IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Karina S
19.12.2024

Outline

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

Executive Summary

- Basic Python methods, machine learning algorithms, and SQL were used throughout the course. Data exploration, preprocessing, and feature selection were performed with Python, while machine learning models were developed for prediction. SQL was employed to manage and query the dataset efficiently.
- The final model predicted landing success with high accuracy. Key factors influencing landing outcomes were identified, and the model provided reliable cost estimations for launches based on the likelihood of first-stage reuse.

Introduction

- Commercial space travel is growing, with SpaceX leading the way by reusing Falcon 9's first stage to reduce costs. At Space Y, our goal is to predict if the first stage will land successfully, helping determine the launch cost.
- The aim of this project is to predict whether the Falcon 9's first stage will land and be reused, as this affects the mission's cost. Key factors will be identified and a model to forecast landing success will be built.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology with API
- Data wrangling to find patterns and label data
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using several classification models

Data Collection

Two primary methods for data collection were employed:

- data was requested from the SpaceX API:

```
GET Request
     ↓
Data in .JSON Format
     ↓
Pandas DataFrame
     ↓
Data in .CSV
```

- web scraping was performed to gather Falcon 9 launch records available online

```
HTML Page
     ↓
BeautifulSoup (Python)
     ↓
Pandas DataFrame
     ↓
Save Data as .CSV
```

Data Collection - SpaceX API

The flowchart shows detailed data collection process with coding:

Complete data and analysis:
[Github: data collection - Space X API]()

```
Step 1: Getting response from API
    spacex_url = "https://api.spacexdata.com/v4/launches/past"
    response = requests.get(spacex_url)
    ↓
Step 2: Converting response to .json file
    data = pd.json_normalize(response.json())
    ↓
Step 3: Use functions to apply outputs to the variables
    getBoosterVersion(data)
    getLaunchSite(data)
    getPayloadData(data)
    getCoreData(data)
    ↓
Step 4: Assign the list into a dictionary, then a dataframe
    launch_dict = {
        'FlightNumber': list(data['flight_number']),
        'Date': list(data['date']),
        'BoosterVersion': BoosterVersion,
        'PayloadMass': PayloadMass,
        'Orbit': Orbit,
        …
    }
    df = pd.DataFrame.from_dict(launch_dict)
    ↓
Step 5: Filter dataframe and export to csv
    data_falcon9 = df[df['BoosterVersion'] != 'Falcon 1']
    data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping

The webscraping process in more details:

```
Step 1: Getting HTML response and creating BeautifulSoup object
    data = requests.get(static_url).text
    soup = BeautifulSoup(data)
    ↓
Step 2: Finding tables
    html_tables = soup.find_all('table')
    ↓
Step 3: Getting column names
    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)
    ↓
```

Complete data and analysis:
[Github: data collection - WebScraping]

```
    ↓
Step 4: Creation of dictionary
    launch_dict = dict.fromkeys(column_names)
    del launch_dict['Date and time ( )']
    launch_dict['Flight No.'] = []
    launch_dict['Launch site'] = []
    ↓
Step 5: Fill the dictionary with launch records
    extracted_row = 0
    for table_number, table in enumerate(soup.find_all('table',
        for rows in table.find_all("tr"):
            …
    ↓
Step 6: Convert dictionary to dataframe and export to CSV
    df = pd.DataFrame(launch_dict)
    df.to_csv('spacex_web_scraped.csv', index=False)
```

Data Wrangling Workflow

```
Step 1: Calculate the number of launches on each site
     df['LaunchSite'].value_counts()
     ↓
Step 2: Calculate the number and occurrence of each orbit
     df['Orbit'].value_counts()
     ↓
Step 3: Calculate the number and occurrence of mission outcome per orbit type
     landing_outcomes = df['Outcome'].value_counts()
     landing_outcomes
     ↓
Step 4: Create a landing outcome label from Outcome column
     df['Class'] = landing_class
     df[['Class']].head(8)
     ↓
Step 5: Export to CSV
     df.to_csv("dataset_part_2.csv", index=False)
```

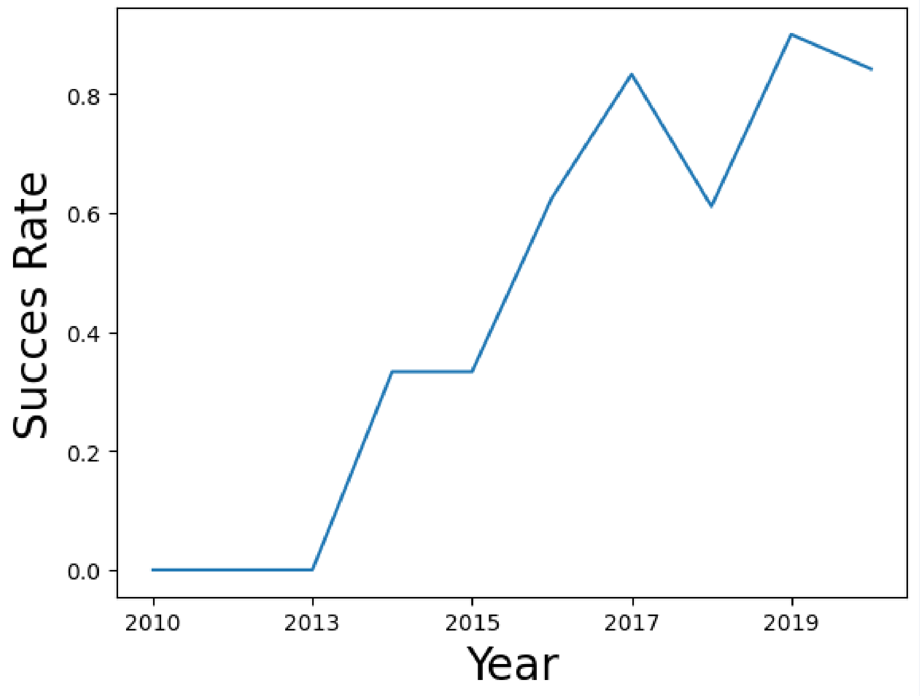
Complete data and analysis:
[Github: data wrangling]

EDA with Data Visualization

Visual charts can help us see the relationship between:

- flight number and payload mass;
- flight number and launch site;
- launch site and payload mass;
- orbit type and success rate;
- flight number and orbit type;
- payload mass and orbit type;
- and year and success rate

This is also a great way to see if there are any trends.



Launch success yearly trend - visual

Complete data and analysis:
[Github: EDA]

EDA with SQL

SQL helped us to get the following data:

Complete data and analysis:
[Github: EDA with SQL]

- Display the names of the unique launch sites
- Display 5 records where launch sites begin with ‘KSC’
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date where the successful landing outcome in drone ship was achieved.
- List the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
- List the total amount of successful and failure mission outcomes
- List the booster versions which have carried the maximum payload mass
- List the monthly records for 2017
- Count the successful landings between june 2010 and march 2017

Build an Interactive Map with Folium

We added the following map objects:

- Markers: Added to pinpoint SpaceX launch sites for easy identification.
- Circle Markers: Highlighted areas around launch sites to show proximity and influence.
- Polylines: Represented trajectories or connections between locations like launch and landing sites.
- Popups: Provided additional details on markers for better context.
- Tile Layers: Enhanced map style (e.g., satellite view) for clarity.
- Layer Controls: Allowed toggling layers to explore data interactively.

These objects were added to the Folium map to make it interactive and show SpaceX launch activities clearly. They help analyze the areas around launch sites and find the best locations for future launches.

Complete data and analysis:
Github: Visual analytics with Folium

Build a Dashboard with Plotly Dash

We created a pie chart displaying the success rate by launch site, with an interactive dropdown to select and view the rate for any specific site.

Additionally, we plotted a scatter graph showing mission outcomes (success/fail) based on payload mass and booster version.

These charts help identify the best launch site and payload mass combination, aiding in predicting mission outcomes.

Complete data and analysis:
Github: Plotly Dashboard

Predictive Analysis (Classification)

1. **Data Preprocessing:** Cleaned data, handled missing values, and engineered features like 'Payload Mass (kg)'.
2. **Model Building:** Tested multiple classifiers (Logistic Regression, SVM, Decision Tree, KNN).
3. **Model Evaluation:** Used accuracy, precision, recall, F1-score, and k-fold cross-validation to assess performance.
4. **Model Improvement:** Applied GridSearchCV for hyperparameter tuning and feature selection.
5. **Best Model:** all used models performed approximately the same way with accuracy of 0.83333

Model Development Flow:

