IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Francisco Ricardo Andraschko
Feb 2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - Exploratory Data Analysis (EDA) with Data Visualization
  - Exploratory Data Analysis (EDA) with SQL
  - Building an interactive map with Folium
  - Building a Dashboard with Plotly Dash
  - Predictive analysis (Classification)
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

Introduction

- Project background and context

  SpaceX has emerged as a pioneering leader in the modern commercial space era by making space travel significantly more affordable. With Falcon 9 launches priced at around 62 million dollars — compared to over 165 million dollars for competitors—the company's ability to reuse the first stage is a game changer. This reusability not only drives down costs but also provides a competitive edge in the aerospace market.

  This project leverages public data and machine learning models to predict whether the Falcon 9 first stage will successfully land. By accurately forecasting landing outcomes, we can better estimate launch costs and gain insights into the factors influencing a safe recovery.

- Problems you want to find answers
  - What are the main characteristics that distinguish successful from failed landings?
  - How do variables like payload mass, launch site, number of flights, and orbital conditions affect landing success?
  - Has the success rate of first stage landings improved over the years?
  - What is the most effective binary classification algorithm for predicting first stage reuse?

Section 1

Methodology

Methodology

- Data collection methodology
  - Utilized SpaceX REST API and web scraping from Wikipedia.
- Data wrangling
  - Filtered the data, dropped unnecessary columns, and handled missing values.
- Data Preparation
  - Applied One Hot Encoding for binary classification.
- Exploratory Data Analysis (EDA)
  - Performed analysis using visualization techniques and SQL.
- Interactive Visual Analytics
  - Employed Folium and Plotly Dash for interactive visualizations.
- Predictive Analysis
  - Built, tuned, and evaluated classification models to ensure optimal results.

Data Collection

Our data collection process integrated two complementary approaches to build a comprehensive dataset for analysis:

- **SpaceX REST API:**
  We accessed the SpaceX REST API (api.spacexdata.com/v4/) to retrieve a JSON file containing detailed launch information (e.g., flight numbers, dates, booster versions, payload mass, etc.). The JSON data was then transformed into a dataframe, cleaned, and exported.

- **Wikipedia Web Scraping:**
  We scraped the Falcon 9 and Falcon Heavy launches page on Wikipedia to extract additional launch details (e.g., launch site, payload, customer, launch outcome, etc.). The HTML content was processed using BeautifulSoup, converted into a dataframe, and exported.

Data Collection – SpaceX API

1. API Call

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
```

2. Receive JSON File

```
data = pd.json_normalize(response.json())
```

3. Transform Data

```
getBoosterVersion(data)
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

4. Create Dictionary

```
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 }
```

5. Create Dataframe

```
data = pd.DataFrame(launch_dict)
```

6. Filter Dataframe

```
data_falcon9 = data[
        data['BoosterVersion'] != 'Falcon 1']
```

7. Export CSV

```
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

GitHub URL -> Data Collection API

Data Collection - Scraping

1. Access URL

```
static_url = "https://en.wikipedia.org/w/index.php?
title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

2. Retrieve HTML Response

```
response = requests.get(static_url)
```

3. Transform Data

```
soup = BeautifulSoup(response.text, 'html.parser')
```

4. Create Dictionary

```
launch_dict= dict.fromkeys(column_names)
# Remove an irrelvant 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']=[]
```

5. Create Dataframe

```
df= pd.DataFrame(
    { key:pd.Series(value) for key, value in launch_dict.items() })
```

6. Export CSV

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

GitHub URL -> Data Collection Web Scraping

Data Wrangling

Our data processing involved transforming raw landing outcome strings into a binary format and performing exploratory analysis. We handled cases where the booster landing outcome was recorded as strings such as "True Ocean," "True RTLS," and "True ASDS" (indicating success) and their corresponding false values (indicating failure). These outcomes were converted into categorical training labels where “1” signifies a successful landing and “0” a failure.

Additionally, we performed exploratory data analysis (EDA) by:

- Calculating the number of launches at each site
- Counting occurrences of each orbit type
- Analyzing mission outcomes by orbit
- Creating a new landing outcome label from the Outcome column
- Exporting the processed data to CSV

GitHub URL -> Data Wrangling

Data Wrangling

1. Calculating the number of launches at each site

```
launch_site_counts = df['LaunchSite'].value_counts()
print(launch_site_counts)
```

```
LaunchSite
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: count, dtype: int64
```

11

2. Counting occurrences of each orbit type

```
orbit_counts = df['Orbit'].value_counts()
print(orbit_counts)
```

```
Orbit
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
HEO       1
ES-L1     1
SO        1
GEO       1
Name: count, dtype: int64
```

3. Analyzing mission outcomes by orbit

```
landing_outcomes = df['Outcome'].value_counts()
print(landing_outcomes)
```

```
Outcome
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: count, dtype: int64
```

4. Creating a new landing outcome label from the Outcome column

```
landing_class = df['Outcome'].apply(lambda x: 0 if x in bad_outcomes else 1).tolist()
```

```
print(landing_class)
```

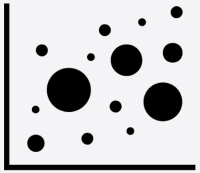
```
[0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
```

5. Export CSV

```
df.to_csv("dataset_part_2.csv", index=False)
```

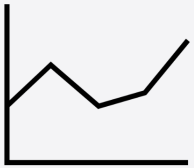
GitHub URL -> Data Wrangling

EDA with Data Visualization



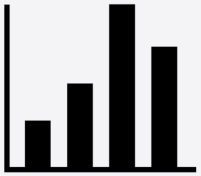
Scatter plots reveal relationships (correlations) between numerical variables, helping to understand how factors like flight number, payload mass, launch site, and orbit type interact.

