IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

RUBEN SOLER

J ANUARY 17TH,2025

Outline

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

Executive Summary

Methodology

1. Data Collection:

- Extracted data from SpaceX's API and web scraping techniques.
- Data wrangling processes to ensure data quality.

2. Exploratory Analysis:

- Interactive visualizations using Folium and Plotly Dash.
- SQL queries to uncover trends and key metrics.

3. Predictive Models:

- Classification models applied to predict launch outcomes.
- Evaluation and optimization to select the most accurate model.

Executive Summary

Results

- Identification of the most successful launch sites and historical trends.
- Analysis of payload impact and orbit type on success rates.
- Development of interactive maps showcasing launch site locations and proximities.
- Accurate classification of launch outcomes, highlighting conditions for success.

Introduction

SpaceX revolutionized the aerospace industry by significantly reducing launch costs through the reusability of th Falcon 9 rocket's first stage. While SpaceX advertises a cost of $62 million per launch, competitors charge upwa $165 million, making reusability a key factor in their cost advantage. The ability to predict whether the first stage successfully land provides crucial insights into operational efficiency and cost savings. Such insights are valuabl companies looking to compete with SpaceX in the launch services market.

This project seeks to leverage data from previous SpaceX launches to build predictive models that can estimate success of a Falcon 9 first-stage landing. By doing so, it empowers alternate companies to make informed decis and enhance their competitiveness.

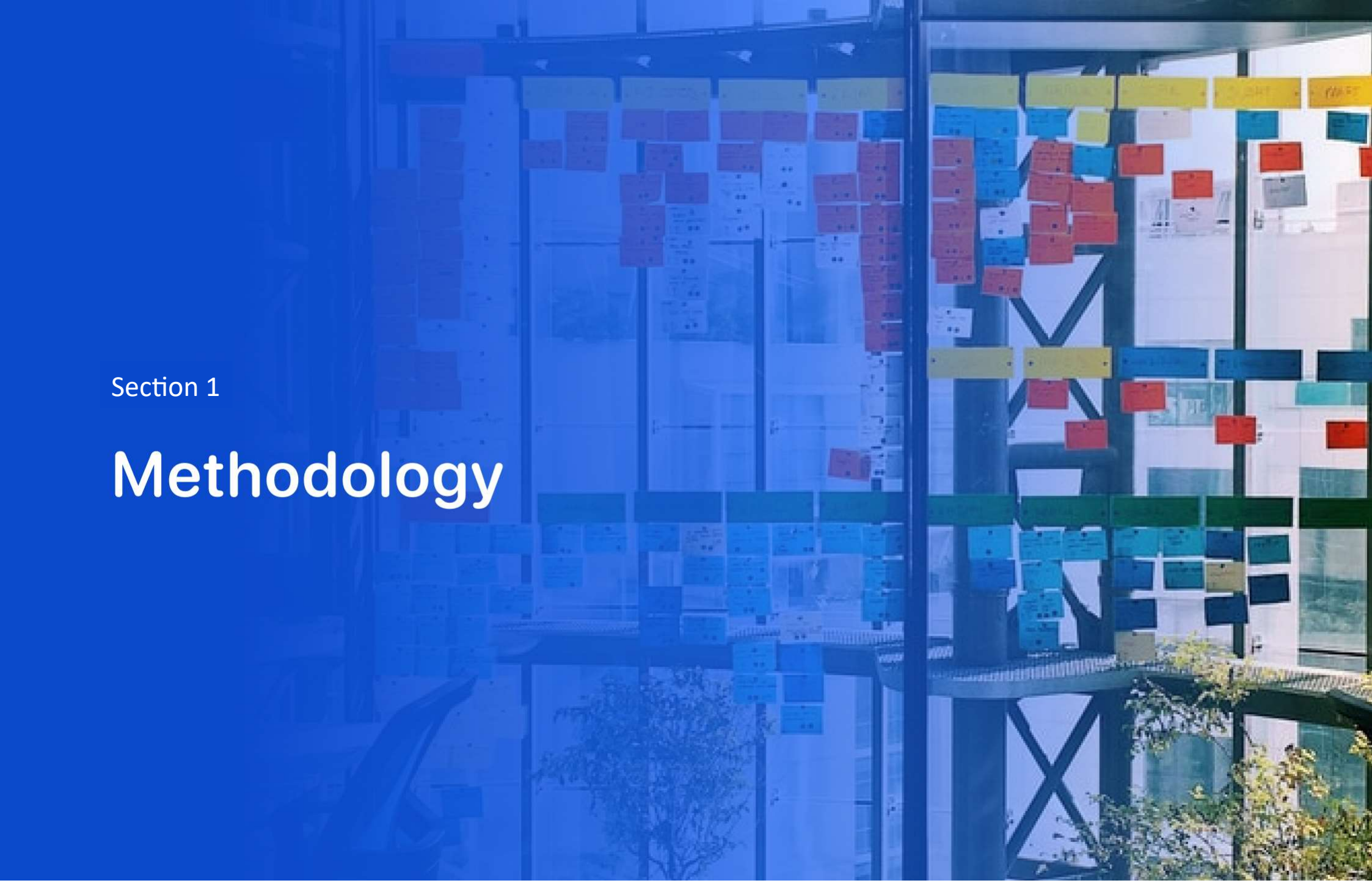
- Problems we want to find answers:

  Can we accurately predict if the Falcon 9 first stage will successfully land based on historical data?

  What features or conditions most influence landing success?

  How do variables such as payload, orbit type, and launch site impact the probability of a successful landing?

  Are there identifiable trends in the historical success rate of Falcon 9 landing ?



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

To analyze and predict the success of Falcon 9 first-stage landings, data was collected using the following methods:

1. SpaceX API:

The SpaceX API was utilized to retrieve detailed historical launch data, including launch dates, payloads, orbit types, launch s
landing outcomes, and booster specifications.

API endpoints were accessed systematically to ensure comprehensive and accurate data retrieval.

A flowchart was created to visualize the data extraction process and the relationships between various data points.

2. Web Scraping:

Web scraping techniques were employed to extract supplementary information from SpaceX's website and other publicly ava
sources.

Data on launch costs, mission objectives, and reusability metrics was scraped to complement the API data.

Python libraries like BeautifulSoup and requests were used for efficient scraping.

Data Collection – SpaceX API

Data Extraction

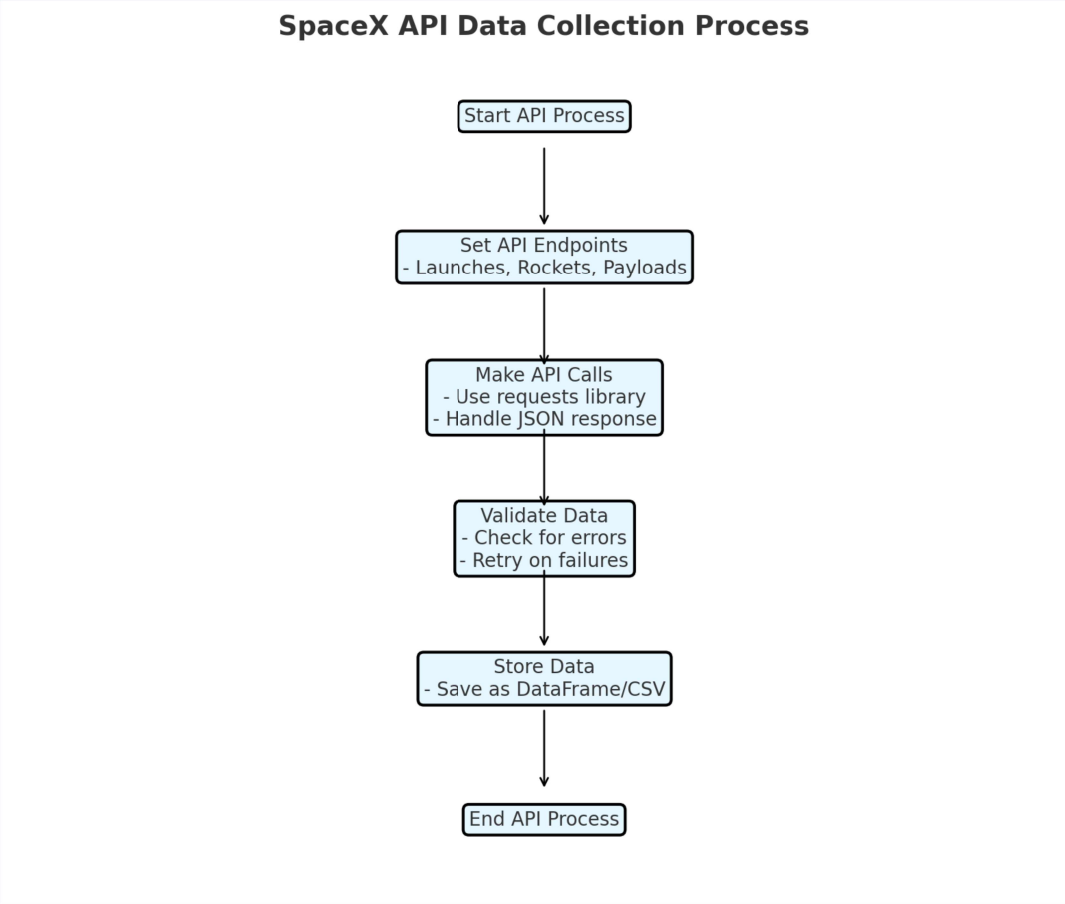
Key Source: SpaceX API.

