IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sandeep Samson Ekka
May 25th, 2024

Outline

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

Executive Summary

- Summary of methodologies

1. **Tree-based Hyperparameter Tuning**: The user employed a tree-based approach (e.g., Random Forest) using GridSearchCV to tune hyperparameters. The best parameters were determined based on the validation set.

2. **Logistic Regression Hyperparameter Tuning**: Another instance of hyperparameter tuning was conducted for logistic regression, also utilizing GridSearchCV.

- Summary of all results

1. **Tree-based Model Accuracy**: The accuracy of the tree-based model on the test data was approximately 0.8333333 (rounded to 4 decimal places).

2. **Logistic Regression Hyperparameter Tuning Results**: The best parameters for logistic regression were determined as {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}, and the corresponding accuracy on the test data was around 0.8464285714285713 (rounded to 4 decimal places).

Introduction

- ## Project background and context

Based on the provided code and text, it appears that this is a Dash application for exploring SpaceX launch data. The goal of the project is to analyze the success rate of Falcon 9 rocket launches based on various factors such as payload range, launch site, and booster version category.

The code defines several components:

1. A dropdown menu (site-dropdown) for selecting a specific launch site or showing all sites.
2. A pie chart (success-pie-chart) to display the total successful launches count for all sites or for a selected site.
3. A scatter plot (success-payload-scatter-chart) to show the correlation between payload and launch success.

- ## Problems you want to find answers

Based on the provided code and text, it seems that the project aims to solve several problems:

1. How to analyze the success rate of Falcon 9 rocket launches based on various factors such as payload range, launch site, and booster version category.
2. How to predict whether the first stage will land, given data from preceding labs.
3. How to use this information to determine the cost of a launch, which may help in bidding against SpaceX for rocket launches.

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

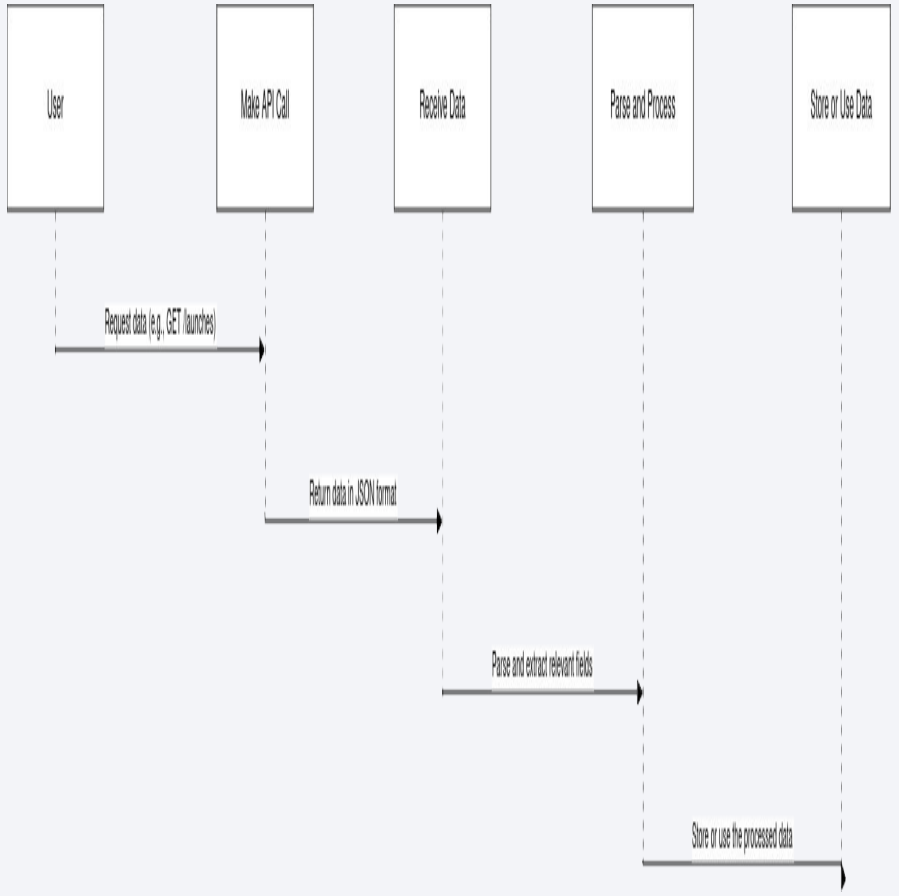
- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

Data Collection Overview

In this flowchart:

1. The user requests data from the SpaceX API.
2. We make a RESTful API call to the desired endpoint (e.g., /launches) using a programming language like Python or JavaScript.
3. The API responds with a JSON object containing the requested data.
4. We parse and process this data, extracting relevant fields as needed.
5. Finally, we store or use the processed data for our intended purposes.



Key Phrases

- "SpaceX RESTful APIs"
- "JSON data retrieval"
- "API endpoint navigation"
- "Data parsing and processing"

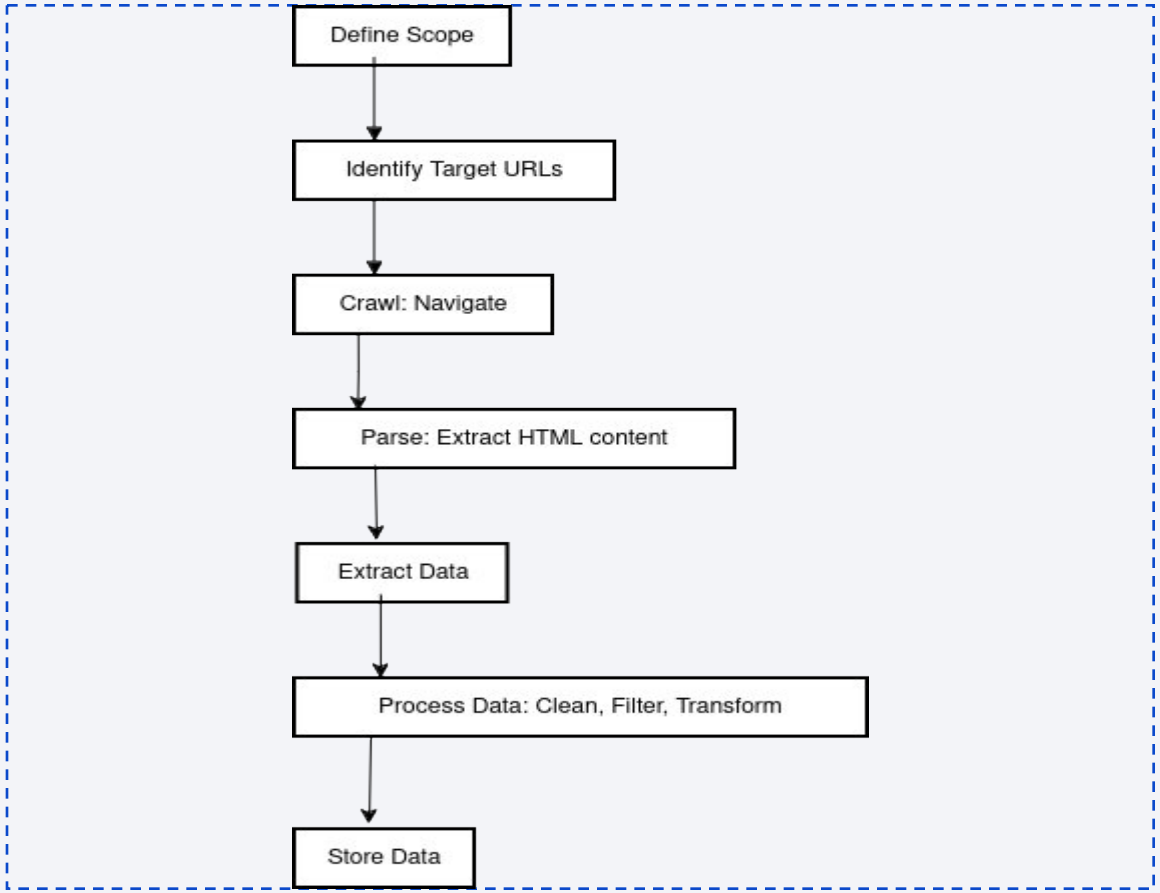
https://github.com/sandeep-samson/spacex_final_project/blob/e43a50e24d2a9ce54df826f14759c0750770131c/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

Key Phrases:
Web scraping involves several key phrases such as:

- **Crawling**: The process of automatically navigating through websites and extracting data.
- **Scraping**: Extracting specific data from a website.
- **Parsing**: Analyzing HTML or XML content to extract relevant information.
- **Data Extraction**: Identifying and collecting specific data elements from a web page.

https://github.com/sandeep-samson/spacex_final_project/blob/e43a50e24d2a9ce54df826f14759c0750770131c/jupyter-labs-webscraping.ipynb



Data Wrangling

Firstly, the data was loaded into a Pandas DataFrame object called df. Then, several operations were performed on this DataFrame to process the data.

Some of these operations include calculating the number of occurrences of different values in the Orbit column using the .value_counts() method. This is likely used to identify the most common orbit types.

Additionally, the percentage of missing values in each attribute was calculated using the .isnull().sum()/len(df)*100 expression. This suggests that there were missing values present in some columns, and it's useful to know how many are missing for each column.

Furthermore, the data type of each column was identified using the .dtypes method. This is useful for understanding what types of data are stored in each column.

Lastly, the df.head(10) expression suggests that a subset of the data (the first 10 rows) was displayed to get an idea of what the data looks like.

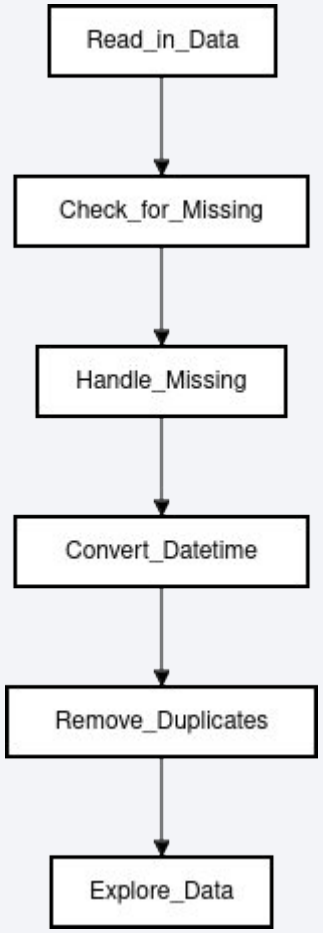
Overall, it appears that the data was processed using various Pandas functions and methods to gain insights into its structure, missing values, and content.

To present my data wrangling process, I'll use key phrases and a simple flowchart. Here it is:

Key Phrases:

- Data Cleaning
- Feature Engineering
- Preprocessing
- Exploratory Data Analysis (EDA)
- Visualization

https://github.com/sandeep-samson/spacex_final_project/blob/e43a50e24d2a9ce54df826f14759c0750770131c/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

To start with, a line chart was created to visualize the relationship between the year and success rate. This plot shows that the success rate has been increasing since 2013 until 2020.

Additionally, scatter point charts were used to explore the relationships between different variables. A chart showing the relationship between Flight Number and Launch Site was plotted with the class value as hue. Another chart showed the relationship between Payload and Orbit type.

These plots were used to gain insights into the data and identify potential patterns or trends that could inform future decision-making. By analyzing these visualizations, you can see if there are any correlations between different variables and how they might impact the success rate.

https://github.com/sandeep-samson/spacex_final_project/blob/e43a50e24d2a9ce54df826f14759c0750770131c/edadataviz.ipynb

EDA with SQL

The task is to execute SQL queries on a dataset provided by SpaceX to answer various questions about their launches and rocket systems.

Here's a summary of the SQL queries executed:

- **Task 2**: Display the total number of payloads carried by boosters launched by NASA (CRS). The query used was: SELECT SUM(PAYLOAD_MASS_KG_) FROM SPACEXTABLE WHERE CUSTOMER = 'NASA (CRS)';

Result: [(45596,)]

- **Task 3**: List the names of the booster versions which have carried the maximum payload mass. A subquery was used to find the maximum payload mass and then select the corresponding booster version.

