IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Toheeb Olufade
06/10/2023

Outline

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

Executive Summary

Overview

- This project analyzed the success rate of SpaceX rocket launches at various launch sites.
- The goal was to optimize the launch success rate and determine features and factors relevant to successful launch in order to bid against SpaceX.

Methodologies

- Data Collection and Cleaning
- Exploratory Data Analysis
- Feature Engineering
- Classification modelling
- Hyperparameter tuning

Results

- Best Performing Model : Decision Tree Algorithm with an accuracy score of 87.32% on the training data and 83.33% on the test data.
- Optimal parameters for this model: {'criterion': 'gini', 'max_depth': 18, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2, 'splitter': 'best'}

Introduction

Background

- Analysis of SpaceX rocket launches data.

Context

- In recent years, space exploration has seen unprecedented growth with private companies like SpaceX leading the way. Understanding the factors influencing launch success is crucial for optimizing future missions.

Problems Addressed

1. Determining key factors influencing successful rocket launches.
2. Leveraging data to enhance launch planning and decision-making.
3. Enhancing reliability for payloads of varying weights.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

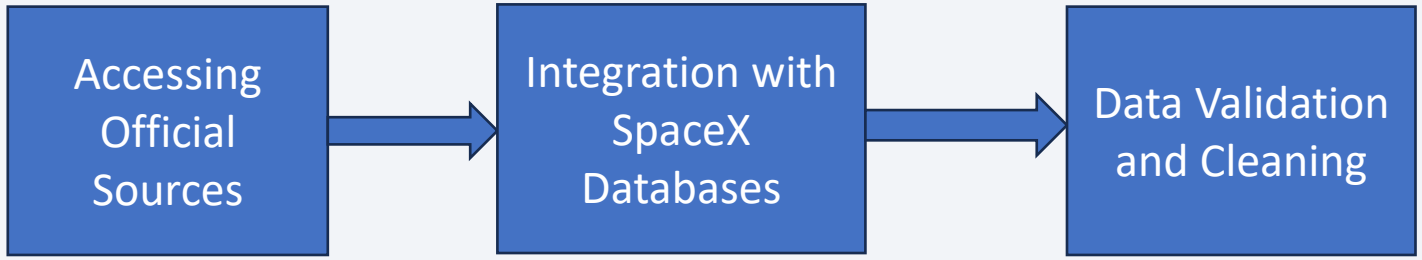
Data Sources

- Obtained from official SpaceX records using SpaceX APIs.
- Obtained from SpaceX Launches Wikipedia page.

Data Gathering Steps

- Obtained SpaceX launch records from official sources and Wikipedia.
- Integrated additional mission details from SpaceX datasets.
- Ensured accuracy and consistency through rigorous validation and cleaning processes.

Data Collection Flowchart

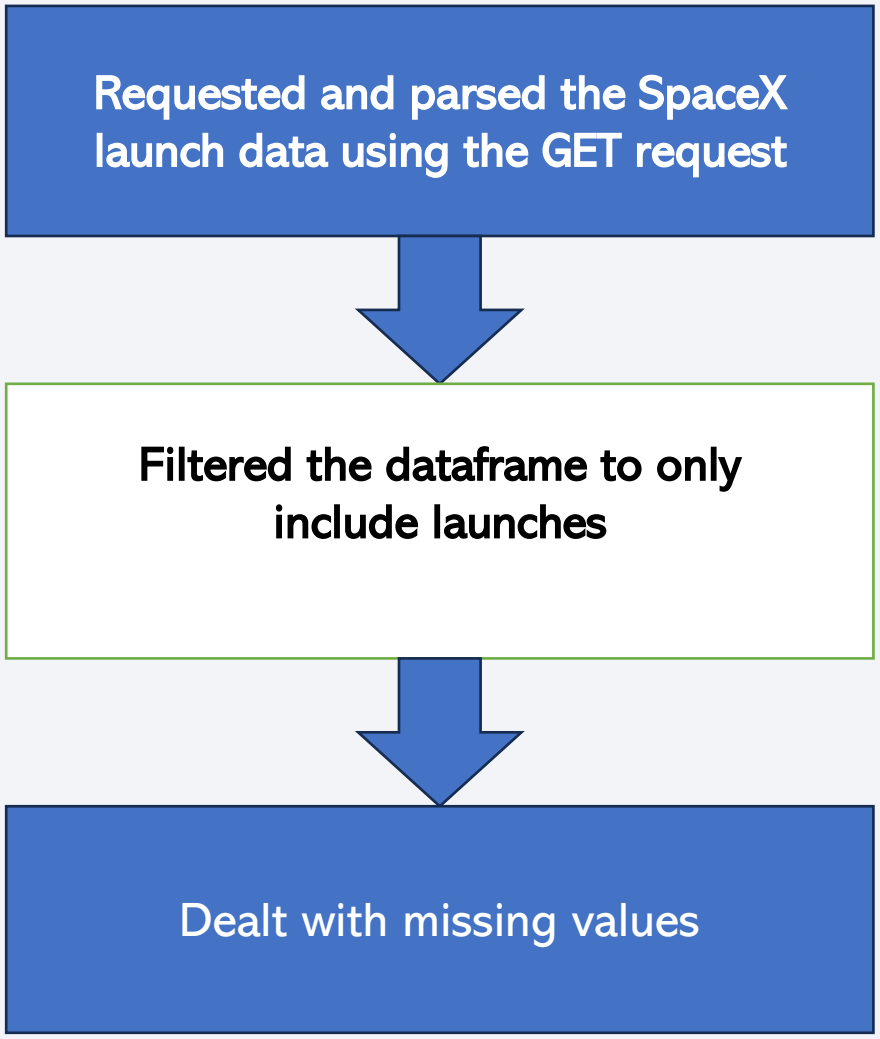


Data Collection – SpaceX API

- Executed RESTful queries to SpaceX API for real-time launch data.
- Converted raw API responses into structured datasets.
- Merged the SpaceX records for comprehensive insights.
- Scrutinized and rectified anomalies for high-quality, reliable data.

GitHub URL

- https://github.com/ToheebA/Rocket-Launch-Project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

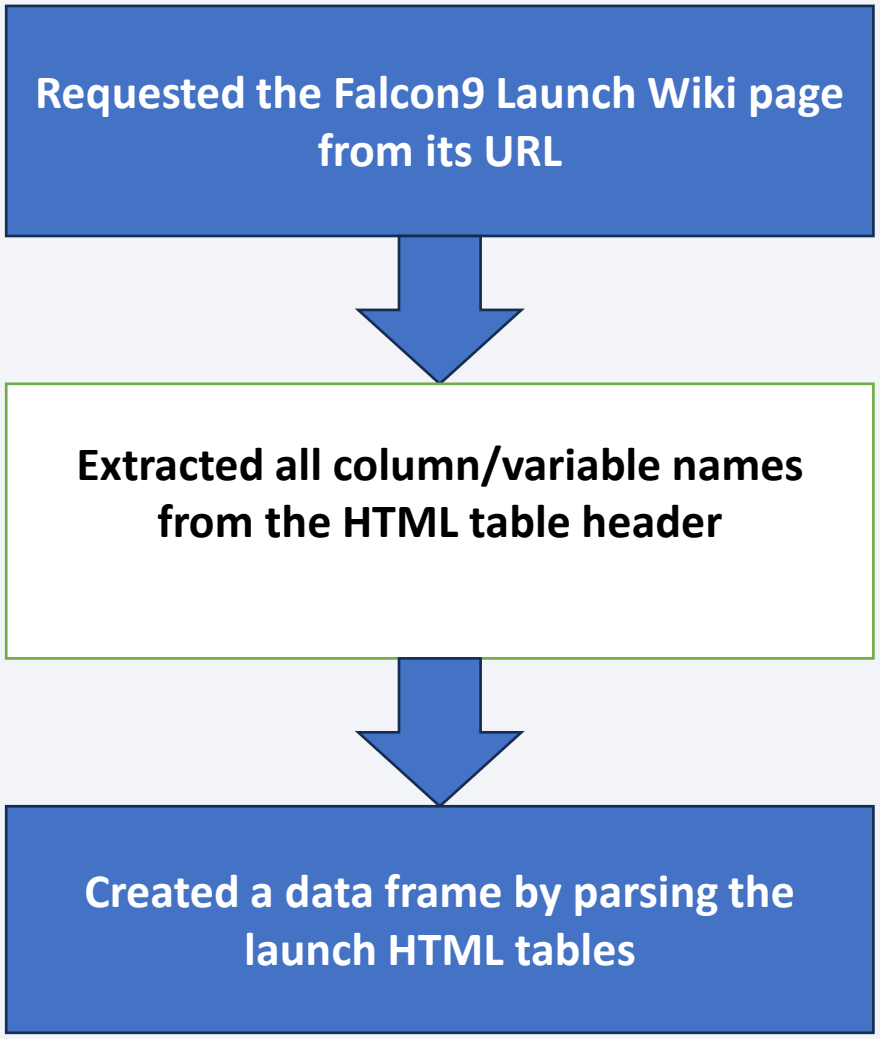


Data Collection - Scraping

- Utilized Python libraries like BeautifulSoup to parse HTML structure from Wikipedia page.
- Extracted launch information including date, mission, and outcomes.
- Processed raw HTML data into structured datasets.

