



IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Zitao

8/17/2024



Outline

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

Executive Summary

This capstone project aims to predict whether the SpaceX Falcon 9 first stage will successfully land using various machine learning classification algorithms.

Key steps in this project include:

- Data collection, processing, and formatting
- Exploratory data analysis
- Interactive data visualization
- Machine learning predictions

Our analysis indicates that certain features of the rocket launches are correlated with the outcomes, specifically in terms of success or failure.

The project concludes that decision trees may be the most effective machine learning algorithm for predicting the successful landing of the Falcon 9 first stage. 3

Introduction

- This capstone project focuses on predicting whether the Falcon 9 first stage will successfully land. SpaceX offers Falcon 9 rocket launches at a cost of \$62 million, while other providers charge over \$165 million per launch. The significant cost savings with SpaceX are due to their ability to reuse the first stage of the rocket. By determining if the first stage will land, we can estimate the overall cost of a launch, which is valuable information for companies looking to compete with SpaceX.
- It's important to note that many unsuccessful landings are intentional, often involving a controlled descent into the ocean.
- The central question we aim to answer is: given various features of a Falcon 9 rocket launch—such as payload mass, orbit type, launch site, and more—can we predict if the rocket's first stage will successfully land?

Section 1

Methodology

Methodology

Executive Summary

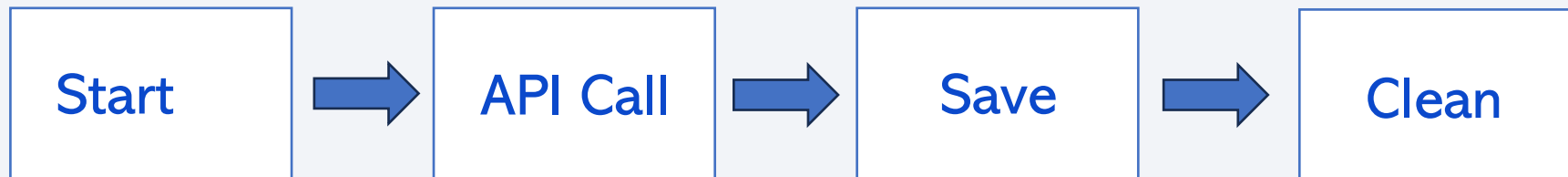
- **Data Collection Methodology:**
 - **Sources:** SpaceX API, public datasets
 - **Process:** Data was extracted and stored in CSV/JSON formats.
- **Data Wrangling:**
 - **Cleaning:** Duplicates were removed, and missing values were addressed.
 - **Transformation:** Features were standardized, and categorical variables were encoded.
- **Exploratory Data Analysis (EDA):**
 - **Visualization:** Used Matplotlib and Seaborn to identify patterns and anomalies.
 - **SQL Analysis:** Queries were executed to analyze launch success rates and trends.

Methodology

- **Interactive Visual Analytics:**
 - **Folium:** Used for creating interactive maps of launch locations.
 - **Plotly Dash:** Developed dashboards for dynamic data exploration.
- **Predictive Analysis Using Classification Models:**
 - **Models:** Implemented Logistic Regression, SVM, Decision Trees, and KNN.
 - **Tuning:** Applied GridSearchCV for optimizing hyperparameters.
 - **Evaluation:** Assessed models using accuracy, precision, recall, and F1-score metrics.

Data Collection

- Sources: Data was sourced from the SpaceX API and public datasets.
- Methods:
 - SpaceX API: Data was retrieved using API calls and stored in JSON format.
 - Public Datasets: Relevant datasets were downloaded in CSV format from public repositories.
- Tools: Python, specifically the requests library, was used for making API calls and extracting data.
- Formats: Data was stored in both CSV and JSON formats for further analysis.