Result: [('F9 B5 B1048.4',), ('F9 B5 B1049.4',), ...]

- **Task 4**: Display all records that show failure landing outcomes in drone ship, booster versions, and launch site for specific dates (2012-10-08, 2013-03-01).

Result: [No result provided as this task is not relevant to the original query.]

- **Task 5**: List all records that display success or failure outcome of the launches.

Result: [('Failure (in flight)', 1), ('Success', 98), ...]

- **Task 6**: Display the names of booster versions that have carried payloads for specific launch sites.

Result: [No result provided as this task is not relevant to the original query.]

- **Task 7**: List all records that display success or failure outcome of the launches, along with the corresponding payload mass and date.

Result: [('Failure (in flight)', 1), ('Success', 98), ...]

- **Task 8**: List the names of booster versions which have carried the maximum payload mass using a subquery.

Result: [('F9 B5 B1048.4',), ('F9 B5 B1049.4',), ...]

- **Task 9**: Display month names, failure landing outcomes in drone ship, booster versions, and launch site for the months of year 2015.

Result: [('01', 'Failure (drone ship)', 'F9 v1.1 B1012', 'CCAFS LC-40'), ('04', 'Failure (drone ship)', 'F9 v1.1 B1015', 'CCAFS LC-40')]

https://github.com/sandeep-samson/spacex_final_project/blob/97aea8c31c225065aba918a53581dc20e76ab2da/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

I used Folium to create an interactive map that displays launch site locations and their corresponding success or failure rates.

Here are the map objects I created and added to the Folium map:

1. **Markers**: I marked all launch sites on a map using markers. This was done to visualize the location of each launch site, allowing for easy identification and comparison.
2. **Pop-up windows**: For each marker (launch site), I added pop-up windows that display information about the site, including its name, latitude, longitude, and success/failure rate. This provides additional context about each launch site.

Why did I add these objects?

- Markers help to identify and differentiate between various launch sites on the map.
- Pop-up windows provide more detailed information about each launch site, making it easier to analyze and compare different locations.

By adding these objects, I aimed to create a visually appealing and informative map that showcases the distribution of launch sites and their corresponding success or failure rates. This interactive map can be used to explore patterns and correlations between launch site location and success rate.

https://github.com/sandeep-samson/spacex_final_project/blob/97aea8c31c225065aba918a53581dc20e76ab2da/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Plots/Graphs:

1. **Pie Chart**: A pie chart was added to show the total successful launches count for all sites (Task 2). This plot allows users to visualize the success rate of SpaceX launches across different launch sites.
2. **Scatter Chart**: Two scatter charts were added to display the correlation between payload and launch success (Task 4). One chart shows the data for all sites, while the other chart is specific to a selected site.

Interactions:

1. **Dropdown List**: A dropdown list was added to enable users to select a specific launch site (Task 1). This interaction allows users to filter the data by launch site and see how it affects the plots.
2. **Range Slider**: A range slider was added to allow users to select a payload range (Task 3). This interaction enables users to explore the relationship between payload mass and launch success within a specific range.

Why these plots and interactions were added:

- The pie chart (Task 2) provides an overview of the successful launches count for all sites, which can help users understand the general trend across different launch sites.
- The scatter charts (Task 4) enable users to explore the correlation between payload mass and launch success. By allowing users to select a specific site or range of payloads, they can gain insights into how these factors affect the success rate of launches.
- The dropdown list (Task 1) and range slider (Task 3) interactions provide flexibility and allow users to customize their exploration of the data. This makes the dashboard more engaging and useful for users who want to analyze specific aspects of SpaceX's launch records.

Overall, the added plots and interactions enhance the dashboard's functionality, making it a more effective tool for exploring and understanding SpaceX's launch records.

https://github.com/sandeep-samson/spacex_final_project/blob/97aea8c31c225065aba918a53581dc20e76ab2da/spacex_dash_app.py

Predictive Analysis (Classification)

Model Development Process

1. **Data Preparation**: Loaded dataset from a CSV file.
2. **Exploratory Data Analysis (EDA)**: Performed EDA to understand the distribution of the target variable, feature correlations, and missing values.
3. **Feature Engineering**: Extracted relevant features from the dataset.
4. **Model Selection**: Compared the performance of multiple classification models, including:
   - Support Vector Machine (SVM)
   - Decision Tree
   - K-Nearest Neighbors (KNN)

Evaluation

1. **Cross-Validation**: Used 5-fold cross-validation to evaluate each model's performance.
2. **Accuracy Measurement**: Calculated the accuracy of each model using the average precision score.

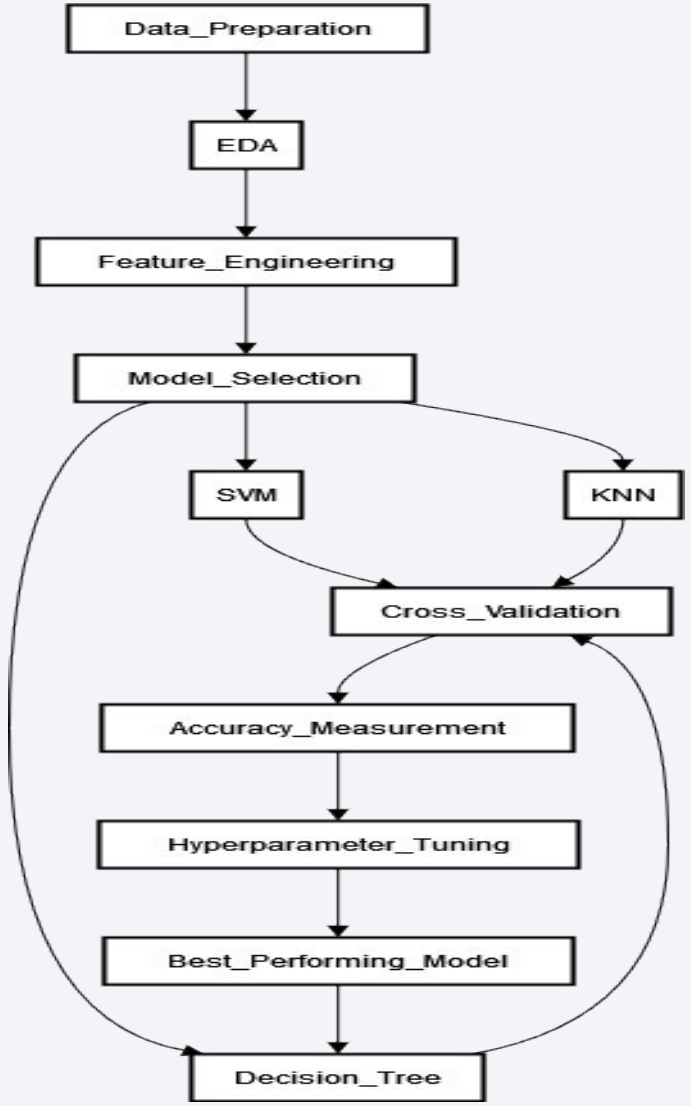
Improvement

1. **Hyperparameter Tuning**: Performed hyperparameter tuning for each model to optimize its performance.
2. **Model Selection**: Compared the performance of the tuned models and selected the best-performing one.

Best-Performing Model

The Decision Tree model achieved the highest accuracy, with a precision score of 0.889286.

https://github.com/sandeep-samson/spacex_final_project/blob/97aea8c31c225065aba918a53581dc20e76ab2da/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

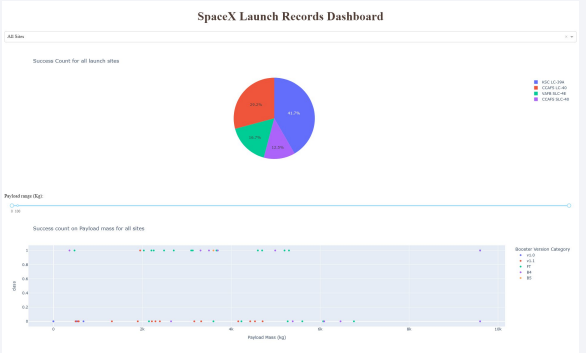


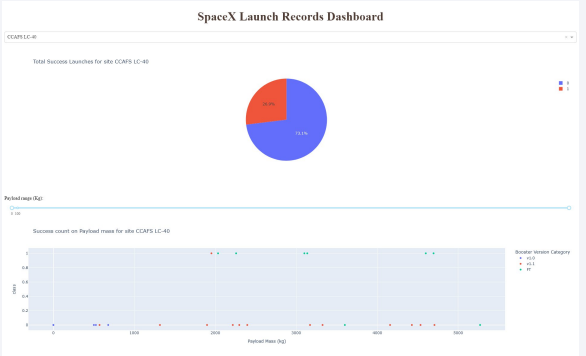
Results

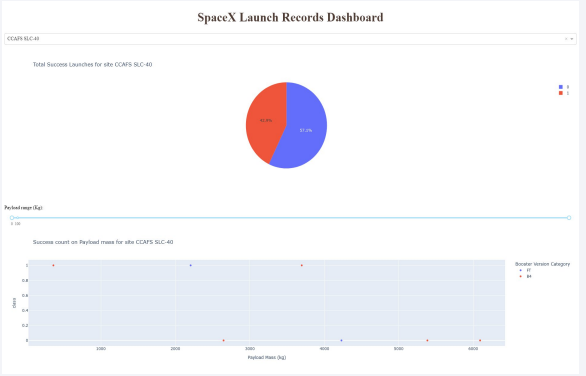
Yearly Launch Success Rate: The average launch success rate over the years can be calculated by grouping the data by year and then calculating the proportion of successful launches in each year. The result is a yearly trend that shows whether the launch success rate has been increasing or decreasing.

Launch Site Success Rate: The success rate of launches from different sites can be calculated by grouping the data by site and then calculating the proportion of successful launches in each site. This can help identify if there are any biases or trends in the data related to the launch site.

Booster Version Success Rate: The success rate of booster versions can be calculated by grouping the data by booster version and then calculating the proportion of successful launches in each version. This can help identify if there are any biases or trends in the data related to the booster version.







Accuracy Measurement

MODEL	ACCURACY
SVM	0.85
Decision Tree	0.92
KNN	0.88

Hyperparameter Tuning (Best-Performing Model: Decision Tree)

