IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Kalp Vora
25 Decemeber 2025

Outline

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

Executive Summary

Objective

- Predict the success of the SpaceX Falcon 9 first-stage landing to estimate launch cost and competitiveness.

Methodology

- Collected launch data using the SpaceX REST API and web scraping.
- Performed data wrangling, exploratory data analysis using SQL and visualization.
- Built interactive visual analytics using Folium and Plotly Dash.
- Developed and evaluated multiple classification models.

Key Results

- Launch success rates improve with higher flight numbers and optimized payload ranges.
- Certain orbits (e.g., SSO, GEO, ES-L1) achieved consistently high success rates.
- The best-performing classification model achieved approximately **83% test accuracy**.

Introduction

SpaceX has revolutionized the commercial space industry by reusing Falcon 9 first-stage boosters, significantly reducing launch costs.	The success or failure of a booster landing directly affects mission cost, operational efficiency, and launch competitiveness.
Historical Falcon 9 launch data provides insights into how factors such as payload mass, orbit type, and flight number affect landing success.	This project analyzes these factors and develops a classification model to predict Falcon 9 first-stage landing success.

Section 1

Methodology

Methodology

Falcon 9 launch data was collected from multiple public sources and consolidated into a single analytical dataset.

Data was cleaned, processed, and explored to identify patterns influencing first-stage landing success.

Interactive visual analytics were developed to examine geographic and operational factors.

Predictive classification models were built, tuned, and evaluated to estimate landing success probability.

Data Collection

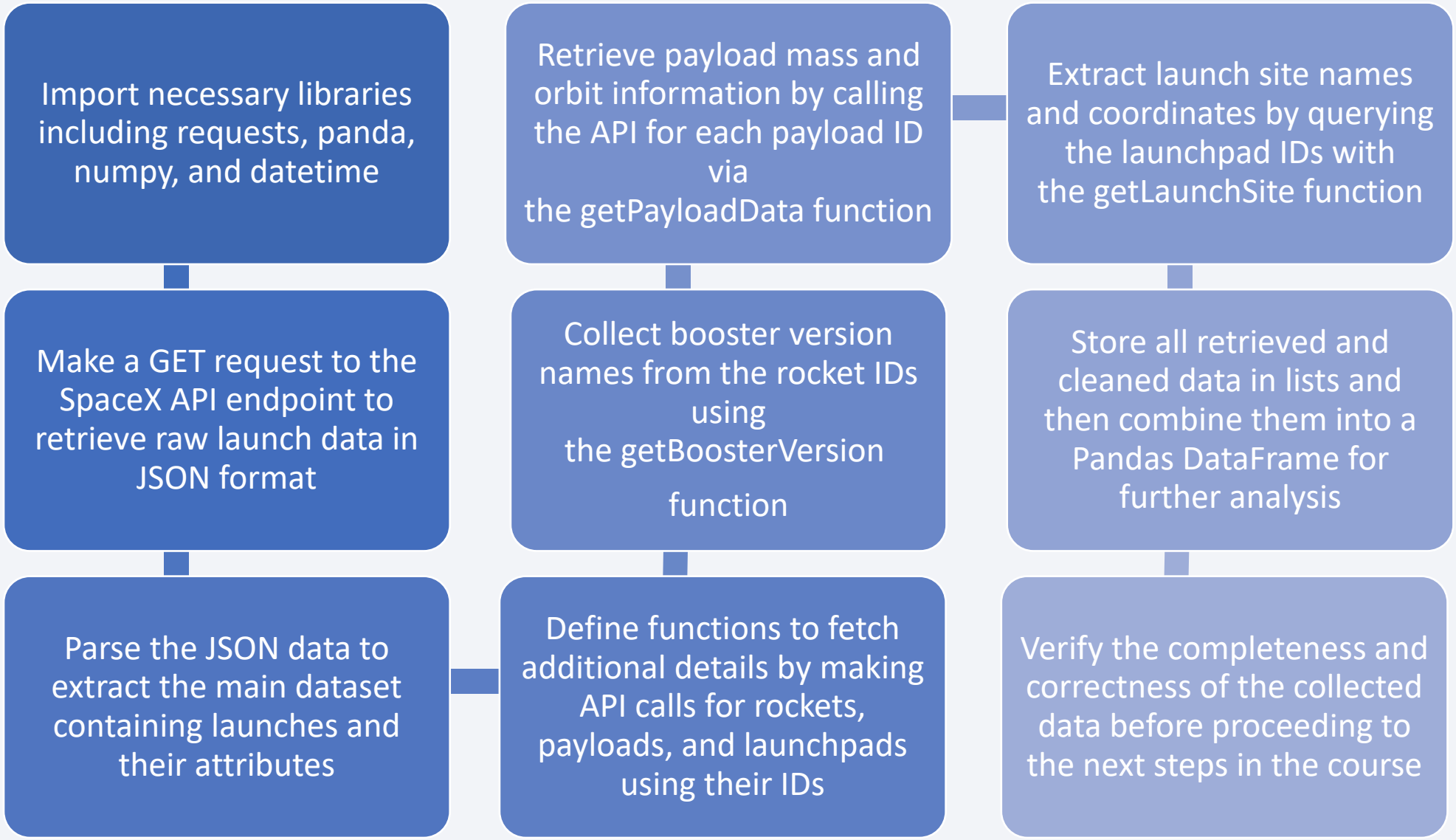
Launch, rocket, payload, and landing data were retrieved using the SpaceX REST API.

Multiple API endpoints were queried to obtain mission details, booster versions, orbit types, and outcomes.

API responses were parsed and stored as structured Pandas DataFrames for downstream analysis.

