Winning Space Race with Data Science

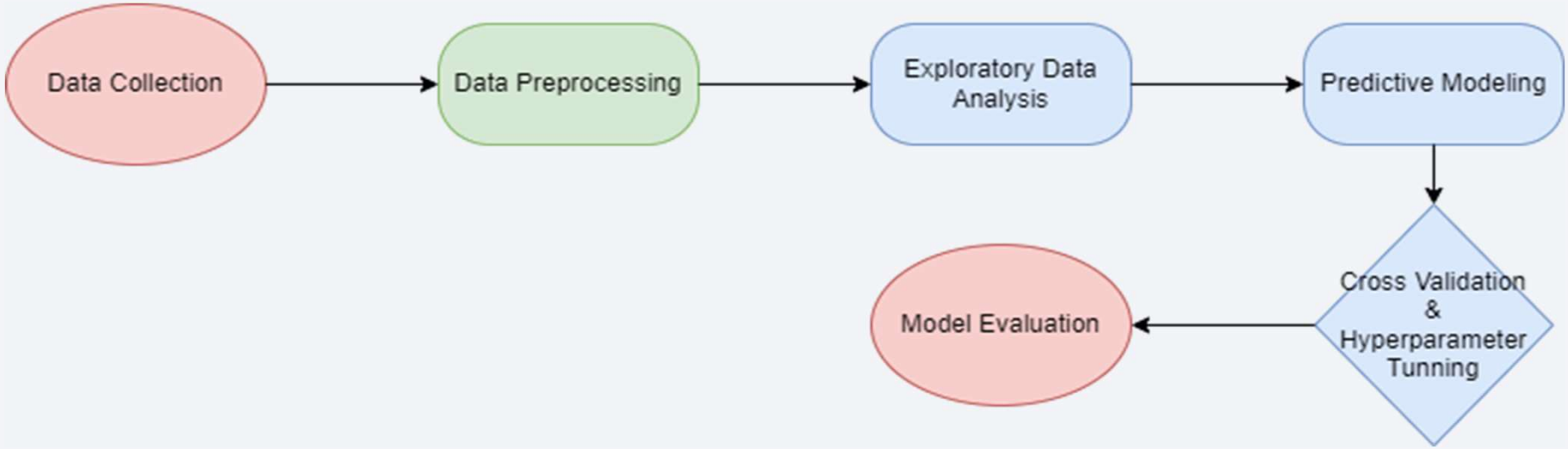
by Harold Merino Silva
August 11, 2024

Outline

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

Executive Summary

- Summary of methodologies



Executive Summary

- Summary of all results
  - **DataFrame of Falcon 9 Historical Launch Records**: Through the SpaceX API and web scraping from Wikipedia, the historical launch records were obtained and integrated into our DataFrame.
  - **Greater Understanding of the DataFrame and Relevant Variables:** Through Exploratory Data Analysis (EDA) using SQL queries and graphical visualizations, we identified the most relevant variables that contribute to a successful landing.
  - **Interactive Graphics Using Plotly Dash:** Interactive visualizations were created using Plotly Dash to explore various aspects of the Falcon 9 launch data.
  - **Best Hyperparameter for SVM, Classification Trees and Logistic Regression**: Using GridSearchCV, the optimal hyperparameters for each method were found.

Introduction

The commercial space age is here, with companies making space travel more affordable for everyone. One of the most successful is SpaceX. A key factor in SpaceX's success is that their rocket launches are relatively inexpensive. SpaceX advertises Falcon 9 rocket launches on its website at a cost of 62 million dollars, while other providers charge upwards of 165 million dollars per launch. Much of the savings comes from SpaceX's ability to reuse the first stage of their rockets, which is a large and expensive component. Unlike other rocket providers, SpaceX's Falcon 9 can recover and reuse this first stage, significantly reducing costs.

Space Y, a competitor to SpaceX founded by billionaire industrialist Allon Musk, aims to determine the competitive price for each launch by predicting whether SpaceX will reuse the first stage of its rockets. This will be achieved using machine learning models trained on publicly available SpaceX data, with the results visualized through an interactive dashboard.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology: The data used in this analysis was collected from public sources, including Wikipedia and the SpaceX API. These sources provided detailed information about the launches conducted by SpaceX, including launch sites, payloads, and whether the first stage was successfully reused or not.
- Perform data wrangling: After gathering the data, a data wrangling process was implemented to clean and organize it. Missing values were handled, categorical variables were encoded, and numerical features were normalized to ensure consistency and readiness for analysis.
- Describe how data was processed: The processing involved standardizing the data by converting it into formats compatible with machine learning models. This included managing missing data, scaling numeric variables, and encoding categorical features like launch sites and booster versions.

Methodology

Executive Summary

- **Perform exploratory data analysis (EDA) using visualization and SQL:** A thorough exploratory data analysis (EDA) was conducted to identify trends, patterns, and relationships within the dataset. SQL queries were employed to retrieve specific insights, and visualizations were generated using libraries like Matplotlib and Seaborn to provide a clearer understanding of the data distributions and key features. To enhance interactivity, Folium maps were created to visualize the geographical locations of launch sites, along with their outcomes. Additionally, a Plotly Dash dashboard was developed to allow users to interactively explore SpaceX's launch data, selecting different launch sites and payload ranges to analyze success rates.
- **Perform interactive visual analytics using Folium and Plotly Dash:** To enhance interactivity, Folium maps were created to visualize the geographical locations of launch sites, along with their outcomes. Additionally, a Plotly Dash dashboard was developed to allow users to interactively explore SpaceX's launch data, selecting different launch sites and payload ranges to analyze success rates.

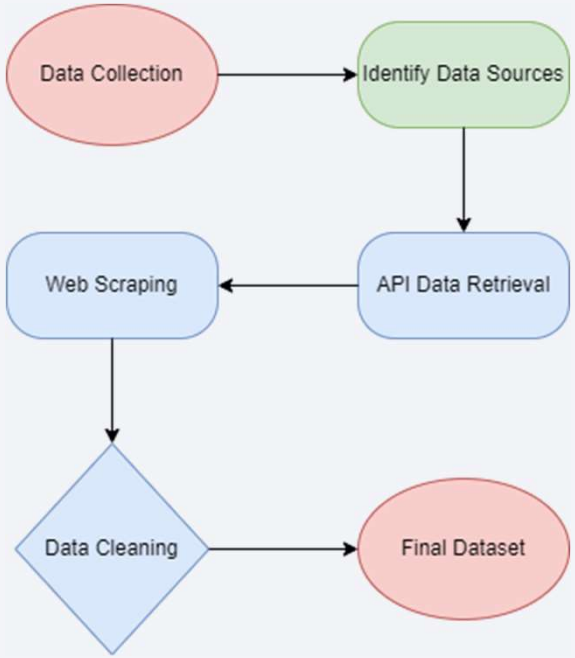
Methodology

Executive Summary

- **Perform predictive analysis using classification models:** Several classification models were implemented to predict whether SpaceX would reuse the first stage of the rocket based on the launch data. These models included logistic regression, decision trees, and k-nearest neighbors (k-NN). The classification models were built using Python's scikit-learn library. Hyperparameter tuning was performed using GridSearchCV to optimize model performance. The evaluation of these models primarily focused on accuracy, which was complemented by confusion matrices to assess the model's effectiveness in classifying successful and failed reuses.