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload vs. Launch Site
- Orbit vs. Flight Number
- Payload vs. Orbit Type
- Orbit vs. Payload Mass



Line graphs track trends over time, providing insight into how the success rate evolves year by year, which is useful for predicting future performance.

- Success Rate vs. Year



Bar graphs compare numerical data against categorical variables (e.g., orbit types) to highlight differences in success rates.

- Success Rate vs. Orbit

GitHub URL -> EDA with Data Visualization

EDA with SQL

The SQL queries performed were:

- Retrieved a distinct list of launch sites from the mission dataset.
- Selected five records where the launch site names begin with 'CCA'.
- Summed the payload mass carried by boosters launched by NASA (CRS).
- Calculated the average payload mass for booster version F9 v1.1.
- Determined the earliest date of a successful ground pad landing.
- Listed booster versions that achieved successful drone ship landings with payload masses between 4000 and 6000.
- Counted the total number of successful versus unsuccessful mission outcomes.
- Identified the booster versions that carried the maximum payload mass using a subquery.
- Extracted records for 2015 showing failed drone ship landings, along with corresponding booster versions and launch sites.
- Ranked landing outcome counts (e.g., Failure (drone ship), Success (ground pad)) between 2010-06-04 and 2017-03-20 in descending order.

GitHub URL -> EDA with SQL

Build an Interactive Map with Folium

Integrating diverse mapping elements, our interactive Folium map offers a comprehensive view of launch site distribution, performance outcomes, and geographical context—all within one dynamic map.

- **Markers & Circles:**
  Placed at each launch site (with labels/popups) to clearly identify locations.
- **Marker Clusters:**
  Grouped nearby markers to reduce clutter in dense areas.
- **Colored Icons:**
  Used **green** for successes and **red** for failures to quickly convey landing outcomes.
- **Lines:**
  Connected launch sites to nearby landmarks (e.g., railways, highways, cities) to show spatial relationships.

GitHub URL -> Interactive Map w/ Folium

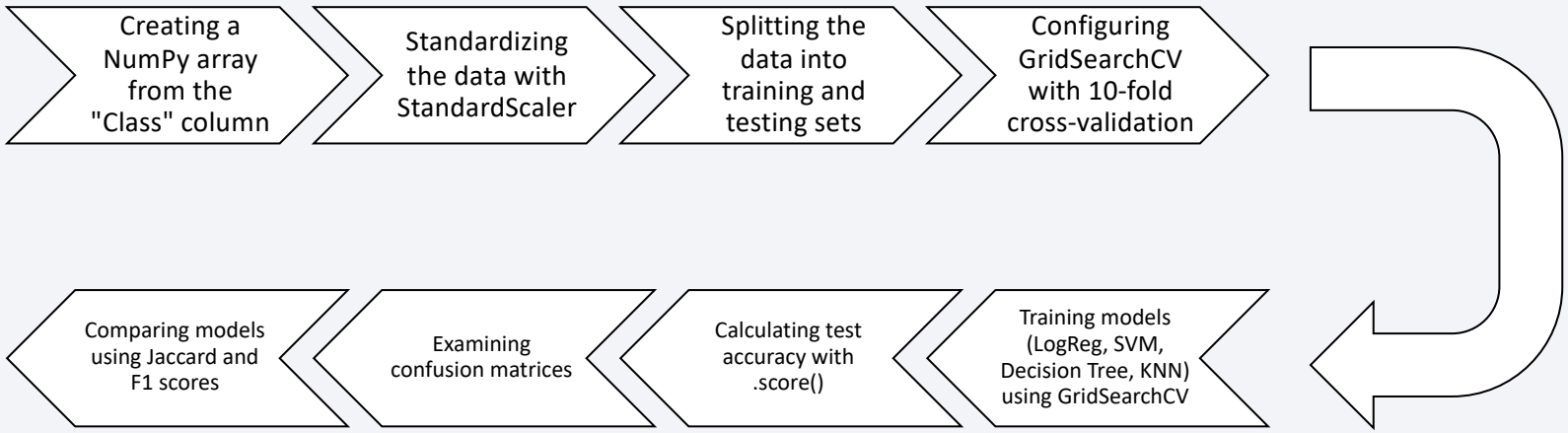
Build a Dashboard with Plotly Dash

An interactive dashboard was created to dynamically explore and analyze launch data:

- **Launch Sites Dropdown:**
  Provides selection of a specific launch site or all sites (using dash_core_components.Dropdown).
- **Success Launches Pie Chart:**
  Displays overall success counts and a breakdown of success versus failure for the chosen launch site (using plotly.express.pie).
- **Payload Mass Range Slider:**
  Allows filtering of data based on a specified payload mass range (using dash_core_components.RangeSlider).
- **Payload Mass vs. Success Rate Scatter Chart:**
  Visualizes the correlation between payload mass and launch success across different booster versions (using plotly.express.scatter).

GitHub URL -> Ploty Dash

Predictive Analysis (Classification)



