IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Mixail Ota>
<February 2025>

Outline

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

Executive Summary

- Summary of methodologies

The methodologies include data collection from various sources, exploratory data analysis (EDA) to understand patterns and trends, and interactive visual analytics using dashboards for better insights. The process then moves to machine learning prediction, where different models such as logistic regression, support vector machines, decision trees, and k-nearest neighbors are tested and optimized using hyperparameter tuning. The final methodology involves evaluating model performance using accuracy scores and confusion matrices.

- Summary of all results

The analysis identified key patterns in the dataset, highlighting significant correlations and dependencies. The interactive visual analytics provided clear insights into data distributions. In the machine learning phase, models were trained and tested, with decision trees achieving the highest accuracy of approximately 87%. Logistic regression and SVM also performed well with accuracy scores around 83-84%. The study concluded that a well-tuned decision tree classifier yielded the best results for predicting Falcon 9 first-stage landings.

Introduction

Project Background and Context

This project analyzes Falcon 9 first-stage landings to predict success rates. SpaceX reduces costs by reusing rocket stages, so accurate predictions can aid cost estimation and competition. The approach includes data collection, EDA, interactive dashboards, and machine learning models.

Problems You Want to Find Answers

- What factors impact successful landings?
- How do launch conditions affect outcomes?
- Which model predicts landings best?
- How can predictions be optimized?
- Can insights help in cost estimation and competition?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

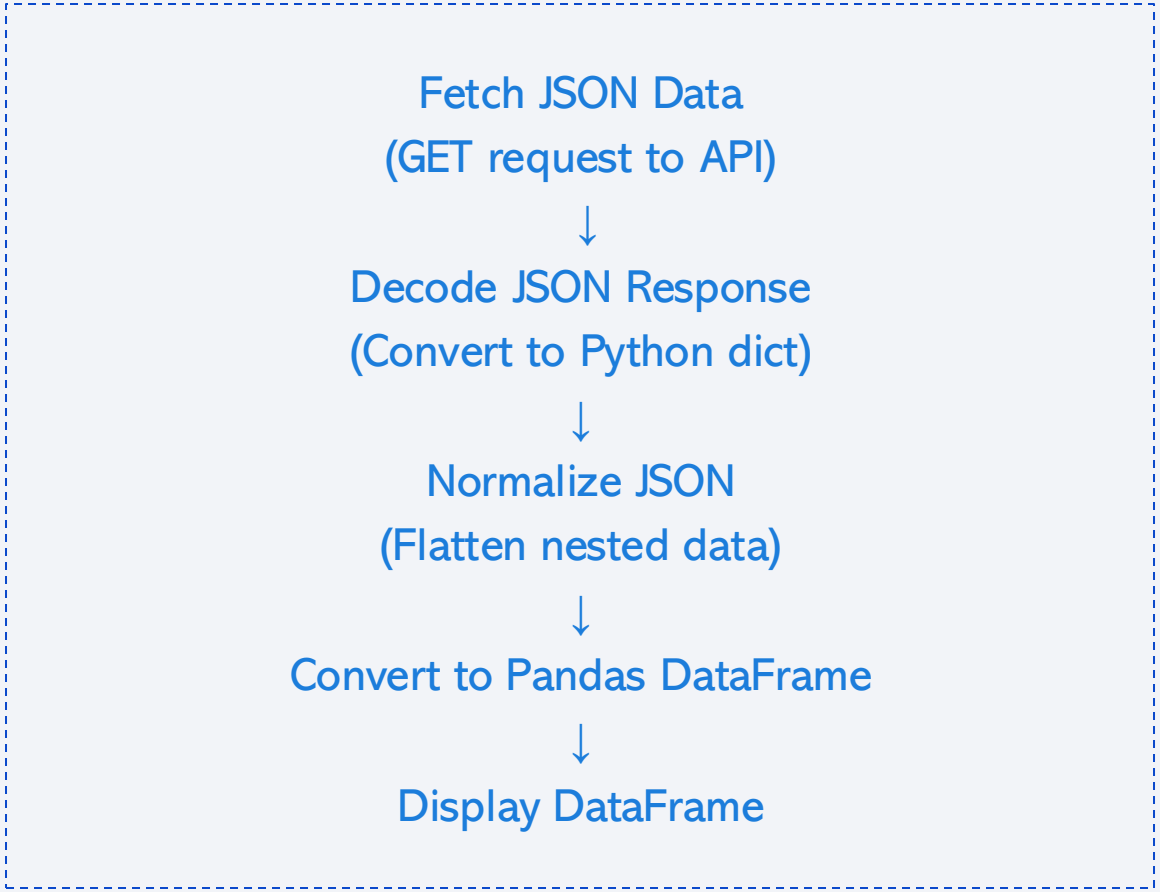
Data Collection Methods

- Data was collected using **GET requests** from the **SpaceX API**.
- The response was decoded as **JSON** and converted into a **pandas DataFrame** using .json_normalize().
- Data cleaning was performed to **check for missing values** and fill them where necessary.
- **Web scraping** was done using **BeautifulSoup** to extract **Falcon 9 launch records** from Wikipedia.
- The extracted **HTML table** was parsed and converted into a **pandas DataFrame** for further analysis.

Data Collection – SpaceX API

- The data was collected using SpaceX's REST API, which provides real-time and historical launch details. API calls were made to retrieve data on past Falcon 9 launches, including launch site, payload details, booster version, landing success, and orbit parameters.

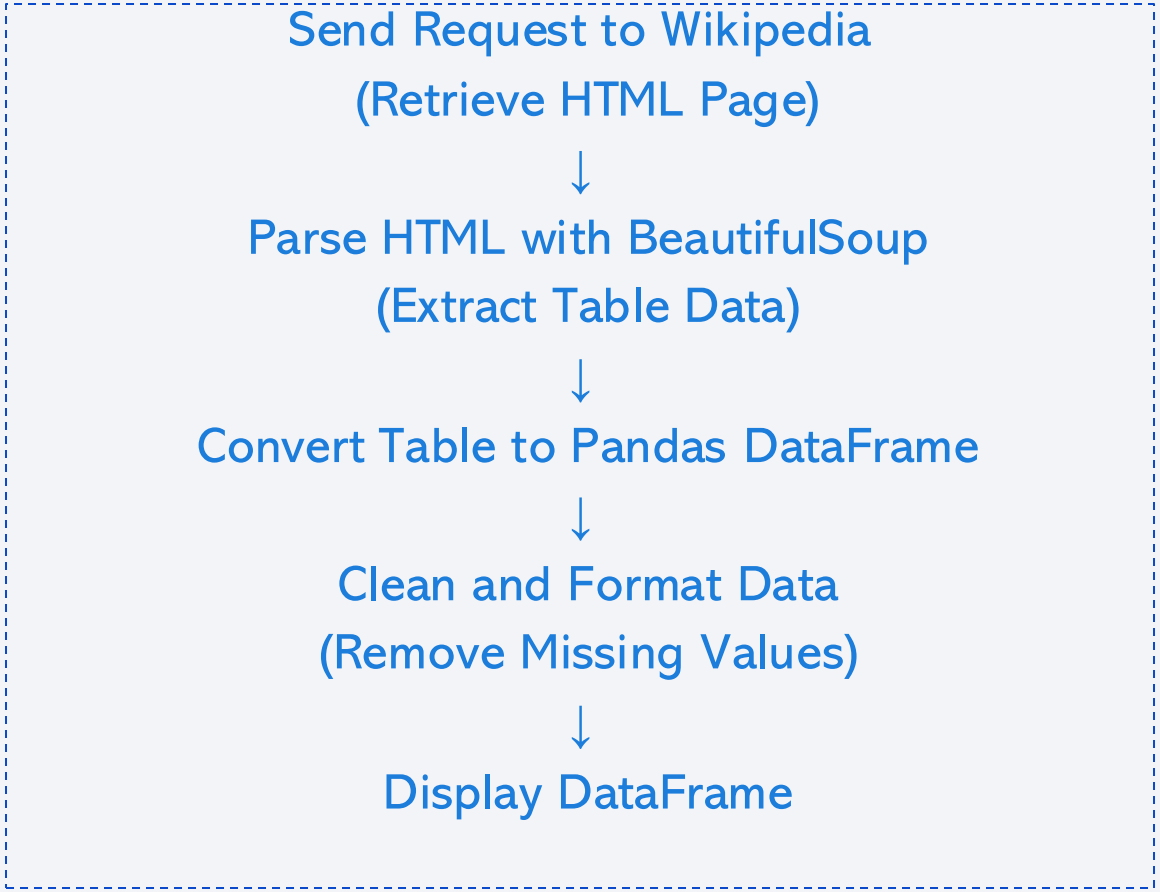
https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/01_Data-Collection.ipynb



Data Collection - Scraping

Web scraping was done using BeautifulSoup to extract Falcon 9 launch records from Wikipedia. The extracted HTML table was parsed and converted into a pandas DataFrame for further analysis.

