IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Heather Navarro
6/28/2024

Outline

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

Executive Summary

- RESTful API & web scraping to gather data
- Plotly Dash dashboard
- Folium interactive map
- Machine learning to make predictions

Introduction

- SpaceX can reuse the first stage of the rocket launch which saves them money in the Space Race. If we can determine if the first stage will land, we can determine the cost of a launch.

- Using data gathered from the web we can explore factors that influence the success or failure of the landing and develop machine learning to find the method that performs the best in predicting it.

- The question on everyone's mind: Will the first stage of the Falcon 9 land successfully?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX web API
  - Web scraping Wikipedia
- Perform data wrangling
  - Utilizing Python, particularly Pandas
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Test SVM, Classification Trees, and Logistic Regression

Data Collection

Input URL to request and parse data using GET request

Use JSON to decode response

Convert to Pandas DataFrame

Filter to only include Falcon 9 launches

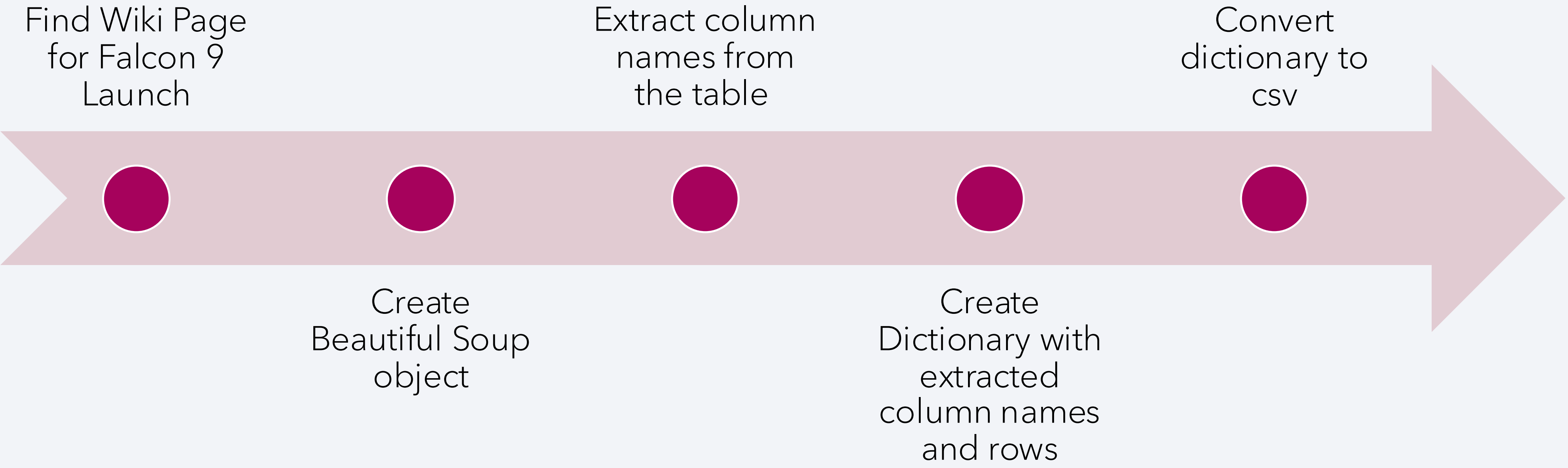
Web scrape Wikipedia for further details using BeautifulSoup

Data Collection – SpaceX API

- **Use the get request to the SpaceX API to collect data, clean it then perform basic data wrangling and formatting.**
- https://github.com/hnavarro25/SpaceX/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Applied web scrapping to Falcon 9 launch records from Wiki with BeautifulSoup
- https://github.com/hnavarro25/SpaceX/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- https://github.com/hnavarro25/SpaceX/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

Calculate the number of launches at each site

Calculate the number and occurrence of each orbits

Calculate the number and coccurance of mission coutcome per orbit type

Create landing outcome label from outcome column

Exporte results to csv

EDA with Data Visualization

- Scatter Plots
  - compare flight number vs payload mass vs launch site
  - compare payload vs launch site
  - compare orbit type vs flight vs payload
- Bar Chart
  - View success rate of each orbit
- Line Plot
  - View success rate and date
- https://github.com/hnavarro25/SpaceX/blob/main/edadataviz.ipynb