GitHub URL -> Machine Learning Prediction

Results

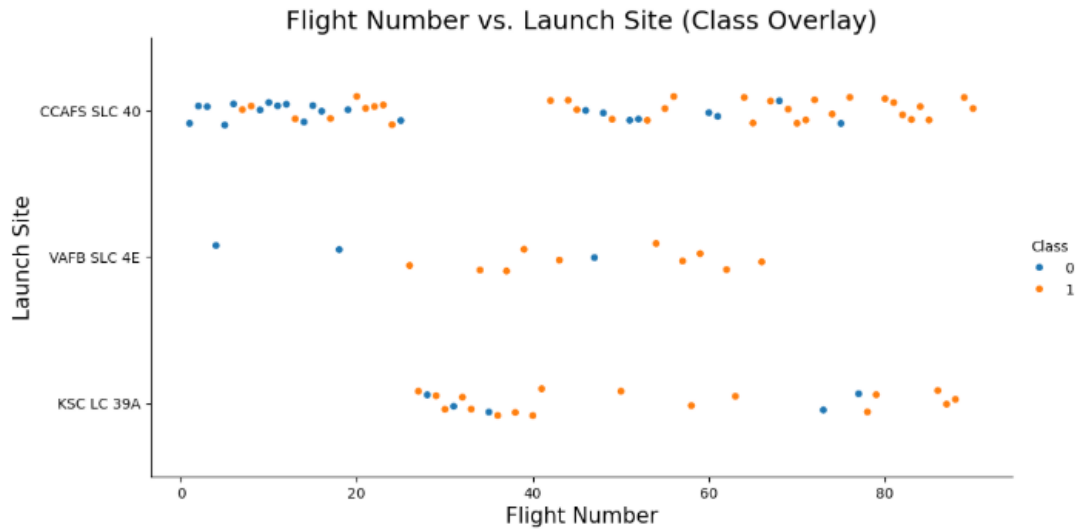
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

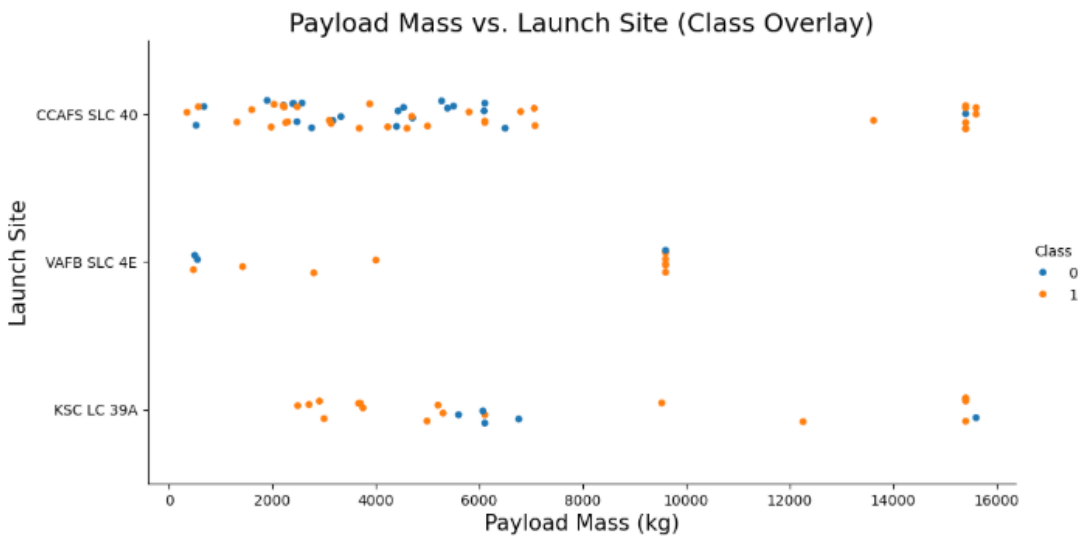
Flight Number vs. Launch Site

- This plot shows flight numbers against launch sites. Early flights generally failed, while later flights succeeded more often. Also, CCAFS SLC 40 is the most frequently used launch site.



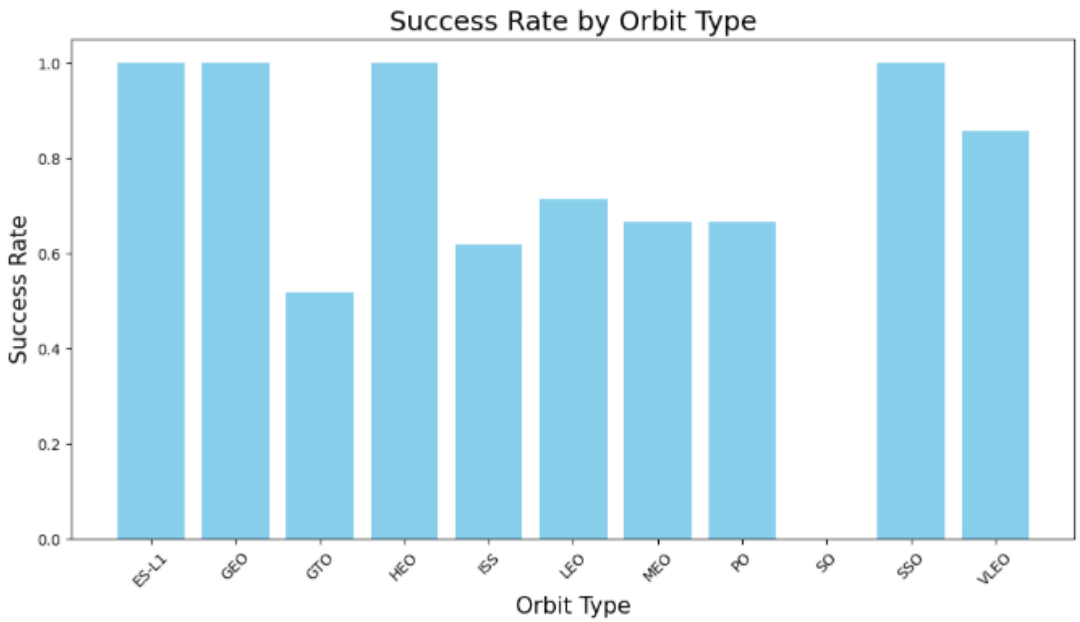
Payload vs. Launch Site

- This chart displays payload mass for each launch site. Higher payloads tend to correlate with increased success, and sites like KSC LC 39A show strong performance even with lighter payloads.