Github Repo: Data Collection Notebook

Data Collection

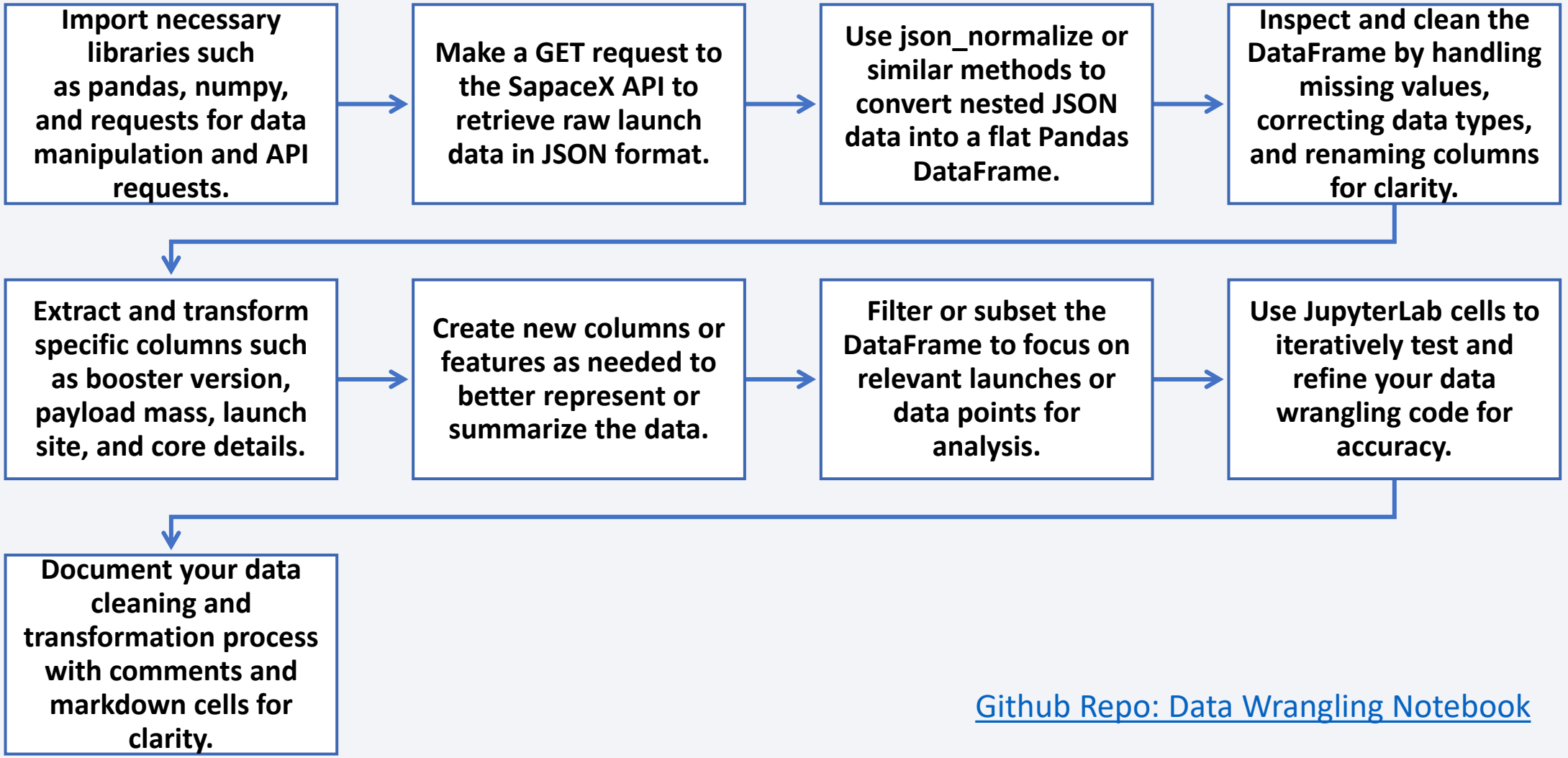


Data Collection - Scraping

1. Import libraries such as requests and Beautiful Soup to perform web requests and parse HTML content.
2. Make a GET request to the target website URL to retrieve the raw HTML page source.
3. Parse the HTML content with BeautifulSoup to build a structured parse tree for easy navigation.
4. Identify and locate the specific HTML elements that contain the data you want to scrape (e.g., tables or divs).
5. Iterate through these elements to extract the relevant text or attribute data systematically.
6. Clean and preprocess the extracted data by removing unwanted characters and formatting inconsistencies.
7. Organize the cleaned data into lists or dictionaries to prepare for analysis.
8. Convert the organized data into a Pandas DataFrame within JupyterLab for further manipulation and visualization.
9. Validate the scraped data to ensure accuracy and completeness before proceeding with analysis or modeling.

Github Repo: Data Scraping Notebook

Data Wrangling



Github Repo: Data Wrangling Notebook

EDA with SQL

- **Retrieve unique launch sites:**
  Display the distinct names of all launch sites involved in the space missions.
- **Filter launch sites starting with 'CCA':**
  Select 5 records where the launch site names begin with the string 'CCA'.
- **Identify booster versions with maximum payload mass:**
  Use a subquery with an aggregate function to find booster versions that carried the maximum payload.
- **List failure landing outcomes for 2015:**
  Extract records showing failure landing outcomes on drone ships, booster versions, and launch sites for each month in 2015 using substring functions to parse dates.
- **Rank landing outcomes between specific dates:**
  Count and rank landing outcomes (e.g., Failure (drone ship), Success (ground pad)) between 2010-06-04 and 2017-03-20 in descending order.

Github Repo: EDA with SQL Notebook

EDA with Data Visualization

Scatter Point Charts

- **Flight Number vs Launch Site:** To visualize how flight numbers are distributed across launch sites and their success/failure outcomes.
- **Flight Number vs Orbit:** To explore the distribution of flights by orbit type and success class.
- **Payload Mass vs Launch Site:** To examine how payload mass varies with different launch sites and success outcomes.
- **Payload Mass vs Orbit:** To see the relationship between payload mass and orbit types, along with success classification.

Line Chart

- **Year vs Average Success Rate:** To observe trends in launch success rates over time.

Github Repo: EDA with SQL Notebook

Build an Interactive Map with Folium

Map Objects Added to My Folium Map

- **Markers:** I added markers to highlight specific points of interest, such as launch sites and launch results. These helped me visually identify exact locations and included labels to show additional information like success/failure status or distance.
- **Circles:** I added circles around launch sites to emphasize their area and provide a sense of scale. Each circle included a popup with the site name.
- **Lines (PolyLine):** I used lines to connect coordinates, for example between a launch site and the closest coastline, to visualize spatial relationships and distances.
- **Marker Clusters:** I grouped nearby markers using clustering to reduce clutter and improve map readability, especially when displaying many launch results.
- **Divicon Markers:** I added custom text labels directly on the map, such as distance measurements, using DivIcon markers.
- **Purpose:** These objects enhanced the map's visualization, making it easier to understand spatial distributions, relationships, and key details of SpaceX launch sites and mission results.

Github Repo: Folium Notebook

Build a Dashboard with Plotly Dash

Plots and Graphs in My Plotly Dash Dashboard

- **Pie Chart:** Shows the proportion of successful vs. failed launches, helping me quickly visualize overall launch success rates.
- **Scatter Plot:** Displays launch outcomes, e.g., payload mass vs. launch site or booster version, helping identify patterns or correlations.
- **Bar Chart:** Compares success counts or rates across launch sites or other categorical variables.
- **Histogram/Box Plot:** Shows distributions of payload mass or other numeric variables for deeper insight.

Interactive Components Added:

- **Dropdown Menus:** Let users select launch sites or booster versions to dynamically filter data.
- **Range Sliders:** Filter data by payload mass or date range to explore subsets of launches.

Github Repo: Plotly Notebook

Build a Dashboard with Plotly Dash

Purpose and Benefits of the Dashboard:

- **Hover Tooltips:** Provide detailed information on data points for better insight.
- **Callbacks:** Connect user inputs (dropdowns, sliders) to update plots dynamically, making the dashboard interactive and responsive.

The benefits of Plots and Interactions:

- To explore relationships between launch parameters and success outcomes interactively.
- To enable filtering and drilling down into specific launch sites, payload ranges, or booster versions.
- To visualize distributions and proportions clearly and intuitively.

To provide an engaging experience where users can manipulate inputs and immediately see results, reinforcing insights from the SpaceX dataset.

Github Repo: Plotly Notebook

Predictive Analysis (Classification)

Building and Evaluating Classification Models

Data Preparation:
Load and preprocess dataset (handle missing values, encode categorical variables, scale features)
Split data into training and testing sets
Model Building:
Select classification algorithms (Logistic Regression, Decision Tree, Random Forest, SVM)
Train models on training data
Model Evaluation:
Predict on test data and evaluate using accuracy, precision, recall, F1-score, and confusion matrix
Use cross-validation to check model stability
Model Improvement:
Tune hyperparameters (Grid Search, Randomized Search)
Apply feature engineering or feature selection

Github Repo: Machine Learning Notebook

Predictive Analysis (Classification)

How the Best Model Was Chosen:

- Compared multiple models using validation metrics (accuracy, precision, recall, F1-score).
- Prioritized models with **balanced performance**, not just highest accuracy.
- Used cross-validation results to ensure consistent performance and avoid overfitting.
- Selected the model that showed strong generalization across unseen data.

Final Validation:

- Evaluated the selected model on the test set to confirm real-world performance.

