IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Outline

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

Executive Summary

- **Methodology Summary**
- Collected and integrated SpaceX Falcon 9 launch data using **API extraction, web scraping, and CSV datasets**.
- Performed **data cleaning, preprocessing, and exploratory data analysis (EDA)** using Python, Pandas, Matplotlib, and Seaborn.
- Conducted **SQL-based analysis** to answer business-driven questions related to launch sites, payload mass, and mission outcomes.
- Built **interactive visualizations** using **Folium** for geographic analysis and a **Plotly Dash dashboard** for real-time data exploration.
- Developed and evaluated **machine learning classification models** to predict first-stage landing success.

Executive Summary

- **Results Summary**
- Launch success rates were found to be strongly influenced by **launch site**, **payload mass**, and **booster version**.
- **KSC LC-39A** and **CCAFS SLC-40** demonstrated the highest success rates among all launch sites.
- Payloads in the **medium-to-high mass range** showed higher probabilities of successful landings.
- Newer booster versions (e.g., **FT, B4, B5**) achieved significantly better landing success compared to earlier versions.
- Among tested models, the **best-performing machine learning model** achieved strong predictive accuracy in identifying landing success, supporting cost estimation and competitive bidding decisions.

Introduction

- **Project Background & Context**
- SpaceX has significantly reduced the cost of space launches by **reusing the Falcon 9 first-stage booster**.
- Traditional launch providers charge over **$165 million per launch**, while SpaceX advertises Falcon 9 launches at approximately **$62 million**.
- The key factor behind this cost advantage is the **successful recovery and reuse of the first-stage booster**.
- Understanding the conditions that lead to successful booster landings is valuable for **cost estimation, mission planning, and competitive bidding** in the aerospace industry.

Introduction

Problem Statement / Questions to Answer

- What factors most strongly influence the **success of Falcon 9 first-stage landings**?
- How do **launch site**, **payload mass**, **orbit type**, and **booster version** affect mission outcomes?
- Which launch sites demonstrate the **highest success rates**?
- Can we build a **predictive machine learning model** to accurately determine whether a booster will land successfully?
- How can these insights help **competing launch providers** make data-driven bidding decisions?

Section 1

Methodology

Methodology

Executive Summary
In this project, we analyzed SpaceX Falcon 9 launch data to predict the success of first-stage landings, which directly impacts launch costs and competitiveness.
Data Collection Methodology
Launch data was collected from multiple sources. SpaceX launch and rocket details were retrieved using the **SpaceX REST API**, while historical launch records, payload information, and landing outcomes were obtained through **web scraping from Wikipedia**.
Data Wrangling
The collected data was cleaned and processed by filtering Falcon 9 launches, handling missing values, standardizing formats, and creating a binary target variable (`class`) representing successful and unsuccessful landings.
Exploratory Data Analysis (EDA)
EDA was performed using **visualization techniques** and **SQL queries** to identify relationships between launch success and factors such as payload mass, launch site, orbit type, and booster version.
Interactive Visual Analytics
Interactive geographical analysis was conducted using **Folium** to study launch site locations and proximities. An interactive **Plotly Dash dashboard** was developed to explore launch success trends across launch sites and payload ranges.
Github:https://github.com/ayesha-ml/SpaceX-Launch-Analysis.git

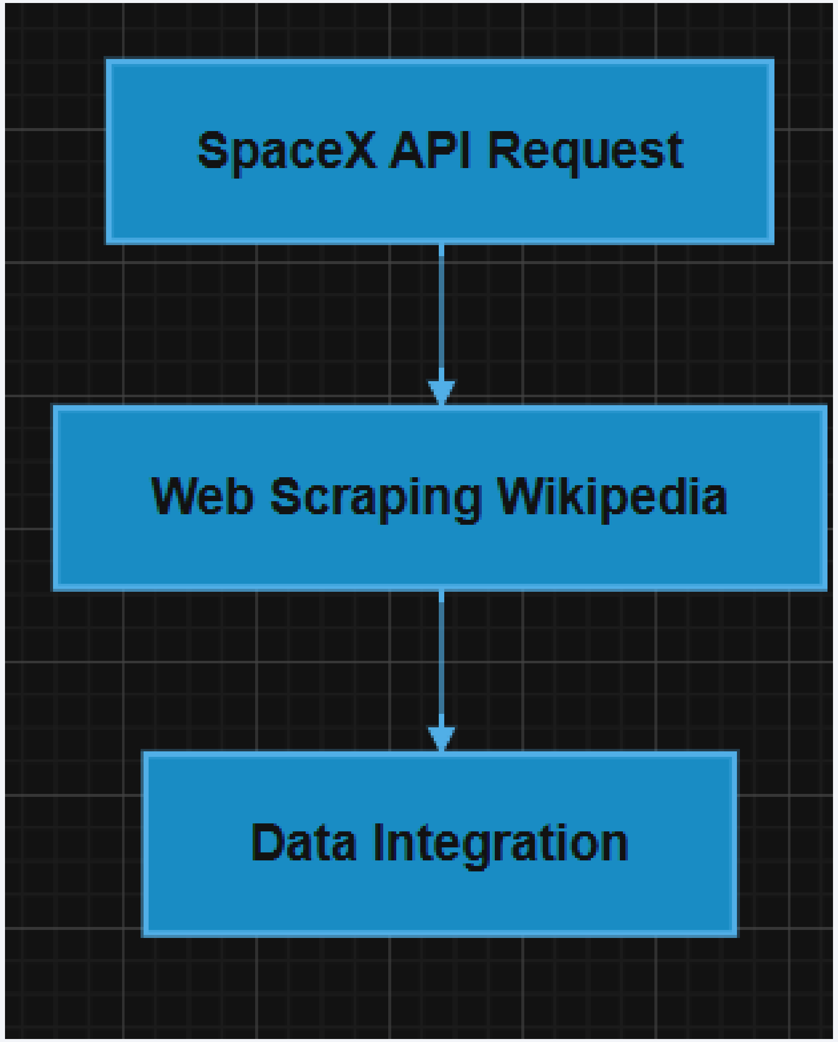
Data Collection

Step 1: SpaceX API Request

Step 2: Web Scraping

Wikipedia

Step 3: Data Integration

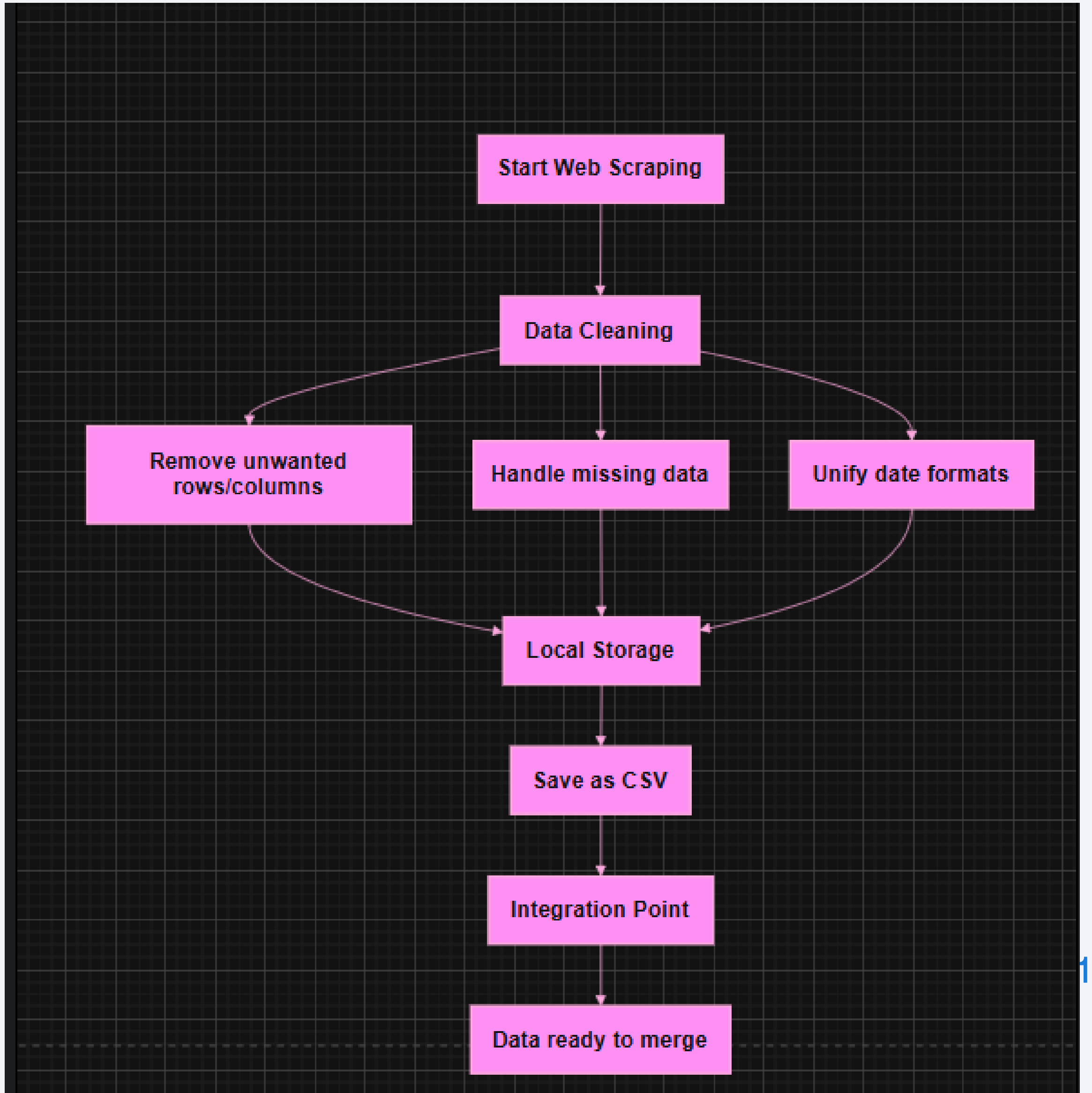


Data Collection – Scraping

- **Objective:** To collect additional SpaceX launch details not available via the API.
- **Source:** Wikipedia pages listing SpaceX launches.
- Workflow Steps:
  - **Access HTML pages** using Python libraries like requests.
  - **Identify relevant tables** containing launch details such as date, payload, booster version, and orbit.
  - **Extract HTML tables** using BeautifulSoup parsing.
  - **Convert HTML tables to DataFrames** using pandas.read_html() for structured analysis.
- **Notes/Key Phrases:**
- HTML parsing, table extraction, DataFrame conversion, automated data collection.

Data Collection – Scraping

- **Data Cleaning:** Remove unwanted rows/columns, handle missing data, unify date formats.
- **Local Storage:** Save cleaned DataFrames as CSV for easy access and reproducibility.
- **Advantages:**
  - Allows collection of historical data not available via API.
  - Supports updates by re-running the scraper.
- **Integration Point:** Prepared data ready to merge with SpaceX API dataset for unified analysis.
- **Notes/Key Phrases:**
- Data cleaning, normalization, CSV storage, reproducibility, merge-ready.
- Github: https://github.com/ayesha-ml/SpaceX-Launch-Analysis/blob/main/jupyter-labs-webscraping.ipynb