Methodology: API calls to retrieve structured datasets, including:

- Launch dates and times.
- Booster specifications (e.g., version, reuse count).
- Payload information (e.g., mass, orbit type).
- Launch site and landing outcomes.

- Tools: Python with libraries such as requests and json to automate and parse API responses.

https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex1_data_collection.ipynb

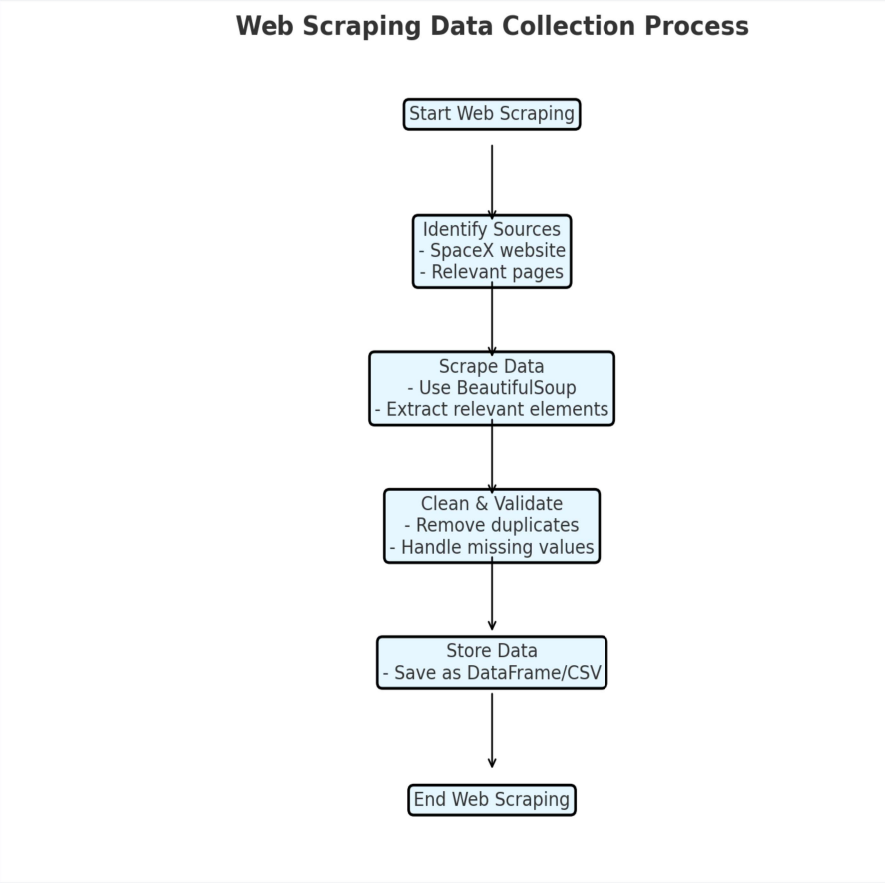


Data Collection - Scraping

Web Scraping

- Source: SpaceX’s official website and other aerospace repositories.
- Purpose: Collect additional context, such as mission objectives, historical notes, and cost benchmarks.
- Tools: Python libraries BeautifulSoup and requests.
- Process: Navigate, parse, and extract relevant HTML content.

https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex2_web_scrapping.ipynb



Data Wrangling

The data processing (wrangling) phase ensured the raw data collected from SpaceX API and web scraping was cleaned, structured, and prepared for analysis. The following steps outline the process:

Data Cleaning:

Handle Missing Values: Identified missing fields in key columns (e.g., landing_outcome). Filled or removed rows based on their significance to the analysis.

Remove Duplicates: Verified unique records for launches, boosters, and payloads.

Standardize Formats: Unified date, numeric, and categorical formats (e.g., orbit type labels).

Data Transformation:

Feature Engineering: Created new variables, such as reused_booster and success_rate.

Categorical Encoding: Converted text-based data (e.g., landing_outcome) into numerical labels.

Data Scaling: Normalized payload mass and other numeric features for machine learning.

Data Wrangling

Data Integration:

Merged datasets from multiple sources (API and web scraping).

Ensured schema consistency between merged tables.

Validation:

Checked data consistency with known values (e.g., total launches).

Verified relationships between features (e.g., orbit vs. payload_mass).

https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex3_data_wrangling.ipynb

EDA with Data Visualization

During the EDA phase, the following charts were plotted to analyze and gain insights into the SpaceX launch data:

1. **Scatter Plots:** Flight Number vs. Launch Site: Observe how launches are distributed across different s Payload Mass vs. Launch Site: Analyze the payload capacity handled by each site.

2. **Bar Charts:** Success Rate vs. Orbit Type: Compare the success rate of launches based on orbit type.

3. **Line Charts:** Launch Success Yearly Trend: Track changes in success rates over time.

4. **Pie Charts:** Proportion of Successful vs. Failed Launches: Provide an overview of overall mission succ rates.

5. **Interactive Maps (Folium):** Launch Site Locations: Visualize launch site distributions and Launch Outco with Color Coding: Highlight successful and failed launches.

   https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex4_Exploration_Data_Analysis.ipynb

EDA with SQL

- **Basic Queries:** Retrieve all records from the launch dataset, Select specific columns such as launch_site, payload_mass_kg, and landing_outcome.
- **Filtering and Conditional Queries:** Find launches that occurred at a specific launch site, Retrieve records where the payload mass exceeds a certain threshold, Filter launches based on success or failure outcomes.
- **Aggregation Queries:** Calculate the total number of launches per launch site, Determine the average payload mass for successful missions, Count the number of successful and failed landings.
- **Grouping Queries:** Group launches by orbit type and calculate success rates, Find the number of launches per year, Group by booster version to analyze performance trends.

EDA with SQL

- **Sorting and Ranking Queries:** Rank landing outcomes by frequency in descending order, Sort payloads by mass to identify the heaviest payloads launched.
- **Join Queries:** Combine data from multiple tables to analyze launch success by booster type, Merge payload data with mission outcomes to identify correlations.
- **Date-Based Queries:** Identify the first and last launch dates for a specific booster version, Count launches per year to analyze growth trends.
- **Specific Condition Queries:** Find the total payload mass carried by NASA missions, Retrieve booster names that had successful drone ship landings with payloads between specific weight ranges.

https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex6_sql.ipynb

Interactive Map with Folium

- Markers: Represent the exact locations of SpaceX launch sites. Purpose: To visually identify wher launches took place and allow for easy comparison of site locations.
- Circles: Indicate areas of interest, such as proximity to infrastructure. Purpose: To visualize the influence range of launch sites and analyze surrounding infrastructure.
- Lines (Polylines): Show the distance between launch sites and key infrastructure elements such a highways, coastlines, and railways. Purpose: To assess logistical accessibility and support decisio making for future launches.
- Color-Coded Markers: Differentiate successful and failed launches at each site. Purpose: To provi at-a-glance understanding of launch success rates by location.
- Pop-ups: Display detailed information about each launch site, including name, total launches, and success rates. Purpose: To offer additional context without cluttering the map visually.

  https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/space sual_analytics.ipynb

Build a Dashboard with Plotly Dash

- **Total Success Launches (Pie Chart)**: Displays the proportion of successful vs. failed launches for the selected launch site. Purpose: To provide an overview of the success rate for all launch sites or a specific site, helping stakeholders understand the reliability of each site.
- **Payload vs. Outcome (Scatter Plot)**: Shows the relationship between payload mass and launch success across different booster versions. Purpose: To analyze how payload weight affects launch success and identify patterns across booster versions.

https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex.py

Predictive Analysis (Classification)

Key Steps in Model Development

Data Preparation:

Cleaned and preprocessed data by handling missing values and encoding categorical variables.

Scaled numerical features such as payload mass to ensure uniformity.

Feature Selection:

Selected key features affecting landing success, such as Payload Mass, Orbit, and Booster Version.

Eliminated irrelevant or highly correlated features to improve model accuracy.

Model Training:

Used machine learning algorithms such as Logistic Regression, Decision Trees, and Random Forest.

Split data into training and testing sets for evaluation.

Predictive Analysis (Classification)

Model Evaluation:

Assessed performance using accuracy, precision, recall, and F1-score.

Plotted confusion matrices to analyze prediction errors.

Model Improvement:

Hyperparameter tuning using GridSearchCV to optimize model performance.