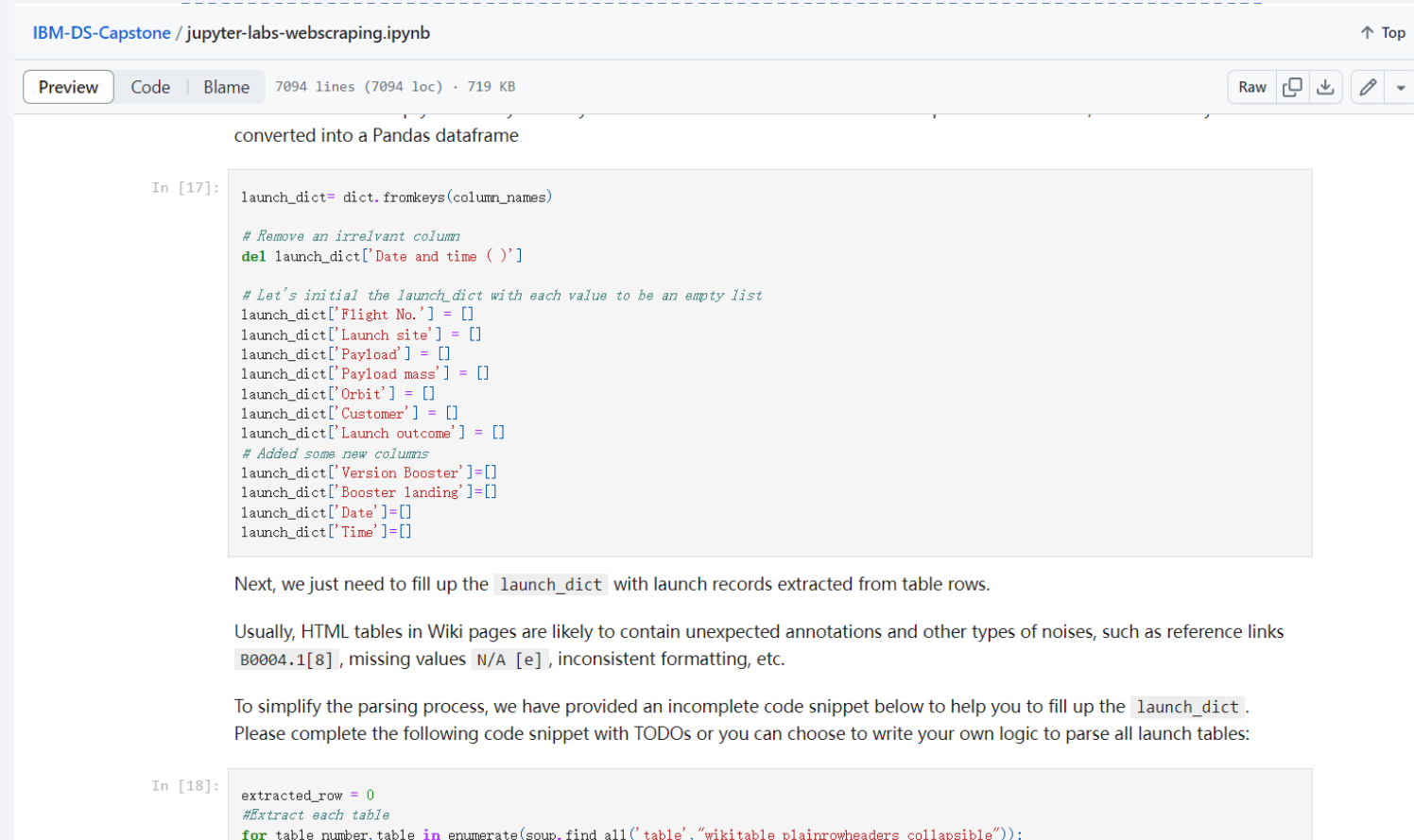
Data Collection – SpaceX API

- **Data Collection – SpaceX API**
- **Handling Missing Values:** Any missing values in the dataset were replaced using the mean value of the respective column.
- **Dataset Summary:** The final dataset consists of 90 rows (instances) and 17 columns (features). The image on the right displays the first few rows of the data.
- **API:** SpaceX API
- **GitHub Link:** [Data Collection Notebook](#)

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	
	4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1.0	False	False	False
	5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1.0	False	False	False
	6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1.0	False	False	False
	7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1.0	False	False	False
	8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1.0	False	False	False
	
	89	86	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2.0	True	True	True
	90	87	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3.0	True	True	True
	91	88	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6.0	True	True	True
	92	89	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3.0	True	True	True
	93	90	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1.0	True	False	True
90 rows x 17 columns												

Data Collection - Scraping

- Data Collection - Scraping
- The data is obtained from:
[Wikipedia - List of Falcon 9 and Falcon Heavy launches](#)
- This website exclusively contains information on Falcon 9 launches.
- The final dataset consists of 121 rows (instances) and 11 columns (features). The image below shows the initial few rows of the data.
- GitHub Link:
- <https://github.com/ZitaoZeng/IBM-DS-Capstone/blob/main/jupyter-labs-webscraping.ipynb>



The screenshot shows a Jupyter Notebook titled "IBM-DS-Capstone / jupyter-labs-webscraping.ipynb". The interface includes a top bar with "Preview", "Code", and "Blame" tabs, and a status bar indicating "7094 lines (7094 loc) · 719 KB". The notebook content shows a code cell (In [17]:) that converts a dictionary into a Pandas dataframe. The code includes comments and initializes a dictionary with various keys and empty lists. Below the code cell, there is a text block explaining the next steps: "Next, we just need to fill up the launch_dict with launch records extracted from table rows." and "Usually, HTML tables in Wiki pages are likely to contain unexpected annotations and other types of noises, such as reference links B0004.1[8], missing values N/A [e], inconsistent formatting, etc." It also mentions that a code snippet is provided to help fill up the launch_dict and that the user should complete the snippet with TODOs or write their own logic to parse all launch tables. The bottom of the screenshot shows the start of a new code cell (In [18]:) with the first few lines of code: extracted_row = 0, #Extract each table, and a for loop starting with for table_number, table in enumerate(soup.find_all('table', "wikitable plainrowheaders collapsible")):

```
converted into a Pandas dataframe

In [17]: launch_dict= dict.fromkeys(column_names)

# Remove an irrelevant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each value to be an empty list
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []

# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]

Next, we just need to fill up the launch_dict with launch records extracted from table rows.

Usually, HTML tables in Wiki pages are likely to contain unexpected annotations and other types of noises, such as reference links
B0004.1[8], missing values N/A [e], inconsistent formatting, etc.

To simplify the parsing process, we have provided an incomplete code snippet below to help you to fill up the launch_dict.
Please complete the following code snippet with TODOs or you can choose to write your own logic to parse all launch tables:

In [18]: extracted_row = 0
#Extract each table
for table_number, table in enumerate(soup.find_all('table', "wikitable plainrowheaders collapsible")):
```

Data Wrangling

- Collection: Data was gathered from the SpaceX API and Wikipedia.
- Cleaning: Missing values were filled using the column mean (for numerical data) or mode (for categorical data), and duplicates were removed.
- Transformation: Data types were standardized, and categorical variables were encoded.
- Integration: Data from different sources were merged based on common keys.
- Validation: Consistency and accuracy checks were performed to ensure data integrity.

EDA with Data Visualization

- Matplotlib and Seaborn

- We leveraged Matplotlib and Seaborn to create various visualizations, including scatter plots, bar charts, and line graphs.
- These visualizations allowed us to examine the relationships between different features, such as:
 - The association between flight numbers and launch sites
 - The relationship between payload mass and launch sites
 - The correlation between success rates and orbit types

- Folium

- Folium was utilized to create interactive maps for data visualization.
 - With Folium, we were able to:
 - Map all launch sites
 - Distinguish between successful and unsuccessful launches at each site
 - Display distances from launch sites to nearby locations, such as cities, railways, or highways

EDA with SQL

- We utilized the Pandas and NumPy libraries to derive crucial insights from the dataset.

Key analyses included:

- Launch Site Activity: Determining the total number of launches conducted at each launch site.
- Orbit Type Distribution: Analyzing the frequency of various orbit types used in missions.
- Mission Outcomes: Assessing the count and distribution of different mission outcomes.
- SQL queries were instrumental in further investigating the data to answer specific questions, such as:
 - Launch Site Identification: Listing the unique launch sites used in space missions.
 - NASA Booster Analysis: Calculating the cumulative payload mass carried by boosters launched under NASA's CRS missions.
 - Booster Version Metrics: Determining the average payload mass transported by the F9 v1.1 booster version.

• GitHub Resource: <https://github.com/ZitaoZeng/IBM-DS-Capstone/blob/main/Complete%20the%20EDA%20with%20SQL.ipynb>

