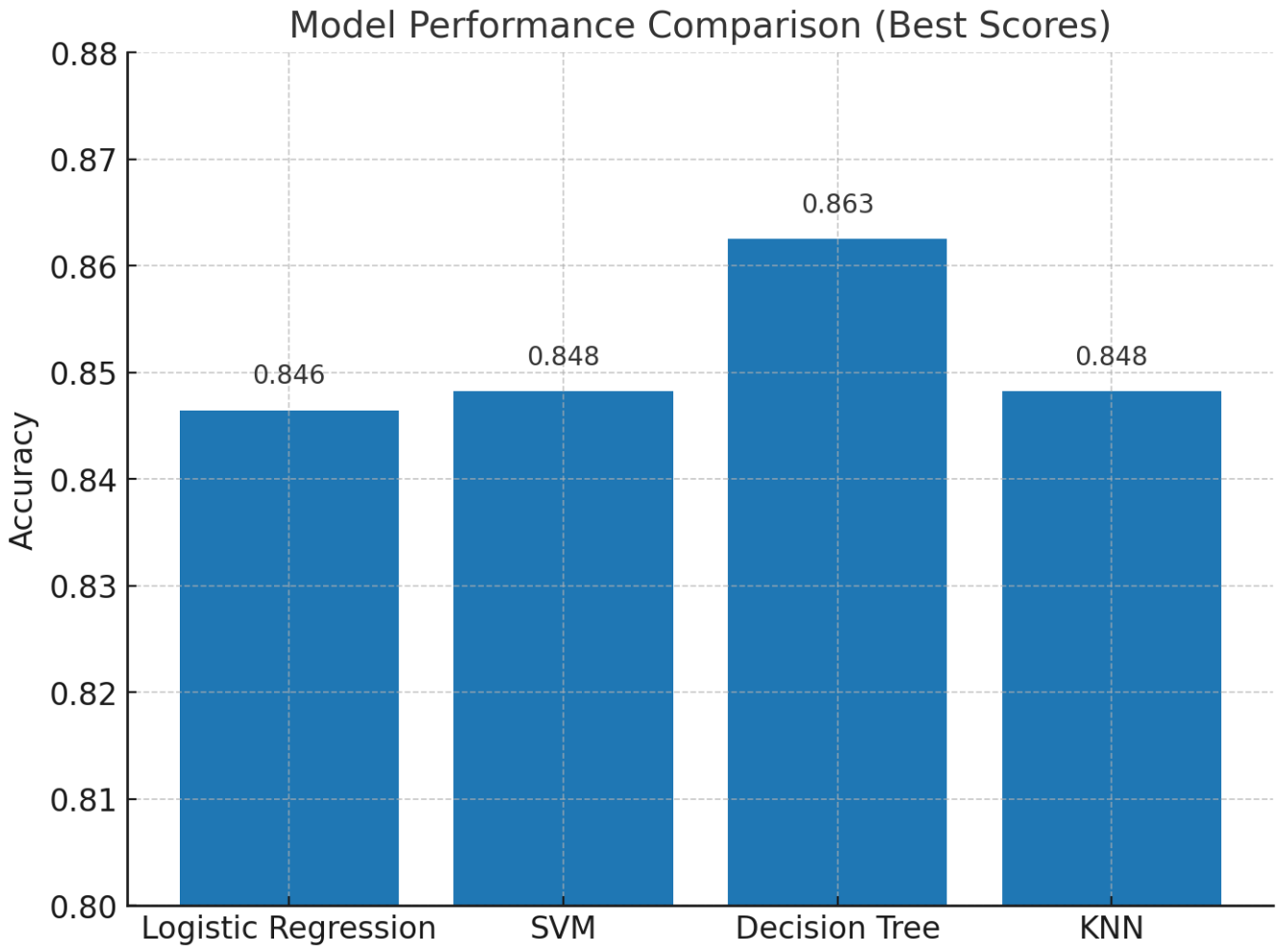
- “FT” booster is the most frequently used and shows a high success rate across different payload masses.
- “v1.0” booster has fewer launches, requiring more data to fully assess performance. Overall, the data suggests no clear link between higher payload masses and lower success rat