Performed cross-validation to reduce overfitting and generalize better.

Best Model Selection:

Compared models based on key performance metrics.

The model with the highest classification accuracy and balanced precision-recall was chosen for deployment.

https://github.com/rubensoler/spacex/blob/1d224d8b4fc62fa8adb867a274538e373dd87789/spacex7_prediction2.ipynb

Results

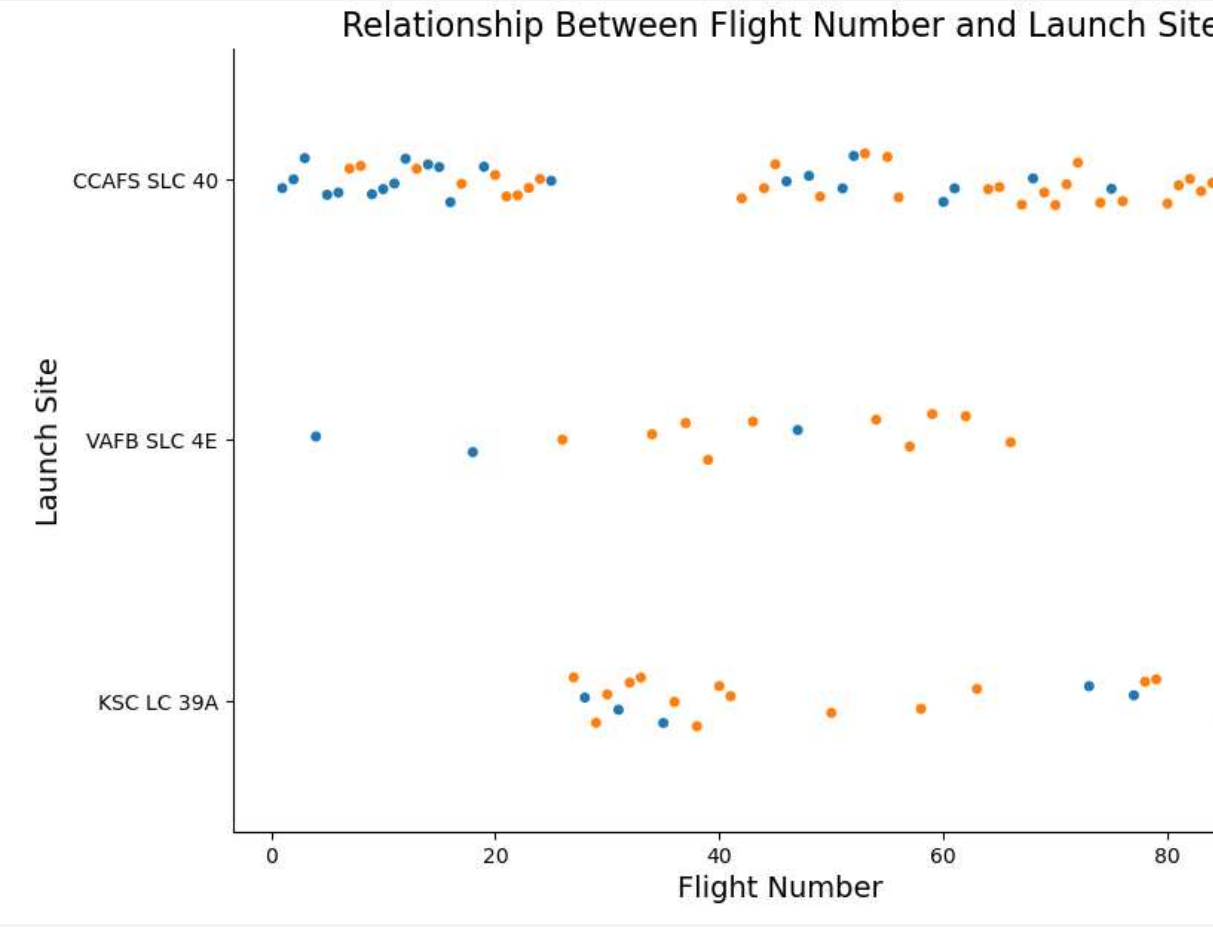
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

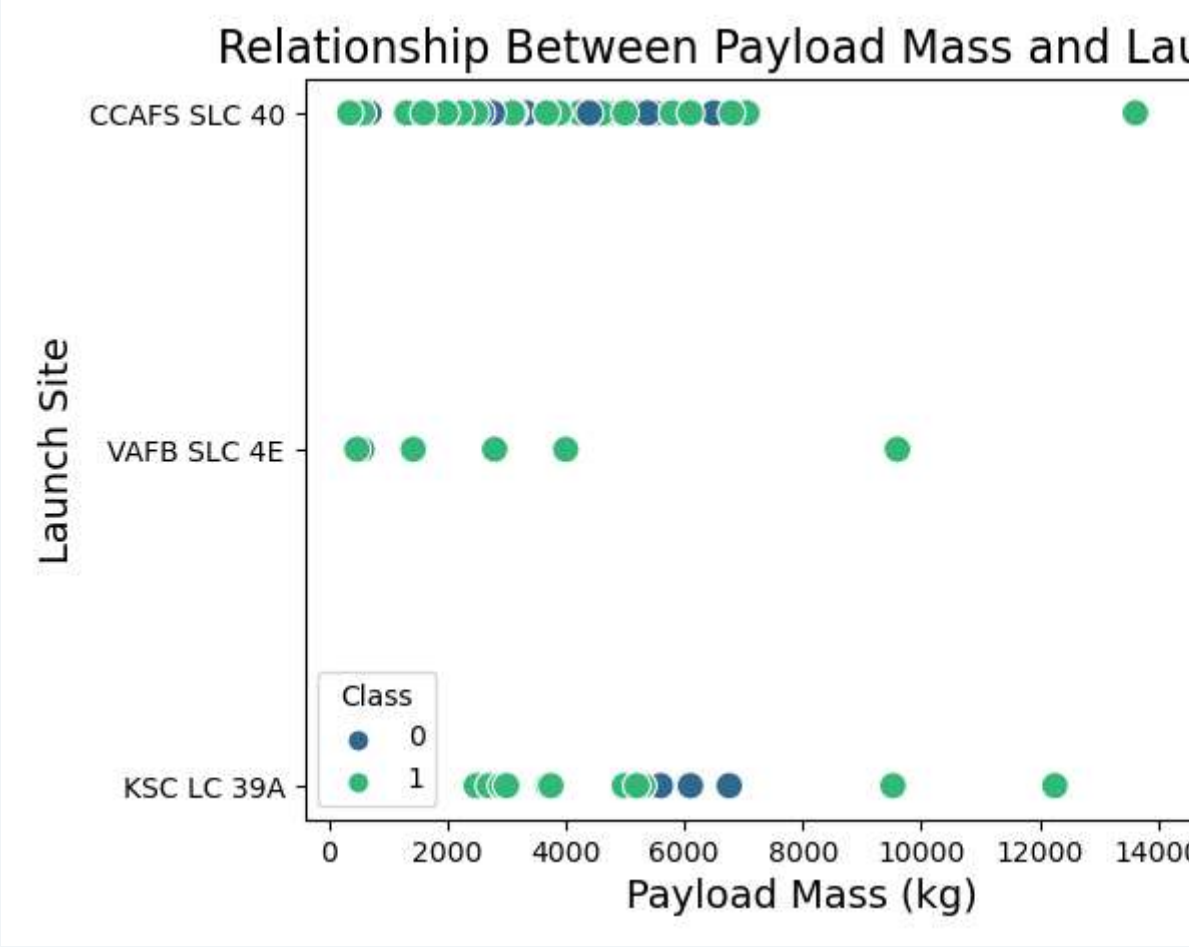
Flight Number vs. Launch Site

- Mayority of flights has used CCAFS SLC 40
- VAFB SLC 4E not in use since last 20 flights aprox.
- Succesful flights has increased significantly during last 20 flights