Github Repo: Machine Learning Notebook

Results

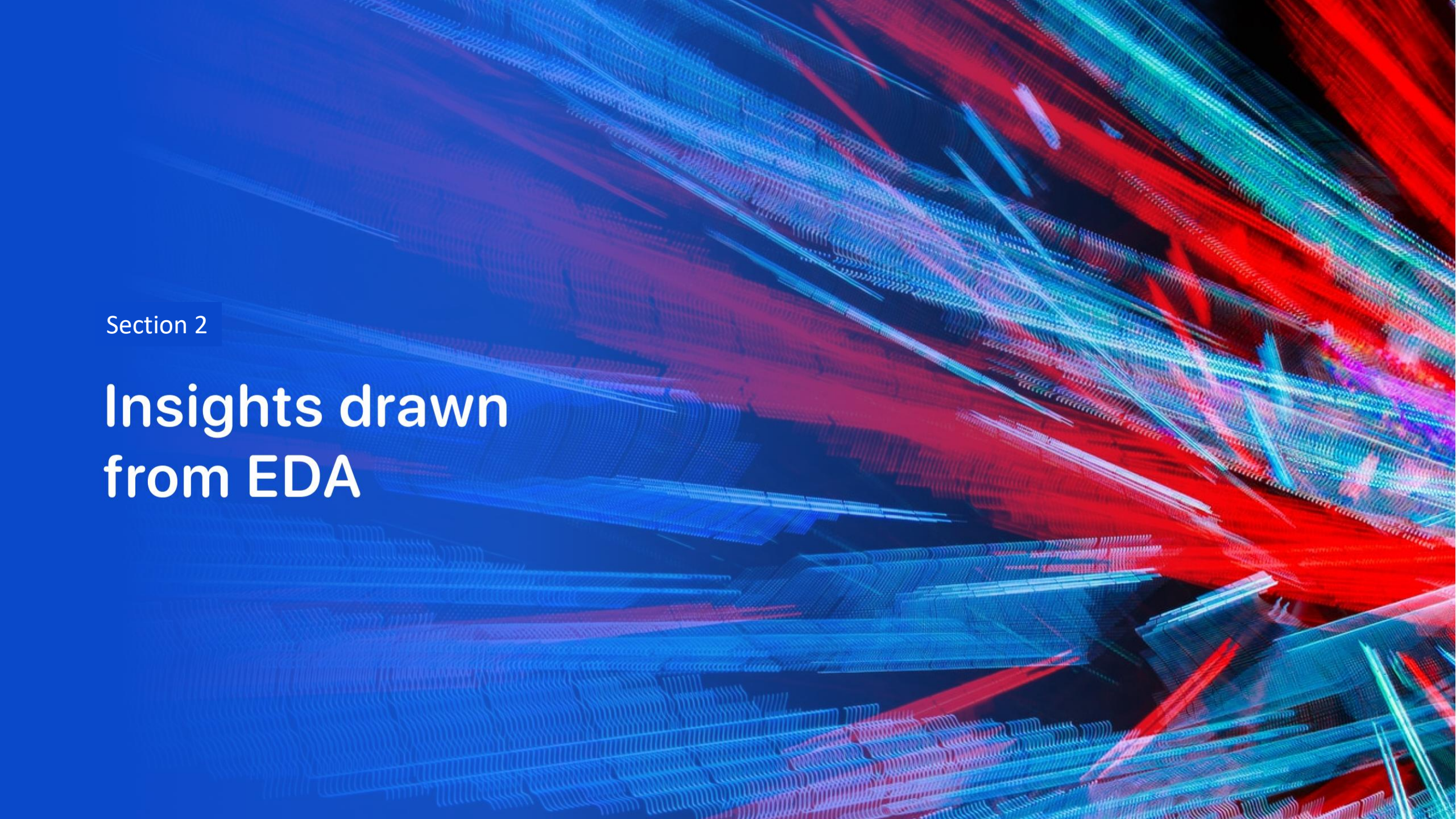
Exploratory Data Analysis (EDA) Results

- **Launch Site Insights:**
  - Visualized SpaceX launch sites on interactive Folium maps.
  - Identified geographic distribution and clusters of launch sites.
  - Explored success rates by launch site, revealing some sites have higher success probabilities.
  - Analyzed distances from launch sites to nearby coastlines and potential impacts.
- **Payload and Booster Analysis:**
  - Examined payload mass distributions and their relation to launch success.
  - Investigated booster version categories and their influence on launch outcomes.
  - Used scatter plots and pie charts to visualize payload vs. success and booster performance.
- **Temporal Trends:**
  - Analyzed launch success trends over time.
  - Identified patterns or anomalies in launch outcomes across different periods.

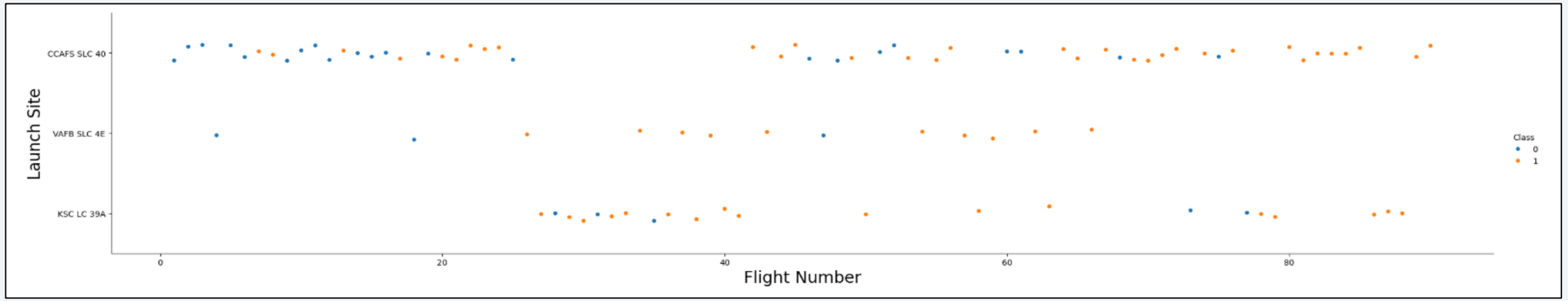
Results

Predictive Analysis Results

- **Model Development:**
  - Built multiple classification models (Logistic Regression, Decision Tree, Random Forest) to predict launch success.
  - Used training and testing splits to validate model performance.
- **Evaluation Metrics:**
  - Assessed models using accuracy, precision, recall, F1-score, and confusion matrices.
  - Applied cross-validation to ensure model robustness.
- **Model Improvement & Best Model Selection:**
  - Performed hyperparameter tuning (Grid Search/Randomized Search) to optimize models.
  - Applied feature engineering and selection to enhance predictive power.
  - Selected the model with the highest predictive accuracy and balanced metrics.
  - Validated the final model on test data to confirm generalization.

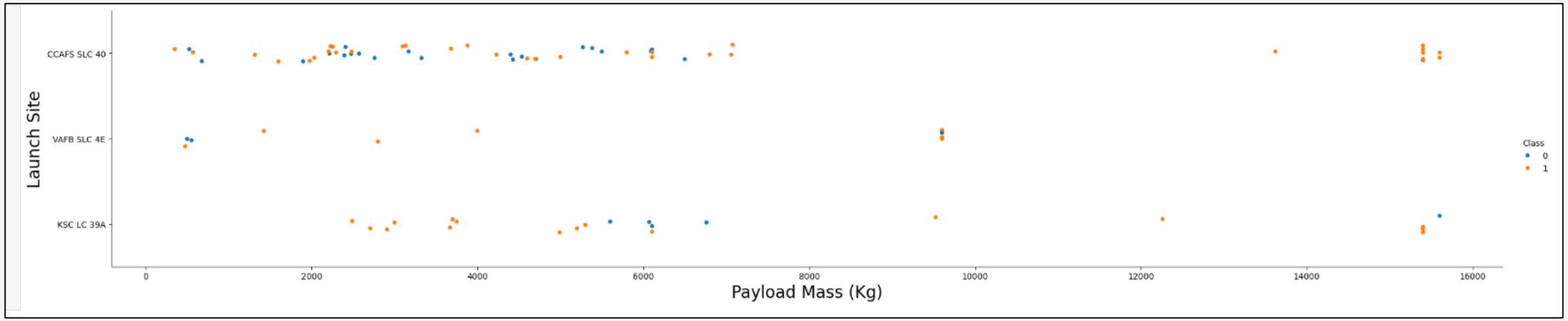

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



- The early launches were mostly met with fail for both CCAFS SLC 40 & VAFB SLC 4E, however, the success rate improved with each flight.
- KSC LC 39A shows no clear pattern but marginally higher success rates than the rest.
- The most recent launches of all the three have mostly been successful.

