IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Febrian Dwi Fazri
22 July 2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - Exploratory Data Analysis with Data Visualization
  - Exploratory Data Analysis 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

Background and Context of the Project:

SpaceX stands out as the most successful company in the commercial space industry, revolutionizing space travel by making it more cost-effective. They offer Falcon 9 rocket launches at $62 million per launch, significantly undercutting competitors whose costs can exceed $165 million per launch. A key factor in SpaceX's cost savings is their ability to reuse the first stage of the rocket. Thus, accurately predicting whether the first stage will land successfully can directly impact the overall launch cost.

Questions to Explore:

- How do variables like payload mass, launch site, number of flights, and orbits influence the success of the first stage landing?
- Is there an increasing trend in the rate of successful landings over the years?
- Which machine learning algorithm performs best for predicting the reuse of the first stage in SpaceX launches?

Section 1

Methodology

Methodology

Executive Summary

Data Collection Methodology:

- I used the SpaceX Rest API for gathering data.
- I also did some web scraping from Wikipedia.

Data Wrangling Process:

- We filtered the data.
- Dealt with any missing values.
- Used One Hot Encoding to prepare the data for binary classification.

Exploratory Data Analysis (EDA):

- Explored the data visually using visualization techniques and SQL queries.

Interactive Visual Analytics:

- Used Folium and Plotly Dash for interactive visual analysis.

Predictive Analysis:

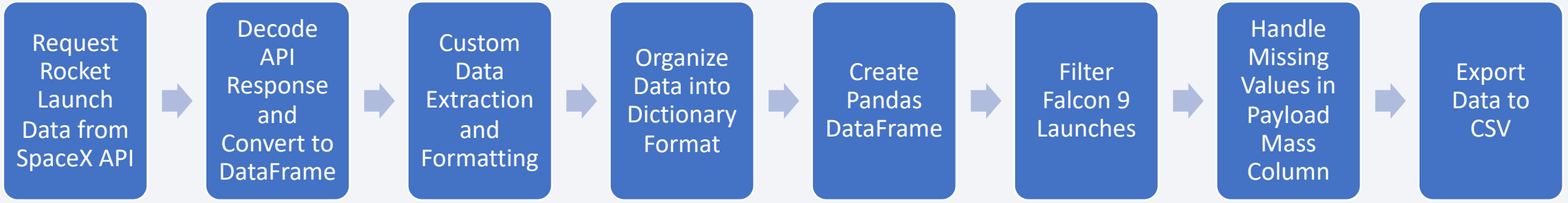
- Built, fine-tuned, and evaluated classification models to achieve the best possible results.

Data Collection

I collected my data using a combination of methods: making API requests to SpaceX's REST API and scraping data from a table on SpaceX's Wikipedia page. This dual approach ensured we gathered comprehensive information about the launches for a thorough analysis.

Data obtained from SpaceX's REST API includes columns like FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, and Latitude.

Data obtained from Wikipedia scraping includes columns such as Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, and Time.



Data Collection – SpaceX API

I collected my data using a combination of methods: making API requests to SpaceX's REST API and scraping data from a table on SpaceX's Wikipedia page. This dual approach ensured we gathered comprehensive information about the launches for a thorough analysis.

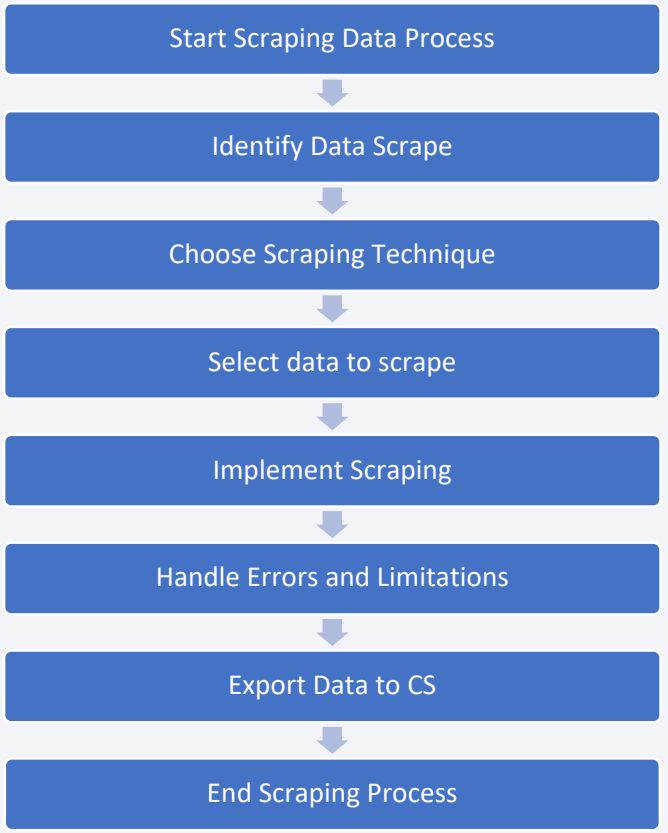
Data obtained from SpaceX's REST API includes columns like FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, and Latitude.

Data obtained from Wikipedia scraping includes columns such as Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, and Time.

Data Collection

Data Collection - Scraping

To gather SpaceX launch data, I first scoped out what I needed—stuff like flight numbers, dates, booster versions, payload masses, orbits, launch sites, and outcomes. Then, I hit up SpaceX's Wikipedia page to grab all that juicy info using web scraping. After snagging the raw HTML data, I sifted through it to fish out the important bits and pieces. Once I had everything sorted, I cleaned up the data, ditching any junk and fixing up missing bits. Finally, I dumped the nicely polished data into a CSV file for easy access later on.

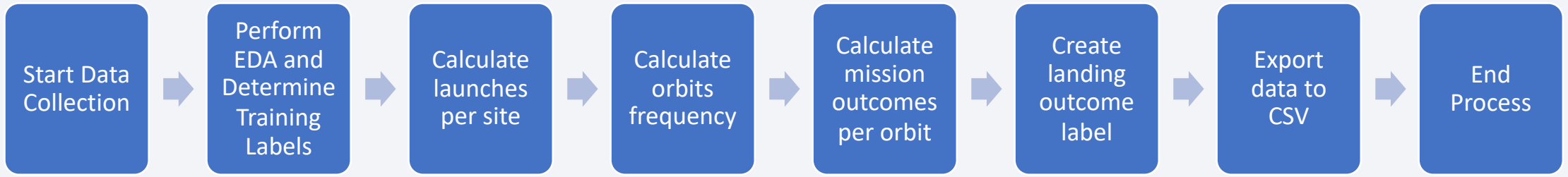


Data Scraping

Data Wrangling

In the dataset, there are various scenarios where the booster did not successfully land. Sometimes, the landing attempt failed due to accidents. For instance, 'True Ocean' indicates a successful landing in a specific ocean region, while 'False Ocean' means an unsuccessful attempt in the ocean. Similarly, 'True RTLS' denotes a successful ground pad landing, whereas 'False RTLS' indicates an unsuccessful ground pad landing. 'True ASDS' signifies a successful landing on a drone ship, whereas 'False ASDS' indicates an unsuccessful attempt on a drone ship.

These outcomes are mainly converted into training labels, where '1' signifies a successful booster landing and '0' denotes an unsuccessful attempt.



Data Wrangling

EDA with Data Visualization

- I plotted several 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
- Scatter plots illustrate how variables relate to each other. If there's a clear relationship, these insights can be used in a machine learning model.
- Bar charts compare different categories against measured values, showcasing specific relationships between them.
- Line charts track data trends over time, providing insights into time-based patterns (time series).

