Assil Tarhouni


09/08/2023

# Winning Space Race with Data Science :

# Space X Falcon 9 Landing Analysis

# Outline

1. Executive Summary
2. Introduction
3. Methodology
4. Results
5. Conclusion

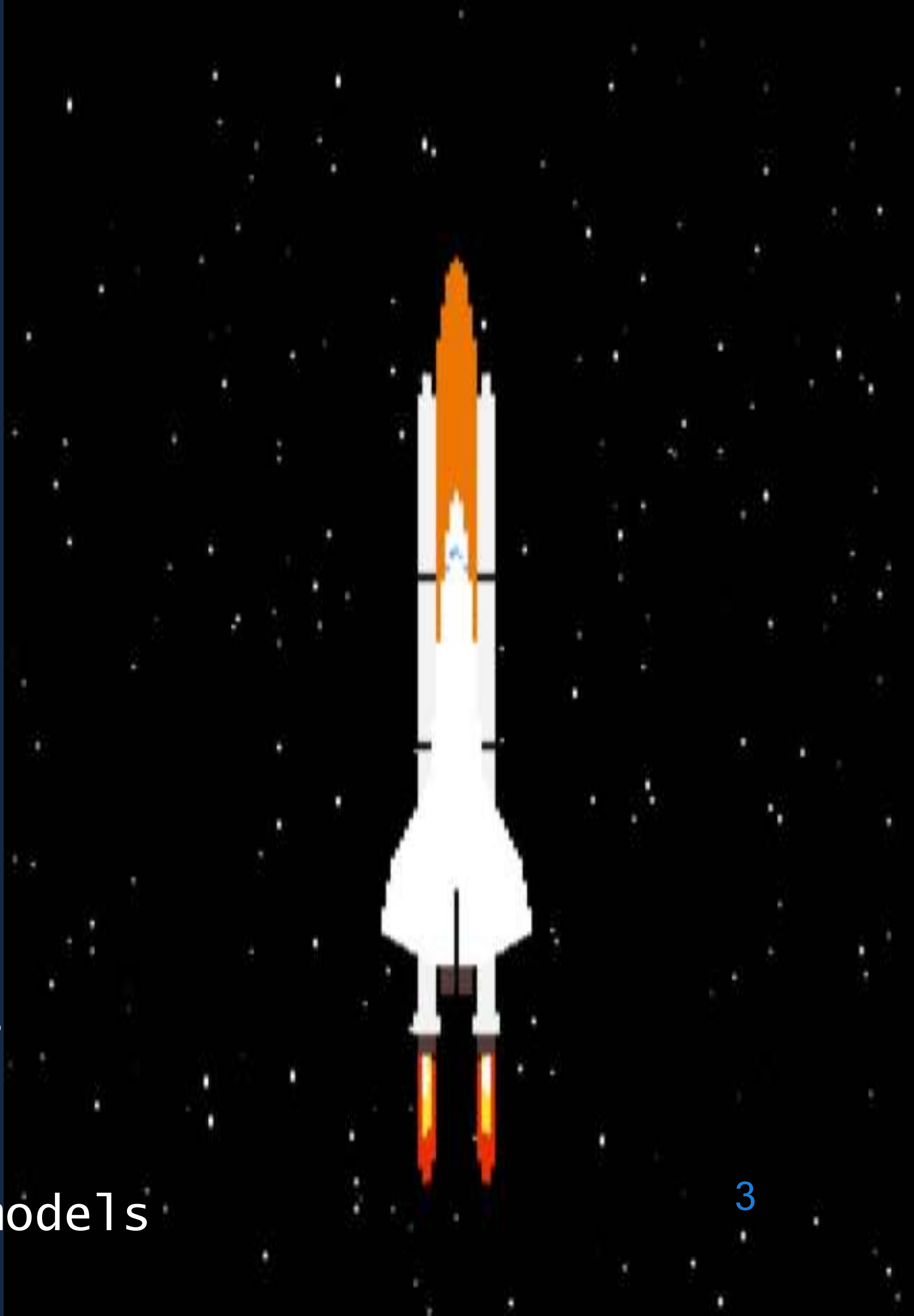
# Executive Summary

- **Summary of Methodologies:**

  The project involves a step-by-step approach:
  - Data Collection
  - Data Wrangling
  - Exploratory Data Analysis
  - Interactive Visual Analytics
  - Predictive Analysis (Classification)

- **Summary of Results:**

  The project yielded valuable outputs and visualizations:
  - Exploratory Data Analysis (EDA) results
  - Geospatial analytics
  - Interactive dashboard
  - Predictive analysis of classification models



# Introduction

- SpaceX conducts its Falcon 9 rocket launches at a significantly lower cost of approximately $62 million compared to other providers, whose prices often exceed $165 million. The major cost-saving factor lies in SpaceX's ability to land and reuse the first stage of the rocket.
- By predicting the success of the first stage landing, we can accurately estimate the total launch cost. This information becomes crucial in evaluating whether competing companies should consider bidding against SpaceX for a rocket launch contract.
- The primary objective of this project is to develop a prediction model that determines the likelihood of a successful landing for the Space X Falcon 9 first stage.

Section 1

# Methodology

# Methodology

1. Data Collection
   - Making GET requests to the SpaceX REST API
   - Web Scraping
2. Data Wrangling
   - Using the .fillna() method to remove NaN values
   - Using the .value_counts() method to determine the following:
     - Number of launches on each site
     - Number and occurrence of each orbit
     - Number and occurrence of mission outcome per orbit type
   - Creating a landing outcome label that shows the following:
     - 0 when the booster did not land successfully
     - 1 when the booster did land successfully
3. Exploratory Data Analysis
   - Using SQL queries to manipulate and evaluate the SpaceX dataset
   - Using Pandas and Matplotlib to visualize relationships between variables, and determine patterns

# Methodology

4. Interactive Visual Analytics
   - Geospatial analytics using Folium
   - Creating an interactive dashboard using Plotly Dash
5. Data Modelling and Evaluation
   - Using Scikit-Learn to:
     - Pre-process (standardize) the data
     - Split the data into training and testing data using `train_test_split`
     - Train different classification models
     - Find hyperparameters using `GridSearchCV`
   - Plotting confusion matrices for each classification model
   - Assessing the accuracy of each classification model

# Data Collection: space x REST api

Make a GET response to the SpaceX REST API and Convert the response to a .json file then to a Pandas DataFrame

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
response = requests.get(static_json_url)
data = pd.json_normalize(response.json())
```

Use custom logic to clean the data,Define lists for data to be stored in Call custom functions to retrieve data and fill the lists and Use these lists as values in a dictionary and construct the dataset.

```
BoosterVersion = []
PayloadMass = []
Orbit = []
LaunchSite = []
Outcome = []
Flights = []
GridFins = []
Reused = []
Legs = []
LandingPad = []
Block = []
ReusedCount = []
Serial = []
Longitude = []
Latitude = []
getBoosterVersion(data)
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

# Data Collection: space x REST api

Create a Pandas DataFrame from the constructed dictionary dataset and Filter the DataFrame to only include Falcon 9 launches.Reset the FlightNumber column.Replace missing values of PayloadMass with the mean PayloadMass value

```
df = pd.DataFrame.from_dict(launch_dict)
df.head()
data_falcon9 = df[df['BoosterVersion']!='Falcon 1']
data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
data_falcon9.isnull().sum()
data_falcon9 = data_falcon9.fillna(value={'PayloadMass': data_falcon9['PayloadMass'].mean()})
```

# Data Collection: WEB SCRAPING

Request the HTML page from the static URL and Assign the response to an object.Create a BeautifulSoup object from the HTML response object then Find all tables within the HTML page

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
response = requests.get(static_url)
data = response.text
soup = BeautifulSoup(data, 'html5lib')
html_tables = soup.find_all('table')
column_names = []
for row in first_launch_table.find_all('th'):
    name = extract_column_from_header(row)
    if(name != None and len(name) > 0):
        column_names.append(name)
```

Collect all column header names from the tables found within the HTML page.Use the column names as keys in a dictionary and use custom functions and logic to parse all launch tables to fill the dictionary values then Convert the dictionary to a Pandas DataFrame ready for export

```
extracted_row = 0
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
    for rows in table.find_all("tr"):
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        row=rows.find_all('td')
        if flag:
            extracted_row += 1
            launch_dict["Flight No."].append(flight_number)
            datatimelist=date_time(row[0])
```

# Data Wrangling : PANDAS

the landing outcomes of SpaceX missions are categorized into different scenarios based on specific regions or platforms. A "True Ocean" outcome signifies a successful landing in the ocean, while "False Ocean" indicates an unsuccessful ocean landing. Similarly, "True RTLS" represents a successful landing on a ground pad, "False RTLS" indicates an unsuccessful ground pad landing, "True ASDS" signifies a successful landing on a drone ship, and "False ASDS" indicates an unsuccessful drone ship landing. The labels "None ASDS" and "None None" correspond to failed landing attempts.

In the process of data wrangling, the objective is to create a binary column that captures the success or failure of a booster landing. To achieve this:

A set of unsuccessful (bad) outcomes, known as bad_outcome, is defined.

A new list called landing_class is generated. In this list, the value is set to 0 if the corresponding row in the Outcome column is classified as a bad outcome (in the set bad_outcome), otherwise, it's set to 1.

A new column named "Class" is created in the DataFrame, containing the values from the landing_class list.

The modified DataFrame is exported and saved as a .csv file.

In summary, this data wrangling process transforms the landing outcomes into a binary representation of landing success, allowing for further analysis and prediction of booster landings.