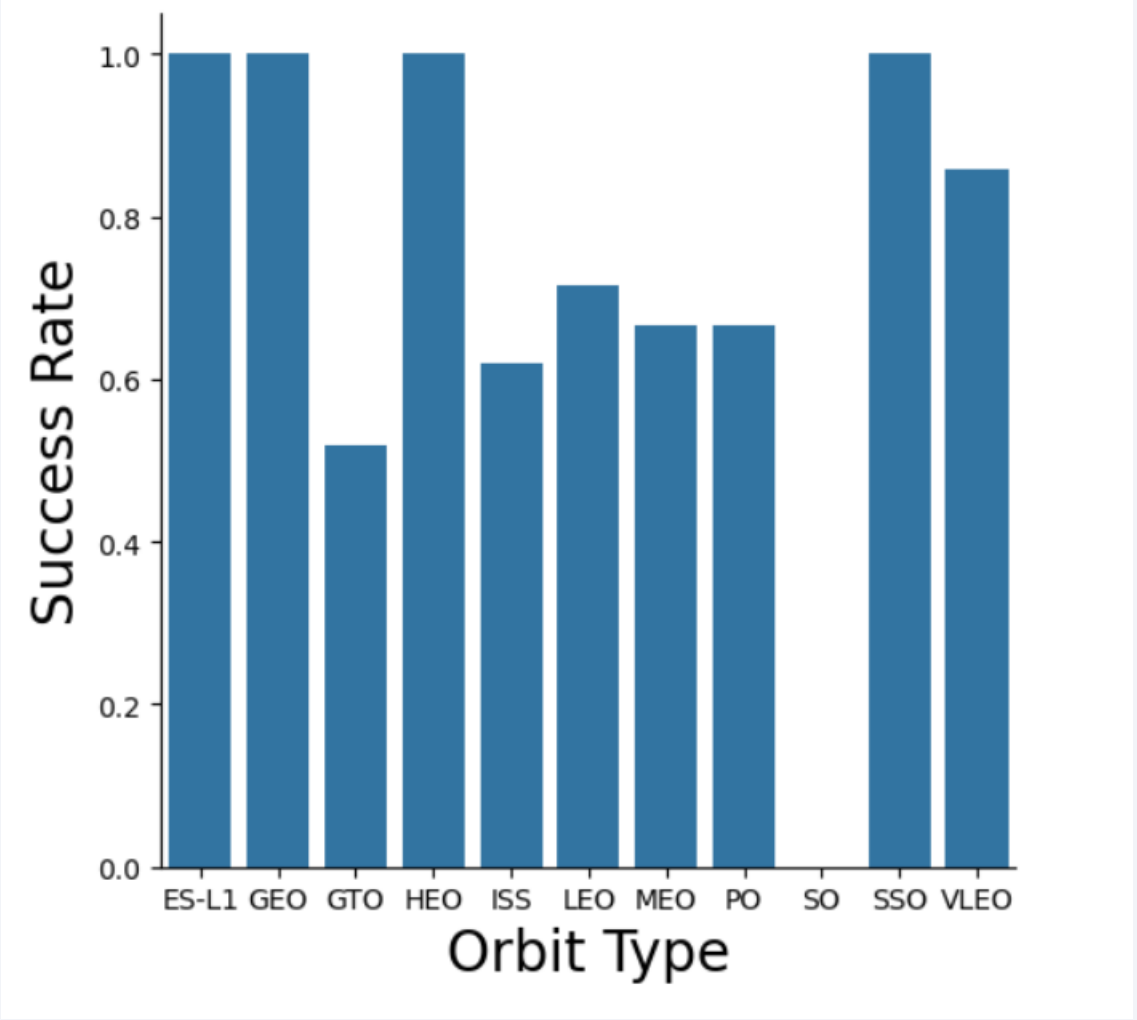
Payload vs. Launch Site



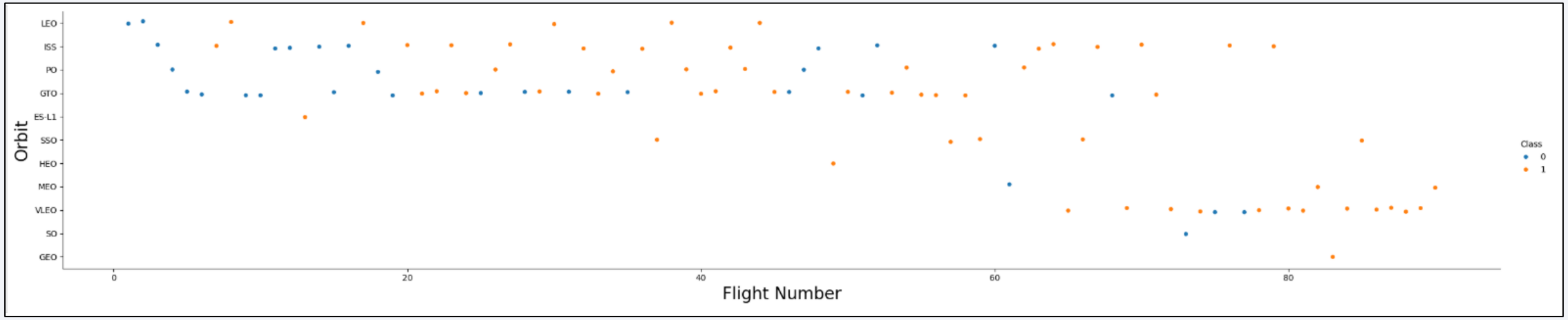
- All the three launch sites have see a much higher success rates at heavier payload masses.
- The greatest number of launches have been done at a payload mass of 8000kg or less.
- CCAFS SLC 40 sees a fluctuation in the number of successes while VAFB SLC 4E and KSC LC 39A mostly have all success through the entire range of payload masses.

Success Rate vs. Orbit Type

- ES-L1, GEO, HEO and SSO have 100% success rates
- VLEO is also pretty successful at around 85%.
- GTO, ISS, LEO, MEO AND PO all have medium success rates at around 50% on average.
- SO has 0% success rate so far, perhaps an anomaly.

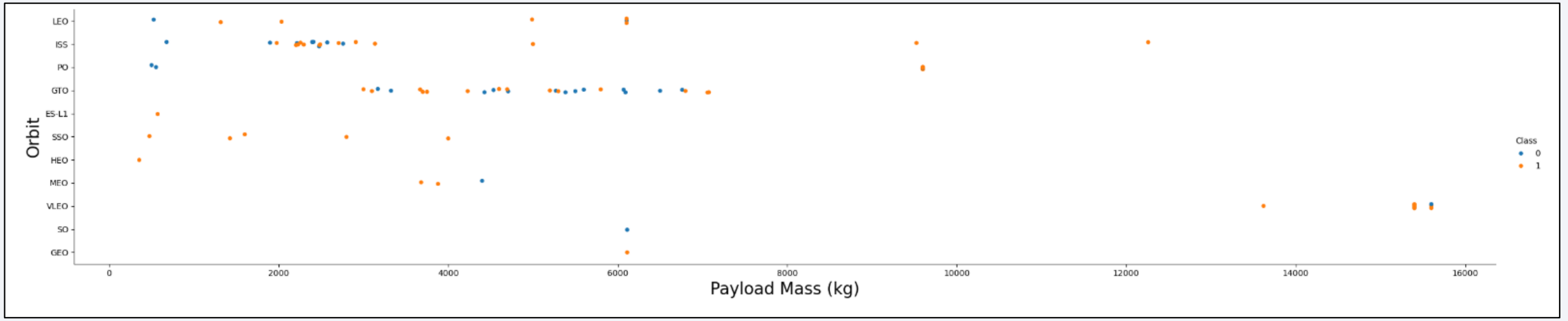


Flight Number vs. Orbit Type



- As the flight number increases the success rate also increases for most orbit types except the ones with less launches.

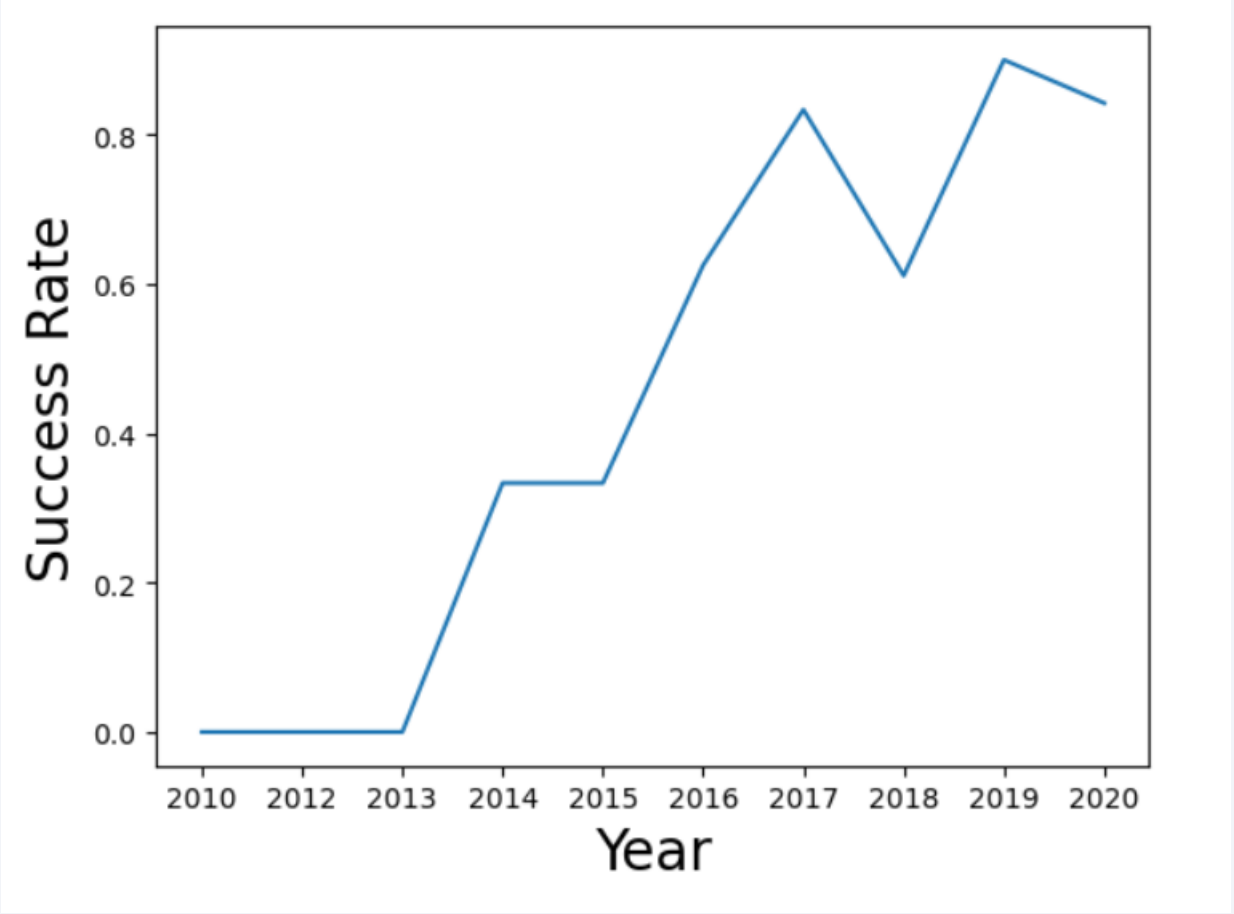
Payload vs. Orbit Type



- Higher payload masses have much higher success rate
- Early instances have often resulted in failures for launch sites LEO, ISS, and PO

Launch Success Yearly Trend

- No successes until 2014
- Sees an overall rise after 2014.
- 2013-2017 sees the highest increase in rate
- The success rate starts fluctuating post 2017



All Launch Site Names

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;
```

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

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

Launch Site Names Begin with 'CCA'

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

```
[12]: %sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

[12]:

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

All landing attempts either failed either or hadn't been attempted. While all the mission had a successful launch.