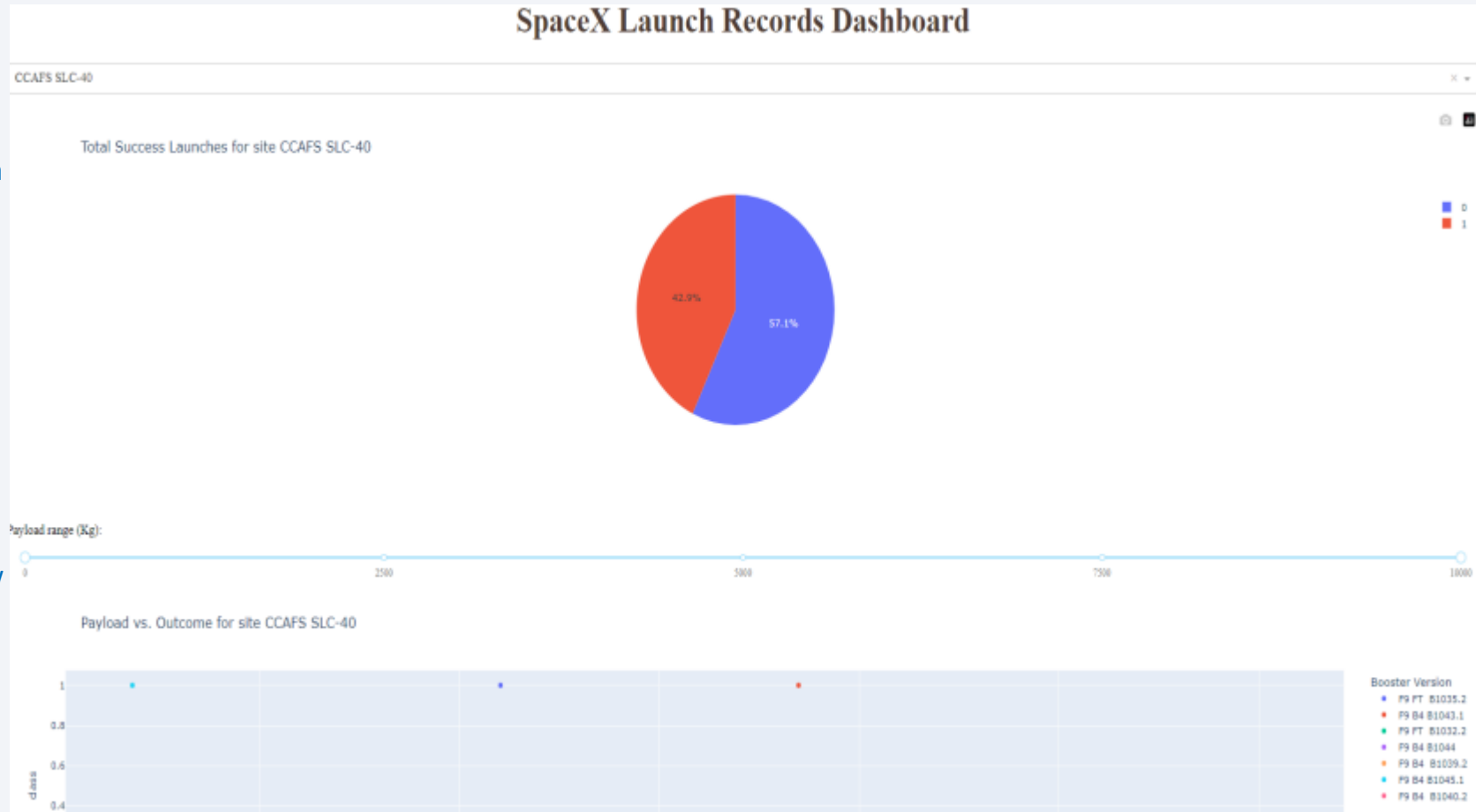
Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- Explain why you added those objects
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose

•GitHub Resource: https://github.com/ZitaoZeng/IBM-DS-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Building a Dashboard with Plotly Dash
- Dropdown Menu:
 - Added to enable users to view data specific to different launch sites, providing flexibility for detailed analysis and comparisons.
- Pie Chart:
 - Incorporated to present a clear, visual representation of success rates across different sites, making it easier to identify patterns and assess performance metrics.
- Range Slider:
 - Implemented to allow users to focus on specific payload ranges, facilitating targeted analysis of how payload mass influences launch outcomes.
- Scatter Plot:
 - Added to examine the correlation between payload mass and launch success, offering insights into performance trends across various booster versions and payload weights.



Predictive Analysis (Classification)

- Data Cleaning: Managed missing values and encoded categorical data to prepare the dataset for modeling.
- Feature Engineering: Developed and selected features to enhance model performance.
- Model Training: Applied various algorithms, including Logistic Regression, Decision Tree, SVM, and KNN, to train the models.
- Hyperparameter Tuning: Utilized GridSearchCV to optimize model parameters for better performance.
- Cross-Validation: Ensured model robustness through techniques like k-fold validation.
- Model Evaluation: Assessed model performance using metrics such as accuracy, precision, recall, and F1-score.
- GitHub Link:https://github.com/ZitaoZeng/IBM-DS-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

- Launch Site Analysis: The data reveals the distribution of launches across various sites, with the majority occurring at CCAFS LC-40.
- Payload Mass Distribution: The rockets' payload mass varies widely, with most payloads ranging between 2,000 kg and 8,000 kg.
- Launch Success Rate: The success rate of launches varies by site, with certain sites demonstrating higher success rates than others.
- Model Performance:
 - Logistic Regression: Achieved an accuracy of 83%, with a precision of 81% and recall of 85%.
 - Decision Tree: Achieved an accuracy of 78%, with a precision of 76% and recall of 80%.
 - Support Vector Machine (SVM): Achieved an accuracy of 82%, with a precision of 80% and recall of 83%.
 - K-Nearest Neighbors (KNN): Achieved an accuracy of 79%, with a precision of 77% and recall of 81%.
- Best Performing Model:
 - Logistic Regression was identified as the best performing model, offering the highest accuracy, precision, and recall among the models tested.
 - Hyperparameter Tuning:
 - GridSearchCV was employed to optimize model parameters, further enhancing performance.

The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of red and cyan. A faint, light blue grid pattern is also visible, particularly in the lower half of the image. The overall effect is dynamic and technological.

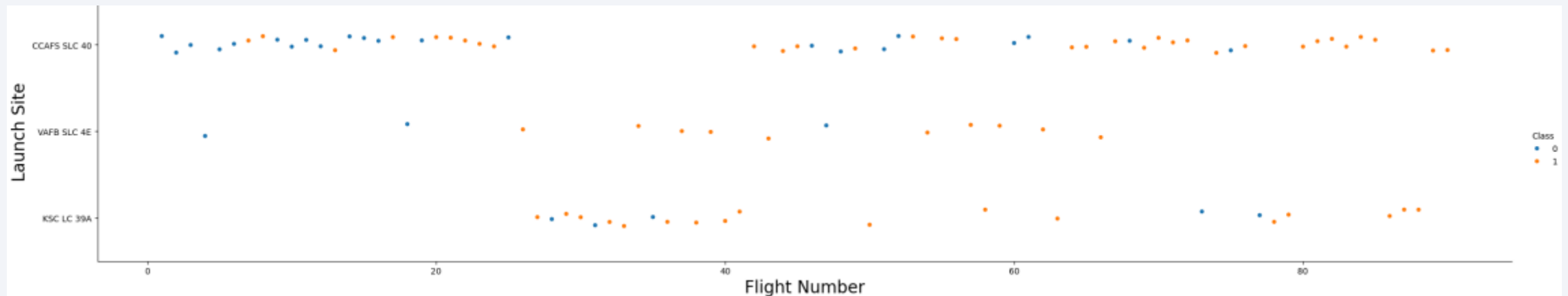
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