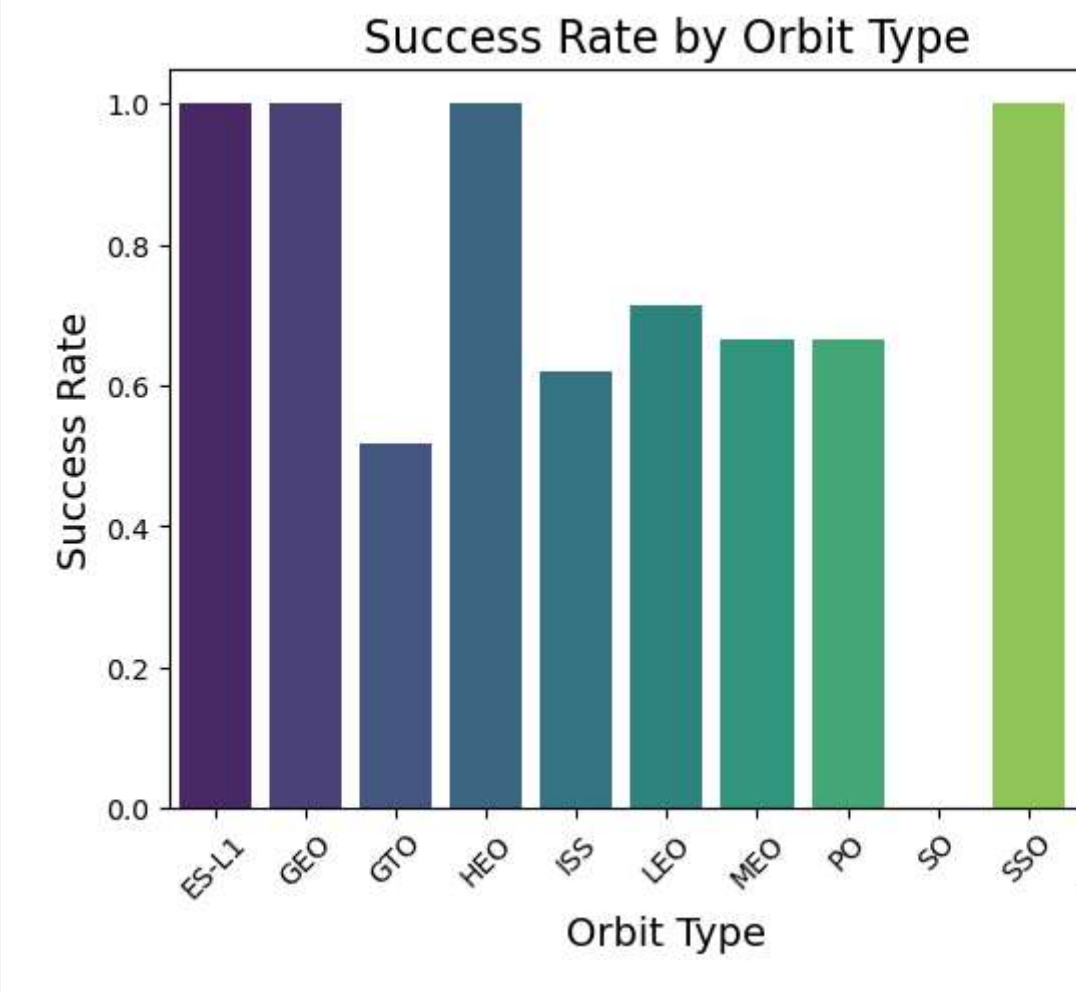
Payload vs. Launch Site

- For the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).



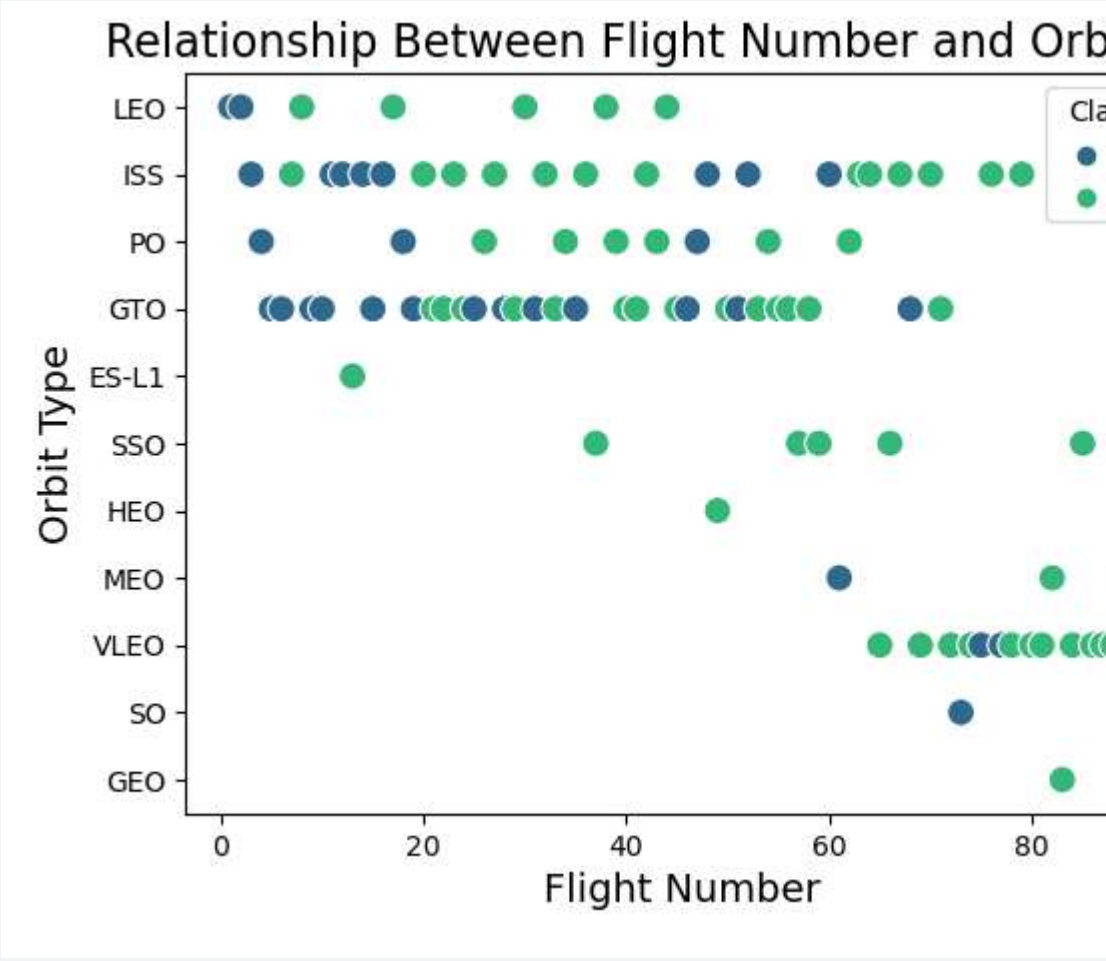
Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO Orbits has had a 100% success rate,
- The orbit with the lowest rate is GTO with 50% and SO is zero.



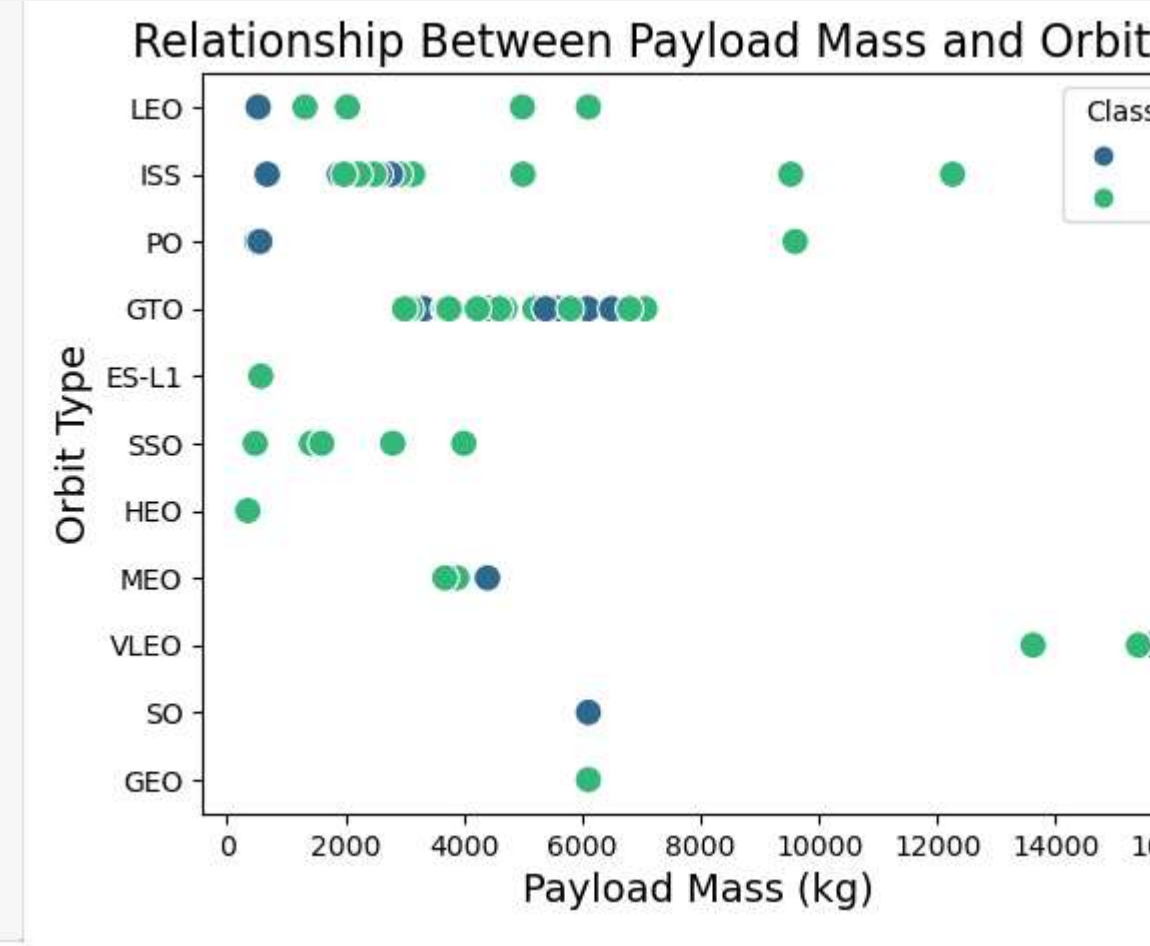
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations



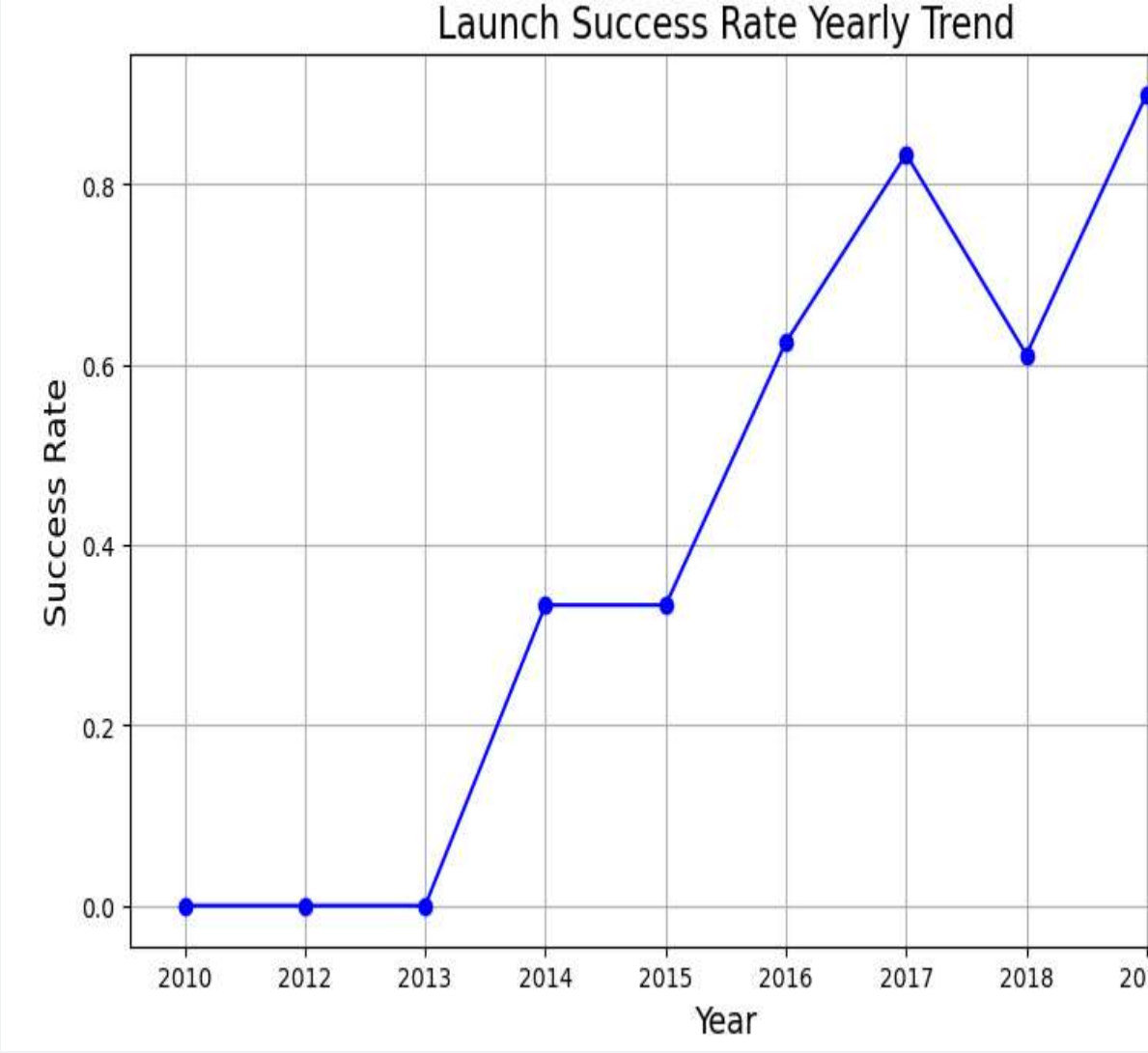
Payload vs. Orbit Type

- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.



Launch Success Yearly Trend

- Success rate for flights had a rapid improvement from 2013 to 2017
- The trend tend to stabilize around 90% of success.



All Launch Site Names

```
In [10]: %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;

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

Out[10]:

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

The word DISTINCT in this query indicates to the database to return different values for the table SPACEXTBL

Launch Site Names Begin with 'CCA'

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

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

Out[12]:

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

The oparator LIKE indicates to the DB to return values that start with the string CCA. The symbol % it's a wildcard indicating that any character aft the word CCA will be accepted for the result.

Total Payload Mass

Calculate the total payload carried by boosters from NASA

Present your query result with a short explanation here

Average Payload Mass by F9 v1.1

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

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

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

Out[18]:

Average_Payload_Mass
2928.4

The SQL query uses AVG to calculate the average payload mass, AS to assign alias to the result, FROM to specify the data source, and WHERE to filter record for the Falcon 9 v1.1 booster version.

First Successful Ground Landing Date

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

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

Out[19]:

First_Successful_Landing
2015-12-22

he SQL query uses MIN to find the earliest date, AS to rename the output column, FROM to specify the data source, and WHERE to filter records for successful landings on the ground pad.

Successful Drone Ship Landing with Payload between 4000 and 6000

In [20]:
```
%%sql SELECT DISTINCT "Booster_Version"
FROM SPACEXTBL
WHERE "Landing_Outcome" = 'Success (drone ship)'
  AND "PAYLOAD_MASS__KG_" > 4000
  AND "PAYLOAD_MASS__KG_" < 6000;
```

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

Out[20]:

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

The SQL query uses SELECT DISTINCT to retrieve unique booster versions, FROM to specify the data source, and WHERE to filter records based on successful drone ship landings with payload masses between 4000 and 6000 kg.

Total Number of Successful and Failure Mission Outcom

```
In [22]: %%sql SELECT "Mission_Outcome", COUNT(*) AS Total
         FROM SPACEXTBL
         GROUP BY "Mission_Outcome";

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

Out[22]:

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

The SQL query uses SELECT to retrieve mission outcomes, COUNT(*) to count the total occurrences of each outcome, AS to rename the result column, FROM to specify the data source, and GROUP BY to aggregate the data based on unique mission outcomes.

Boosters Carried Maximum Payload

The SQL query uses SELECT to retrieve booster versions, FROM to specify the data source, and WHERE to filter records where the payload mass is equal to the maximum payload mass, which is determined using a subquery with MAX.

In [23]:
```
%%sql SELECT "Booster_Version"
FROM SPACEXTBL
WHERE "PAYLOAD_MASS__KG_" = (
    SELECT MAX("PAYLOAD_MASS__KG_")
    FROM SPACEXTBL
);
```

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

Out[23]:

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

In [24]:
```
%%sql SELECT SUBSTR("Date", 6, 2) AS Month, "Landing_Outcome", "Booster_Version", "Launch
FROM SPACEXTBL
WHERE "Landing_Outcome" LIKE '%Failure (drone ship)%'
  AND SUBSTR("Date", 1, 4) = '2015';
```

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

Out[24]:

Month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

he SQL query uses SUBSTR to extract the month from the date, AS to rename it, SELECT to retrieve relevant columns, FROM to specify the data source, WHERE to filter records for failed drone ship landings, and another SUBSTR to ensure only records from the year 2015 are selected.

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

The SQL query uses SELECT to retrieve landing outcomes, COUNT(*) to count occurrences of each outcome, AS to rename the result column, FROM to specify the data source, WHERE to filter records within the given date range, GROUP BY to aggregate the data by landing outcome, and ORDER BY to sort the results in descending order of outcome count.

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

In [26]:
```
%%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.
```

Out[26]:

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 sites in USA

Key elements visible on the map:

Launch Sites in Florida (East Coast):

CCAFS (Cape Canaveral Space Force Station): Indicated on the Florida coast.

LC-40 and LC-39A: Marked as key launch complexes in Cape Canaveral, which are frequently used for SpaceX and other space missions.

Launch Sites in California (West Coast):

VAFB (Vandenberg Space Force Base): Located near Los Angeles, used for polar and sun-synchronous orbit launches.

SLC (Space Launch Complex) 4E: A specific launch pad at Vandenberg, often used by SpaceX for Falcon 9 missions.