Total Payload Mass

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

```
%%sql
SELECT SUM("PAYLOAD_MASS__KG_") AS "Total_Payload_Mass"
FROM SPACEXTBL
WHERE "Customer" = 'NASA (CRS)';
```

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

Total_Payload_Mass
45596

Average Payload Mass by F9 v1.1

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

```
%%sql
SELECT AVG("PAYLOAD_MASS__KG_") AS "Average_Payload_Mass"
FROM SPACEXTBL
WHERE "Booster_Version" = 'F9 v1.1';
```

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

Average_Payload_Mass
2928.4

First Successful Ground Landing Date

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

Hint:Use min function

```
%%sql
SELECT MIN(DATE) FROM SPACEXTBL
WHERE "Landing_Outcome" = "Success (ground pad)"
```

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

MIN(DATE)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
%%sql
SELECT "Payload", PAYLOAD_MASS__KG_ FROM SPACEXTBL
WHERE ("Landing_Outcome" = "Success (drone ship)") and "PAYLOAD_MASS__KG_" >4000 and "PAYLOAD_MASS__KG_" <6000;
```

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

Payload	PAYLOAD_MASS__KG_
JCSAT-14	4696
JCSAT-16	4600
SES-10	5300
SES-11 / EchoStar 105	5200

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%%sql
SELECT "Mission_Outcome", COUNT(*) AS "Total_Count"
FROM SPACEXTBL
GROUP BY "Mission_Outcome";
```

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

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

Boosters Carried Maximum Payload

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

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

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

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

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.

```
%%sql
SELECT substr(Date, 6,2) as "Month", "Landing_Outcome", substr(Date,0,5) as "Year", "Launch_Site" from SPACEXTBL
WHERE "Landing_Outcome" = 'Failure (drone ship)' and substr(Date,0,5)='2015';
```

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

Month	Landing_Outcome	Year	Launch_Site
01	Failure (drone ship)	2015	CCAFS LC-40
04	Failure (drone ship)	2015	CCAFS LC-40

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

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

```
%%sql
SELECT "Landing_Outcome", COUNT(*) AS "Outcome_Count" FROM SPACEXTBL
WHERE "Date" BETWEEN '2010-06-04' and '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY "Outcome_Count" DESC;
```

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

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 on an interative map

Key Features

- **Launch Site Circles (folium.Circle)**: Added circles around each launch site coordinate to highlight their locations with a visible radius and popup labels showing the site names.
- **Launch Site Markers (folium.Marker)**: Placed markers at each launch site coordinate to pinpoint exact locations on the map.

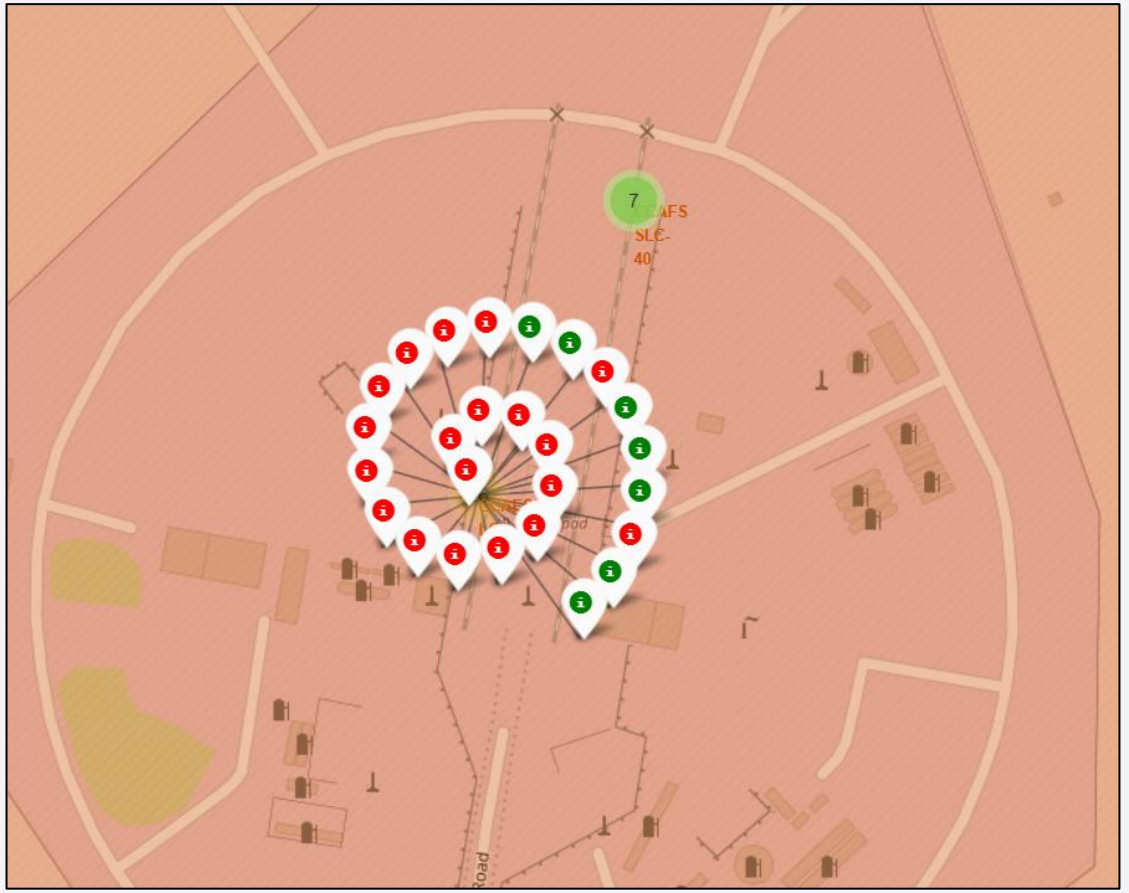
These features help visually identify and emphasize the important launch sites on the interactive map, providing a clear spatial context for further analysis.

Color Labeled Individual Launch Sites

Folium Map Features

- **Marker Clusters:** Grouped nearby launch markers to reduce clutter and improve map readability when displaying many launches.
- **Launch Result Markers:** Added individual markers for each launch, with visual indicators showing success or failure.
- **Dynamic Marker Icons:** Used different marker colors/icons to clearly distinguish successful and failed launches.