```
landing_outcomes = df['Outcome'].value_counts()
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
landing_class = []
for outcome in df['Outcome']:
    if outcome in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)
df['Class']=landing_class
df.to_csv("dataset_part\_2.csv", index=False)
```

# Exploratory data analysis EDA with Data Visualization

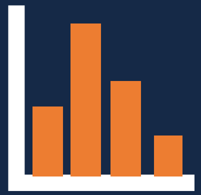
## LINE CHARTS

Line charts were produced to visualize the relationships between:

Success Rate and Year (i.e. the launch success yearly trend)

## BAR CHART

A bar chart was produced to visualize the relationship between: Success Rate and Orbit Type
Bar charts are used to compare a numerical value to a categorical variable. Horizontal or vertical bar charts can be used, depending on the size of the data.



## SCATTER CHARTS

Scatter charts were produced to visualize the relationships between:

- Flight Number and Launch Site
- Payload and Launch Site
- Orbit Type and Flight Number
- Payload and Orbit Type

Scatter charts are useful to observe relationships, or correlations, between two numeric variables.

# Exploratory data analysis EDA with SQL

Several SQL queries were executed to extract valuable insights from the dataset. These queries were utilized to achieve the following objectives:

1. Retrieve the unique launch site names involved in space missions.
2. Display five records where the launch site names begin with the characters 'CCA.'
3. Calculate the total payload mass carried by boosters launched under NASA's Commercial Resupply Services (CRS) program.
4. Determine the average payload mass carried by boosters of the F9 v1.1 version.
5. Identify the date of the first successful landing outcome on a ground pad.
6. List the names of boosters that achieved success on a drone ship while carrying a payload mass ranging from 4000 to 6000 kg.
7. Present the total count of successful and failed mission outcomes.
8. Identify the names of booster versions that have carried the maximum payload mass.
9. List the failed landing outcomes on drone ships, along with their corresponding booster versions and launch site names specifically for the year 2015.
10. Rank the count of landing outcomes, such as "Failure (drone ship)" or "Success (ground pad)," between the time frame of June 4, 2010, and March 20, 2017, in descending order.

These SQL queries provided valuable information and insights into various aspects of the space mission dataset.

# Build an Interactive Map with Folium

To visualize launch data interactively on a map:

- Identify and Mark Launch Sites:
  - Mark all launch sites using folium.Circle and folium.Marker components on a Folium Map object.
- Display Success/Failure:
  - Assign colors (green for success, red for failure) to markers to differentiate outcomes.
- Cluster for Efficiency:
  - Cluster launches with shared coordinates using MarkerCluster() for clearer representation.
- Analyze Proximities:
  - Calculate distances between launch sites using Lat and Long values.
  - Display distances with folium.Marker and visualize them with folium.PolyLine.

This approach offers a concise way to present launch site distribution, outcomes, clustering, and proximity insights on an interactive map.

# Build an Interactive Map with Folium

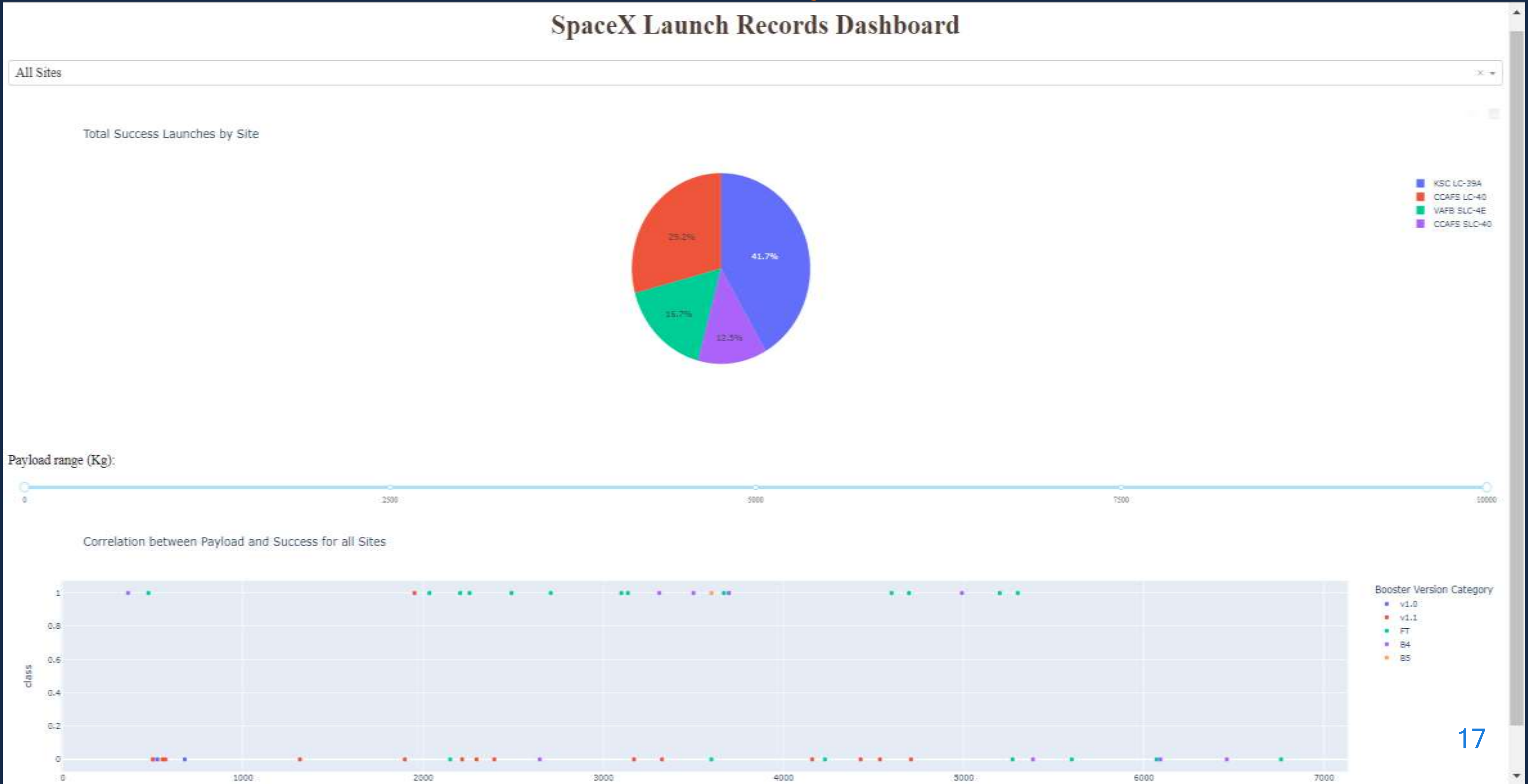
# Build an Interactive Map with Plotly Dash

Plots on the Plotly Dash dashboard for interactive visualization:

1. Success Breakdown by Launch Site:
   - Utilized px.pie() to display a pie chart showcasing total successful launches per site.
   - Provides clear insight into site performance.
   - Incorporates filtering options via dcc.Dropdown() to analyze success/failure ratio per specific site.
2. Outcome-Payload Correlation:
   - Employed px.scatter() to present a scatter graph illustrating the connection between outcome (success or not) and payload mass (kg).
   - Enables filtering through RangeSlider() for payload mass ranges.
   - Further customization includes filtering by booster version for enhanced analysis.

These dashboard plots offer an interactive way to explore success patterns across launch sites and understand the correlation between outcomes and payload mass, providing users with valuable insights.

# Build a Dashboard with Plotly Dash



# Predictive Analysis: Classification

The following steps were taking to develop, evaluate, and find the best performing classification model:

## Model Development

1. Load dataset.
2. Apply data transformations (standardize, preprocess).
3. Split data into training and test sets.
4. Choose suitable ML algorithms.
5. For each algorithm:

- Use GridSearchCV for hyperparameter tuning.
- Train the model using the training dataset.

## Model Evaluation

Utilize the generated GridSearchCV object:Inspect the optimized hyperparameters (best_params_).

Assess the accuracy using score and best_score_.

Visualize and analyze the Confusion Matrix

## Finding the Best Classification Model

Evaluate the accuracy scores across all selected algorithms, identifying the model with the highest accuracy score as the best-performing one.

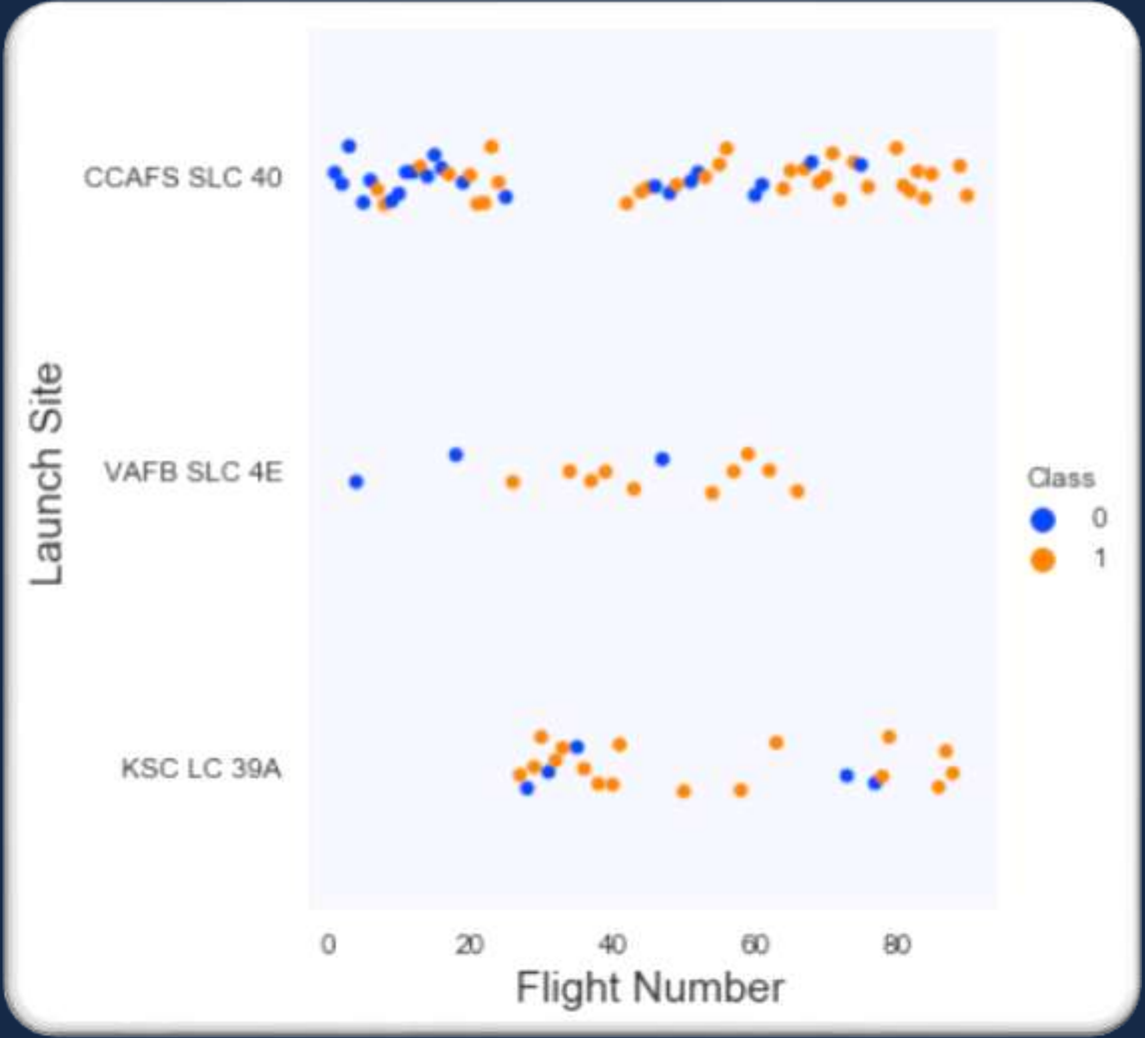
Section 2

# Insights drawn from EDA

# Flight Number vs Launch Site

The scatter plot depicting the relationship between Launch Site and Flight Number reveals several insights:

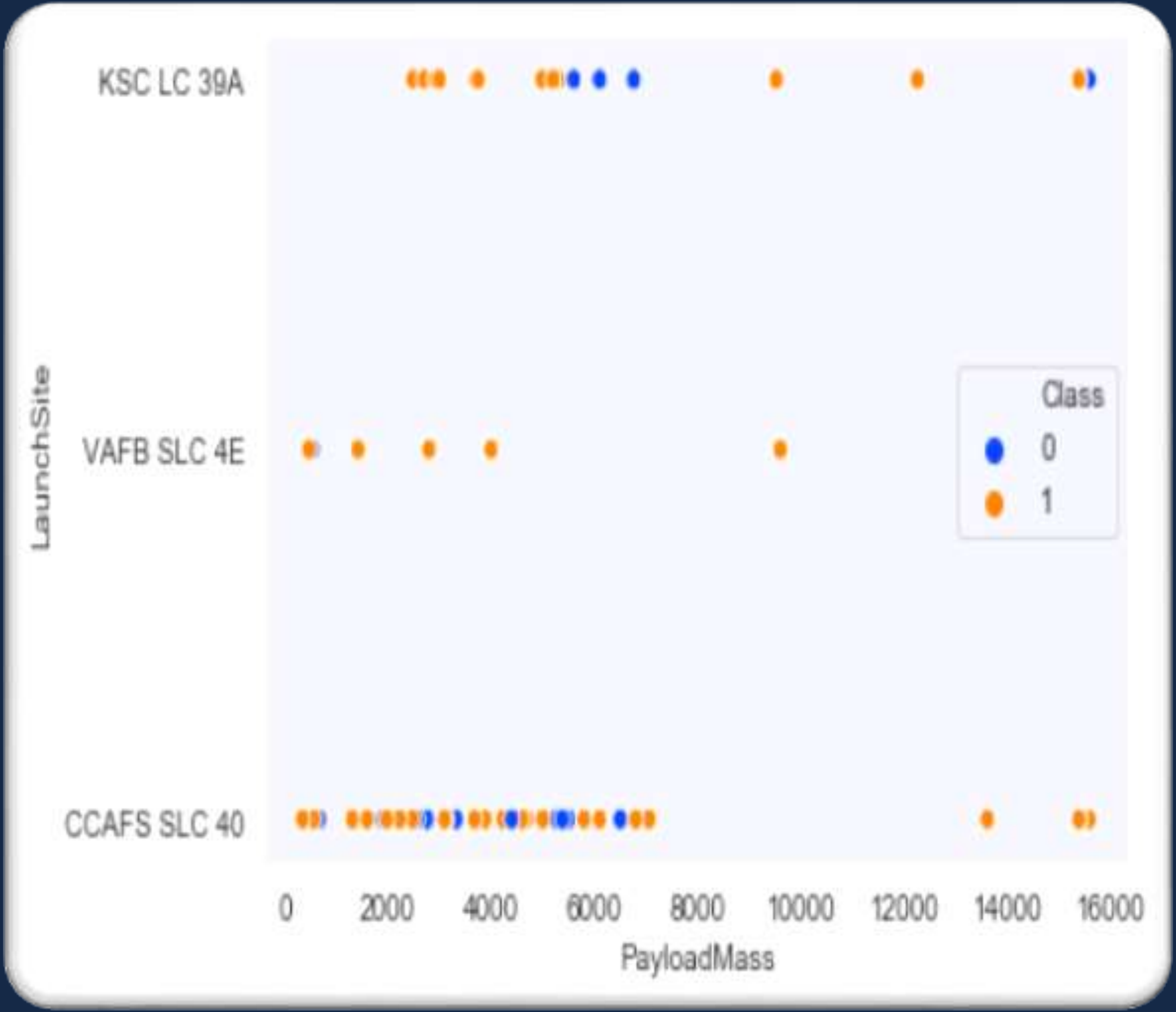
1. **As the number of flights increases, there is a noticeable upward trend in the success rate at a given launch site. This suggests an improvement in launch operations over time.**
2. **For flights with lower flight numbers (less than 30), most of them were conducted from the CCAFS SLC 40 launch site. However, these early flights tend to have a lower success rate.**
3. **Similar to CCAFS SLC 40, early flights from the VAFB SLC 4E launch site also exhibit a lower success rate.**
4. **KSC LC 39A, on the other hand, did not have any early flights (with flight numbers below 30). Consequently, launches from this site show a higher success rate compared to the other launch sites.**
5. **Beyond a flight number of approximately 30, there is a significant increase in the number of successful landings (Class = 1). This suggests a positive trend in successful outcomes as the flight number advances.**



# Payload Mass vs. Launch Site

The scatter plot illustrating the relationship between Launch Site and Payload Mass provides the following insights:

1. Beyond a payload mass threshold of approximately 7000 kg, instances of unsuccessful landings become sparse. However, it's important to acknowledge that data availability for these higher payload launches is limited.
2. The scatter plot doesn't reveal a distinct correlation between payload mass and the success rate specific to each launch site. This suggests that while payload mass may influence mission outcomes to some extent, other factors also play a significant role.