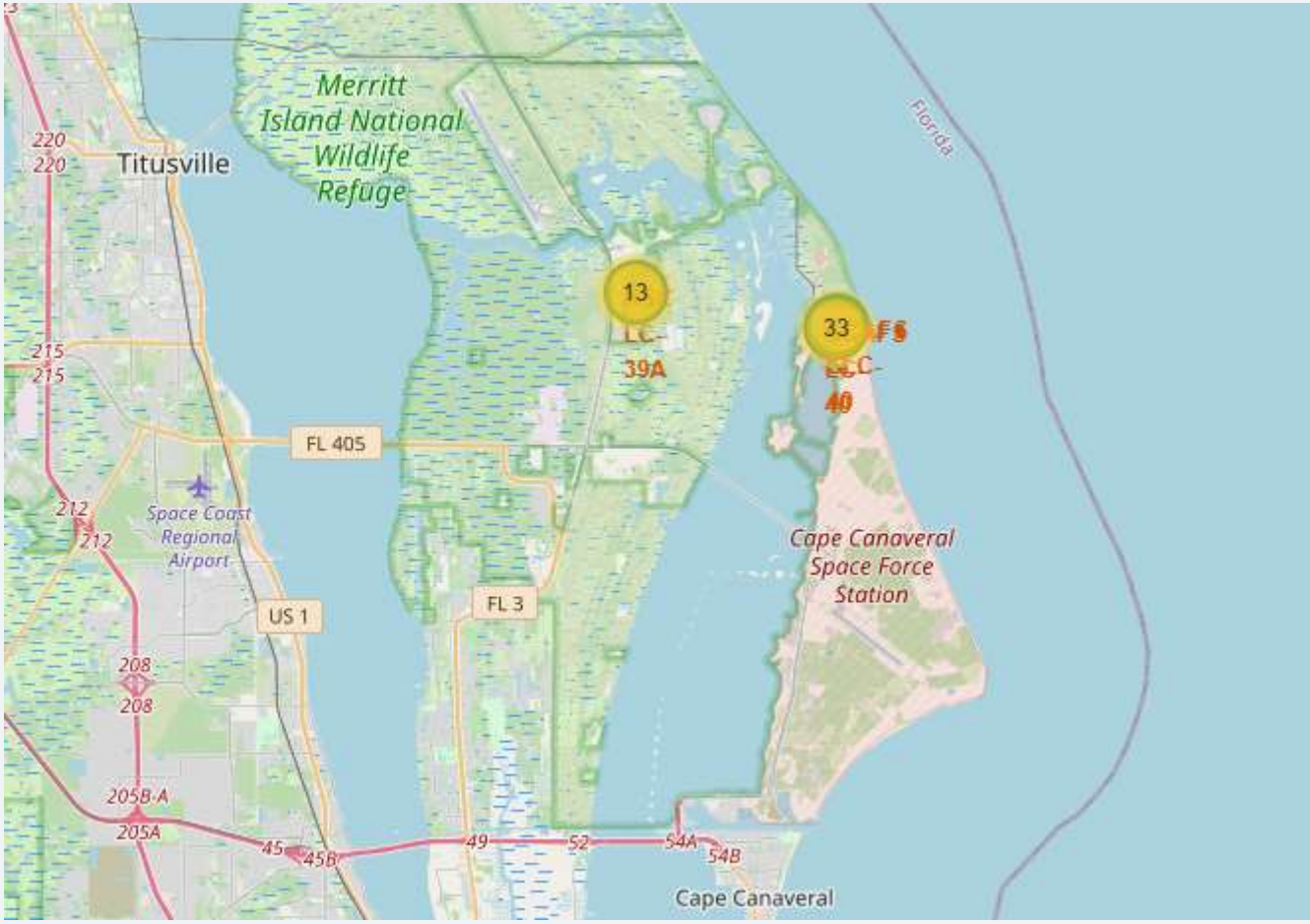
Cape Canaveral – Launches locations

The map shows the Cape Canaveral region in Florida, highlighting key launch sites and nearby landmarks. Some notable elements in the image include:

Launch Complexes (LC) Identified:

LC-39A (13 launches): Located in the northern section of the map, within the Kennedy Space Center (KSC).

LC-40 (33 launches): Located further south in the Cape Canaveral Space Force Station (CCSFS).



Launch site and its proximities

Important Elements in the Screenshot:

LC-39A Marker:The launch site is marked with an orange circle labeled "13 LC-39A," indicating 13 launches from this complex.A green boundary around the marker suggests an area of interest or safety perimeter.

Transportation Proximities:

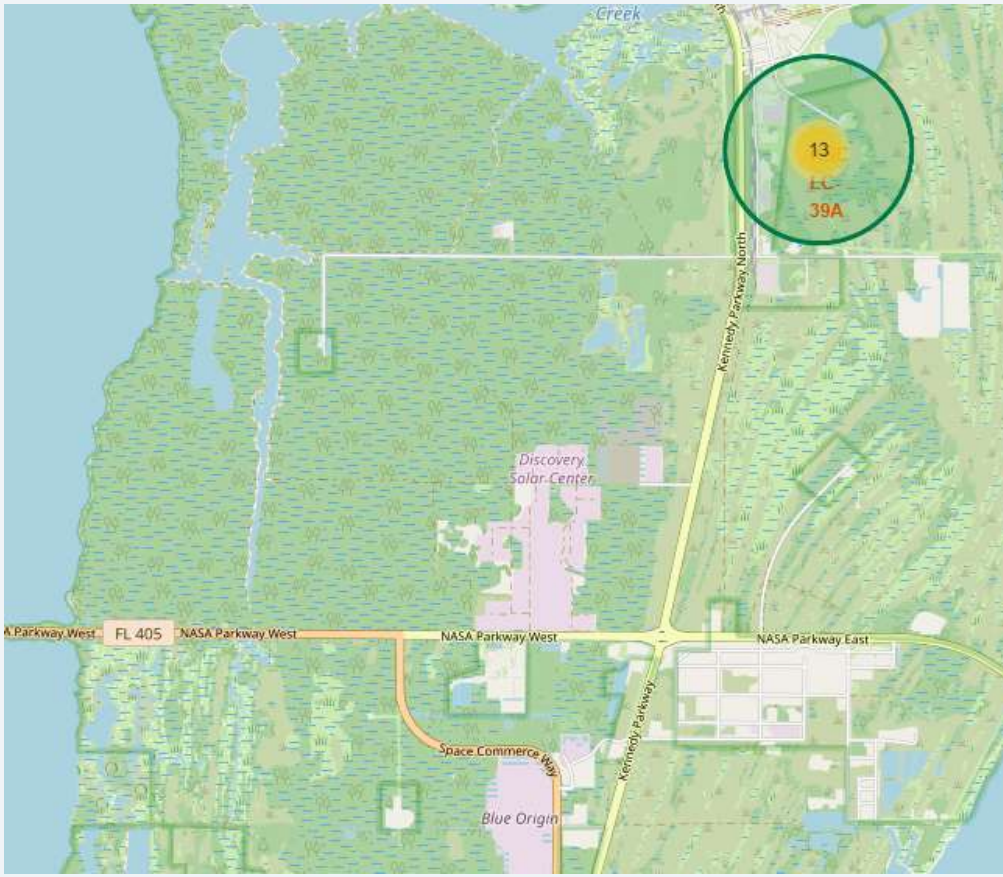
NASA Parkway (East & West):This major roadway provides access to the Kennedy Space Center facilities. It connects to FL 405, a key highway running westward toward Titusville and other mainland locations.