Data Collection, Scraping and Wrangling

- Identify Data Sources: Data was sourced from the SpaceX API and Wikipedia, chosen for their structured and detailed records on SpaceX launches.
- API Data Retrieval: Launch details, including payloads and outcomes, were programmatically extracted from the SpaceX API using Python.
- Web Scraping: Supplementary data was gathered from Wikipedia using Python's BeautifulSoup for historical and additional launch details.
- Data Cleaning and Wrangling: Collected data was cleaned to remove duplicates, fill missing values, and ensure consistency across datasets.
- Dataset Merging: The datasets from the SpaceX API and Wikipedia were merged into a final, clean dataset for analysis.



https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/1.1%20jupyter-labs-spacex-data-collection-api.ipynb
https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/1.2%20jupyter-labs-webscraping.ipynb
https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/1.3%20labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with SQL

SQL Queries Implemented

- SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
- SELECT * FROM SPACEXTABLE WHERE “Launch_Site” LIKE “CCA%” LIMIT 5
- SELECT SUM(“PAYLOAD_MASS__KG_”) AS total_payload_mass FROM SPACEXTABLE WHERE "Customer“ LIKE ''%NASA (CRS)%”
- SELECT AVG(“PAYLOAD_MASS__KG_”) AS mean_plm_KG FROM SPACEXTABLE WHERE “Booster_Version” = 'F9 v1.1’
- SELECT MIN(“DATE”) FROM SPACEXTABLE WHERE “Landing_Outcome” = 'Success (ground pad)’
- SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)’ AND "Payload_Mass__KG_" BETWEEN 4000 AND 6000

https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/2.1%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

EDA with SQL

SQL Queries Implemented

- SELECT SUM("Landing_Outcome" like '%Success%') AS 'Success', SUM("Landing_Outcome" like '%Failure%') AS 'Failure' FROM SPACEXTABLE
- SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (select MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)
- SELECT SUBSTR("DATE",6,2) AS 'Month', "Landing_Outcome","Booster_Version", "Launch_Site" FROM SPACEXTABLE WHERE SUBSTR("DATE",1,4) = '2015' AND "Landing_Outcome" = 'Failure (drone ship)'
- SELECT "Landing_Outcome", COUNT(*) AS 'count' FROM SPACEXTABLE WHERE "DATE" BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY count DESC

https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/2.1%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

EDA with Data Visualization

- Scatter Point Chart (Hue: Class Value):
  - **Flight Number vs Launch Site :** to identify patterns between the number of flights and the launch site, as well as the relationship between the number of flights and the success rate at different launch sites.
  - **Payload Mass vs Launch Site:** to reveal how payload mass might affect the success rate of missions at different launch sites
  - **Flight Number vs Orbit:** to understand if certain orbits have a higher success rate as the number of flights increases.
  - **Payload Mass vs Orbit:** to identify if certain payload masses are more successful in reaching particular orbits.

https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/2.2%20edadataviz.ipynb

EDA with Data Visualization

- Bar Chart: This chart helps compare the success rate across different orbit types, providing insight into which orbits are more successful in missions.
- Line Chart: It shows the trend of success rates over the years, providing insight into whether SpaceX missions have improved over time.

https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/2.2%20edadataviz.ipynb

Build an Interactive Map with Folium

- Markers
  - Launch Site Markers
  - Mission Outcome Markers
  - Distance Markers
- Circles
  - Launch Site Circle
- Lines
  - Coastline, Highway, Railway and City Lines
- Marker Clusters
- Mouse Position

Build an Interactive Map with Folium

- **Launch Site Identification:** The markers and circles around the launch sites help users easily identify the exact locations of SpaceX's various launch facilities. The popup labels and text annotations provide clear, instant information
- **Mission Outcome Visualization:** By adding color-coded markers (green for success, red for failure), the map visually communicates the outcomes of SpaceX missions. This is crucial for spotting patterns in success rates based on location.
- **Distance Measurement:** The distance markers and connecting lines between the launch sites and the proximities provide insights into the geographical context of the launches, such as how close a site is to the ocean, which could be relevant to the mission outcome.
- **Cluster Management:** The MarkerCluster object helps to manage the visibility of the markers when there are multiple points of interest in a small geographical area, ensuring that the map remains readable.
- **Interactive Exploration:** The mouse position tool allows for precise exploration of the map, helping users to examine the coordinates of various points of interest.

https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/3.1%20lab_jupyter_launch_site_location.ipynb

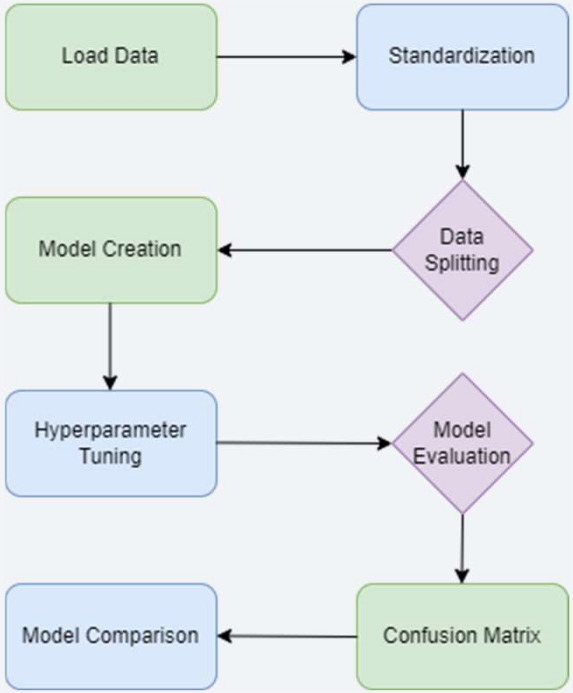
Build a Dashboard with Plotly Dash

- **Dropdown for Launch Site Selections:** Allows users to select a specific launch site or view data for all sites.
- **Pie Chart for Success Rates by Sites:** Displays the ratio of successful and failed launches by site.
- **Range Slider for Payload Mass Selection:** Enables users to filter data based on a specific range of payload masses.
- **Scatter Plot for Correlation Between Payload and Success:** Shows the relationship between payload mass and launch success, with color coding for the booster version category.

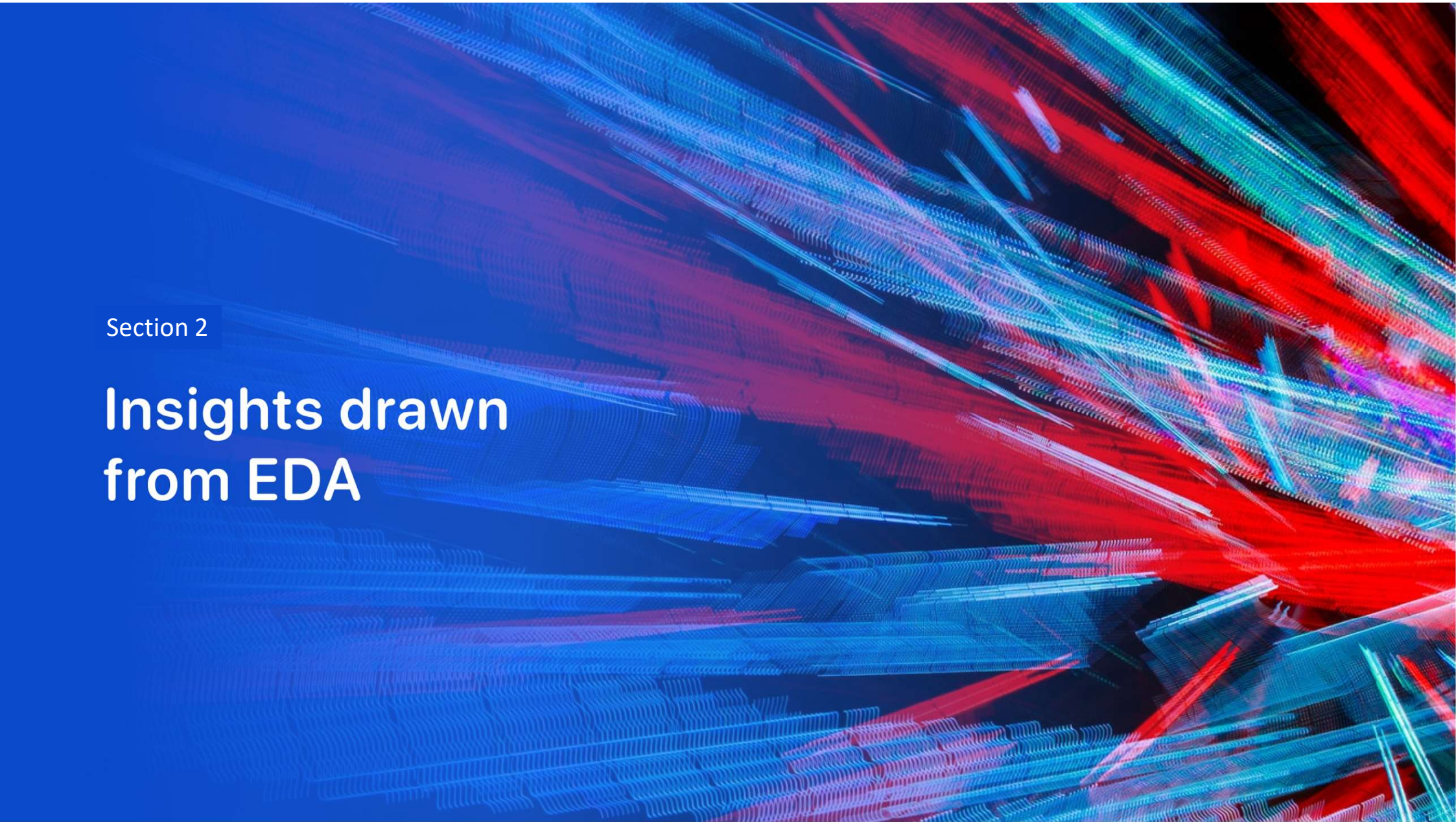
https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/3.2%20Build%20an%20Interactive%20Dashboard%20with%20Ploty%20Dash%20app.py