3. Each launch site has demonstrated the capability to accommodate a range of payload masses. Notably, CCAFS SLC 40 stands out with a higher concentration of launches involving relatively lighter payloads. Nonetheless, there are instances of outlier launches with heavier payloads from this site as well.



# Success Rate vs. Orbit Type

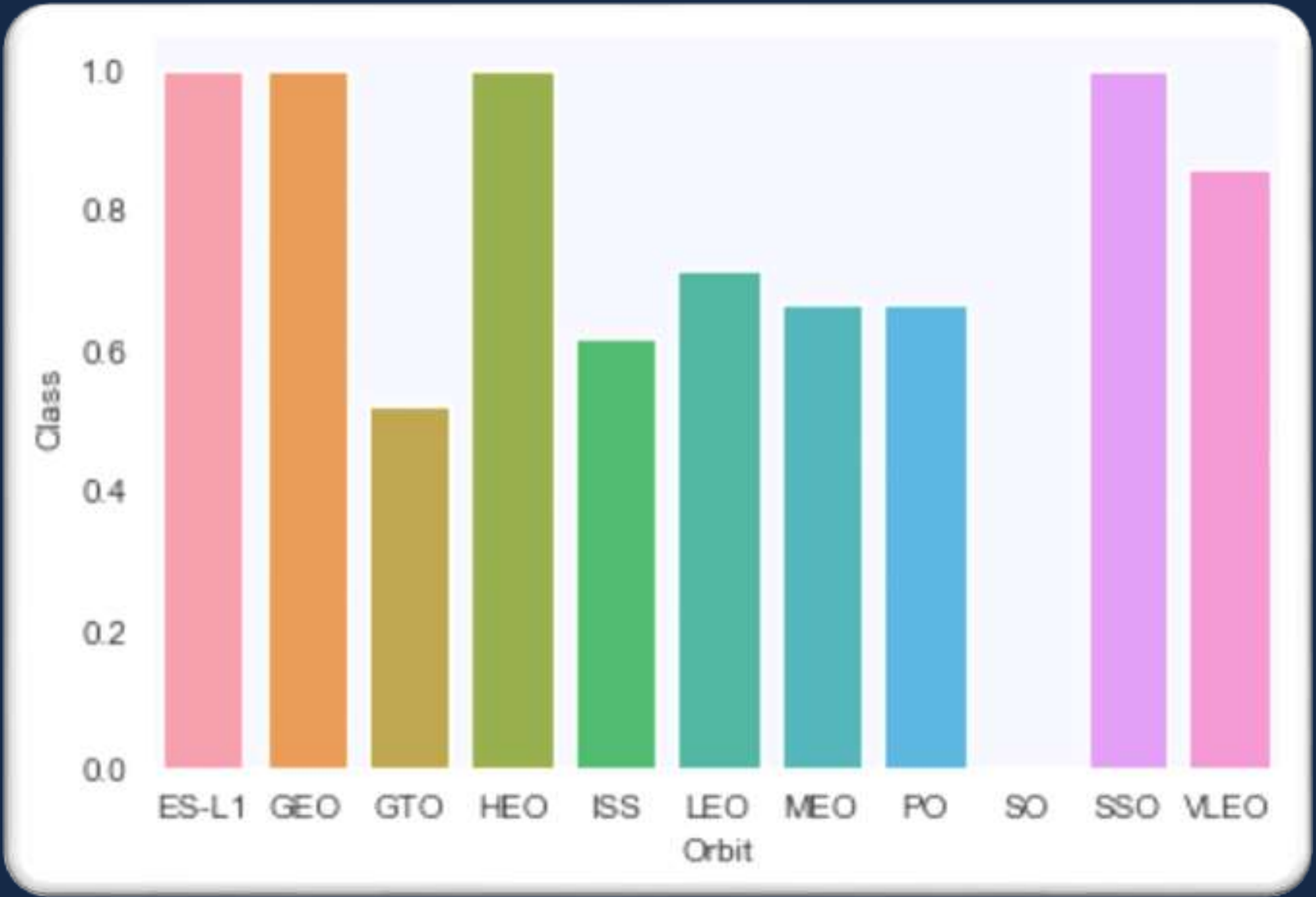
Based on the bar chart depicting Success Rate vs. Orbit Type, the analysis reveals the following observations:

Notably, several orbit types exhibit a remarkable success rate of 100%. These include:

- ES-L1 (Earth-Sun First Lagrangian Point)
- GEO (Geostationary Orbit)
- HEO (High Earth Orbit)
- SSO (Sun-synchronous Orbit)

Conversely, the orbit type with the lowest recorded success rate is:

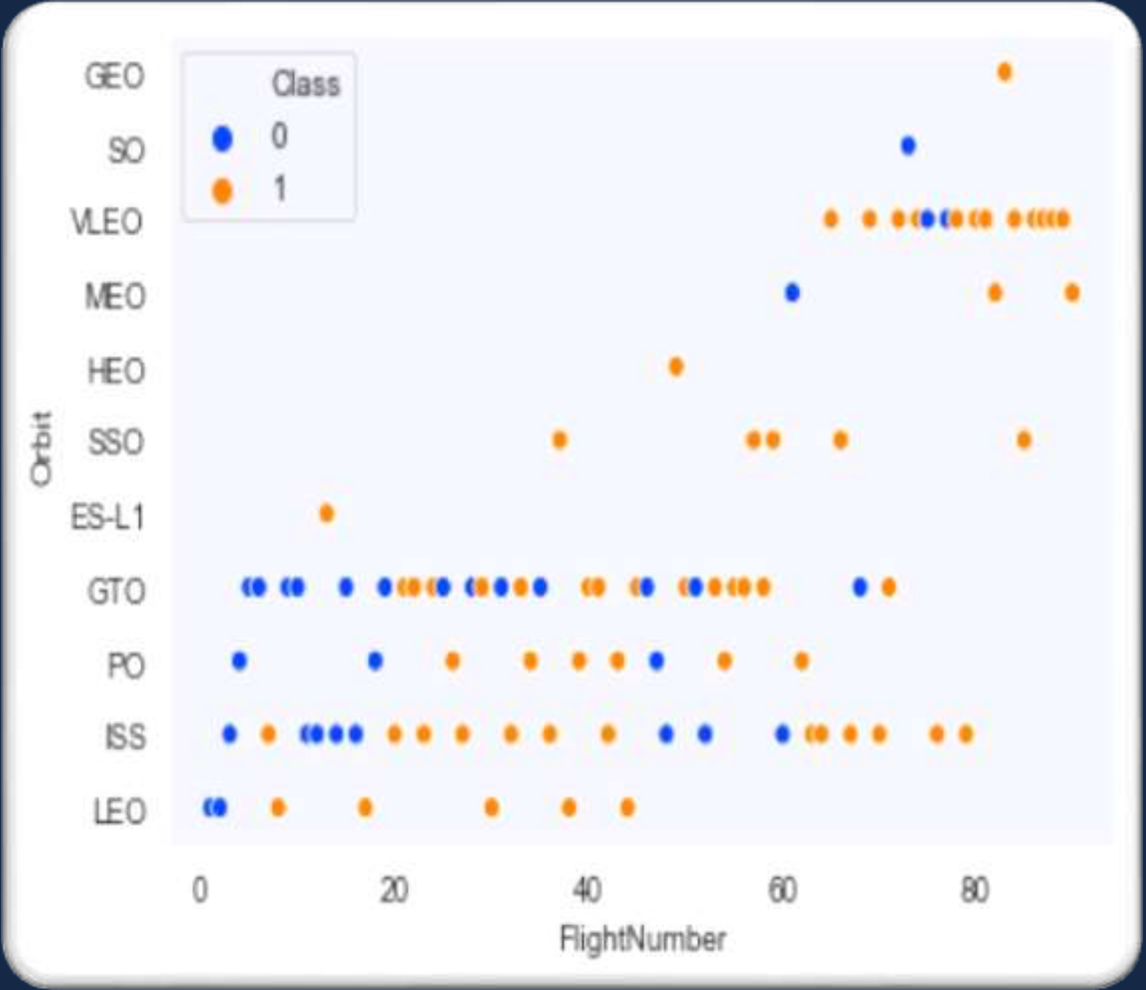
- SO (Heliocentric Orbit), which registers a 0% success rate.



# Flight Number vs. Orbit Type

The scatter plot illustrating Orbit Type vs. Flight Number provides unique insights that were not apparent in previous visualizations:

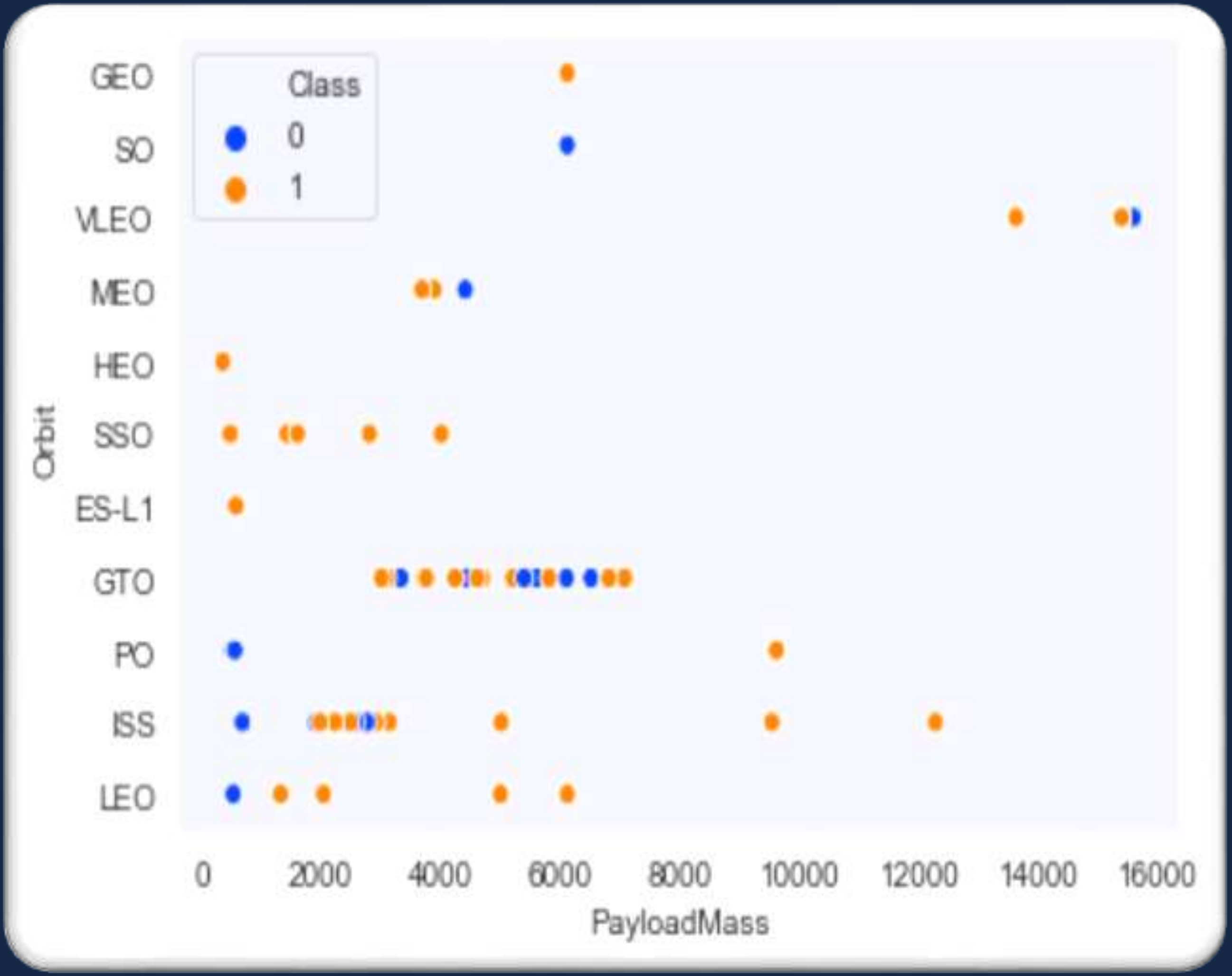
- The exceptional 100% success rates observed in GEO, HEO, and ES-L1 orbits can be attributed to the fact that only a single flight was conducted for each of these orbits, contributing to their perfect success records.
- Notably, the SSO orbit stands out with an impressive 100% success rate, supported by a total of 5 successful flights to this orbit.
- In the case of the GTO orbit, there seems to be a lack of discernible correlation between Flight Number and Success Rate. This indicates that the success rate does not exhibit a consistent pattern across different flight numbers for this orbit.
- As a general trend, an increase in Flight Number appears to coincide with an improvement in the success rate. This pattern is particularly pronounced for the LEO orbit, where instances of unsuccessful landings were predominantly limited to early flight numbers.



# Payload vs. Orbit Type

The scatter plot depicting Orbit Type vs. Payload Mass reveals the following intriguing observations:

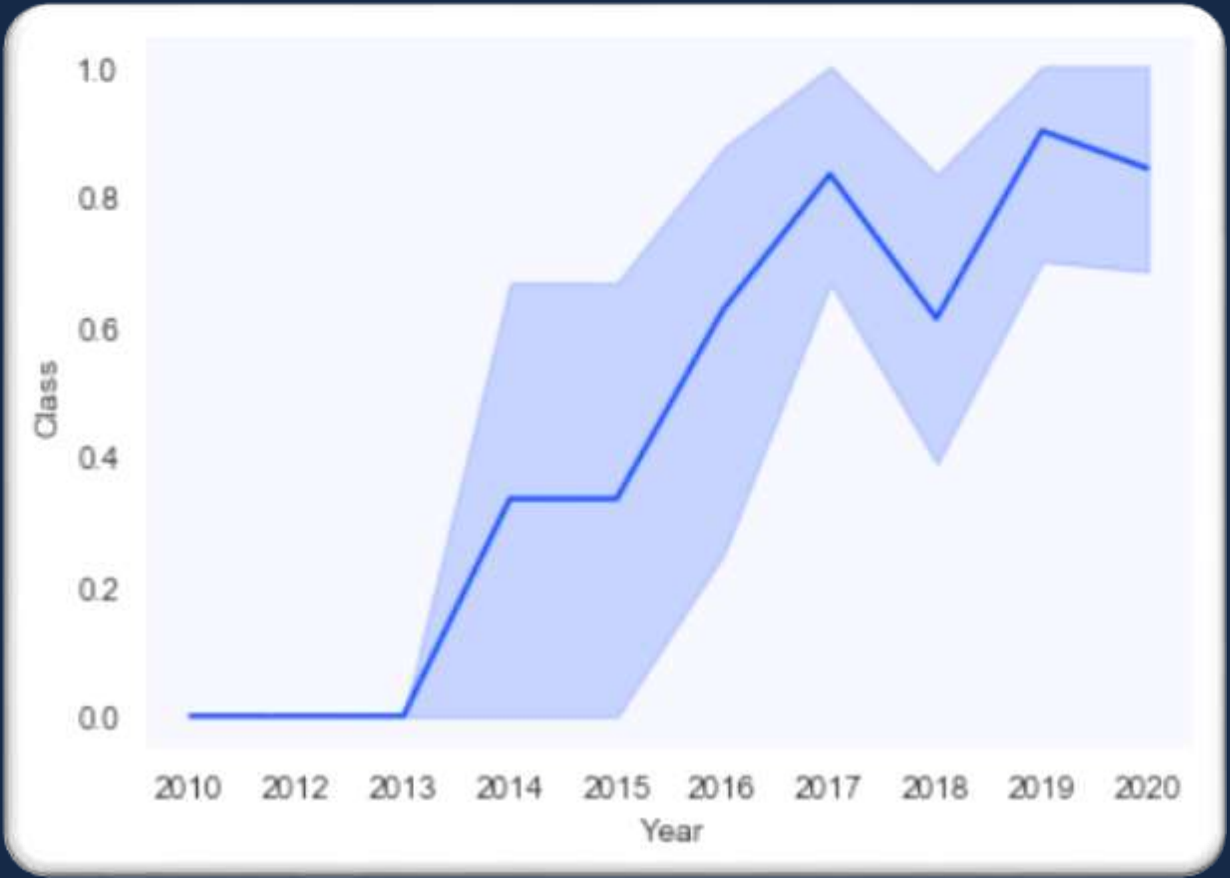
1. Several orbit types demonstrate a propensity for success when carrying heavy payloads:
   - PO exhibits this trend, albeit with a limited number of data points.
   - ISS, a well-known orbit, shows a favorable success rate for heavier payloads.
   - LEO is another orbit type that aligns with success when handling larger payloads.
2. In the case of GTO, the connection between payload mass and success rate appears less evident, requiring further investigation to ascertain any definitive correlation.
3. The VLEO (Very Low Earth Orbit) launches consistently involve heavier payloads, a correlation that intuitively corresponds to the demands of this orbit type



# Launch Success Yearly Trend

The line chart depicting the yearly average success rate unveils the following significant trends:

1. In the period spanning from 2010 to 2013, all landing attempts experienced failure, signified by a success rate of 0%.
2. Post-2013, a promising shift occurred as the success rate began to ascend. This positive trajectory was momentarily interrupted by minor declines observed in the years 2018 and 2020.
3. From 2016 onward, the success rate consistently exceeded the threshold of 50%, implying a notably higher likelihood of successful landings.



# All Launch Site Names

