Data Science Capstone Project

Micko Kok

10/11/2025

Outline

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

Executive Summary

Executive Summary

1. **Summary of Methodologies**
    a. Data Collection
    b. Data Wrangling
    c. Exploratory Data Analysis with Data Visualisation
    d. Exploratory Data Analysis with SQL
    e. Building an Interactive Map with Folium
    f. Building a Dashboard with Plotly Dash
    g. Predictive Analysis
2. **Summary of all Results**
    a. Exploratory Data Analysis on the Results
    b. Interactive Analytics Demo Screenshots
    c. Predictive Analysis Results

Introduction

Introduction

Project Background

SpaceX is one of the most successful companies in the commercial space industry, known for significantly reducing the cost of space travel. A typical Falcon 9 rocket launch costs about $62 million, compared to over $165 million charged by other launch providers. This major cost reduction is largely due to SpaceX's ability to reuse the rocket's first stage after launch.

In this project, we aim to predict whether the first stage of a Falcon 9 rocket will successfully land and be reused. By using publicly available data and machine learning techniques, we can estimate the likelihood of first-stage recovery. Accurately predicting reusability can help estimate launch costs and provide valuable insights into the efficiency and sustainability of commercial space missions.

Introduction

Questions to Explore

1. What is the best binary classification algorithm that can be utilised to determine the cost of a launch?
2. Is the rate of successful landings annually incremental?
3. How do variables such as payload, mass, launch site, number of flights, and ortbits affect the success rate of the first stage landings?

Methodology

Methodology

1. **Data collection methodology:**
    a. Using SpaceX Rest API
    b. Using Web Scrapping from Wikipedia
2. **Performed data wrangling**
    a. Filtering the data
    b. Dealing with missing values
    c. Using One Hot Encoding to process the data for binary classification
3. **Performed exploratory data analysis (EDA)**
    a. Visualization
    b. SQL
4. **Performed interactive visual analytics**
    a. Folium
    b. Plotly Dash
5. **Performed predictive analysis using classification models**
    a. Building, tuning and evaluation of classification models to ensure the best results

Data Collection

The data collection process involved a combination of both:

- API requests from the SpaceX REST API
- Data scraped directly from SpaceX's Wikipedia entry.

Both of these data collection methods were vital to obtain all the **necessary data** pertaining to the **launches**, in order to conduct a more **detailed analysis further on**.

SpaceX Rest API Data Columns

- FlightNumber
- Date
- BoosterVersion
- PayloadMass
- Orbit
- LaunchSite
- Outcome
- Flights
- GridFins

- Reused
- Legs
- LandingPad
- Block
- ReusedCount
- Serial
- Longitude
- Latitude

SpaceX Rest API Data Collection Pipeline

Request data from the SpaceX API

Sent an HTTP GET request to the SpaceX REST API endpoint containing rocket launch information.

Decode the response content

Used .json() to parse the API response and convert it into a Python dictionary.

Normalize nested JSON data

Applied .json_normalize() to flatten nested JSON structures into a tabular format.

Extract relevant launch information

Created and applied custom functions to retrieve specific attributes.

Construct a clean data dictionary

Organized the extracted fields into a dictionary.

Create a pandas DataFrame

Converted the dictionary into a DataFrame.

Filter for Falcon 9 launches

Applied conditional filtering to keep only rows where the rocket type was "Falcon 9".

Handle missing values

Filled missing values in the Payload Mass (kg) column with the column's calculated mean using .fillna(df["PayloadMass"].mean()).

Export the cleaned dataset

Saved the processed and filtered dataset as a .csv file for further analysis.

GitHub Source: DataCollection_API

SpaceX’s Wikipedia Web Scraped Data Columns

- Flight No.
- Launch site
- Payload
- PayloadMass
- Orbit
- Customer
- Launch outcome
- Version Booster
- Booster landing

- Date
- Time

SpaceX’s Wikipedia Web Scraped Data Collection Pipeline

Request HTML content from Wikipedia

Sent an HTTP GET request to the Wikipedia page containing the Falcon 9 launch history table.

Create a BeautifulSoup object

Parsed the HTML response using BeautifulSoup to easily navigate and extract elements from the webpage’s DOM.

Extract column names from the HTML table header

Identified the main launch data table and extracted all column headers to define the structure of the dataset.

Collect table data

Iterated through table rows, extracted data from each cell, cleaned and formatted the raw text values.

Construct a data dictionary

Organized the extracted fields into a dictionary.

Create a pandas DataFrame

Converted the dictionary into a DataFrame.

Export the cleaned dataset

Saved the processed and filtered dataset as a .csv file for further analysis.