Section 5

# Predictive Analysis (Classification)

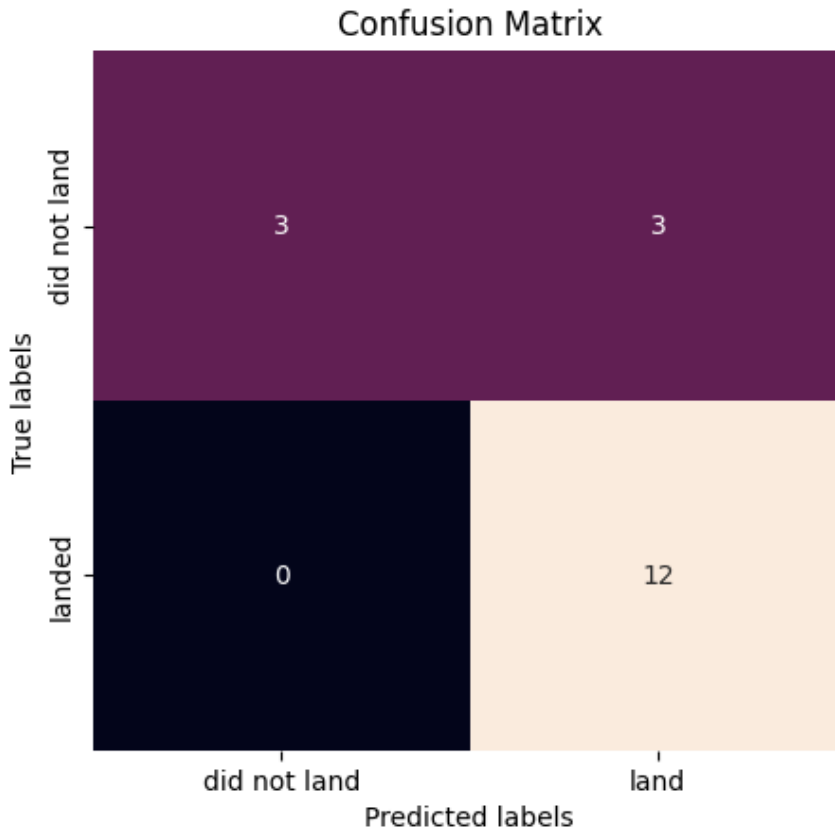
# Classification Accuracy

- The **Decision Tree** achieved the **highest accuracy (0.8625)**, outperforming all other models.
- **SVM (0.8482)** and **KNN (0.8482)** performed almost identically, slightly better than **Logistic Regression (0.8464)**.
- The differences between models are relatively small, but the **Decision Tree shows a clear edge** in predictive performance.



# Confusion Matrix

- Since all models performed identically on the test set, their confusion matrices are the same.
- The models correctly predicted **12 successful landings** when the true label was also successful.
- They correctly predicted **3 unsuccessful landings** when the true label was unsuccessful.
- However, they incorrectly predicted **3 successful landings** when the true label was unsuccessful (**false positives**).



# Conclusions

- **Task:** Develop a machine learning model for **SpaceY** to compete with SpaceX.
- **Goal:** Predict whether **Stage 1 will successfully land**,
- **Data Sources:** SpaceX API and Wikipedia (via web scraping).
- **Process:**
  - Labeled and stored data in a **DB2 SQL database**.
  - Built a **dashboard for visualization**.
  - Trained a machine learning model with an **accuracy of 83%**.
- **Application:** SpaceY can use this model to estimate Stage 1 landing success **before launch**, helping decide whether to proceed.
- **Next Step:** Collecting **more data** could improve accuracy and help identify the **best-performing model**.

# Thank you!