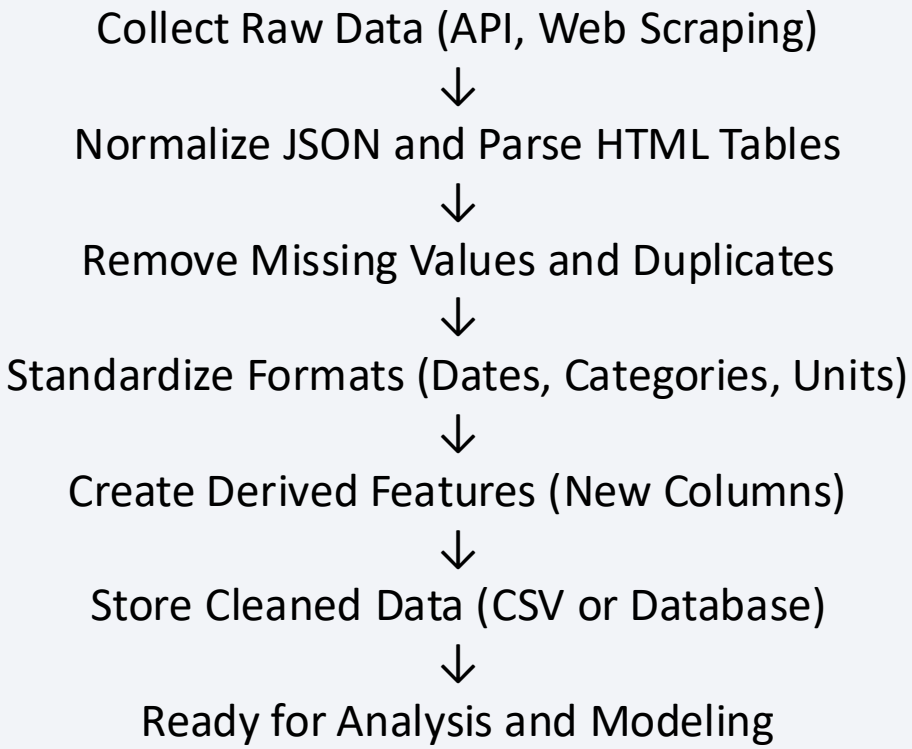
https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/01_Webscraping.ipynb



Data Wrangling

- Raw data from the SpaceX API and Wikipedia was collected in JSON and HTML formats. The JSON data was normalized into a structured pandas DataFrame, while the HTML tables were extracted and parsed using BeautifulSoup. Missing values were handled by filling or removing incomplete records.
- Duplicates were dropped, and formats like dates, categories, and numerical values were standardized. New columns were created for better analysis, and the cleaned data was stored in CSV files or a database, making it ready for exploratory analysis and machine learning.

https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/01_Data-Collection.ipynb



EDA with Data Visualization

- **Bar Charts** were used to compare categorical data, such as **success rates for different launch sites and orbit types**. This helped in identifying which locations or orbits had the highest probability of a successful launch.
- **Line Charts** were used to observe **changes in launch success rates over time**. This visualization allowed for trend analysis, showing how launch performance evolved over the years.

https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/02_Exploratory-Data-Analysis.ipynb

EDA with SQL

Summary of SQL Queries in EDA

- **Explored booster performance** by listing booster versions that carried the maximum payload mass.
- **Filtered failed landings** on drone ships for the year 2015 to analyze unsuccessful missions.
- **Ranked landing outcomes** between 2010 and 2017 based on frequency.
- **Counted total successful and failed missions** to assess overall success rates.
- **Retrieved unique launch sites and booster versions** for further categorization.
- **Calculated average and total payload mass** to understand payload distribution.
- **Identified the first successful ground pad landing** to track early mission successes.
- **Created new tables** with cleaned data for further structured analysis.

https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/02_Exploratory-Data-Analysis.ipynb

Build an Interactive Map with Folium

- **Markers** were added to indicate specific **launch site locations** on the map.
- **Circles** were used to highlight **launch sites**, with varying radii representing different areas of interest.
- **Lines** were drawn to visualize **connections between launch sites and impact locations**, helping to analyze distances and trajectories.

https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/03_Interactive-Visual-Analytics-and-Dashboards.ipynb

Build a Dashboard with Plotly Dash

Summary of Dashboard Plots and Interactions

- **Pie Chart** was used to show the **proportion of successful vs. failed launches** for each site, helping to compare success rates visually.
- **Dropdown Menu** was added to allow users to **filter launch data by site**, making it easier to focus on specific locations.
- **Slider** was implemented to dynamically **adjust the payload mass range**, enabling users to analyze how payload weight impacts launch success.

https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Logistic Regression was trained as a **baseline classifier** to evaluate initial performance.

Support Vector Machine (SVM) was tested with different **hyperparameters** to improve classification accuracy.

Decision Tree was built to **capture complex decision boundaries** for better classification.

Data Preprocessing
(Cleaning & Standardization)
↓
Train-Test Split
(Dividing Data for Evaluation)
↓
Model Training
(Logistic Regression, SVM, Decision Tree)
↓
Hyperparameter Tuning
(Optimizing Performance)
↓
Model Evaluation
(Accuracy, Confusion Matrix)
↓
Best Model Selection
(Final Classifier Chosen

https://github.com/mishaneta/IBM-Data-Science-Specialization/blob/main/04_Machine-Learning-Prediction-Part-5-v1.ipynb

Results

Exploratory Data Analysis Results

Launch success rates were analyzed using **bar charts** for different launch sites and orbit types.

•**Temporal trends** in launch outcomes were examined using **line charts** to identify changes over time.

•**Geospatial analysis** mapped **launch site locations and impact zones** for spatial insights.