GitHub Source: DataCollection_WebScraping

Data Wrangling

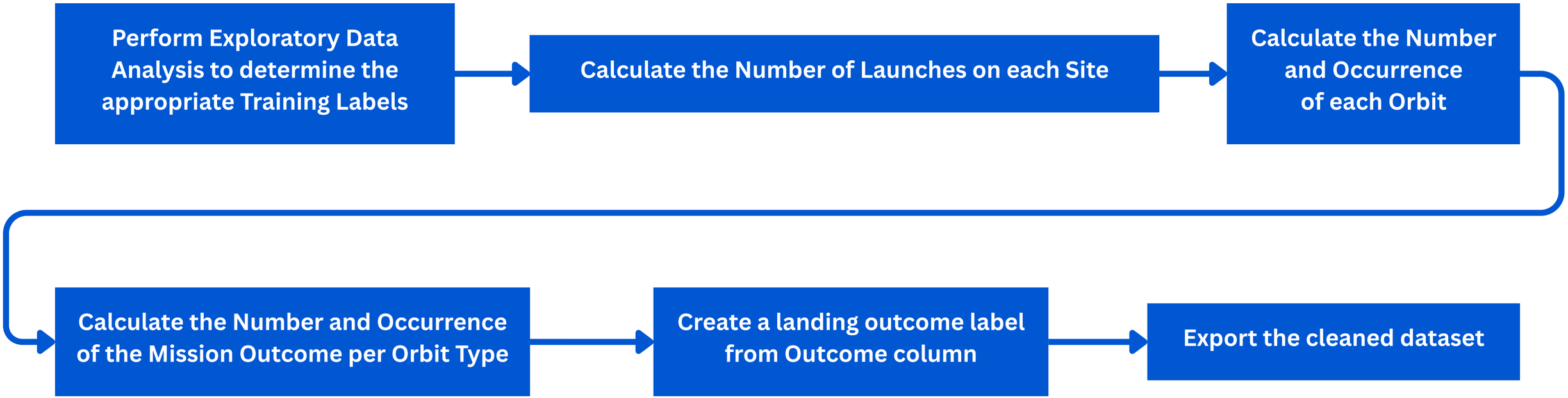
In the dataset, there are multiple cases where the rocket booster did not land successfully. Each record includes a landing outcome describing the type and result of the landing attempt. For example:

- **True Ocean:** indicates a successful landing in a designated ocean region, while **False Ocean** indicates an unsuccessful attempt in the same context.
- **True RTLS (Return To Launch Site):** denotes a successful ground pad landing, whereas **False RTLS** represents a failed ground pad landing.
- **True ASDS (Autonomous Spaceport Drone Ship):** indicates a successful landing on a drone ship, and **False ASDS** indicates an unsuccessful one.

For modeling purposes, these categorical outcomes were converted into binary training labels:

- 1 = successful booster landing
- 0 = unsuccessful booster landing

Data Wrangling Pipeline



GitHub Source: DataWrangling

EDA with Data Visualisation

Plotted Charts

- **Flight Number vs. Payload Mass**
- **Flight Number vs. Launch Site**
- **Payload Mass vs. Launch Site**
- **Orbit Type vs. Success Rate**
- **Flight Number vs. Orbit Type**
- **Payload Mass vs Orbit Type**
- **Success Rate Yearly Trend**

Chart Types and Purpose

- **Scatter plots:** show the relationship between variables. If a correlation exists, these features could be used in machine learning models.
- **Bar charts:** compare discrete categories, highlighting the relationship between specific categories and their measured values.
- **Line charts:** display trends over time, making them useful for analyzing time series data.

GitHub Source: EDA_DataVisualisation

EDA with SQL

SQL Queries Performed

- Display the names of unique launch sites in the space mission dataset.
- Display 5 records where launch sites begin with the string 'CCA'.
- Calculate the total payload mass carried by boosters launched by NASA (CRS).
- Calculate the average payload mass carried by the booster version F9 v1.1.
- List the date of the first successful landing on a ground pad.

- List the names of boosters that were successfully landed on a drone ship and had a payload mass between 4000 and 6000 kg.
- Count the total number of successful and failed mission outcomes.
- List the names of booster versions that have carried the maximum payload mass.
- List failed landing outcomes on drone ships, including booster versions and launch site names, for the months in 2015.
- Rank the count of landing outcomes (e.g., Failure on drone ship or Success on ground pad) between 2010-06-04 and 2017-03-20 in descending order.

GitHub Source: EDA_SQLite

Building an Interactive Map with Folium

Markers of All Launch Sites:

- Added a marker with a circle, popup label, and text label for NASA Johnson Space Center, using its latitude and longitude coordinates as the initial map location.
- Added markers with circles, popup labels, and text labels for all launch sites, displaying their geographical locations and proximity to the Equator and nearby coasts.

Colored Markers for Launch Outcomes:

- Added colored markers to indicate launch success (green) or failure (red).
- Used Marker Clusters to identify launch sites with relatively high success rates.

Distances Between Launch Sites and Nearby Features:

- Added colored lines to show distances between the launch site VAFB SLC-4E and nearby features such as railways, highways, coastlines, and the closest city.

GitHub Source: InteractiveVisualAnalytics_Folium

Building a Dashboard with Plotly Dash

Launch Sites Dropdown List

- Added a dropdown menu to enable selection of a specific launch site.

Pie Chart Showing Successful Launches

- Added a pie chart displaying the total count of successful launches across all sites.
- When a specific launch site is selected, the chart shows the proportion of successful vs. failed launches for that site.

Slider for Payload Mass Range

- Added a slider to filter launches by payload mass range.

Scatter Chart of Payload Mass vs. Success Rate for Different Booster Versions

- Added a scatter chart to visualize the correlation between payload mass and launch success, highlighting differences across booster versions.

GitHub Source: PlotlyDash_Dashboard

Predictive Analysis Pipeline

Created a NumPy array from the “Class” column in the dataset

Standardized the data using StandardScaler, then fitted and transformed the features

Split the dataset into training and testing sets using the train_test_split() function.

Created a GridSearchCV object with cv = 10 to find the best hyperparameters.

Create a landing outcome label from Outcome column

Applied GridSearchCV on Logistic Regression (LogReg), Support Vector Machine (SVM), Decision Tree, and K-Nearest Neighbors (KNN) models.

Calculated the accuracy on the test data using the .score() method for all models.

Examined the confusion matrix for each model to evaluate classification performance.

Compared models using Jaccard Score and F1 Score metrics to determine which method performed best.