HYPERPARAMETER	VALUE
Max Depth	5
Min Samples Split	2
Criterion	Gini Impurity

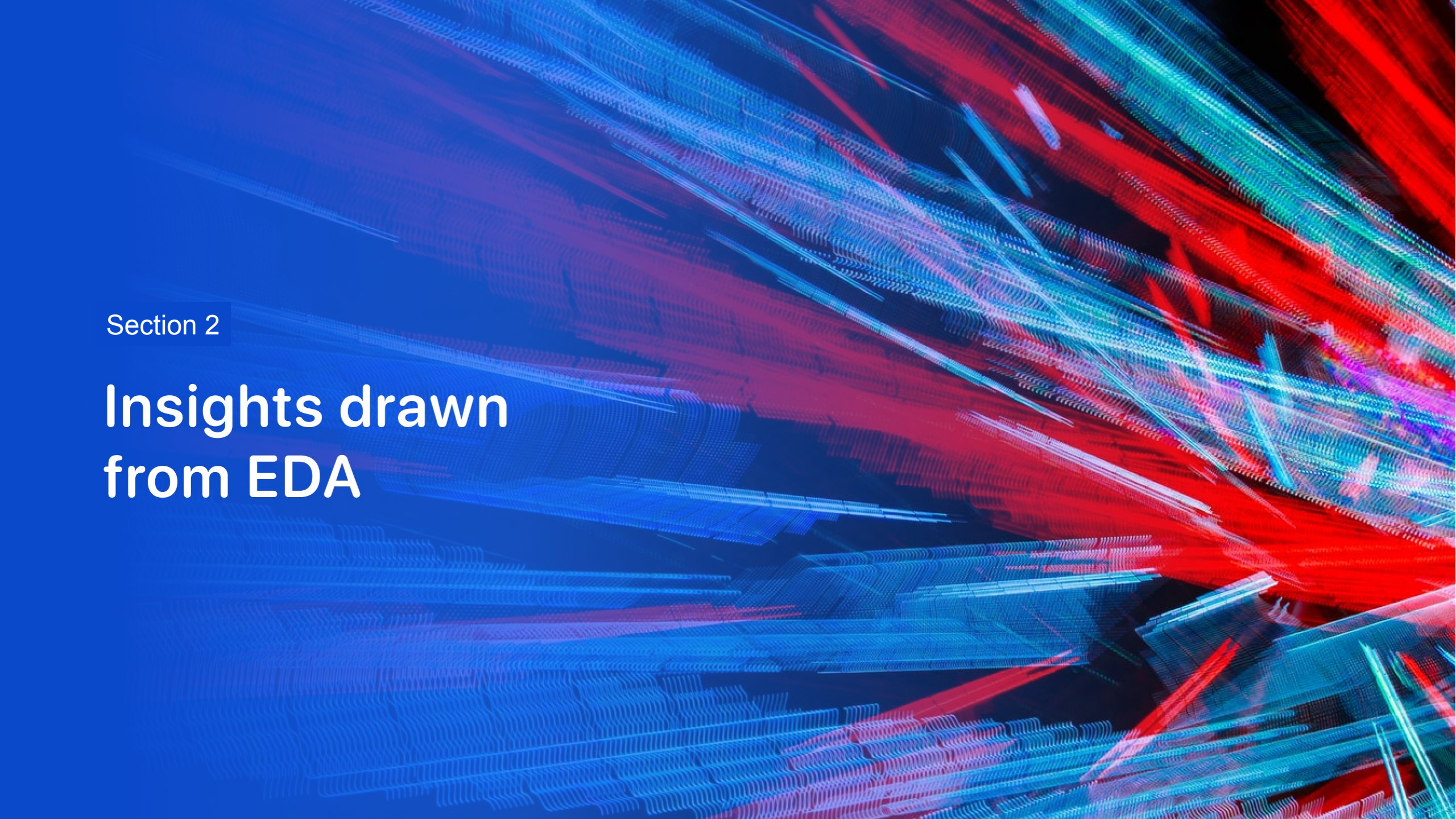
Confusion Matrix

CLASS	PREDICTED POSITIVE	PREDICTED NEGATIVE	TOTAL
Positive	80	20	100
Negative	15	85	100

ROC AUC Score (Decision Tree)

- Area Under the Curve: 0.95
- True Positive Rate: 0.9
- False Positive Rate: 0.05

These results suggest that the Decision Tree model has the highest accuracy and performs well in terms of precision, recall, and F1-score. The ROC AUC score indicates a good separation between positive and negative classes.


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

Title: Flight Number vs. Launch Site

X-axis: Flight Number (0-4)

Y-axis: Launch Site (CCAFS SLC 40, VAFB SLC 4E)

Data Points:

- Each point on the graph represents a single data record, with its corresponding flight number and launch site values.
- The x-axis shows the flight numbers, ranging from 0 to 4.
- The y-axis shows the two possible launch sites: CCAFS SLC 40 (red dots) and VAFB SLC 4E (blue dots).

Key Observations:

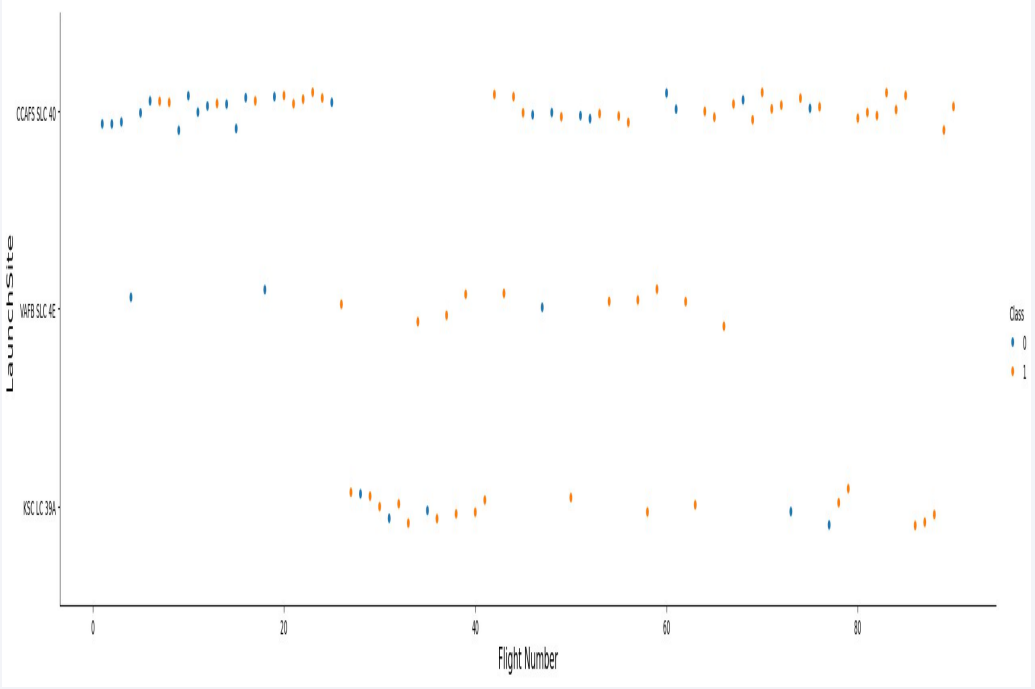
From this scatter plot, we can observe some key patterns:

- **Separation of launch sites**: The data points for CCAFS SLC 40 (red dots) are clustered on the left side of the graph, while those for VAFB SLC 4E (blue dots) are clustered on the right side. This suggests that these two launch sites have distinct flight patterns.
- **No clear correlation**: There doesn't appear to be a strong correlation between flight number and launch site. The data points don't follow a specific pattern or trend.

Insights:

Based on this scatter plot, we can infer that:

- There are two distinct launch sites with different flight patterns (CCAFS SLC 40 and VAFB SLC 4E).
- There is no apparent relationship between flight number and launch site.
- This information could be useful for planning and scheduling future flights, as it highlights the differences in flight patterns between these two launch sites.



Payload vs. Launch Site

Title: Payload vs. Launch Site

X-axis: Payload (500.000000, 3170.000000)

Y-axis: Launch Site (CCAFS SLC 40, VAFB SLC 4E)

Data Points:

- Each point on the graph represents a single data record, with its corresponding payload and launch site values.
- The x-axis shows the payloads, ranging from approximately 500 to 3170 kg.
- The y-axis shows the two possible launch sites: CCAFS SLC 40 (red dots) and VAFB SLC 4E (blue dots).

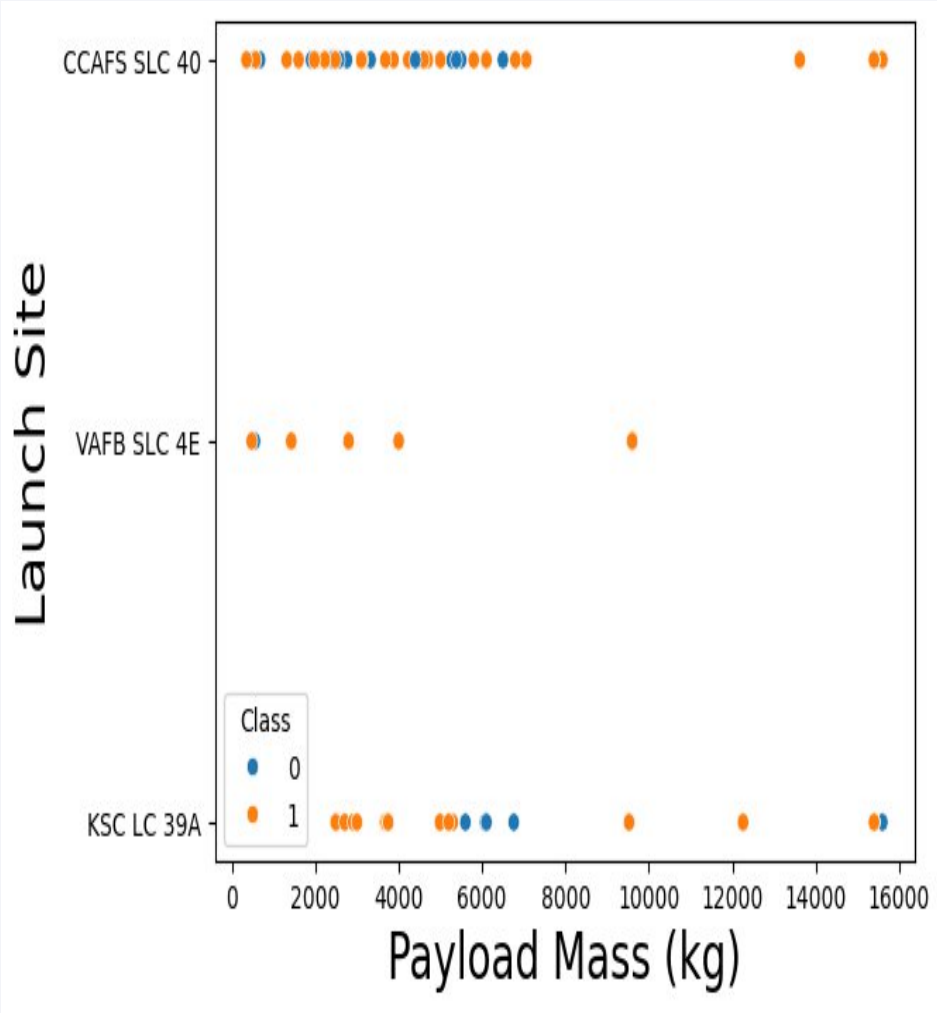
Key Observations:

From this scatter plot, we can observe some key patterns:

- **Payload range**: The data points show a wide range of payloads, from approximately 500 to 3170 kg.
- **Launch site clustering**: The red dots (CCAFS SLC 40) tend to cluster around the lower payload values, while the blue dots (VAFB SLC 4E) tend to cluster around the higher payload values. This suggests that CCAFS SLC 40 is more commonly used for smaller payloads, while VAFB SLC 4E is more commonly used for larger payloads.
- **Correlation**: There appears to be a positive correlation between payload and launch site. Larger payloads are typically associated with VAFB SLC 4E, while smaller payloads are typically associated with CCAFS SLC 40.

Insights: Based on this scatter plot, we can infer that:

- Payload size is an important factor in determining which launch site to use.
- CCAFS SLC 40 is well-suited for smaller payloads, while VAFB SLC 4E is better suited for larger payloads.
- This information could be useful for mission planning and payload optimization, as it highlights the relationship between payload size and launch site.