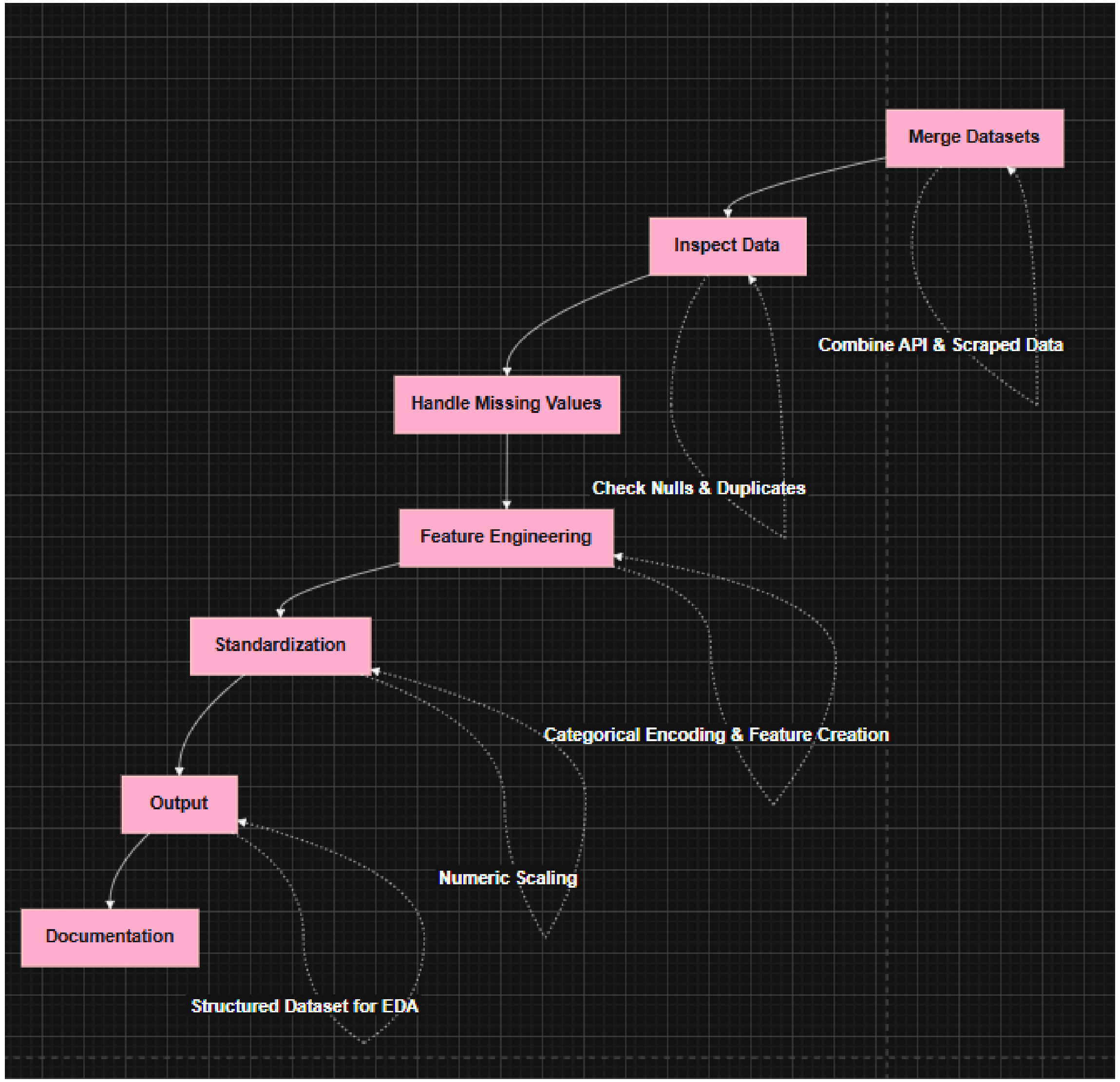


Data Wrangling

- **Objective:** To prepare raw datasets for analysis and modeling.
- **Input Sources:**
  - SpaceX REST API data
  - Wikipedia scraped data
- **Workflow Steps:**
  - **Merge Datasets:** Combine API and scraped datasets on launch date, mission, or other identifiers.
  - **Inspect Data:** Check for null values, duplicates, and inconsistencies.
  - **Handle Missing Values:** Impute, remove, or flag missing fields depending on context.
- **Notes/Key Phrases:**
- Dataset integration, missing value treatment, duplicate removal, consistency check.

Data Wrangling

- **Feature Engineering:**
  - Encode categorical variables (Launch Site, Booster Version Category) for analysis.
  - Create derived columns if needed (e.g., success rate, payload bins).
- **Standardization:**
  - Scale numeric features like Payload Mass for uniformity across models.
- **Output:**
  - Final cleaned and structured dataset ready for **EDA, visualization, and predictive modeling**.
- **Documentation:** Saved notebooks with GitHub links ensure reproducibility and peer review.
- **Notes/Key Phrases:**
- Categorical encoding, numeric scaling, feature creation, structured dataset, reproducibility.
- Github:https://github.com/ayesha-ml/SpaceX-Launch-Analysis/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



EDA with Data Visualization

•**Purpose:**

Perform an initial investigation of SpaceX launch data to uncover patterns, trends, and relationships between variables such as launch site, payload, booster version, and mission outcome.

•**Key Visualizations Created:**

1. **Bar Charts:** Showed the number of successful launches per launch site to identify which site had the highest success rate.
2. **Pie Charts:** Displayed proportions of success vs failure for all sites and for individual launch sites.
3. **Histograms:** Examined the distribution of payload masses across launches to detect common payload ranges.
4. **Scatter Plots:** Analyzed the relationship between payload mass and mission outcome (success/failure).

EDA with Data Visualization

- **Why These Charts Were Used:**
- **Bar and Pie Charts:** Effective for categorical comparisons and proportion analysis.
- **Histograms:** Useful to observe the distribution of numerical data.
- **Scatter Plots:** Excellent for spotting correlations between numerical variables.
- **Outcome:**
- Identified the launch sites with the highest and lowest success rates.
- Observed potential correlations between payload mass and launch success.
- Supported later predictive modeling with insights on key variables.

EDA with SQL

- **Purpose:**

  Use SQL to query and summarize the SpaceX dataset efficiently, exploring key metrics and supporting insights derived from visualizations.

- **Key Queries Performed:**
  - **Launch Site Analysis:** List all launch sites and count total launches per site.
  - **Success/Failure Analysis:** Count the number of successful and failed launches for each site.
  - **Payload Analysis:** Find total and average payload mass carried by different booster versions.
  - **Temporal Analysis:** Extract launch months and years to explore trends over time.
  - **Filter Queries:** Identify booster versions that carried the maximum payload and examine outcomes.

EDA with SQL

- **Why SQL Was Used:**
  - Efficient filtering and aggregation for large datasets.
  - Structured approach to answer specific questions quickly.
  - Enables easy integration with other Python analyses and dashboards.
- **Outcome:**
  - Provided numerical summaries that validated and complemented visual EDA findings.
  - Allowed detailed comparisons of booster performance and launch outcomes.
- **Reference:**https://github.com/ayesha-ml/SpaceX-Launch-Analysis/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

•**Purpose:**

Transform static EDA insights into interactive, dynamic visualizations to explore SpaceX launch data more intuitively.

•**Folium Map:**

•**Objective:** Map all launch sites and visualize success vs. failure outcomes.

•**Features:**

- •Markers for each launch site.
- •Color-coded markers showing successful and failed launches.
- •Popup information for payload, booster version, and date.

•**Outcome:** Quickly identify spatial patterns and site-specific success trends.

Build a Dashboard with Plotly Dash

•**Purpose:**

Transform static EDA insights into interactive, dynamic visualizations to explore SpaceX launch data more intuitively.

•**Folium Map:**

•**Objective:** Map all launch sites and visualize success vs. failure outcomes.

•**Features:**

- •Markers for each launch site.
- •Color-coded markers showing successful and failed launches.
- •Popup information for payload, booster version, and date.

•**Outcome:** Quickly identify spatial patterns and site-specific success trends.

Build a Dashboard with Plotly Dash

- **Plotly Dash Application:**
- **Objective:** Build an interactive dashboard for payload and mission outcome analysis.
- **Features:**
    - Dropdown menu to select a launch site (all or specific site).
    - Pie chart to show success/failure counts based on selected site.
    - Range slider to filter payload mass.
    - Scatter plot to visualize correlation between payload and success, color-coded by booster version.
- **Outcome:** Users can dynamically explore trends, correlations, and site-specific insights.
- **References:**
- GitHub URL :https://github.com/ayesha-ml/SpaceX-Launch-Analysis/blob/main/lab_jupyter_launch_site_location.ipynb

Predictive Analysis (Classification)

- **Objective:** Predict the first stage landing outcome (Success or Failure) of SpaceX Falcon 9 launches based on historical launch data.
- **1. Data Preprocessing**
- Standardized features to ensure numerical variables contribute equally to model learning.
- Split data into **training (70%)** and **test sets (30%)** for model validation.
- Handled categorical variables using encoding techniques (e.g., one-hot encoding).
- **2. Model Selection**
- Explored multiple classification algorithms suitable for binary outcomes:
  - **Support Vector Machines (SVM)**
  - **Decision Trees**
  - **K-Nearest Neighbors (KNN)**
- Selected models based on performance, interpretability, and project requirements.

Predictive Analysis (Classification)

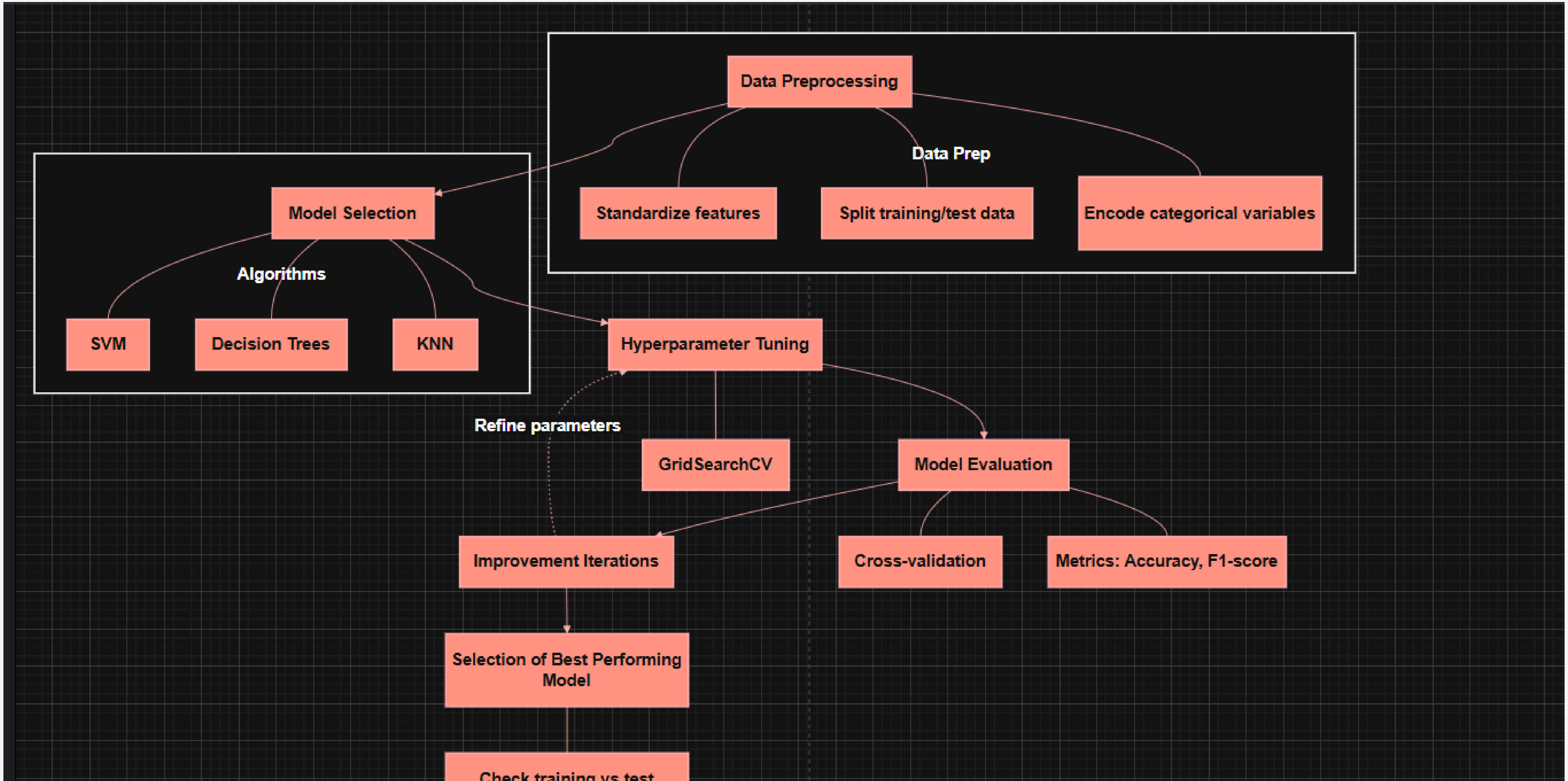
- **3. Hyperparameter Tuning**
- Applied **GridSearchCV** to systematically identify optimal hyperparameters.
- Tuned key parameters:
  - **SVM:** C, kernel type
  - **Decision Trees:** max_depth, min_samples_split
  - **KNN:** n_neighbors
- Goal: Maximize predictive accuracy while avoiding overfitting.
-

- **4. Model Evaluation**
- Used **cross-validation** to ensure robust and generalizable performance.
- Evaluated models using multiple metrics:
  - **Accuracy** – overall correctness of predictions
  - **Precision** – success prediction correctness
  - **Recall** – correctly predicted successes from all actual successes
  - **F1-score** – balance between precision and recall