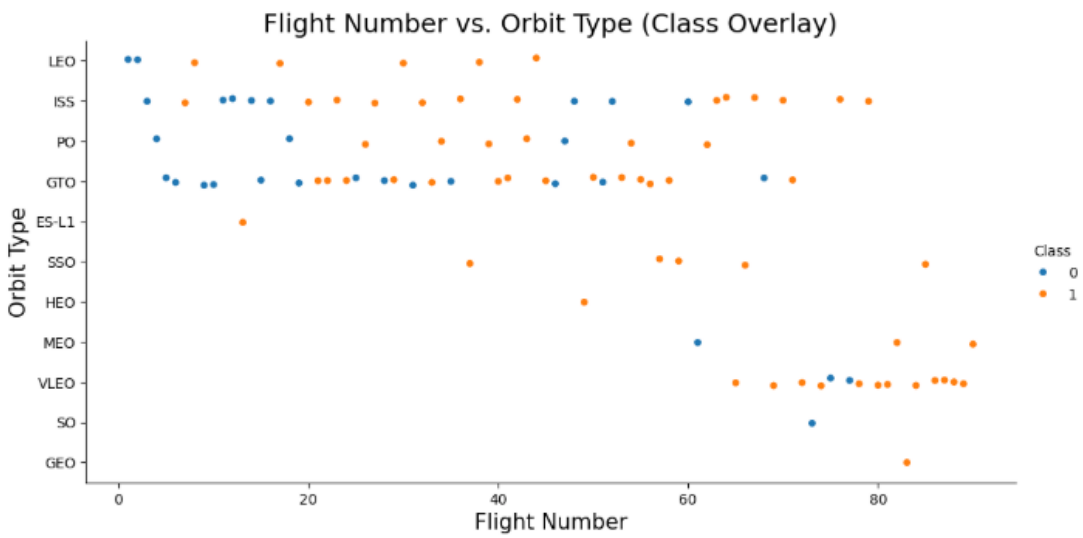
Success Rate vs. Orbit Type

- Here, average success rates are compared across orbit types. Orbits such as ES-L1, GEO, HEO, and SSO reach 100% success, while SO shows very low performance. Other orbits fall between 50% and 85%.



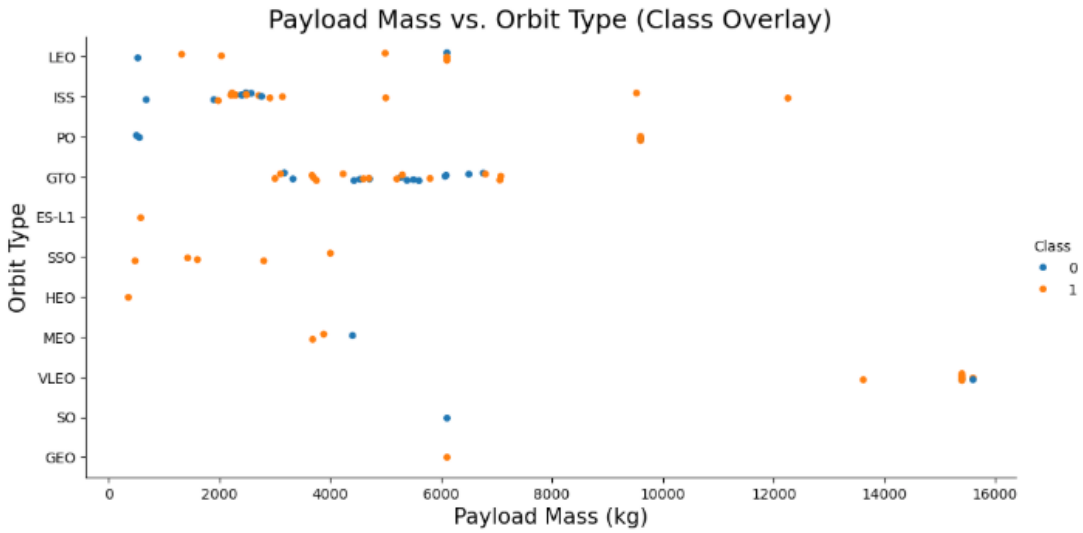
Flight Number vs. Orbit Type

- This scatter plot examines flight numbers in relation to orbit types. In LEO, success appears to improve with more flights, but no clear trend is observed for orbits like GTO.



Payload vs. Orbit Type

- This visualization relates payload mass to orbit type. It indicates that heavy payloads negatively impact GTO success, while they favor better outcomes in Polar and LEO (ISS) orbits.



Launch Success Yearly Trend

- This line chart tracks the annual average success rate, showing a steady increase from 2013 up to 2020.

All Launch Site Names

SQL Query

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE WHERE "Launch_Site" IS NOT NULL;
```

Results

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

Explanation:

- This query extracts distinct launch site names from the dataset, providing a complete list of all locations where launches occurred.

Launch Site Names Begin with 'CCA'

SQL Query

```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%'LIMIT 5;
```

Results

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

Explanation:

- Displays 5 records where launch site names start with 'CCA'. This helps verify data consistency for sites like CCAFS SLC-40.

Total Payload Mass

SQL Query

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS Total_Payload_Mass FROM SPACEXTABLE WHERE "Customer" LIKE '%NASA (CRS)%';
```

Results

Total_Payload_Mass
48213

Explanation:

- Calculates the sum of payload masses for boosters operated by NASA (CRS), offering an aggregate measure of payload capacity for these missions.

Average Payload Mass by F9 v1.1

SQL Query

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") AS Average_Payload_Mass FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

Results

Average_Payload_Mass
2928.4

Explanation:

- Computes the average payload mass for launches using booster version F9 v1.1, providing insight into typical payload sizes for this version.

First Successful Ground Landing Date

SQL Query

```
%sql SELECT MIN(Date) AS First_Successful_Ground_Pad_Landing
                    FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';
```

Results

First_Successful_Ground_Pad_Landing
2015-12-22

Explanation:

- Uses the MIN function to identify the earliest date of a successful ground pad landing, marking a key operational milestone.