EDA Visualization

EDA with SQL

I conducted several SQL queries:

- Showed the names of unique launch sites in the space missions
- Displayed 5 records where launch sites start with 'CCA'
- Calculated the total payload mass carried by boosters launched by NASA (CRS)
- Found the average payload mass carried by booster version F9 v1.1
- Listed the date of the first successful landing outcome on a ground pad
- Identified boosters that successfully landed on a drone ship with payload mass between 4000 and 6000
- Counted the total number of successful and failed mission outcomes
- Listed booster versions that carried the maximum payload mass
- Identified failed landing outcomes on a drone ship, including booster versions and launch site names in 2015
- Ranked landing outcomes (Failure (drone ship) or Success (ground pad)) between June 4, 2010, and March 20, 2017, in descending order.

EDA SQL

Build an Interactive Map with Folium

Markers of all Launch Sites:

- Added markers with circles, popup labels, and text labels for NASA Johnson Space Center using its latitude and longitude coordinates as the starting location.
- Added markers with circles, popup labels, and text labels for all launch sites using their latitude and longitude coordinates to display their geographical locations and their proximity to the Equator and coasts.

Colored Markers of the launch outcomes for each Launch Site:

- Added colored markers for successful launches (green) and failed launches (red) using Marker Cluster to visualize which launch sites have relatively high success rates.

Distances between a Launch Site to its proximities:

- Added colored lines to illustrate distances between Launch Site KSC LC-39A (as an example) and its nearby features such as railways, highways, coastlines, and the closest city.

Map with Folium

Build a Dashboard with Plotly Dash

Launch Sites Dropdown List:

- Added a dropdown list so you can choose the Launch Site you're interested in.

Pie Chart showing Success Launches (All Sites/Certain Site):

- Added a pie chart to display the total count of successful launches across all sites. If you select a specific Launch Site, it also shows the breakdown of Success vs. Failed launches for that site.

Slider of Payload Mass Range:

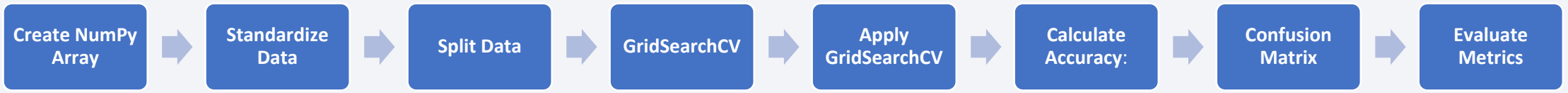
- Added a slider that lets you choose the range of Payload Mass you want to explore.

Scatter Chart of Payload Mass vs. Success Rate for the different Booster Versions:

- Added a scatter chart to visualize how Payload Mass correlates with Launch Success for different Booster Versions.

Predictive Analysis (Classification)

I start by creating a NumPy array from the "Class" column in my dataset. Next, I standardize my data using StandardScaler, which both fits and transforms it for consistency. Then, I split my dataset into training and testing sets using the train_test_split function. I proceed by setting up a GridSearchCV object with cv = 10 to explore and determine the best parameters for my models: Logistic Regression, SVM, Decision Tree, and KNN. After finding optimal parameters, we apply GridSearchCV to each model. Following this, I calculate the accuracy of each model on my test data using the .score() method. To further evaluate performance, I analyze the confusion matrix for each model, gaining insights into prediction accuracy. Finally, we determine the best-performing method by comparing metrics such as Jaccard score and F1 score.



Results

Exploratory Data Analysis (EDA) Results:

In our exploration of the data, we uncovered several key insights:

Launch Sites Analysis: We calculated the number of launches from each site, revealing significant variations in launch frequency across different sites.

Orbit Analysis: We analyzed the distribution of launch orbits, identifying the most common types and their respective frequencies.

Payload Mass Insights: Investigating the payload mass distribution highlighted its variability across launches, essential for understanding mission capabilities.

Success Rates Over Time: We explored how the success rates of missions have evolved over the years, providing insights into SpaceX's operational improvements.

Correlation Analysis: Scatter plots between variables like Payload Mass and Success Rates suggested potential relationships, informing our predictive modeling.

Predictive Analysis Results:

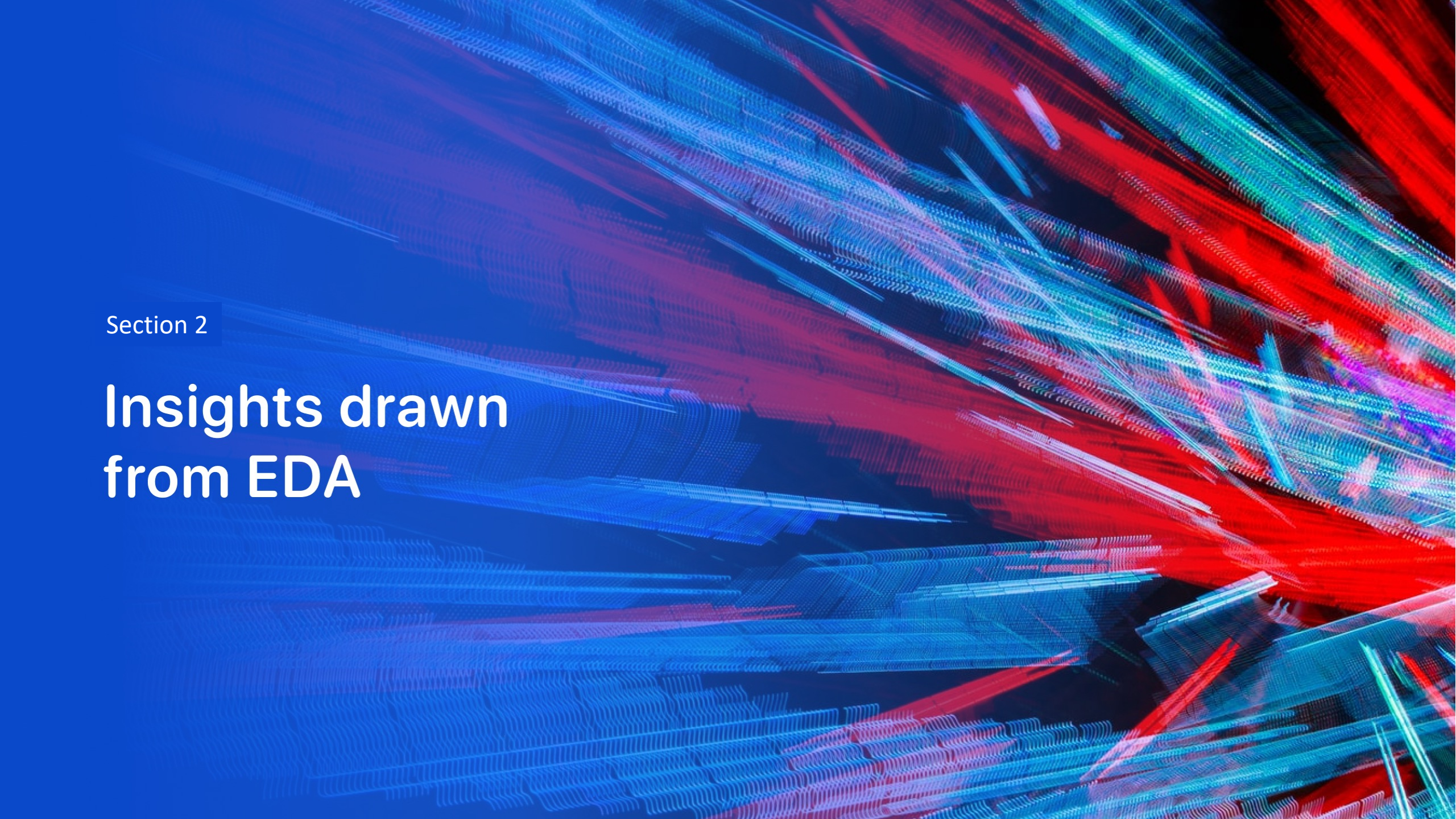
In our predictive analysis using classification models, we evaluated several algorithms to determine the best performer:

Logistic Regression: Achieved an accuracy score of 87%, with a Jaccard index of 83% and F1-score of 91% on the test set.