Predictive Analysis (Classification)

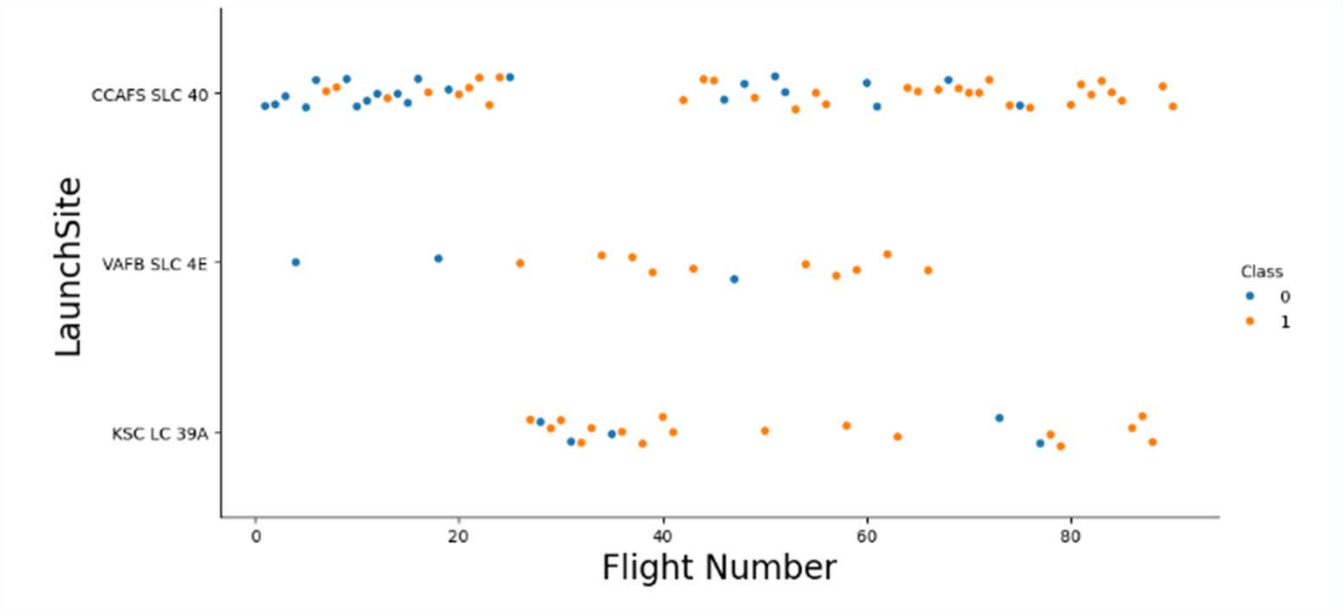
- **Data Preparation:** Loaded the datasets and created a Numpy array for the target variable “Y” from the “Class” column.
- **Data Splitting:** Split the dataset into training and testing sets using 80/20 split to create a robust evaluation process.
- **Model Selection and Hyperparameter Tuning:** Created a logistic regression model (“LogisticRegression”) and used “GridSearchCV” with cross-validation (cv=10) to identify the best hyperparameters from a set of defined options. This approach systematically evaluated different configurations to find the best-performing model.
- **Model Evaluation:** Evaluated the model on validation data during the hyperparameter tuning process by examining the best parameters and achieving an accuracy score on the training validation set. After selecting the best model, it was evaluated on the test data, achieving an accuracy of 0.833.
- **Confusion Matrix:** Used a confusion matrix to further evaluate the model's performance on test data. It was found that logistic regression effectively distinguished between the different classes but showed some false positives (incorrectly predicting a landing when it didn't occur).
- **Model Comparison:** The same procedure was applied to Support Vector Machine (SVM), Decision Tree (tree_cv), and KNN models, all resulting in similar confusion matrices.



https://github.com/HM3R1NO/IBM_Data_Science_Capston/blob/main/3.2%20Build%20an%20Interactive%20Dashboard%20with%20Ploty%20Dash%20app.py