GitHub Source: Predictive Analysis

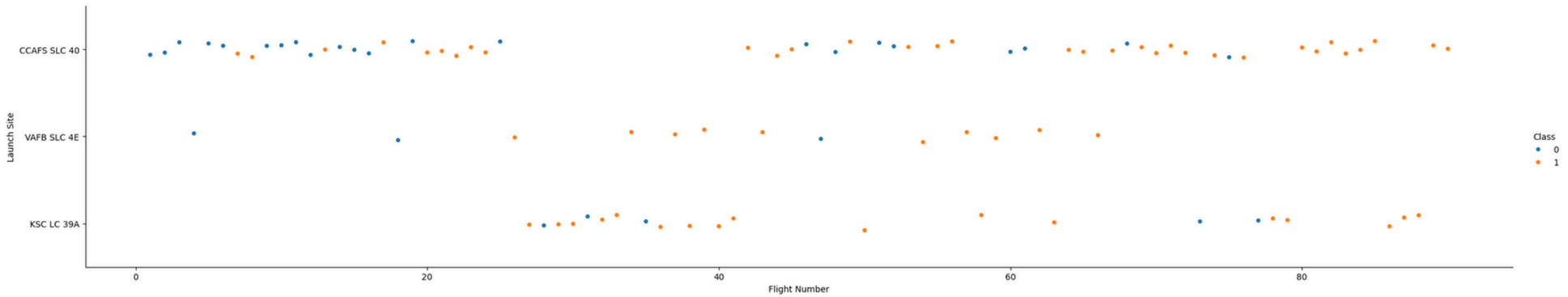
Results

Results

1. **Exploratory Data Analysis Results**
    a. EDA with Data Visualisation
    b. EDA with SQL
2. **Interactive Analytics Demo Results**
    a. Interactive Map with Folium
    b. Plotly Dashboard
3. **Predictive Analysis Results**

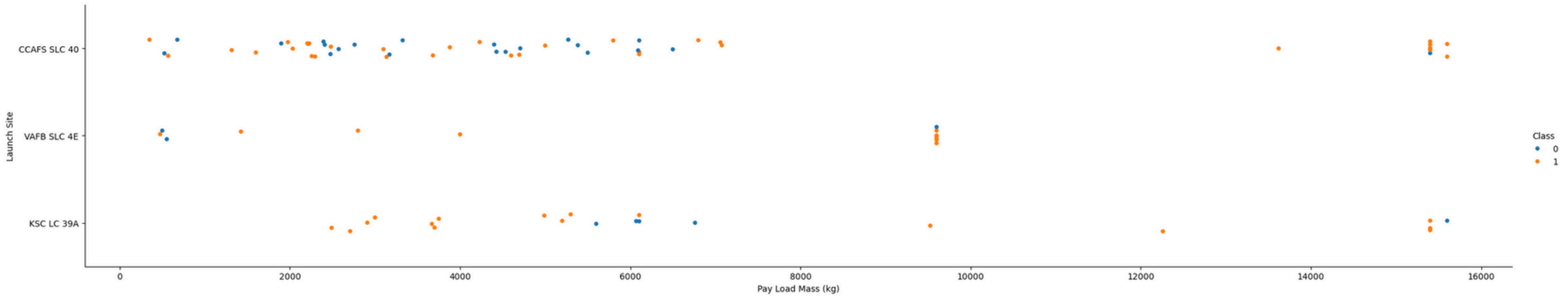
Exploratory Data Analysis Results

Flight Number vs Launch Site



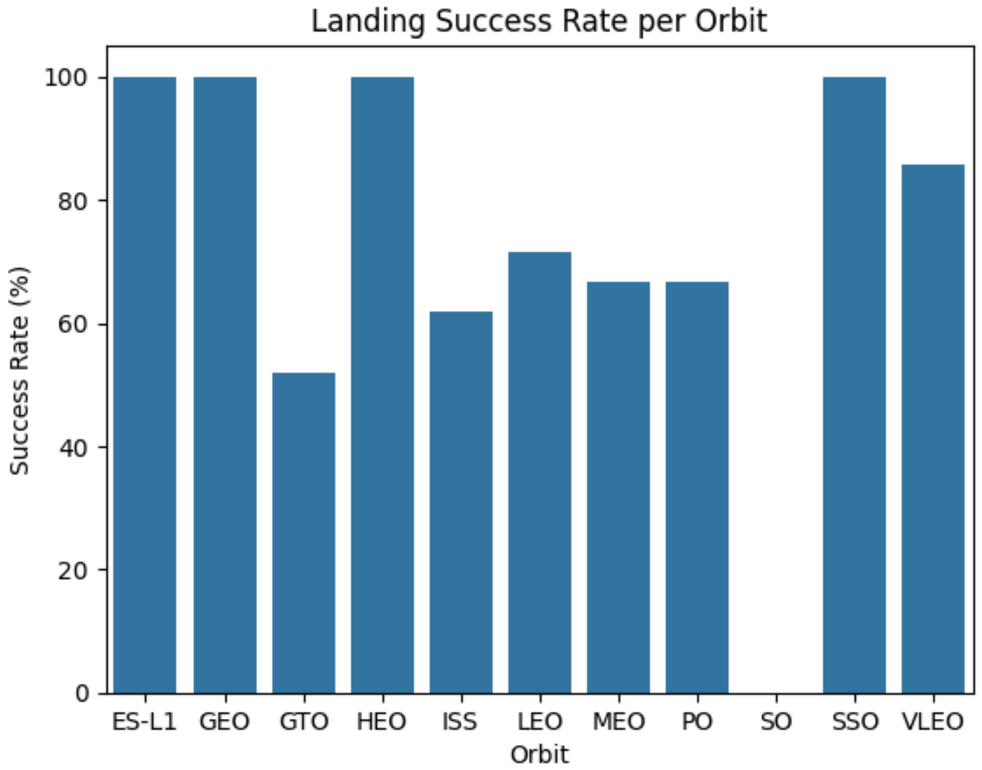
- The earliest flights all failed, while the most recent ones succeeded.
- The CCAFS SLC-40 launch site accounts for about half of all launches.
- Both VAFB SLC-4E and KSC LC-39A show higher success rates.
- This suggests that each new launch tends to have a higher likelihood of success.