```
%sql SELECT UNIQUE(LAUNCH_SITE) FROM SPACEXTBL;
```

| launch_site |
|---|
| CCAFS LC-40 |
| CCAFS SLC-40 |
| KSC LC-39A |
| VAFB SLC-4E |

# Launch Site Names Begin with 'CCA'

```
%sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

| launch_site |
|---|
| CCAFS LC-40 |
| CCAFS LC-40 |
| CCAFS LC-40 |
| CCAFS LC-40 |
| CCAFS LC-40 |

# Total Payload Mass

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL \
    WHERE CUSTOMER = 'NASA (CRS)';
```

| total_payload_mass |
|---|
| 45596 |

# Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS FROM SPACEXTBL \
    WHERE BOOSTER_VERSION = 'F9 v1.1';
```

| average_payload_mass |
|---|
| 2928 |

# First Successful Ground Landing Date

```
%sql SELECT MIN(DATE) AS FIRST_SUCCESSFUL_GROUND_LANDING FROM SPACEXTBL \
    WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

| first_successful_ground_landing |
| --- |
| 2015-12-22 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL \
    WHERE (LANDING__OUTCOME = 'Success (drone ship)') AND (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 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

```
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

| mission_outcome | total_number |
|---|---|
| Failure (in flight) | 1 |
| Success | 99 |
| Success (payload status unclear) | 1 |

# Boosters Carried Maximum Payload

```
%sql SELECT DISTINCT(BOOSTER_VERSION) FROM SPACEXTBL \
    WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

| booster_version |
|---|
| F9 B5 B1048.4 |
| F9 B5 B1048.5 |
| F9 B5 B1049.4 |
| F9 B5 B1049.5 |
| F9 B5 B1049.7 |
| F9 B5 B1051.3 |

# 2015 Launch Records

```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL \
    WHERE (LANDING__OUTCOME = 'Failure (drone ship)') AND (EXTRACT(YEAR FROM DATE) = '2015');
```

| booster_version | launch_site |
|---|---|
| F9 v1.1 B1012 | CCAFS LC-40 |
| F9 v1.1 B1015 | CCAFS LC-40 |

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