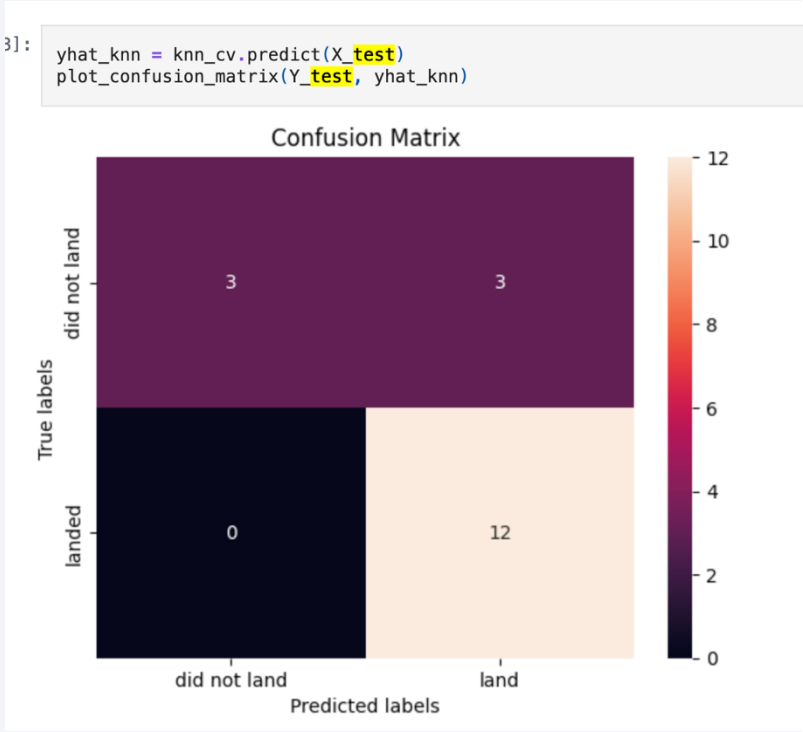
Predictive Analysis Results

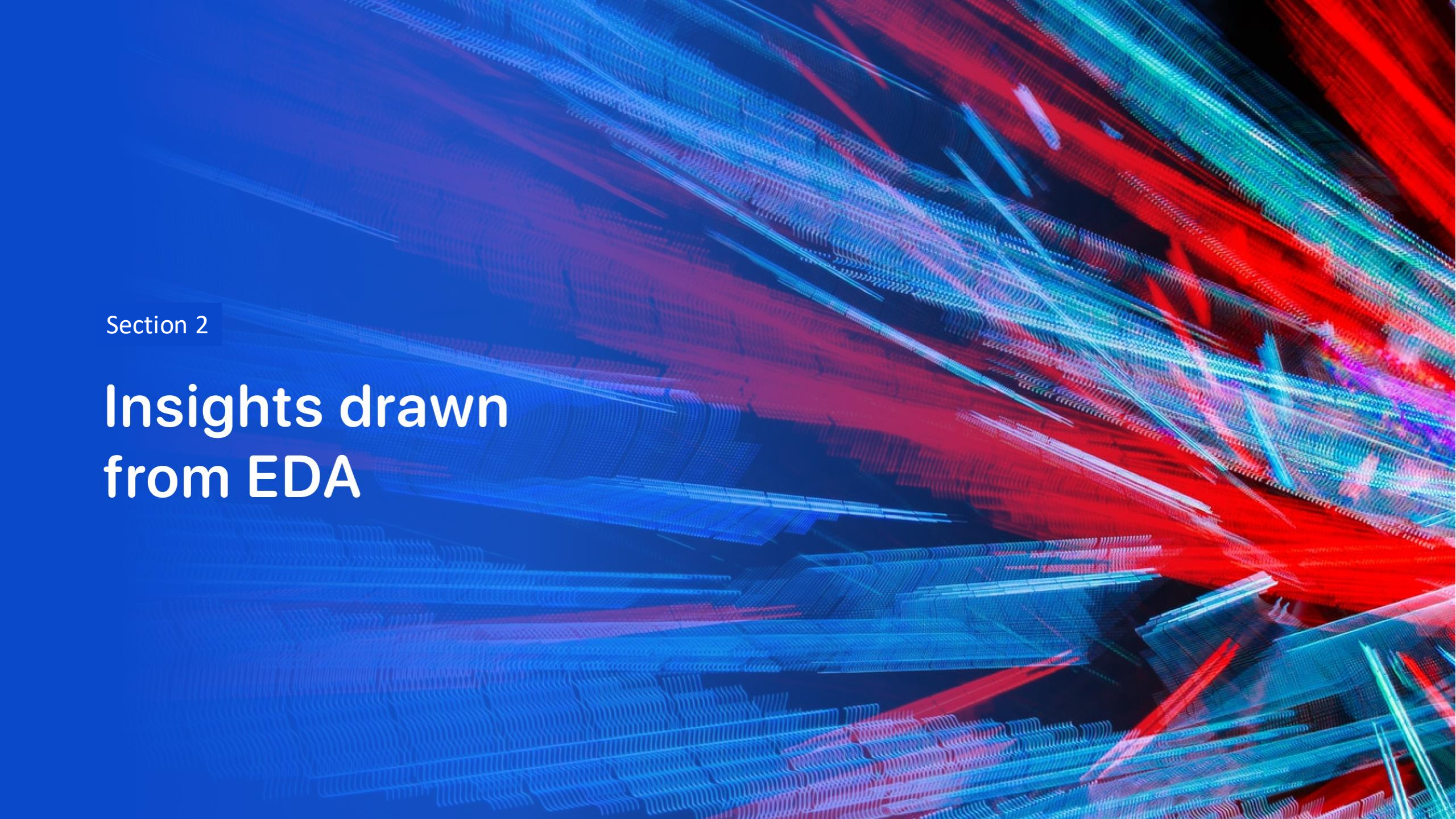
•**Logistic Regression** was used as a **baseline model** for predicting successful launches.

•**Support Vector Machine (SVM)** was trained with **hyperparameter tuning** to improve accuracy.

•**Decision Tree models** were built to **capture complex relationships** in the data.

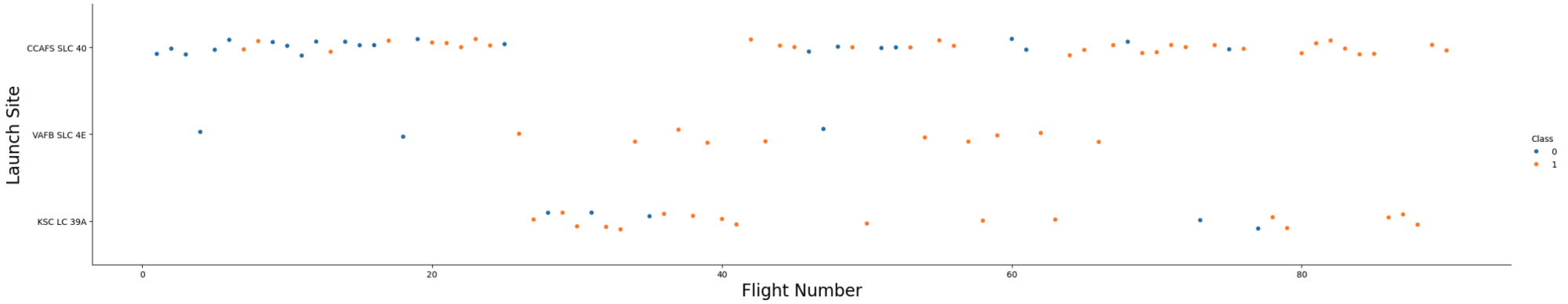
These models were compared using **accuracy scores and confusion matrices** to determine the best classifier for predicting launch success.




Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

```
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site", fontsize=20)
plt.show()
```

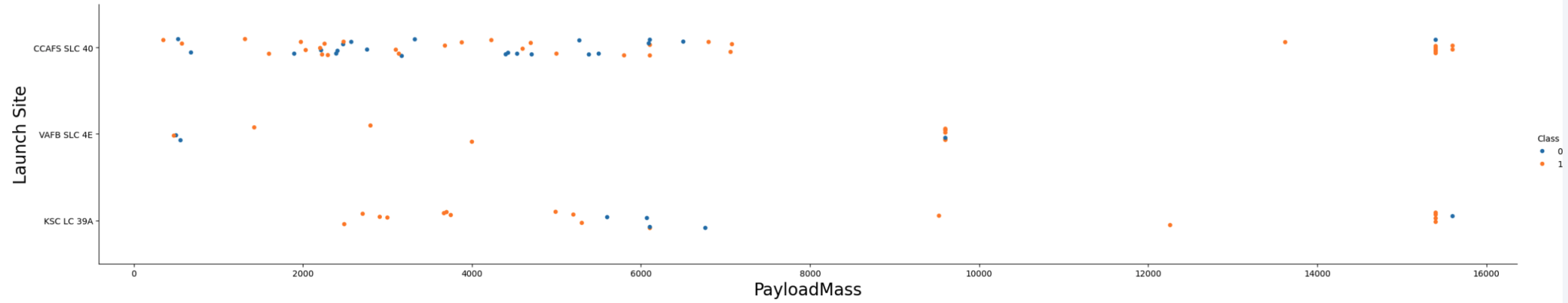


The plot shows that **launch success rates have improved over time**, with **early missions experiencing more failures** compared to later ones. CCAFS SLC 40 and KSC LC 39A have the **highest number of launches**, with KSC LC 39A showing a **higher success rate overall**. VAFB SLC 4E has **fewer launches**, suggesting it is used for specific missions. The trend indicates that **SpaceX has optimized its launch processes over time**, leading to a **higher probability of successful landings** at certain sites.