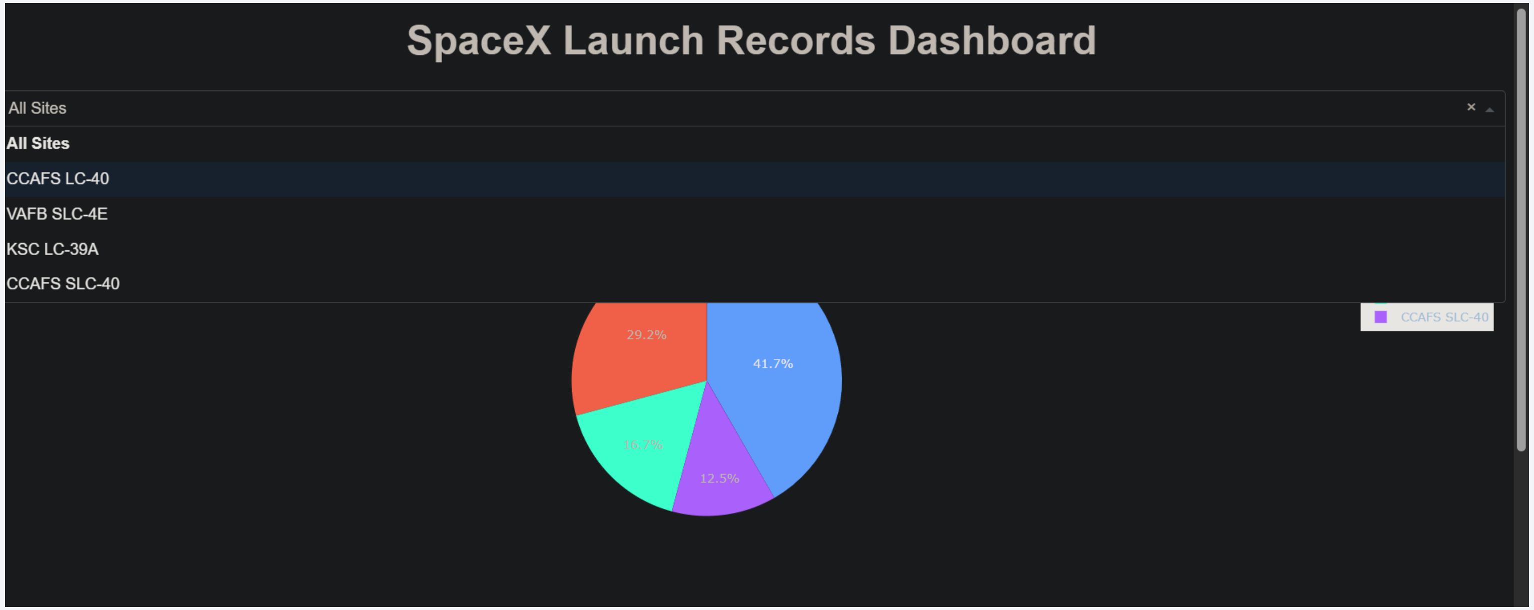
Predictive Analysis (Classification)

- **5. Improvement Iterations**
- Iteratively adjusted models based on validation performance.
- Fine-tuned hyperparameters to maximize predictive reliability.
- Compared model performances across algorithms to identify strengths and weaknesses.
-
- **6. Selection of Best Performing Model**
- Identified the model with the **highest test set accuracy** as the final model.
- Ensured consistency between training and test performance to avoid overfitting.
- Final model chosen based on **accuracy, reliability, and real-world applicability**.

Predictive Analysis (Classification)



Results


SpaceX Launch Records Dashboard
All Sites
All Sites
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
CCAFS SLC-40
29.2%
41.7%
16.7%
12.5%

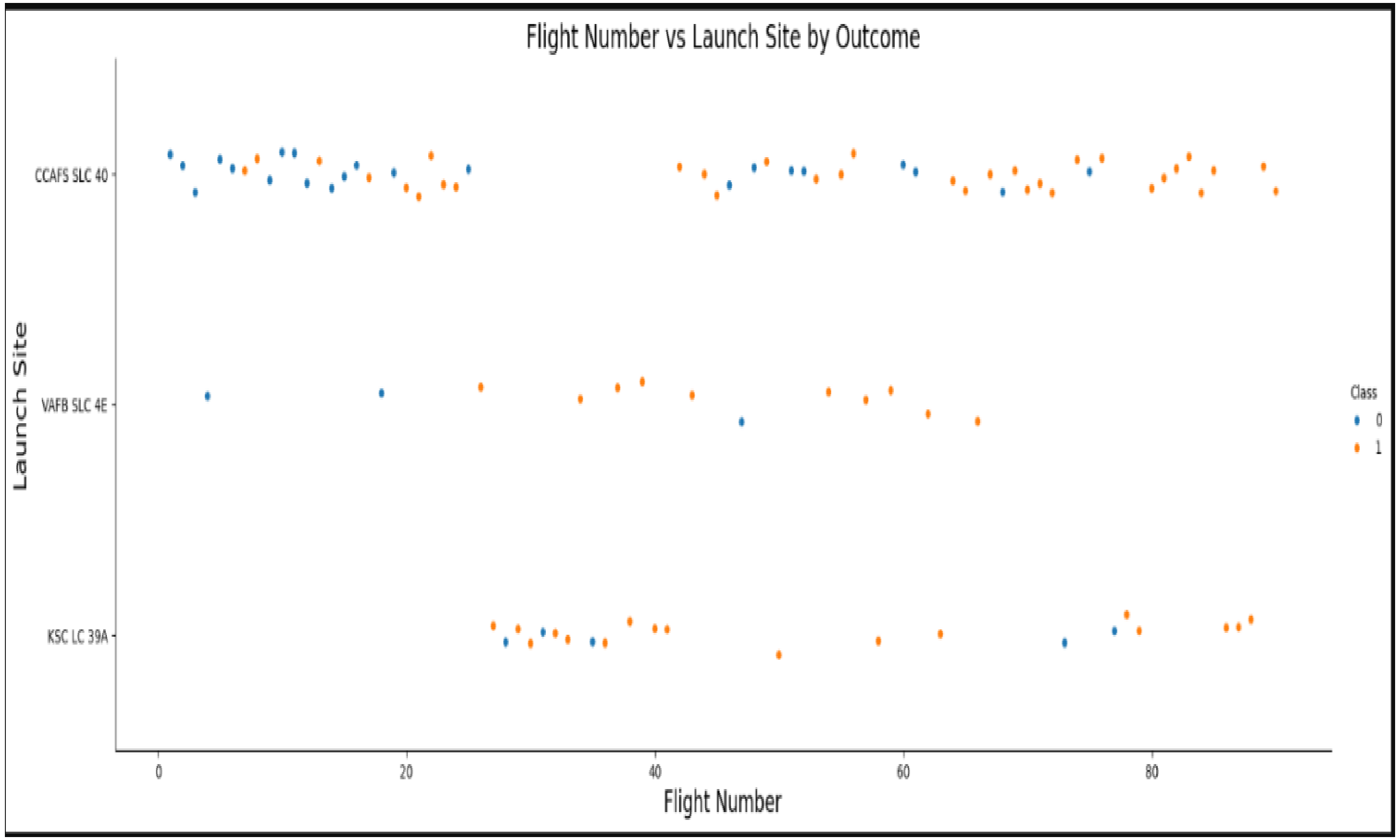

Payload range (Kg):
0
2500
5000
7500
Payload vs Launch Outcome
Success
Failure
class_label
v1.1
FT
B4
B5

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

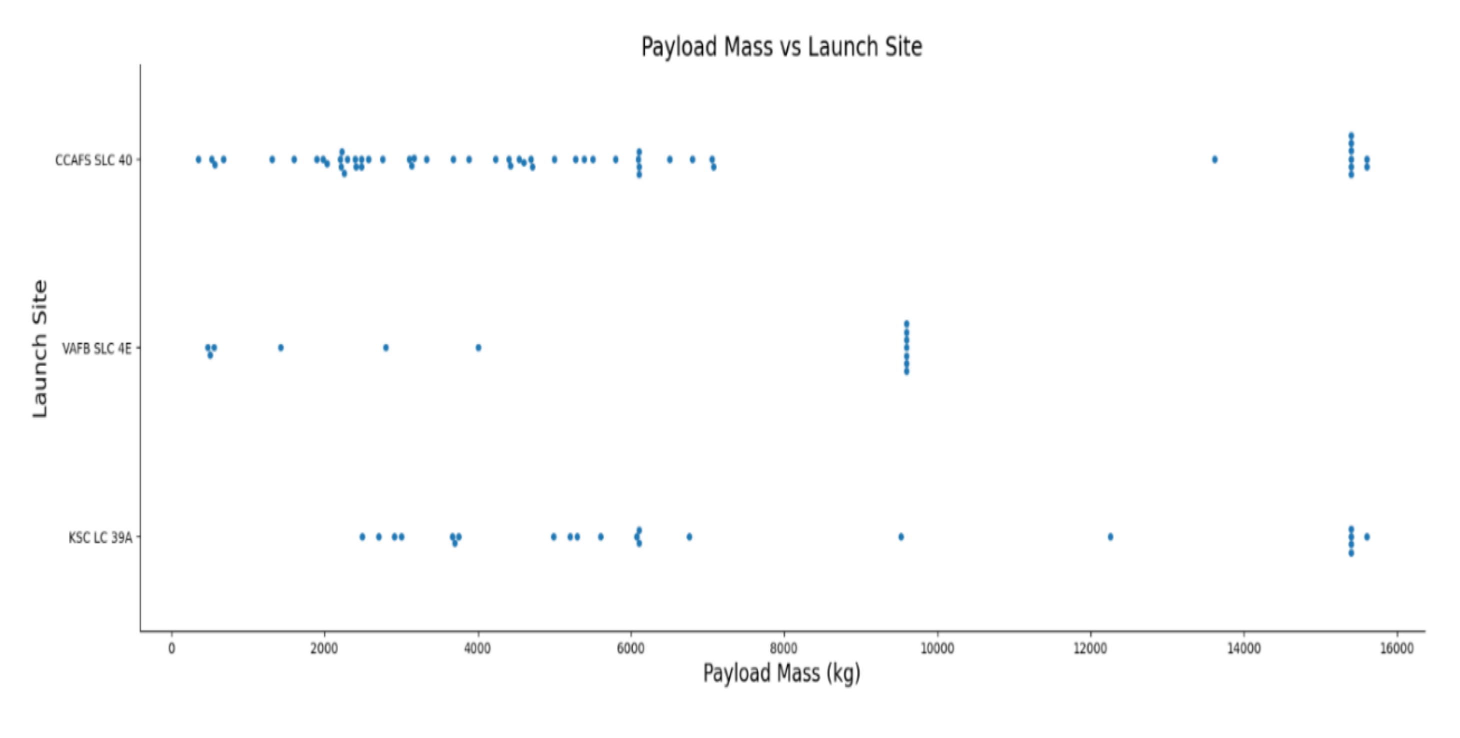
- **Varied Outcomes at Key Launch Sites:** Both CCAFS SLC-40 and KSC LC-39A exhibit a combination of successful (orange) and failed (blue) landings, suggesting that factors beyond the launch site itself likely affect landing success.
- **Steady Activity Across Flight Numbers:** Launches span a wide range of flight numbers at all sites, indicating consistent launch activity over time without a clear trend of increasing or decreasing landing success.



Payload vs. Launch Site

•**Payload Distribution:** Most launches from CCAFS SLC-40 carry payloads under 10,000 kg, whereas VAFB SLC-4E and KSC LC-39A handle a broader range of payload masses, reflecting diverse mission profiles.

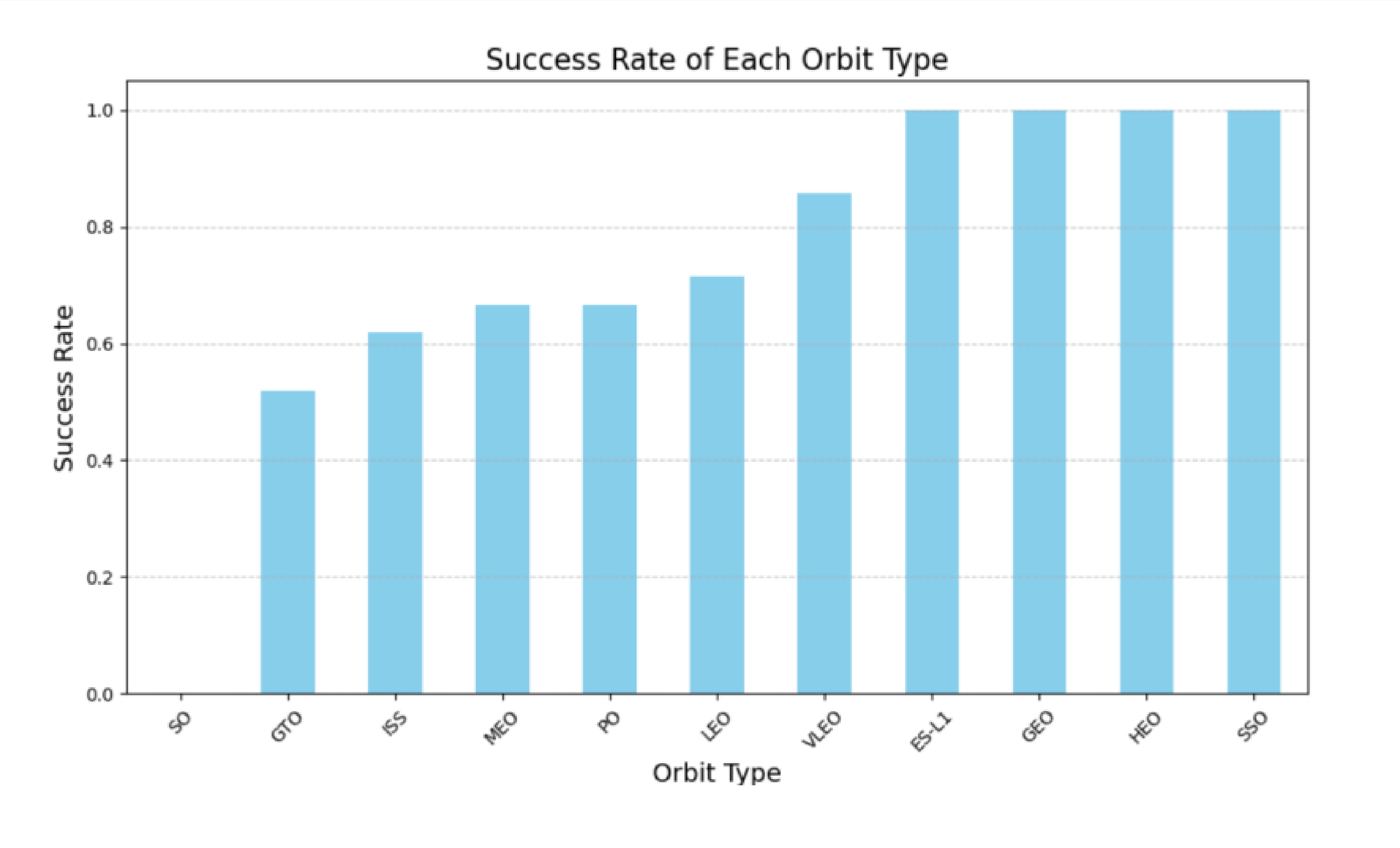
•**High-Capacity Launches:** KSC LC-39A frequently handles heavier payloads, with multiple launches exceeding 15,000 kg, highlighting its capability for high-capacity missions.