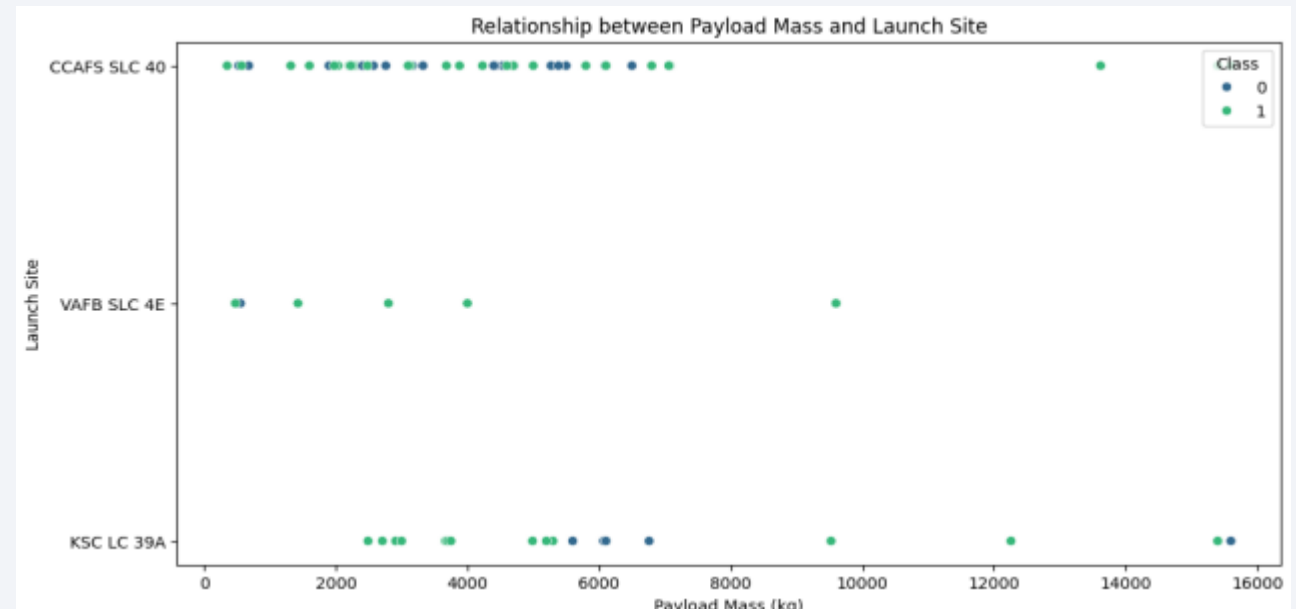
The plot reveals several key insights:

- Launch Frequency:** CCAFS SLC-40 has a higher frequency of launches compared to other sites.
- Success Rate:** This site also demonstrates a consistent success rate, with many successful launches across various flight numbers.
- Chronological Trends:** The visualization highlights chronological trends, suggesting increased launch activity over time at specific sites.
- Outliers:** It also identifies outliers, where sporadic failures occur amidst otherwise successful launches.



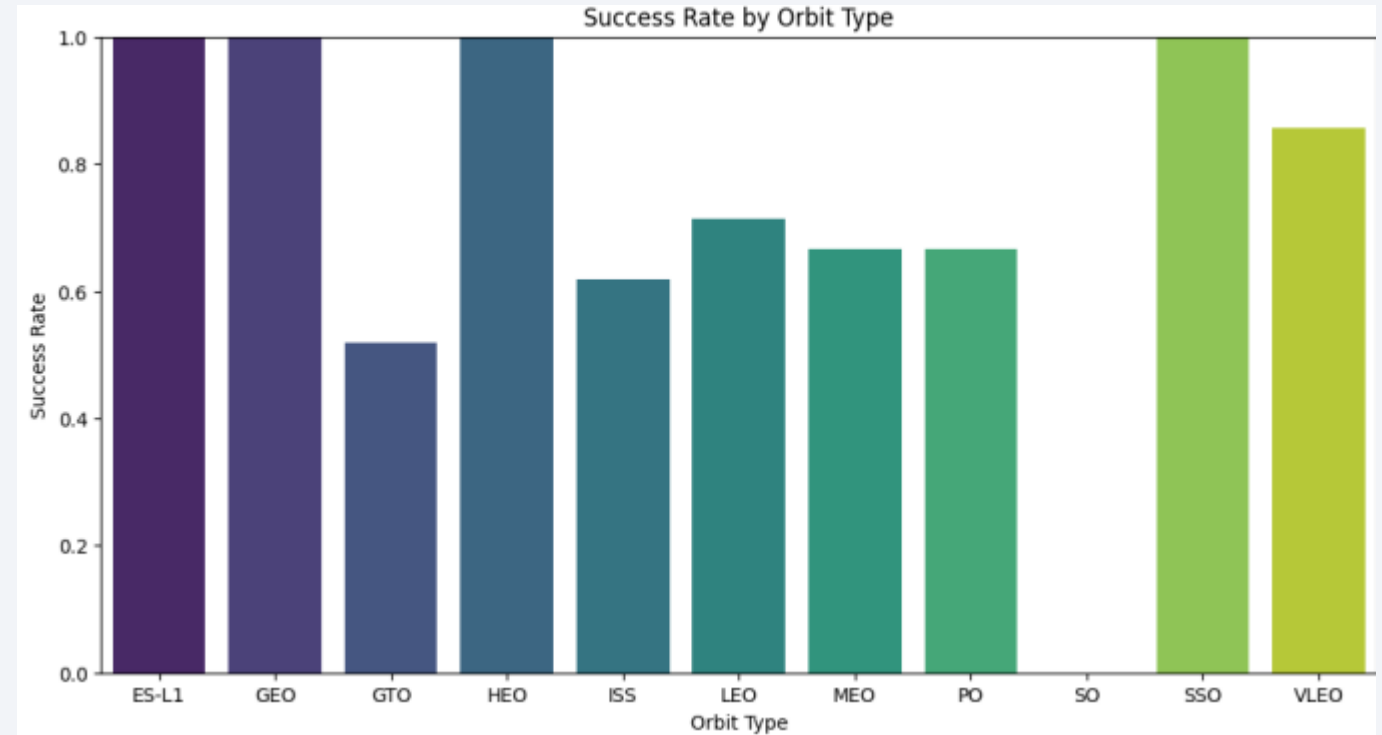
Payload vs. Launch Site

- This visualization reveals several key insights:
- **Payload Accommodation:** CCAFS SLC-40 handles a wide range of payload masses, maintaining a high frequency of launches.
- **Site-Specific Activity:** KSC LC-39A and VAFB SLC-4E also show significant activity, albeit with varied payload capacities.
- **Launch Success Across Payloads:** Successful launches (orange points) are spread across all payload ranges, suggesting that payload mass alone does not have a straightforward impact on launch success.
- **Handling Heavier Payloads:** The concentration of successful launches at higher payload masses indicates effective handling of heavier payloads at certain sites.