EDA with SQL

- Loaded SpaceX dataset into PostgreSQL db to perform SQL queries:
  - Name of unique launch sites
  - Total payload mass carried by boosters launched
  - Avg payload mass carried by booster
  - Total successful and failure missions
  - Failed landing outcomes in drone ship
- https://github.com/hnavarro25/SpaceX/blob/main/jupyter-labs-eda-sql-coursera_sqllite%20(1).ipynb

Build an Interactive Map with Folium

- Marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- Using color-labeled marker clusters, idenitifed which launch sites have high success.
- Calculated distances between a launch site to railways, highways, coastlines and distance from cities.

Build a Dashboard with Plotly Dash

- Interactive dashboard with pie charts and scatter plots.
    - Pie charts used to show total launches by certain sites
    - Scatter graphs to see Outcome and Payload Mass for different booster versions
- https://github.com/hnavarro25/SpaceX/blob/main/lab_jupyter_launch_site_location%20(1).ipynb

Predictive Analysis (Classification)

- Load data - Define variables and target (Class) - Standardize the data - Split into training and test data - Find best hyperparameter for SVM, Logistic Regression, KNN, Decision Tree - Find method performs best
- Once I fixed the max_features parameter from 'auto' to 'log2' (there is no auto option) for decision tree it resolved the error.
- Decision tree did best at 94%, all other models performed at 83%
- https://github.com/hnavarro25/SpaceX/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

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

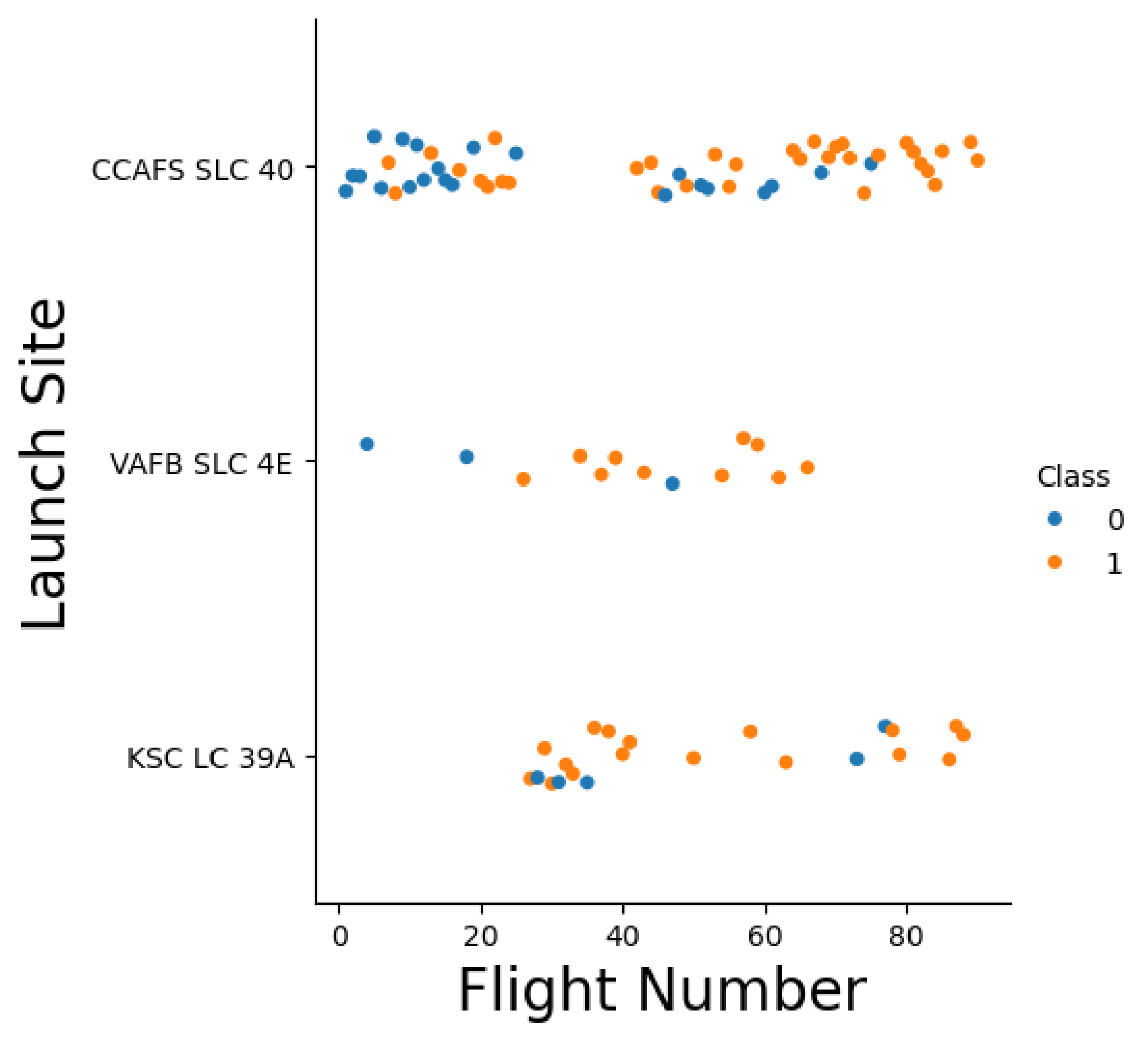
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site