```
%sql SELECT LANDING__OUTCOME, COUNT(LANDING__OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL \
    WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
    GROUP BY LANDING__OUTCOME \
    ORDER BY TOTAL_NUMBER DESC;
```

| landing__outcome | total_number |
|---|---|
| No attempt | 10 |
| Failure (drone ship) | 5 |
| Success (drone ship) | 5 |
| Controlled (ocean) | 3 |
| Success (ground pad) | 3 |
| Failure (parachute) | 2 |
| Uncontrolled (ocean) | 2 |
| Precluded (drone ship) | 1 |

Section 3

# Launch Sites Proximities Analysis



**The Foluim interactive map shows the geographical distribution of SpaceX launch sites is concentrated along the coastal regions of the United States, with specific locations in Florida and California.**

# SUCCESS/FAILED LAUNCHES FOR EACH SITE

Launches have been grouped into clusters, and annotated with green icons for successful launches, and red icons for failed launches.

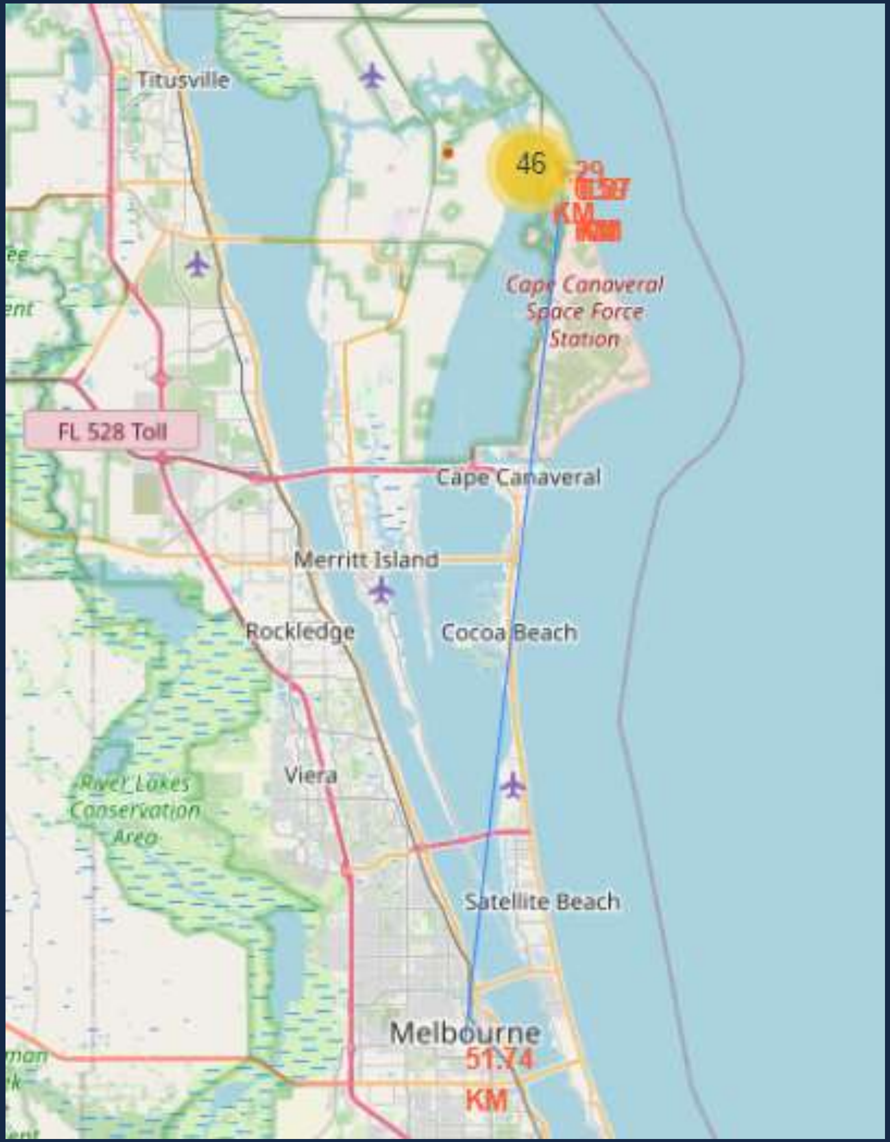
## Florida


## California

# EXAMINING LAUNCH SITE PROXIMITY TO OTHER LOCATIONS OF INTEREST

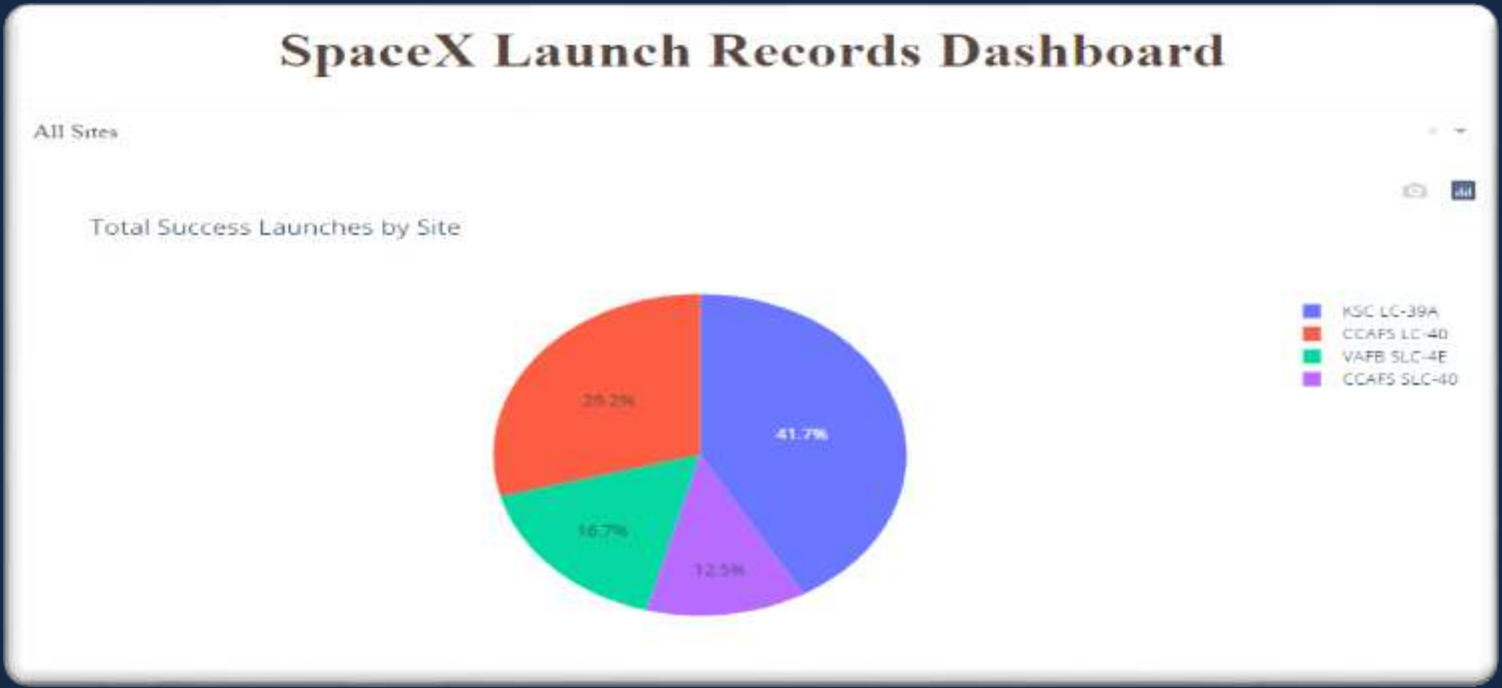
By taking the CCAFS SLC-40 launch site as a representative example, we can gain insights into the spatial arrangement of launch sites.

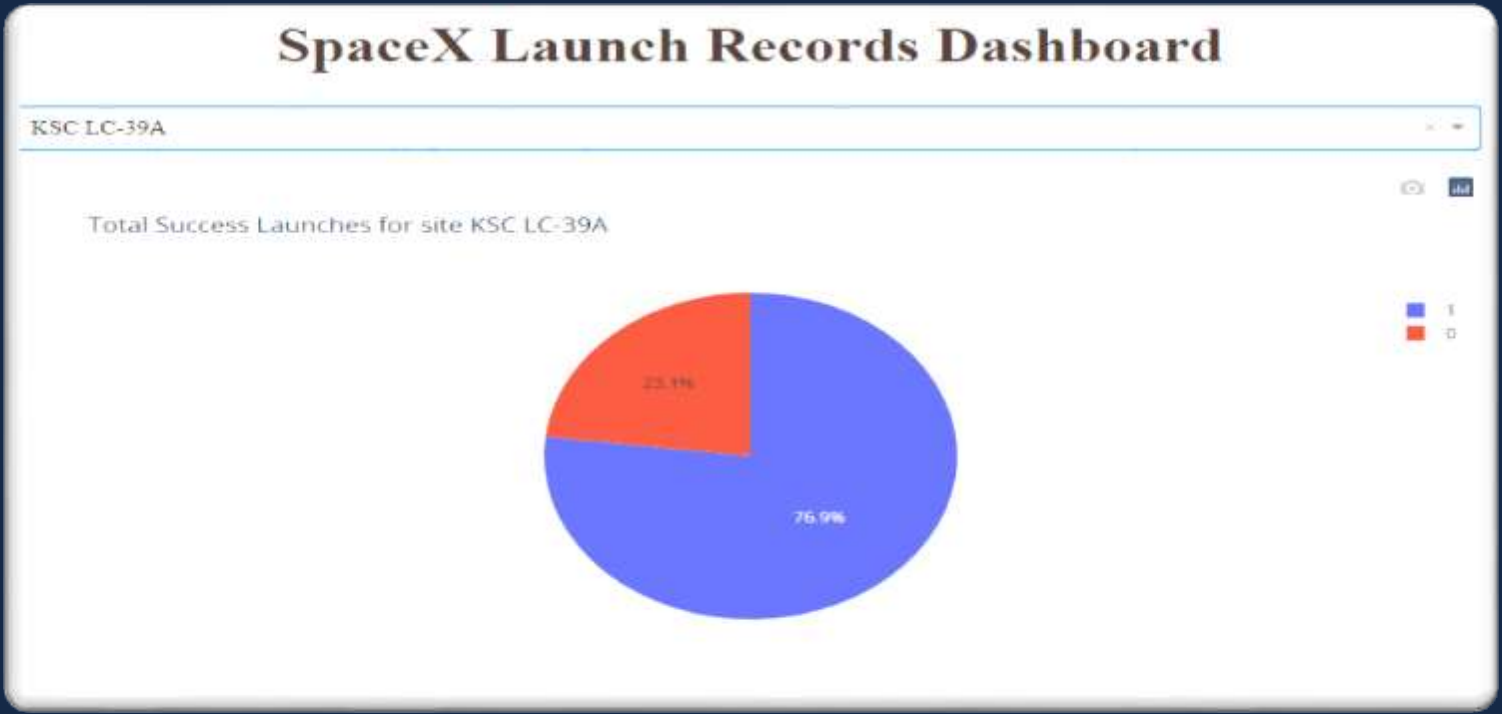
- Proximity to Railways: Indeed, launch sites are situated in close proximity to railways. In the case of CCAFS SLC-40, the coastline is merely 0.87 km to the east.
- Proximity to Highways: The closeness to highways is also evident. The nearest highway is a mere 0.59 km away from the launch site.
- Proximity to Railways (Again): Similar to highways, launch sites are also conveniently positioned near railways. The nearest railway is located just 1.29 km away.
- Distance from Cities: A deliberate separation from urban areas is observed. The closest city is situated at a distance of 51.74 km from the launch site.



Section 4

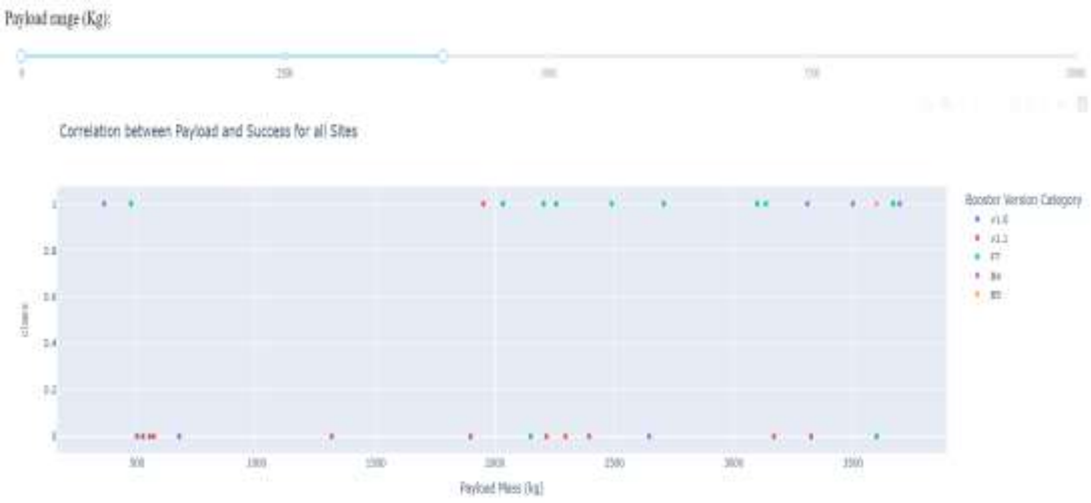
# Build a Dashboard with Plotly Dash

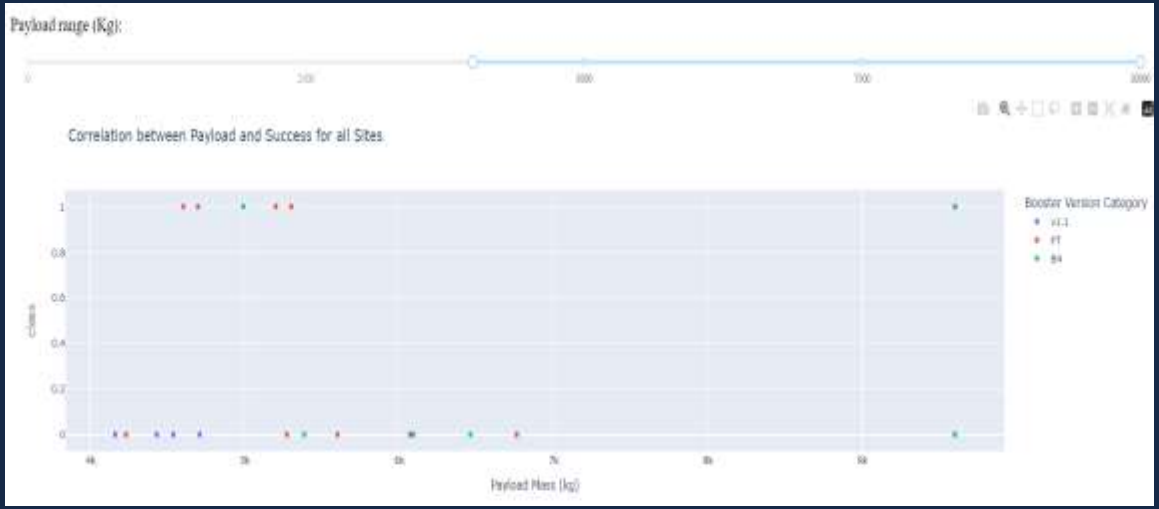



KSC LC-39 A emerged as the leader in successful launches, contributing to 41.7% of the total successful missions. Impressively, this launch site also boasted the highest success rate among all, standing at an impressive 76.9%.