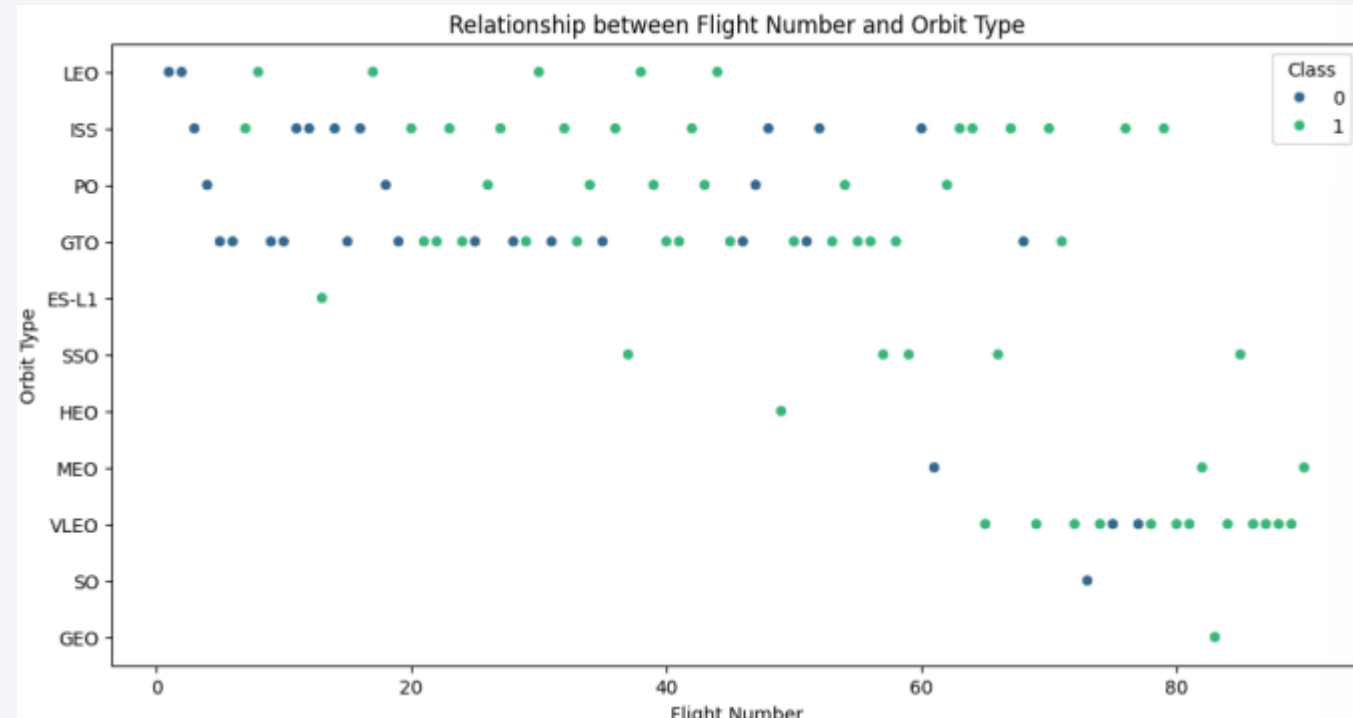
Success Rate vs. Orbit Type

- The chart reveals that orbits such as **ES-L1**, **SSO**, and **GEO** exhibit high success rates close to 1, indicating strong reliability for launches targeting these orbits. Conversely, orbits like **PO** and **GTO** show lower success rates, suggesting potential challenges or higher risks associated with missions aimed at these destinations.



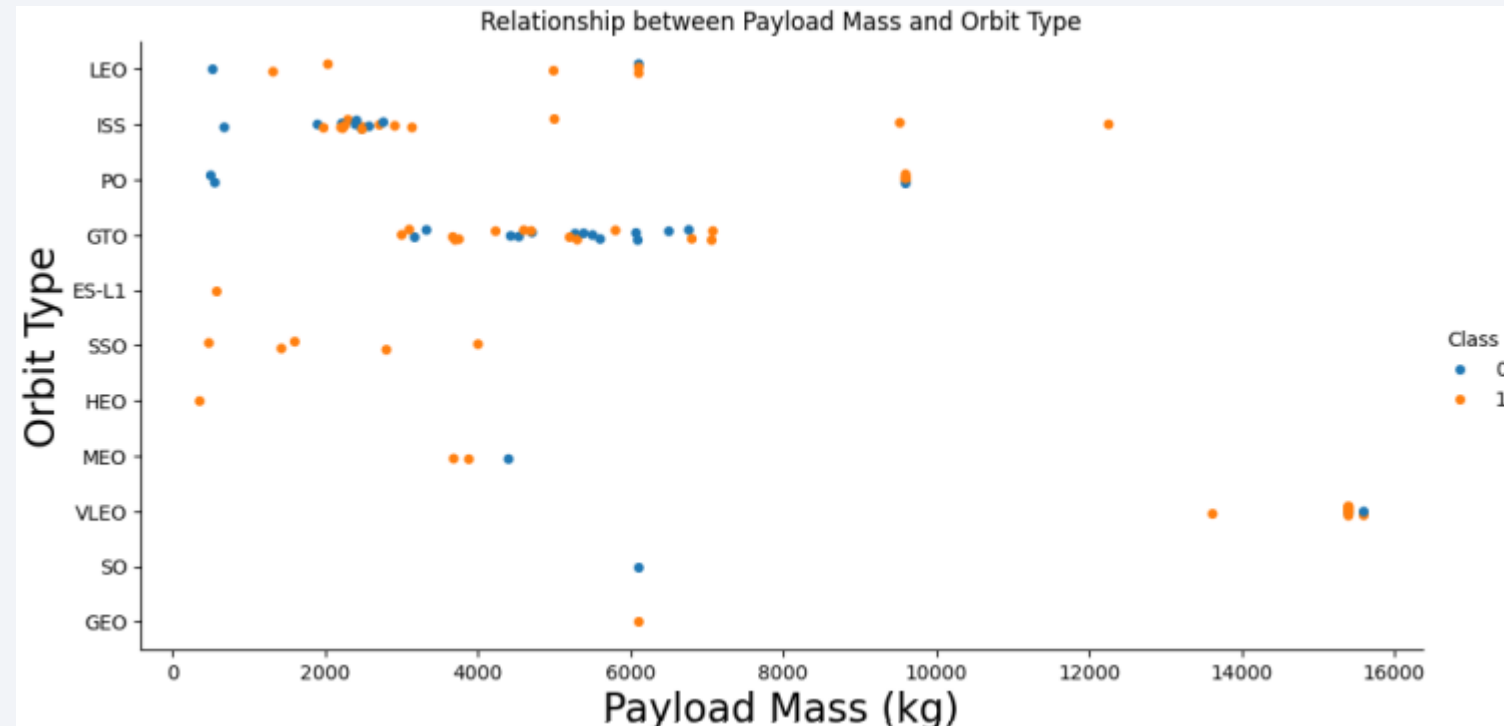
Flight Number vs. Orbit Type

- The plot shows that both successful and failed launches are spread across various flight numbers and orbit types, indicating no clear pattern that links flight number progression with increased success rates in specific orbits. However, certain orbits like **GTO** and **LEO** exhibit a higher concentration of launches, reflecting their frequent use.