```
Data Collection
       ↓
Data Cleaning
       ↓
Feature Engineering
       ↓
Model Selection → Model Training
       ↓
Model Evaluation → Hyperparameter Tuning
       ↓
Feature Selection
       ↓
Best Performing Model Selection
       ↓
Final Model Training & Evaluation
```

Complete data and analysis:
[Github: Machine Learning Prediction]

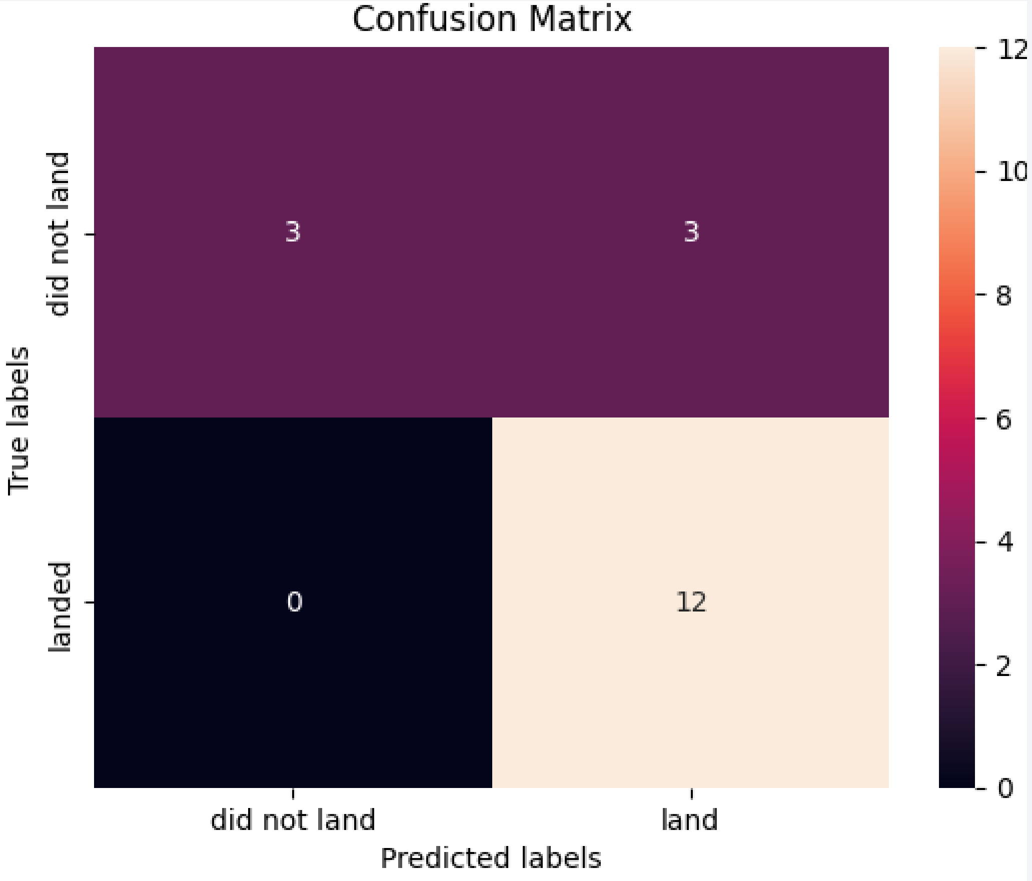
Results

Exploratory Data Analysis Results:

- Analyzed launch data to identify trends and patterns.
- Found key insights like launch site frequencies, payload mass distributions, and orbit types.
- Summary statistics provided a good foundation for predicting mission success.

Predictive Analysis Results:

- Built classification models to predict mission success.
- Evaluated models based on accuracy and performance metrics.
- All models performed approximately the same

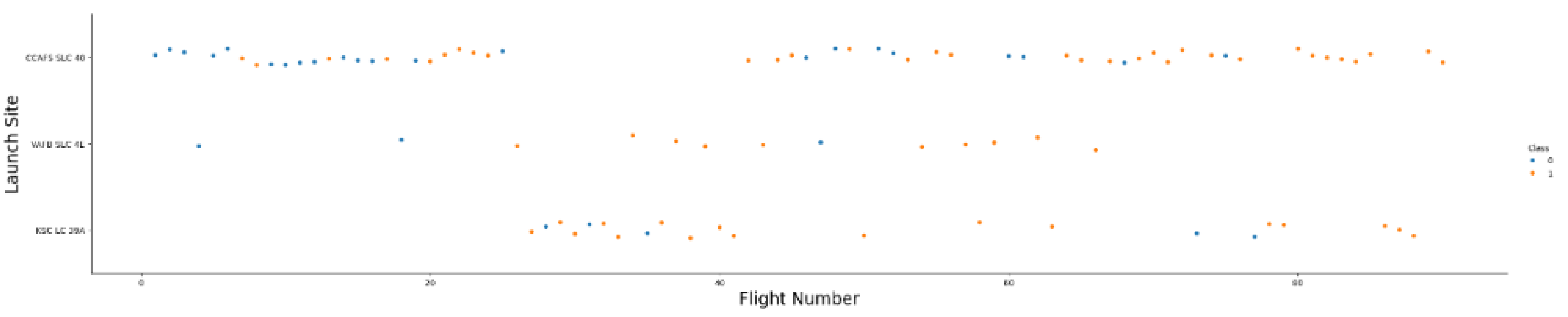


Section 2

Insights drawn from EDA

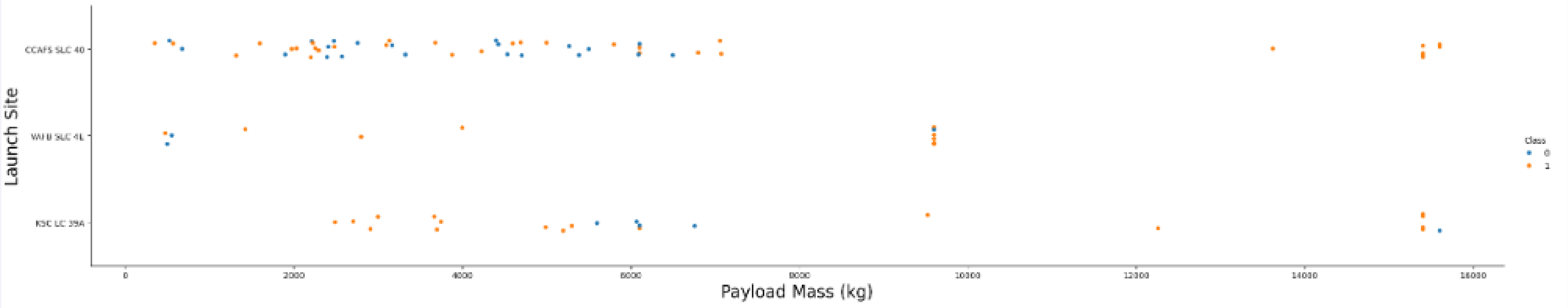
Flight Number vs. Launch Site

There are far more launches from the CCAFS SLC 40 launch site than any of the other sites.



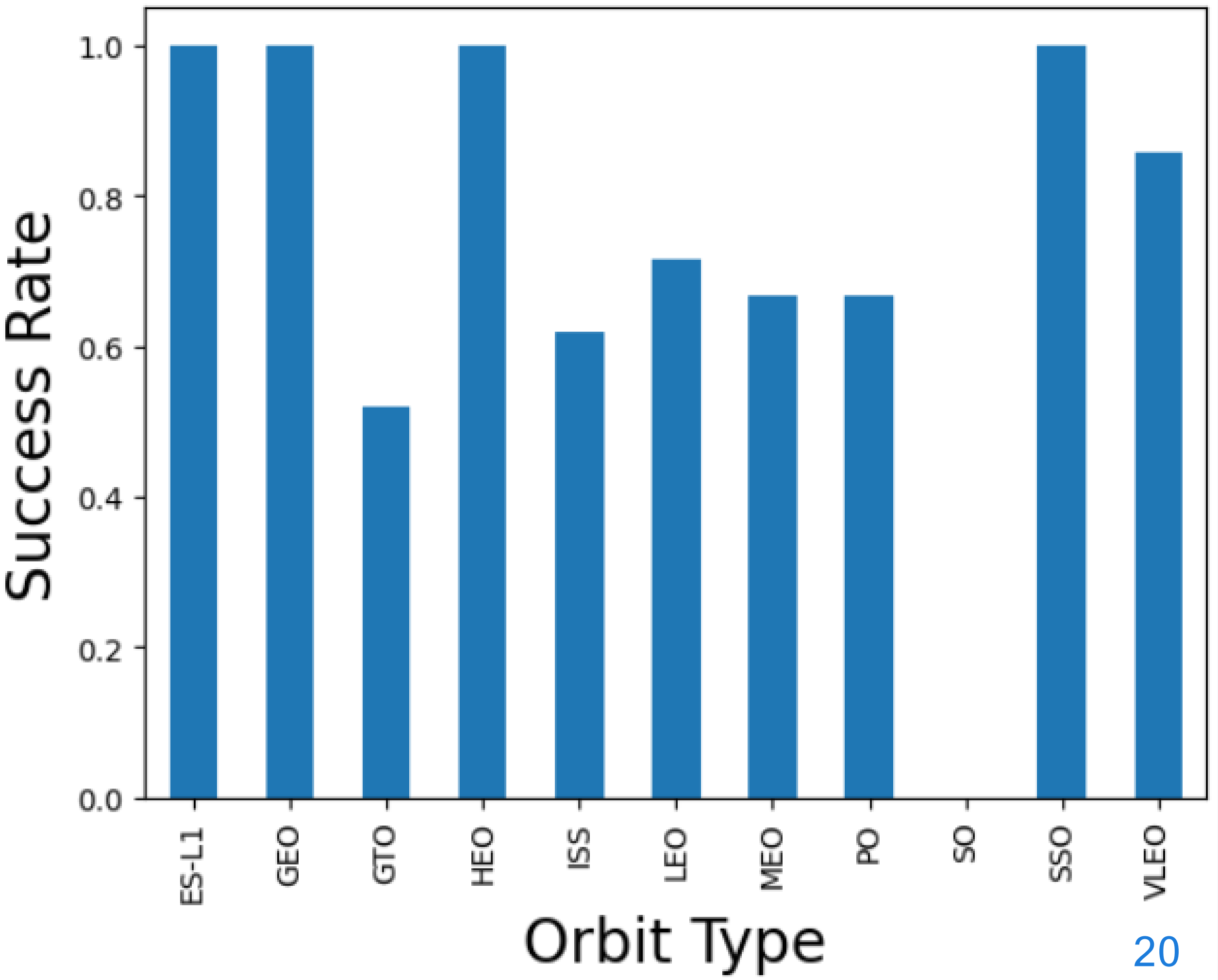
Payload vs. Launch Site

The site with the highest number of the lighter payload launches is CCAFS SLC 40.