Payload vs. Launch Site

- The plot indicates that **successful launches (orange) occur across various payload masses**, but certain patterns emerge. **KSC LC 39A** appears to handle **higher payloads (above 10,000 kg)** with a relatively **higher success rate**. **CCAFS SLC 40**, which has the most launches, sees **a mix of successes and failures across different payload ranges**. **VAFB SLC 4E** has fewer launches, with most payloads being on the lower end. This suggests that **some launch sites may be optimized for heavier payloads**, while others handle a broader range of missions with varying success rates. 🚀

```
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("PayloadMass",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



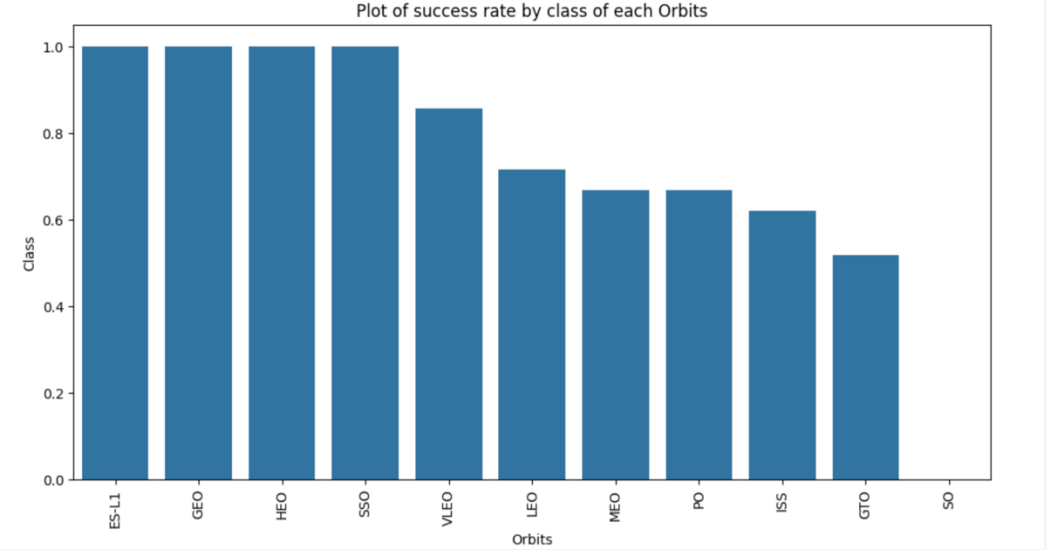
Success Rate vs. Orbit Type

The bar chart shows that **success rates vary across different orbit types**. **ES-L1, GEO, HEO, and SSO** have **the highest success rates (close to 100%)**, indicating that missions targeting these orbits are more reliable. In contrast, **GTO and SO have the lowest success rates**, suggesting that launches to these orbits face greater challenges.

LEO, MEO, PO, and ISS orbits have moderate success rates, implying **some variability in mission outcomes**.

```
grouped_orbits = df.groupby(by=['Orbit'])['Class'].mean().sort_values(ascending=False).reset_index()
fig, ax=plt.subplots(figsize=(12,6))
ax = sns.barplot(x = 'Orbit', y = 'Class', data=grouped_orbits)
ax.set_title('Plot of success rate by class of each Orbits', fontdict={'size':12})
ax.set_ylabel('Class', fontsize = 10)
ax.set_xlabel('Orbits', fontsize = 10)
ax.set_xticklabels(ax.get_xticklabels(), fontsize = 10, rotation=90);
```

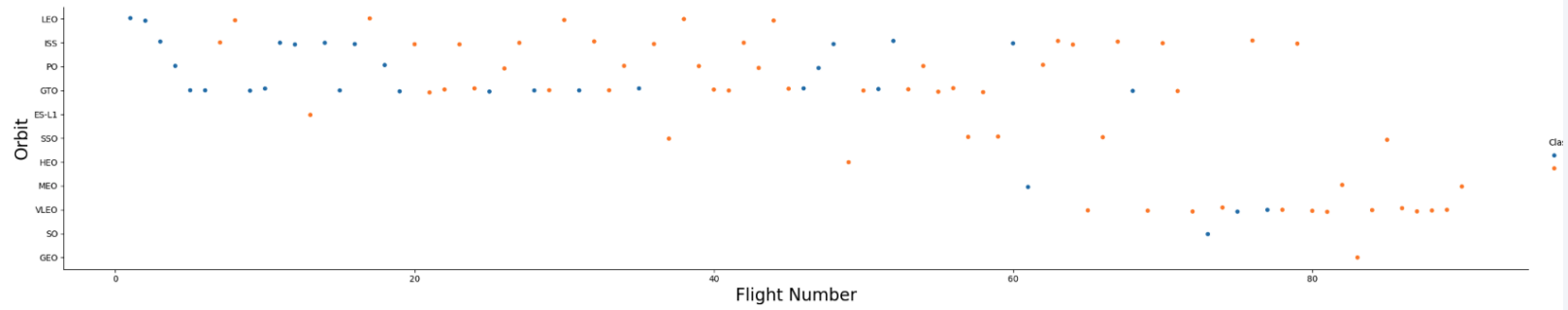
```
/var/folders/24/vx1b85kj3mvcthgs93rkvsvc0000gn/T/ipykernel_57584/1611272804.py:7: UserWarning: set_ticklabels() should only be used v
i.e. after set_ticks() or using a FixedLocator.
  ax.set_xticklabels(ax.get_xticklabels(), fontsize = 10, rotation=90);
```



Flight Number vs. Orbit Type

The plot shows the **relationship between Flight Number and Orbit Type**, with successful launches (orange) and failures (blue) distributed across different orbits. **Early flights had more failures**, particularly in **LEO, GTO, and ISS orbits**, indicating that **initial missions faced more challenges**. Over time, the success rate improved, especially for **higher-altitude orbits like GEO, HEO, and SSO**, suggesting **better mission optimization**. The variation in success rates across orbits implies that **some orbit types are more difficult to reach, requiring technological advancements for consistent success**.

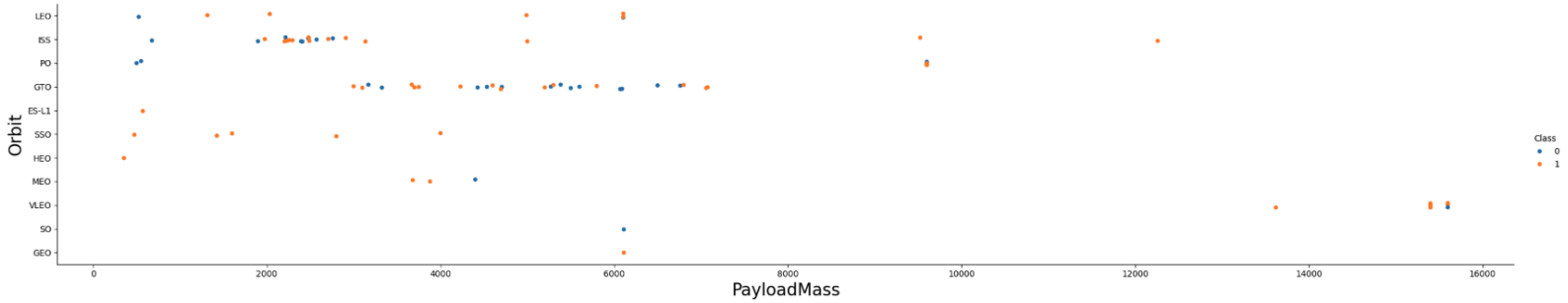
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the c
sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



Payload vs. Orbit Type

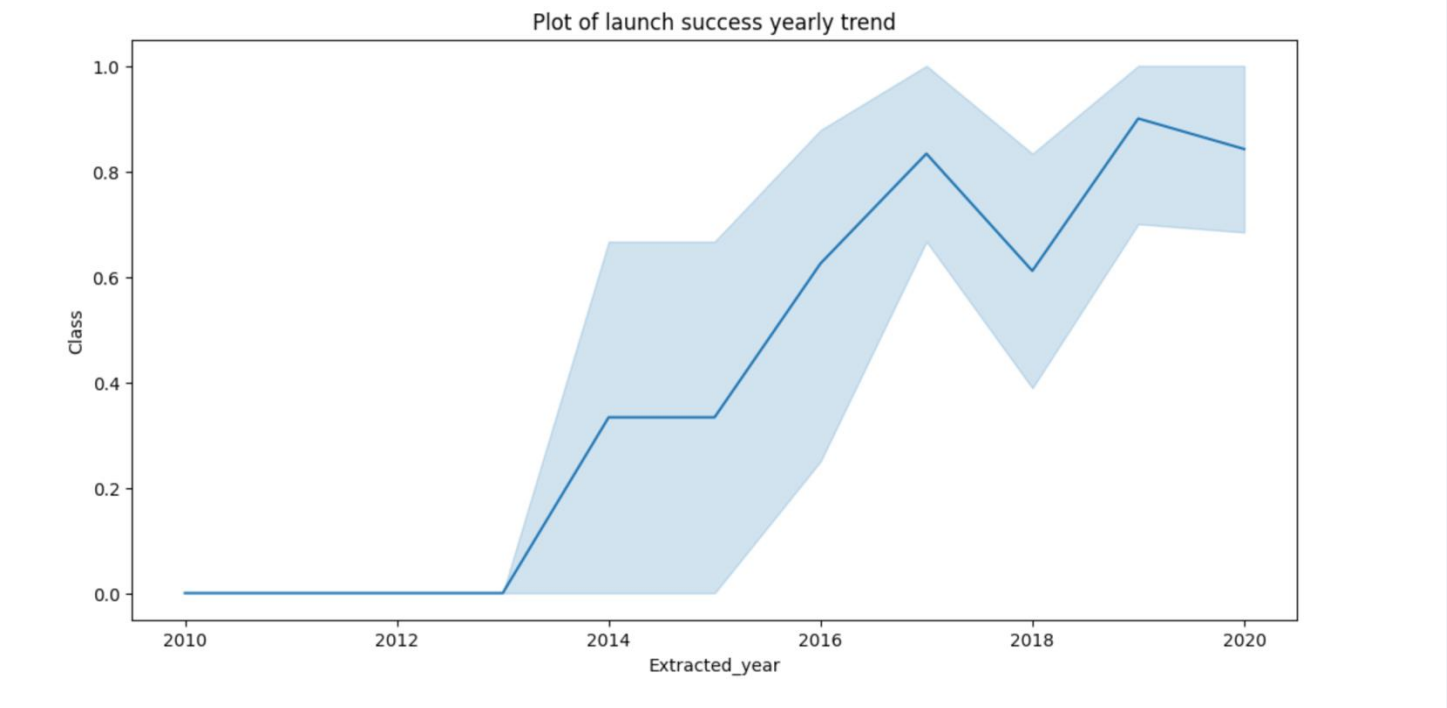
The plot suggests that **certain orbits, like GTO and GEO, handle heavier payloads**, as seen with payloads above **10,000 kg**. However, **there are fewer data points for these higher payloads**, meaning the trend is based on limited observations rather than extensive testing. **LEO and ISS have more launches with a wide range of payloads**, but failures still occur at different payload levels. This indicates that while **some orbits seem better suited for heavier payloads, more launches would be needed to confirm this pattern with confidence**.