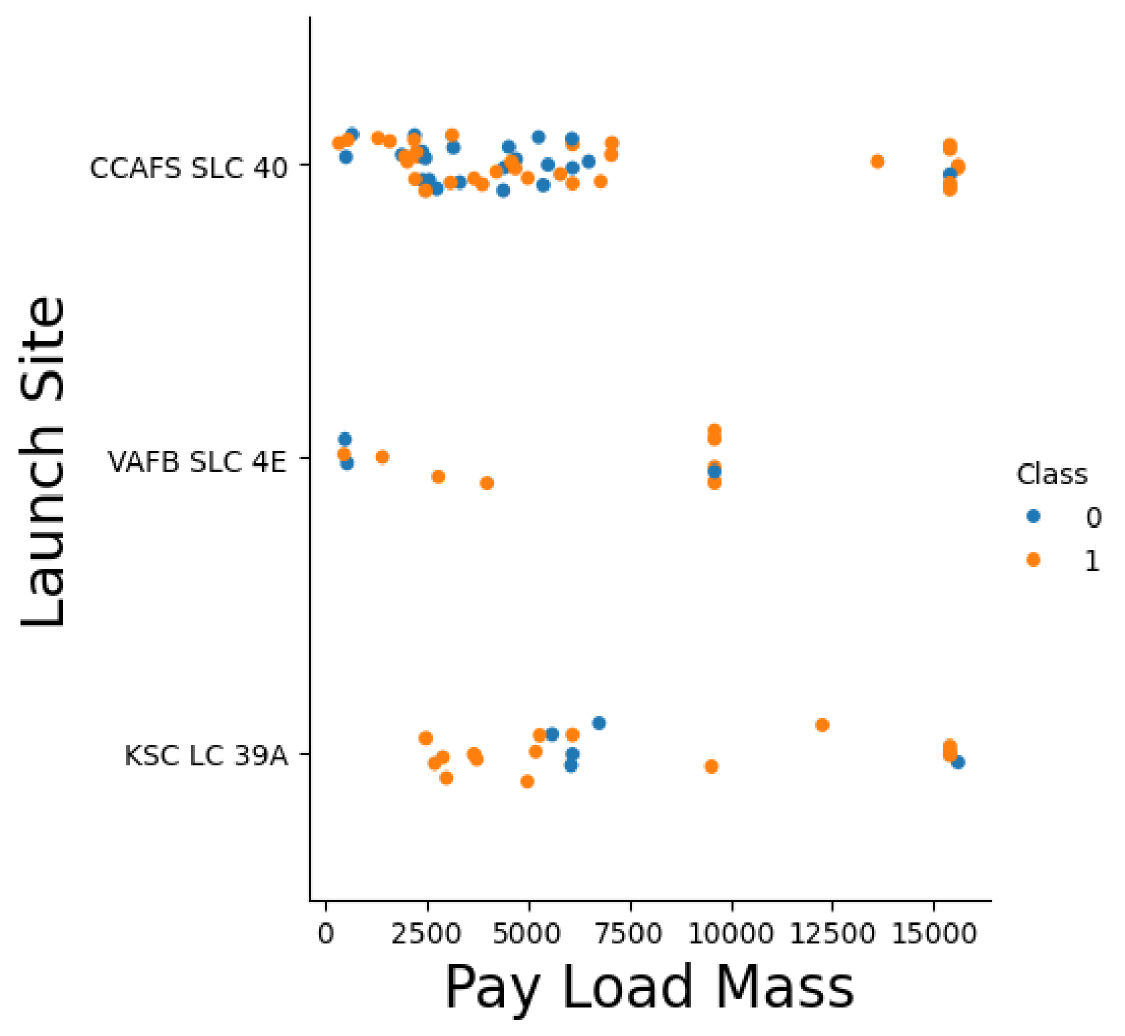
- The higher the flight number the higher the odds of success.



Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site

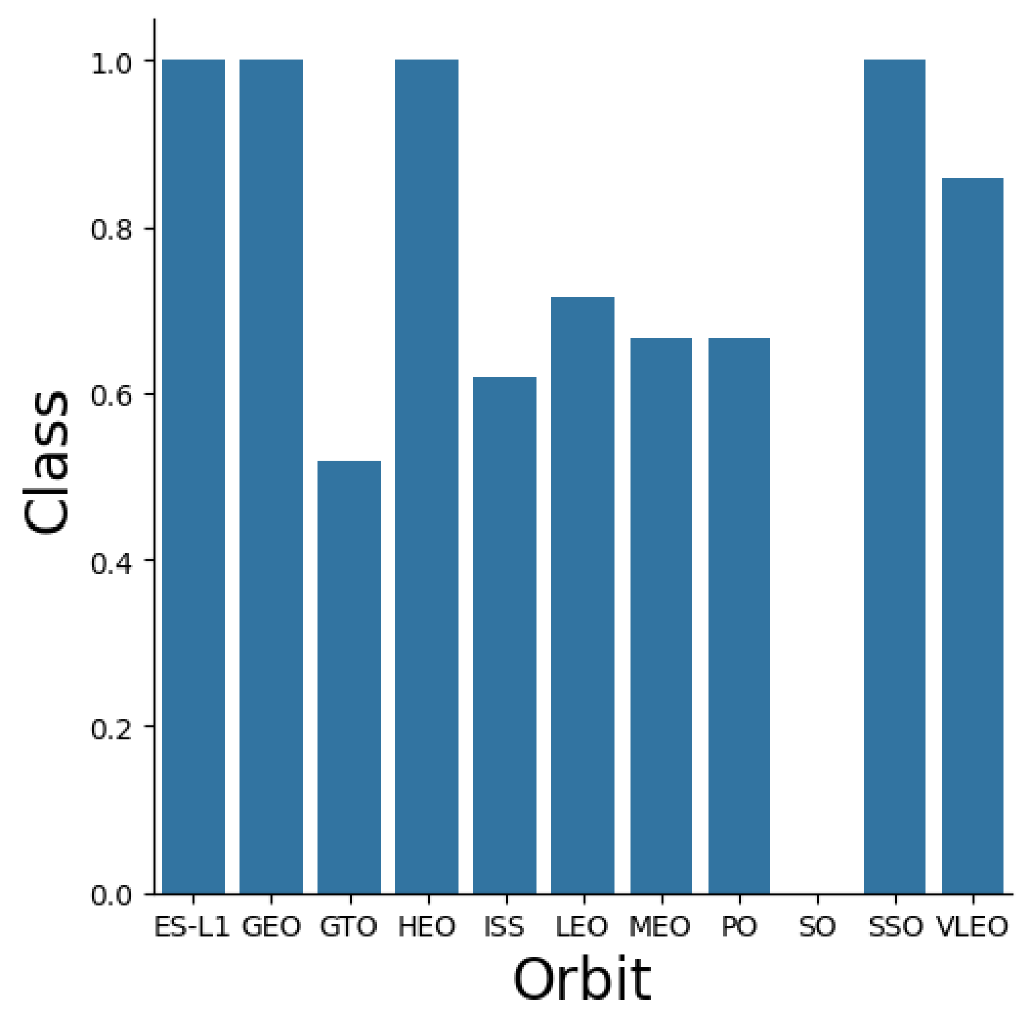
- VAFB-SLC launchsite has no rockets higher pay load mass.