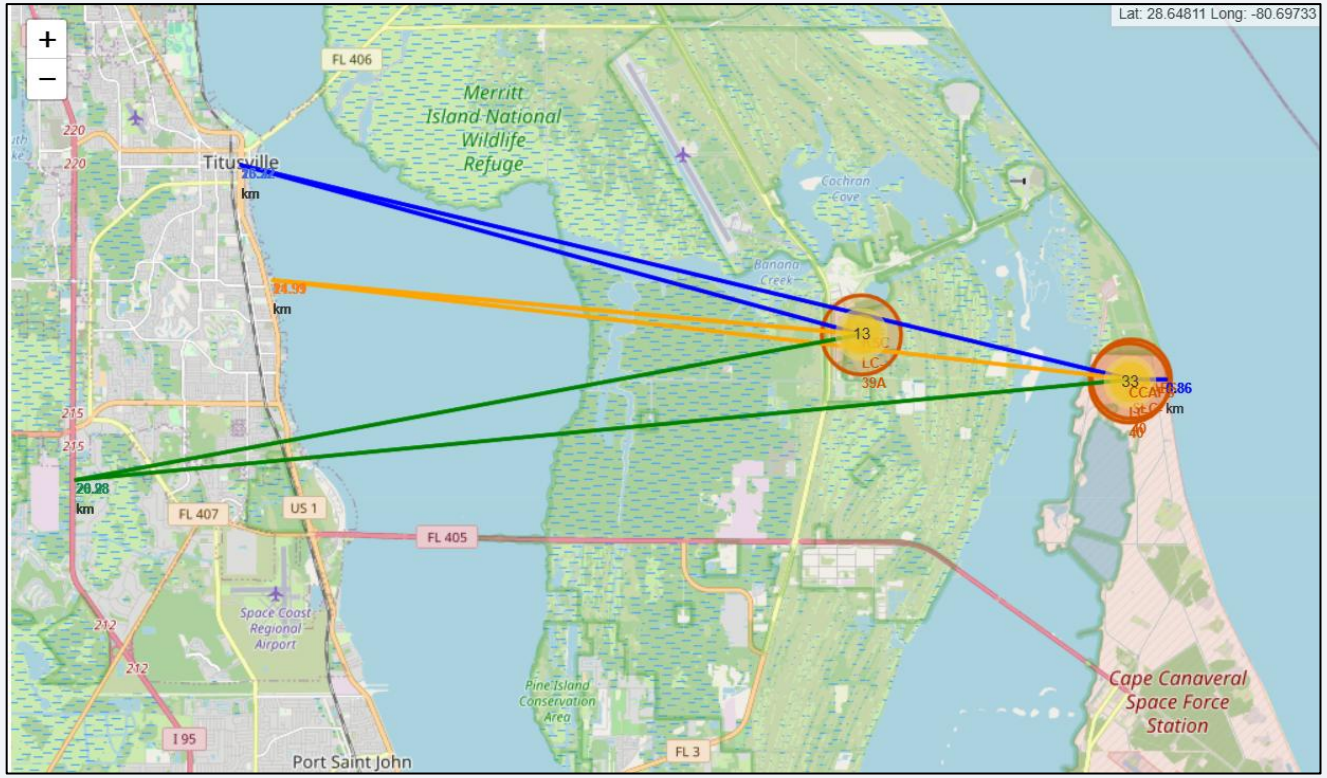
These features make the map more interactive, readable, and effective for analyzing launch outcomes.

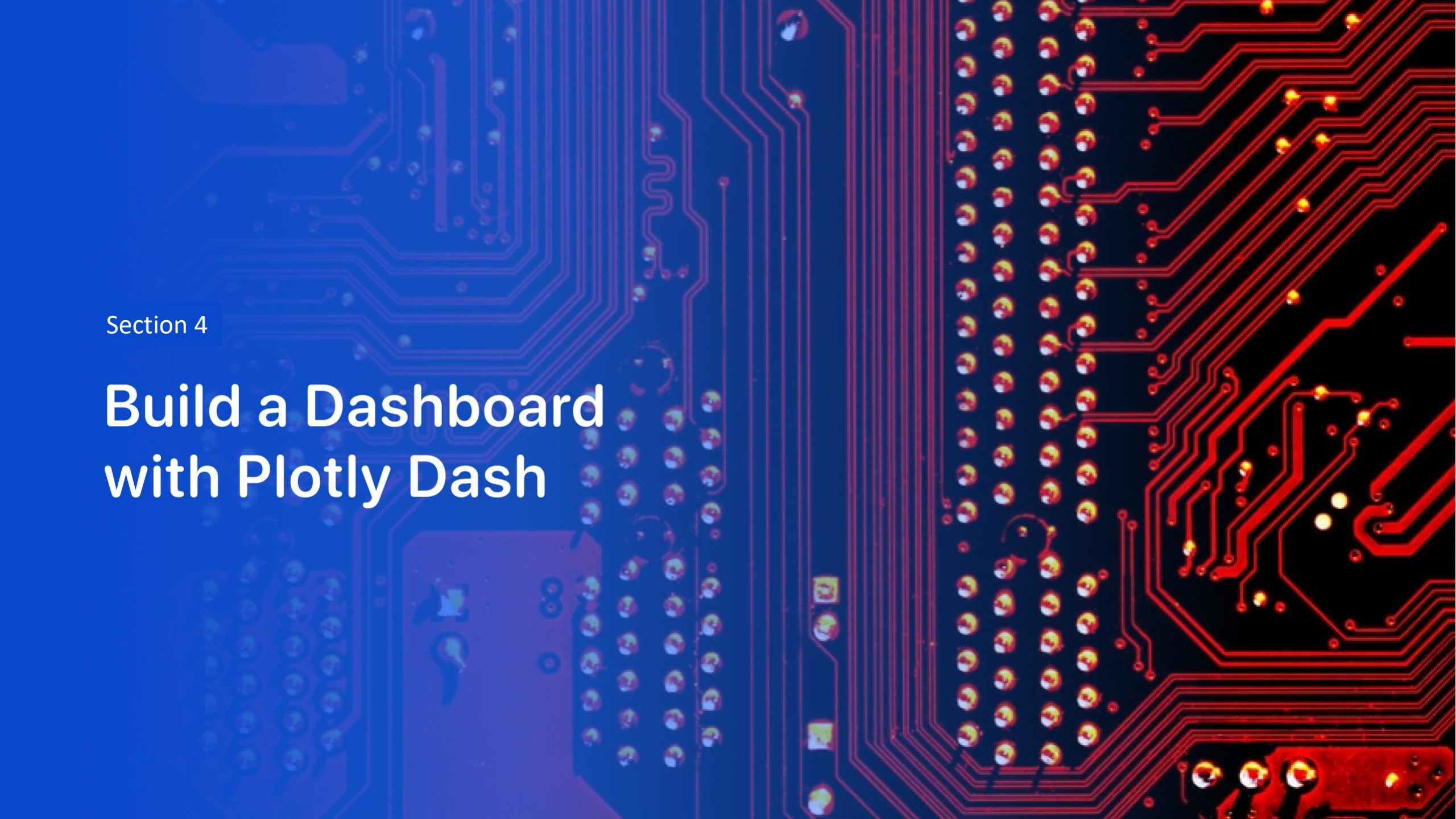


Proximity to nearby cities and transport

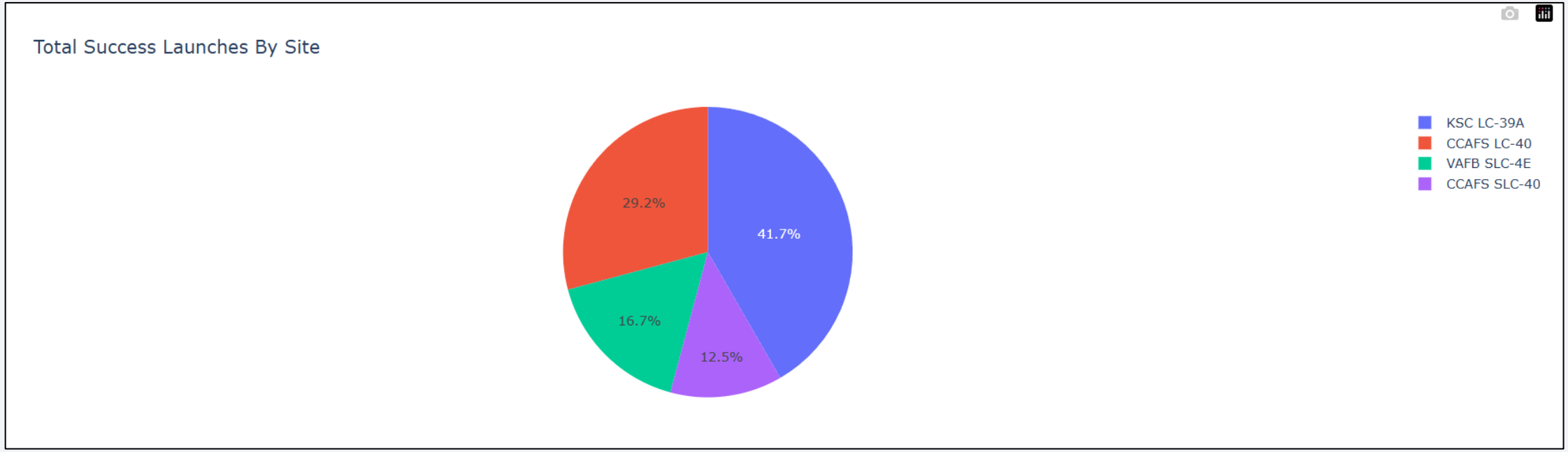
Distance and Spatial Annotations

- **Distance Markers (DivIcon):** Added custom markers showing the distance between each launch site and its nearest coastline.
- **Connecting Lines (PolyLine):** Drew lines from launch sites to coastlines to visualize spatial relationships.
- **Enhanced Visual Annotations:** Combined launch site markers, coastline points, and distance labels for clear geographic context.