Payload vs. Launch Site



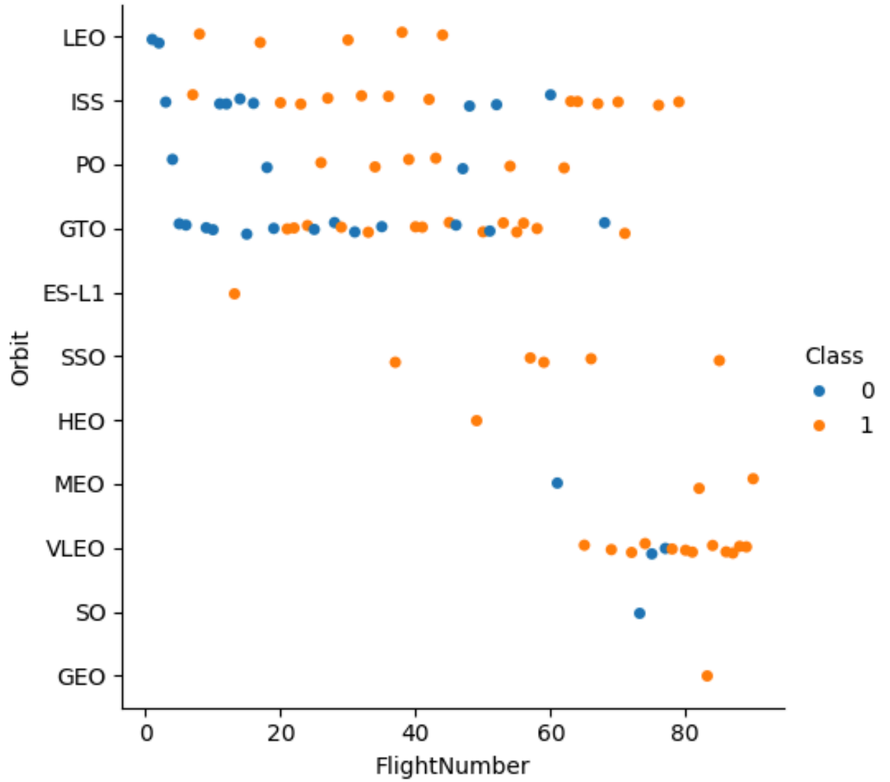
- For every launch site, a higher payload mass generally corresponds to a higher success rate.
- Most launches carrying payloads over 7,000 kg were successful.
- KSC LC-39A also maintains a 100% success rate for payloads under 5,500 kg.

Success rate vs. Orbit type



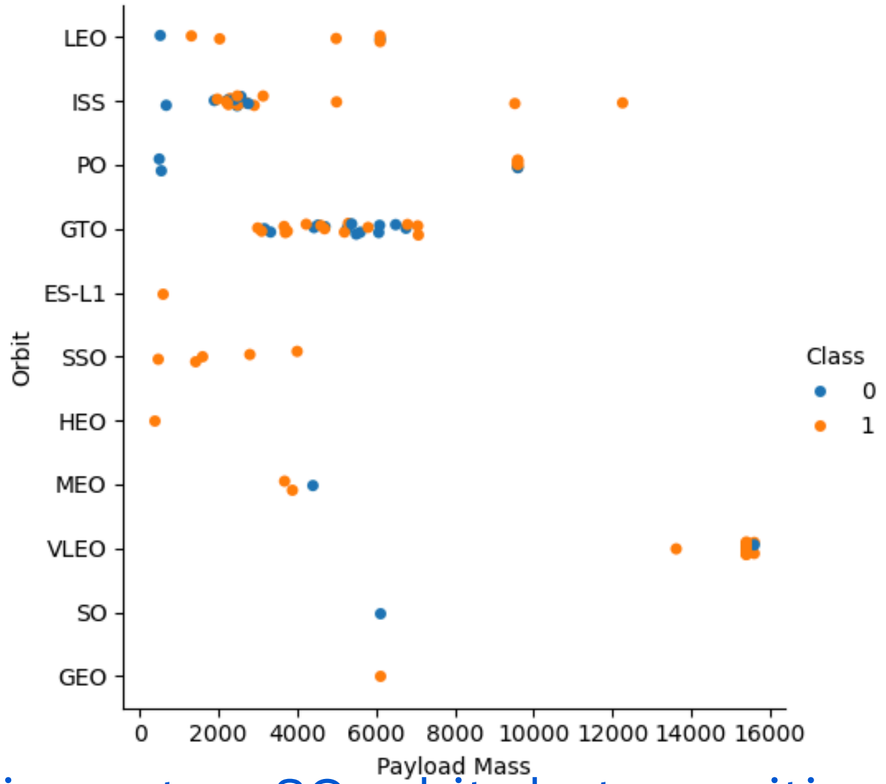
- Orbits with a 100% success rate:
  - ES-L1, GEO, HEO, SSO
- Orbit with a 0% success rate:
  - SO
- Orbits with success rates between 50% and 85%:
  - GTO, ISS, LEO, MEO, PO