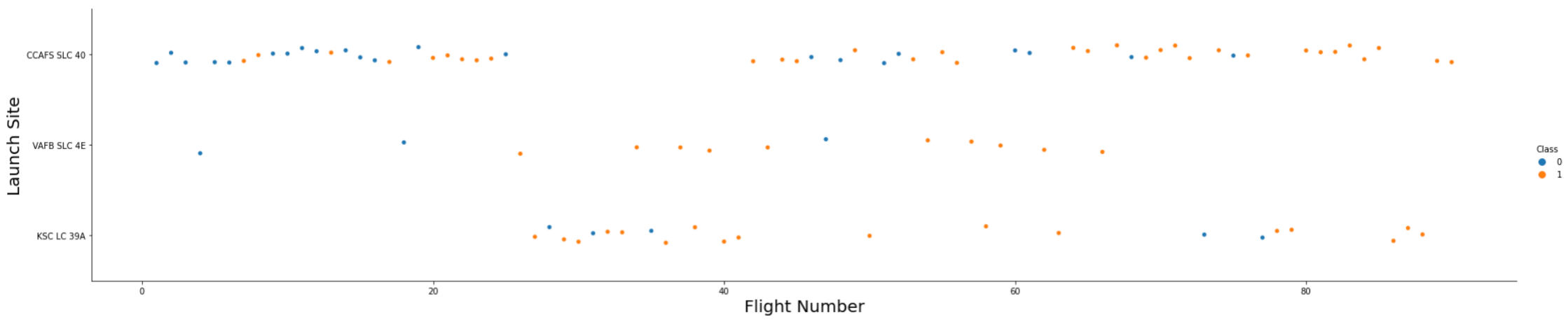
Support Vector Machine (SVM): Utilizing the Sigmoid kernel, SVM achieved an accuracy score of 88%, with a Jaccard index of 85% and F1-score of 92%.

Decision Tree: With optimal parameters, Decision Tree achieved an accuracy score of 91%, a Jaccard index of 88%, and an F1-score of 94%.

K-Nearest Neighbors (KNN): After tuning, KNN achieved an accuracy score of 86%, with a Jaccard index of 82% and F1-score of 90%.


Section 2
Insights drawn from EDA

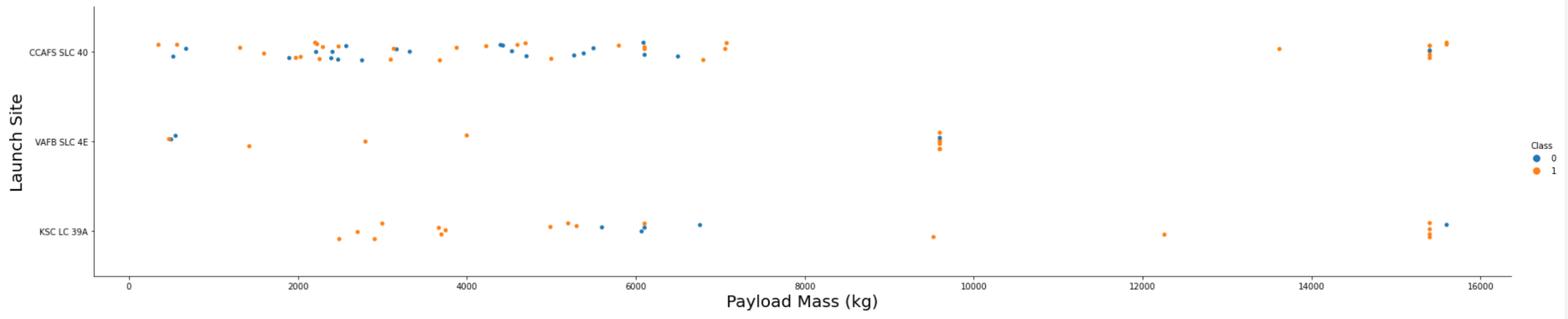
Flight Number vs. Launch Site



So, here's the deal: the first flights?

- Total bust. But the latest ones? Nailed it every time.
- You know that launch site at CCAFS SLC 40? Like, half of all launches happen there.
- Now, VAFB SLC 4E and KSC LC 39A? They've got way better success rates.
- Seems like every new launch just keeps getting better at succeeding, right?

Payload vs. Launch Site



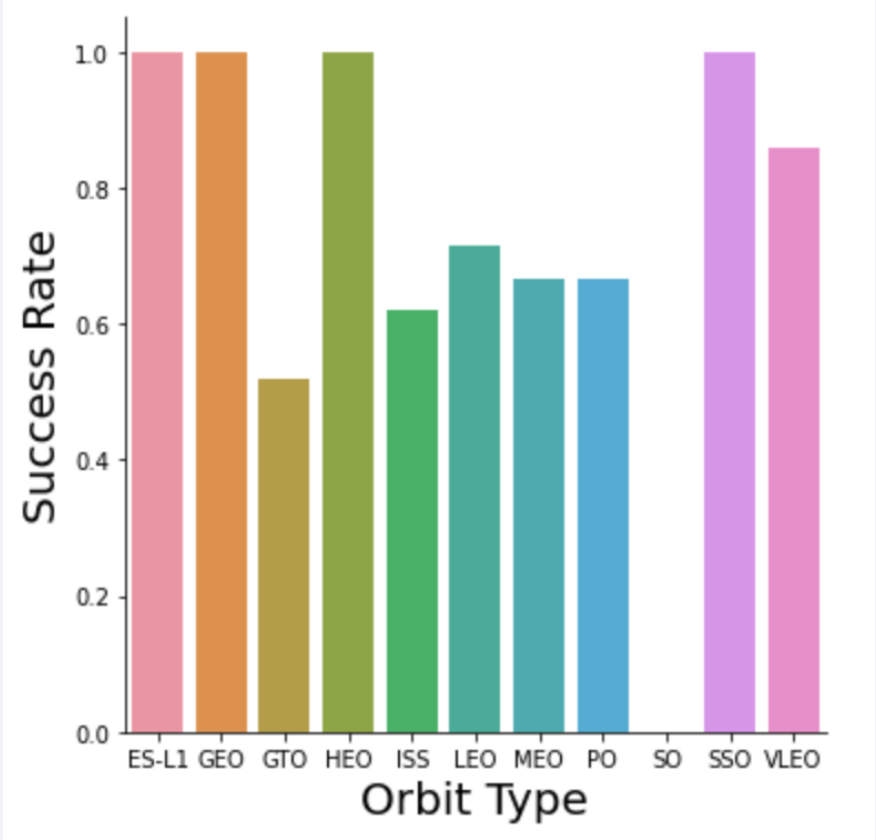
Here's the scoop:

- When it comes to launch sites, the heavier the payload, the better the success rate.
- Most of the launches carrying over 7000 kg of payload were total wins.
- And check this out: KSC LC 39A has a perfect track record with payloads under 5500 kg too.