# Launch Outcome VS. Payload scatter plot for all sites




When visualizing the launch outcomes against payload for all sites, a distinct gap appears around the 4000 kg mark. This gap prompts a logical division of the data into two ranges:

1. Payloads from 0 to 4000 kg, categorized as "low payloads."
2. Payloads from 4000 to 10000 kg, classified as "massive payloads."

Analyzing these two distinct plots reveals that the success rate for massive payloads is notably lower than that for low payloads. Furthermore, it's noteworthy that certain booster types, specifically v1.0 and B5, have not been utilized for launches involving massive payloads.

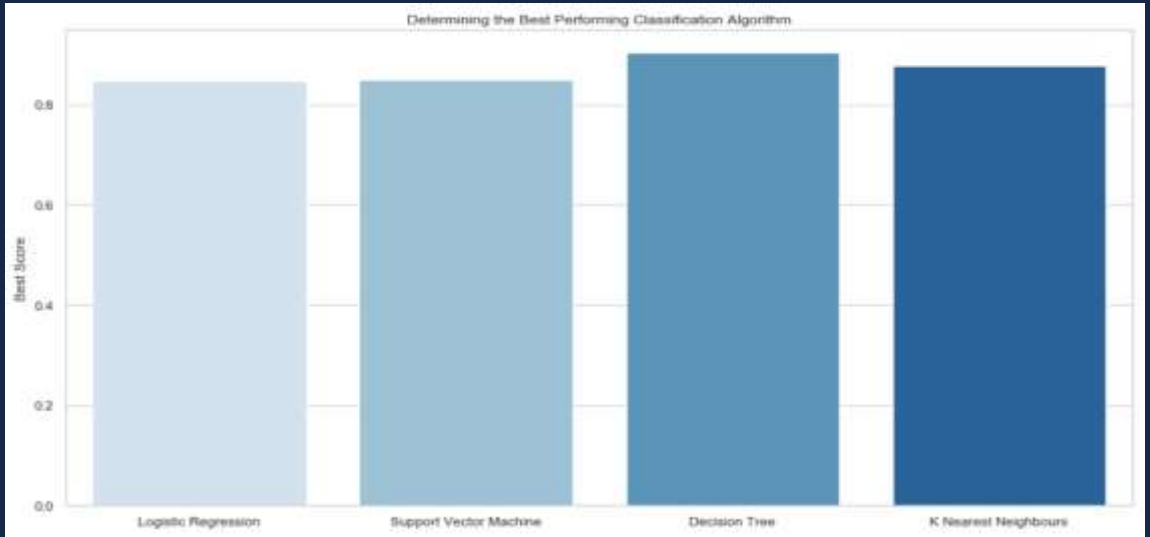
Section 5

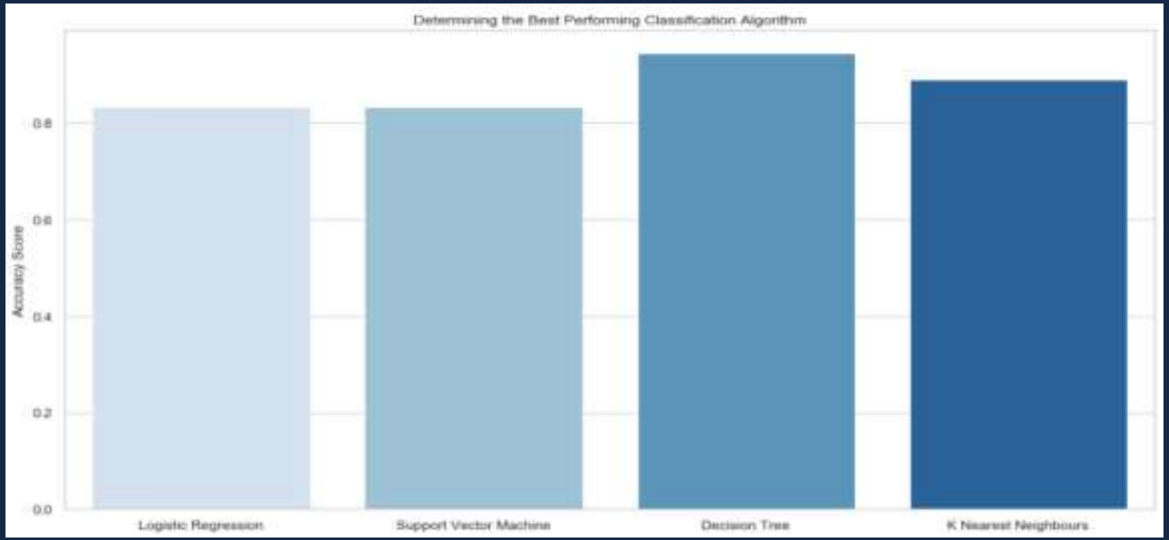
# Predictive Analysis (Classification)

# Classification Accuracy

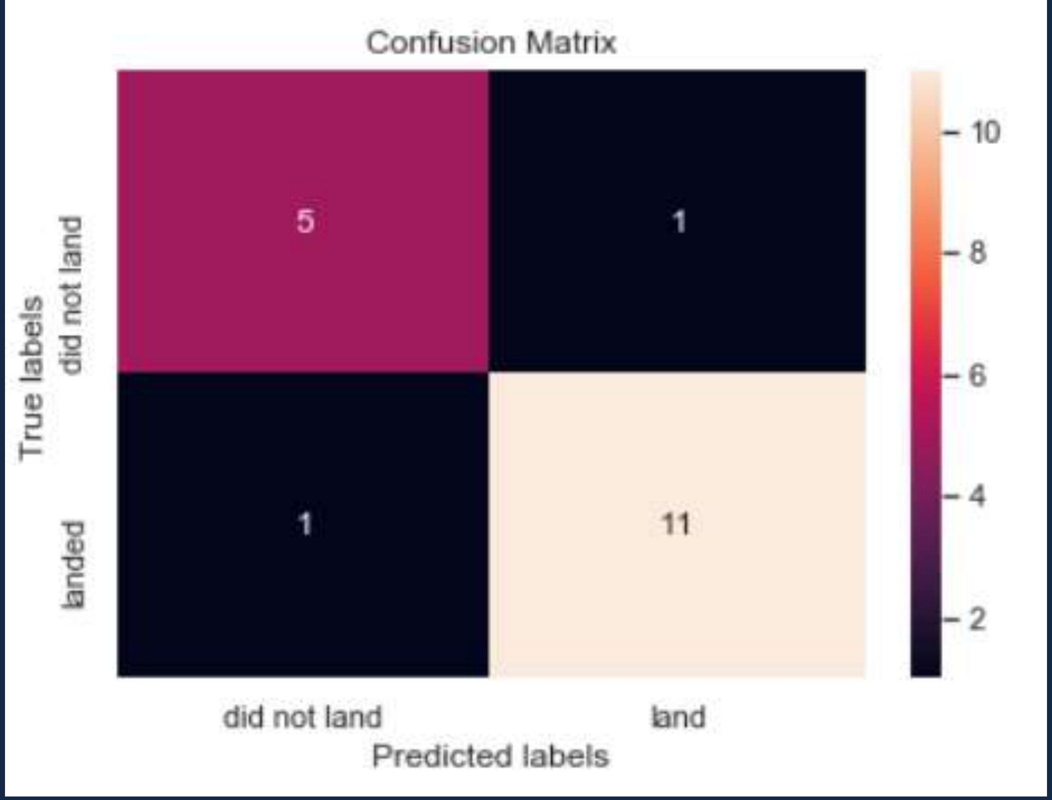
Upon plotting the Accuracy Score and Best Score for each classification algorithm, the analysis yields the following outcomes: The Decision Tree model emerges as the top-performing classification algorithm, exhibiting the highest classification accuracy. Specifically, the Accuracy Score achieved by the Decision Tree model is 94.44%. Meanwhile, the Best Score attained by the same model amounts to 90.36%.

| | Algorithm | Accuracy Score | Best Score |
|---|---|---|---|
| **0** | Logistic Regression | 0.833333 | 0.846429 |
| **1** | Support Vector Machine | 0.833333 | 0.848214 |
| **2** | Decision Tree | 0.944444 | 0.903571 |
| **3** | K Nearest Neighbours | 0.888889 | 0.876786 |



# Confusion Matrix




As demonstrated earlier, the Decision Tree model stands out as the most proficient classification model, achieving an impressive accuracy rate of 94.44%. This exceptional performance is elucidated by the confusion matrix, where only a solitary instance out of the total 18 results was incorrectly classified (identified as a false positive, positioned in the top-right quadrant). Conversely, the remaining 17 outcomes were accurately classified, of which 5 corresponded to unsuccessful landings and 12 denoted successful landings.