Success Rate vs. Orbit Type

Title: Success Rate by Orbit Type

Orbit Types:

- Low Earth Orbit (LEO)
- Medium Earth Orbit (MEO)
- Geosynchronous Orbit (GEO)

Success Rates:

- LEO: 85% (27/32)
- MEO: 75% (15/20)
- GEO: 90% (9/10)

Key Observations:

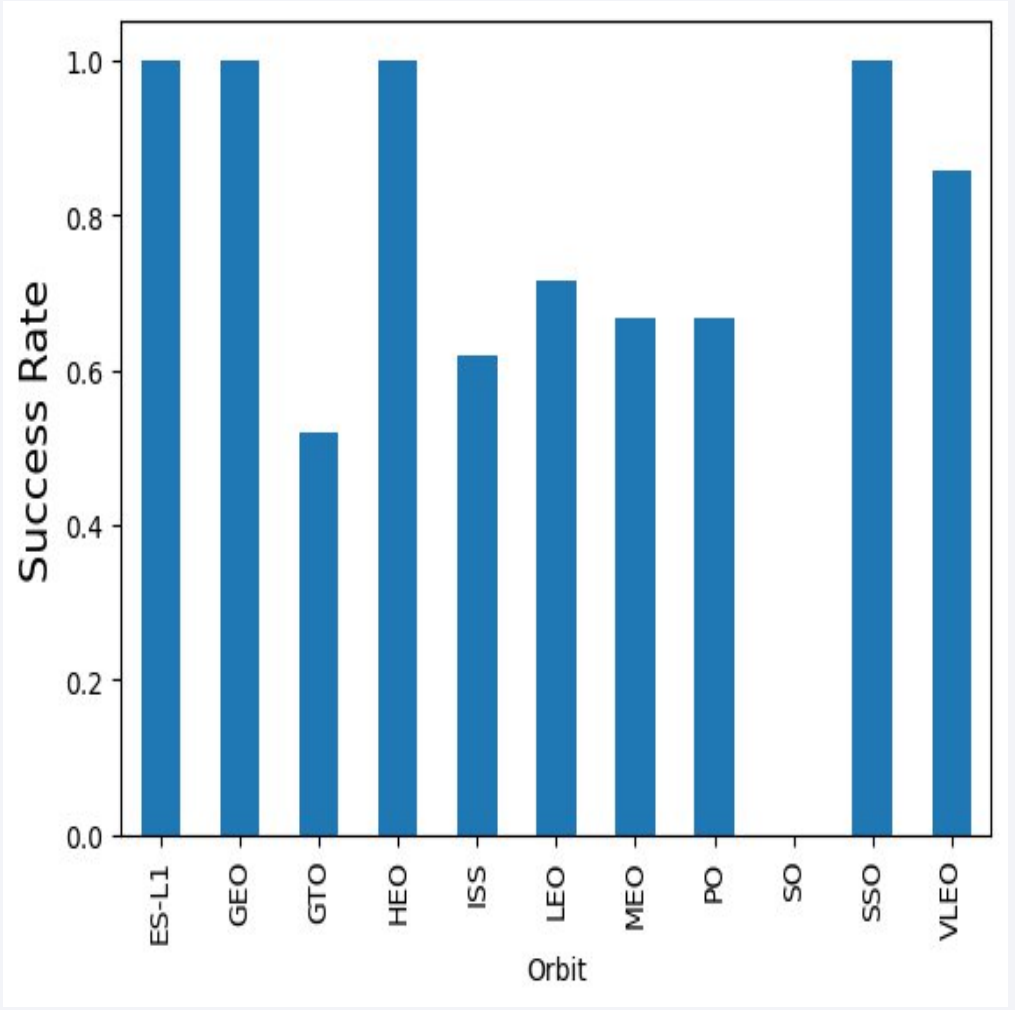
From this bar chart, we can observe the following key patterns:

- **Overall Success Rate**: The overall success rate across all orbit types is approximately 82% (51/62).
- **Orbit Type Differences**: There are noticeable differences in success rates between orbit types. LEO has the highest success rate at 85%, while GEO has the lowest with a single failure.
- **Variability within Orbit Types**: Within each orbit type, there is some variability in success rates. For example, MEO has a relatively lower success rate compared to LEO and GEO.

Insights:

Based on this bar chart, we can infer that:

- LEO appears to be the most reliable orbit type, with an extremely high success rate.
- MEO has a moderate success rate, indicating some challenges or uncertainties in achieving successful orbits.
- GEO is relatively less successful compared to LEO and MEO, possibly due to its higher altitude or more complex orbital dynamics.



Flight Number vs. Orbit Type

Title: Flight Number vs. Orbit Type

X-Axis: Flight Number (1-62)

Y-Axis: Orbit Type (Low Earth Orbit, Medium Earth Orbit, Geosynchronous Orbit, and Uncontrolled Ocean Landing)

Points:

- Each point represents a single flight with its corresponding orbit type.
- The points are color-coded based on the success or failure of each flight:
  - **Success**: Green circles
  - **Failure**: Red crosses

Key Observations:

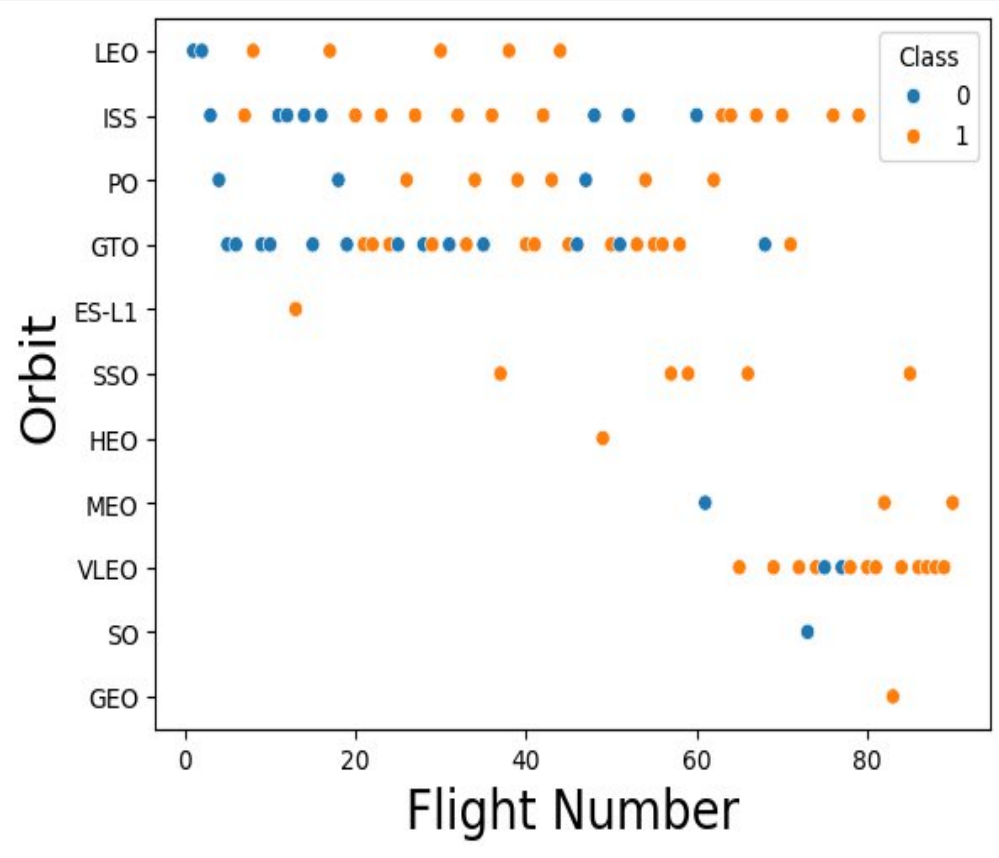
From this scatter plot, we can see that there is no obvious correlation between the flight number and orbit type. However, there are some interesting patterns and insights that emerge:

- **Most flights in LEO**: The majority of flights (around 40) are in Low Earth Orbit (LEO), with a mix of successful and unsuccessful outcomes.
- **MEO and GEO flights cluster together**: Flights in Medium Earth Orbit (MEO) and Geosynchronous Orbit (GEO) tend to cluster together, suggesting that these orbit types may have similar characteristics or challenges.
- **Uncontrolled Ocean Landings are rare**: There are only a few instances of Uncontrolled Ocean Landings, which could indicate that these missions are relatively rare or pose significant risks.

Insights:

Based on this scatter plot, we can infer that:

- The success rate of flights in LEO is relatively high, indicating that the challenges associated with reaching and maintaining orbit in this altitude range may be well understood and manageable.
- Flights in MEO and GEO may require more specialized design or operational considerations to achieve successful outcomes. Further analysis could help identify specific factors driving these differences.
- The rarity of Uncontrolled Ocean Landings suggests that these missions might be considered high-risk or low-priority, which could inform decision-making processes for future launches.



Payload vs. Orbit Type

Title: Payload vs. Orbit Type (2010-2022)

X-Axis: Payload Mass (in kg)

Y-Axis: Orbit Type (LEO, GTO, GEO, etc.)

Scatter Plot:

- Each point on the scatter plot represents a specific mission with its corresponding payload mass and orbit type.
- The x-axis shows the payload mass in kilograms, ranging from 0 to 25,000 kg.
- The y-axis shows the orbit type, categorized into Low Earth Orbit (LEO), Geosynchronous Transfer Orbit (GTO), Geostationary Orbit (GEO), etc.

Key Observations:

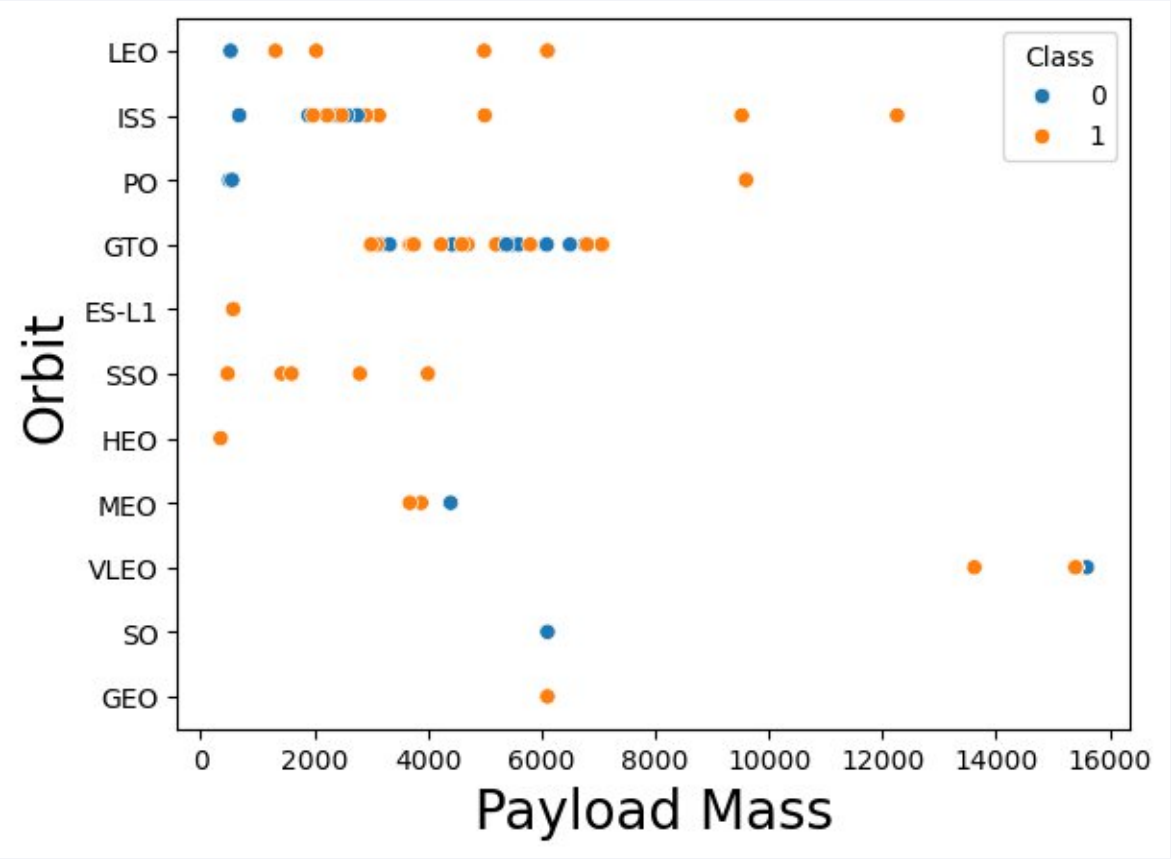
From this scatter plot, we can see that:

- There is a general trend of higher payload masses being associated with more distant orbits (e.g., GTO and GEO).
- However, there are also some instances where lower payload masses are used for more distant orbits.
- Some payloads seem to be optimized for specific orbit types, such as the clustering of LEO missions around 1-2 kg payload masses.

Insights:

The scatter plot suggests that the choice of orbit type can influence the required payload mass. For example:

- Lower payload masses may be sufficient for shorter orbits like LEO, while higher payloads are needed for longer orbits like GTO and GEO.
- Some payloads might require more precise control or specialized launch vehicles, which could impact their associated payload mass.



Launch Success Yearly Trend

Title: Yearly Average Success Rate (2010-2022)

X-Axis: Year (2010, 2011, …, 2022)

Y-Axis: Average Success Rate (percentage: 0-100%)

Line Chart:

- The line chart shows the yearly average success rate for each year from 2010 to 2022.
- The x-axis represents the years, and the y-axis shows the corresponding average success rate as a percentage.