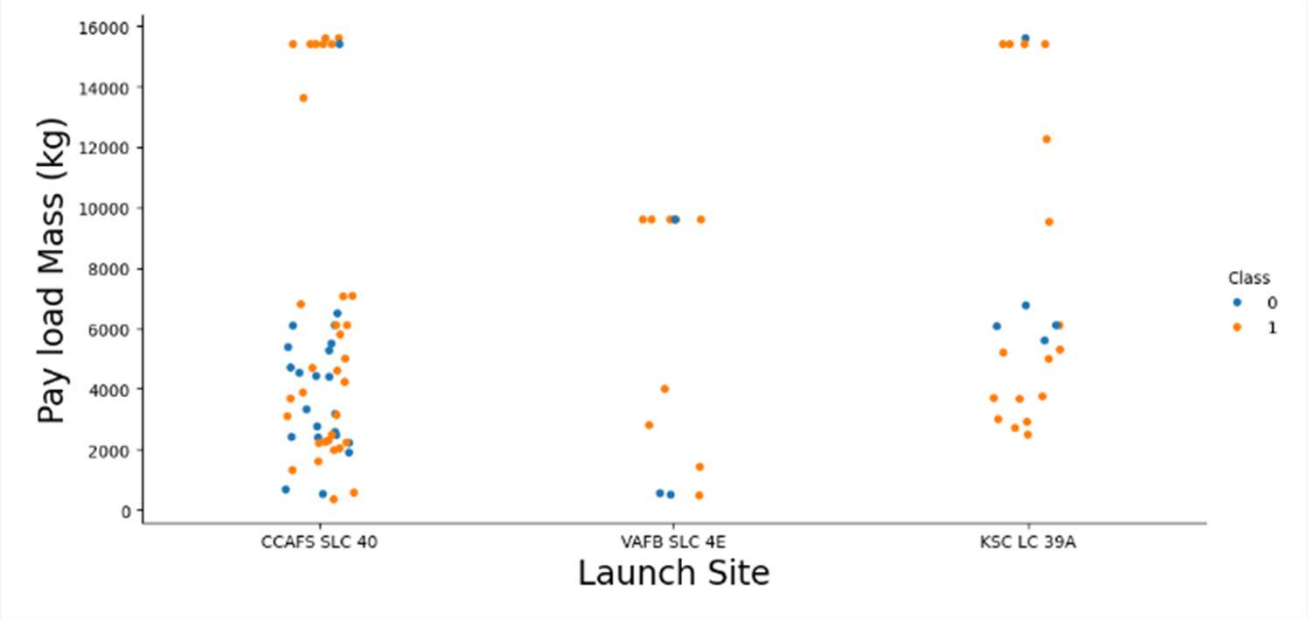

Section 2
Insights drawn
from EDA

Flight Number vs. Launch Site



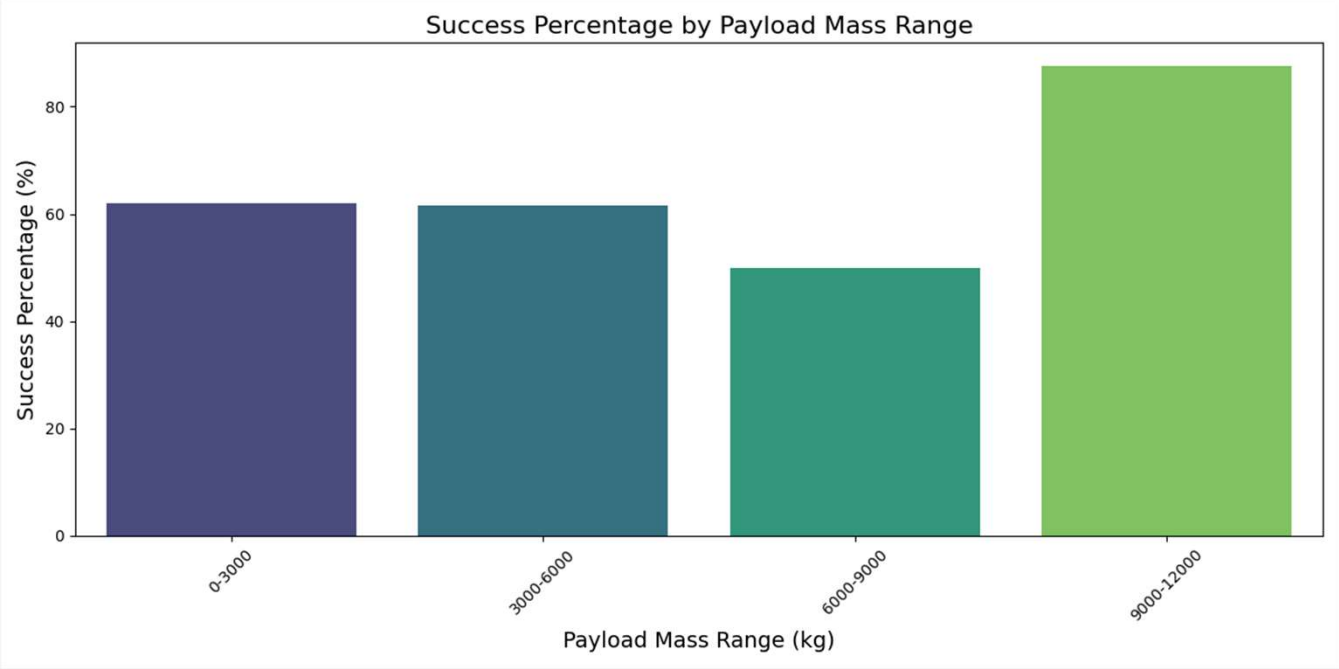
- This chart visualizes the distribution of flights across different launch sites, with color representing whether the mission was successful or not (Class).

Payload vs. Launch Site



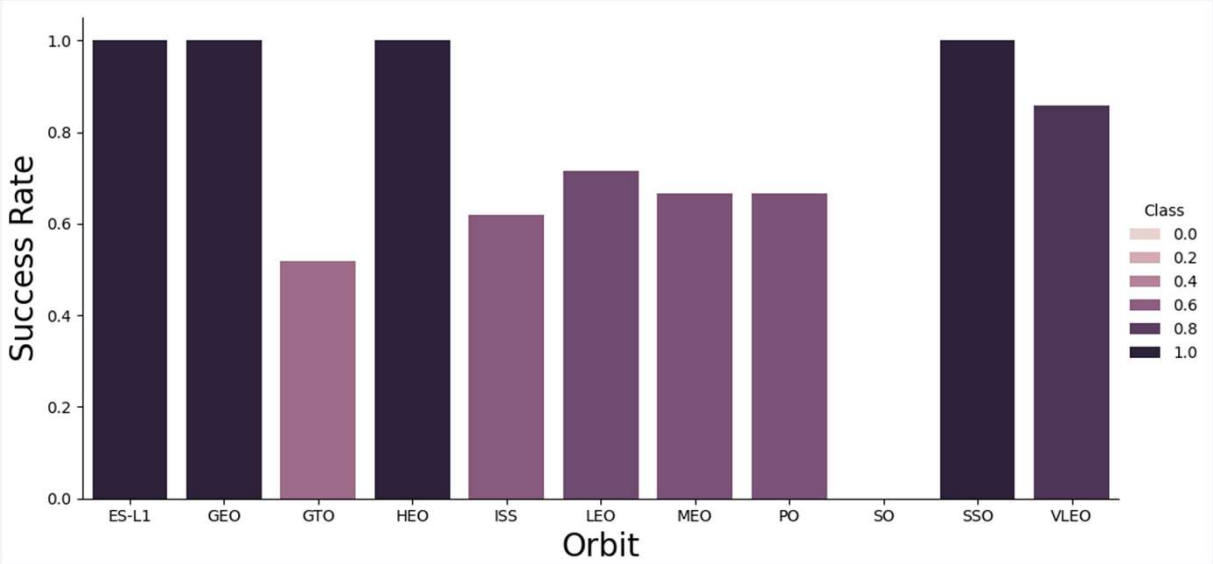
- This chart examines the correlation between payload mass and launch site, with color coding to differentiate between successful and unsuccessful missions (Class).

Payload vs. Success Rate



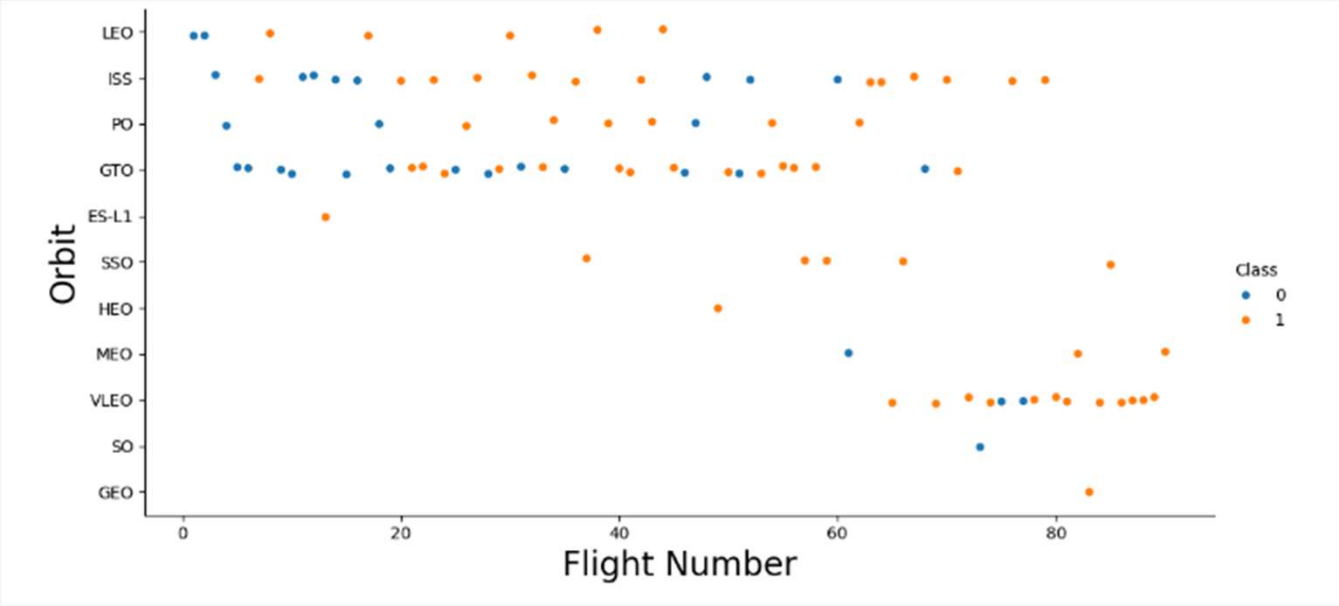
- The bar chart illustrating success percentage by payload mass range reveals how launch success varies with different payload masses.

Success Rate vs. Orbit Type



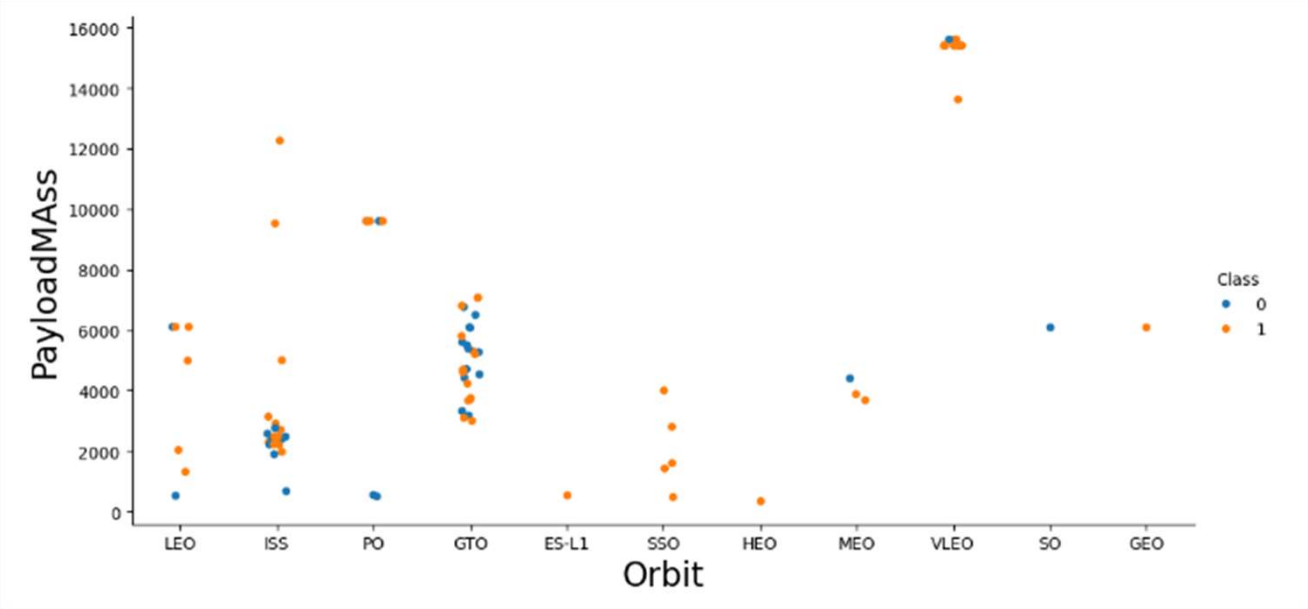
- The bar chart shows the average success rate for each orbit type, based on the mean value of the Class column.