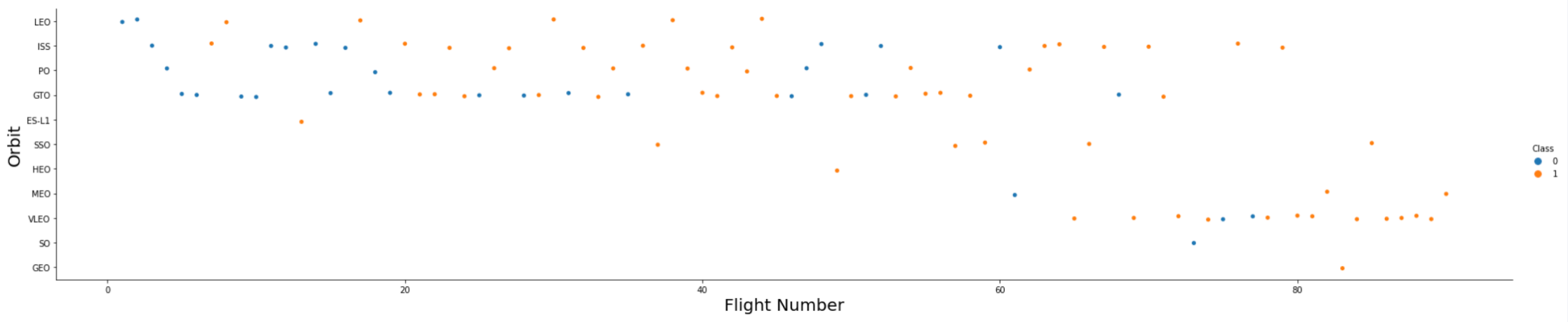
Success Rate vs. Orbit Type

Alright, here's the breakdown:

- We've got some orbits that are totally ace with a 100% success rate:
- ES-L1
- GEO
- HEO
- SSO
- But then, there are orbits like SO that just haven't had any luck—zero success rate.
- And in between, orbits like GTO, ISS, LEO, and MEO are doing okay with success rates ranging from 50% to 85%.

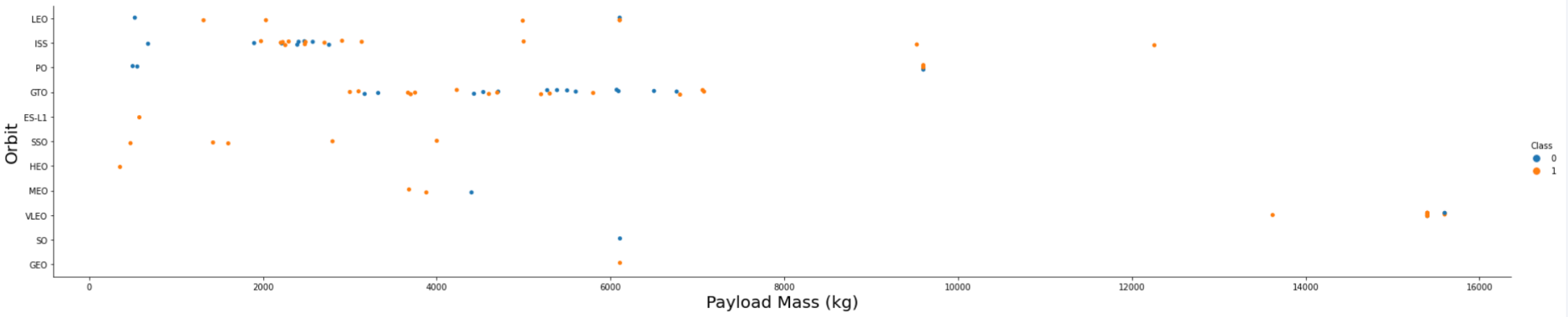


Flight Number vs. Orbit Type



So, in the LEO orbit, it looks like success rates go up as more flights happen. But in the GTO orbit, flight numbers don't seem to make a difference to success rates.

Payload vs. Orbit Type

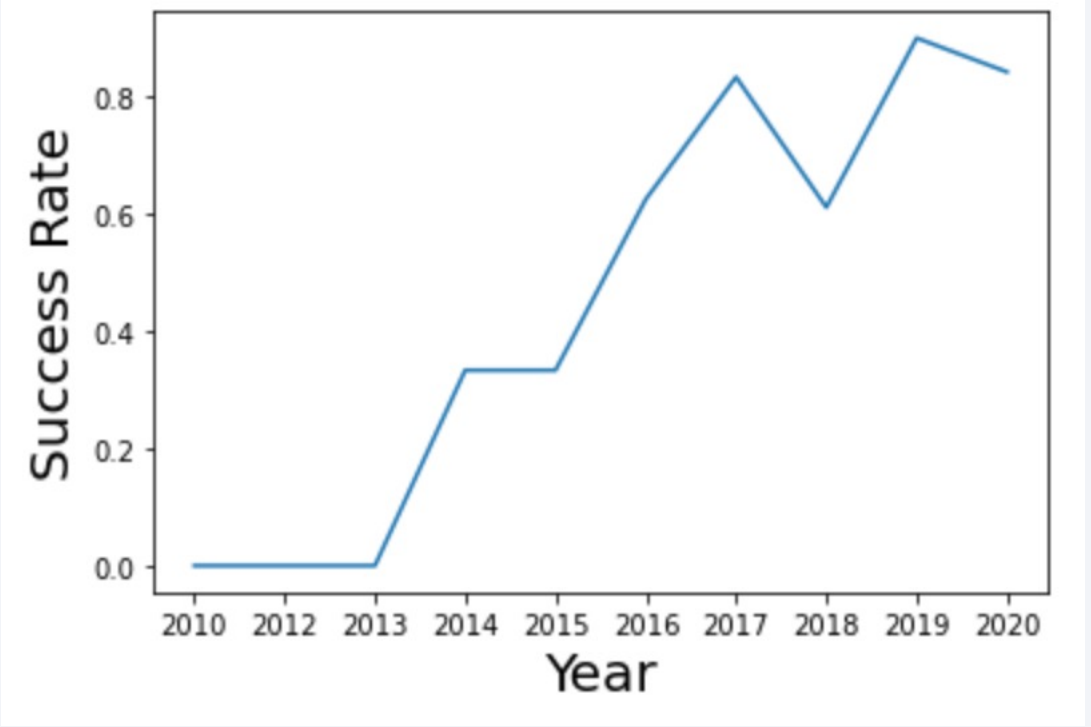


Check this out:

Heavy payloads kinda mess with GTO orbits, but they're actually a big plus for GTO and Polar LEO (ISS) orbits.

Launch Success Yearly Trend

You can see that the success rate has been on the rise since 2013, peaking in 2020.