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL Query

```
%sql SELECT "Booster_Version" FROM SPACEXTABLE
            WHERE "Landing_Outcome" = 'Success (drone ship)'
            AND "PAYLOAD_MASS__KG_" > 4000  AND "PAYLOAD_MASS__KG_" < 6000;
```

Results

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

Explanation:

- Lists booster versions that achieved successful drone ship landings with payload masses between 4000 and 6000, highlighting specific performance cases.

Total Number of Successful and Failure Mission Outcomes

SQL Query

```
%sql SELECT "Mission_Outcome", COUNT(*) AS Count FROM SPACEXTABLE GROUP BY "Mission_Outcome";
```

Results

Mission_Outcome	Count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Explanation:

- Groups and counts mission outcomes to show the overall numbers of successes versus failures across all missions.

Boosters Carried Maximum Payload

SQL Query

```
%sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (
                    SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE);
```

Results

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

Explanation:

- Identifies the booster versions that carried the maximum payload mass using a subquery, pinpointing record-holders in payload capacity.

2015 Launch Records

SQL Query

```
%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)';
```

Results

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

Explanation:

- Extracts records for 2015 detailing failed drone ship landings, including booster versions and launch site names, to uncover patterns in these failures.

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

SQL Query

```
%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;
```

Results

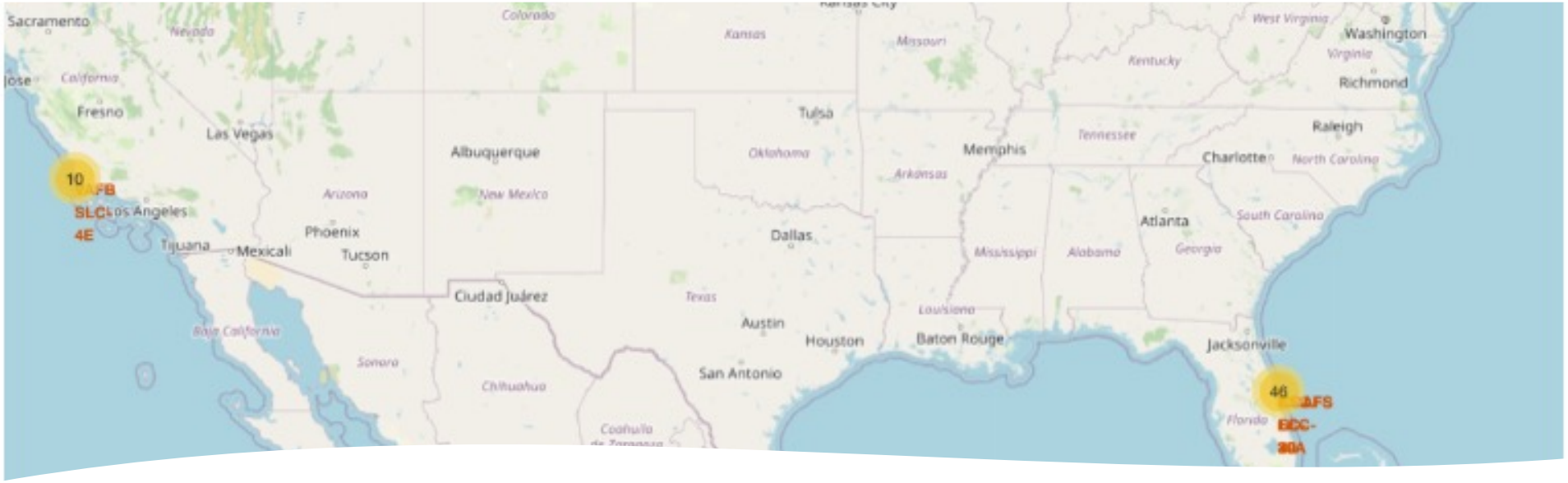
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Explanation:

- Ranks landing outcomes by their frequency within the specified date range, highlighting the most common outcomes in descending order.