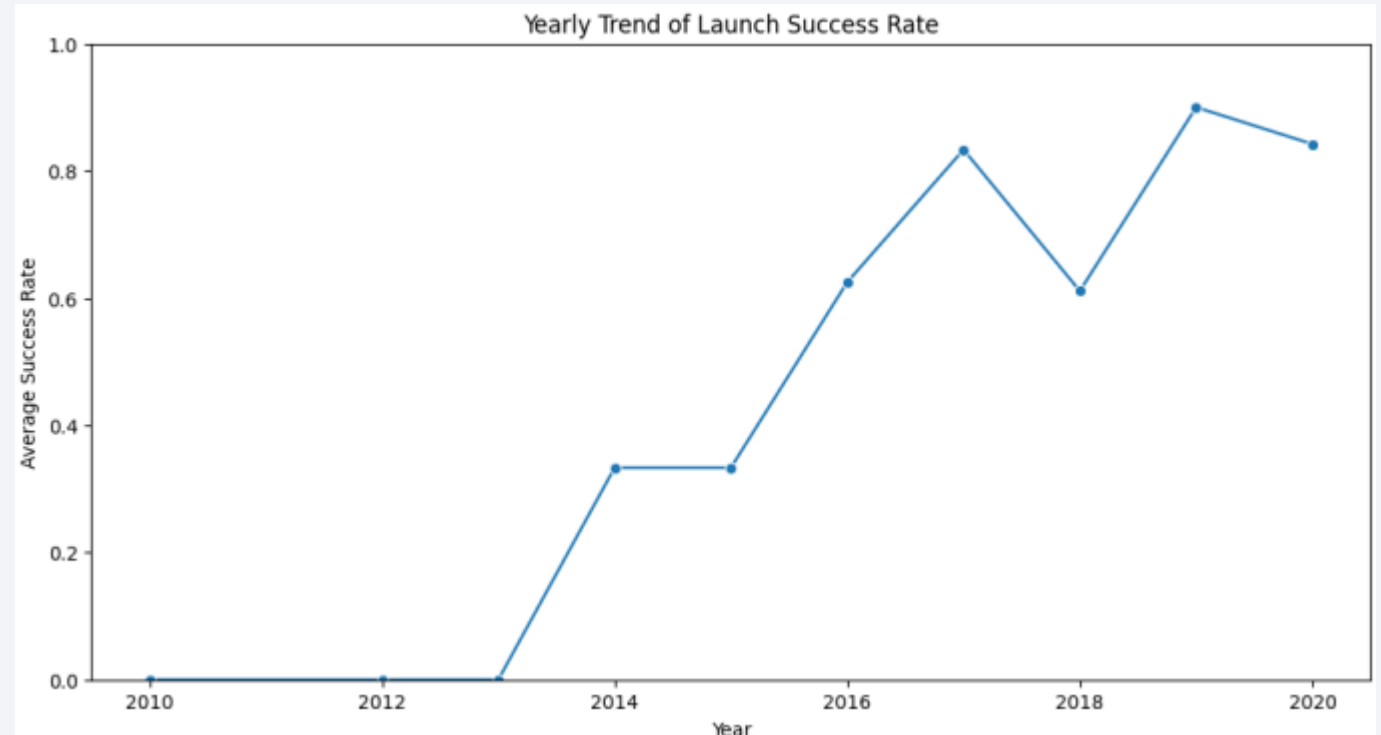
Payload vs. Orbit Type

- The plot reveals several key insights:
- **Orbit-Specific Patterns:** Certain orbit types, such as **GTO**, display a concentration of launches with varying payload masses, while orbits like **LEO** and **GEO** exhibit a broader distribution of payload masses.
- **Success Across Payload Ranges:** Successful launches (orange points) are observed across most orbit types, indicating that success is not strictly dependent on payload mass. However, some orbit types show a clustering of successful launches, suggesting that specific payload ranges might be more favorable for achieving successful missions.



Launch Success Yearly Trend

- The line chart illustrates the yearly trend of SpaceX launch success rates from 2010 to 2020. The x-axis represents the years, while the y-axis shows the average success rate for each year.
- **Trend Analysis:** The plot highlights a significant upward trend in launch success rates over the decade. Starting from a success rate of 0 in 2010, there is a notable increase beginning in 2014, followed by consistent growth with occasional fluctuations.
- **High Success Rates:** The years 2018 and 2020 stand out with particularly high success rates, nearing 1.0, indicating nearly perfect launch success.



All Launch Site Names

- This is the result:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with 'CCA'

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

Total_Payload_Mass

48213

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

Average_Payload_Mass
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

First_Successful_Landing

2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here

```
* sqlite:///my_data1.db
Done.
  Landing_Outcome  Count
Controlled (ocean)    5
Failure              3
Failure (drone ship)  5
Failure (parachute)   2
No attempt           21
No attempt            1
Precluded (drone ship) 1
Success              38
Success (drone ship)  14
Success (ground pad)  9
Uncontrolled (ocean)  2
```

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here

```
List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

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

* sqlite:///my_data1.db
Done.
Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7
```

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
* 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

- 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

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

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The background is a deep blue gradient.

Section 3

Launch Sites Proximities Analysis

Folium Map 1

- The screenshot displays a Folium map with markers indicating all SpaceX launch sites globally. Key elements include markers for **VAFB SLC-4E** on the west coast and multiple sites in Florida (**CCAFS** and **KSC**).
- This map provides a clear visual representation of the geographic distribution of SpaceX launch sites, highlighting their strategic locations across the United States for optimal launch coverage.



Folium Map 2

- The screenshot shows a Folium map with SpaceX launch sites marked and color-coded circles indicating launch outcomes: orange for successes and other colors for failures.
- This visual highlights the distribution and success rate of launches at each site, allowing for quick assessment and comparison.



Folium Map Screenshot 3

- The screenshot displays Vandenberg Space Launch Complex 4, highlighting its proximity to the coastline with a calculated distance. The map also marks nearby infrastructure, which is essential for understanding logistical and operational relationships.