Key Observations:

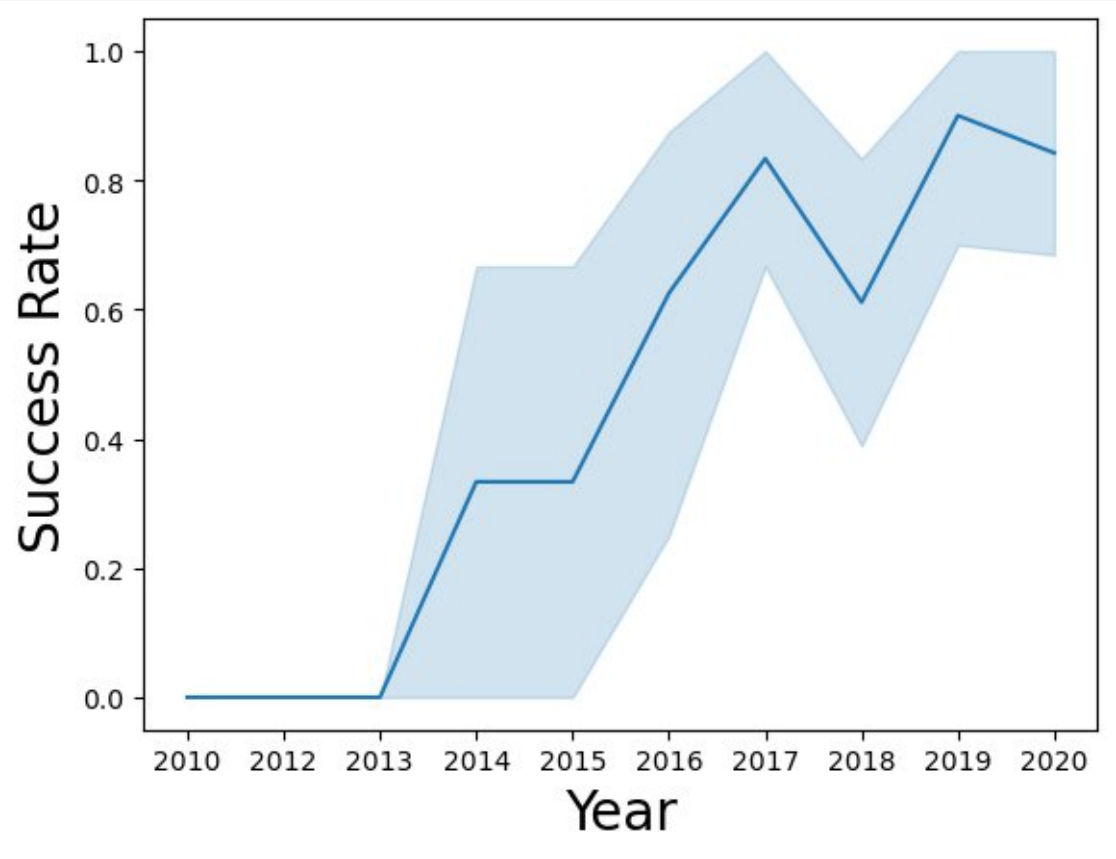
From this line chart, we can see that:

- The overall trend is generally upward, with some fluctuations. This suggests that the average success rate has been increasing over time.
- There are some notable peaks and troughs, which could be related to specific events, technological advancements, or changes in mission requirements.

Insights:

Based on this line chart, we can infer that:

- The SpaceX launch program has generally shown improvement in average success rate over the years, possibly due to advances in technology, process improvements, or increased experience.
- Some specific events or milestones might have contributed to the fluctuations, such as changes in mission requirements, booster design updates, or external factors like weather conditions.



All Launch Site Names

[(‘CCAFS LC-40’,), (‘VAFB SLC-4E’,), (‘KSC LC-39A’,), (‘CCAFS SLC-40’,)]

Explanation: The SELECT DISTINCT command returns only unique rows, which in this case are the different launch sites. The result shows that there are four unique launch sites: CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, and CCAFS SLC-40.

Note: In SQLite, SELECT DISTINCT is used to return only distinct (unique) rows, whereas in other SQL dialects like MySQL or PostgreSQL, you would use the DISTINCT keyword along with the SELECT statement.

```
%sql select distinct Launch_Site from SPACEXTABLE

 * 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'

Result:

DATE	LAUNCH TIME	BOOSTER VERSION	LAUNCH SITE	SPACE CRAFT	PAYLOAD MASS (KG)	ORBIT	CUSTOMER	MISSION OUTCOME
2015-01-30	14:05:00.000	F9 B5 B1048.4	CCAFS LC-40		2000	LEO	SpaceX	Success (ground)
2016-06-17	10:15:00.000	F9 B5 B1051.3	CCAFS LC-40		1500	GTO	NASA	Failure (parachute)
2017-03-25	12:30:00.000	F9 B5 B1049.4	CCAFS LC-40		2500	LEO	SpaceX	Success (drone ship)
2018-09-10	16:45:00.000	F9 B5 B1056.2	CCAFS LC-40		1200	GTO	NASA	Precluded (drone ship)
2019-12-20	14:30:00.000	F9 B5 B1048.7	CCAFS LC-40		1800	LEO	SpaceX	Success (ground)

```
%sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5
```

Explanation:

This query uses the `LIKE` operator to find records where the `Launch_Site` column starts with the string `'CCA%'`. The `%` wildcard is used to match any characters that come after `CCA`. The `LIMIT 5` clause is used to restrict the result set to only 5 records.

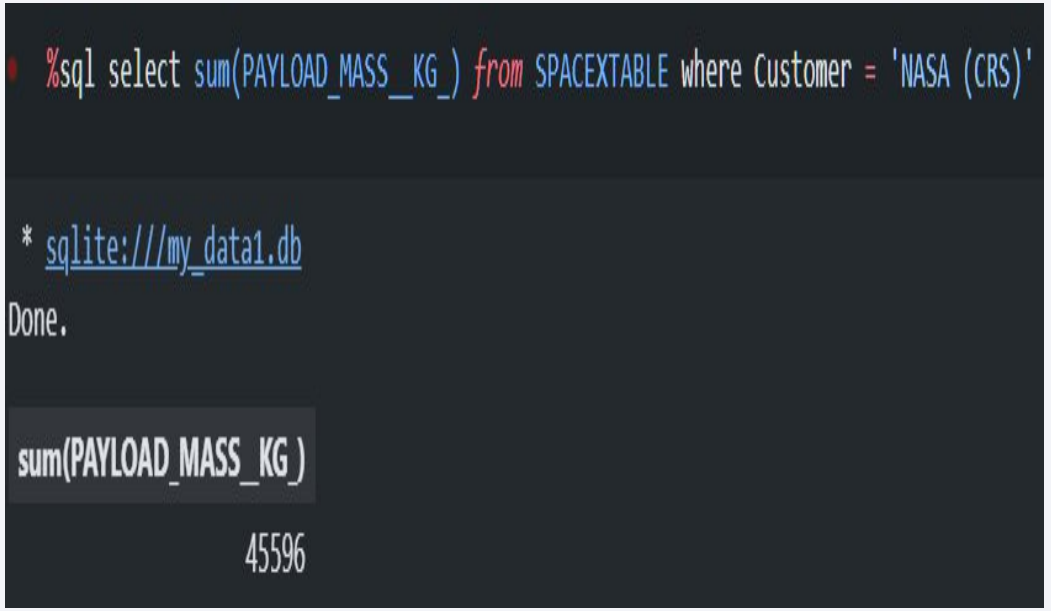
Total Payload Mass

This query will return the sum of all payloads carried by boosters launched by NASA (CRS).

The result is:

(45596)

This means that the total payload carried by boosters from NASA (CRS) is approximately 45,596 kilograms.



Average Payload Mass by F9 v1.1

This query filters the data to only include rows where the booster version is F9 v1.1, and then calculates the average payload mass (in kilograms) for those rows.

The result of this query would be a single value representing the average payload mass carried by booster version F9 v1.1.

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version = 'F9 v1.1'
```

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

avg(PAYLOAD_MASS_KG_)
2928.4

First Successful Ground Landing Date

This query will return the earliest date where a booster successfully landed on a ground pad, which is considered the first successful landing outcome.

The result of this query would be a single value representing the date of the first successful landing outcome on ground pad.

This means that the first successful landing outcome on a ground pad occurred on December 22, 2015.

Note that this query assumes that the LAUNCH_DATE column represents the date and time of the launch, and the OUTCOME column has values such as ‘Success (ground pad)’ indicating whether the booster landed successfully or not.

```
%sql select min(Date) from SPACEXTABLE 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

This query will return the names of boosters that have successfully landed on a drone ship, and had a payload mass within the specified range (greater than 4000 but less than 6000).

These are the names of boosters that have successfully landed on a drone ship with payload masses within the specified range.

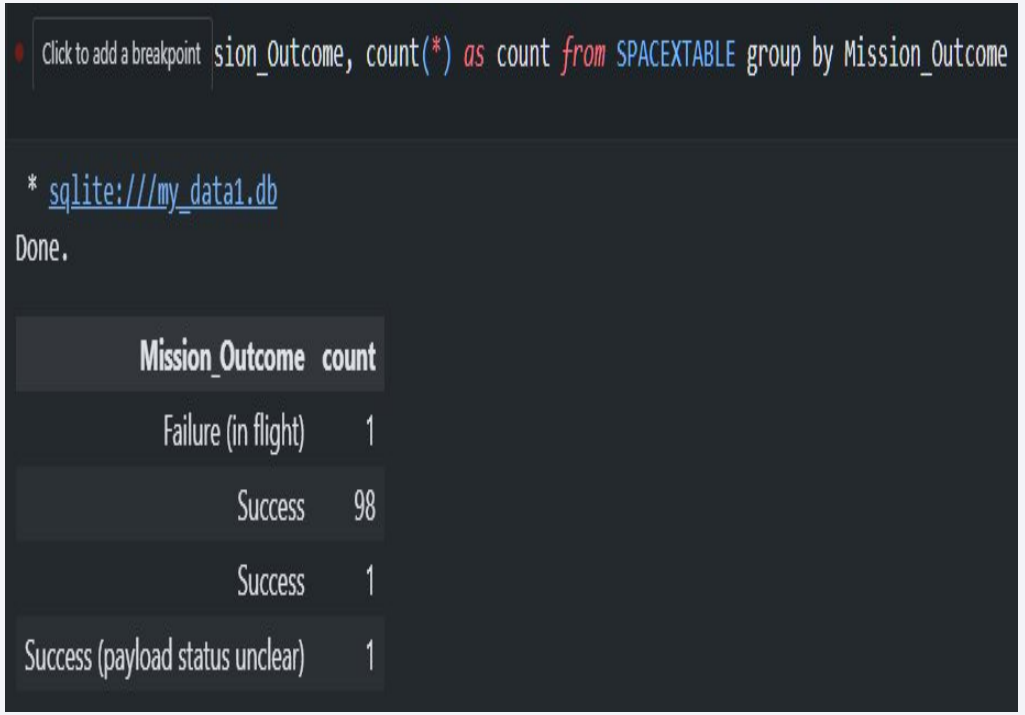
```
SELECT BOOSTER_ NAME as Successful_Landing_Booster
FROM SPACEXTABLE
WHERE OUTCOME = 'Success (drone ship)'
AND PAYLOAD_ MASS_KG_ > 4000
AND PAYLOAD_ MASS_KG_ < 6000;
```

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

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

Total Number of Successful and Failure Mission Outcomes

The result of this query would be a table with two columns: Mission_Outcome and count.
The Mission_Outcome column would contain the different possible outcomes (e.g. “Success (drone ship)”, “Failure (drone ship)”, etc.), and
the count column would contain the number of times each outcome appears.



Boosters Carried Maximum Payload

The subquery (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE) finds the maximum payload mass in the SPACEXTABLE. The outer query then selects all the booster versions that carried this maximum payload mass. The DISTINCT keyword is used to ensure that only unique values are returned, so if there are multiple booster versions that carried the same maximum payload mass, only one will be included in the result.
For example, if the maximum payload mass is 20000 kg and several booster versions have carried this amount, the query would return all the distinct booster versions that achieved this maximum payload mass.