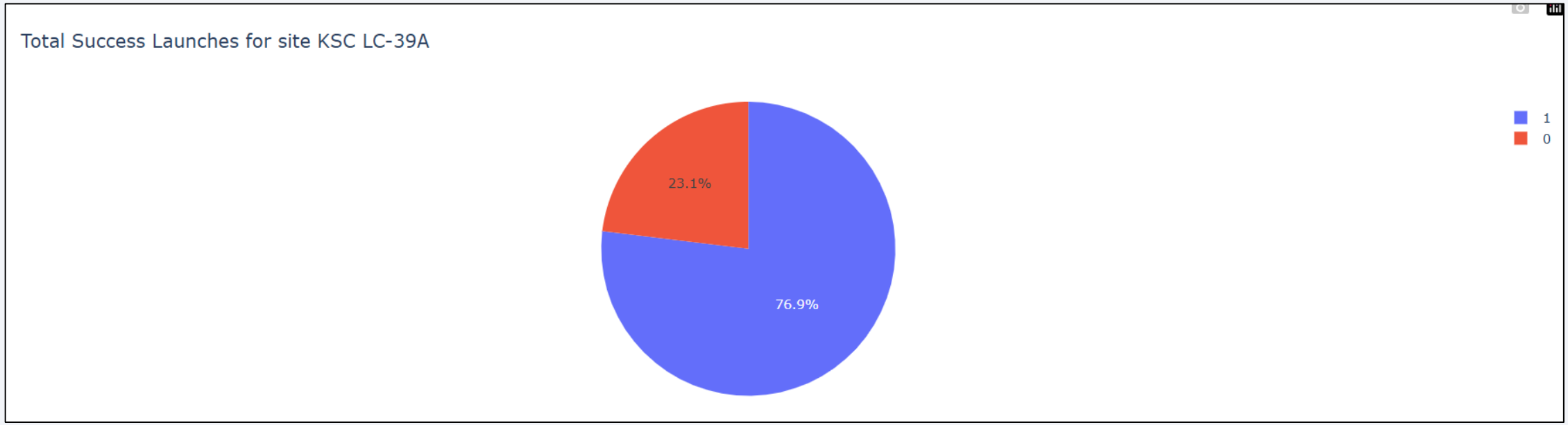

Section 4
Build a Dashboard with Plotly Dash

Total Successful Launches for each Launch Site



- This chart show the distribution of all the successful launches by each site, KSC LC 39A contributes the most to this while CCAFS SLC 40 contributes the least.

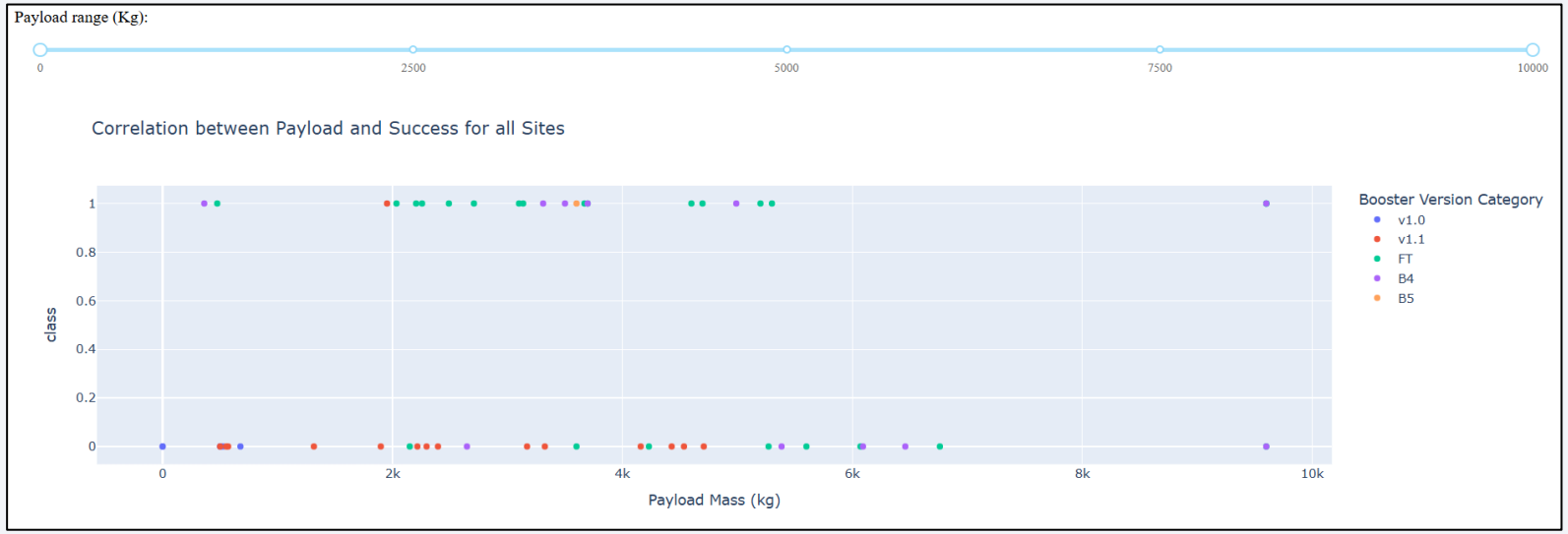
Launch Site with the Highest Success Ratio

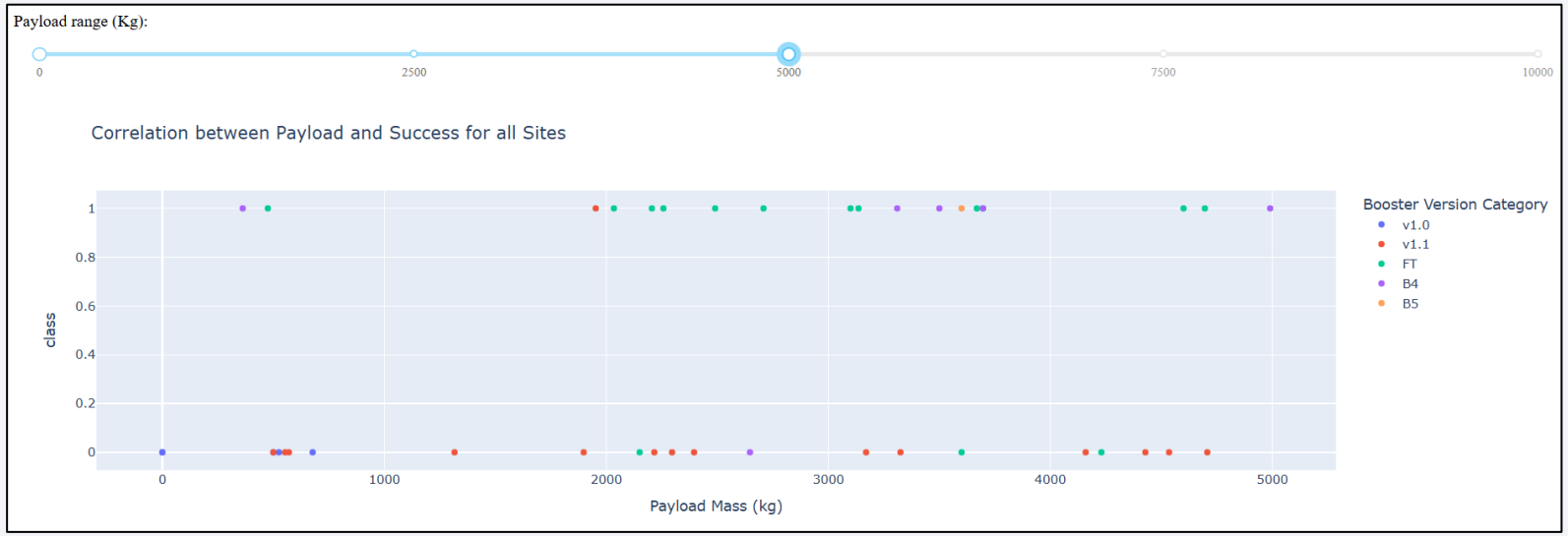


- KSC LC 39A has the highest ratio of successful launches, with just over ¾ of its launches being successful.