There are no mass(greater than 10000) rockets launched from VAFB-SLC.



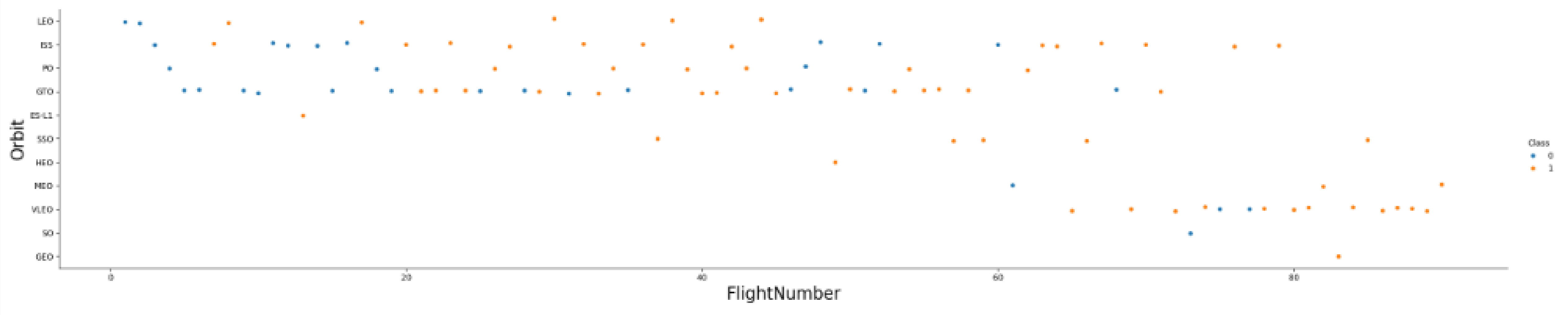
Success Rate vs. Orbit Type

The orbits with the highest success rate are ES-L1, GEO, HEO and SSO, followed by VLEO with a slightly lower success rate.



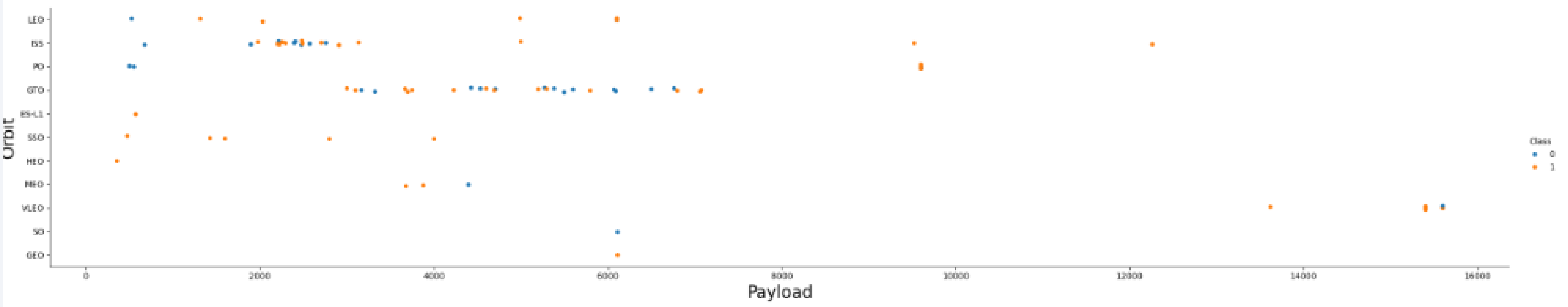
Flight Number vs. Orbit Type

There seems to be a shift in the preferred orbit type in the latest launches, the trend seems to move to newer orbits such as VLEO or SSO.



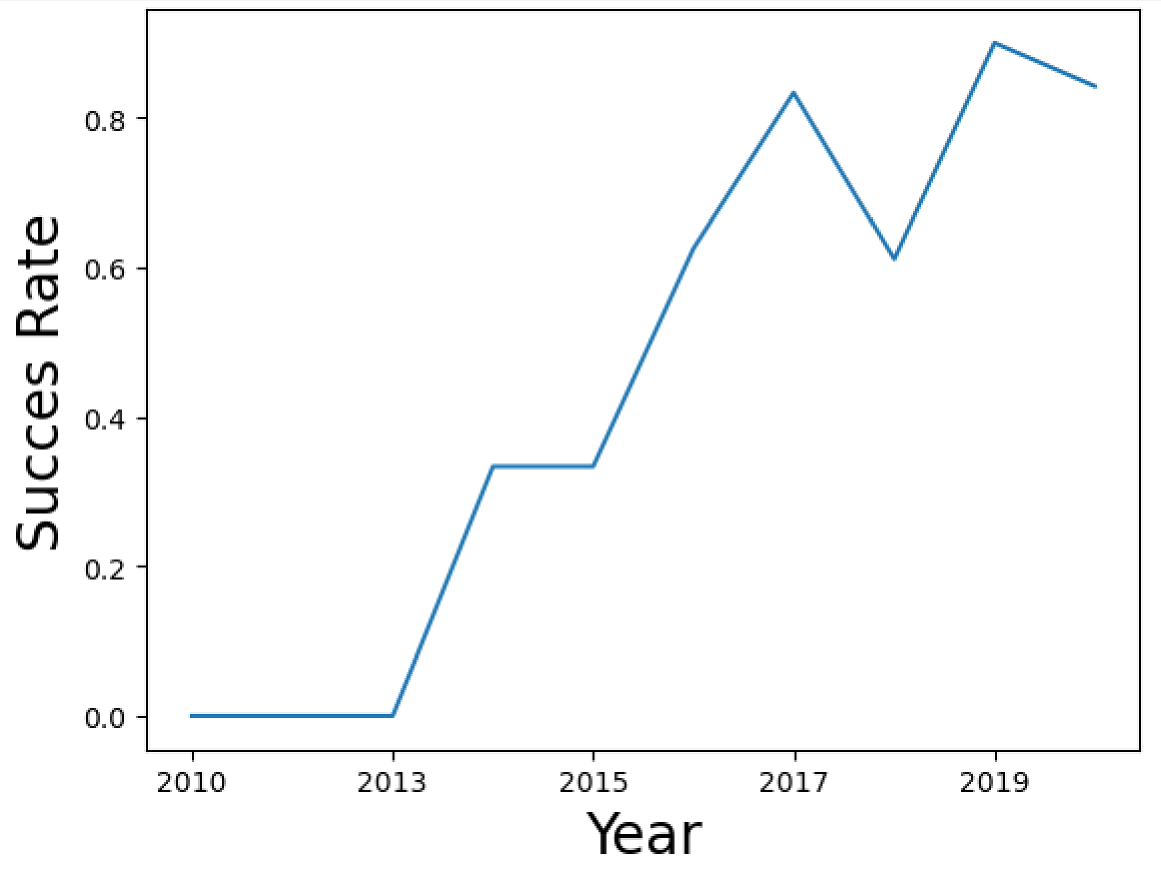
Payload vs. Orbit Type

With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.



Launch Success Yearly Trend

Based on the graph, the success rate kept increasing since the year 2013, having only a slight dip in the year 2018. However the trend is positive.



All Launch Site Names

To start our Exploratory Data Analysis (EDA), we first explored the data by examining the launch sites. We did this by running a DISTINCT query to identify the different launch sites. The results are shown below:

Launch Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Launch Site Names Begin with 'KSC'