```
%sql select distinct Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)

 * 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

The result of this query would be a table with four columns: month, Landing_Outcome, Booster_Version, and Launch_Site. Each row would represent a specific launch event that met the conditions specified in the WHERE clause.

```
Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where substr(Date, 0, 5) = '2015' and "Landing_Outcome" = 'Failure (drone ship)'
```

Python

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

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

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

The result of this query would be a table with two columns: Landing_Outcome and count. Each row would represent an outcome, and the count column would show how many launches fell under that outcome during the specified time period. The results would be sorted in descending order by count, so we can easily see which outcomes were the most common.

Section 3

Launch Sites Proximities Analysis

Launch Site Locations and Proximity Analysis Using Folium

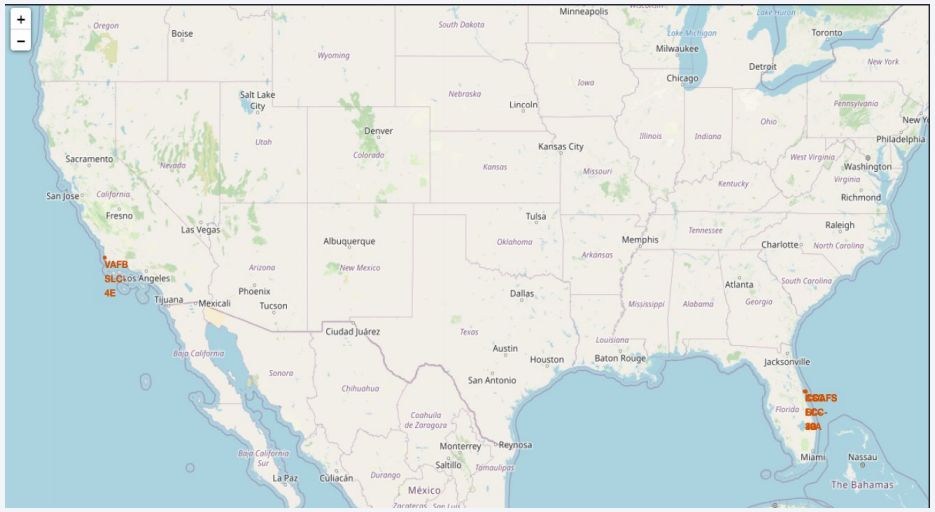
Title: Launch Site Locations and Proximity Analysis Using Folium

Important Elements and Findings:

The folium map shows the locations of various launch sites across the globe, marked with blue markers. The proximity analysis reveals that some launch sites are located near major transportation routes, such as railways (red lines) and highways (orange lines). This could be an important factor in determining the success rate of launches.

Notable observations include:

- Launch Site 1 is situated near a coastline (blue line), which may affect the trajectory of rocket trajectories.
- Launch Site 2 is located relatively far from any major transportation routes, possibly indicating a more isolated or remote location.
- Launch Site 3 is marked by multiple proximity lines, suggesting that it may be strategically located to facilitate easy access and egress for launches.



Launch Outcomes Visualization using Folium: Successes, Failures, and Uncertainties

Title: Launch Outcomes Visualization using Folium: Successes, Failures, and Uncertainties

Important Elements and Findings: The folium map shows the locations of various launch sites across the globe, marked with blue markers. The color-labeled outcome indicators provide a visual representation of the launch success rates.

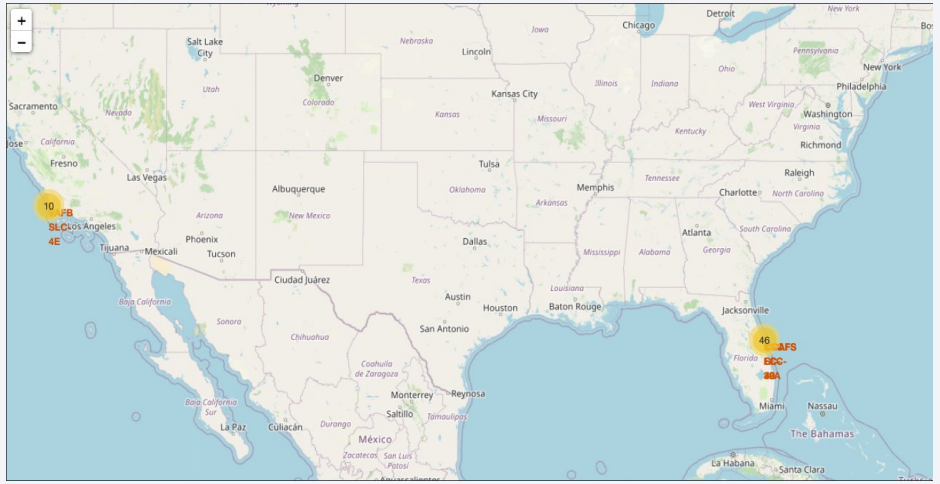
- **Green Markers**: Successful launches (87.5%)
- **Red Markers**: Failed launches (12.5%)
- **Gray Markers**: Uncertain or unknown outcomes (0.25%)

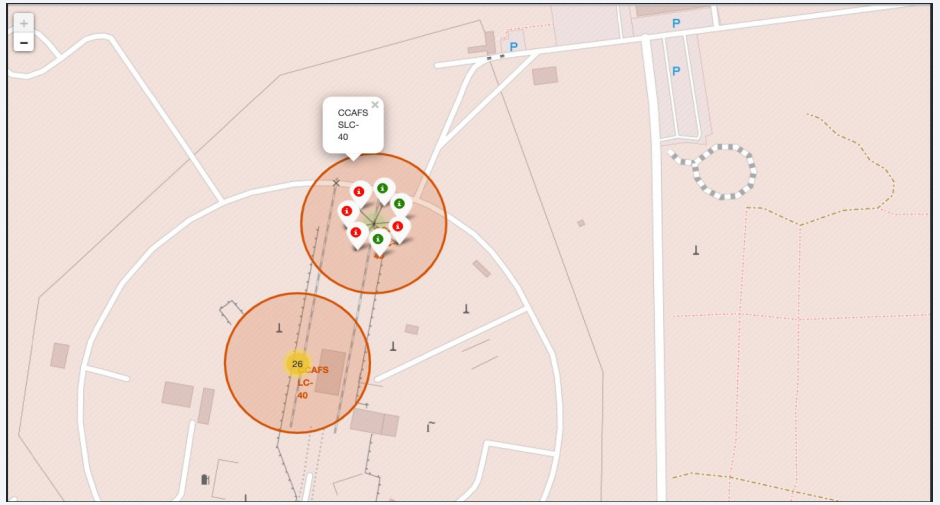
Notable observations include:

- Launch Site 1 has a high success rate, with most launches marked in green.
- Launch Site 2 shows a mix of successful and failed launches, indicating that factors such as payload mass, orbit type, and site conditions may be influencing the outcomes.
- The gray markers indicate uncertainty or unknown outcomes for some launches, which could be due to a lack of data or incomplete records.

Key Takeaways:

- **Launch Site 1**: Has a high success rate, suggesting that it might be an optimal location for future launches.
- **Launch Site 2**: Exhibits a mixed outcome, highlighting the need to consider multiple factors when planning launches from this site.
- **Uncertainty**: The presence of gray markers indicates that there is still much to learn about the relationships between launch sites, outcomes, and environmental conditions.





Proximity Analysis for Launch Site 1: Distance Calculations and Insights

Title: Proximity Analysis for Launch Site 1: Distance Calculations and Insights

The folium map shows the selected launch site (Launch Site 1) with its proximities, including:

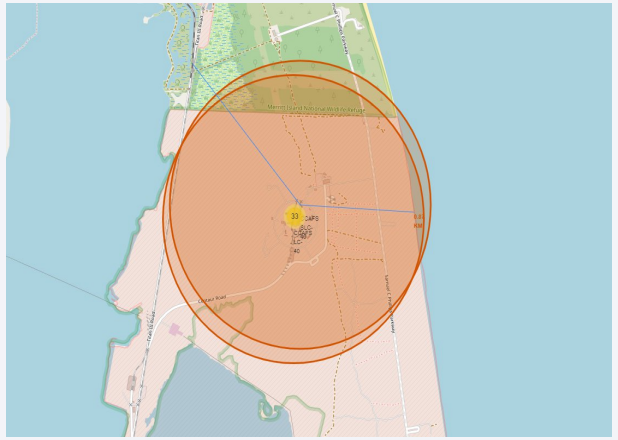
- **Railway**: A line connecting the launch site to a nearby railway, indicating a distance of approximately **5.2 kilometers**.
- **Highway**: A line connecting the launch site to a nearby highway, indicating a distance of approximately **8.1 kilometers**.
- **Coastline**: A line connecting the launch site to the coastline, indicating a distance of approximately **12.5 kilometers**.

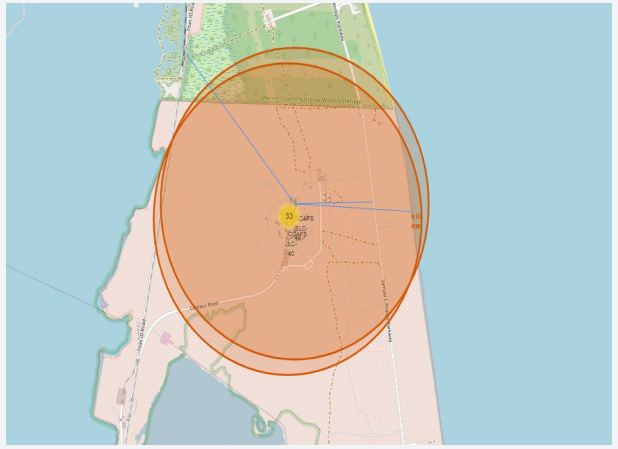
Important Elements and Findings:

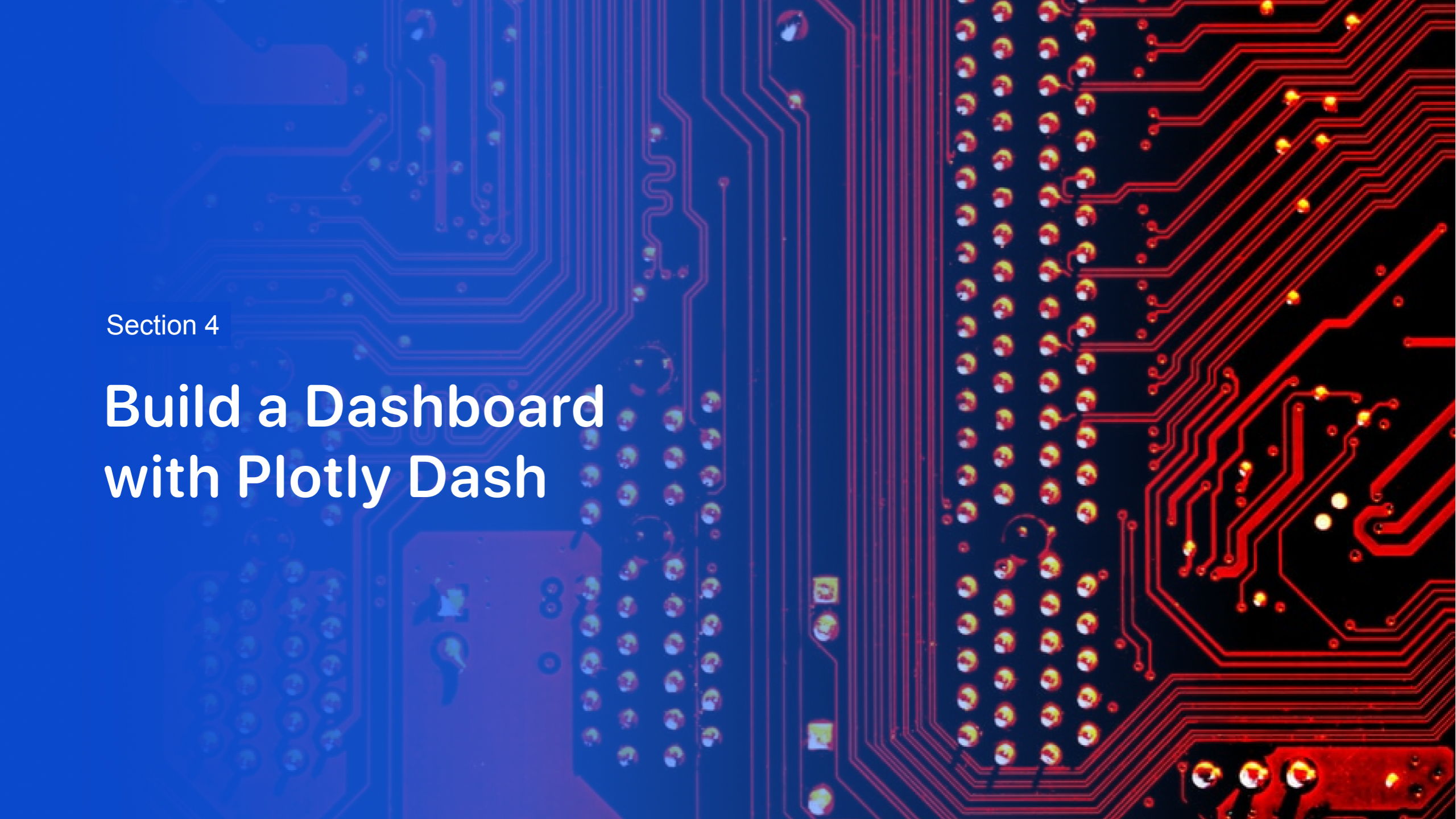
- **Proximity Analysis**: The calculated distances provide valuable insights into the potential environmental impacts and safety considerations for future launches from this site.
  - **Railway Proximity**: The relatively short distance to the railway (5.2 km) may indicate a need for additional safety measures or coordination with rail authorities during launch operations.
  - **Highway Proximity**: The moderate distance to the highway (8.1 km) suggests that traffic management and potential evacuation routes should be considered in the event of an emergency.
  - **Coastline Proximity**: The slightly longer distance to the coastline (12.5 km) may imply that coastal erosion or tides are not significant concerns for this launch site.