Flight Number vs. Orbit type



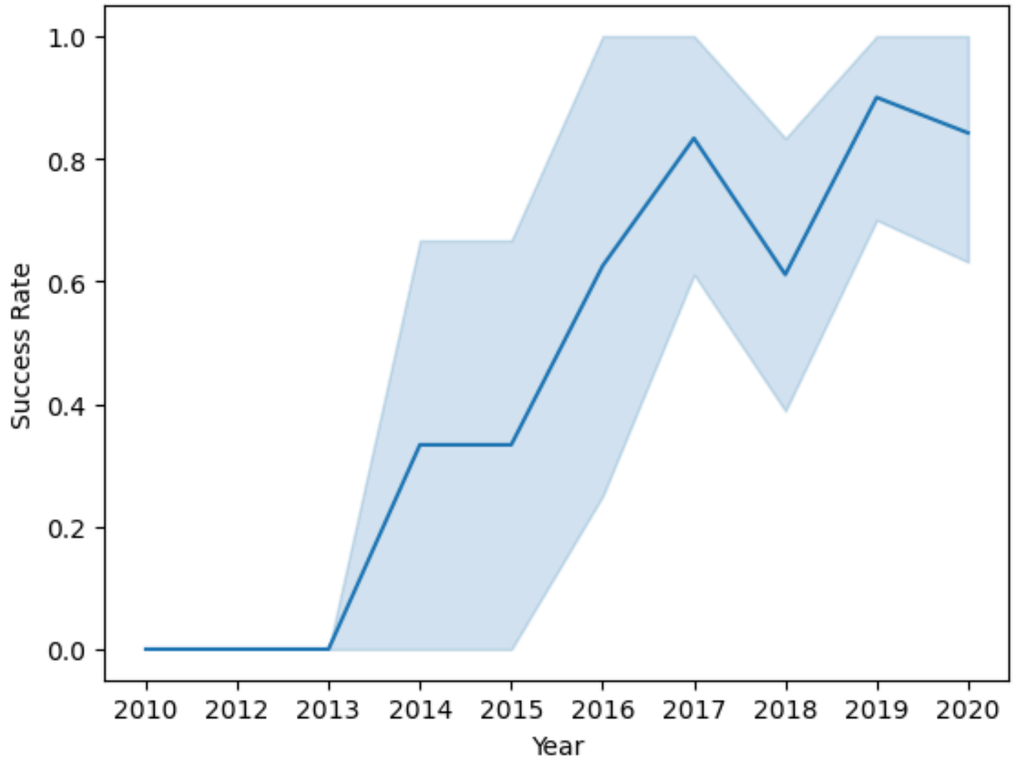
- In LEO, success appears to be related to the number of flights; in contrast, there seems to be no such relationship in GTO.

Payload Mass vs. Orbit type



Heavy payloads have a negative impact on SO orbits but a positive effect on GTO and Polar LEO (ISS) orbits.

Launch success yearly trend



Since 2013, the success rate has shown a consistent upward trend, continuing through 2020.

EDA with SQL

All launch site names

```
[10]: %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
```

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

[10]:

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

Displayed the names of the unique launch sites used in the space missions.

Launch site names begin with `CCA`

[11]:
```
%sql SELECT * FROM SPACEXTABLE where "Launch_Site" like 'CCA%' limit 5
```

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

[11]:

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

Displayed the five records where the launch site names begin with "CCA".

Total payload mass

[34]:
```
%sql select SUM("PAYLOAD_MASS__KG_") as total_payload from SPACEXTABLE where "Customer" = 'NASA (CRS)'
```

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

[34]:

total_payload
45596

Displayed the total payload mass carried by boosters launched by NASA (CRS).

Average payload mass by F9 v1.1

[13]:
```
%sql select AVG("PAYLOAD_MASS__KG_") as average_payload from SPACEXTABLE where "Booster_Version" like "F9 v1.1%"
```

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

[13]:

average_payload
2534.6666666666665

Explanation:

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

First successful ground landing date

```
[14]: %sql select min(Date) from SPACEXTABLE where "Landing_Outcome" like "Success%"
```

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

[14]:

min(Date)
2015-12-22

Explanation:

- Listing the date when the first successful landing outcome in ground pad was achieved.

Total number of successful and failure mission outcomes

[31]:
```
%sql select "Booster_Version" from SPACEXTABLE where ("Landing_Outcome" like "Success (drone ship)") and ("PAYLOAD_MASS__KG_" between 4000 and 6000)
```

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

[31]:

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

Explanation:

- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.

Total number of successful and failure mission outcomes

[33]:
```
%sql select mission_outcome, count(*) as total_number_of_outcomes from SPACEXTABLE group by mission_outcome;
```

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

[33]:

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

Explanation:

- Listing the total number of successful and failure mission outcomes.

Boosters carried maximum payload

```
[18]: %sql select "Booster_Version" from SPACEXTABLE where "PAYLOAD_MASS__KG_" = (select MAX("PAYLOAD_MASS__KG_") from SPACEXTABLE)
 * sqlite:///my_data1.db
Done.
```

[18]:

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:

- Listing the names of the booster versions which have carried the maximum payload mass.

2015 launch records

[30]:
```
%sql select substr(Date, 6, 2) as month, "Landing_Outcome", "Booster_Version", "Launch_Site" from SPACEXTABLE where substr(Date, 0, 5) = '2015' and "Landing_Outcome" like "%Failure (drone ship)%"
```

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

[30]:

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:

- Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015.