Flight Number vs. Orbit Type



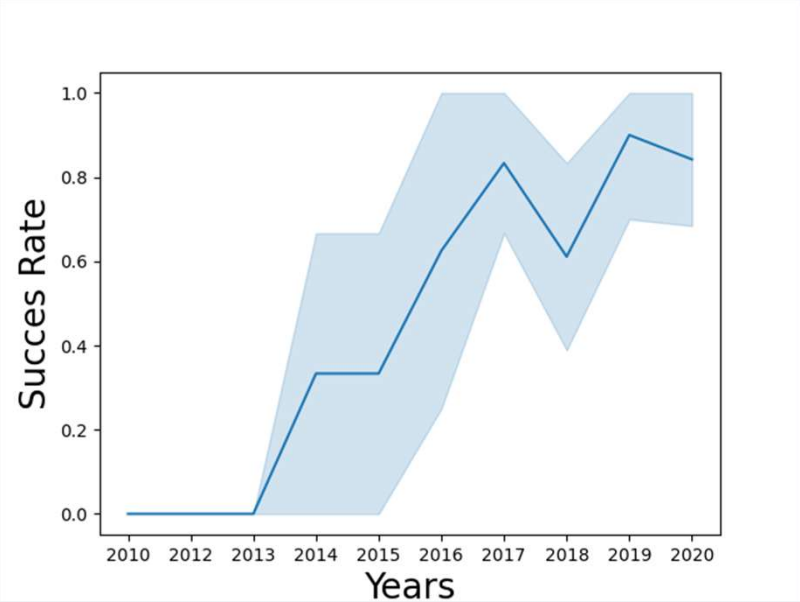
- This chart visualizes the relationship between flight number and orbit type, with color indicating mission success (Class).

Payload vs. Orbit Type



- This chart explores the relationship between payload mass and orbit type, with color representing the success or failure of the mission (Class).

Launch Success Yearly Trend



- The line chart plots the success rate of launches over time, using the year extracted from the launch date.

All Launch Site Names

```
%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

- This query retrieves the unique values from the column "Launch_Site" in the SPACEXTABLE. The DISTINCT keyword ensures that duplicate entries are removed, so only distinct launch site names are returned.

Launch Site Names Begin with 'CCA'

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

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

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

- The result of this query will show the first 5 records for launches that occurred at any launch site starting with "CCA" (e.g., "CCAFS SLC 40"), providing insight into launches conducted at those locations.

Total Payload Mass

```
''' include "NASA (CRS), Kacific 1" to boosters launched by NASA(CRS)'''
%sql select sum("PAYLOAD_MASS__KG_") as total_payload_mass from SPACEXTABLE where "Customer" like '%NASA (CRS)%'
```

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

total_payload_mass
48213

- The query provides insight into the cumulative payload mass delivered for NASA's CRS missions.

Average Payload Mass by F9 v1.1

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

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

mean_plm_KG
2928.4

- The result gives the average payload mass for launches using the Falcon 9 version 1.1 booster, providing insights into the typical payload capacity for this specific booster configuration.

First Successful Ground Landing Date

```
%sql select min("Date") from SPACEXTABLE where "Landing_Outcome" = 'Success (ground pad)'
```

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

min("Date")
2015-12-22

- This date represents the first successful landing on a ground pad recorded in the dataset.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select "Booster_Version" from SPACEXTABLE where "Landing_Outcome" = 'Success (drone ship)' and "Payload_Mass__KG_" between 4000 and 6000
```

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

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

- This query identifies the specific versions of the Falcon 9 rocket that achieved successful landings on drone ships while carrying payloads in the specified mass range.

Total Number of Successful and Failure Mission Outcomes

```
%sql select sum("Landing_Outcome" like '%Success%') as 'Success', sum("Landing_Outcome" like '%Failure%') as 'Failure' from SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

Success	Failure
61	10

- The query counts the number of successful and failed landings recorded in the SPACEXTABLE by using conditional aggregation.

Boosters Carried Maximum Payload

```
%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

- The query retrieves the distinct booster versions associated with the maximum payload mass recorded in the SPACEXTABLE

2015 Launch Records

```
%sql select  substr("DATE",6,2) as 'Month', "Landing_Outcome","Booster_Version", "Launch_Site" from SPACEXTABLE where substr("DATE",1,4) = '2015' and "Landing_Outcome" = 'Failure (drone ship)'
 * 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

- The query extracts and displays the month, landing outcome, booster version, and launch site for all entries in 2015 where the landing outcome was a "Failure (drone ship)".

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