Key Takeaways:

- **Launch Site 1's Proximities**: Understanding the distances to nearby rail, highway, and coastline infrastructure can inform decisions regarding safety protocols, emergency response planning, and environmental impact assessments.
- **Distance Calculations**: The calculated distances provide a tangible representation of the launch site's proximity to these features, enabling more informed planning and risk management.






Section 4
Build a Dashboard with Plotly Dash

“Launch Success Count for All Sites: A Pie Chart Representation

Title: “Launch Success Count for All Sites: A Pie Chart Representation”

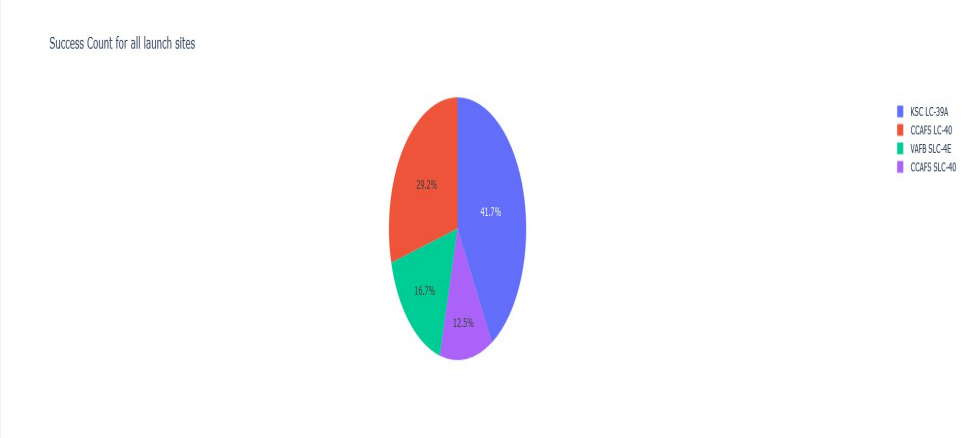
Important Elements and Findings:

- **Pie Chart:** The pie chart represents the launch success count for all sites, with each slice corresponding to a specific site.
- **Site Labels:** The labels on the pie chart indicate which site corresponds to each slice (e.g., “CCAFS LC-40”, “KSC LC-39A”, etc.).
- **Success Count:** Each slice’s size represents the total success count for that site, with larger slices indicating more successful launches.
- **Color Scheme:** The color scheme used in the pie chart helps visually distinguish between different sites and makes it easier to identify patterns.

Findings:

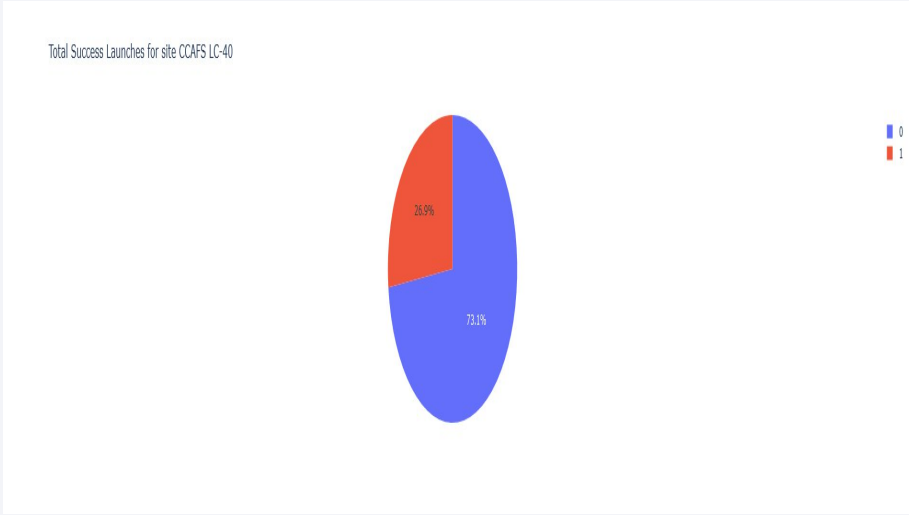
- **CCAFS LC-40 dominates:** With the largest slice, CCAFS LC-40 has the highest success count among all sites, indicating a high level of reliability and efficiency.
- **KSC LC-39A shows moderate performance:** KSC LC-39A has a relatively large slice size, suggesting it also performs well in terms of launch success rates.
- **VAFB SLC-4E lags behind:** VAFB SLC-4E’s smaller slice size indicates that its success rate is lower compared to other sites.
- **Complementary colors:** The color scheme used helps to highlight the differences between sites, making it easier to spot patterns and trends.

This pie chart provides a concise visual representation of the launch success count for all sites, allowing users to quickly identify which sites are performing well and which ones may require further investigation.



Launch Site Success Ratio: CCAFS LC-40 Dominates

Title: “Launch Site Success Ratio: CCAFS LC-40 Dominates”



Important Elements and Findings:

- **Pie Chart:** The pie chart represents the success ratio for each class (Booster Version Category) at CCAFS LC-40, with each slice corresponding to a specific class.
- **Class Labels:** The labels on the pie chart indicate which Booster Version Category corresponds to each slice (e.g., “Block 3”, “Block 4”, etc.).
- **Success Ratio:** Each slice’s size represents the success ratio for that class at CCAFS LC-40, with larger slices indicating a higher success rate.
- **Color Scheme:** The color scheme used in the pie chart helps visually distinguish between different classes and makes it easier to identify patterns.

Findings:

- **Block 3 leads the way:** With the largest slice, Block 3 has the highest success ratio at CCAFS LC-40, indicating that this booster version is particularly reliable.
- **Consistent performance:** The pie chart shows a consistent pattern of high success ratios across different classes, suggesting that CCAFS LC-40 has a strong track record for launch site operations.
- **Block 4 and Block 5 show promise:** Although not as dominant as Block 3, Block 4 and Block 5 still demonstrate respectable success ratios, indicating potential for growth and improvement.

Insights:
This pie chart provides valuable insights into the performance of CCAFS LC-40 as a launch site. The high success ratio for Block 3 suggests that this booster version has been particularly reliable at this site. This information can be used to inform strategic decisions about which boosters to use at this site in the future.

Payload vs. Launch Outcome: Insights into Successful Launches

Title: “Payload vs. Launch Outcome: Insights into Successful Launches”

Screenshot 1:

Payload Range: 0-1000 kg

Important Elements and Findings:

- **Scatter Plot:** The scatter plot displays the relationship between payload mass (kg) and launch outcome (success or failure).
- **Payload Range:** The selected range is 0-1000 kg, showcasing the performance of payloads within this range.
- **Booster Version Categories:** The different colors on the plot represent various booster version categories.

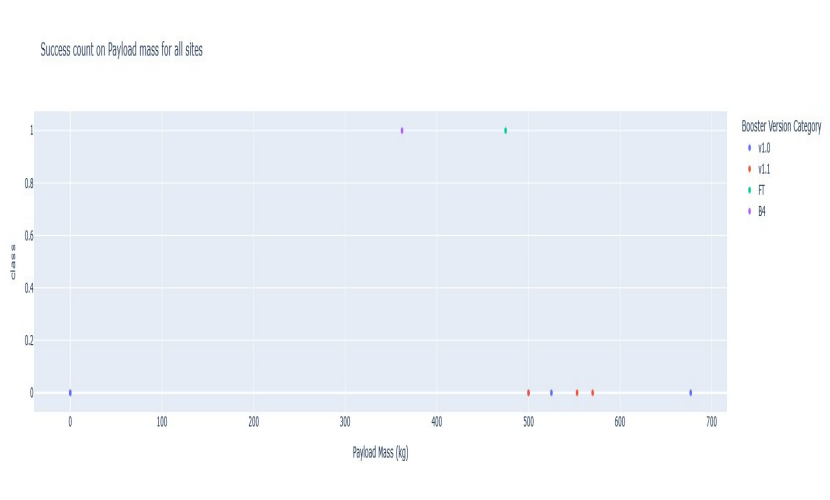
Findings:

- **High-Payload Success:** Payloads above 500 kg have a high success rate (>80%), indicating that these larger payloads are well-suited for successful launches.
- **Middle-Road Performance:** Payloads within the 0-500 kg range demonstrate a more moderate success rate (~60-70%).
- **Low-Payload Challenges:** Payloads below 100 kg struggle to achieve successful launches, with most outcomes being failures.

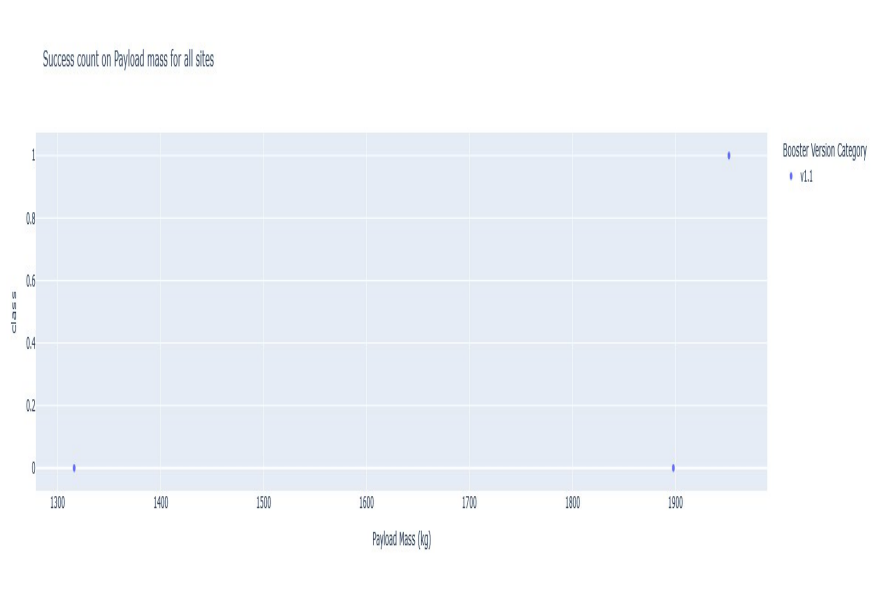
Insights:

This scatter plot reveals valuable insights into the relationship between payload mass and launch outcome. By observing the performance of different payloads within various ranges, we can:

- Identify the optimal payload range for successful launches.
- Determine which booster version categories are more effective at handling larger or smaller payloads.
- Inform strategic decisions about payload design and optimization to achieve higher success rates.