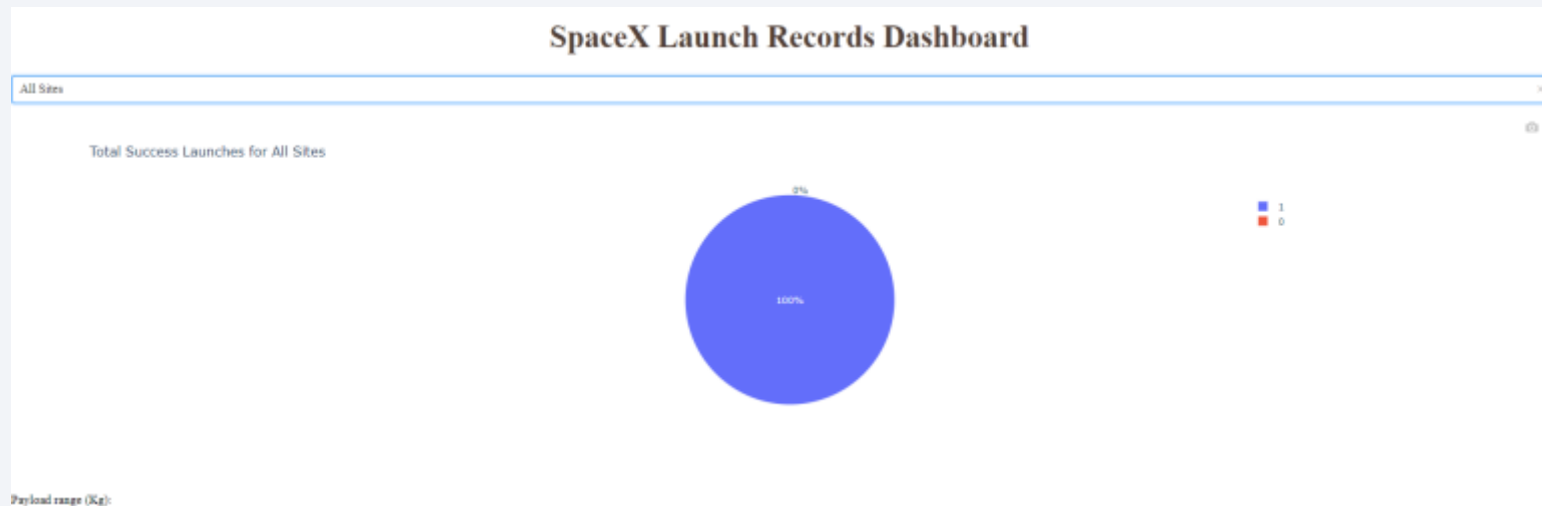


Section 4

Build a Dashboard with Plotly Dash

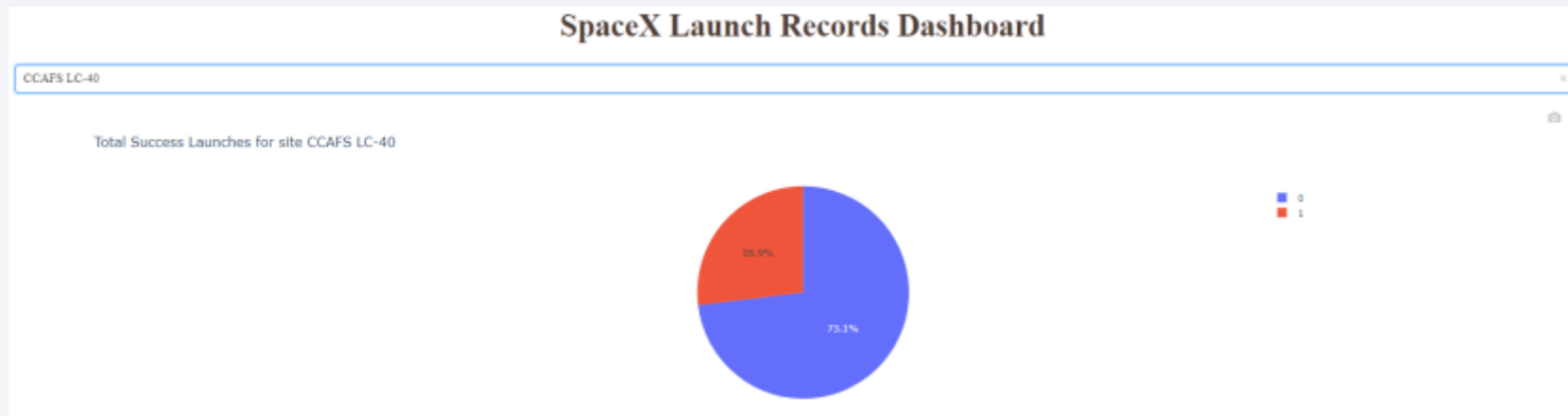
Dashboard 1

- The screenshot displays the "SpaceX Launch Records Dashboard" with a dropdown menu to select launch sites and a pie chart showing the total successful launches. The dropdown defaults to "All Sites," and the pie chart, with a legend, indicates 100% successful launches (blue).



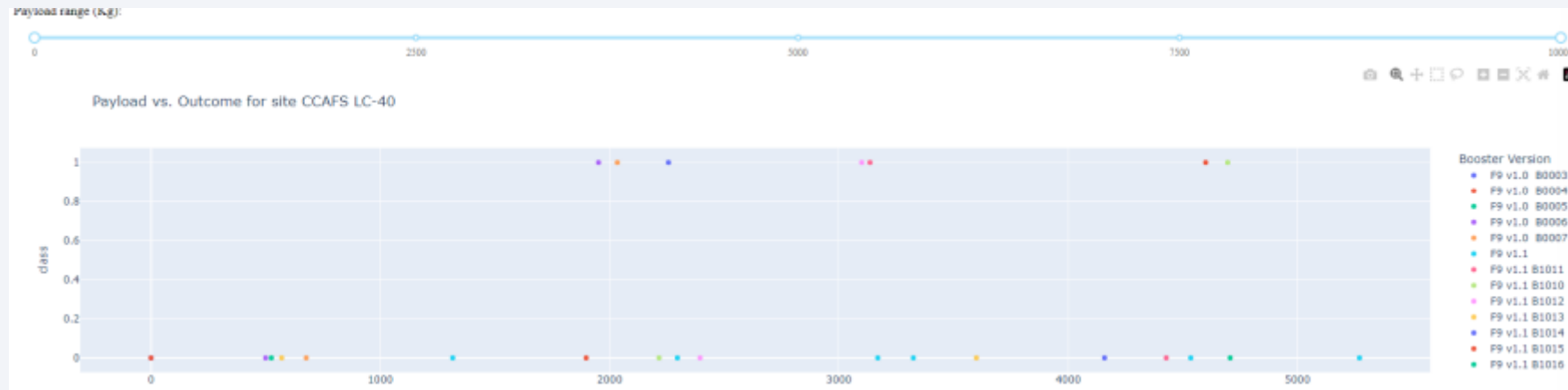
Dashboard 2

- The screenshot shows the "SpaceX Launch Records Dashboard" with the launch site "CCAFS LC-40" selected. The pie chart indicates that 73.1% of launches were successful (blue) and 26.9% failed (red) at this site. The dropdown menu allows users to select different launch sites, and the legend clarifies the success (1) and failure (0) rates.