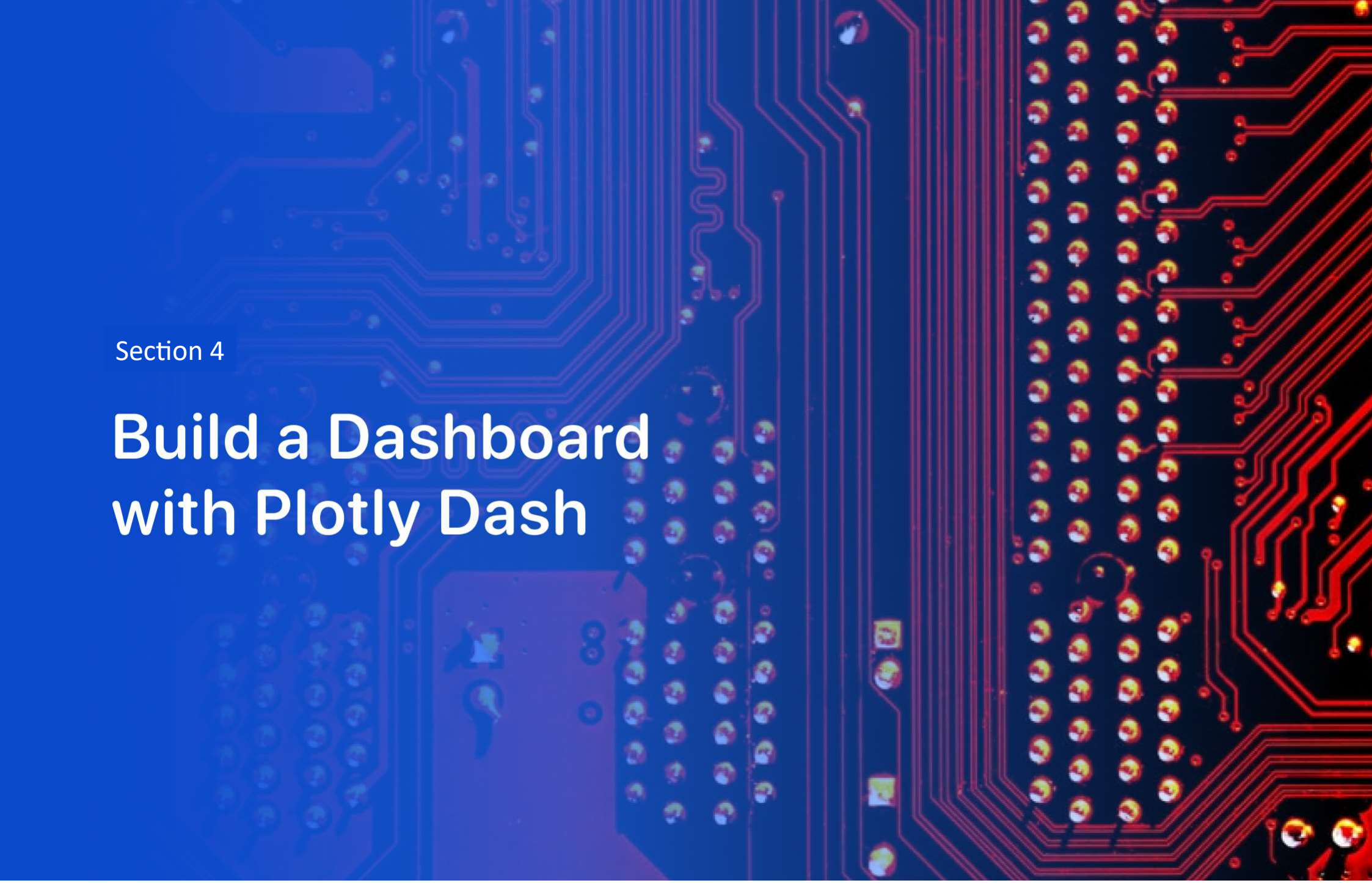
Railway (Kennedy Parkway North):A visible railway line runs parallel to the parkway, likely used for transporting heavy equipment and materials to the launch site.

Coastline Proximity:The map shows the site is relatively close to the Atlantic Ocean, which provides a natural launch corridor for missions requiring eastward trajectories.

Nearby Landmarks:Discovery Solar Center: A facility dedicated to solar energy research.

Blue Origin Facility: A private aerospace company with operations adjacent to NASA's complex, involved in space vehicle development.

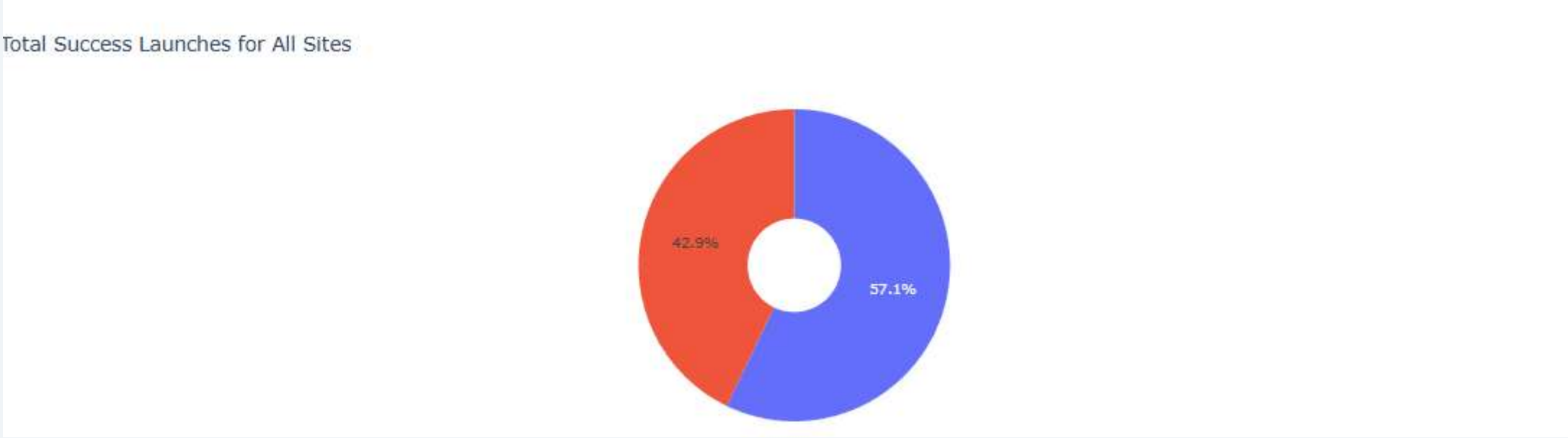




Section 4

Build a Dashboard with Plotly Dash

Total Success Launches for All Sites,



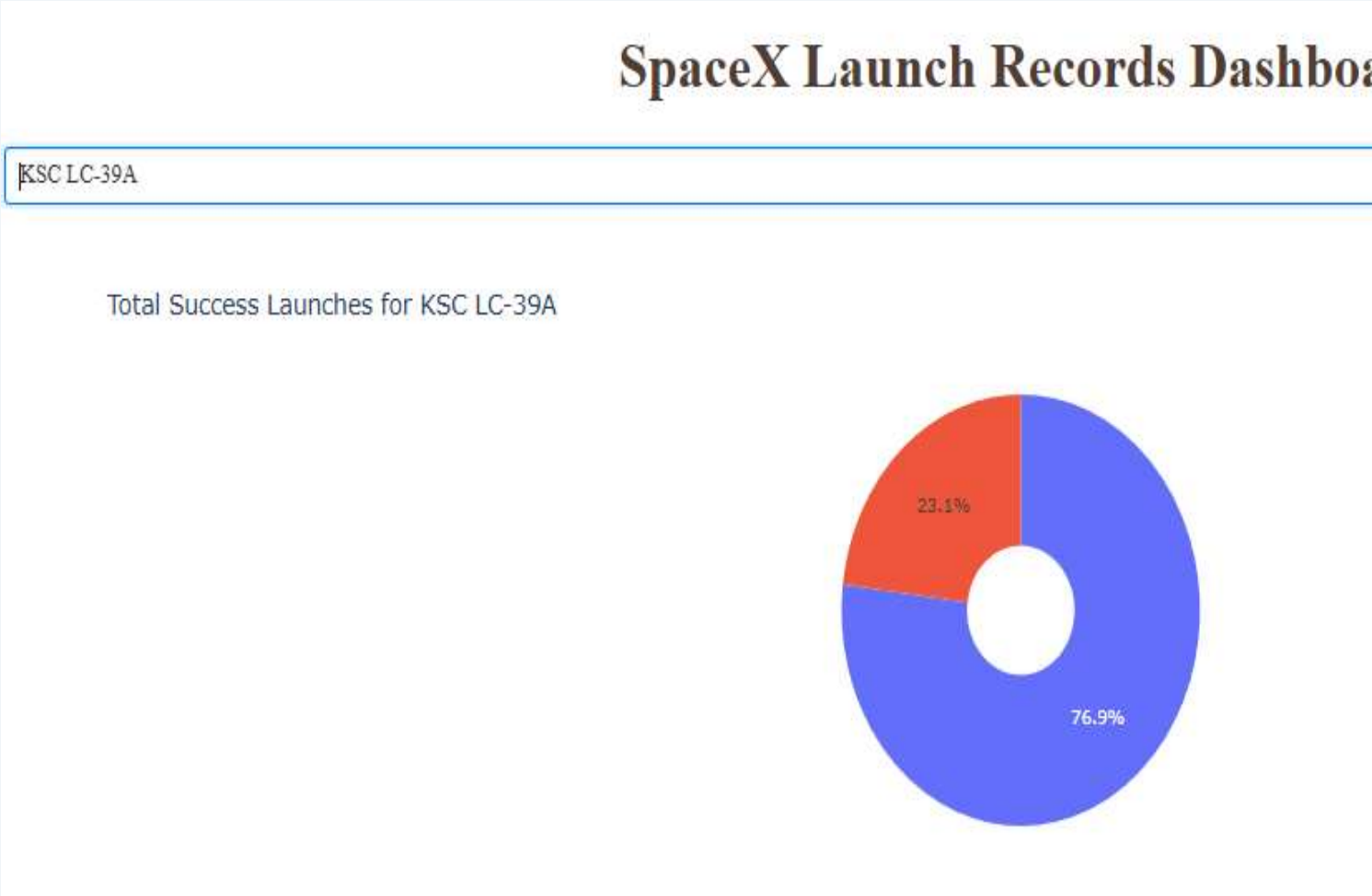
The chart represents the proportion of successful vs. unsuccessful launches acros
sites.