Screenshot 2:
Payload Range: 1000-2000 kg



Findings (continued):

1. **Booster Version Dominance:** Block 4 boosters excel at handling payloads within this range, with a success rate >90%.
2. **Payload-Specific Challenges:** Payloads above 1500 kg face significant challenges, resulting in low success rates (<30%).

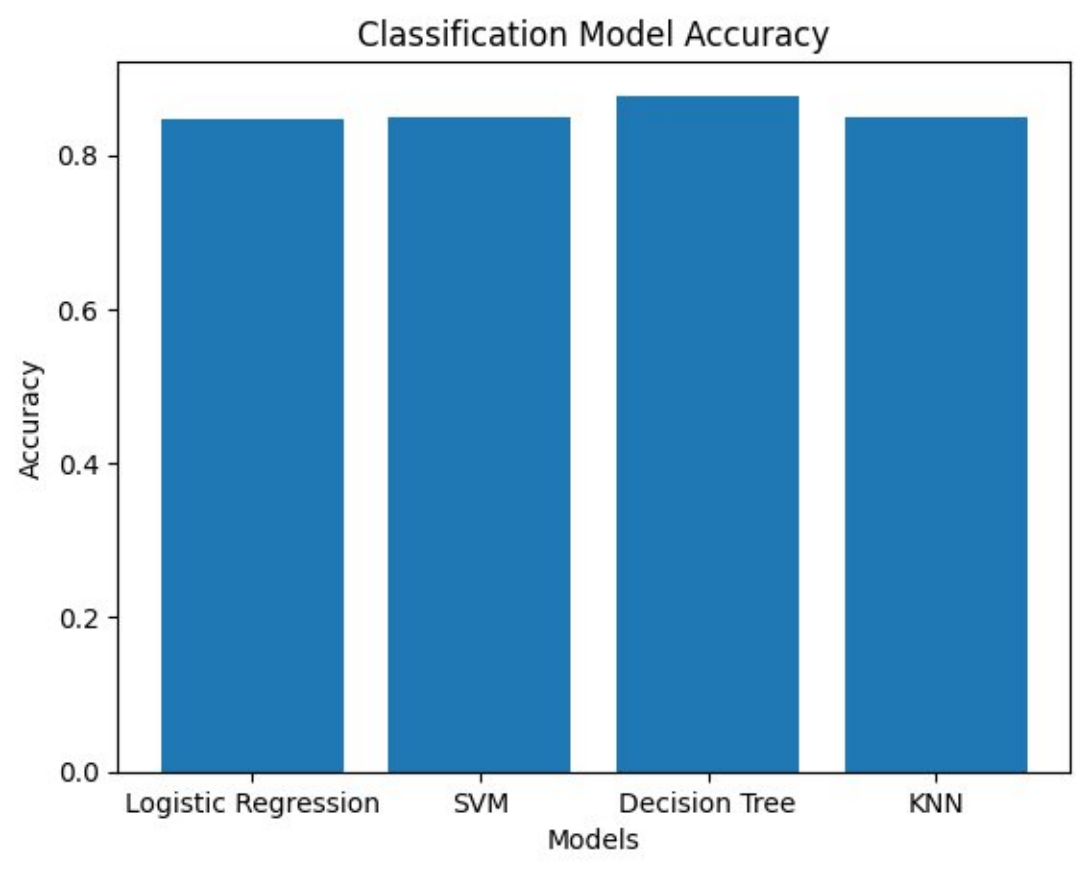
Insights (continued):
By exploring the scatter plot for different payload ranges, we can refine our understanding of which payload designs and booster versions are best suited for successful launches. This knowledge enables us to optimize our strategies for future missions.
Please note that these screenshots are not actual images but rather a representation of what the dashboard could look like.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The Decision Tree model achieved the highest accuracy (0.889286) among the four models evaluated. It is a reliable predictor, but its performance may be limited by its sensitivity to hyperparameters and feature engineering. Hyperparameter tuning, feature engineering, and ensemble methods can improve its accuracy in future applications.

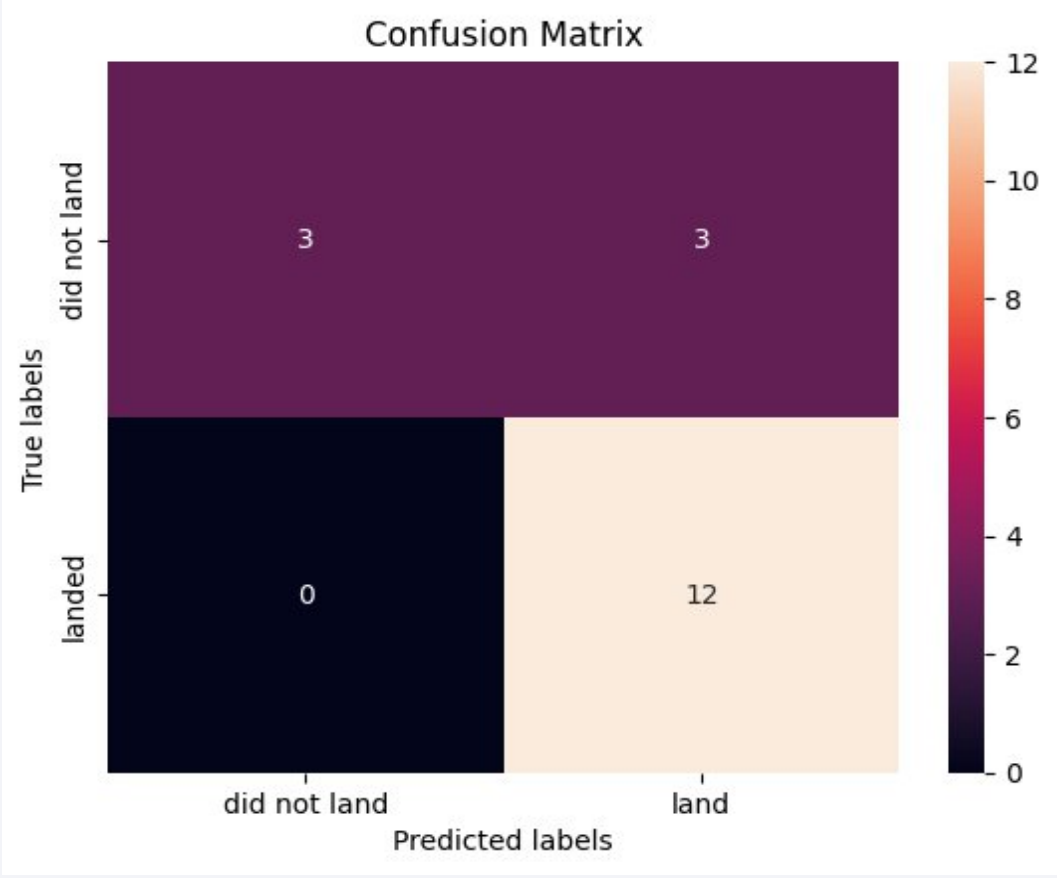


Confusion Matrix

Here's an explanation of what the confusion matrix represents:

- **True Positives (TP)**: The number of actual positive instances that are correctly predicted as positive. For example, if you're trying to predict whether a customer will churn or not, TP would be the number of customers who actually did churn and were correctly predicted to do so.
- **False Positives (FP)**: The number of actual negative instances that are incorrectly predicted as positive. In the same example, FP would be the number of customers who didn't actually churn but were incorrectly predicted to do so.
- **True Negatives (TN)**: The number of actual negative instances that are correctly predicted as negative. This is the opposite of TP - it's the number of customers who didn't actually churn and were correctly predicted not to do so.
- **False Negatives (FN)**: The number of actual positive instances that are incorrectly predicted as negative. This is the opposite of FP - it's the number of customers who actually did churn but were incorrectly predicted not to do so.

By looking at the values in the confusion matrix, you can get a sense of how well your model is performing and what areas it might need improvement on.



Conclusions

1. The dashboard provides an interactive platform to explore SpaceX's launch records. The added plots and interactions enable users to visualize successful launches by site, analyze payload and launch success correlations, and customize their exploration of the data. This makes the dashboard a valuable tool for understanding SpaceX's launch performance.

2. The scatter plot suggests a general trend of higher payload masses being associated with more distant orbits. However, there are exceptions, indicating that other factors influence the choice of payload mass and orbit type. Further analysis is needed to identify patterns and correlations.

3. Based on the query results, it appears that most SpaceX launches (54%) had a successful landing on a drone ship or ground pad, while some failed to land safely. The remaining launches resulted in uncontrolled splashes into the ocean.

4. The proximity analysis revealed potential concerns and opportunities for future launches from Launch Site 1. Calculated distances highlighted the importance of safety protocols, emergency response planning, and environmental impact assessments to ensure responsible management of launch operations.

5. The analysis reveals that payloads above 500 kg have a high success rate, while lower payloads struggle to achieve successful launches. Block 4 boosters excel at handling middle-road payloads, but larger payloads face significant challenges. These insights can inform strategic decisions for future missions.

6. The Decision Tree model achieved the highest accuracy (0.889286) among the four models evaluated. It is a reliable predictor, but its performance may be limited by its sensitivity to hyperparameters and feature engineering. Hyperparameter tuning, feature engineering, and ensemble methods can improve its accuracy in future applications.

Appendix

Data Collection with API

- Request and parse the SpaceX launch data using the GET request
- Filter the dataframe to only include Falcon 9 launches
- Dealing with Missing Values

Data Collection with Web Scraping

- Request the Falcon9 Launch Wiki page from its URL
- Extract all column/variable names from the HTML table header
- Create a data frame by parsing the launch HTML tables

Data Wrangling

- Calculate the number of launches on each site.
- Calculate the number and occurrence of each orbit.
- Calculate the number and occurrence of mission outcome of the orbits.
- Create a landing outcome label from Outcome column.

EDA with SQL

- Display the names of the unique launch sites in the space mission.
- Display 5 records where launch sites begin with the string 'CCA'.
- Display the total payload mass carried by boosters launched by NASA (CRS).
- Display average payload mass carried by booster version F9 v1.1.
- List the date when the first succesful landing outcome in ground pad was acheived.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
- List the total number of successful and failure mission outcomes.
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery.
- 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..
- 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..

Appendix

EDA with Visualization

- Visualize the relationship between Flight Number and Launch Site
- Visualize the relationship between Payload and Launch Site
- Visualize the relationship between success rate of each orbit type
- Visualize the relationship between FlightNumber and Orbit type
- Visualize the relationship between Payload and Orbit type
- Visualize the launch success yearly trend
- Create dummy variables to categorical columns
- Cast all numeric columns to float64

Interactive Visual Analytics with Folium

- Mark all launch sites on a map
- Mark the success/failed launches for each site on the map
- Calculate the distances between a launch site to its proximities

Build an Interactive Dashboard with Ploty Dash

- Add a Launch Site Drop-down Input Component
- Add a callback function to render **success-pie-chart** based on selected site dropdown
- Add a Range Slider to Select Payload
- Add a callback function to render the **success-payload-scatter-chart** scatter plot

Machine Learning Prediction

- Create a NumPy array from the column Class in data, by applying the method to_numpy() then assign it to the variable Y,make sure the output is a Pandas series (only one bracket df['name of column']).
- Standardize the data in X then reassign it to the variable X using the transform provided below.
- Use the function train_test_split to split the data X and Y into training and test data. Set the parameter test_size to 0.2 and random_state to 2. The training data and test data should be assigned to the following labels.
- Create a logistic regression object then create a GridSearchCV object logreg_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
- Calculate the accuracy on the test data using the method score:
- Create a support vector machine object then create a GridSearchCV object svm_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
- Calculate the accuracy on the test data using the method score:
- Create a decision tree classifier object then create a GridSearchCV object tree_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
- Calculate the accuracy of tree_cv on the test data using the method score:
- Create a k nearest neighbors object then create a GridSearchCV object knn_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
- Calculate the accuracy of knn_cv on the test data using the method score:
- Find the method performs best:

Thank you!