The next step in our EDA with SQL was to get an idea of the amount of data we were working with. In order to have a first impression of this, we selected one of the launch sites seen previously and got five results for it:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

Total Payload Mass

Next we wanted to get an overall idea of the payload mass carried by the boosters launched by NASA. In order to get the result, we used an SQL query in which we filtered the results by CRS tag:

total_payload_mass_carried_by_NASA_kg
45596

Average Payload Mass by F9 v1.1

To explore payload mass, we focused on the average mass carried by the booster version F9 v1.1. We filtered the data by booster version and calculated the average payload mass using an SQL query:

average_payload_mass_carried_by_booster_version_F9_v1_1_kg
2928.4

First Successful Drone Ship Landing Date

For the next step in our EDA we wanted to know the date of the first successful drone ship landing. We filtered the data by and SQL query by the landing outcome:

first_successful_landing_date
2016-04-08

Successful Drone Ship Landing with Payload between 4000 and 6000

Next, we wanted to identify the boosters that successfully landed on a drone ship while carrying a payload between 4000 and 6000 kg. To achieve this, we applied two filters in our query: one for payload weight and another for the landing outcome:

Booster_Version
F9 FT B1019
F9 FT B1025.1
F9 FT B1031.1
F9 FT B1032.1
F9 FT B1035.1
F9 B4 B1039.1
F9 B4 B1040.1
F9 FT B1035.2
F9 B4 B1043.1

Total Number of Successful and Failure Mission Outcomes

Next, we aimed to determine the total number of successful and failed mission outcomes. To do this, we filtered the query by outcome and created separate results for successes and failures. The results are as follows:

MISSION_OUTCOME	TOTAL_NUMBER
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

We wanted to identify the boosters that carried the maximum payload mass. To do this, we first ran a DISTINCT query and then filtered the results to focus on boosters that carried the highest payloads. The result is as follows:

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

2017 Launch Records

For the next step in our EDA, we aimed to display the month names, successful landing outcomes on the ground pad, booster versions, and launch sites for each month in 2017. This query involved filtering by outcome and selecting four columns. The result is as follows:

Month	Booster_Version	Launch_Site	Landing_Outcome
02	F9 FT B1031.1	KSC LC-39A	Success (ground pad)
05	F9 FT B1032.1	KSC LC-39A	Success (ground pad)
06	F9 FT B1035.1	KSC LC-39A	Success (ground pad)