Rank success count between 2010-06-04 and 2017-03-20

```
[20]: %sql select "Landing_Outcome", COUNT(*) as outcome_count from SPACEXTABLE where "Date" between '2010-06-04' and '2017-03-20' group by "Landing_Outcome" order by outcome_count desc
```

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

[20]:

Landing_Outcome	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

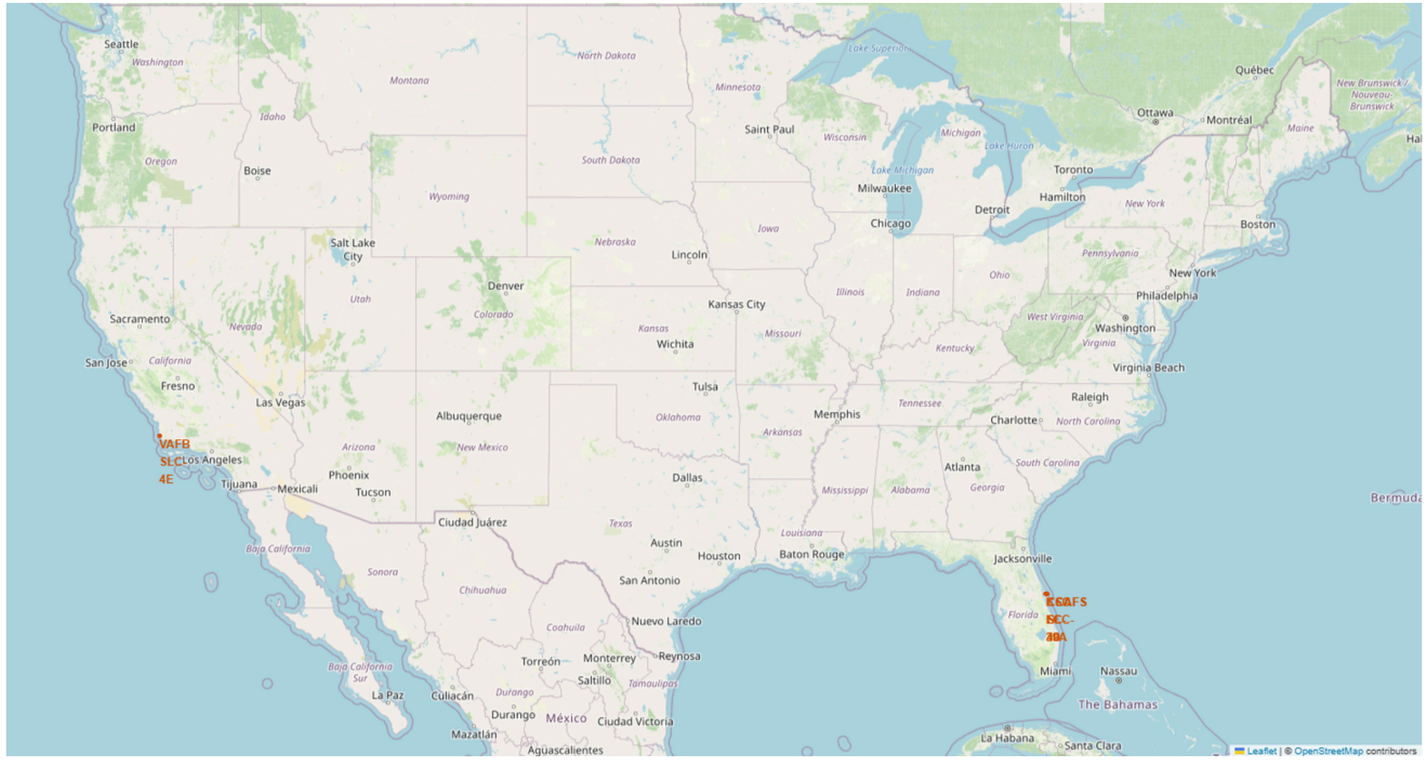
Explanation:

- Ranking 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.

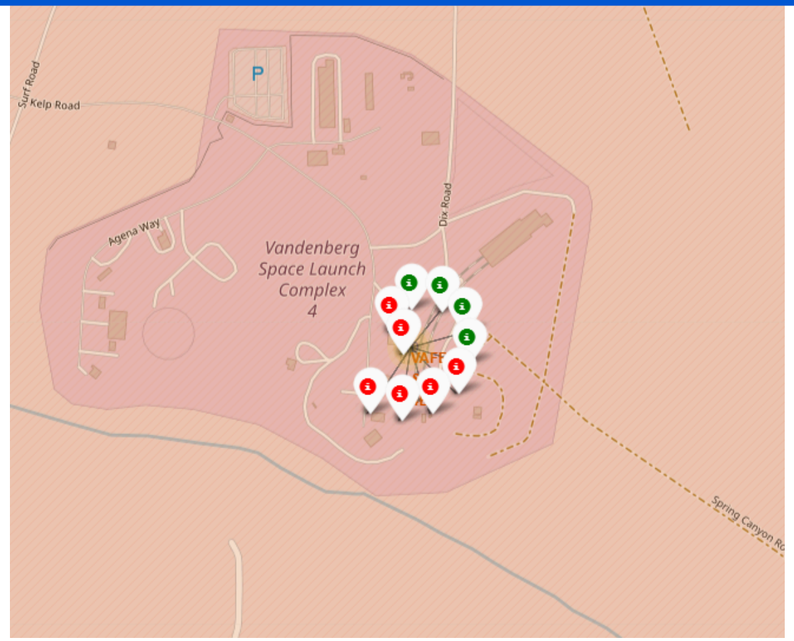
Interactive Map with Folium

All launch sites’ location markers on a global map

- Most launch sites are located near the Equator. The Earth’s surface moves fastest at the Equator, at approximately 1,670 km/h. Any object on the Equator is already moving at this speed due to the Earth’s rotation. When a rocket is launched from the Equator, it carries this rotational speed into orbit, thanks to inertia. This additional speed helps the spacecraft achieve and maintain orbit more efficiently.
- Additionally, all launch sites are situated close to the coast. Launching rockets over the ocean minimizes the risk of debris falling or explosions occurring near populated areas.