```
%sql select "Landing_Outcome", Count(*) as 'count' from SPACEXTABLE where "DATE" between '2010-06-04' and '2017-03-20' group by "Landing_Outcome" order by count desc
```

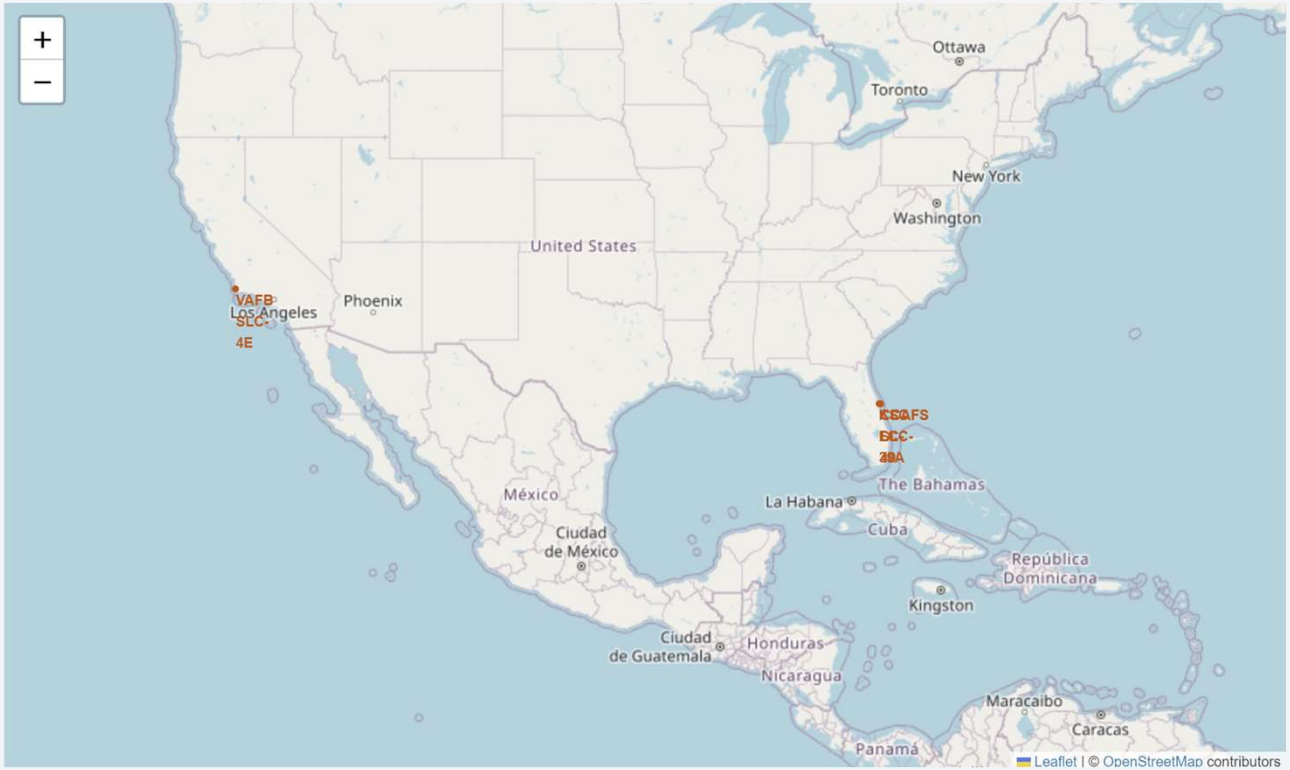
```
 * sqlite:///my_data1.db
Done.
```

Landing_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

- This summary provides insights into the distribution of landing outcomes over the specified time period, showing which outcomes were most common and which were less frequent.

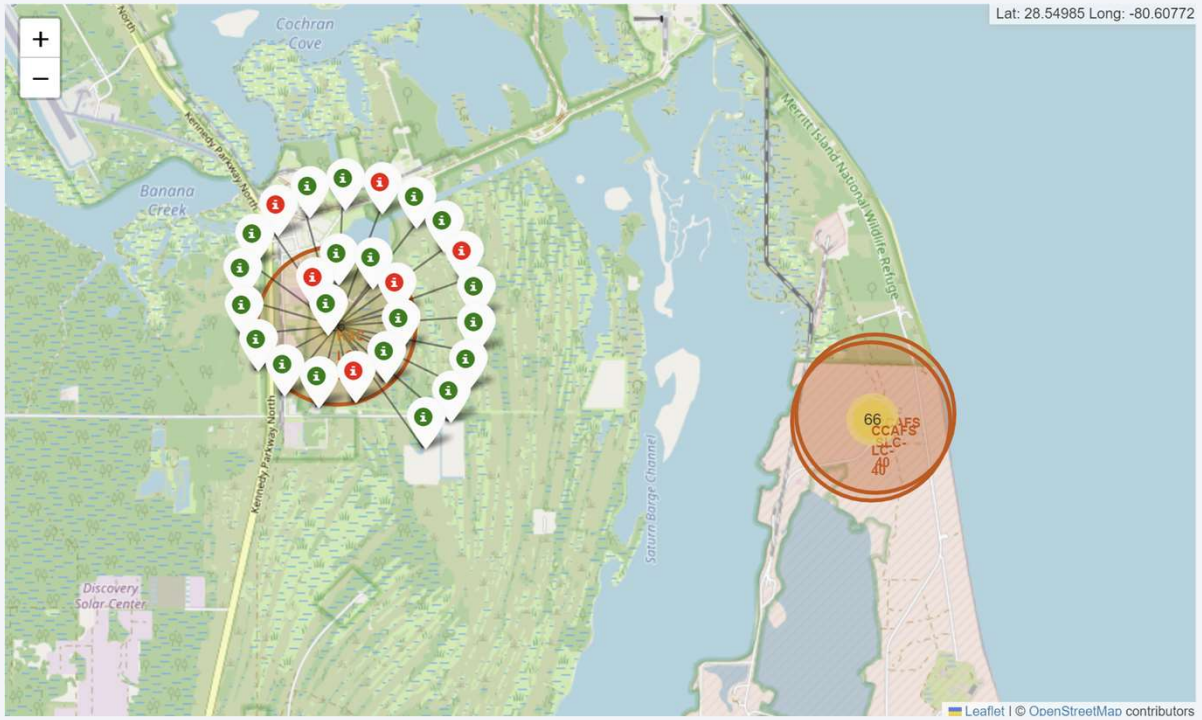

Section 3
Launch Sites
Proximities Analysis

SpaceX Launch Sites



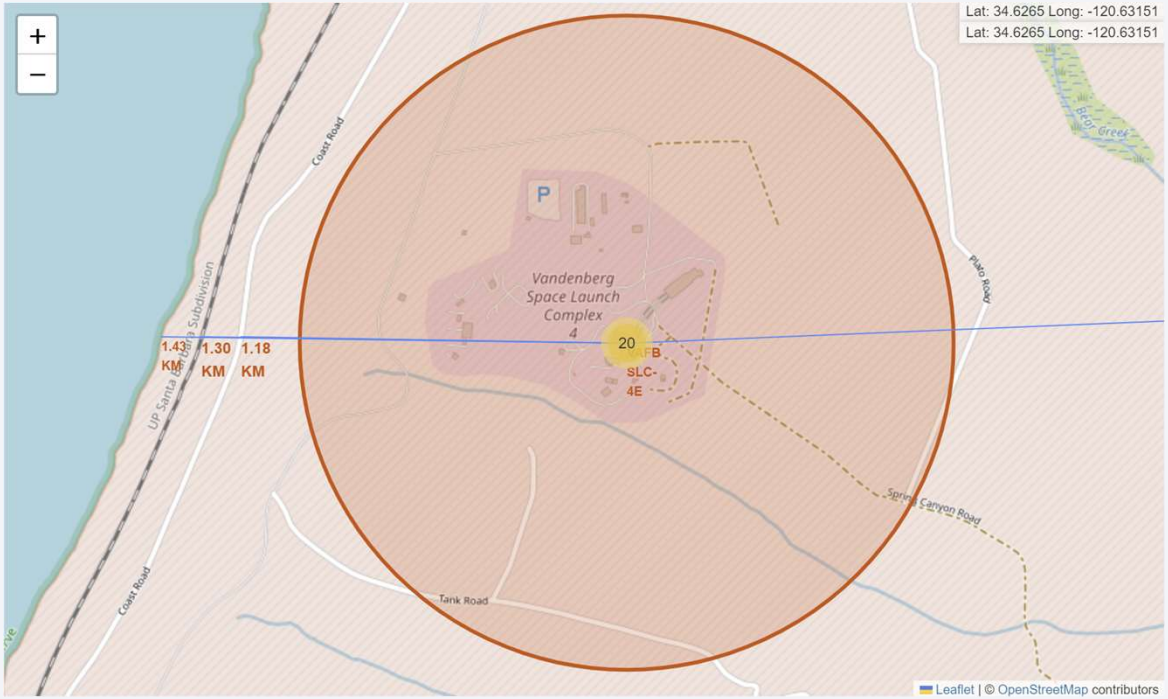
- There are launch site markers placed on the map, corresponding to SpaceX's known launch locations.
- The map shows the geographical spread of SpaceX's operations across the U.S., ensuring coverage for different types of missions.

Visualization of SpaceX Mission Outcomes: Success and Failure by Launch Site

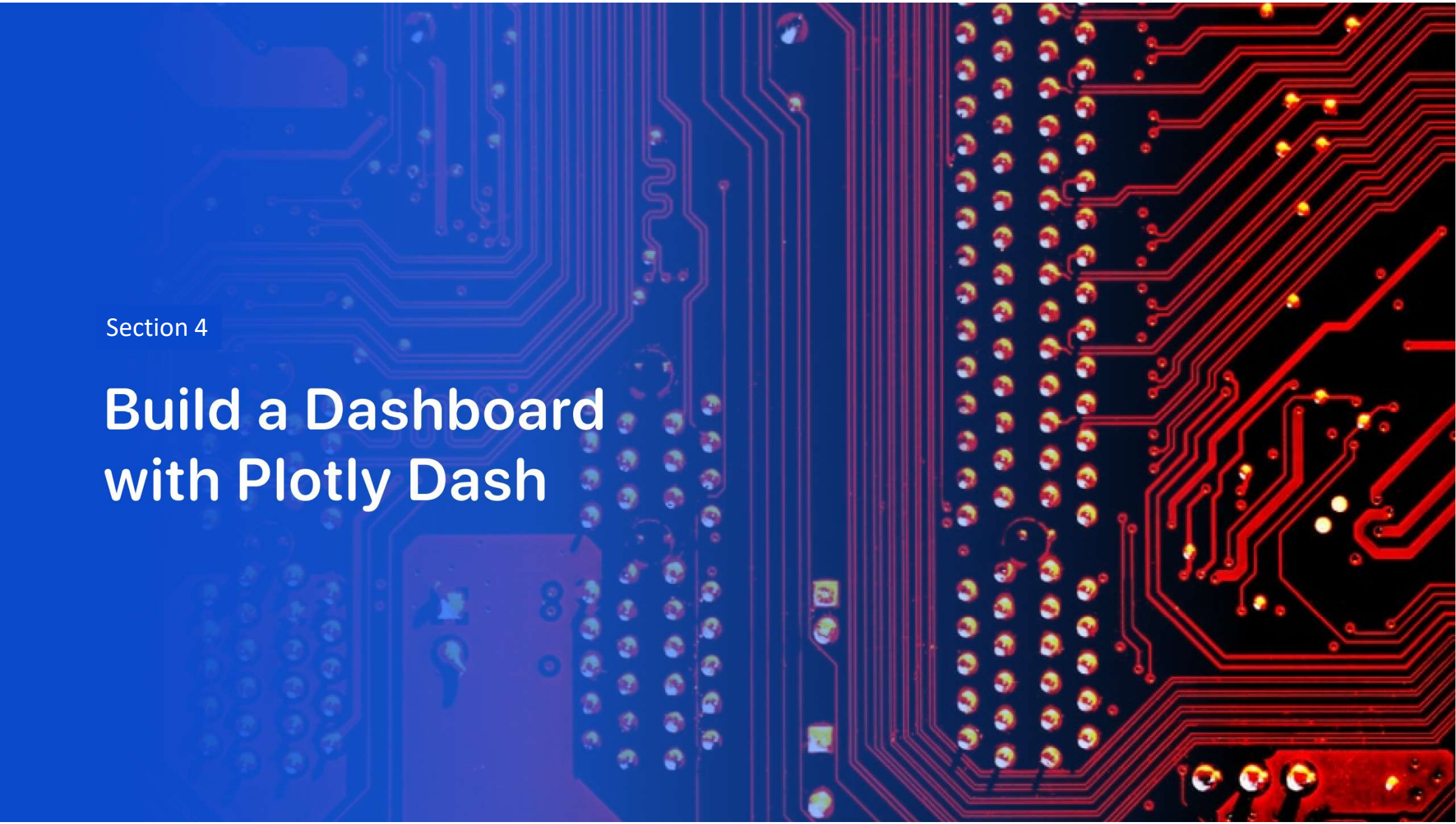


- **Color-Labeled Launch Outcomes:** These markers represent succes or failure of missions at different launch sites.
- **Cluster Functionality:** The clustering feature simplifies the visualization by grouping nearby markers together.

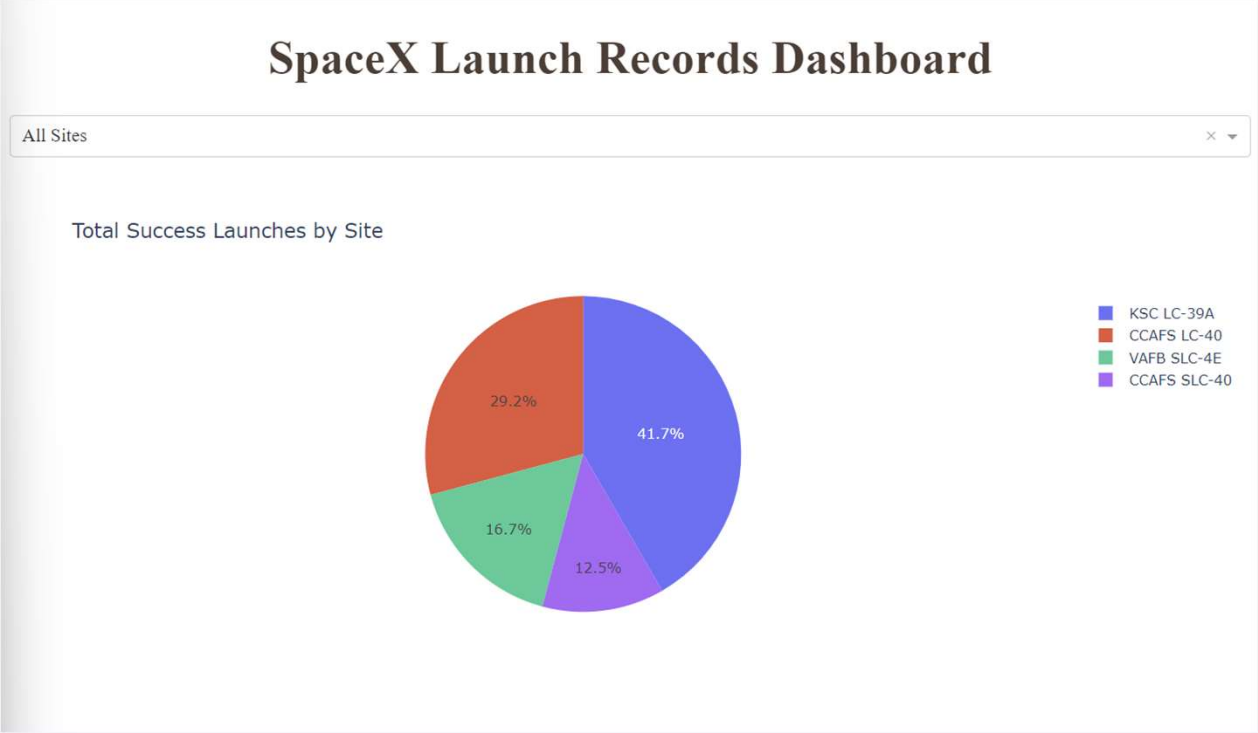
Critical Infrastructure Near Rocket Launch Sites



- **Distance Labels**: Surrounding points of interest (POIs), such as railway lines, highways, and coastlines, are shown with their distances from the launch site
- **Proximity Lines:** Lines are drawn from the launch site to the surrounding POIs, visually linking the site to the points of interest.
- This analysis of the map provides valuable context for understanding why certain locations are chosen for space launches, considering their proximity to critical infrastructure.