GitHub URL

- https://github.com/ToheebA/Rocket-Launch-Project/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

Data Cleaning:
- Addressed missing values, duplicates, and outliers to ensure data quality.

Standardization:

- Uniformly formatted data for consistency in analysis

Normalization:

Categorical Encoding:

- Converted categorical variables into numerical format for modeling.

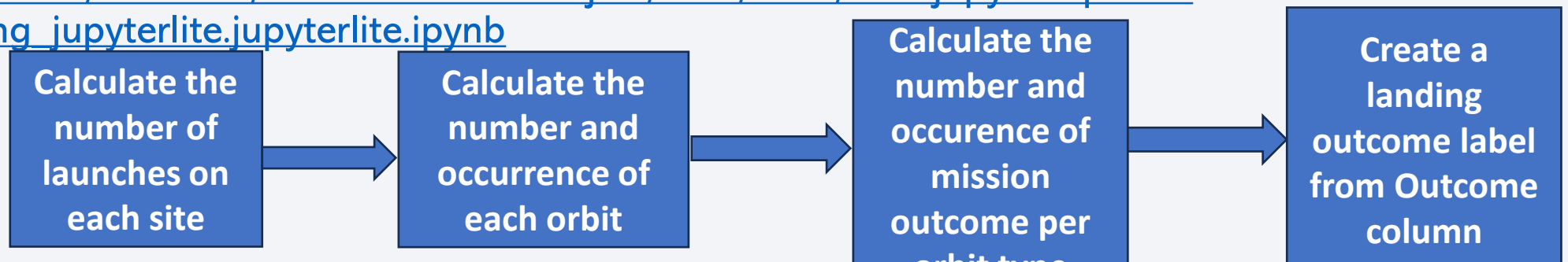
Feature Engineering:

- Created new features to enhance predictive capabilities.

GitHub URL

https://github.com/ToheebA/Rocket-Launch-Project/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

Data Wrangling Flowchart



EDA with Data Visualization

The following charts were created:

- Scatterplot of FlightNumber vs PayloadMass
- Scatterplot of FlightNumber vs LaunchSite
- Scatterplot of PayloadMass vs LaunchSite
- Barchart of Orbit types with their respective Success rate
- Scatterplot of FlightNumber vs Orbit
- Scatterplot of PayloadMass vs Orbit
- Line chart of Year vs Average Success Rate

GitHub URL

https://github.com/ToheebA/Rocket-Launch-Project/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

The following SQL queries were made:

- Query to display the names of the unique launch sites in the space mission.
- Query to display 5 records where launch sites begin with the string ‘CCA’
- Query to display the total payload mass carried by boosters launched by NASA(CRS).
- Query to display average payload mass carried by booster version F9 v1.1
- Query to list the date when the first successful landing outcome in ground pad was achieved
- Query to list the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

GitHub URL

https://github.com/ToheebA/Rocket-Launch-Project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

The following map objects were added:

- Markers: Represented launch sites and coastline points.
- Circles: Highlighted specific areas for visual emphasis.
- Lines: Illustrated paths or boundaries for better spatial understanding.

Reasons for adding map objects:

- Markers: Provided visual reference points for launch sites and coastline, enhancing map interactivity and clarity.
- Circles: Emphasized specific regions, such as launch sites, for important context.
- Lines: Outlined boundaries or paths to improve spatial comprehension, e.g., equator line

GitHub URL

https://github.com/ToheebA/Rocket-Launch-Project/blob/main/lab_jupyter_launch_site_location.jupyterlite%20(1).ipynb

Build a Dashboard with Plotly Dash

The following plots/graphs and interactions were added to the dashboard:

- Pie Chart: Displayed success rates for different launch sites.
- Scatter Chart: Showed the correlation between payload and launch success.
- Dropdown: Allowed users to select specific launch sites for detailed information.
- Range Slider: Enabled users to filter data based on payload range.

Reasons for adding elements:

- Pie Chart: Provided a quick overview of success rates, aiding in site comparison.
- Scatter Chart: Offered insights into the relationship between payload and success, aiding in decision-making.
- Dropdown: Facilitated site-specific exploration for deeper understanding.
- Range Slider: Empowered users to customize data view based on payload criteria.

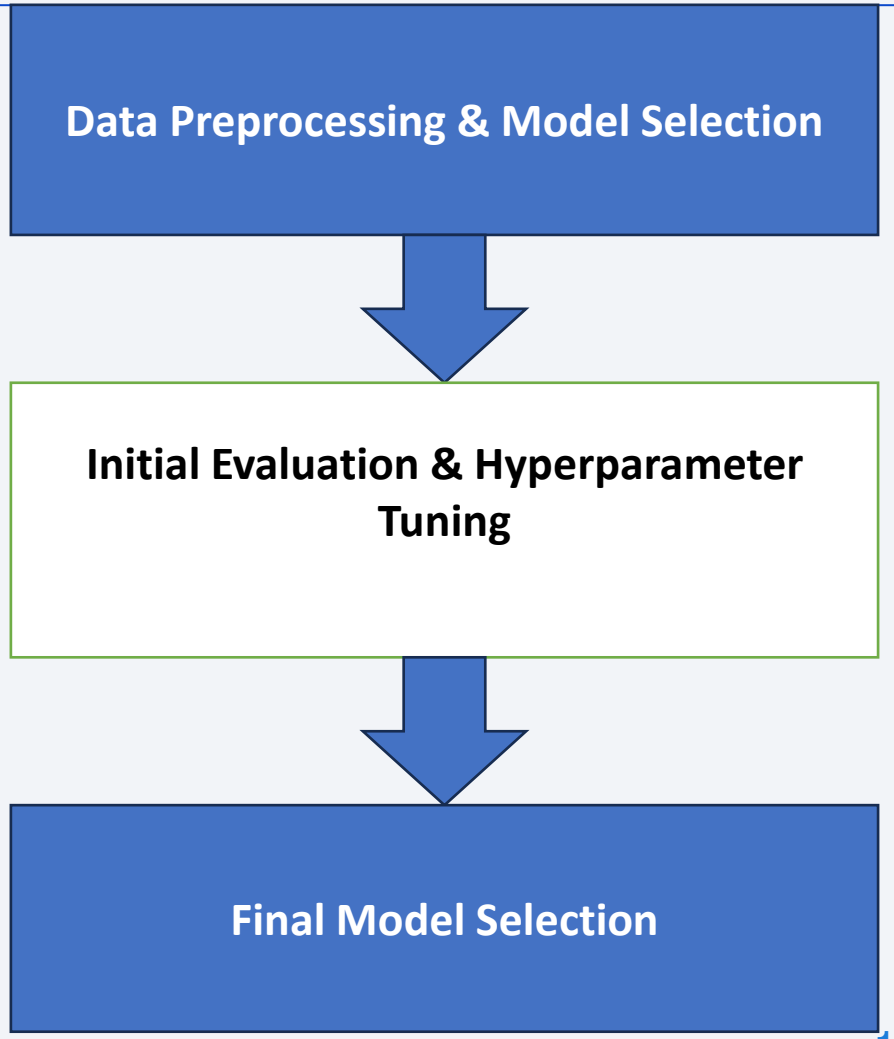
GitHub URL

https://github.com/ToheebA/Rocket-Launch-Project/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Model Development Process

1. Data Preprocessing:
   - Handled missing values and categorical variables.
   - Applied feature scaling.
2. Model Selection:
   - Explored Logistic Regression, SVM, Decision Trees, and K-Nearest Neighbor
3. Initial Evaluation:
   - Assessed models with default parameters using cross-validation
4. Hyperparameter Tuning:
   - Conducted GridSearchCV to find optimal parameters.
5. Final Model Selection:
   - Decision Trees exhibited highest performance



GitHub URL

https://github.com/ToheebA/Rocket-Launch-Project/blob/main/labs_module_4_SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

Exploratory Data Analysis Results:

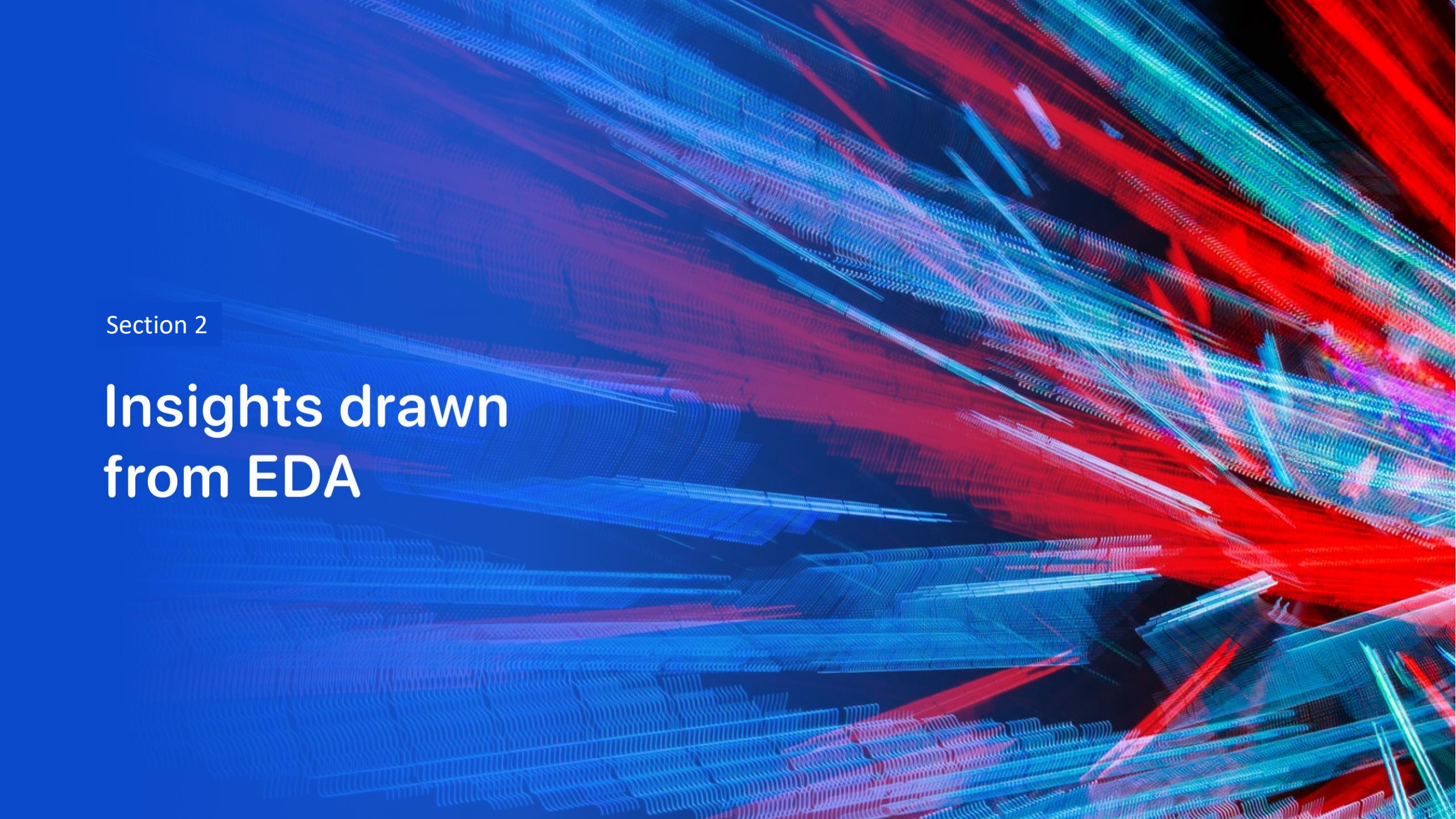
- Summary statistics.
- Distribution of launch outcomes.
- Correlation between key variables.

Interactive Analysis:

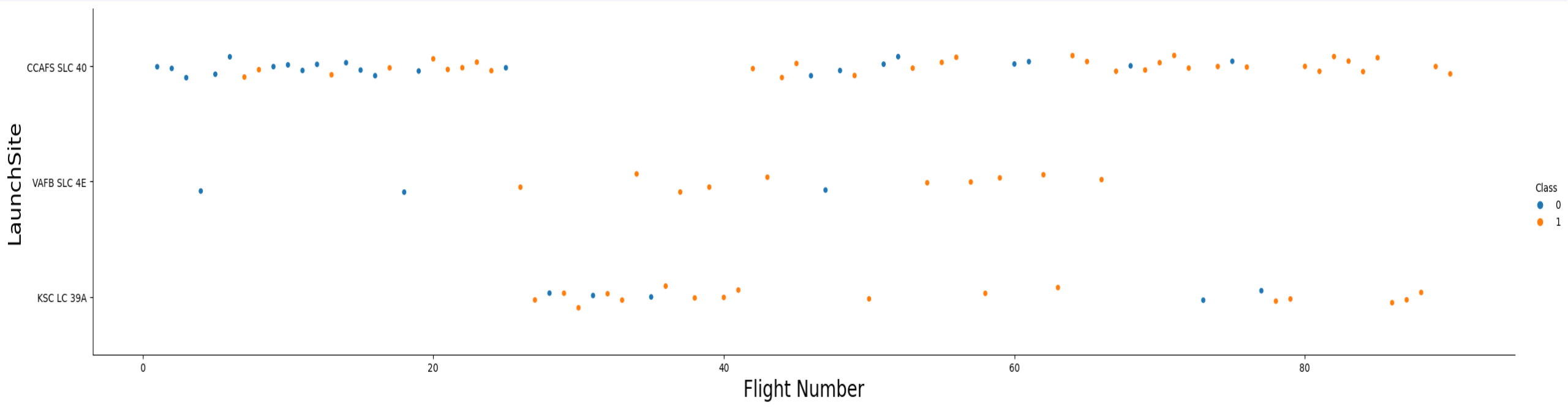
- User-friendly interface.
- Dynamic filters for site selection.
- Real-time data updates.

Predictive Analysis Results

- Classification model performance.
- Success rate predictions.
- Model evaluation metrics.