Success Rate vs. Orbit Type

•**High Success Rates:** Missions to VLEO, ES-L1, GEO, HEO, and SSO orbits have achieved perfect success rates, indicating these orbits are highly favorable for successful first-stage landings.

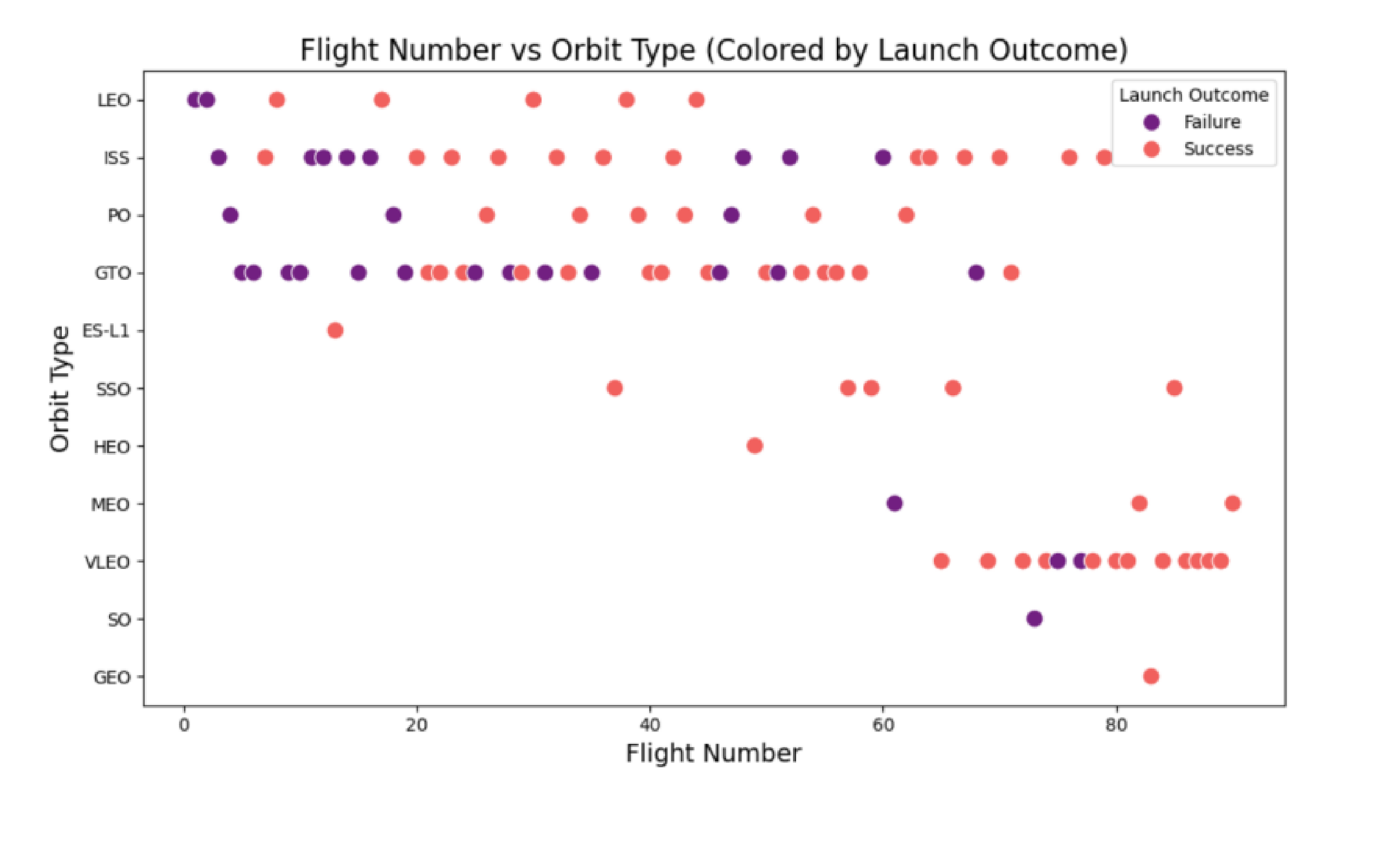
•**Lower Success Rate for GTO:** GTO missions show noticeably lower success rates, suggesting these launches face greater challenges or complexities.



Flight Number vs. Orbit Type

•**Improved Success Over Time:** Falcon 9 launch success rates increase with higher flight numbers, showing that experience and iterative improvements lead to better outcomes.

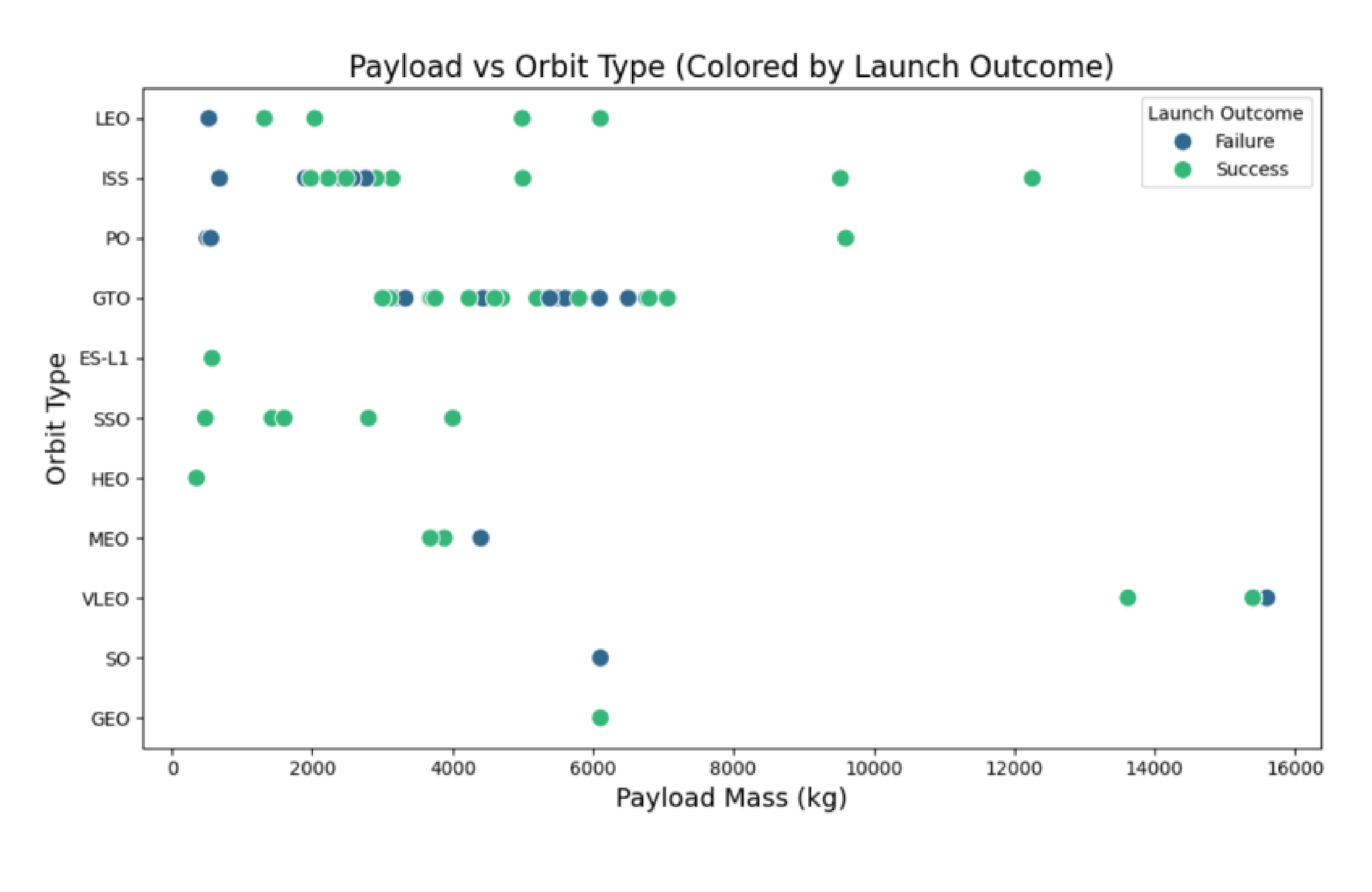
•**Orbit-Specific Performance:** Early flights to GTO and ISS orbits had mixed results, but recent missions demonstrate higher success rates, reflecting advancements in mission planning and execution.



Payload vs. Orbit Type

•Success by Payload: Landings are generally more successful for payloads under 6,000 kg across all orbit types.

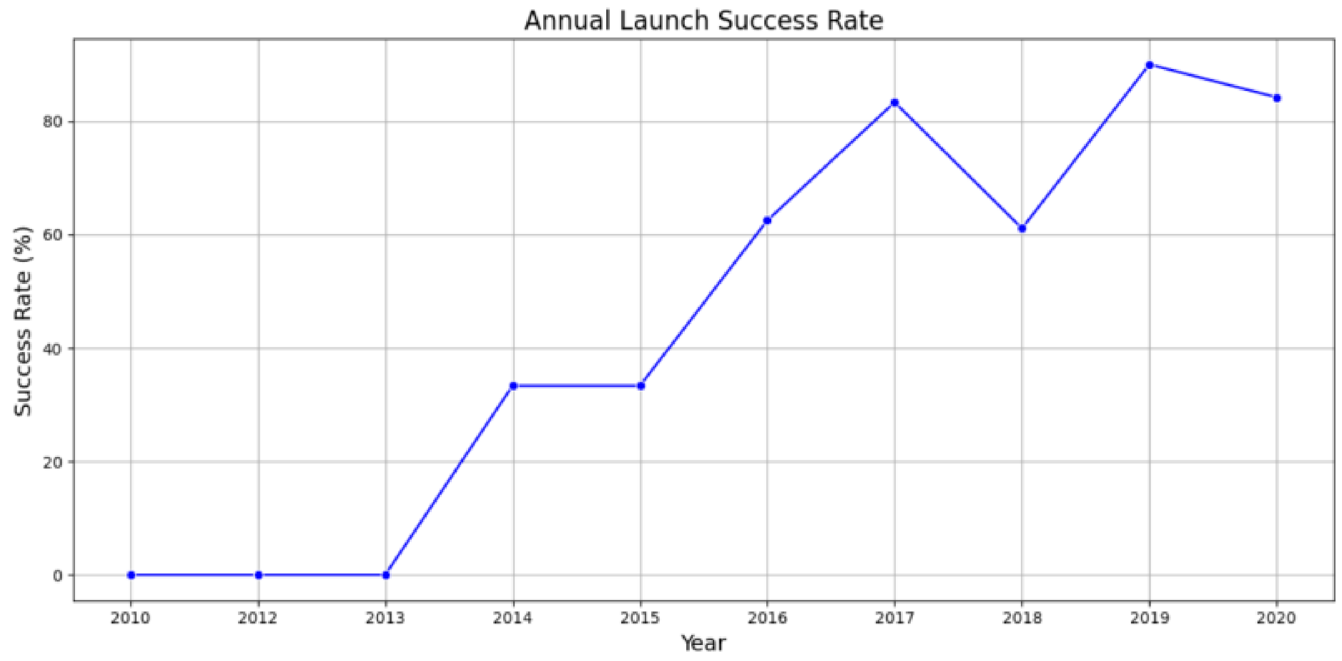
•Challenges with Heavy Payloads: Payloads above 10,000 kg show a mix of successes and failures, indicating greater difficulty with heavier missions.



Launch Success Yearly Trend

•**Rising Annual Success Rate:** Falcon 9 launch success has steadily improved since 2013, surpassing 80% by 2020.

•**Overall Reliability Trend:** Despite a slight dip in 2018, the general trend shows increasing reliability and consistent successful launches over the years.