Section 3

Launch Sites Proximities Analysis

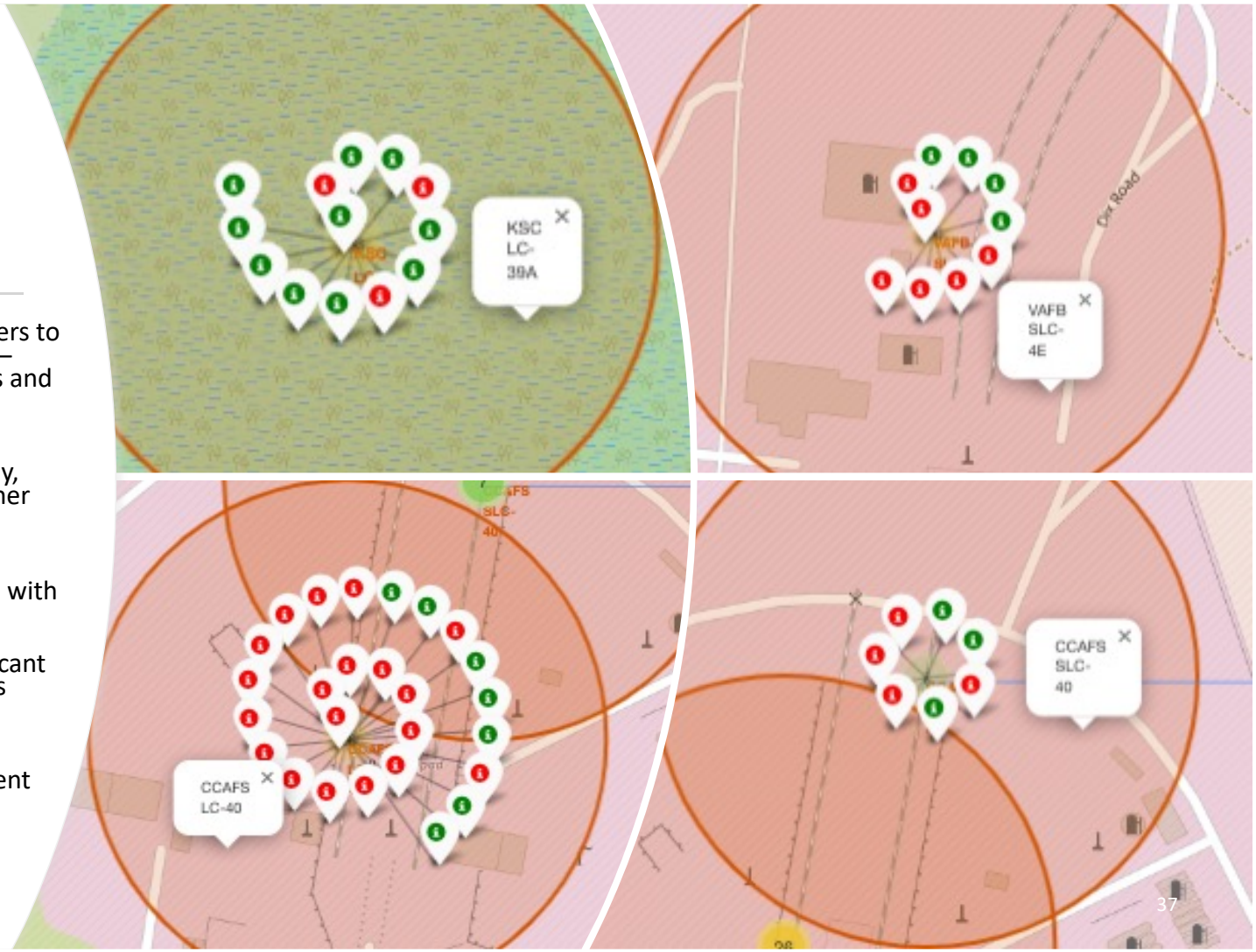


Global Map of Launch Sites

- This global map displays all launch sites using markers and circles with labels and popups. Marker clusters help reduce clutter in dense regions, making it easy to identify the geographic distribution of launch sites.

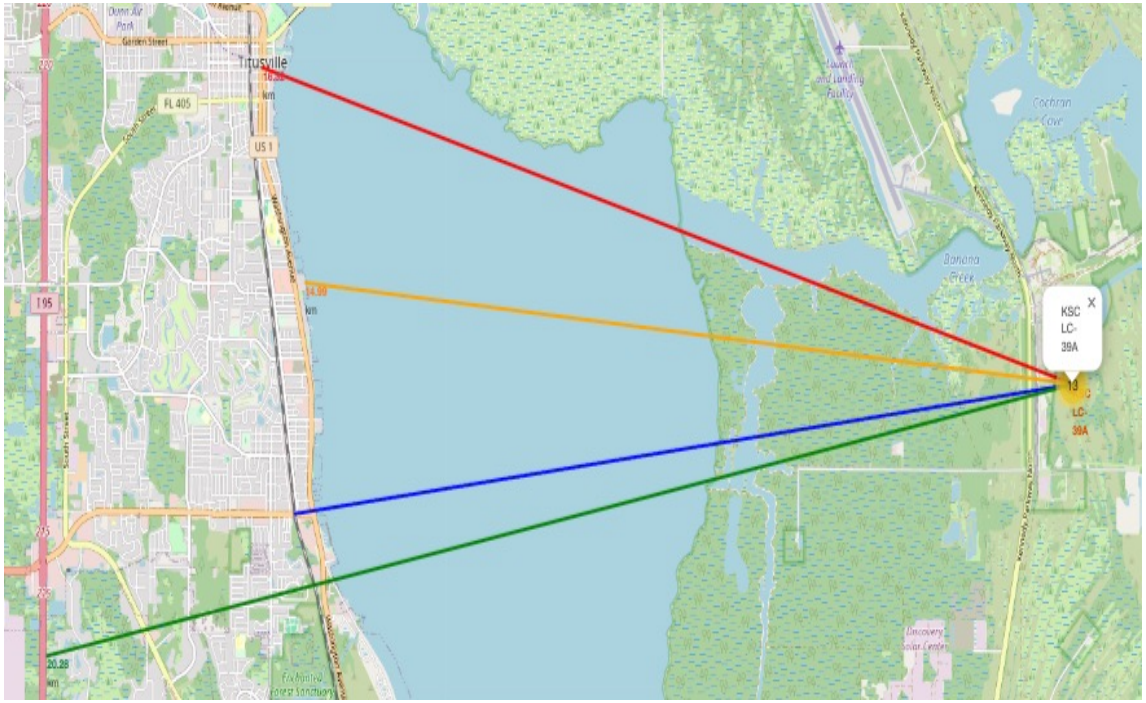
Colored Markers of Launch Outcomes

- This view uses colored markers to represent launch outcomes—green for successful landings and red for failures. This visual differentiation allows quick assessment of performance across different sites. Notably, the following sites show higher occurrences:
- CCAFS SLC-40: Exhibits the highest number of launches, with a mix of outcomes.
- VAFB SLC-4E: Displays significant activity with a strong success rate.
- KSC LC-39A: Also shows high launch frequency and excellent success performance.