All Launch Site Names

%sql SELECT DISTINCT launch_site FROM SPACEXDATASET;

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

Launch Site Names Begin with 'CCA'

%sql select * from SPACEXDATASET where launch_site like 'CCA%' limit 5;

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

Total Payload Mass

```
%sql SELECT SUM(payload_mass__kg_) AS total_payload_mass
        FROM SPACEXDATASET WHERE customer = 'NASA (CRS)';
```

average_payload_mass
2534

Average Payload Mass by F9 v1.1

```
%sql SELECT avg(payload_mass__kg_) AS average_payload_mass
        FROM SPACEXDATASET WHERE booster_version LIKE '%F9 v1.1%';
```

average_payload_mass
2534

First Successful Ground Landing Date

```
%sql SELECT MIN(date) AS first_successful_landing
    FROM SPACEXDATASET
    WHERE landing__outcome = 'Success (ground pad)';
```

first_successful_landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT booster_version FROM SPACEXDATASET
        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(*) as total_number
        from SPACEXDATASET 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 booster_version from SPACEXDATASET
        where payload_mass__kg_ = (select max(payload_mass__kg_)
        from SPACEXDATASET);
```

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

2015 Launch Records

%%sql select monthname(date) as month, date, booster_version, launch_site, landing__outcome from SPACEXDATASET where landing__outcome = 'Failure (drone ship)' and year(date)=2015;

MONTH	DATE	booster_version	launch_site	landing__outcome
January	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
April	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

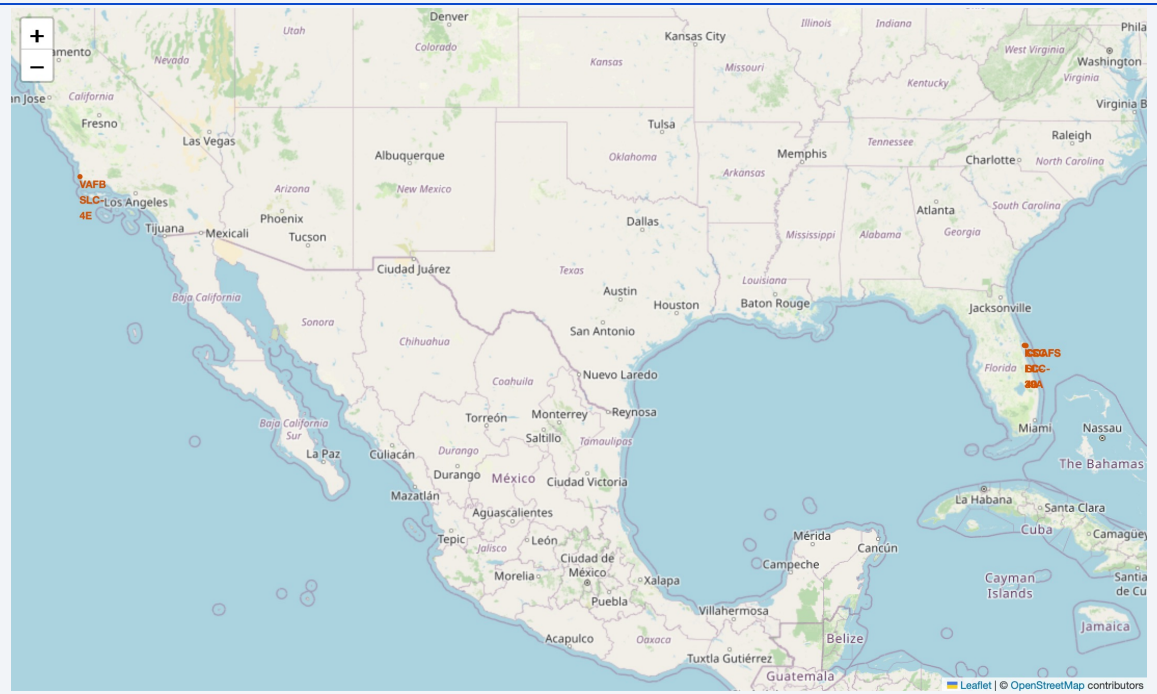
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