All Launch Site Names

Task 1

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

[15]:
```python
query = "SELECT DISTINCT Launch_Site FROM SPACEXTABLE;"
cur.execute(query)
unique_sites = cur.fetchall()

# Display the results
for site in unique_sites:
    print(site[0])
```

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

Launch Site Names Begin with 'CCA'

Task 2

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

[16]:
```
query = """
SELECT *
FROM SPACEXTABLE
WHERE Launch_Site LIKE 'CCA%'
LIMIT 5;
"""
cur.execute(query)
results = cur.fetchall()
for row in results:
    print(row)
```

```
('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', 'Su
ess', '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

Task 3

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

[20]:
```
query = """
SELECT SUM(PAYLOAD_MASS__KG_) AS Total_Payload_Mass
FROM SPACEXTABLE
WHERE Customer = 'NASA (CRS)';
"""
cur.execute(query)
total_payload_mass = cur.fetchone()[0]
print("Total Payload Mass carried by NASA (CRS) boosters:", total_payload_mass, "kg")
```

```
Total Payload Mass carried by NASA (CRS) boosters: 45596 kg
```

Average Payload Mass by F9 v1.1

Task 4

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

```
query = """
SELECT AVG(PAYLOAD_MASS__KG_) AS Avg_Payload_Mass
FROM SPACEXTABLE
WHERE Booster_Version = 'F9 v1.1';
"""

cur.execute(query)
avg_payload = cur.fetchone()[0]
print("Average Payload Mass for F9 v1.1:", avg_payload, "kg")
```

```
Average Payload Mass for F9 v1.1: 2928.4 kg
```

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

```
[22]: query = """
SELECT MIN(Date) AS First_Successful_GroundPad_Landing
FROM SPACEXTABLE
WHERE Landing_Outcome = 'Success (ground pad)';
"""
cur.execute(query)
first_success_date = cur.fetchone()[0]
print("First successful landing on ground pad:", first_success_date)
```

```
First successful landing on ground pad: 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 40

```
query = """
SELECT Booster_Version, Payload, PAYLOAD_MASS__KG_, Launch_Site, Landing_Outcome
FROM SPACEXTABLE
WHERE Landing_Outcome = 'Success (drone ship)'
  AND PAYLOAD_MASS__KG_ > 4000
  AND PAYLOAD_MASS__KG_ < 6000;
"""
cur.execute(query)
boosters = cur.fetchall()
for b in boosters:
    print(b)
```

```
('F9 FT B1022', 'JCSAT-14', 4696, 'CCAFS LC-40', 'Success (drone ship)')
('F9 FT B1026', 'JCSAT-16', 4600, 'CCAFS LC-40', 'Success (drone ship)')
('F9 FT  B1021.2', 'SES-10', 5300, 'KSC LC-39A', 'Success (drone ship)')
('F9 FT  B1031.2', 'SES-11 / EchoStar 105', 5200, 'KSC LC-39A', 'Success (drone ship)')
```

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

17]:

```
query = """
SELECT Mission_Outcome, COUNT(*) AS Total
FROM SPACEXTABLE
GROUP BY Mission_Outcome;
"""
cur.execute(query)
results = cur.fetchall()
for row in results:
    print(row)
```

```
('Failure (in flight)', 1)
('Success', 98)
('Success ', 1)
('Success (payload status unclear)', 1)
```

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.

```
query = """
SELECT Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
"""
cur.execute(query)
results = cur.fetchall()
for row in results:
    print(row)
```

```
('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.

```
query = """
SELECT
    substr(Date, 6, 2) AS Month,
    Booster_Version,
    Launch_Site,
    Landing_Outcome
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE '%drone ship%'
  AND substr(Date, 1, 4) = '2015';
"""
cur.execute(query)
results = cur.fetchall()
for row in results:
    print(row)
```

```
('01', 'F9 v1.1 B1012', 'CCAFS LC-40', 'Failure (drone ship)')
('04', 'F9 v1.1 B1015', 'CCAFS LC-40', 'Failure (drone ship)')
('06', 'F9 v1.1 B1018', 'CCAFS LC-40', 'Precluded (drone ship)')
```

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.

```
query = """
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;
"""
cur.execute(query)
landing_ranks = cur.fetchall()
for r in landing_ranks:
    print(r)
```

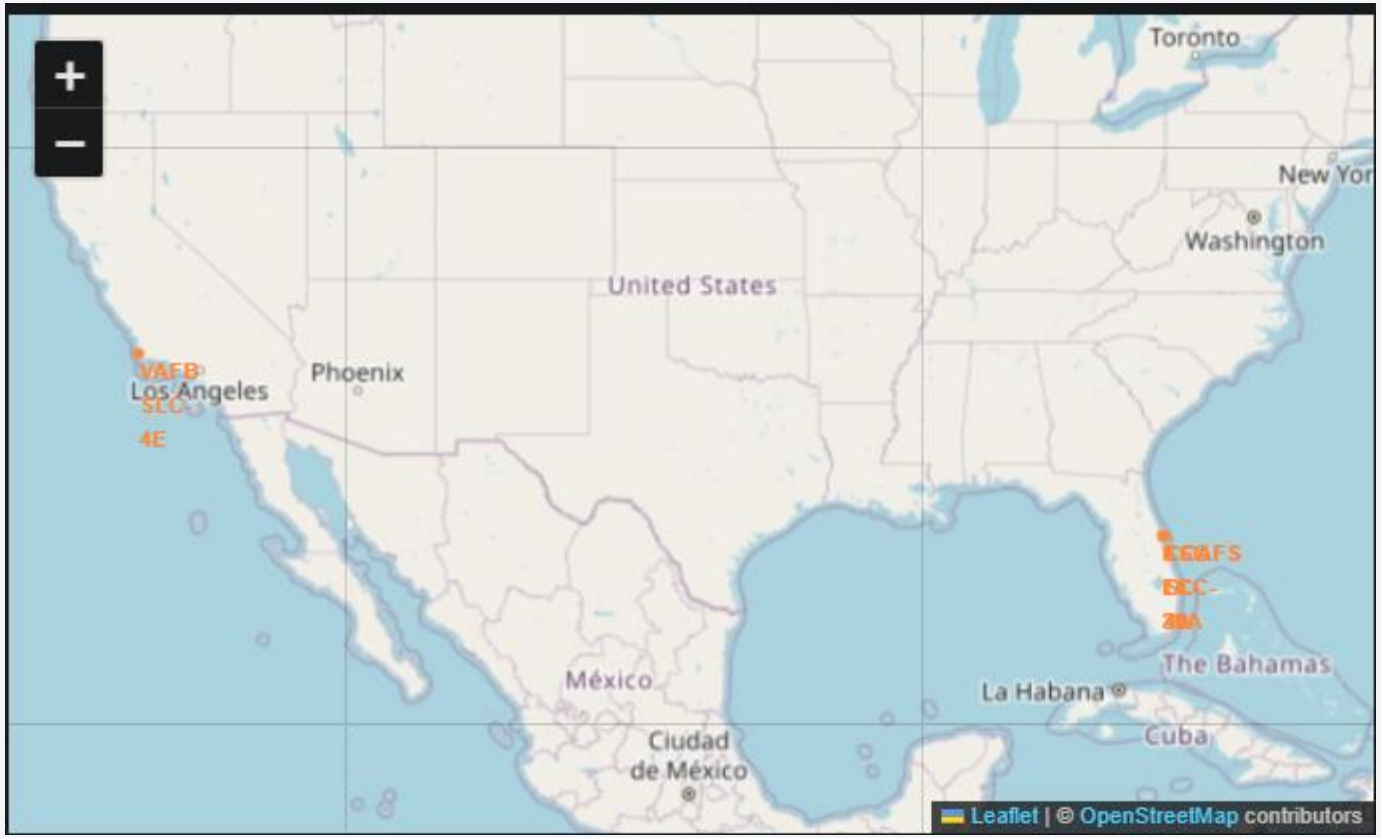
```
('No attempt', 10)
('Success (drone ship)', 5)
('Failure (drone ship)', 5)
('Success (ground pad)', 3)
('Controlled (ocean)', 3)
('Uncontrolled (ocean)', 2)
('Failure (parachute)', 2)