The blue segment (labeled 0) accounts for 57.1%, representing successful launch

The red segment (labeled 1) accounts for 42.9%, representing unsuccessful launc

Site with highest succesful launches

The site with higher success launches is KSC LC-39A



Payload vs. Outcome for All Sites

Most successful launches (y = 1) occur across the entire payload range, indicating a reliable performance for a variety of payload masses.

- The highest payload mass (~9000 kg) has a successful outcome, indicating the booster handled heavy loads effectively.

Failure Trends: Failures are observed in the lower payload range (< 4000 kg), suggesting potential reliability issues with lighter payloads or earlier booster versions.

Booster Performance: Different booster versions are tested across various payload capacities, with newer versions appearing to achieve higher success rates.

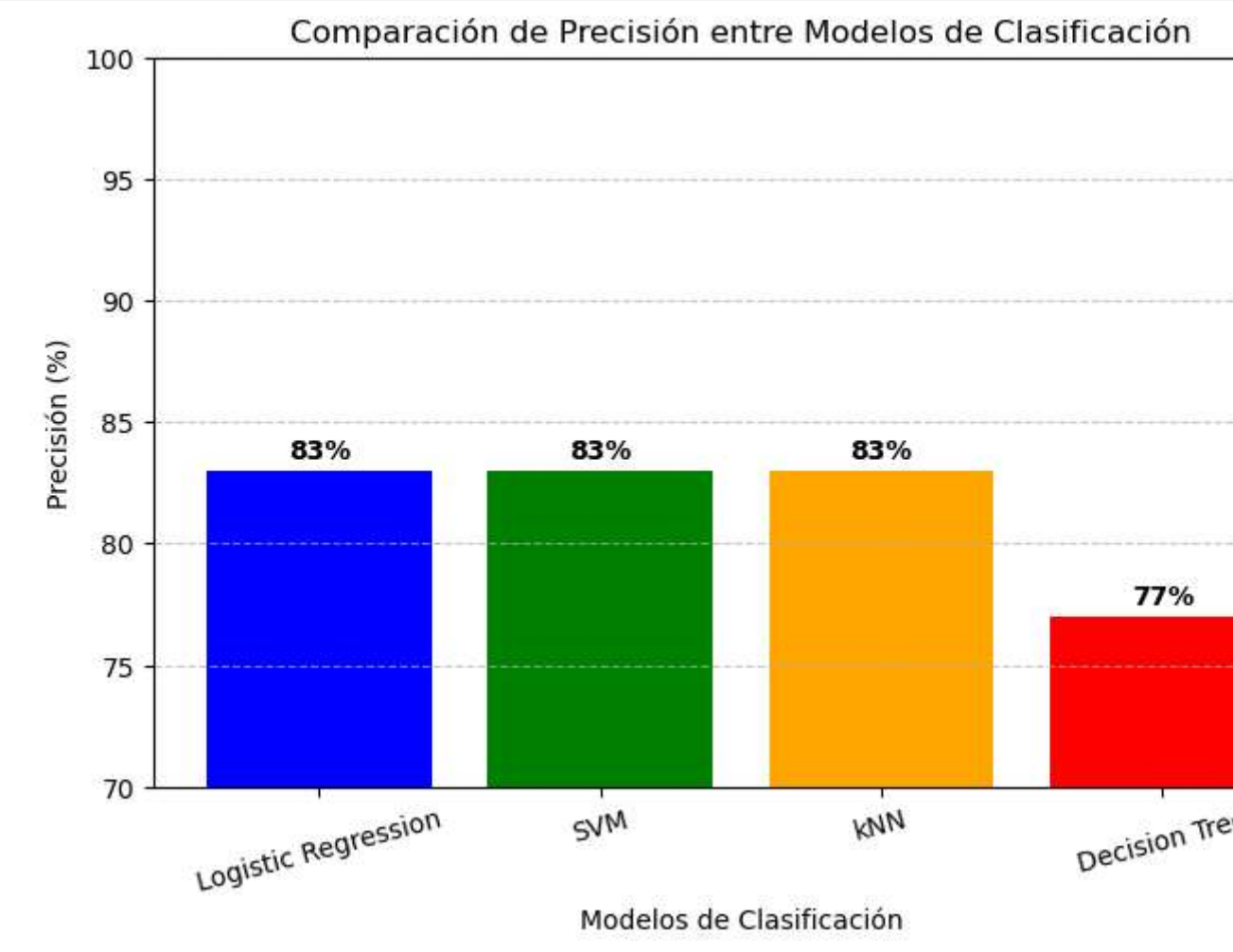
Payload Range Slider Utility: The ability to adjust the payload range provides a dynamic way to analyze trends and identify potential weight-related success patterns

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Logistic regression, SVM and KNN has de highest classification accuracy.



Confusion Matrix

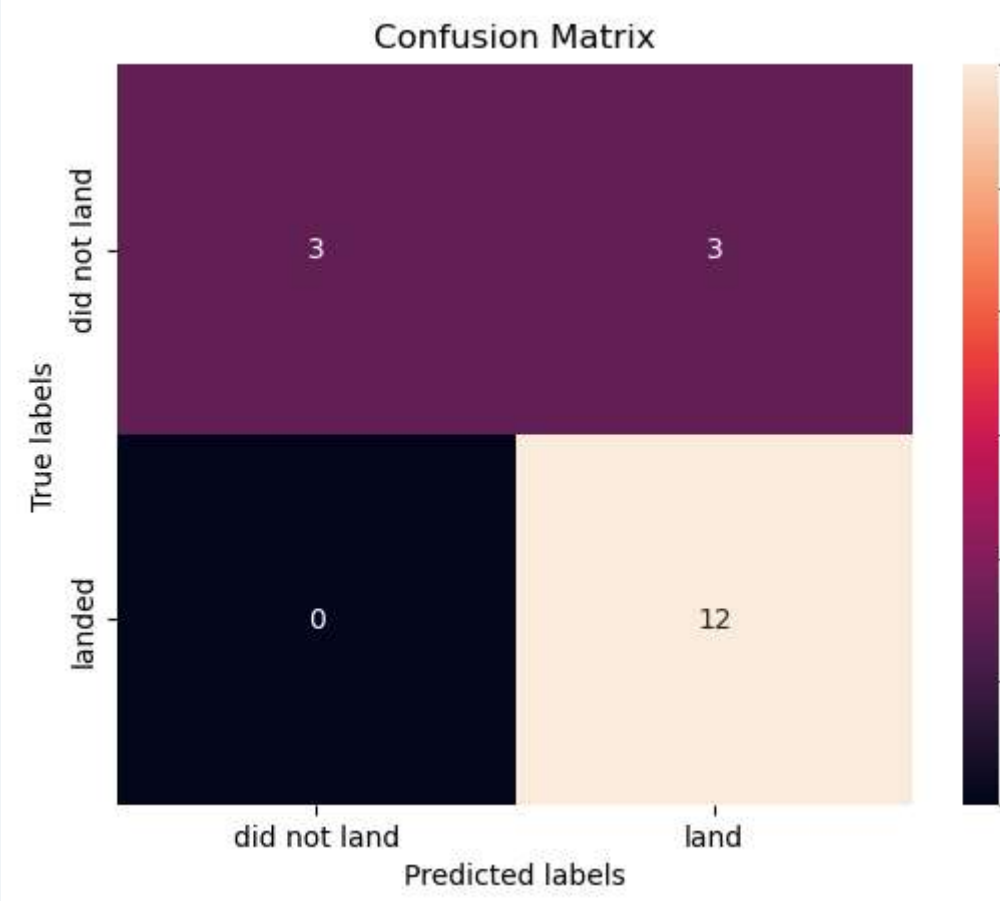
- The confusion matrix presents the model's performance in predicting whether a rocket "landed" or "did not land." Here's the breakdown of the values:

  True Positives (TP): The model correctly preddited "landed" 12 times.

- True Negatives (TN): The model correctly predicted "did not land" 3 times.

- False Positives (FP): The model incorrectly predicted "land" for 3 cases where the actual label was "did not land." This indicates that the model is over-predicting successful landings, which may lead to overconfidence.

- False Negatives (FN): There are no cases where the model predicted "did not land" for rockets that actually landed. This means the model is not missing any successful landings.



Conclusions

The model performs well in predicting actual landings (high recall).

No false negatives indicate a strong ability to identify successful landings.

The model has some false positives, which means it may incorrectly classify some non-landings as successful.

The best accuracy of the models is 83%. It `s not very high, probably due to lack of data.

...

Thank you!