08	F9 B4 B1039.1	KSC LC-39A	Success (ground pad)
09	F9 B4 B1040.1	KSC LC-39A	Success (ground pad)
12	F9 FT B1035.2	CCAFS SLC-40	Success (ground pad)

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

For the final step of our EDA with SQL, we aimed to rank the landing outcomes between June 2010 and March 2017. To achieve this, we applied two filters: one for the date and one for the outcome. We then used a GROUP BY clause to order the results in descending order. The result is as follows:

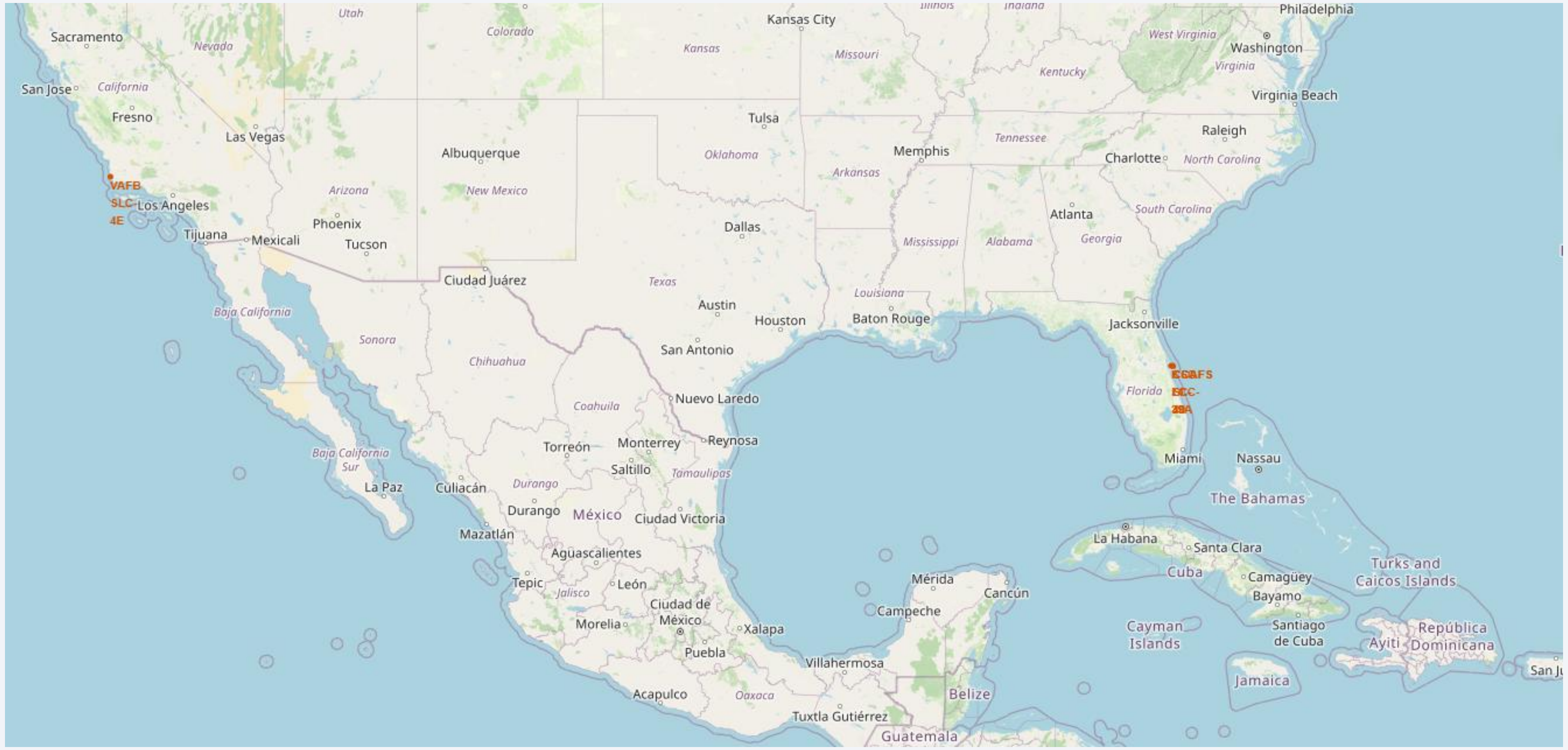
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

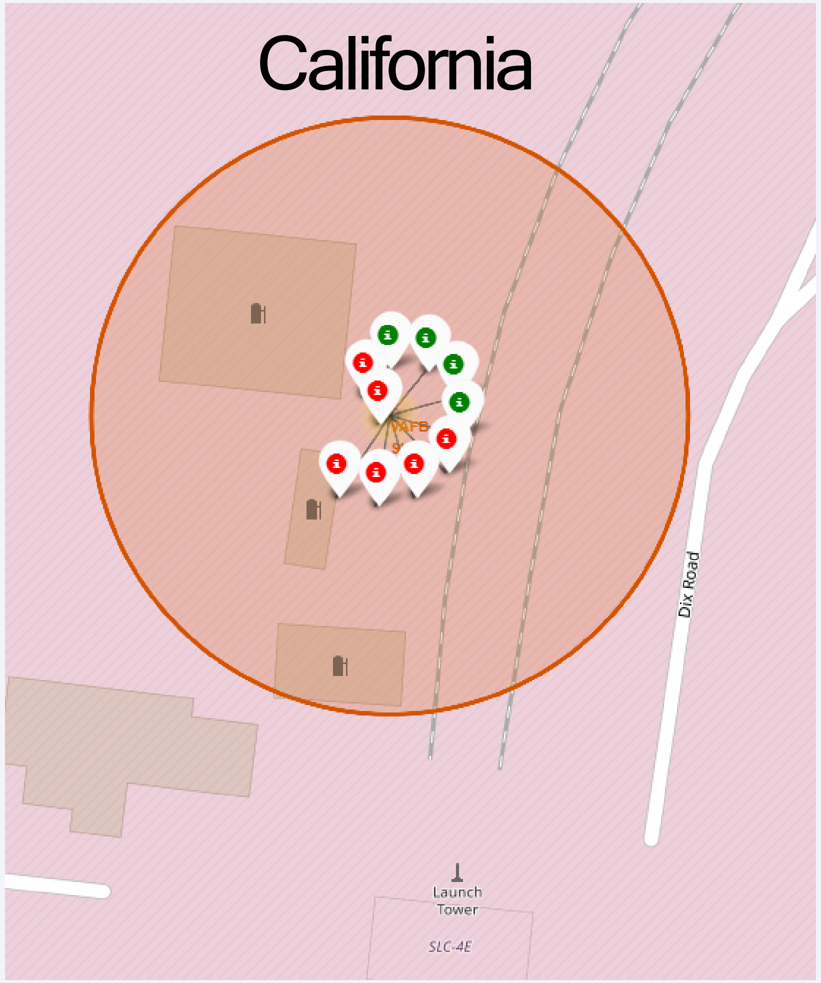
Launch Sites Proximities Analysis

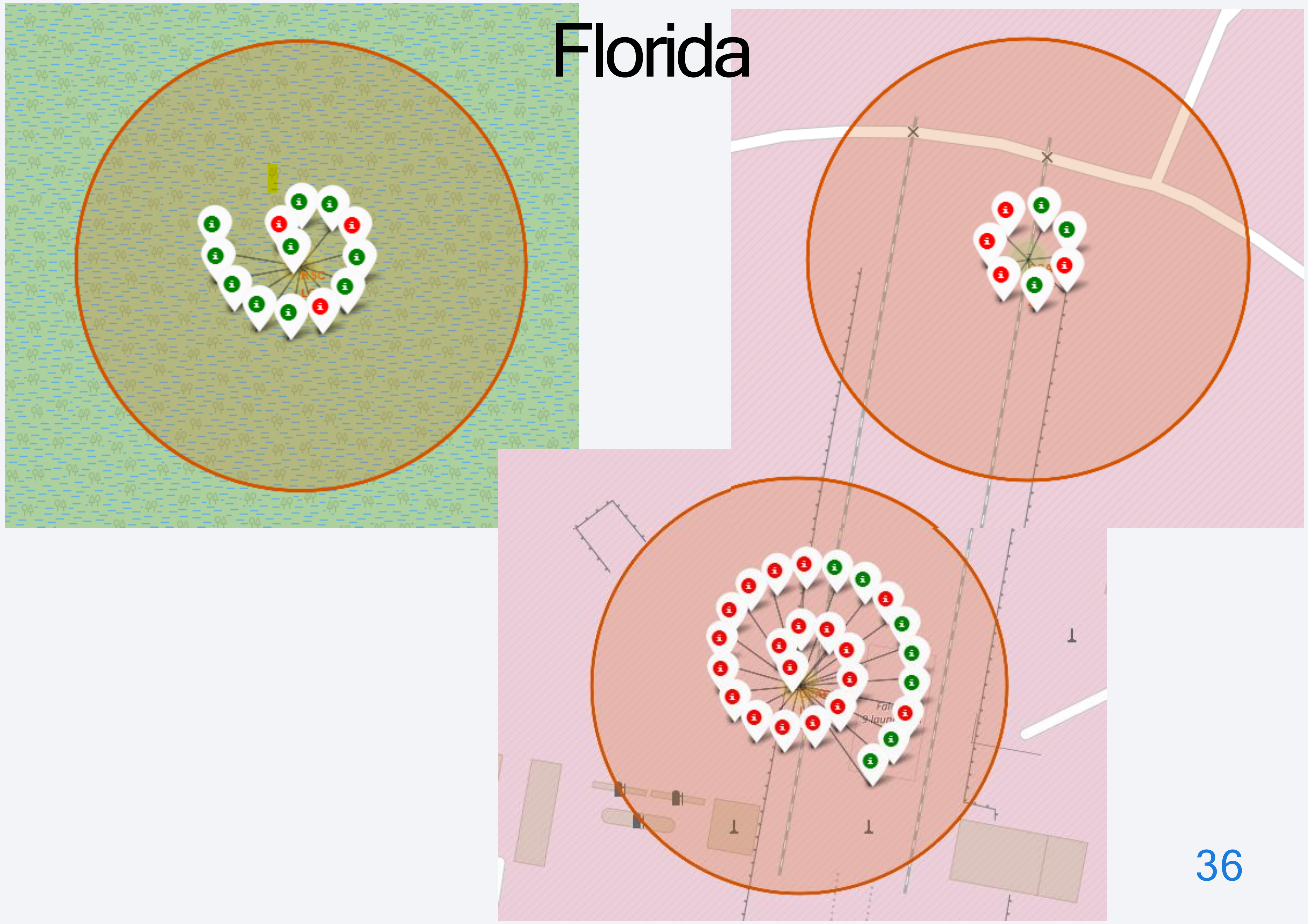
All launch sites

As shown on the map, all the launch site locations are in the US, specifically along both the east and west coastlines. Additionally, an interesting observation is that the sites are positioned near the equator, which is advantageous for space missions, as launches from near the equator can take advantage of the Earth's rotational speed to boost the rocket's velocity. This positioning helps optimize fuel efficiency and launch success rates.

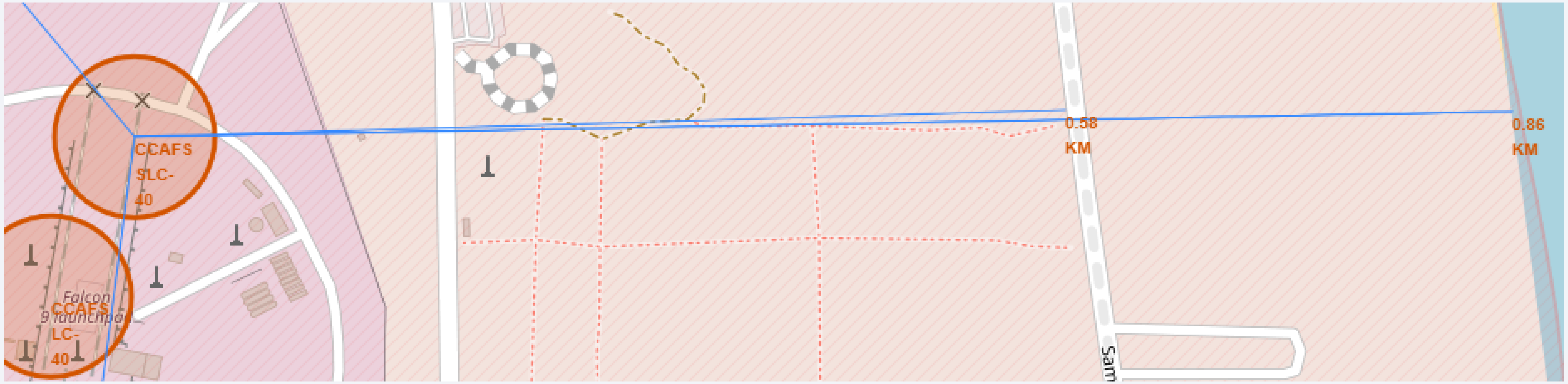


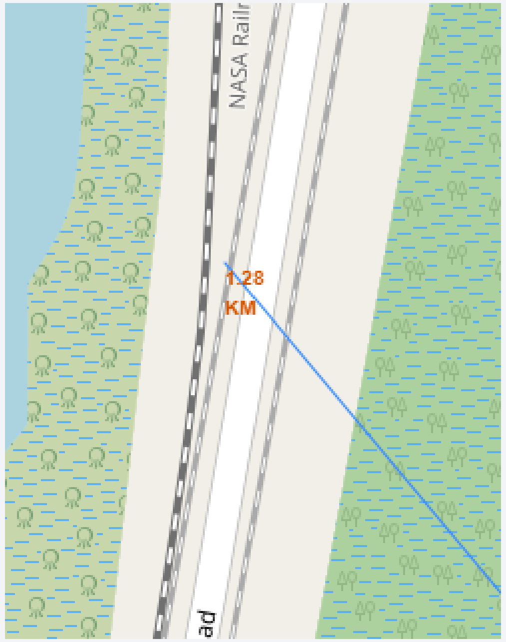