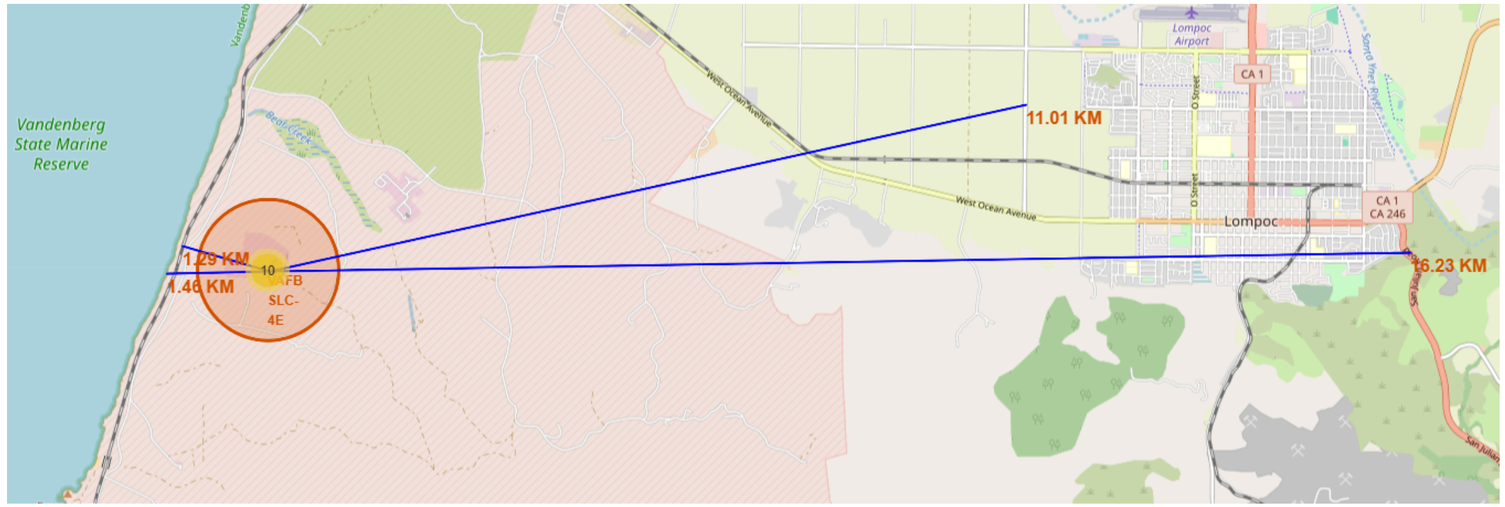


Colour-labeled launch records on the map



- The color-coded markers allow us to easily identify which launch sites have relatively high success rates.
  - Green marker: Successful launch
  - Red marker: Failed launch
- The launch site VAFB SLC-4E has more failed launches than successful ones as there are 6 failed launches out 10 total launches.

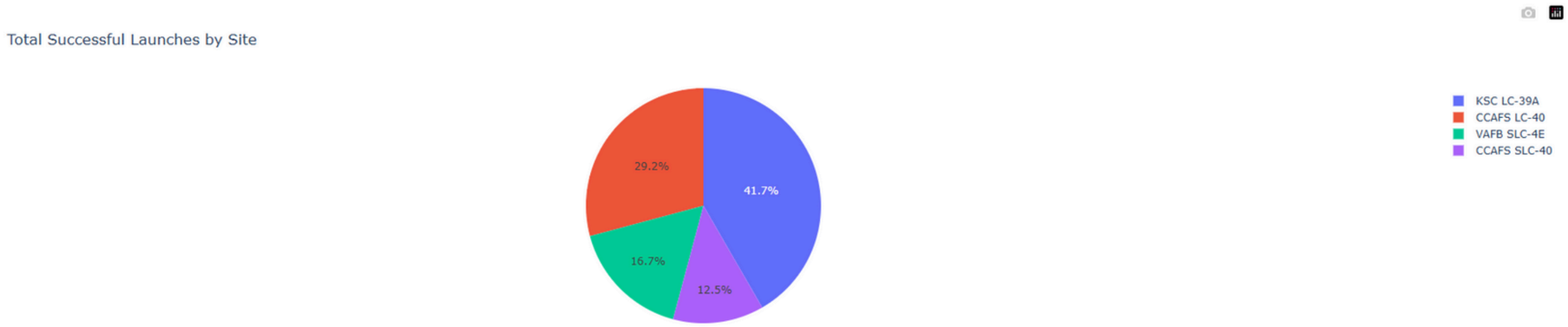
Distance from the launch site KSC LC-39A to its proximities



- From the visual analysis of the launch site VAFB SLC 4E, we observe that it is:
  - Relatively close to the railway (1.29 km)
  - Relatively close to the highway (6.23 km)
  - Relatively close to the coastline (1.46 km)
- Additionally, the launch site KSC LC-39A is relatively close to its nearest city, Lompoc (11.01 km).
- The rocket failure radius does not seem to encompass any major infrastructure; however, the nearby railway at VAFB SLC 4E could pose a risk in the event of a rocket failure.