```
%%sql select landing__outcome, count(*) as count_outcomes
from SPACEXDATASET
where date between '2010-06-04' and '2017-03-20'
group by landing__outcome
order by count_outcomes desc;
```

landing__outcome	count_outcomes
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

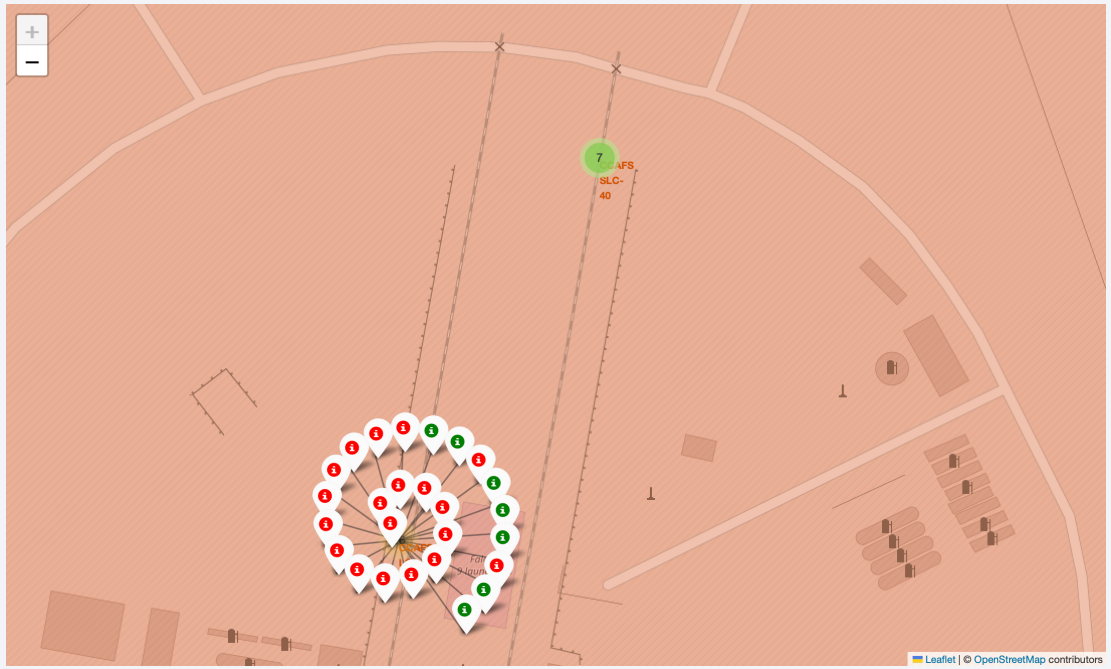

Section 3
Launch Sites
Proximities Analysis

Plotting all the launch sites on a world map



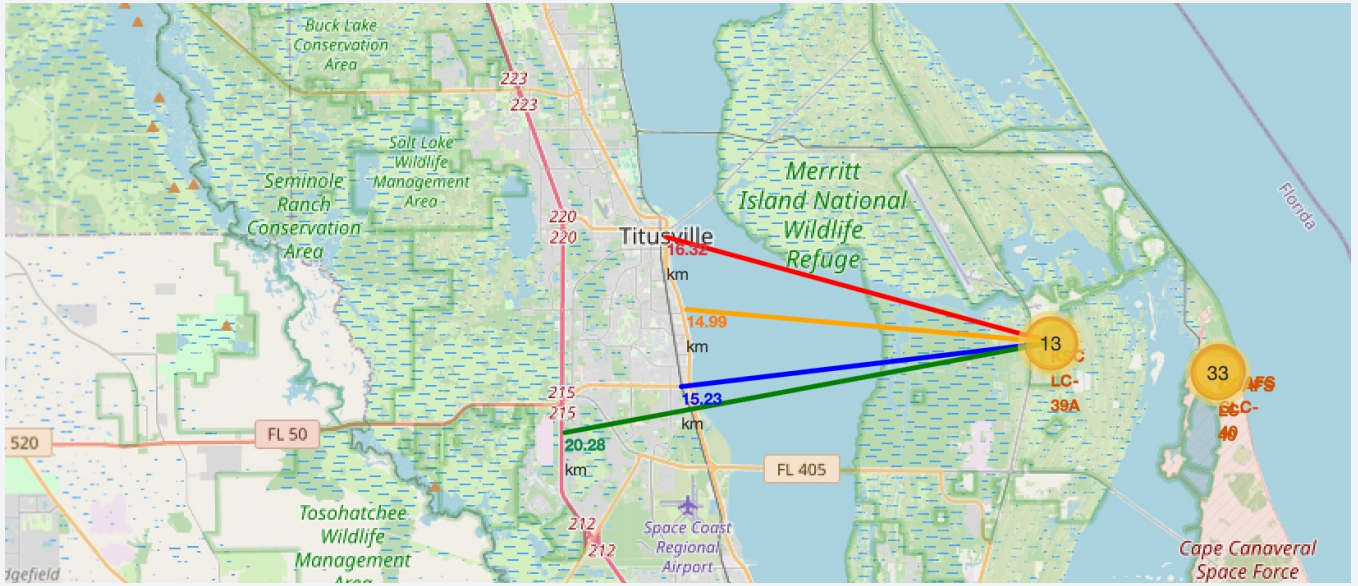
- Most launch sites are near the Equator, where Earth's rotation speed is the fastest—about 1670 km/hour. When rockets launch from here, they inherit some of that speed, thanks to inertia, which helps them maintain orbit.
- All launch sites are super close to the coast. Launching rockets over the ocean reduces the chance of debris falling near people and keeps things safe.

Looking at the map with color-coded launch records



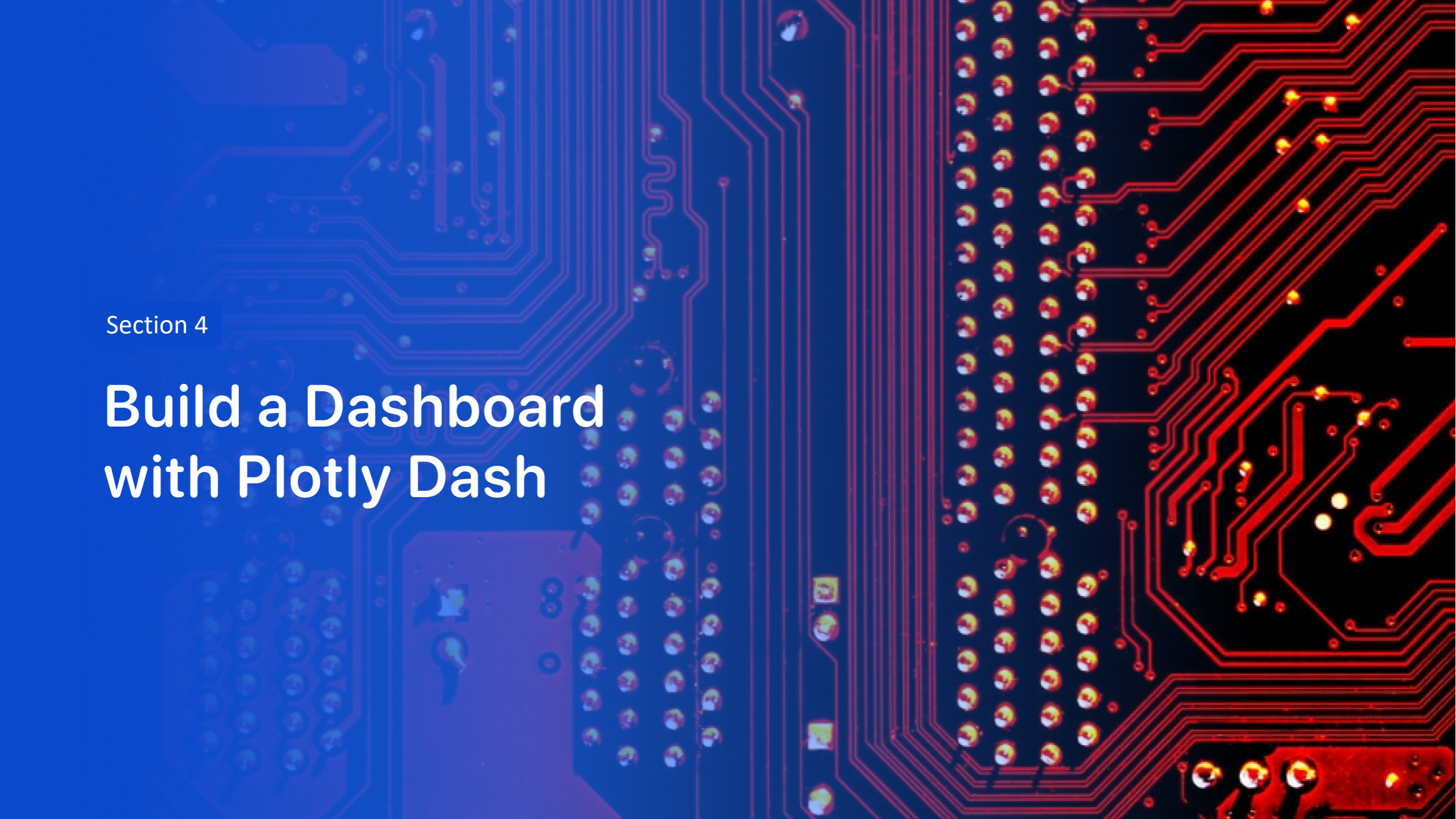
- It's easy to spot which launch sites have done well.
- Green markers mean the launch was a success.
- Red markers show where things didn't go as planned.
- KSC LC-39A stands out with a super high success rate.

Checking out the distance from KSC LC-39A to nearby spots

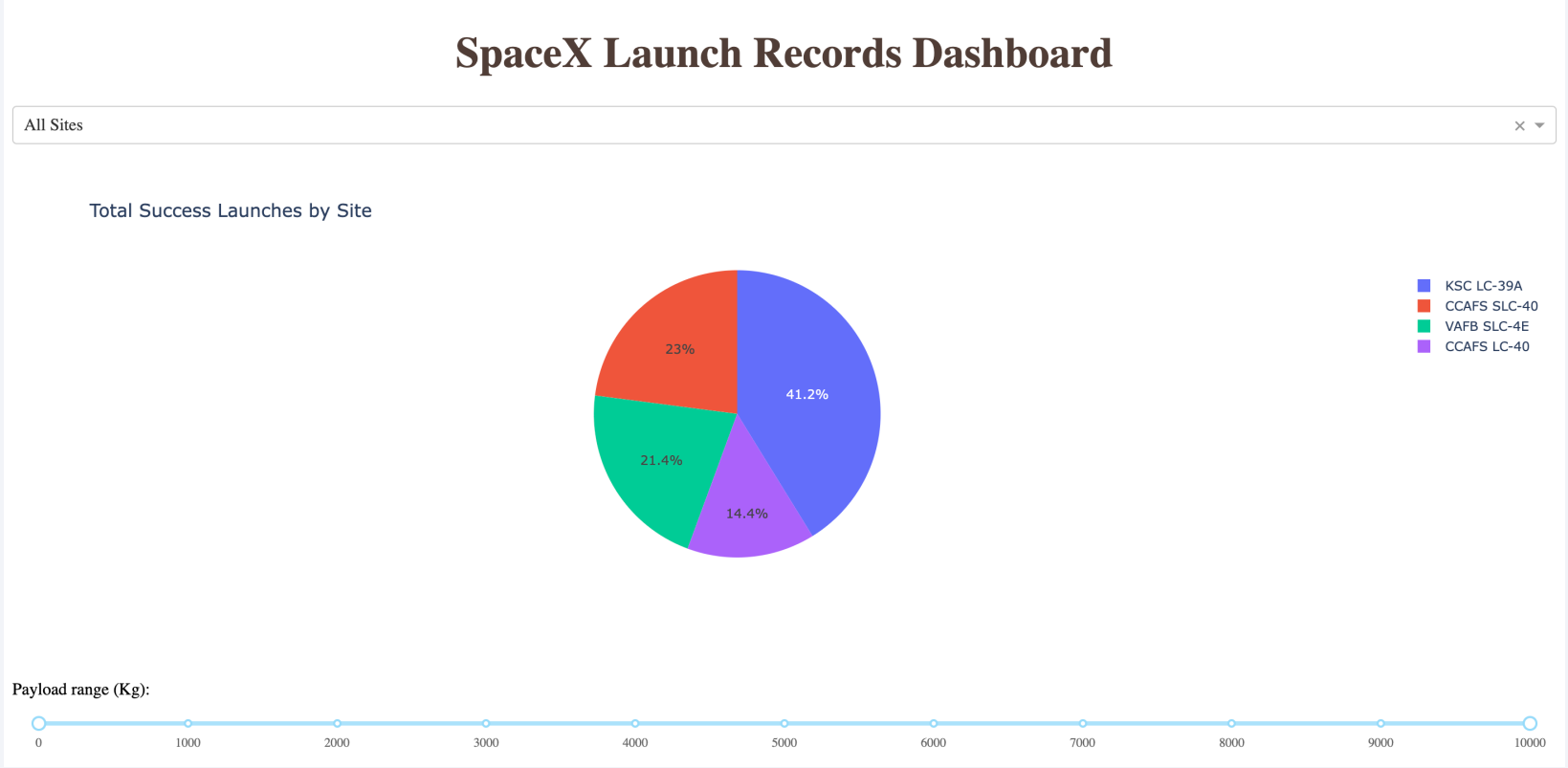


- It's pretty close to a railway (about 15.23 km),