Success Rate vs. Orbit Type

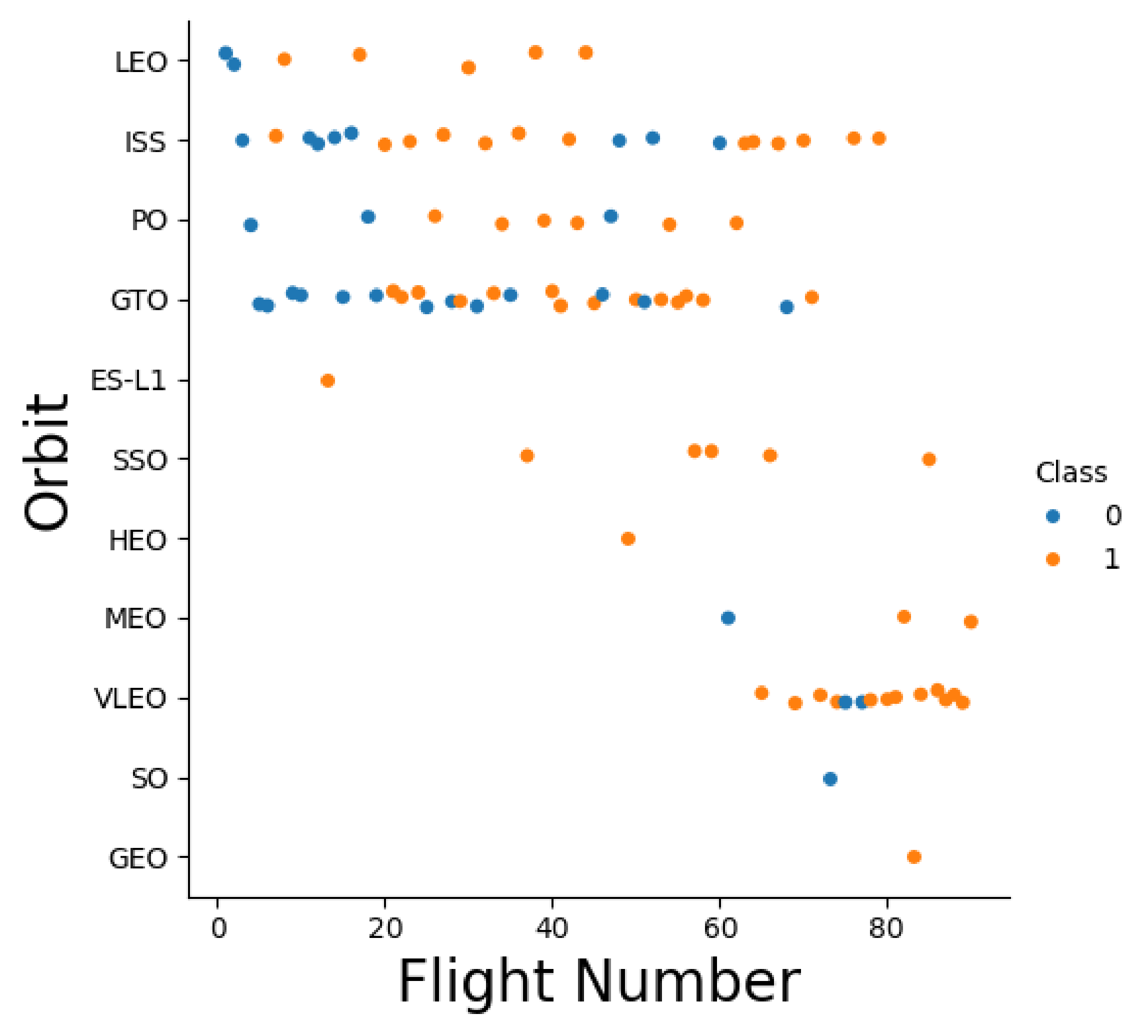
- Show a bar chart for the success rate of each orbit type

- ES-L1, GEO, HEO and SSO have better odds then the other orbits.