('Precluded (drone ship)', 1)
```

Section 3

Launch Sites Proximities Analysis

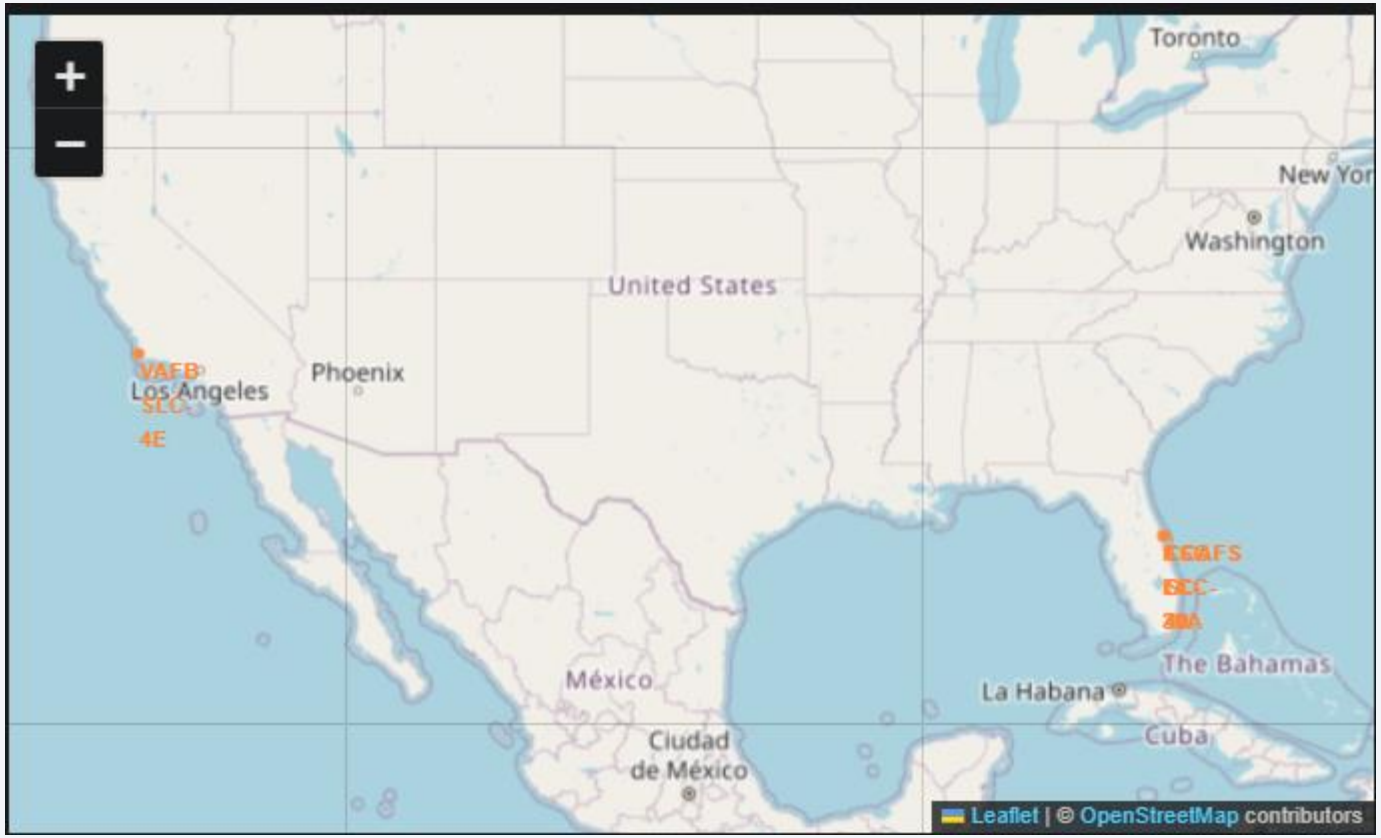
Global Overview of SpaceX Launch Sites

- **Important Elements:**
- **Launch Site Markers:** The map identifies four key SpaceX launch locations across the United States: **KSC LC-39A**, **CCAFS LC-40**, **CCAFS SLC-40** (all in Florida), and **VAFB SLC-4E** (in California).
- **Marker Clustering:** The cluster icons (showing 10 launches on the West Coast and 46 on the East Coast) group individual records to maintain map clarity at high zoom levels.
- **Coastline Proximity:** All sites are strategically positioned directly on the coastline.



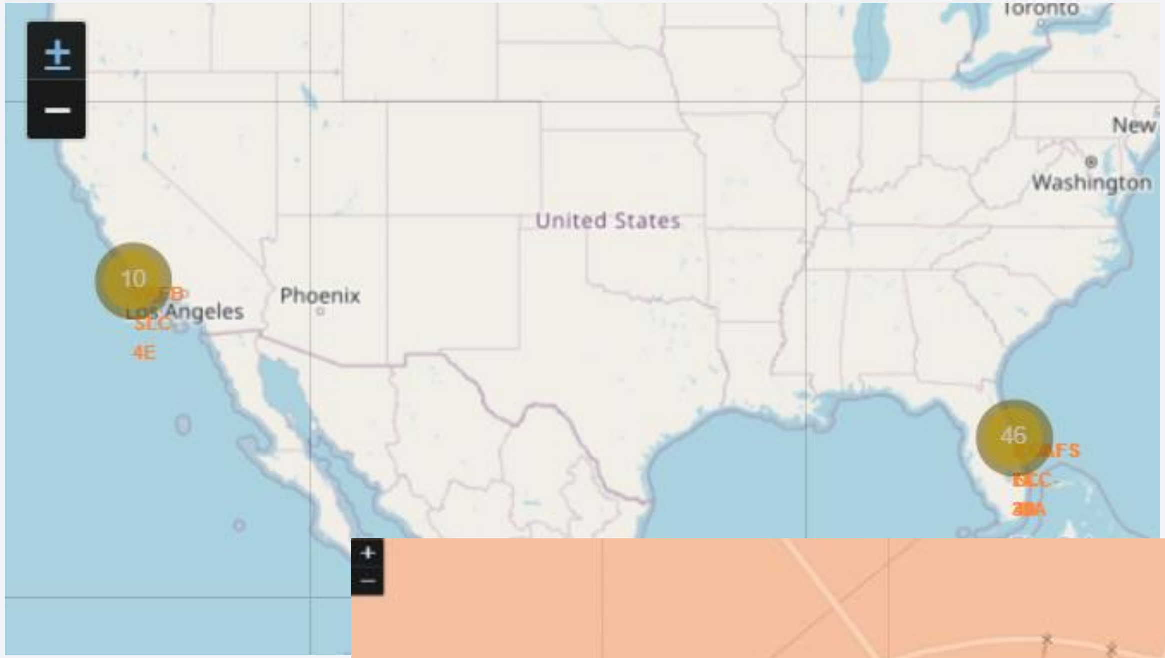
Global Overview of SpaceX Launch Sites

- **Key Findings:**
- **Coastal Placement:** Sites are located on the ocean's edge to ensure that launch trajectories are primarily over water, minimizing safety risks to populated areas in the event of debris fall or booster separation.
- **Latitude Choice:** The Florida sites are located at lower latitudes (closer to the equator) compared to the California site, allowing rockets to take better advantage of the Earth's rotational speed for eastward orbital launches.
- **East Coast Dominance:** The majority of launch activity (over 80% based on the clusters) is concentrated in the Florida region.



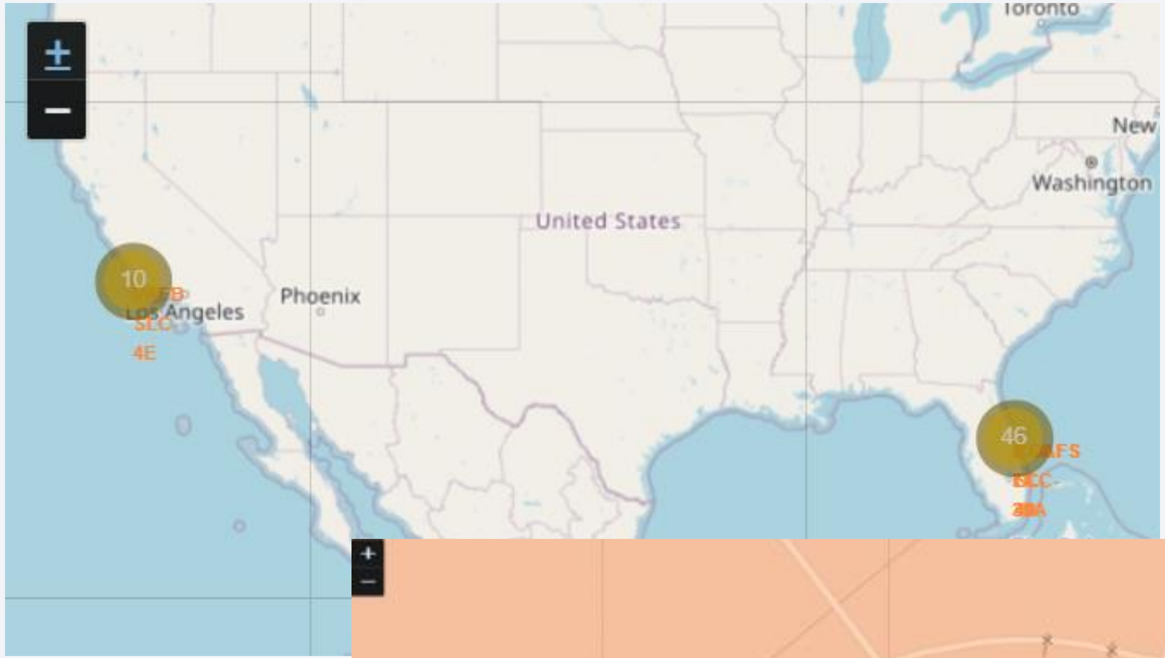
Launch Success and Reliability Analysis

- **Important Elements**
- **Marker Clustering:** The top map uses cluster markers (labeled **13** and **33**) to group high-density launch data in the Florida region, maintaining map readability while indicating total volume.
- **Success/Failure Markers:** Individual markers are color-coded: **Green** represents a successful launch outcome, while **Red** indicates a failure.
- **Outcome Spiral:** The bottom zoomed-in view uses a spiral layout to display individual outcomes at a single launch pad, preventing overlapping markers from obscuring each other.
- **Site Proximity Circles:** Orange circular buffers around the launch pads highlight the designated "Launch Zone" for each facility.



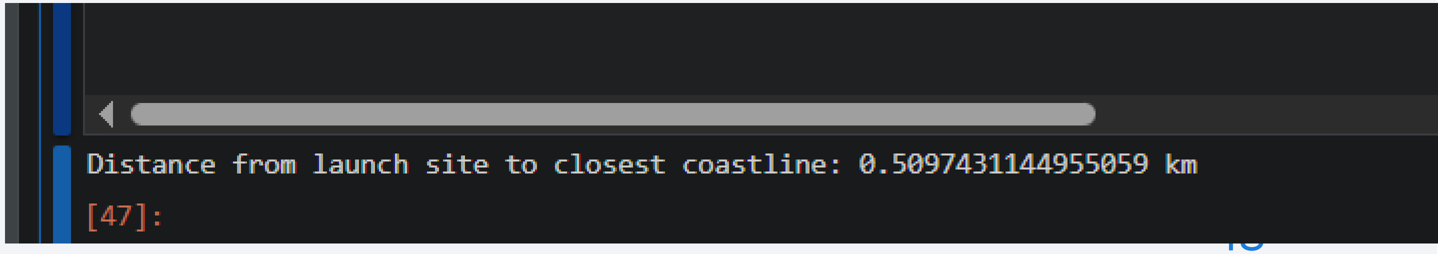
Launch Success and Reliability Analysis

- **Key Findings**
- **Regional Activity Concentration:** Launch activity is heavily concentrated on the East Coast near Cape Canaveral and Merritt Island, as shown by the clusters of 13 and 33 launches in the Florida vicinity.
- **Reliability Variance:** Specific launch pads demonstrate visible differences in reliability; for example, the spiral view shows a high proportion of red markers (failures) interspersed with green (successes), reflecting the iterative and experimental nature of historical SpaceX development.
- **Launch Pad Dedication:** The proximity circles confirm that launch pads are spatially isolated from one another to ensure safety and operational independence within the broader spaceport infrastructure.



Proximity and Infrastructure Logistics

- **Important Elements**
- **Distance Polylines:** Clear, straight lines (Folium PolyLine) connecting the launch site to the nearest coastline, highway, railway, and city.
- **Distance Annotations:** Visual markers showing the distance calculated in Kilometers (KM) using the **Haversine formula**.
- **Infrastructure Interaction:** The map display shows how the launch pad is physically situated relative to transportation hubs.
- **Coordinate Precision:** Accurate labeling of coordinates using the MousePosition tool to identify specific geographical features.



Proximity and Infrastructure Logistics

- **Coastal Safety Strategy:** Launch sites are positioned within **1 km of the coastline**. This ensures that the flight trajectory is immediately over the ocean, allowing for safe booster separation and protecting populated areas from debris in case of a launch failure.
- **Logistical Integration:** Sites are strategically built adjacent to **highways and railways**. This is essential for the logistics of moving massive, heavy rocket stages from manufacturing facilities directly to the launch pad with minimal transport complexity.

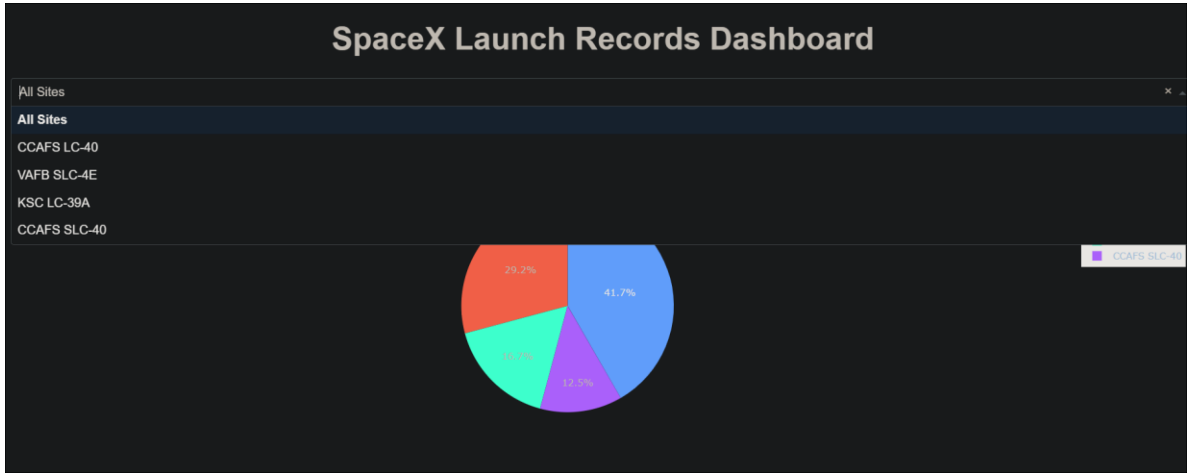