- and not too far from a highway (around 20.28 km).
- It's also near the coastline (just 14.99 km away).
- Plus, it's not far from Titusville, the closest city (about 16.32 km).

When a rocket fails, it can cover 15-20 km in just seconds. That could be risky if it heads toward populated areas.

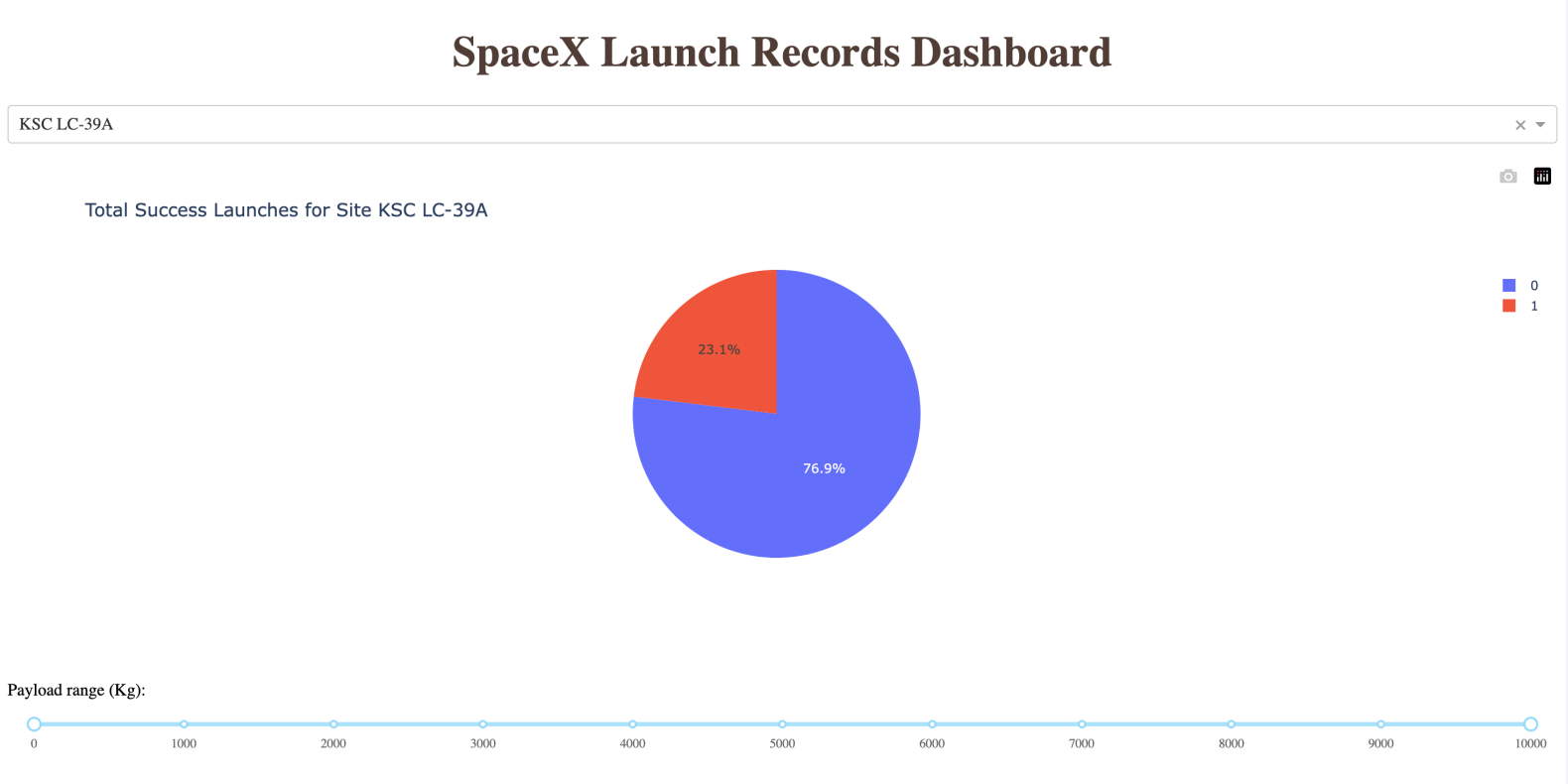

Section 4
Build a Dashboard with Plotly Dash

The Total Number of Successful Launches From Each Site



The chart makes it obvious that out of all the locations, KSC LC-39A has had the most successful launches.

The Launch Site That’s Been Kicking It With The Highest Success Ratio



Basically, KSC LC-39A has nailed it with a 76.9% launch success rate—10 successes and only 3 hiccups along the way.

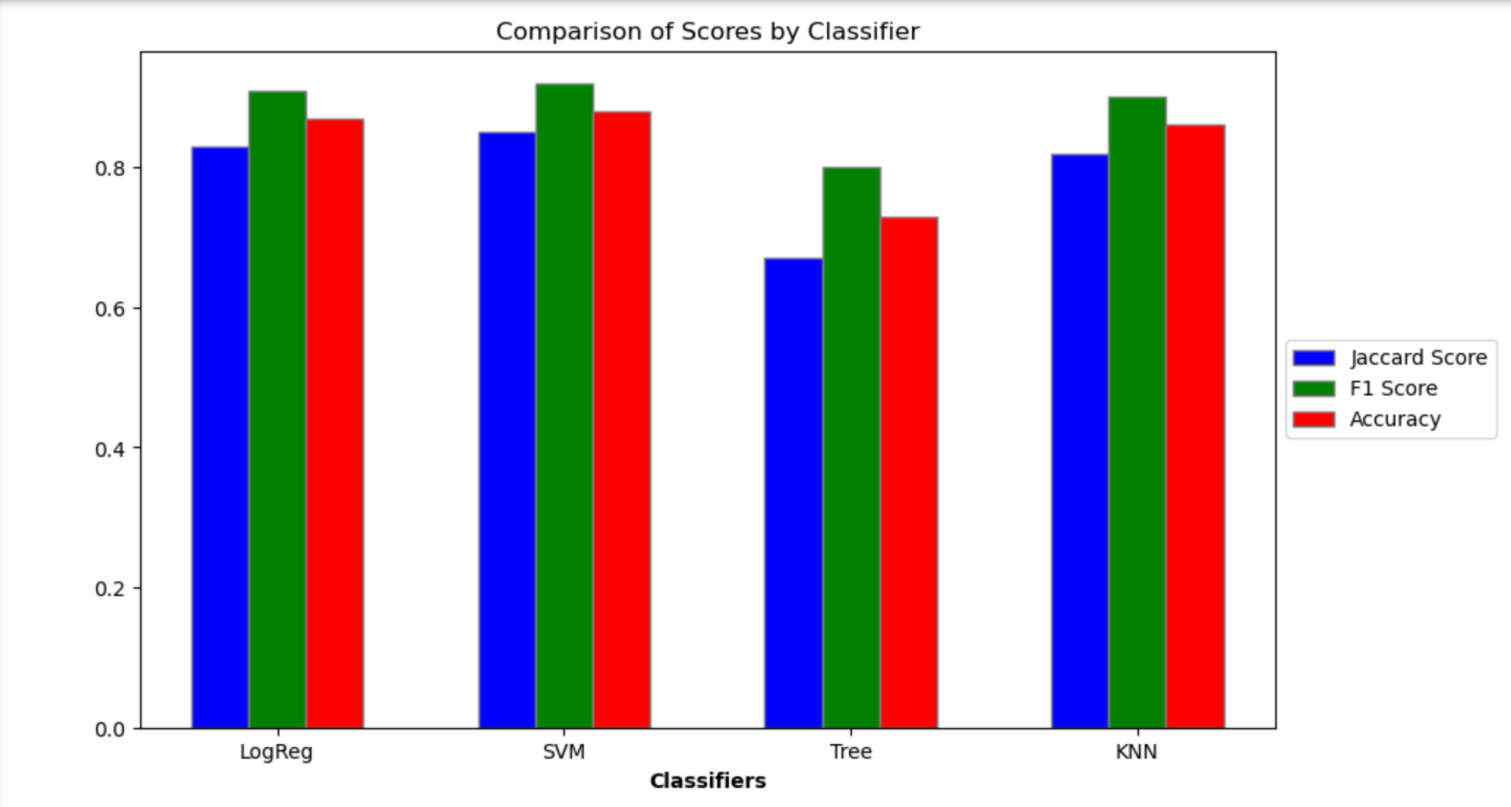
Payload vs. Launch Outcome

The charts basically say that payloads weighing between 2000 and 5500 kg have the best success rate.

Section 5

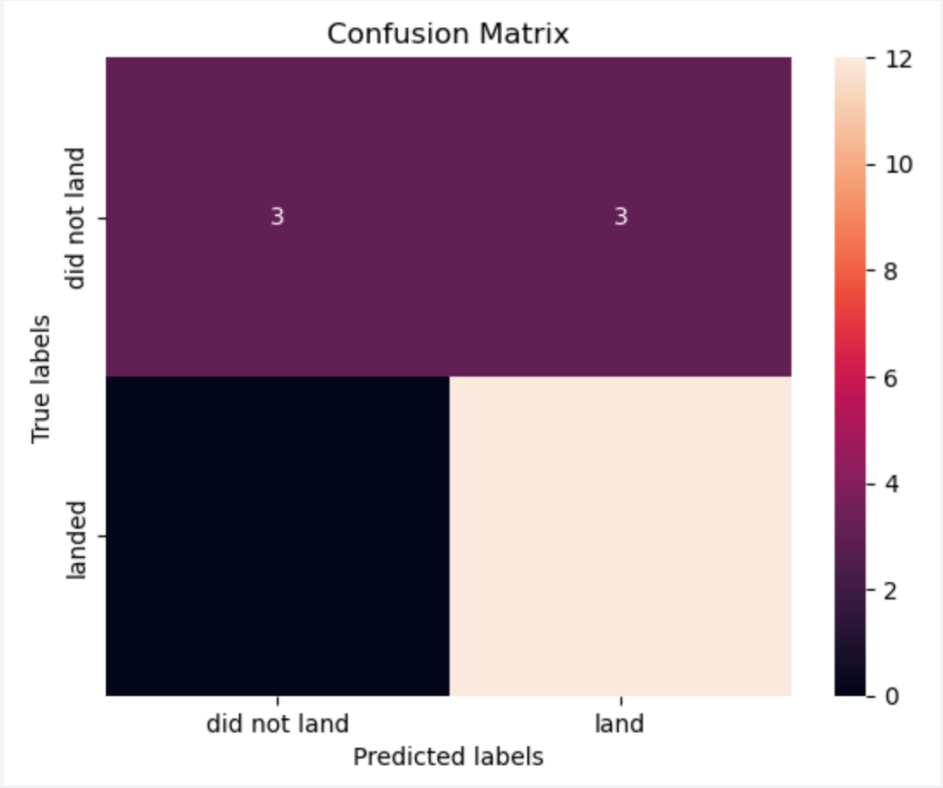
Predictive Analysis (Classification)

Classification Accuracy



So, after looking at all the scores from the dataset, it's pretty clear that the Decision Tree Model is the top dog. Not only does it have higher scores overall, but it also boasts the highest accuracy.

Confusion Matrix



Checking out the confusion matrix, it's clear that logistic regression can tell the classes apart. The main issue here is dealing with false positives.

Conclusions

To wrap it up:

- The Decision Tree Model rocks for this dataset.
- Smaller payload masses tend to perform better than larger ones.
- Most launch sites are near the Equator and close to the coast.
- Success rates improve over the years.
- KSC LC-39A tops the charts for launch success rates.
- Orbits ES-L1, GEO, HEO, and SSO have a perfect 100% success rate.

Appendix

Data Collection: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/Data%20Collection%20API.ipynb

Data Collection with Web Scraping: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/Data%20Collection%20with%20Web%20Scraping.ipynb

Data Wrangling: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/Data%20Wrangling.ipynb

EDA with Data Visualization: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/EDA%20with%20Data%20Visualization.ipynb

Interactive Visual Analytics with Folium: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

Machine Learning Prediction: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/Machine%20Learning%20Prediction.ipynb

SpaceX Dash: https://github.com/Pirouette-pi/datasciencecapstone/blob/main/spacex_dash_app.py

Thank you!