Successful and failed outcomes marked on launch sites

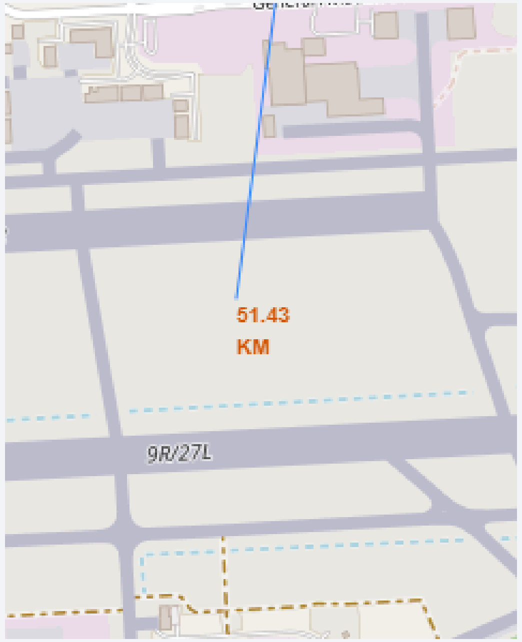




Distance from launch sites to landmarks







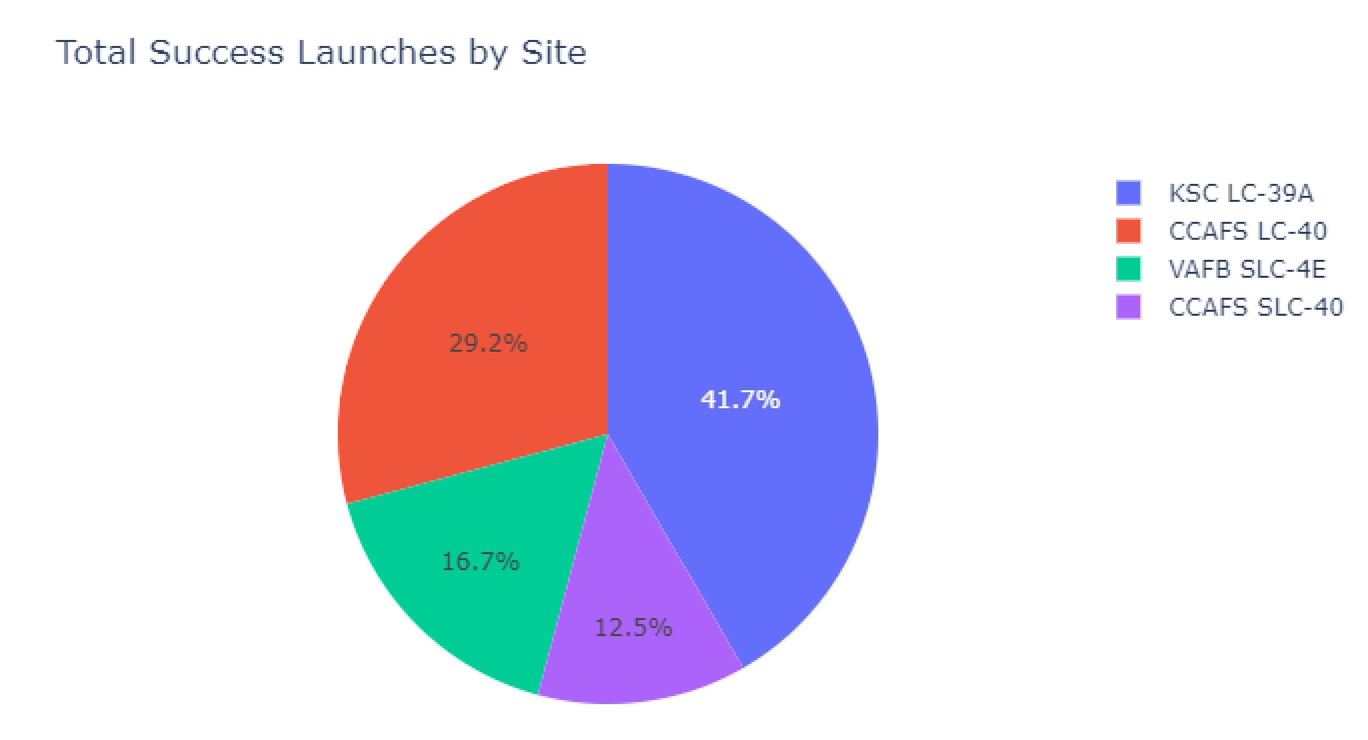
The launch site in Florida is located nearby railways and highway for reducing and simplifying transport of material and people, also near the coastline to redirect landings. The launch site is located far away from the city to lessen the danger of unsuccessful launch/landing.

Section 4

Build a Dashboard with Plotly Dash

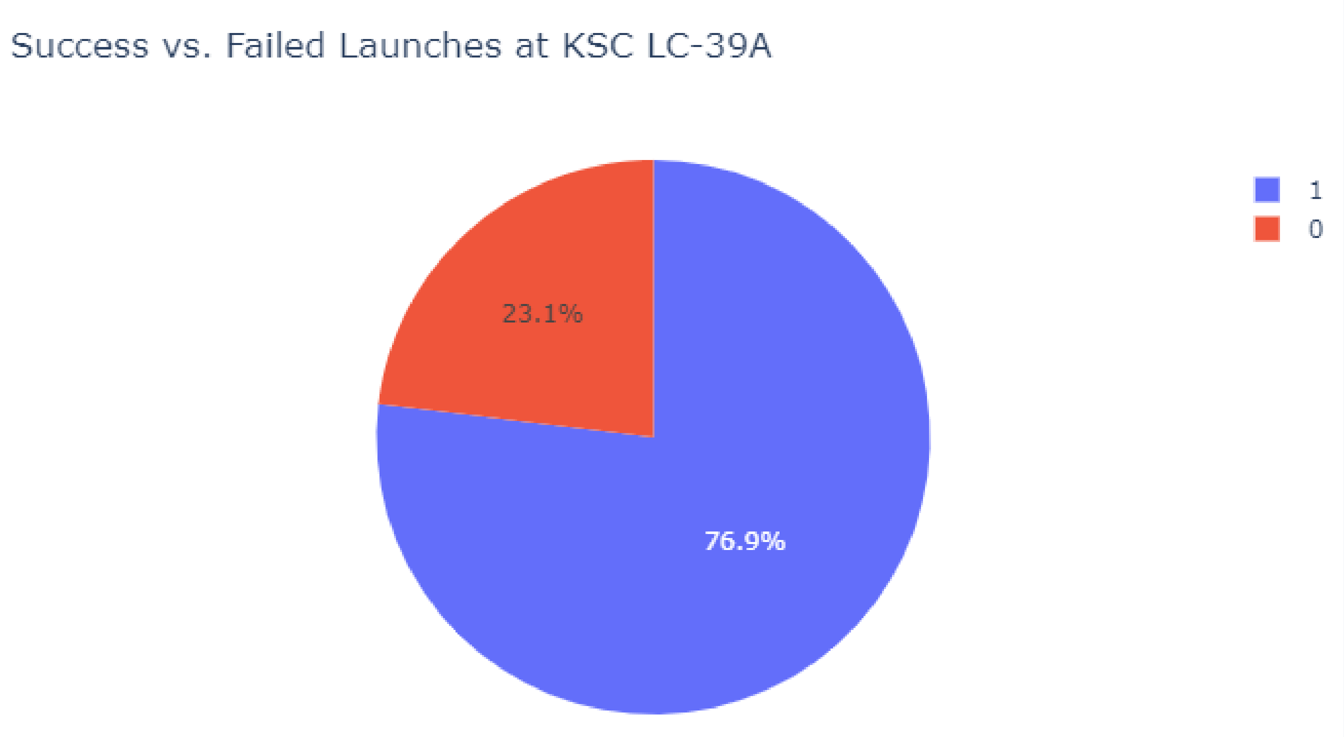
Success percentage by launch site

The majority of successful launches are performed from KSC LC-39A. The least successful launch site is CCAFS SLC-40



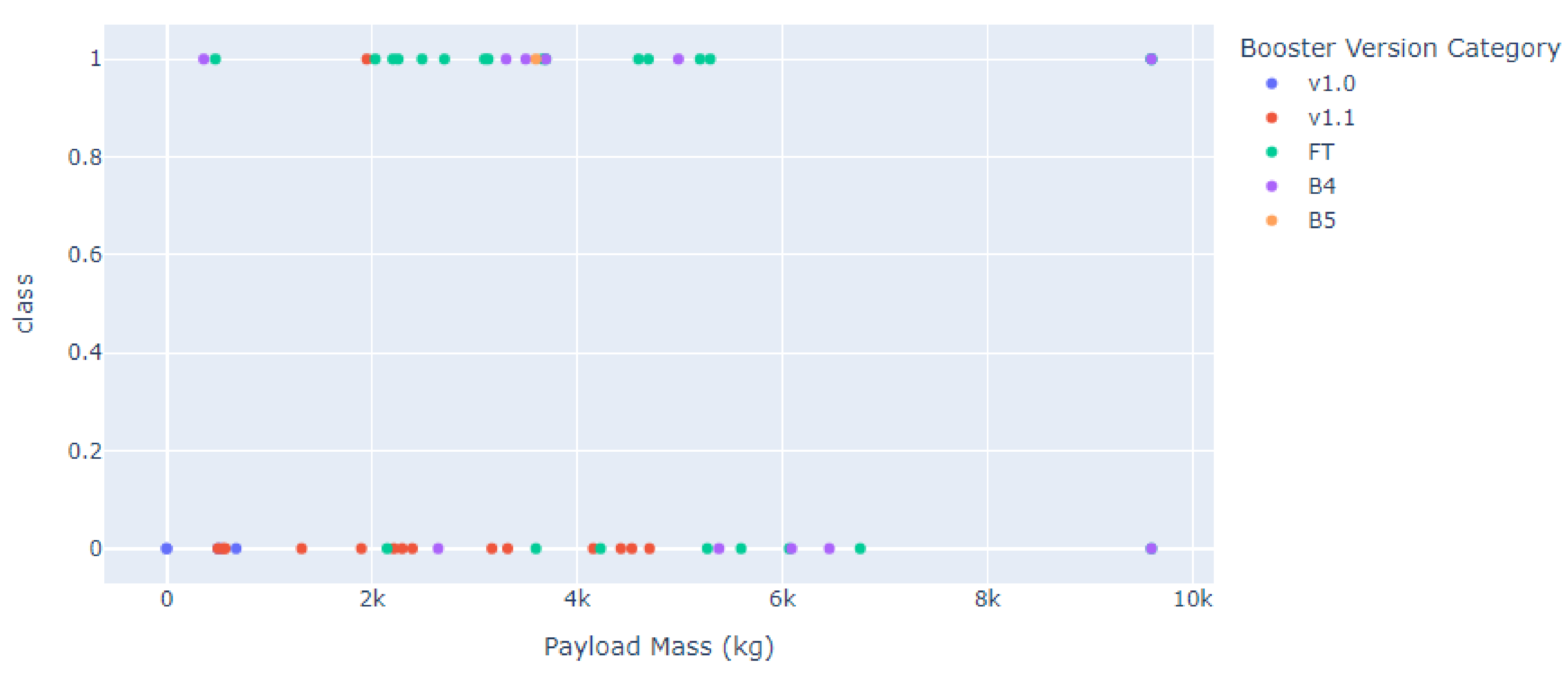
Highest success rate by a single launch site

Another look at KSC LC-39A. It has both highest success and lowest failure rate



Payload vs Launch Outcome for all sites

Low payload < 4000kg launces are more successful, than high payload ones



Section 5

Predictive Analysis (Classification)

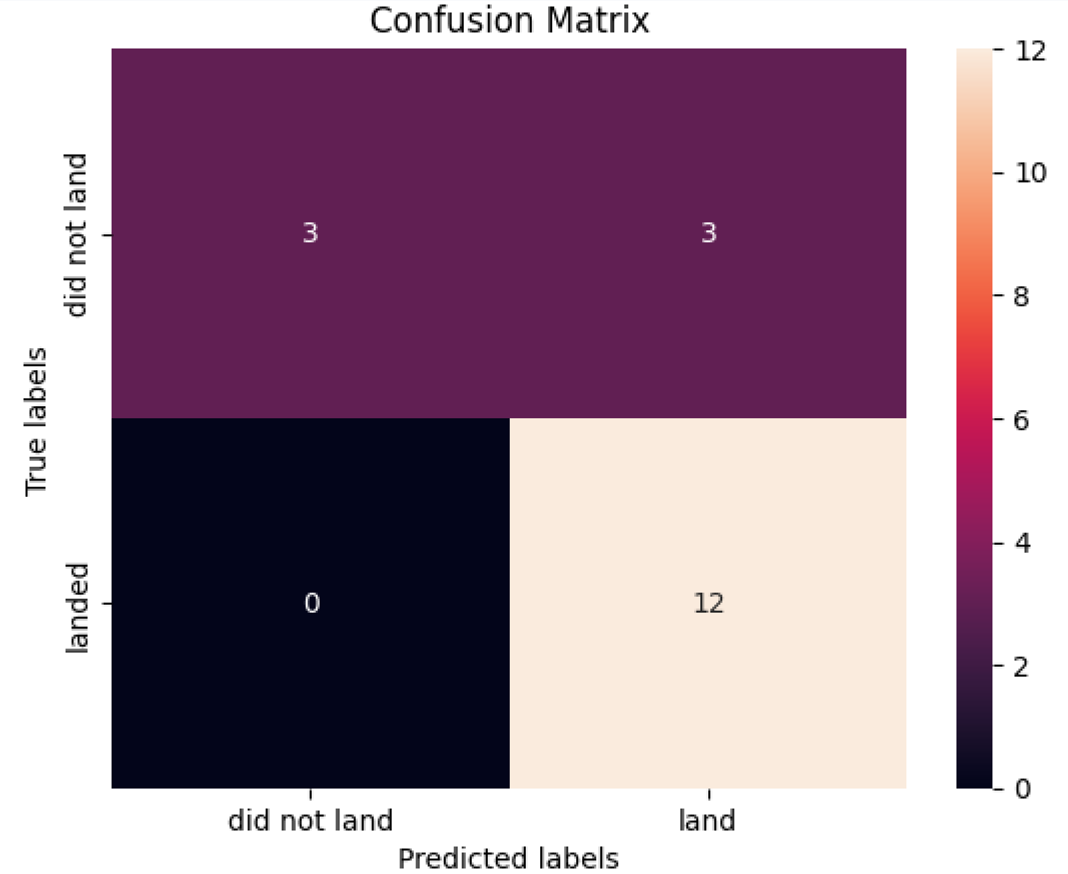
Classification Accuracy

The accuracy values depend on factors like data quality, model selection, and feature engineering. Since all models have the same accuracy, it suggests that the models are equally effective with the data and features provided, or that the problem is straightforward. Further adjustments or optimizations could help highlight differences between the models

- Accuracy for Logistics Regression method: 0.8333333333333334
- Accuracy for Support Vector Machine method: 0.8333333333333334
- Accuracy for Decision tree method: 0.8888888888888888
- Accuracy for K nearest neighbors method: 0.8333333333333334

Confusion Matrix

Since the accuracy values are identical for all four methods, the confusion matrices are also the same. This means that each model is making the same predictions, with the same numbers of true positives, false positives, true negatives, and false negatives.
As a result, the models are performing equally well or poorly on the given data, leading to identical accuracy and confusion matrix results



Conclusions

The models show similar performance, with no clear best since they produce almost identical results. Lighter payloads tend to perform better than heavier ones. KSC LC 39A has the highest success rate. Orbits like ES-L1, GEO, HEO, and SSO also have the best success rates. Overall, launches have improved over time, showing a positive trend.

Thank you!