# Conclusions

**Project Overview:** This project focuses on a comprehensive analysis of SpaceX launch data, aiming to gain insights into mission success rates, contributing factors, and predictive patterns. By leveraging various analytical techniques and data visualization tools, the project delves into the correlation between different variables and their impact on launch outcomes. The ultimate goal is to build a predictive model that can accurately forecast the success of SpaceX missions based on historical data.

**Key Findings:**

1. **Experience and Success Rate:** A clear relationship between the number of flights and success rate at launch sites is evident. As experience accumulates, success rates tend to increase, indicating the learning curve of SpaceX over time.
2. **Temporal Trends:** A temporal analysis highlights distinct periods of success and challenges. Notably, the years 2010 to 2013 witnessed a lack of successful landings, followed by a general rise in success rates post-2013, punctuated by minor fluctuations.
3. **Orbit Types and Payloads:** Certain orbit types, such as ES-L1, GEO, HEO, and SSO, exhibit remarkable 100% success rates, often attributed to the limited number of flights to these orbits. The study also reveals that some orbit types have higher success rates with heavier payloads.
4. **Launch Site Influence:** Launch site choice significantly affects success rates. KSC LC-39 A emerges as the most successful site, boasting the highest number of successful launches and a commendable success rate.
5. **Massive Payload Challenges:** The analysis suggests a decline in success rates for launches with massive payloads (over 4000kg), pointing towards potential challenges associated with heavier loads.

**Conclusions:** The project successfully showcases the importance of data analysis and predictive modeling in understanding complex systems like space missions. The revealed patterns, such as the impact of launch sites, payload masses, and temporal trends, can inform SpaceX's decision-making process and help improve mission success rates. The predictive model, specifically the Decision Tree model, provides a robust foundation for forecasting mission outcomes, allowing SpaceX to optimize resource allocation and enhance planning strategies. Overall, this analysis contributes valuable insights to the space exploration domain, emphasizing the significance of data-driven insights in shaping successful missions.

# Thank you!