Proximity Analysis of a Selected Launch Site

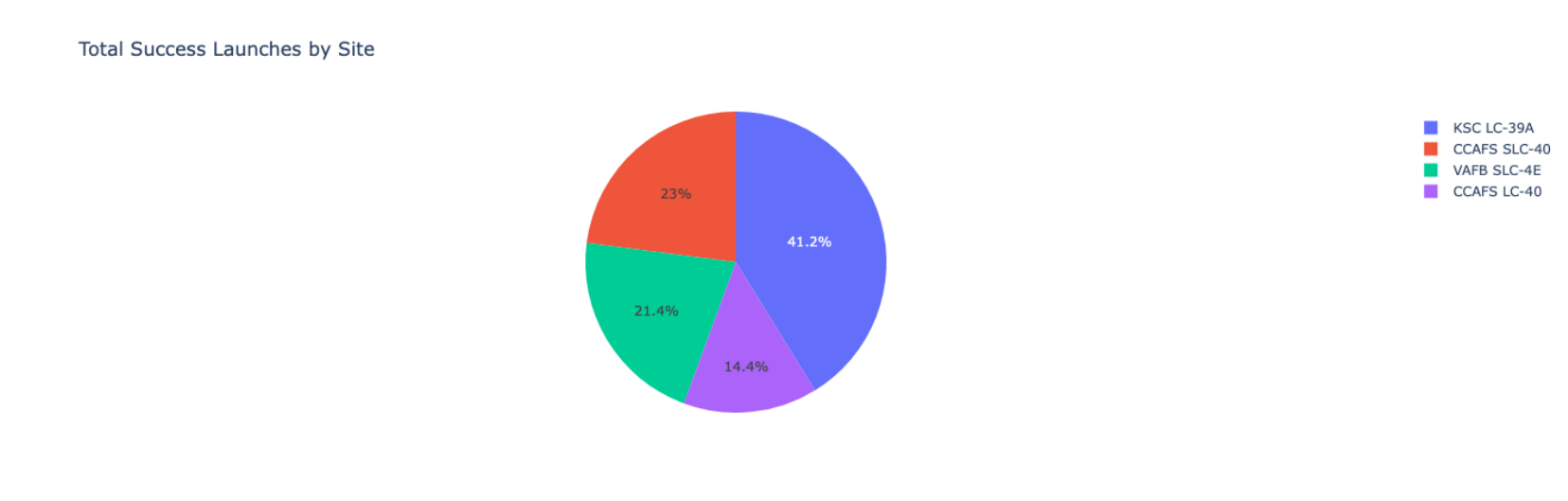
- From the visual analysis of launch site KSC LC-39A, the following distances were observed:
  - Railway: 15.23 km
  - Highway: 20.28 km
  - Coastline: 14.99 km
  - Nearest city (Titusville): 16.32 km
- Given that a failed rocket can cover 15-20 km in a few seconds, these proximities highlight a potential risk to populated areas.



Section 4

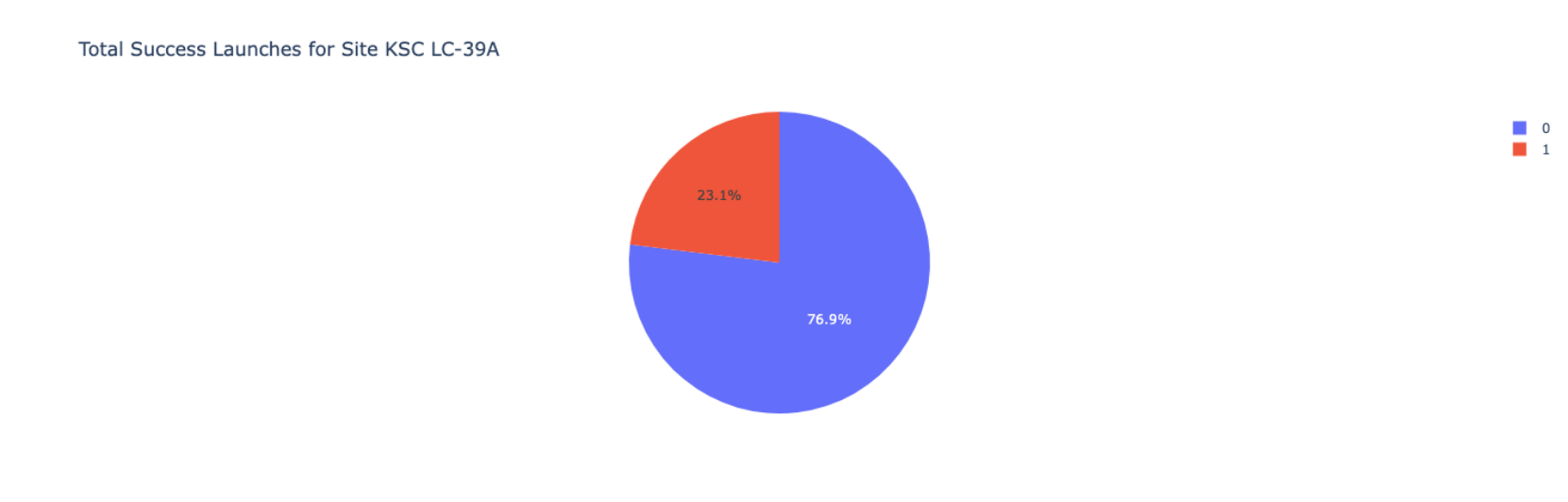
Build a Dashboard with Plotly Dash

Dashboard – Overall Launch Success



Displays the total number of successful launches across all sites, highlighting which sites contribute most to the overall success count.