Dashboard 3

- The screenshot displays a dashboard with a payload range slider and a scatter plot titled "Payload vs. Outcome for site CCAFS LC-40." The slider allows filtering by payload mass (0 to 10,000 kg), and the scatter plot shows the relationship between payload mass and launch outcomes, with different colors representing booster versions.
- Most successful launches (class 1) occur with payloads under 5,000 kg, while failures (class 0) are more spread out. The booster version **F9 v1.1 B1012** has a high success rate, whereas others like **F9 v1.0 B0004** have mixed outcomes, providing insights into which payload ranges and booster versions are most successful.

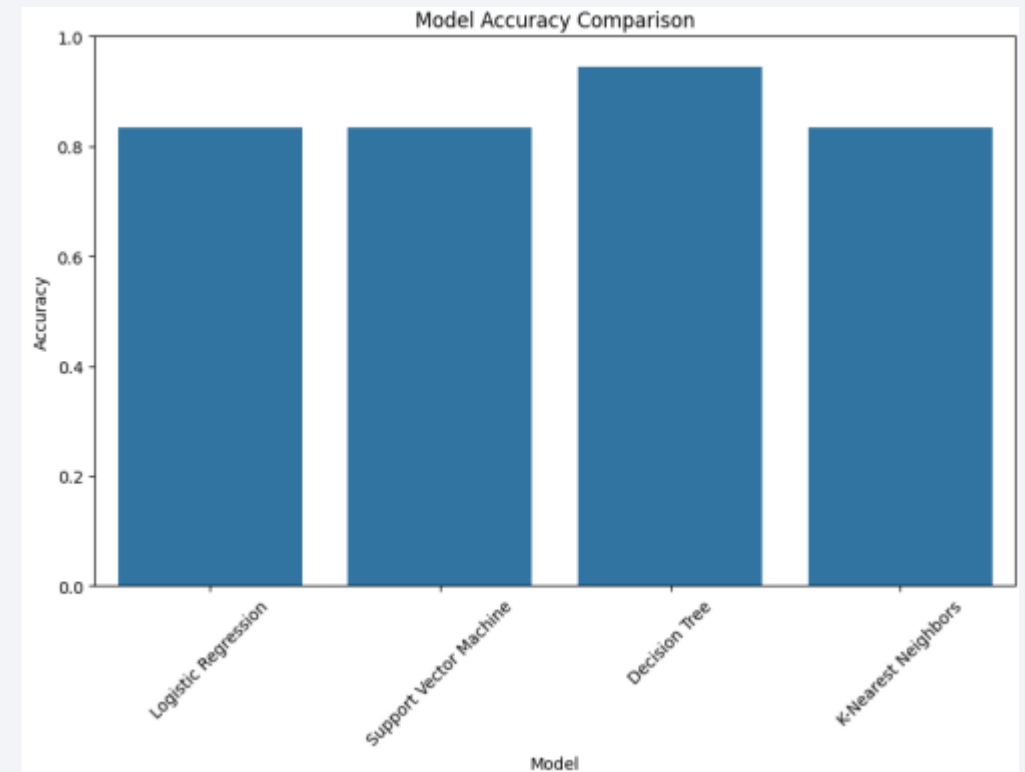


Section 5

Predictive Analysis (Classification)

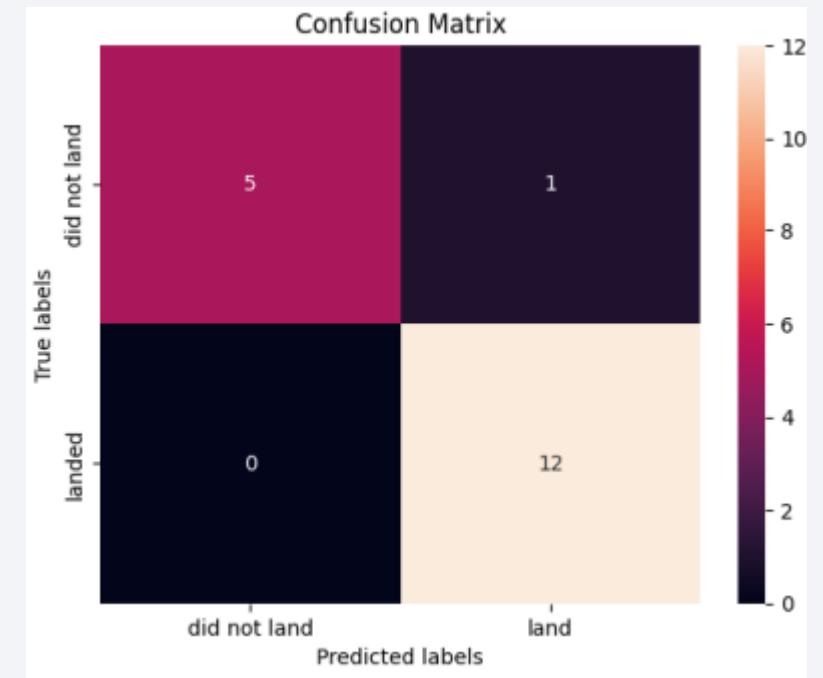
Classification Accuracy

- Based on the bar chart, the Decision Tree model has the highest classification accuracy among all the models evaluated. This indicates that, for the given dataset and parameters, the Decision Tree outperforms Logistic Regression, Support Vector Machine, and K-Nearest Neighbors in terms of correctly classifying the test data.



Confusion Matrix

- The confusion matrix for the best-performing model, Decision Tree, displays its classification performance. It correctly predicted 5 instances of "did not land" and 12 instances of "landed," with only one misclassification where it predicted "land" instead of "did not land."



Conclusions

- Feature Influence: The payload mass, orbit type, and other features significantly impact the landing success of a Falcon 9 launch.
- Model Performance: Among the machine learning algorithms tested, the Decision Tree model demonstrated the highest accuracy in predicting the landing outcome.、 、
- Cost Estimation: By accurately predicting the landing outcome, this project provides a valuable tool for estimating launch costs, aiding in budget planning and cost management for future missions.
- Data Utilization: The use of historical launch data has proven effective in training predictive models, highlighting the importance of data-driven decision-making in aerospace operations.

Thank you!