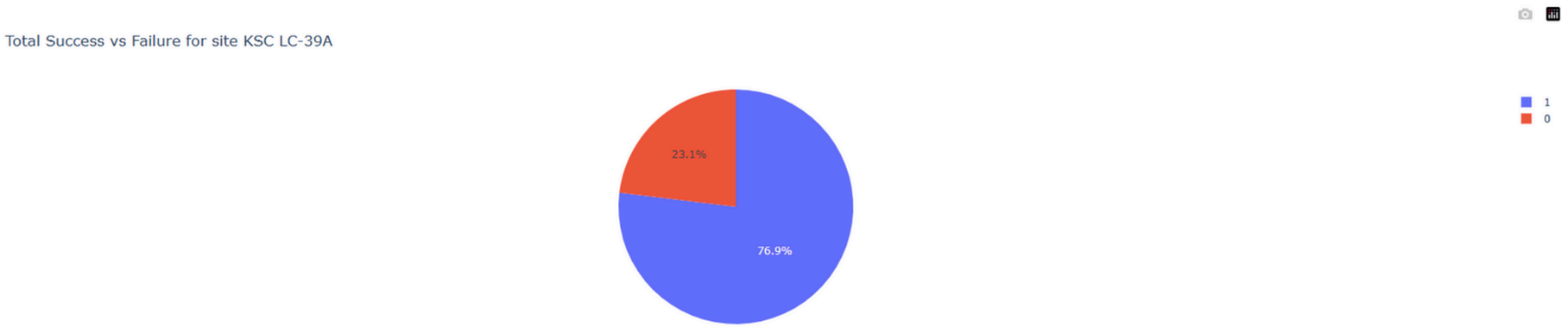
Plotly Dashboard

Launch success count for all sites



The chart clearly shows that, among all the sites, KSC LC-39A has the highest number of successful launches.

Launch site with highest launch success ratio



KSC LC-39A has the highest launch success rate at 76.9%, with 10 successful launches and only 3 failures.

Payload Mass vs. Launch Outcome for all sites

The charts indicate that payloads weighing between 2,000 kg and 5,500 kg achieve the highest success rates.

Predictive Analysis

Scores of the Test Set

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.800000	0.800000	0.692308	0.800000
F1_Score	0.888889	0.888889	0.818182	0.888889
Accuracy	0.846429	0.848214	0.875000	0.848214

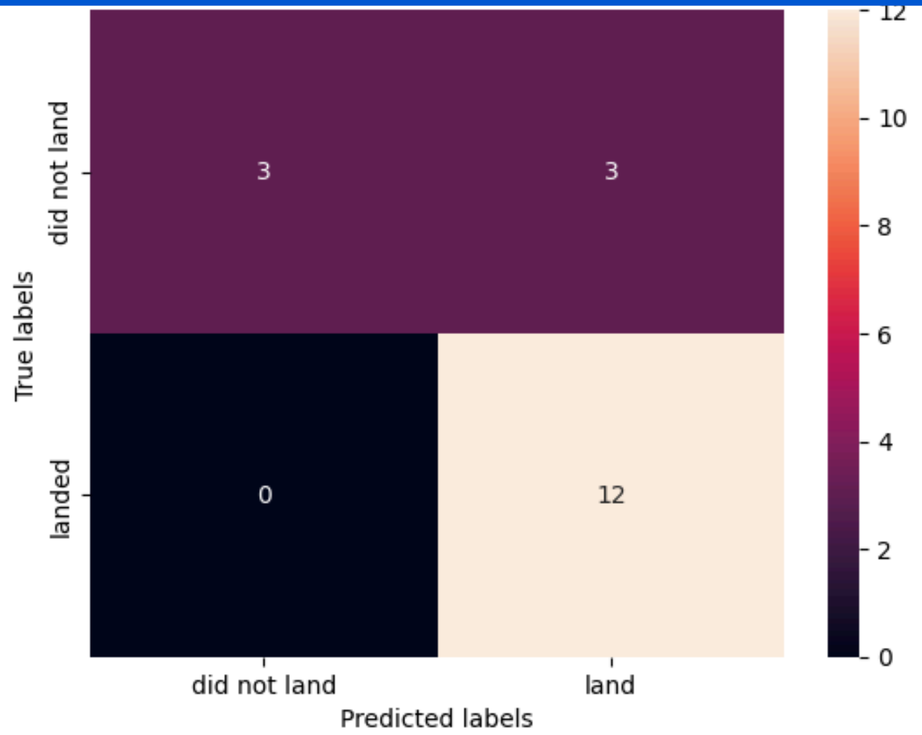
Based on the Test Set scores, we cannot determine which method performs best. The similar scores may be due to the small test sample size (18 samples). Therefore, we evaluated all methods on the entire dataset. The results confirm that the Decision Tree model is the best, achieving not only higher scores but also the highest accuracy.

Scores of the Whole Date Set

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.833333	0.845070	0.761194	0.819444
F1_Score	0.909091	0.916031	0.864407	0.900763
Accuracy	0.866667	0.877778	0.822222	0.855556

The results on the full dataset confirm that the Decision Tree model is the best, achieving not only higher scores but also the highest accuracy.

Confusion Matrix



We can see that logistic regression correctly classifies most samples, but the main issue is false positives, which occur 3 times. Overall, the model distinguishes the classes well, with only a few misclassifications.

Conclusion

- The Decision Tree model is the best algorithm for this dataset. Launches with lower payload masses perform better than those with larger payloads.
- Most launch sites are located near the Equator, and all sites are in close proximity to the coast.
- The success rate of launches has increased over the years, with KSC LC-39A achieving the highest success rate among all sites.
- Additionally, launches to ES-L1, GEO, HEO, and SSO orbits have achieved a 100% success rate.