```
sns.catplot(y="Orbit", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("PayloadMass",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



Launch Success Yearly Trend

The line chart shows a **clear upward trend in launch success rates over time**, indicating **significant improvements in SpaceX's mission reliability**. From **2013 onward**, success rates steadily increased, with some fluctuations. The **dip around 2018** suggests occasional setbacks, but the overall trajectory remains positive. The shaded region represents **confidence intervals**, showing variability in success rates. This trend highlights **technological advancements and improved operational efficiency**, leading to **higher mission success rates over the years**.



All Launch Site Names

This SQL query retrieves a **list of unique launch sites** from the SPACEXTBL table.

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

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

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

Launch Site Names Begin with 'CCA'

This SQL query retrieves launch records from the SPACEXTBL table, filtering for launch sites that start with "CCA" and limiting the output to 5 rows.

```
[4]: %sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
 * sqlite:///my_data1.db
Done.
```

[4]:

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

```
[5]: %sql SELECT SUM("Payload_Mass__kg_") AS Total_Payload_Mass FROM SPACEXTBL;
```

Total Payload Mass

This SQL query calculates the total payload mass from the `SPACEXTBL` table by summing all values in the `"Payload_Mass__kg_"` column and returns it as `"Total_Payload_Mass"`. 🚀

```
5]: %sql SELECT SUM("Payload_Mass__kg_") AS Total_Payload_Mass FROM SPACEXTBL;
 * sqlite:///my_data1.db
Done.
5]:
```

Total_Payload_Mass
619967

Average Payload Mass by F9 v1.1

This SQL query calculates the **average payload mass** for launches using the **"F9 v1.1"** booster version by averaging the values in the "Payload_Mass__kg_" column from the SPACEXTBL table.

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

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

Average_Payload_Mass
2928.4

First Successful Ground Landing Date

This SQL query retrieves the **earliest (first) successful landing date** on a **ground pad** by selecting the **minimum date** (MIN("Date")) from the SPACEXTBL table where the "Landing_Outcome" is **'Success (ground pad)'**.

```
[17]: %sql SELECT MIN("Date") AS First_Successful_Landing FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (ground pad)';
 * sqlite:///my_data1.db
Done.
[17]:
```

First_Successful_Landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

This SQL query retrieves a **list of unique booster versions** that had a **successful landing on a drone ship** and carried a **payload mass between 4000 kg and 6000 kg**. It filters the SPACEXTBL table based on **landing outcome and payload range** to identify which booster versions met these criteria.

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

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

[18]:

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

Total Number of Successful and Failure Mission Outcomes

This SQL query **counts the total number of successful missions** by selecting the number of rows where "Mission_Outcome" is 'Success' in the SPACEXTBL table.

```
[19]: #List the total number of successful and failure mission outcomes
      %sql SELECT COUNT("Mission_Outcome") FROM SPACEXTBL WHERE "Mission_Outcome" = 'Success'

       * sqlite:///my_data1.db
      Done.
[19]: COUNT("Mission_Outcome")
                             98
```

Boosters Carried Maximum Payload

This SQL query retrieves the names of **booster versions** that carried the **maximum payload mass**. It uses a **subquery** to first find the highest payload mass in the "Payload_Mass__kg_" column, then selects the **distinct booster versions** associated with that maximum value from the SPACEXTBL table.

```
[20]: # List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
      %sql SELECT DISTINCT "Booster_Version" FROM SPACEXTBL WHERE "Payload_Mass__kg_" = (SELECT MAX("Payload_Mass__kg_") FROM SPACEXTBL);
```

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

[20]:

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

2015 Launch Records

This SQL query retrieves records of **failed landings on a drone ship** in the year **2015**. It extracts the **month** from the "Date" column using strftime('%m', "Date"), and selects the **landing outcome, booster version, and launch site** from the SPACEXTBL table. The filter conditions ensure that only records where "Landing_Outcome" is 'Failure (drone ship)' and the year is **2015** are included.