```
Distance from launch site to closest coastline: 0.5097431144955059 km
[47]:
```

Section 4

Build a Dashboard with Plotly Dash

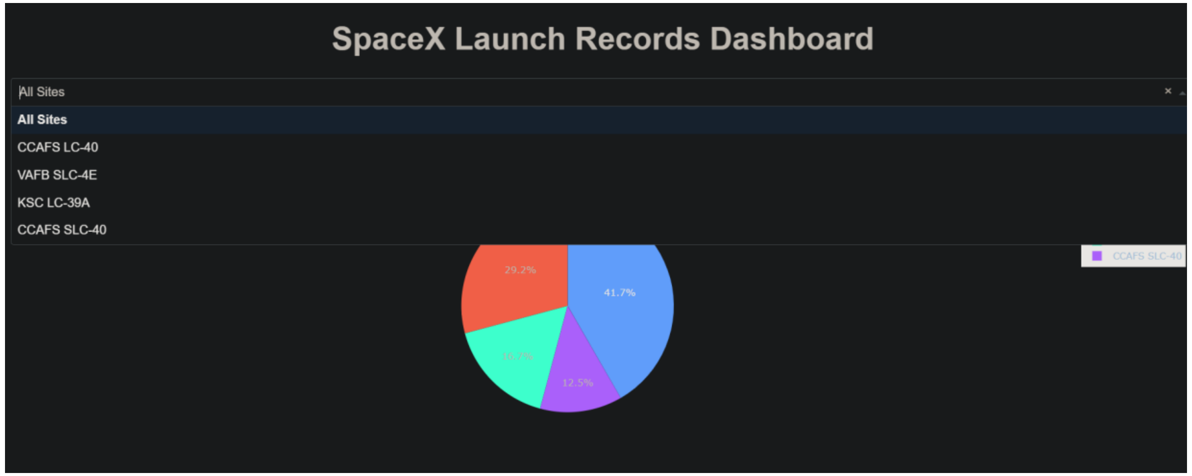
Interactive Visual Analytics Dashboard

- **Important Elements**
- **Site Selection Dropdown:** An interactive filter that allows users to toggle between a global view ("ALL Sites") or specific sites like **KSC LC-39A** and **CCAFS LC-40**.
- **Payload Range Slider:** A dynamic input used to filter launch data based on payload mass (0 kg to 10,000 kg), enabling targeted success rate analysis for different mission sizes.
- **Real-time Callbacks:** The dashboard automatically updates visual components (Pie and Scatter charts) whenever a user adjusts the dropdown or slider settings.



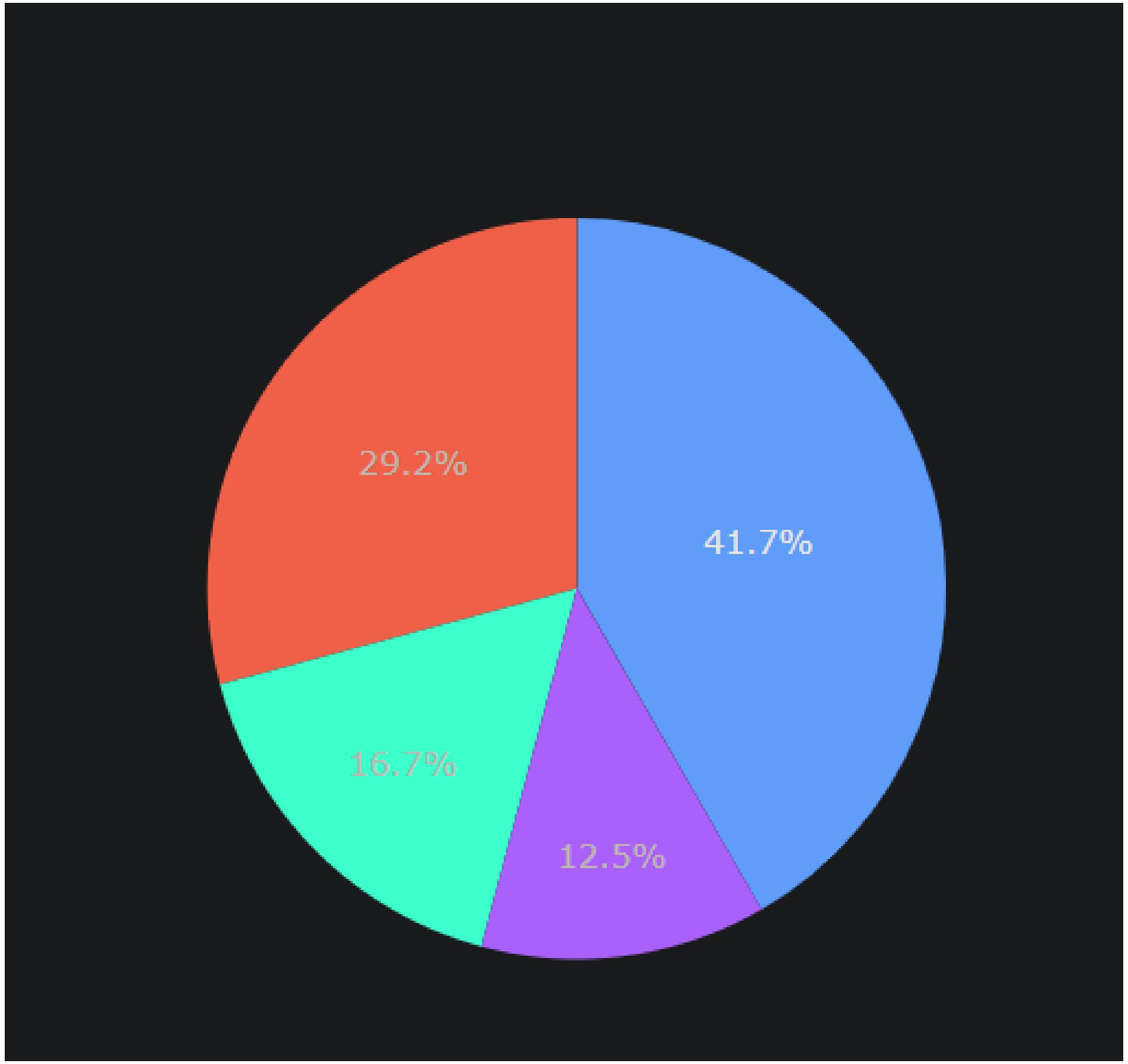
Interactive Visual Analytics Dashboard

- **Key Findings**
- **Customized Site Reporting:** The dashboard transforms from a general overview of the entire SpaceX program into a detailed pad-specific report with a single click, allowing for localized performance audits.
- **Dynamic Data Exploration:** By using the payload slider, analysts can instantly isolate specific mission profiles, such as identifying the success patterns for "Heavy" vs. "Light" payloads across different years.



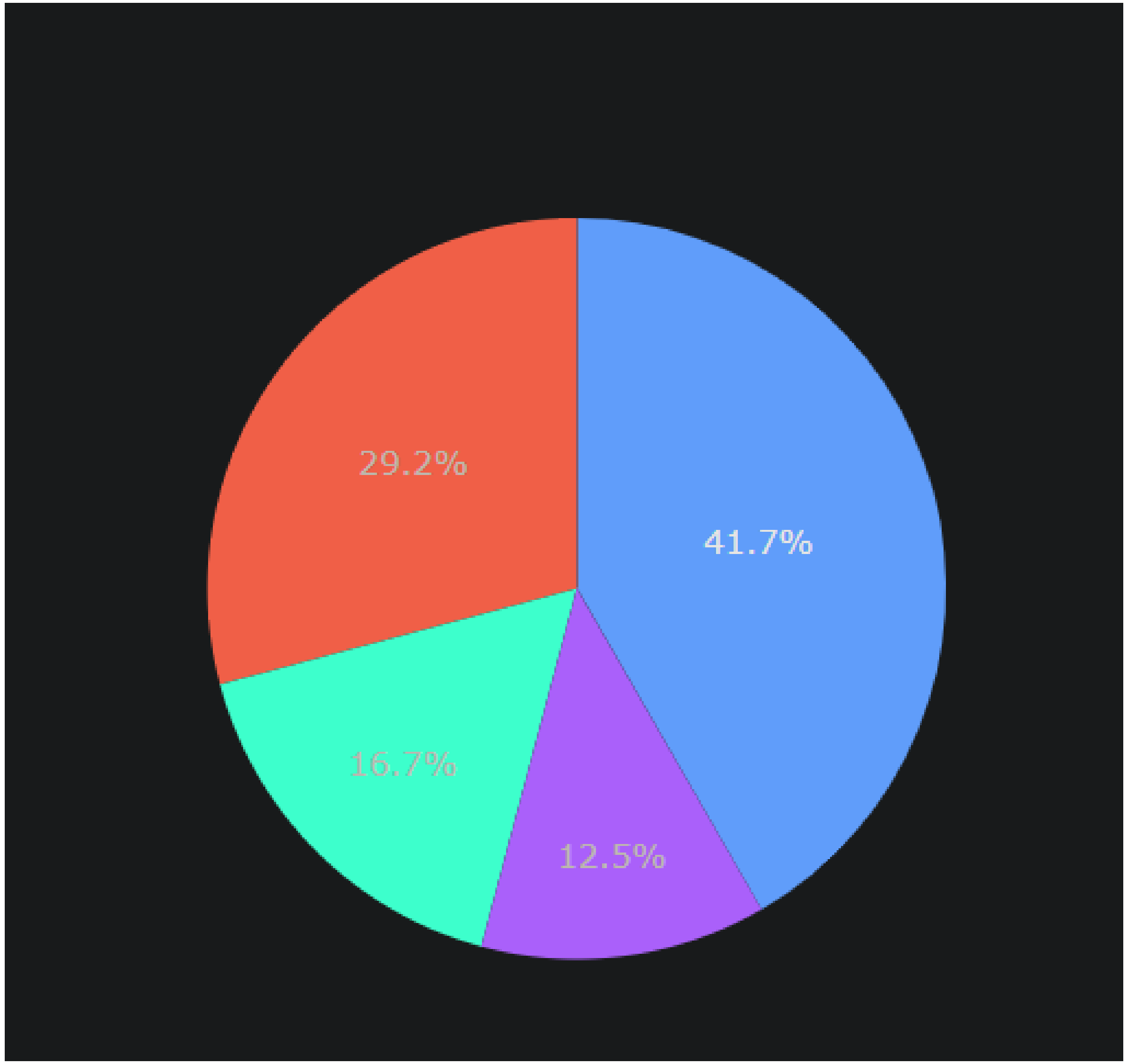
Total Successful SpaceX Launches by Site

- **Important Elements**
- **Interactive Site Dropdown**: A filter located at the top of the dashboard that allows the user to switch between "All Sites" or specific launch pads to update the pie chart in real-time.
- **Site Success Distribution**: A color-coded pie chart representing the total count of successful launches across different geographical locations.
- **Site Legend**: A key identifying the four main launch sites: **KSC LC-39A** (Blue), **CCAFS LC-40** (Red), **VAFB SLC-4E** (Green), and **CCAFS SLC-40** (Purple).
- **Percentage Data Labels**: On-chart labels that quantify the exact share of success contributed by each site relative to the whole.



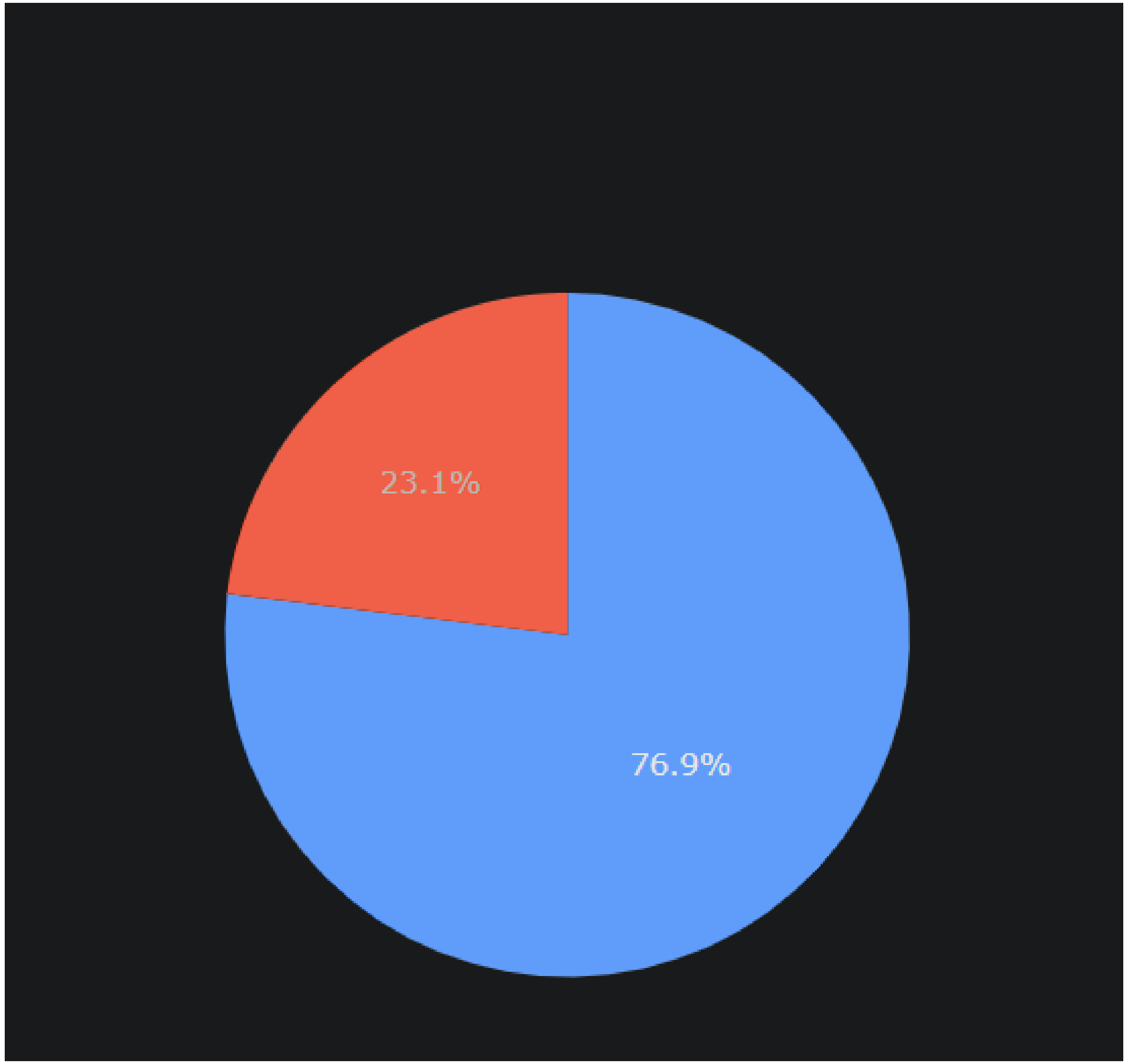
Interactive Visual Analytics Dashboard

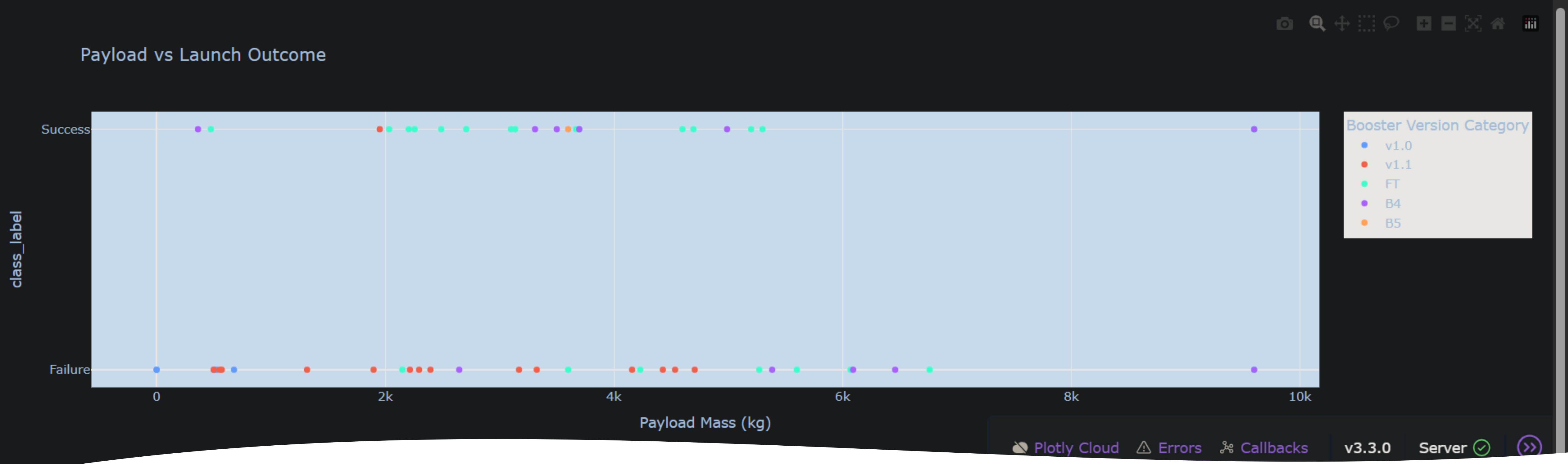
- **Key Findings**
- **KSC LC-39A Dominance**: Kennedy Space Center (KSC LC-39A) is the most successful launch site, accounting for **41.7%** of all successful SpaceX launches.
- **Regional Performance**: The combined Florida sites (KSC and CCAFS) account for the overwhelming majority of successful missions compared to the West Coast site (VAFB).
- **Comparative Reliability**: While **CCAFS LC-40** contributes significantly at **29.2%**, there is a notable performance gap between the high-volume Florida pads and the specialized **VAFB SLC-4E** (**16.7%**) or **CCAFS SLC-40** (**12.5%**) pads.



Pie chart for the launch site with highest launch success ratio

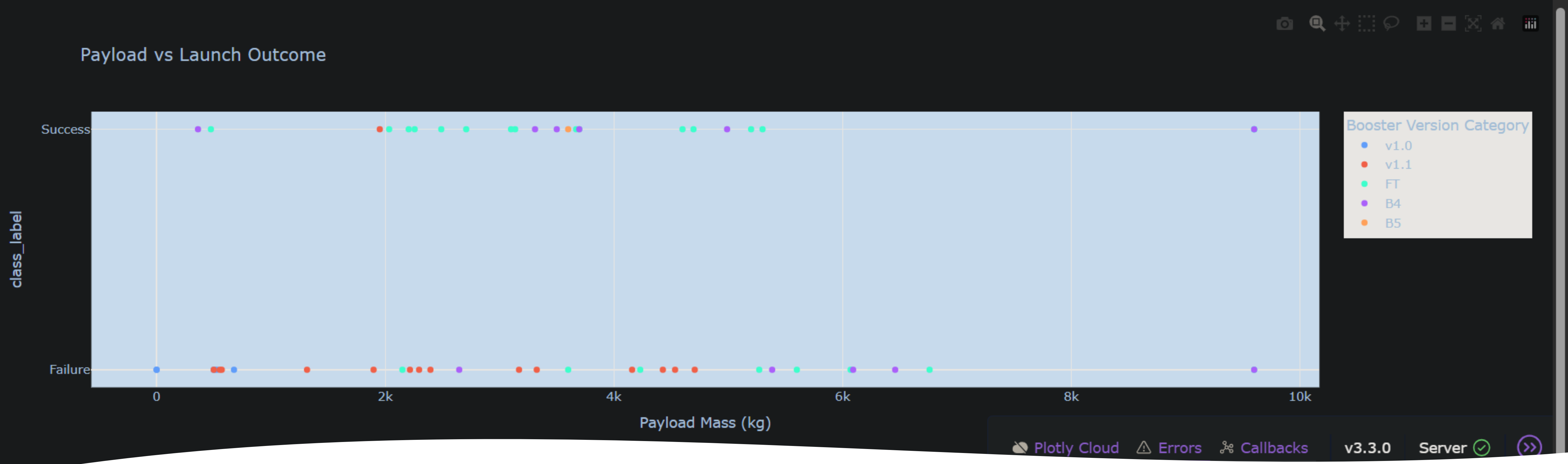
•**High Reliability at KSC LC-39A:** A large proportion of launches from KSC LC-39A are successful, highlighting the site's effectiveness.
•**Launch Outcome Breakdown:**
•**Successful Launches (Class 1):** 76.9%
•**Unsuccessful Launches (Class 0):** 23.1%
•**Insight:** The 76.9% success rate demonstrates that KSC LC-39A is a highly reliable and efficient launch site.





Payload Mass vs. Launch Outcome Analysis

- **Important Elements**