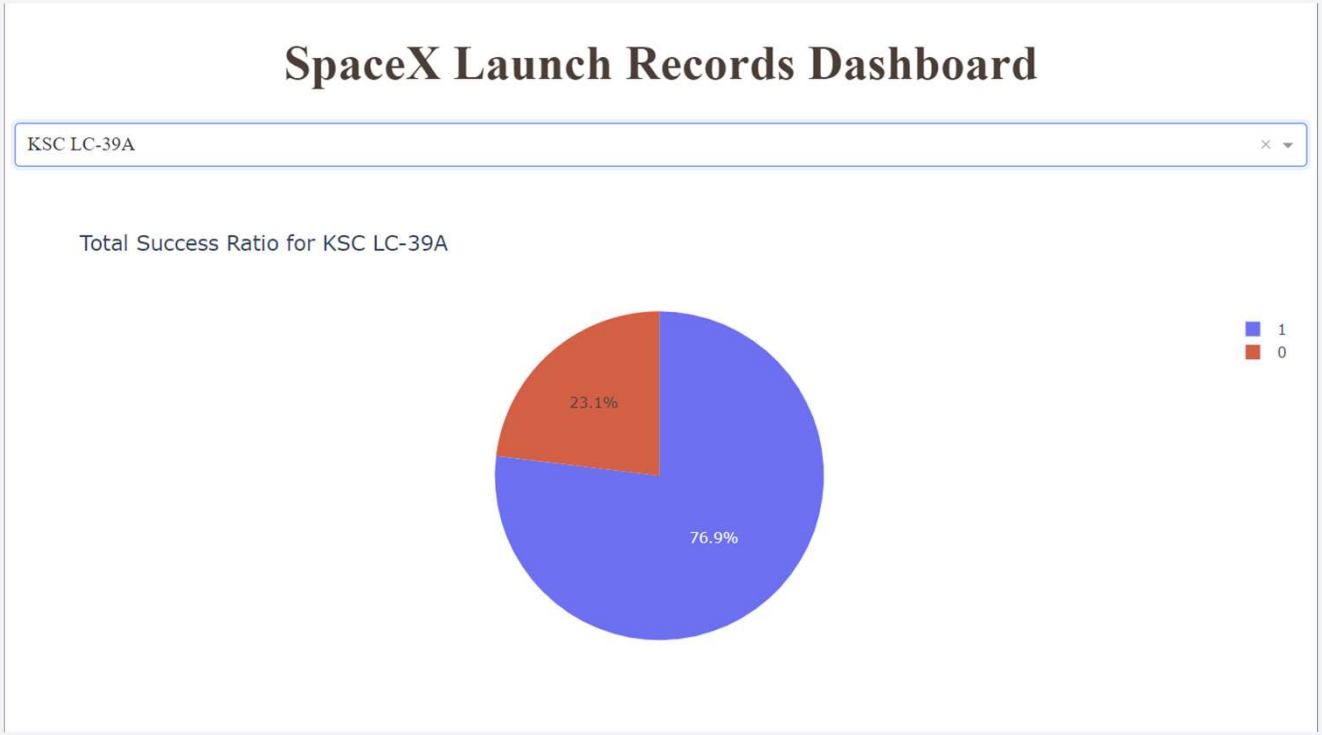

Section 4
Build a Dashboard with Plotly Dash

Pie Chart: Total Success Launches by Site



- This pie chart displays the distribution of successful launches across all launch sites. Each slice represents the proportion of successful launches for each site, giving an overview of which sites have the highest number of successful missions.

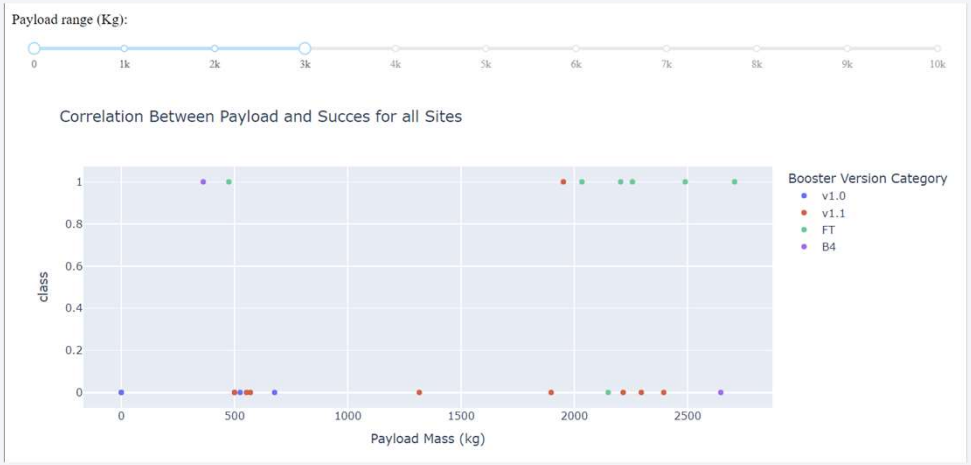
Pie Chart: Total Success Ratio for the Highest Success Launch Site



- This pie chart focuses on a specific launch site with the highest success rate (as selected from the dropdown). It shows the ratio of successful versus failed launches for this site. This visualization helps in understanding how successful launches are distributed for the top-performing site.

Scatter Plot: Payload vs. Launch Outcome for All Sites

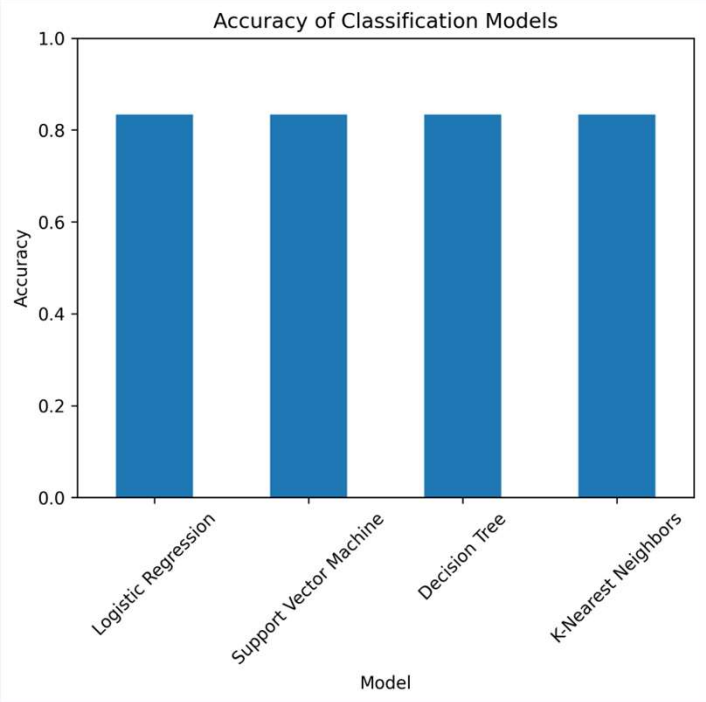
- This scatter plot illustrates the relationship between the payload mass and the success of launches across all sites. Different payload ranges, adjusted using the slider, highlight how payload mass correlates with launch outcomes (success vs. failure). This chart can help identify any trends or patterns in payload sizes relative to launch success rates.



Section 5

Predictive Analysis (Classification)

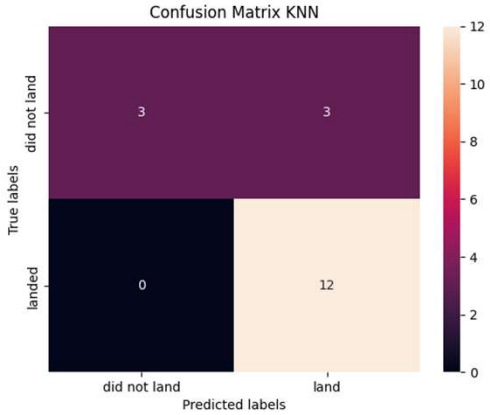
Classification Accuracy

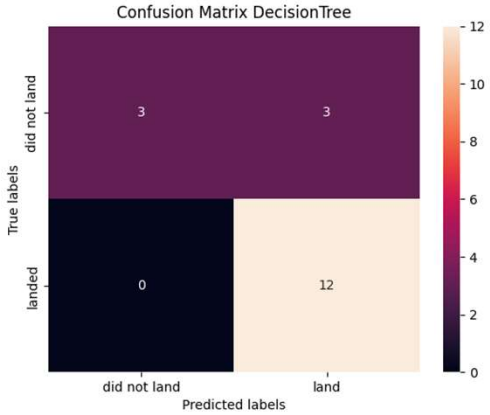


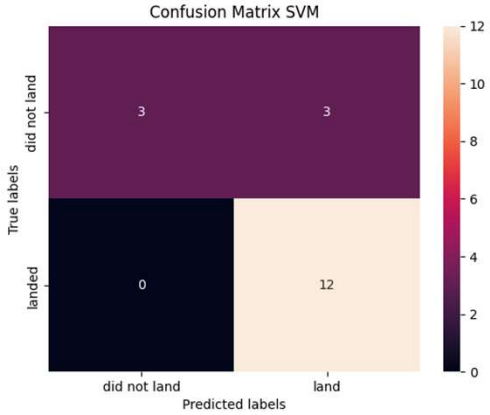
- The results across the models are nearly identical because the dataset is relatively small and lacks enough diversity.

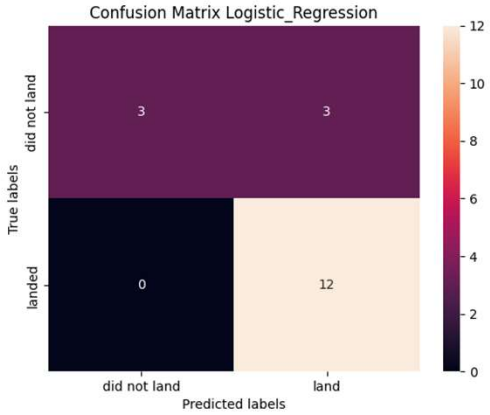
Confusion Matrix