Dashboard – Top Site Success Ratio



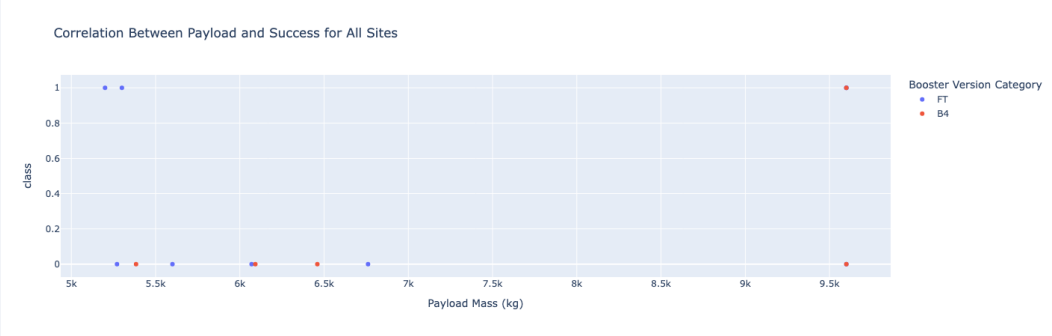
Focuses on KSC LC-39A, the site with the highest success rate, showing the breakdown of successes (76.9%) vs. failures (23.1%).

Dashboard – Payload vs. Launch Outcome

- For payloads from 0 to 5000 kg, the success rate is high and consistent. In contrast, the 5000 to 10000 kg range shows greater variability, indicating heavier payloads pose more challenges for successful landings.

Low weighted payload (0 – 5000 kg)

Heavy weighted payload (5000 – 10000 kg)



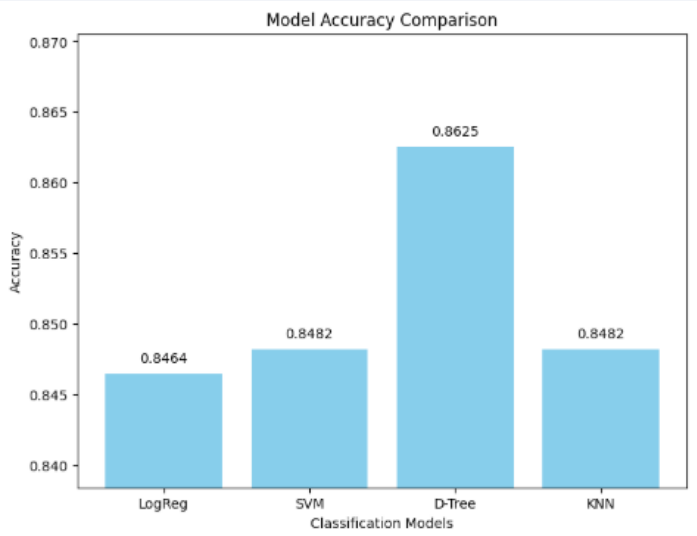
Section 5

Predictive Analysis (Classification)

Classification Accuracy

```
model_scores = {
    "Logistic Regression": logreg_cv.best_score_,
    "Support Vector Machine": svm_cv.best_score_,
    "Decision Tree": tree_cv.best_score_,
    "K-Nearest Neighbors": knn_cv.best_score_
}
best_model = max(model_scores, key=model_scores.get)
print("Model performances:", model_scores)
print(f"The best performing model is: {best_model} with accuracy {model_scores[best_model]:.4f}")

Ans.:
Model performances: {
  'Logistic Regression': 0.8464285714285713,
  'Support Vector Machine': 0.8482142857142856,
  'Decision Tree': 0.8625,
  'K-Nearest Neighbors': 0.8482142857142858}
The best performing model is: Decision Tree with accuracy 0.8625
```

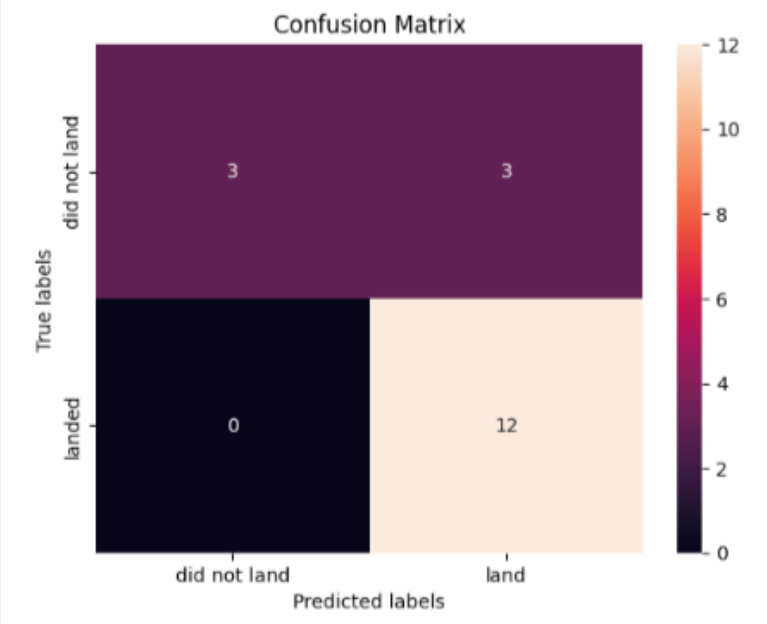


- This bar chart compares the accuracy of the four classification models. The Decision Tree model achieved the highest accuracy at 86.25%, suggesting it is the best performer for predicting Falcon 9 first stage landing outcomes.

Confusion Matrix

		Actual class	
		P	N
Predicted class	P	TP	FP
	N	FN	TN

- The confusion matrix for the Decision Tree model shows high true positive and true negative counts, with very few false positives and false negatives. This indicates robust performance and reliable prediction of landing outcomes.



Conclusions

- The Decision Tree model outperformed other classifiers, showing the highest accuracy and strong predictive performance.
- Launches with lower payload masses (especially between 2000–5000 kg) tend to be more successful, while heavier payloads (5000–10000 kg) exhibit greater variability in outcomes.
- Most launch sites are strategically located near the Equator and the coast, with KSC LC-39A demonstrating the highest success rate.
- Certain orbits (ES-L1, GEO, HEO, SSO) consistently achieve 100% success, highlighting the importance of orbital selection.
- An overall trend of increasing success over the years suggests that accumulated operational experience significantly contributes to better outcomes.

Thank you!