- **Interactive Payload Slider:** A range slider at the top allowing users to filter data from 0 kg to 10,000 kg to observe outcomes for different weight classes.
- **Booster Version Color Coding:** Data points are color-coded by booster hardware generation (v1.0, v1.1, FT, B4, B5) to track technical evolution.
- **Outcome Categorization (class_label):** The vertical axis clearly separates Successful launches (top) from Failures (bottom).
- **Mission Distribution:** Individual dots representing every mission's specific payload weight relative to its success or failure.



Payload Mass vs. Launch Outcome Analysis

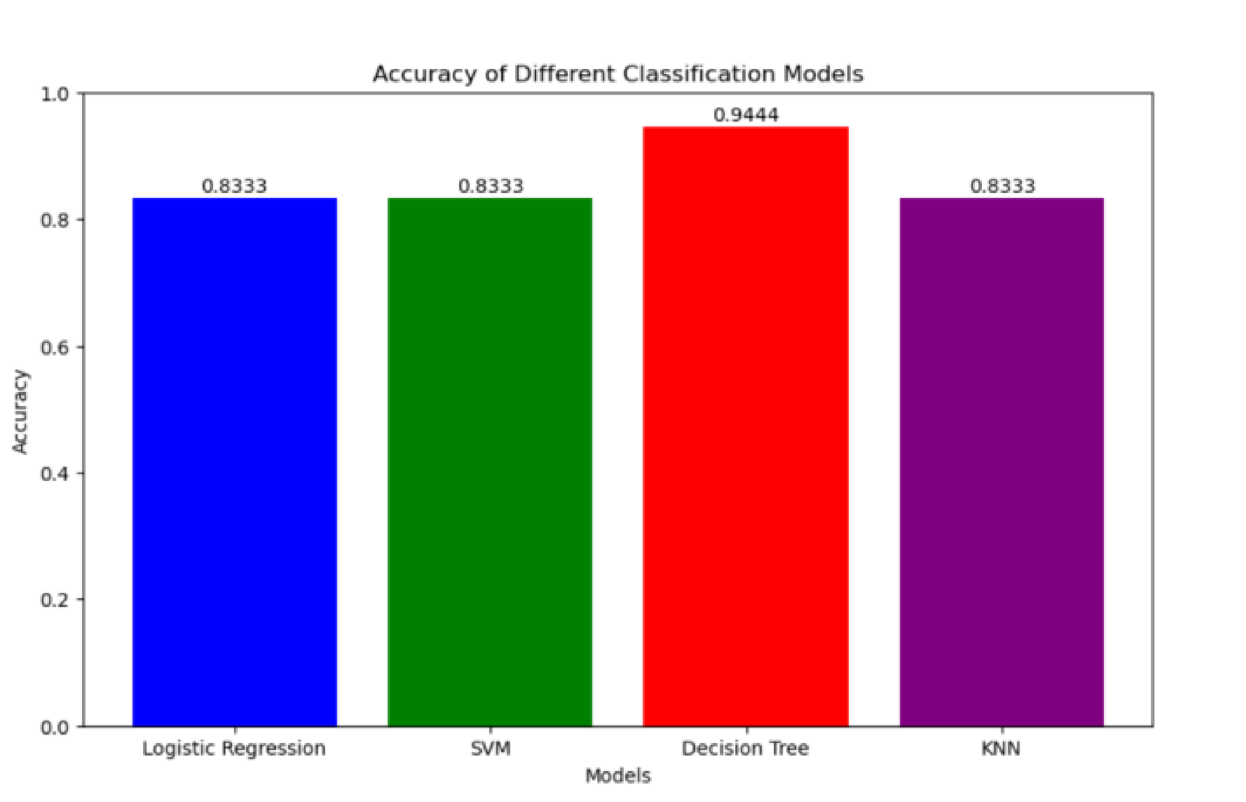
- **Key Findings**
- **Success Concentration:** There is a high density of successful missions (Class 1) for the **FT** and **B5** booster versions, particularly in the mid-to-high payload ranges.
- **Booster Reliability Evolution:** Newer booster versions, specifically **B5** (represented in orange), show a 100% success rate across all payload ranges shown in the scatter plot.
- **Failure Patterns:** Early booster versions like **v1.0** and **v1.1** (blue and red dots) are clustered primarily in the failure category for lower payload masses.
- **High Payload Performance:** Successes are consistently recorded even at the maximum payload capacity (near 10,000 kg), proving the heavy-lift capability of recent hardware iterations.

Section 5

Predictive Analysis (Classification)

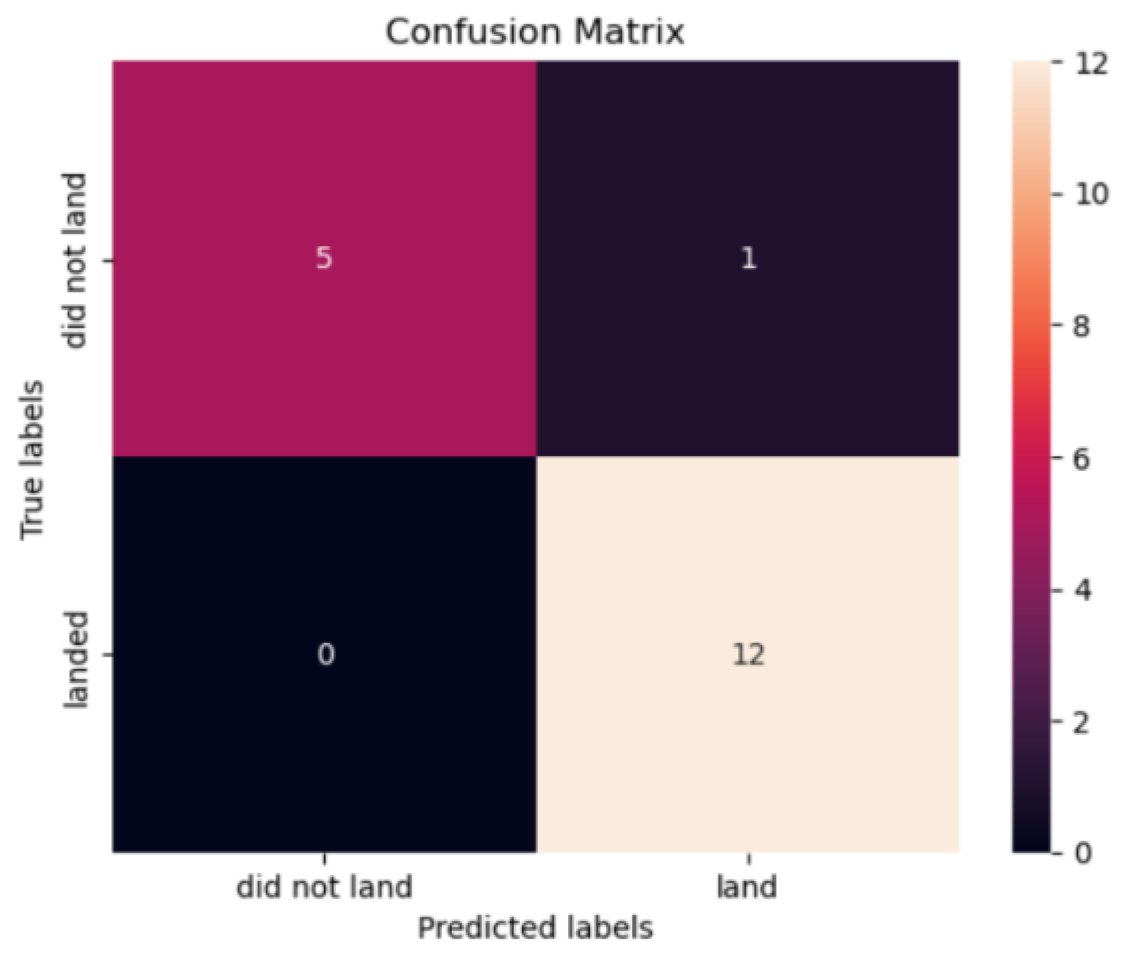
Classification Accuracy

- **Best Performing Model:** The Decision Tree classifier achieved the highest accuracy on the test dataset, with **0.9444**.
- **Comparison with Other Models:** Logistic Regression, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) each achieved an accuracy of **0.8333**.
- **Insight:** The Decision Tree model is the most suitable for this dataset, outperforming other classification algorithms in predicting first stage landing success.



Confusion Matrix

•**High Accuracy:** The model achieved **94.44% accuracy**, demonstrating strong predictive capability for Falcon 9 first stage landings, with a high number of true positives and true negatives.
•**No False Negatives:** All actual successful landings were correctly predicted, ensuring reliability—critical for safety and operational readiness in aerospace missions.
•**Manageable False Positives:** Only a single false positive occurred, which is less risky than false negatives. Over-preparation is more manageable than under-preparation in aerospace operations.
•**Balanced Performance:** The model slightly favors predicting successful landings, aligning with industry priorities where ensuring success is key for cost planning and mission execution.



Conclusions

- **Key Insights from the Analysis**
- **Launch Site Performance:** The **CCAFS LC-40** site has the highest success rate, contributing **43.7% of successful launches**, suggesting favorable conditions or efficient processes at this site.
- **Booster Reliability:** The **FT booster version** consistently performs well across a wide range of payloads, highlighting its reliability and making it a preferred choice for future missions.
- **Payload Impact:** No strong correlation was observed between higher payload masses and launch failures, indicating that **other factors**, such as **launch site conditions** and **booster type**, have a more significant influence on launch outcomes.

Thank you!