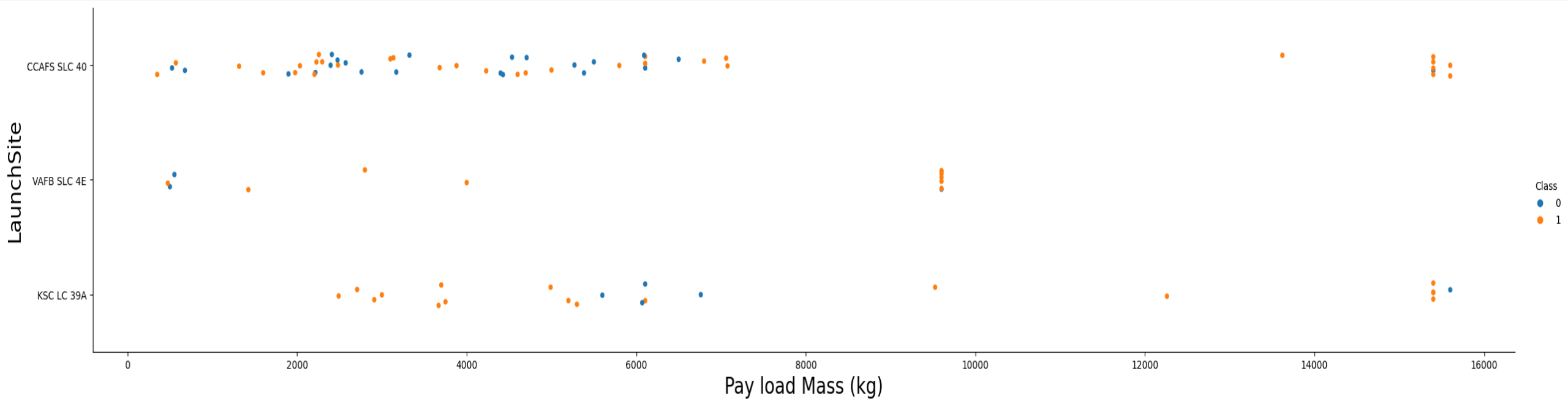

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



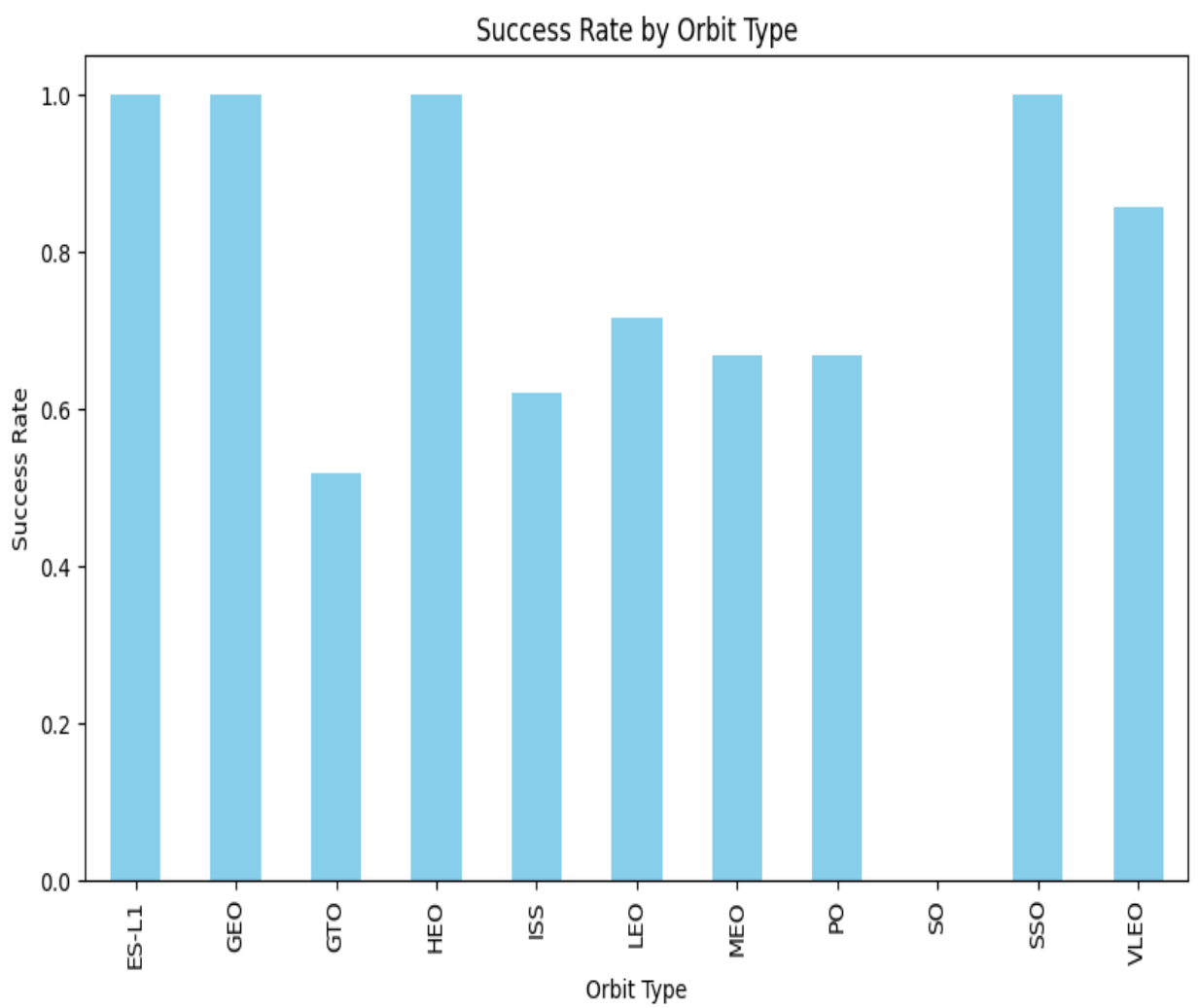
- It is observed from the plot that site KSC LC 39A did not have an any falcon 9 launch until around flight number 23.
- Site CCAFS SLC 40 has the highest number of falcon 9 launches and highest number of unsuccessful launches.
- Site VAFB SLC 4E has just three unsuccessful falcon 9 launches.

Payload vs. Launch Site



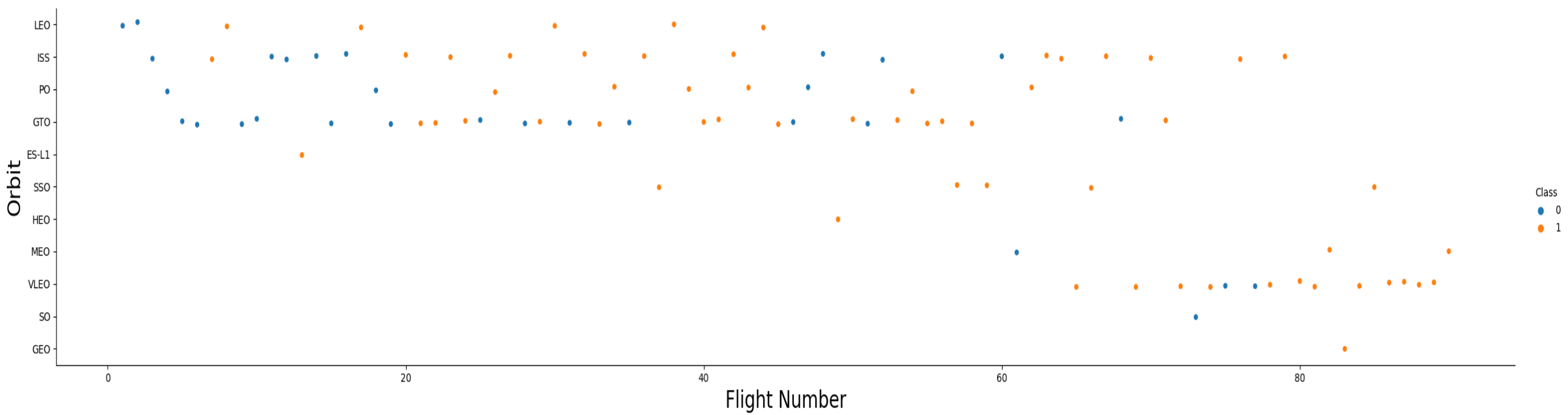
- It is observed from the plot that site CCAFS SLC 40 did not have falcon 9 launches with payload mass between 7000kg and 13000kg.
- Site VAFB SLC 4E did not have falcon 9 launches with payload mass greater than 10000kg.
- Site KSC LC 39A did not have falcon 9 launches with payload mass lesser than 2000kg.

Success Rate vs. Orbit Type



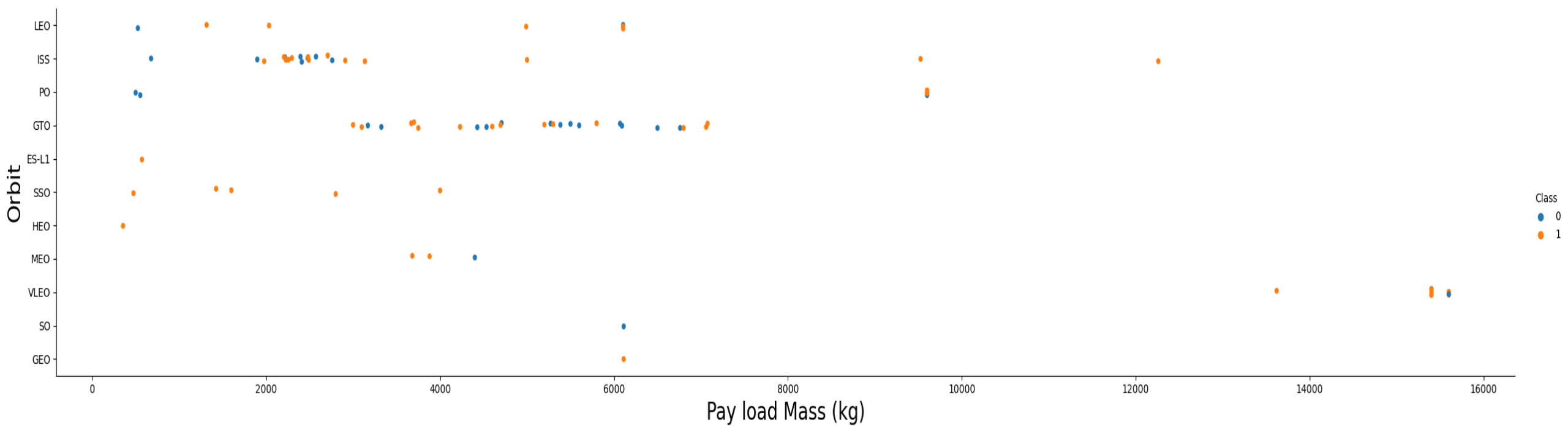
- It is observed that ES-L1, GEO, HEO and SSO orbits have the highest success rate of 100%.
- GTO orbit has the lowest success rate of 50%.