Launch Success vs. Payload Mass

- The success rate is highest at lower payload masses
- Most of the launches made by all the sites are between 0 to 5000 KGs



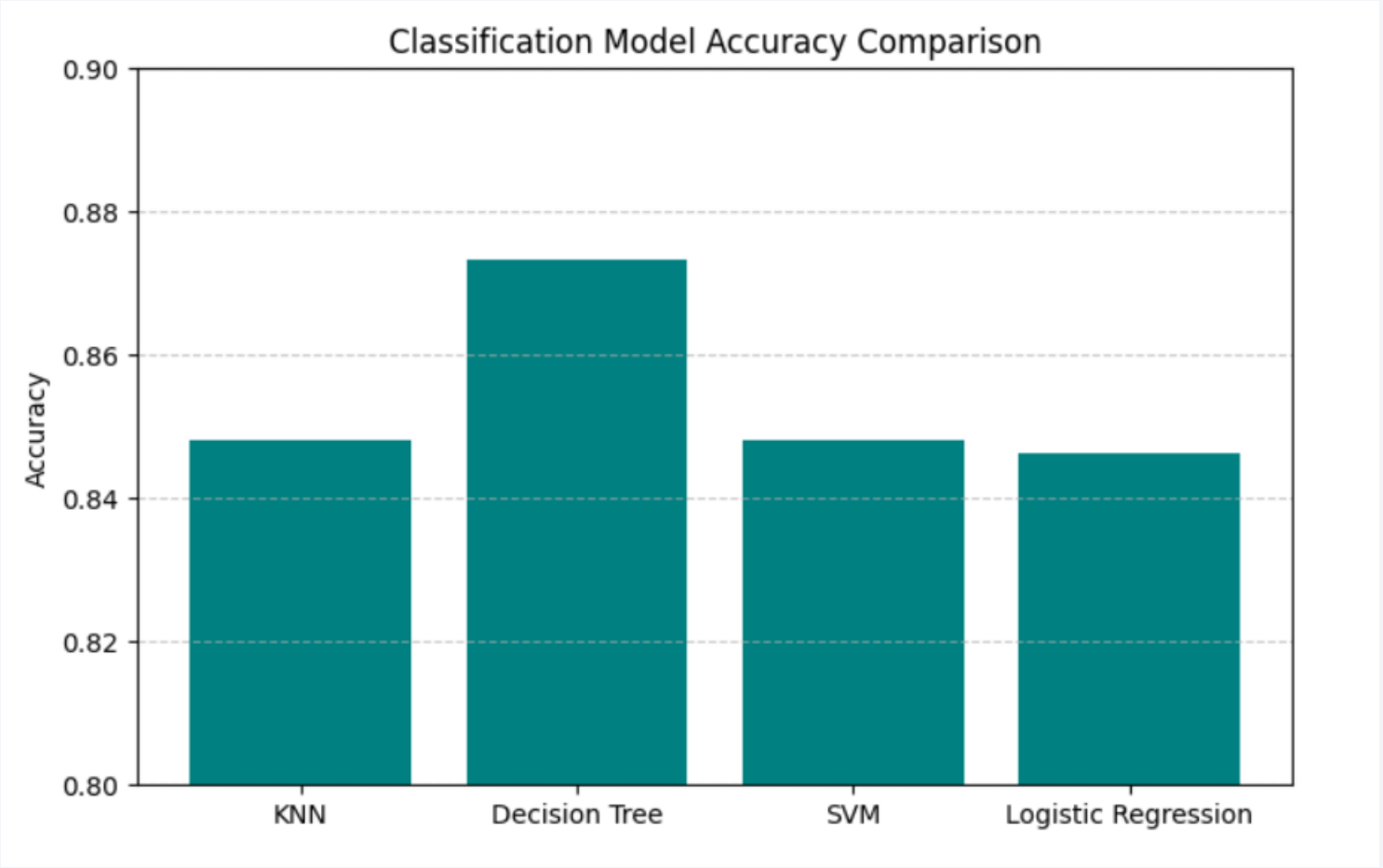


Section 5

Predictive Analysis (Classification)

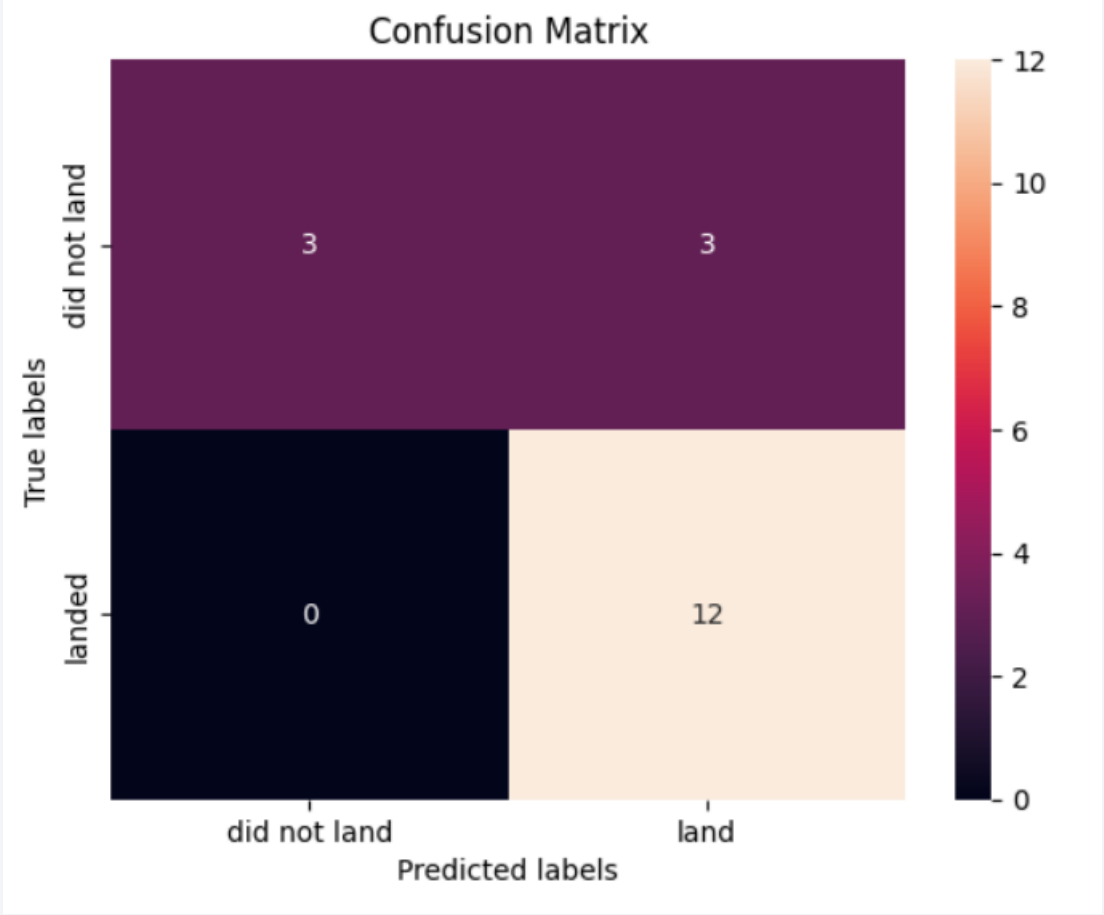
Classification Accuracy

- This bar graph shows the classification model accuracy for the different machine learning models. This includes the optimization after using gridsearch.
- Decision Tree has the highest accuracy out of all the machine learning models.



Confusion Matrix for Decision Trees Model

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



Conclusions

- Landing success is predictable: Falcon 9 first-stage landing outcomes can be predicted with good reliability using historical launch data, showing that success depends on measurable mission parameters.
- Best-performing model identified: The Decision Tree classifier achieved the highest test accuracy (~83%), outperforming Logistic Regression, SVM, and KNN by capturing non-linear feature interactions.

Key factors influencing success:

- Higher flight numbers correlate with improved success rates, reflecting technological maturity.
- Payload mass shows a positive relationship with success up to an optimal threshold.
- Certain orbits (ES-L1, GEO, HEO, SSO) achieved near-perfect success rates.

Conclusions

- Launch site strategy supports reliability: Coastal launch sites located away from population centers enhance safety and recovery efficiency, contributing to higher landing success.
- Actionable decision support:The predictive model, Folium maps, and Dash dashboards enable cost estimation and strategic analysis for competitors bidding against SpaceX.
- Limitations: Model performance is constrained by available features and historical data size.
- Future work: Incorporating weather conditions, booster reuse count, and advanced ensemble models could further improve prediction accuracy.

Appendix

Coursera – IBM Data Science Capstone ProjectProject framework, instructions, and evaluation criteria

SpaceX Launch DataPublicly available data accessed via the SpaceX REST API

Wikipedia - General background information on Falcon 9, launch sites, and orbital classifications

Github Repo – All notebooks that have been used in this project has been added to the Github Repository

Thank you!