[21]:
```
# 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.
%sql SELECT strftime('%m', "Date") AS Month, "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTBL WHERE "Landing_Outcome" LIKE 'Failure (drone ship)'
```

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

[21]:

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

This SQL query ranks the **count of landing outcomes** (e.g., "Failure (drone ship)", "Success (ground pad)") within the date range **2010-06-04 to 2017-03-20**. It groups the data by "Landing_Outcome", counts occurrences of each outcome, and assigns a **rank** based on the count in **descending order** using the RANK() function.

```
# Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.¶
%sql SELECT "Landing_Outcome", COUNT(*) AS Outcome_Count, RANK() OVER (ORDER BY COUNT(*) DESC) AS Rank FROM SPACEXTBL WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
```

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

Landing_Outcome	Outcome_Count	Rank
No attempt	10	1
Success (drone ship)	5	2
Failure (drone ship)	5	2
Success (ground pad)	3	4
Controlled (ocean)	3	4
Uncontrolled (ocean)	2	6
Failure (parachute)	2	6
Precluded (drone ship)	1	8

Section 3

Launch Sites Proximities Analysis

Launch Sites Locations Analysis with Folium

This map visualizes the geographic distribution of SpaceX launch sites across the United States using clustered markers and labels.

Key Elements:

1. Clustered Markers:
   - The numbers (e.g., 10 near Los Angeles, 46 near Florida) indicate the number of launches in those regions.
   - The clustering helps in grouping nearby launch sites for better visualization.
2. Major Launch Locations:
   - Florida (Cape Canaveral / Kennedy Space Center): The most frequently used launch site (46 launches).
   - Vandenberg Space Force Base (California): A secondary launch hub (10 launches).
   - NASA Johnson Space Center (Texas): Marked but not a primary launch site, serving more as a mission control center.

Findings:

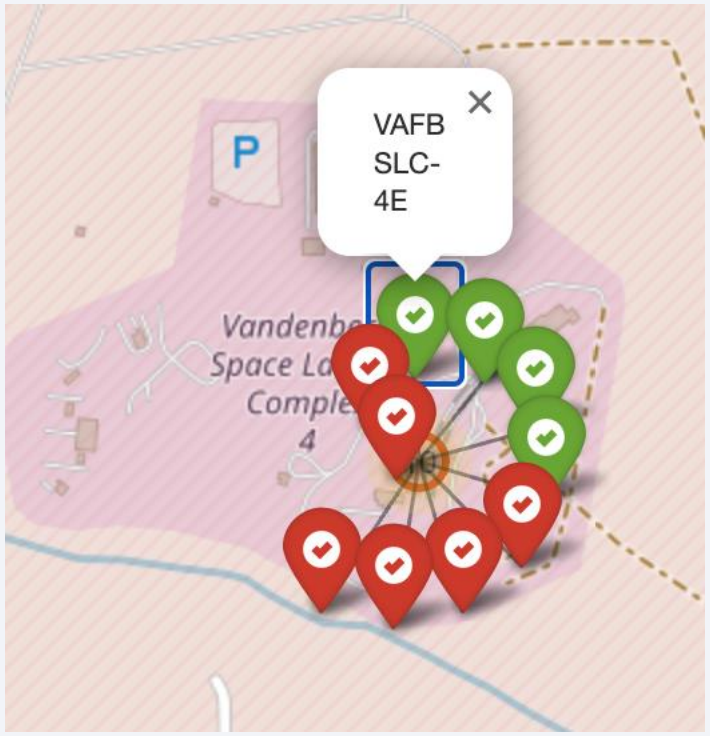
Most launches occur in Florida, confirming that Cape Canaveral is the primary launch site due to its proximity to the equator, allowing for efficient orbital launches.

California (Vandenberg) is used for polar and sun-synchronous orbits, making it ideal for Earth observation and military missions.

The NASA JSC marker in Texas is notable but not an actual launch site, highlighting its role in mission control rather than launches.

This map provides a clear overview of where SpaceX conducts most of its operations and how launch frequency varies by location.

Mark the success/failed launches for each site on the map



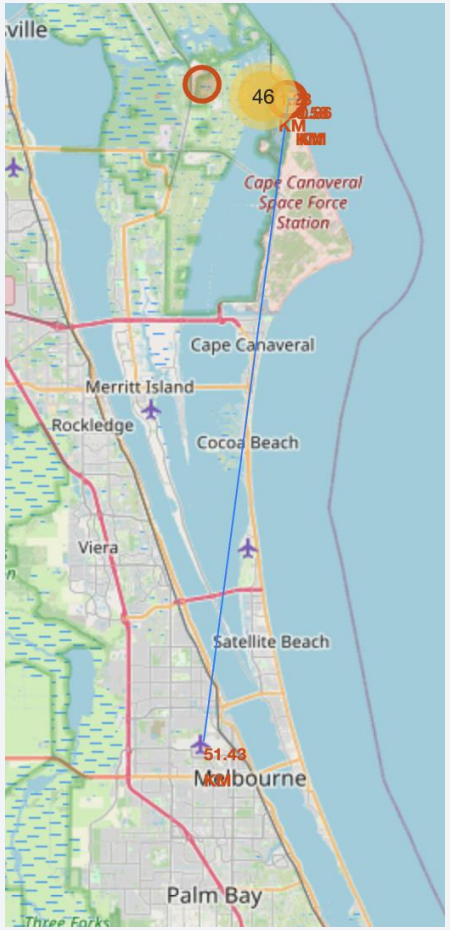
Calculate the distances between a launch site to its proximities

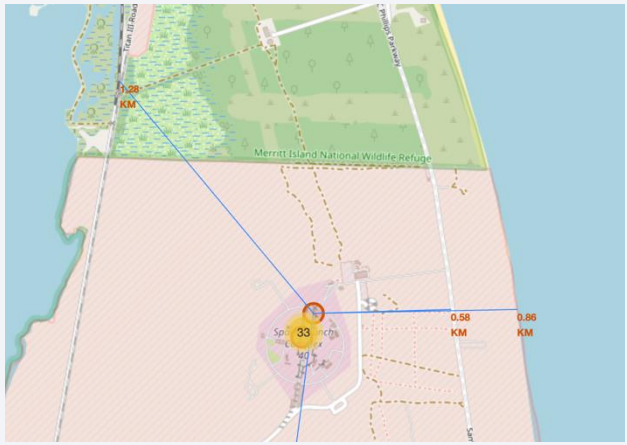
Analysis of Launch Site (Cape Canaveral or Kennedy Space) Proximity

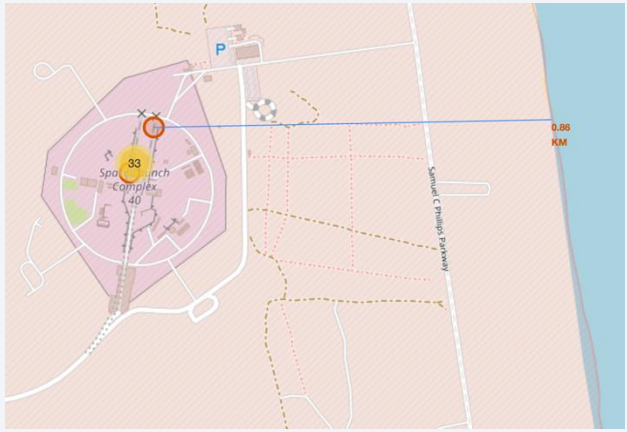
The launch site is **strategically located** near key infrastructure:

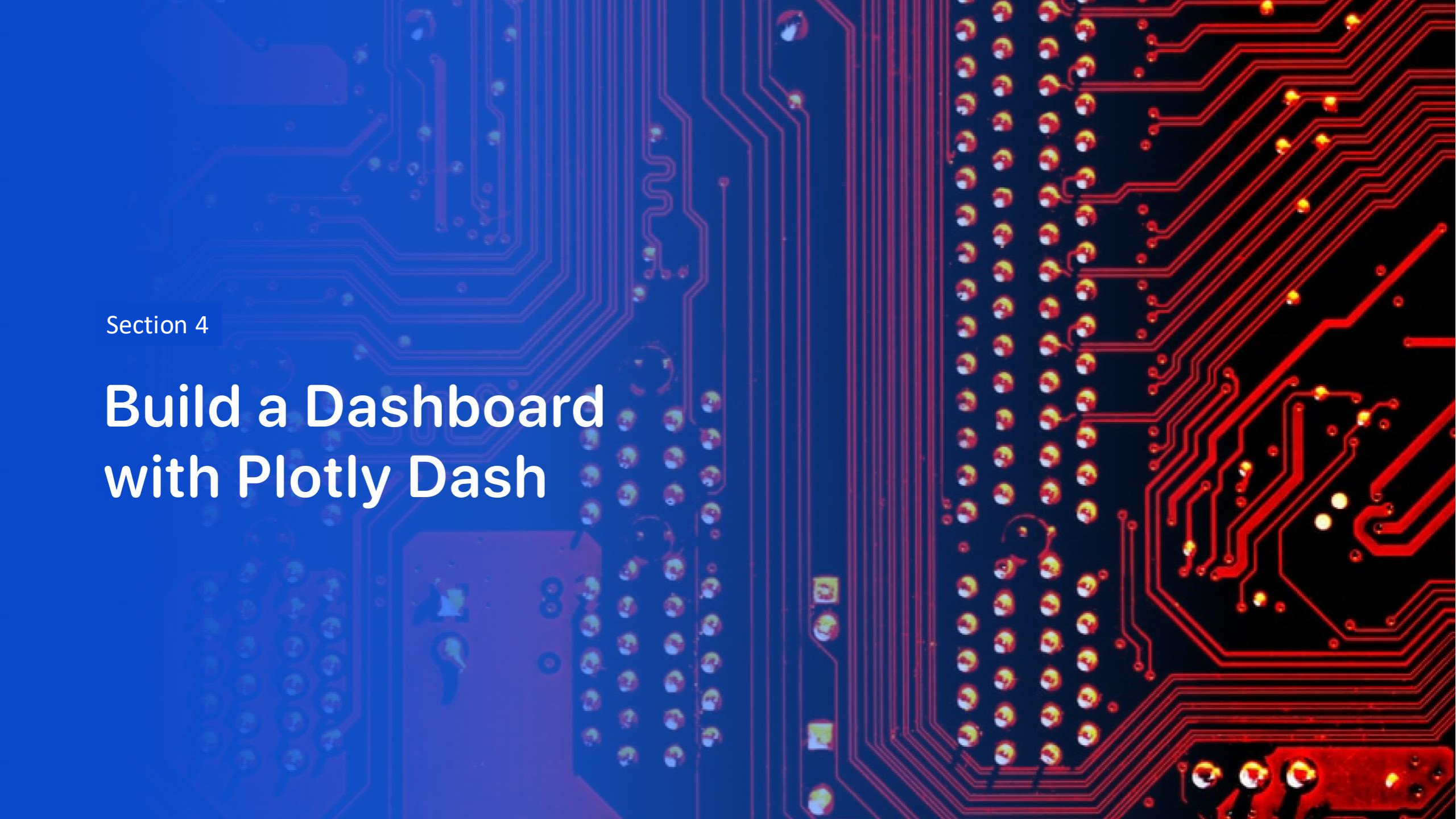
- **Coastline (~0.86 KM):** Ensures safe overwater launches, minimizing risk to populated areas.
- **Highways (~0.58–1.28 KM):** Allows efficient transport of equipment and personnel.
- **Railway (visible in some images):** Supports heavy payload and rocket part transportation.

This setup **optimizes logistics, safety, and operational efficiency**, making the site ideal for frequent launches. 🚀








Section 4
Build a Dashboard with Plotly Dash

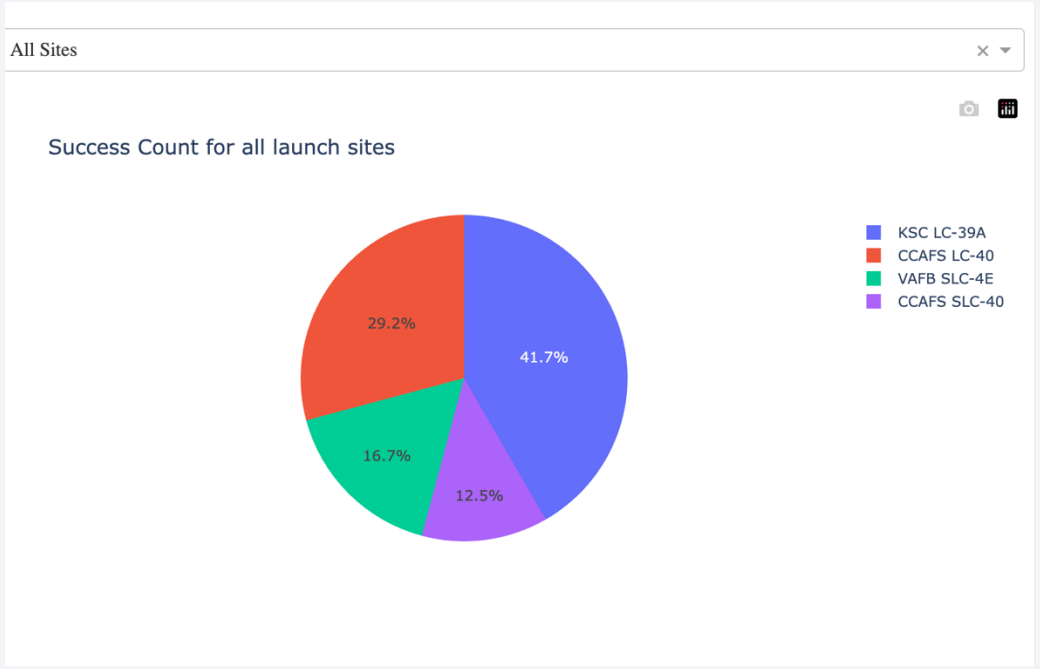
Success count for all launch sites

Analysis of Launch Success Count for All Sites

- **Key Elements:**
- **Pie Chart Representation:** The chart visualizes the **success count distribution across different launch sites**.
- **Color Legend:** Each launch site is represented with a unique color to distinguish its contribution.
- **Percentage Labels:** Each slice indicates the **proportion of successful launches** from the respective site.

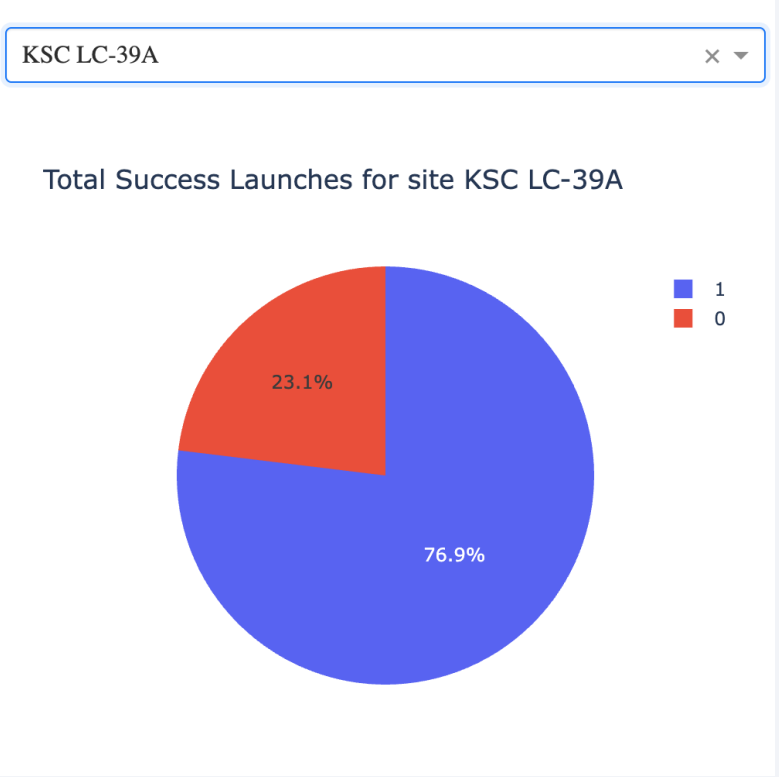
Findings:

- **KSC LC-39A (41.7%)** has the **highest number of successful launches**, making it a key launch site.
- **CCAFS LC-40 (29.2%)** follows as another significant launch site.
- **VAFB SLC-4E (16.7%)** and **CCAFS SLC-40 (12.5%)** contribute a smaller share of successful launches.
- The distribution suggests that **Florida-based launch sites (KSC & CCAFS) dominate in terms of successful missions**, reinforcing their strategic importance for SpaceX operations. 🚀



Pie chart showing the Launch site with the highest launch success ratio

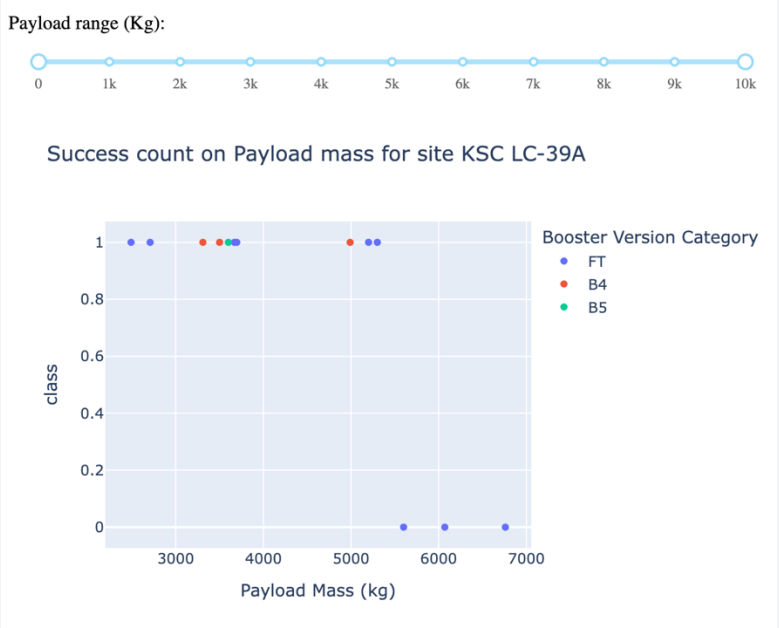
- **KSC LC-39A has a 76.9% success rate**, making it the most reliable launch site.
- **Failures (23.1%) are minimal**, showing consistent improvements.
- **Optimized infrastructure** supports frequent and high-priority missions.



Success count on Payload mass for site KSC LC-39A

Successful launches dominate, especially for payloads **below 6000 kg**.

- **Higher payloads (~5000-7000 kg) show some failures**.
- **Different booster versions (FT, B4, B5) perform across various payloads**.
- The graph helps **analyze payload limits and booster reliability**.

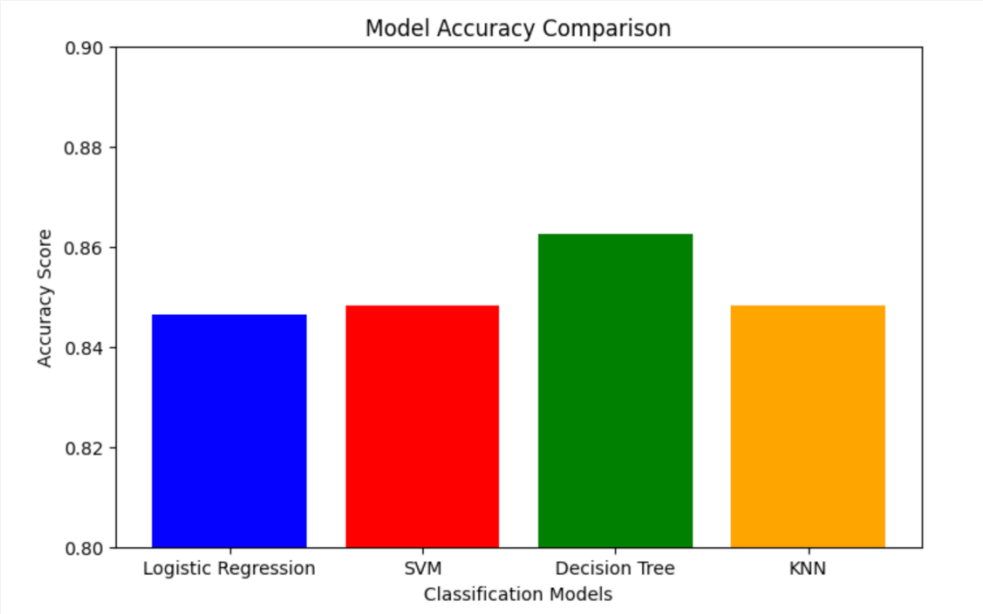


Section 5

Predictive Analysis (Classification)

Classification Accuracy

The **Decision Tree model** achieved the **highest accuracy**, outperforming **Logistic Regression, SVM, and KNN**. The results suggest that **tree-based methods may be more effective** for this classification task.

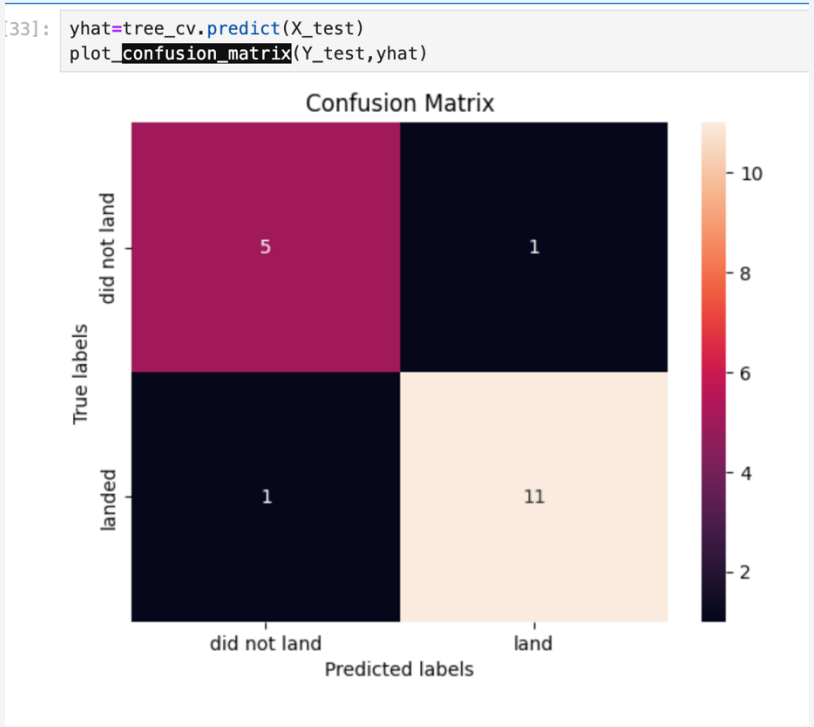


Confusion Matrix

- **True Positives (Bottom-right, 11):** Correctly predicted successful landings.
- **True Negatives (Top-left, 5):** Correctly predicted failures.
- **False Positives (Top-right, 1):** Incorrectly predicted a failure as a success.
- **False Negatives (Bottom-left, 1):** Incorrectly predicted a success as a failure.

Findings:

- The model performs **well with high accuracy**, correctly classifying most landings.
- **Few misclassifications (1 false positive, 1 false negative)** indicate **room for slight improvements**.



Conclusions

1. **Decision Tree achieved the highest accuracy** among tested models.
2. **The confusion matrix showed minimal misclassifications**, proving strong model performance.
3. **Feature analysis highlighted key factors** influencing landing success.
4. **Further tuning may optimize accuracy and performance.**

Appendix

Data Sets

SpaceX API JSON Data (Contains Falcon 9 launch records).

Wikipedia Scraped Data (Includes mission details for Falcon 9 launches).

Processed DataFrames (Used for EDA and model training).

Thank you!