Flight Number vs. Orbit Type



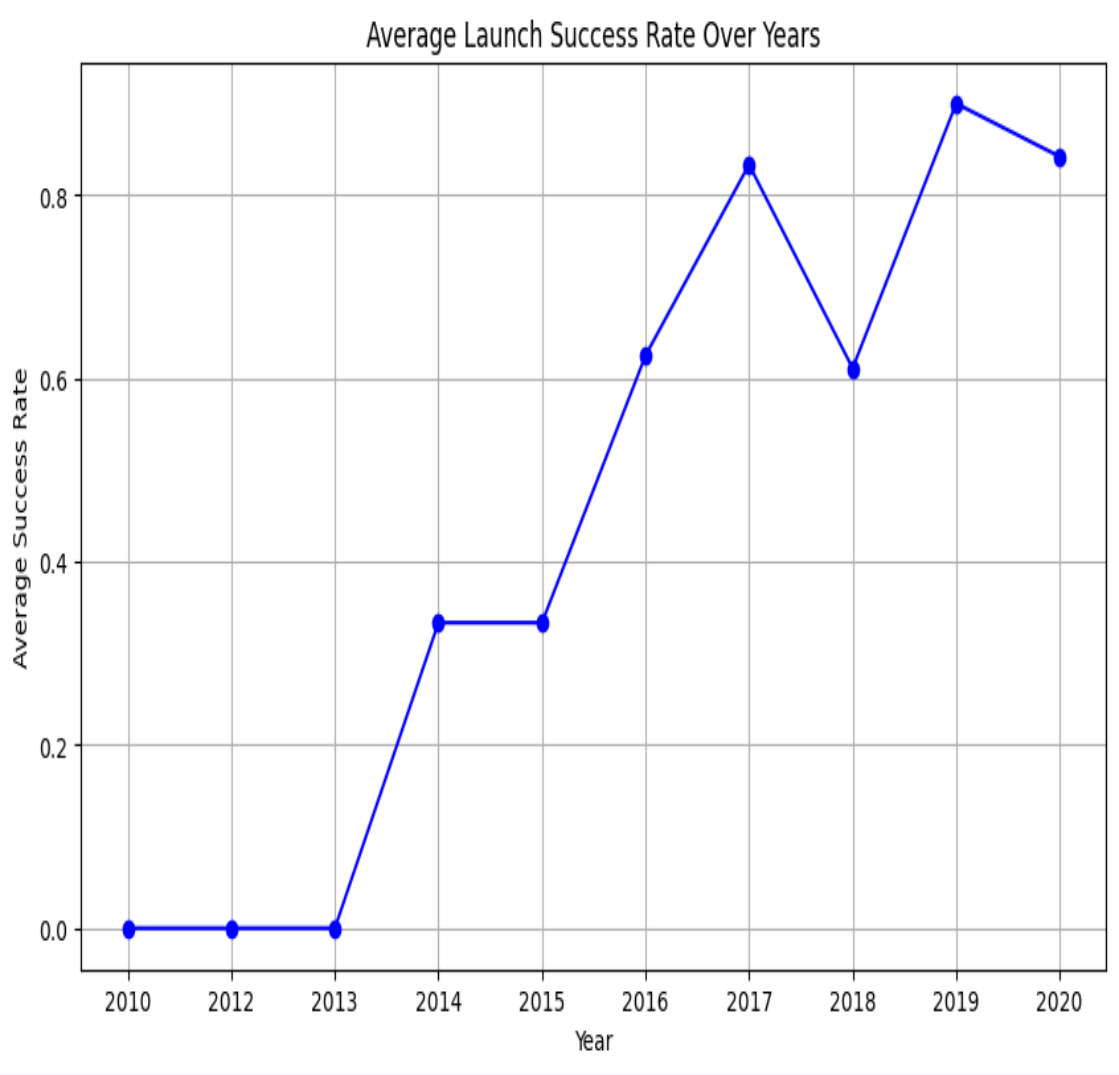
- It is observed that the success of LEO orbit appears related to the number of flights while on the other hand, there seems to be no relationship between flight number when in GTO orbit.
- It is observed that GEO, SO, VLEO, and MEO orbits did not have falcon 9 launches until around flight number 60.

Payload vs. Orbit Type



- The scatter plot shows that heavy payload increases the chances of successful landing or positive landing rate for Polar, LEO and ISS.
- However it is undistinguishable for GTO as both positive landing rate and negative landing(unsuccessful mission) are both here.

Launch Success Yearly Trend



- The average success rate kept increasing from 2013 until 2020 with a single drop in 2018.

All Launch Site Names

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

- There are four unique launch sites:
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40

Launch Site Names Begin with 'CCA'

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

SUM("PAYLOAD_MASS__KG_")
45596

- The total payload mass carried by boosters launched by NASA (CRS) is 45596kg.

Average Payload Mass by F9 v1.1

AVG("PAYLOAD_MASS__KG_")
2928.4

- The average payload mass carried by booster version F9 v1.1 is 2928.4kg.

First Successful Ground Landing Date

MIN(Date)
2015-12-22

- The date for the first successful landing outcome in ground pad is 2015-12-22.

Successful Drone Ship Landing with Payload between 4000 and 6000

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

- The figure above shows the names of boosters that have success in drone ship and have payload mass greater than 4000 but less than 6000.

Total Number of Successful and Failure Mission Outcomes

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

- The table above shows the total number of successful and failure mission outcomes.

Boosters Carried Maximum Payload

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

- The table shows 12 distinct booster versions that carried the maximum payload mass.

2015 Launch Records

Month Names	Date	Landing_Outcome	Booster_Version	Launch_Site
October	2015-10-01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	2015-04-14	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- The query shows two records for the months in 2015.

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

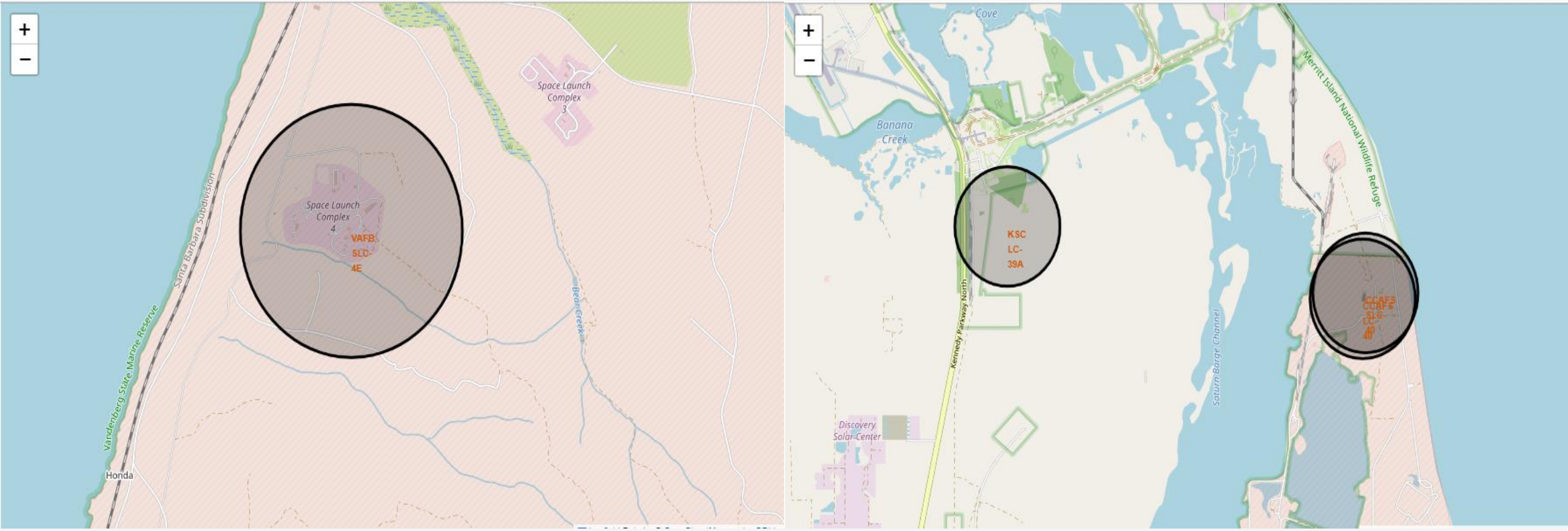
Landing_Outcome	Count
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

- The table above shows no attempt landing outcome has the highest count of 10 while failure (parachute) landing outcome has the lowest count of 1.