- The confusion matrices are identical because the small dataset leads to similar predictions from each model. This results in similar false positive rates. A larger dataset might reveal more differences in model performance.









Conclusions

- Payload Mass Impact: The percentage of success for payload masses over 9000 kg is above 80%, while for payloads below 6000 kg, it hovers around 60%. This indicates a higher success rate for heavier payloads compared to lighter ones.
- SpaceX's Success Trend: SpaceX's success rate has generally improved over the years, but it appears to be approaching a plateau, suggesting that further significant improvements may be challenging.
- Top Launch Site: KGS LC 39A has had the highest success rate among all launch sites, highlighting it as the most reliable location for launches.
- Best Orbits for Success: The orbits GEO, HEO, SSO, and ES L1 have demonstrated the highest success rates, indicating that these orbits are associated with better launch outcomes.
- Strategic Launch Site Selection Based on Transportation Accessibility: The map demonstrates that certain launch sites are chosen for their accessibility to vital transportation routes, which can reduce costs and improve the efficiency of launch operations.
- Consistent Model Results: The confusion matrices and accuracy metrics are consistent across models due to the small dataset, indicating that a larger dataset is needed to capture meaningful differences in model performance.
- Model Overfitting: Even with a 20% train and 80% test split, with a low volume of data, models may become overfitted and be ineffective for future predictions.

Appendix

- For notebooks, dataset and scripts, follow this GitHub repository link:

https://github.com/HM3R1NO/IBM_Data_Science_Capston.git

Thank you!