Section 3

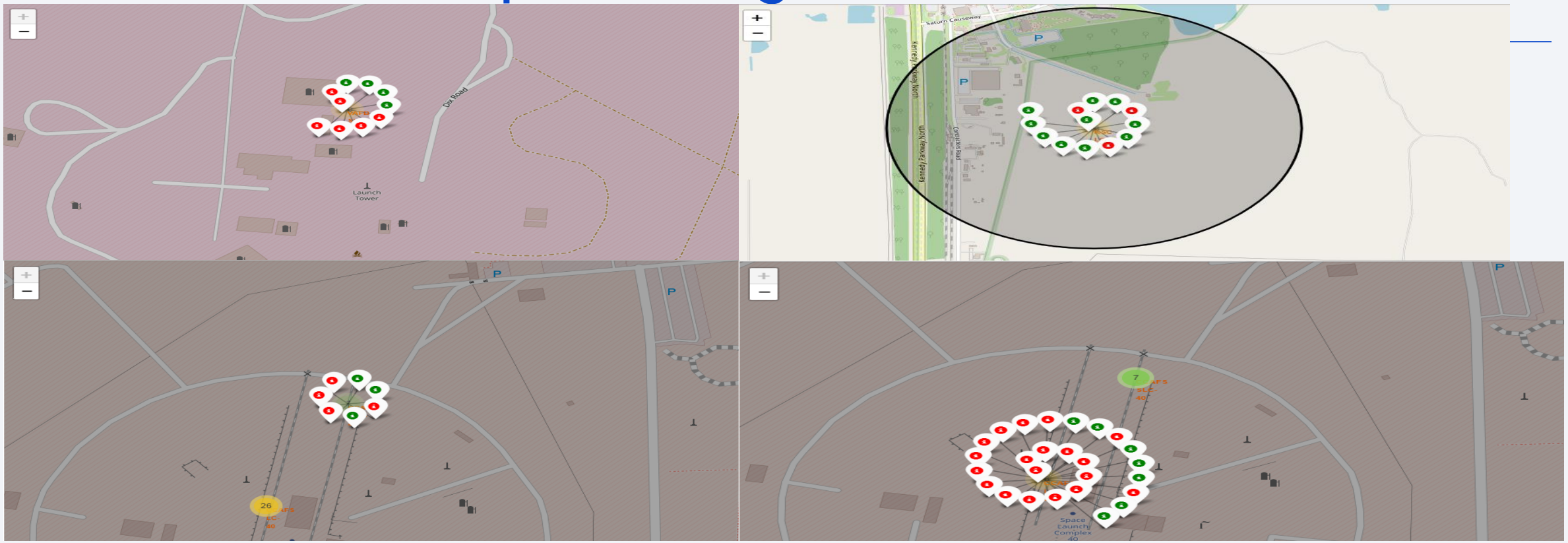
Launch Sites Proximities Analysis

Folium Map Showing the four Launch Site Locations



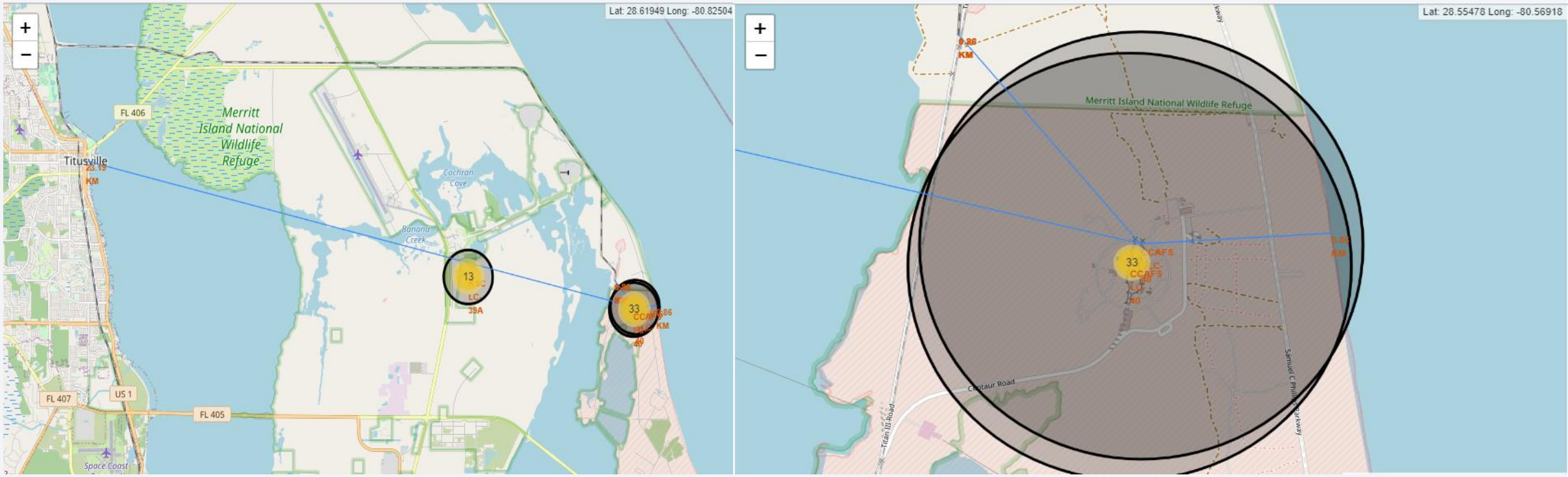
The screenshot displays four launch sites (CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, and VAFB SLC-4E) on a Folium map, showing their precise geographical coordinates and potential proximity to significant landmarks or bodies of water.

Folium Map Showing the Launch Outcomes

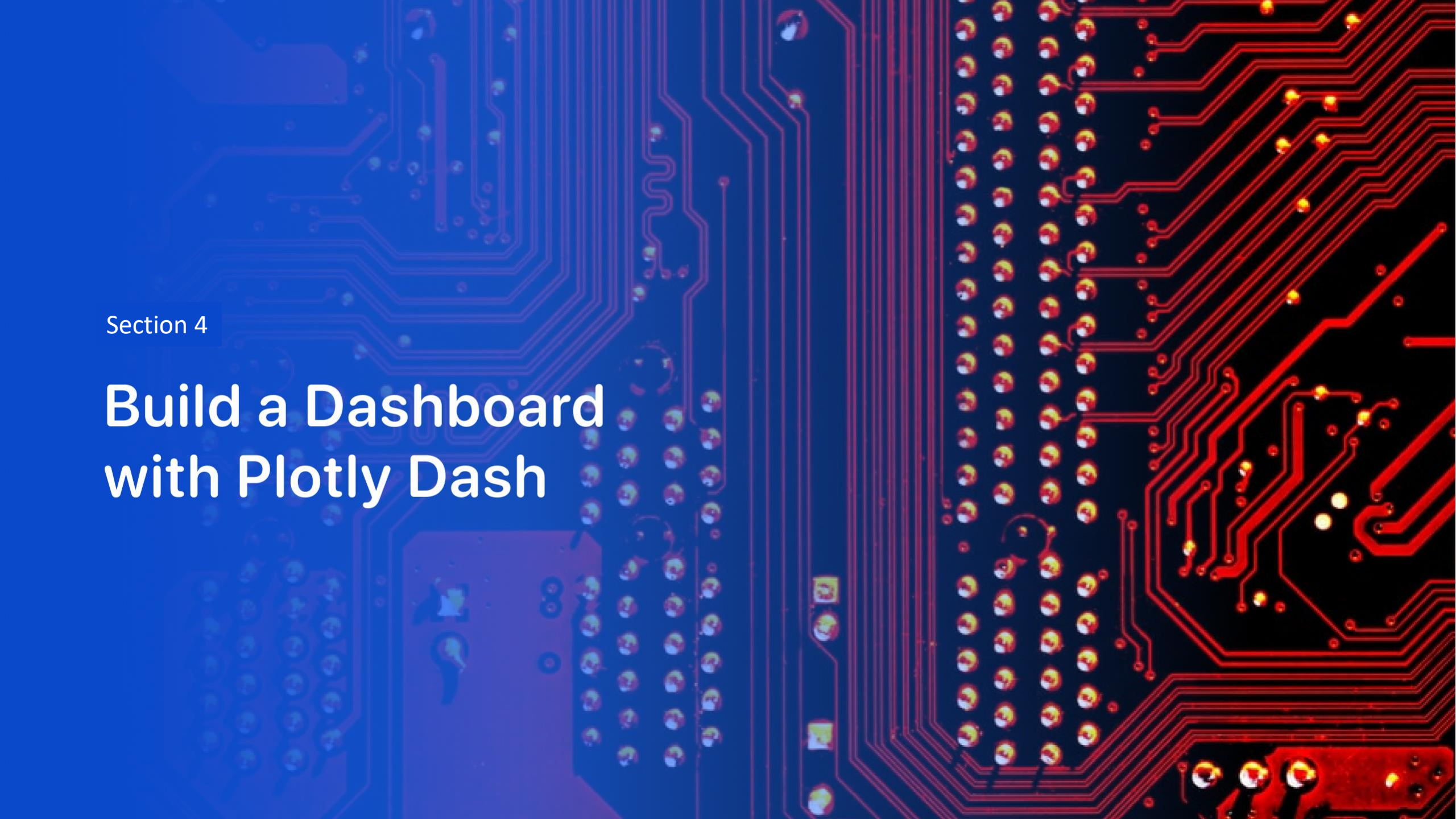


The color-coded markers on the Folium map distinguish between successful (green) and failed (red) launch outcomes, offering a visual overview of success rates at different launch sites.

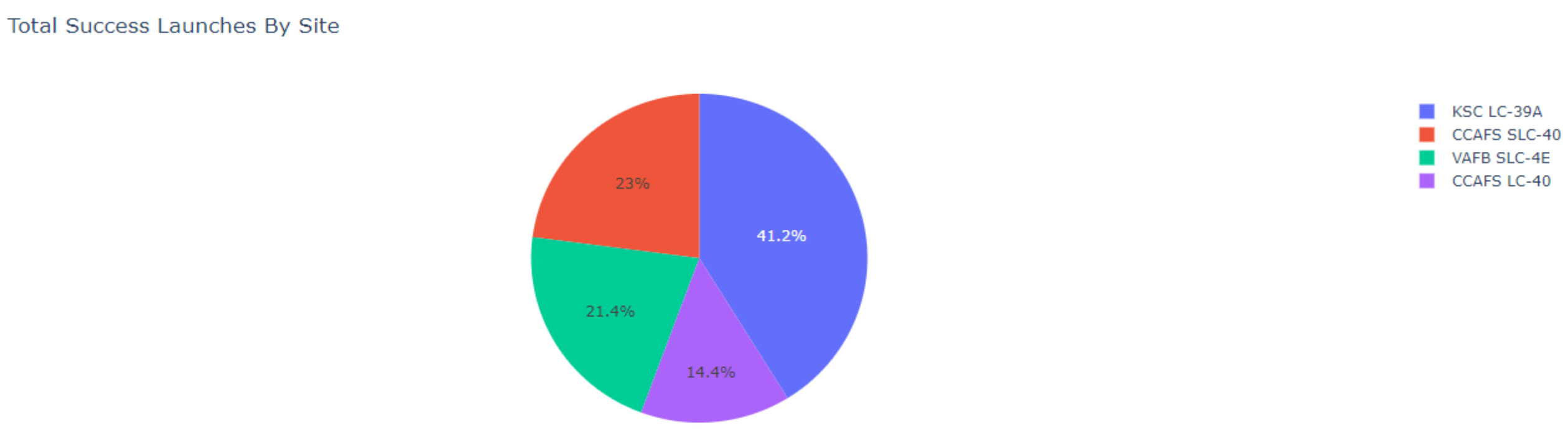
Folium Map Showing the Proximities of a Launch Site to Certain Landmarks



The Folium map offers a visual overview of a launch site's distance to significant landmarks. The labelled markers indicate the proximity to the coastline, nearest city, and nearest railway, providing valuable insights for launch logistics and safety evaluations. It is also observed that the launch sites are situated at considerable distances from cities and towns for safety reasons.

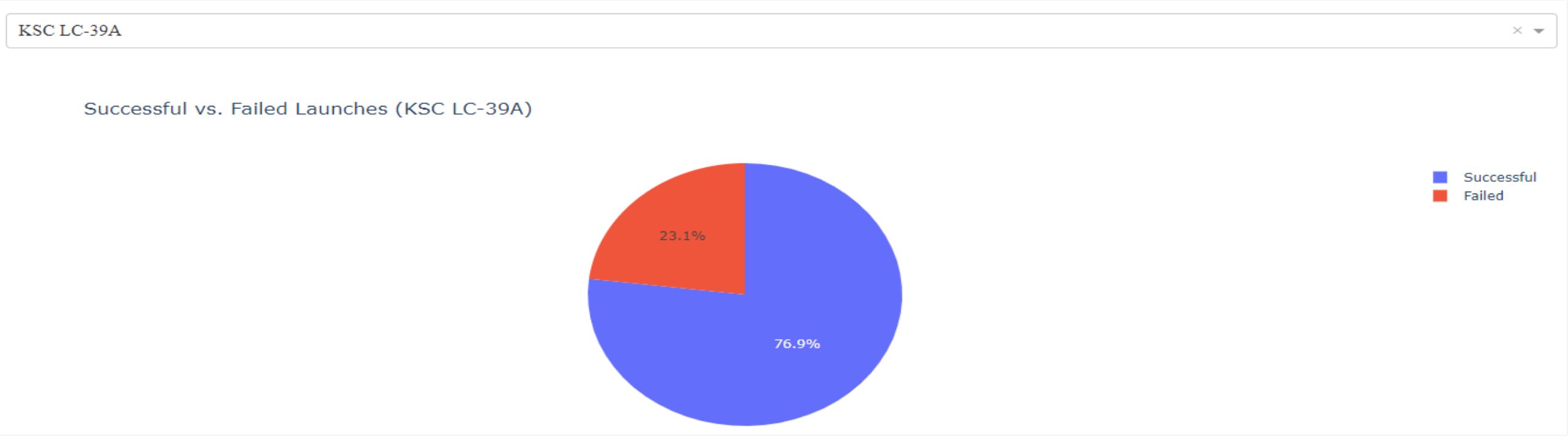

Section 4
Build a Dashboard with Plotly Dash

Dashboard Screenshot showing Total Success Launches by Site



The pie chart displayed in the Dash app provides a visual representation of launch success counts for all sites. It effectively conveys the distribution of successful and failed launches across different launch sites. This information is essential for evaluating the overall performance of each site and making informed decisions about future launches.

Dashboard Screenshot showing Success Rate for the most Successful Site



The pie chart displayed in the Dash app highlights the launch success ratio for the specific site with the highest success rate. It offers a clear visual representation of the proportion of successful and failed launches at this site.

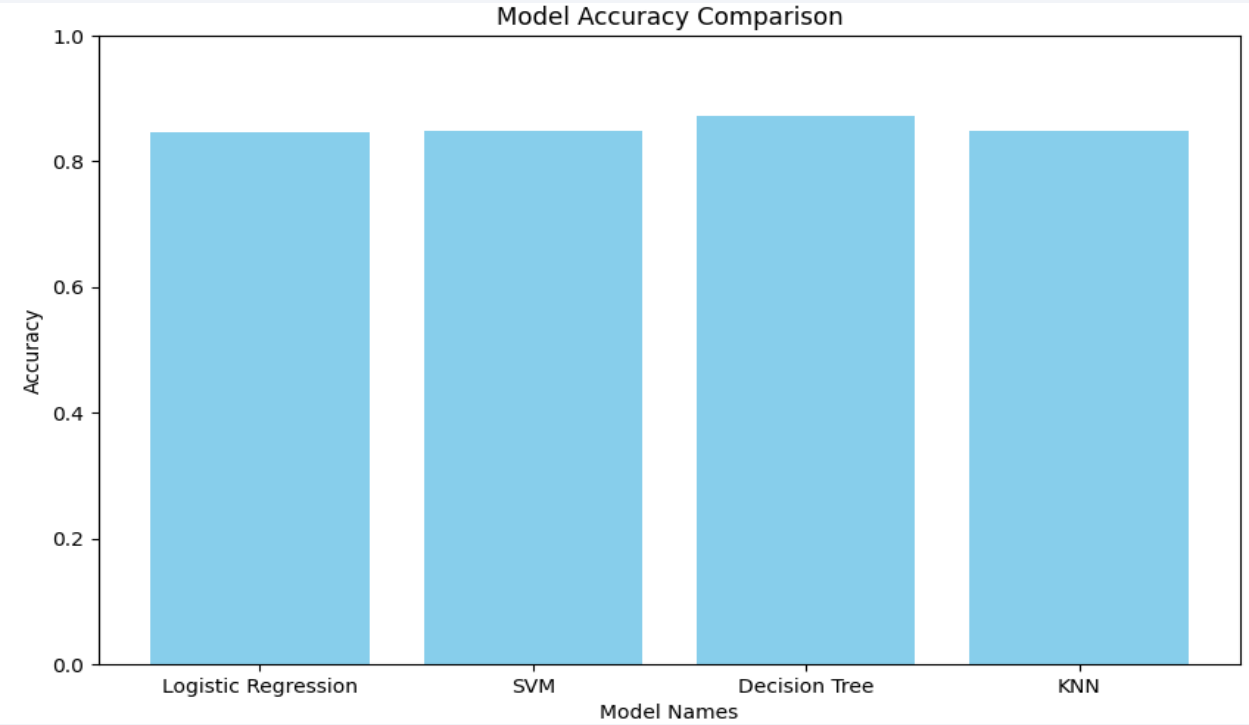
Dashboard Screenshot Showing the Correlation between various Payloads and Success Rates for all Sites

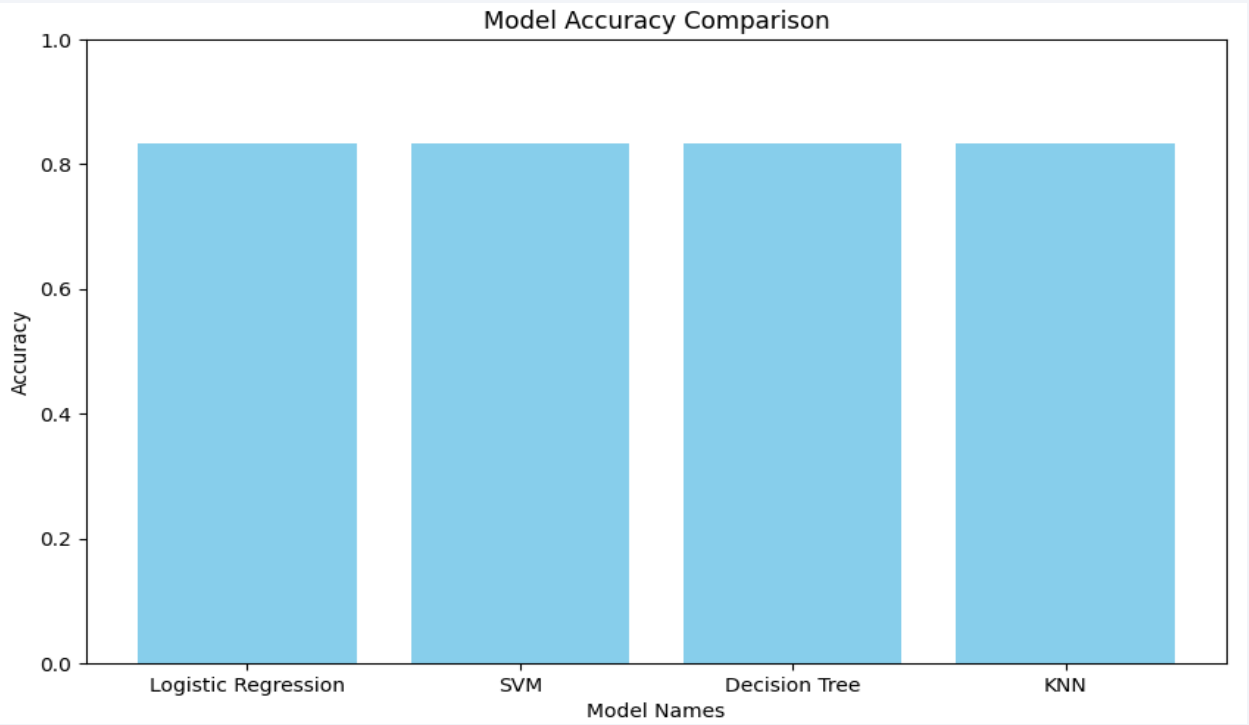
The scatter plot depicts the relationship between payload mass and launch outcomes, offering insights into the influence of payload on launch success rates. The range slider allows users to refine payload selections for a more focused analysis.

Section 5

Predictive Analysis (Classification)

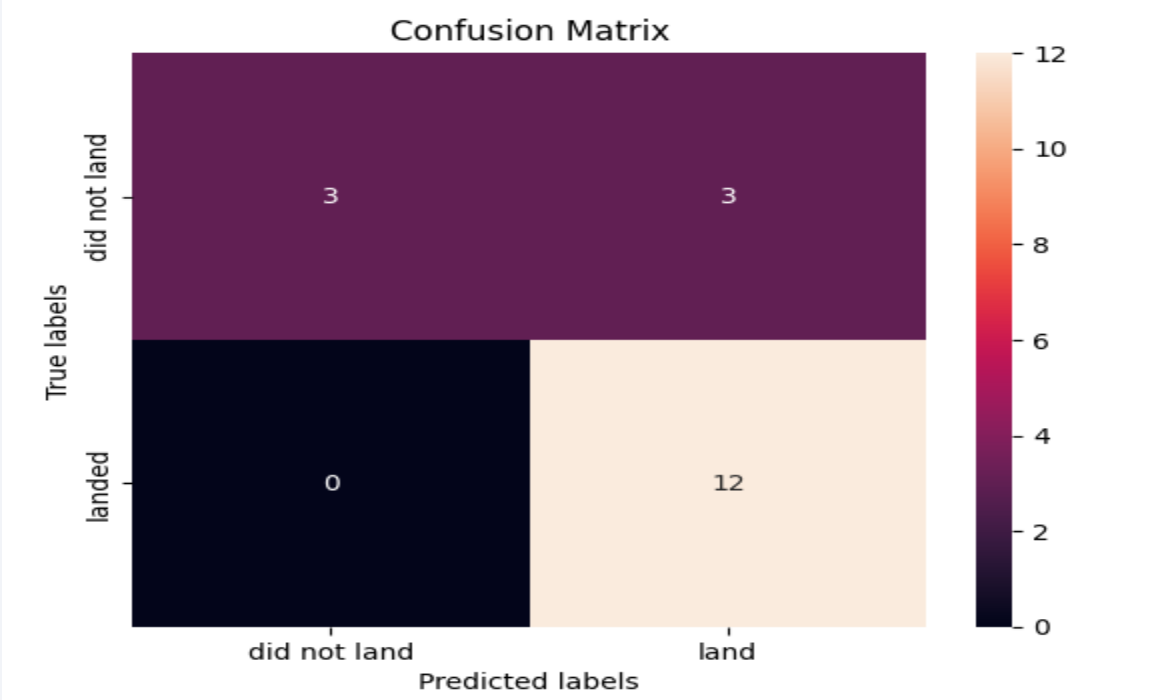
Classification Accuracy





Decision Tree model has the highest classification accuracy on the train data as shown in the figure on the left while the four models have the same accuracy on the test data as shown in the figure on the right.

Confusion Matrix



The confusion matrix indicates that the model is performing well in predicting successful rocket landings, with no instances of false negatives. However, it does have some false positives, meaning it occasionally predicts a successful landing when it doesn't occur. Overall, it is a reliable model for predicting successful landings.

Conclusions

- Rich Data, In-Depth Insights: Leveraged comprehensive SpaceX data to gain profound insights into launch outcomes, site locations, and key factors influencing success.
- Effective Data Preparation: Meticulously cleaned and prepared the dataset, ensuring high data quality for meaningful analysis.
- Dynamic Visualizations: Utilized interactive maps and insightful plots to visually represent launch site locations, success rates, and payload correlations, enhancing data understanding.
- Accurate Predictive Model: Developed a high-performing classification model, demonstrating the power of data-driven decision-making in space rocket launches.
- Strategic Insights for Future Missions: The analysis provides valuable information for SpaceX, aiding in strategic planning and decision-making for upcoming missions.

Appendix

```
@app.callback(
    Output('success-pie-chart', 'figure'),
    [Input('site-dropdown', 'value')]
)
def update_pie_chart(selected_site):
    if selected_site == 'ALL':
        # Calculate successful launches percentage for each site
        success_percentages = []

        for site in spacex_df['Launch Site'].unique():
            site_data = spacex_df[spacex_df['Launch Site'] == site]
            total_launches = len(site_data)
            successful_launches = len(site_data[site_data['class'] == 1])
            success_percentage = (successful_launches / total_launches) * 100
            success_percentages.append({'Site': site, 'Success Percentage': success_perc

        success_percentages_df = pd.DataFrame(success_percentages)
        fig = px.pie(
            success_percentages_df,
            names='Site',
            values='Success Percentage',
```

```
app.callback(
    Output('success-payload-scatter-chart', 'figure'),
    [Input('site-dropdown', 'value'),
     Input('payload-slider', 'value')]

ef update_scatter_chart(selected_site, payload_range):
    if selected_site == 'ALL':
        # Filter data for all sites within selected payload range
        filtered_data = spacex_df[(spacex_df['Payload Mass (kg)'] >= payload_range[0]) &
                                  (spacex_df['Payload Mass (kg)'] <= payload_range[1])]
    else:
        # Filter data for specific site within selected payload range
        filtered_data = spacex_df[(spacex_df['Launch Site'] == selected_site) &
                                  (spacex_df['Payload Mass (kg)'] >= payload_range[0]) &
                                  (spacex_df['Payload Mass (kg)'] <= payload_range[1])]

    # Create scatter chart figure
    fig = px.scatter(
        filtered_data,
        x='Payload Mass (kg)',
```

```
dcc.Dropdown(
    id='site-dropdown',
    options=[
        {'label': 'All Sites', 'value': 'ALL'},
        {'label': 'CCAFS LC-40', 'value': 'CCAFS LC-40'},
        {'label': 'CCAFS SLC-40', 'value': 'CCAFS SLC-40'},
        {'label': 'KSC LC-39A', 'value': 'KSC LC-39A'},
        {'label': 'VAFB SLC-4E', 'value': 'VAFB SLC-4E'}
    ],
    value='ALL',  # Default value is 'ALL'
    style={'width': '100%'}  # Set width to 100% for full page
),
html.Br(),
```

```
html.P("Payload range (Kg):"),
dcc.RangeSlider(
    id='payload-slider',
    min=0,
    max=10000,
    step=2500,  # Set step size to 2500kg
    marks={i: str(i) for i in range(0, 10001, 2500)},
    value=[0, 10000]
),
```

- %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
- %sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
- %sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Customer" = 'NASA (CRS)';
- %sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
- %sql SELECT MIN(Date) FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';

Appendix

- %sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;
- %sql SELECT "Mission_Outcome", COUNT(*) as "Total" FROM SPACEXTABLE GROUP BY "Mission_Outcome";
- %sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE);
- %%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'
    END AS "Month Names",
    "Date",
    "Landing_Outcome",
    "Booster_Version",
    "Launch_Site"
FROM SPACEXTABLE
WHERE substr(Date, 1, 4) = '2015' AND "Landing_Outcome" = 'Failure (drone ship)';
```

Thank you!