Flight Number vs. Orbit Type

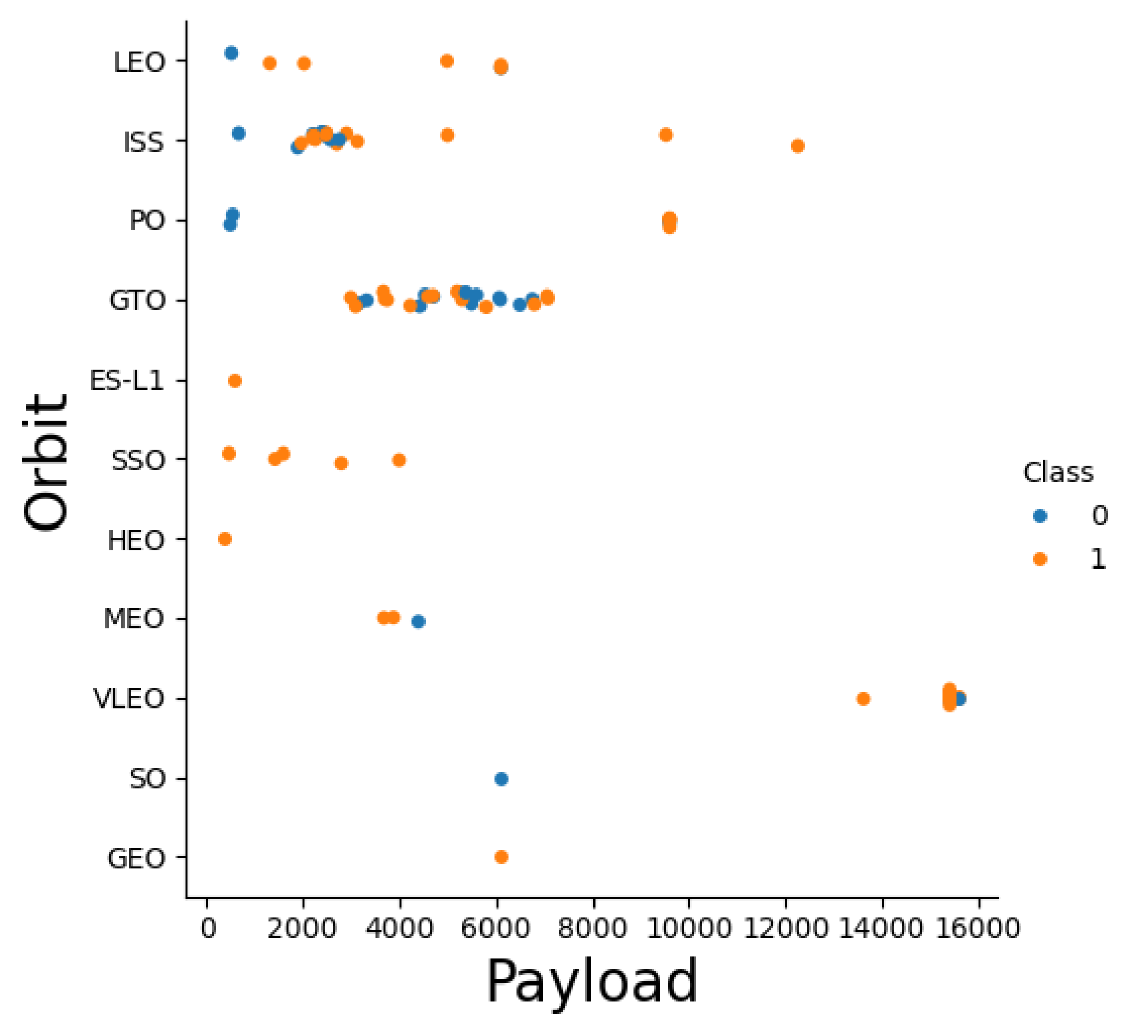
- Show a scatter point of Flight number vs. Orbit type
- In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit



Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type

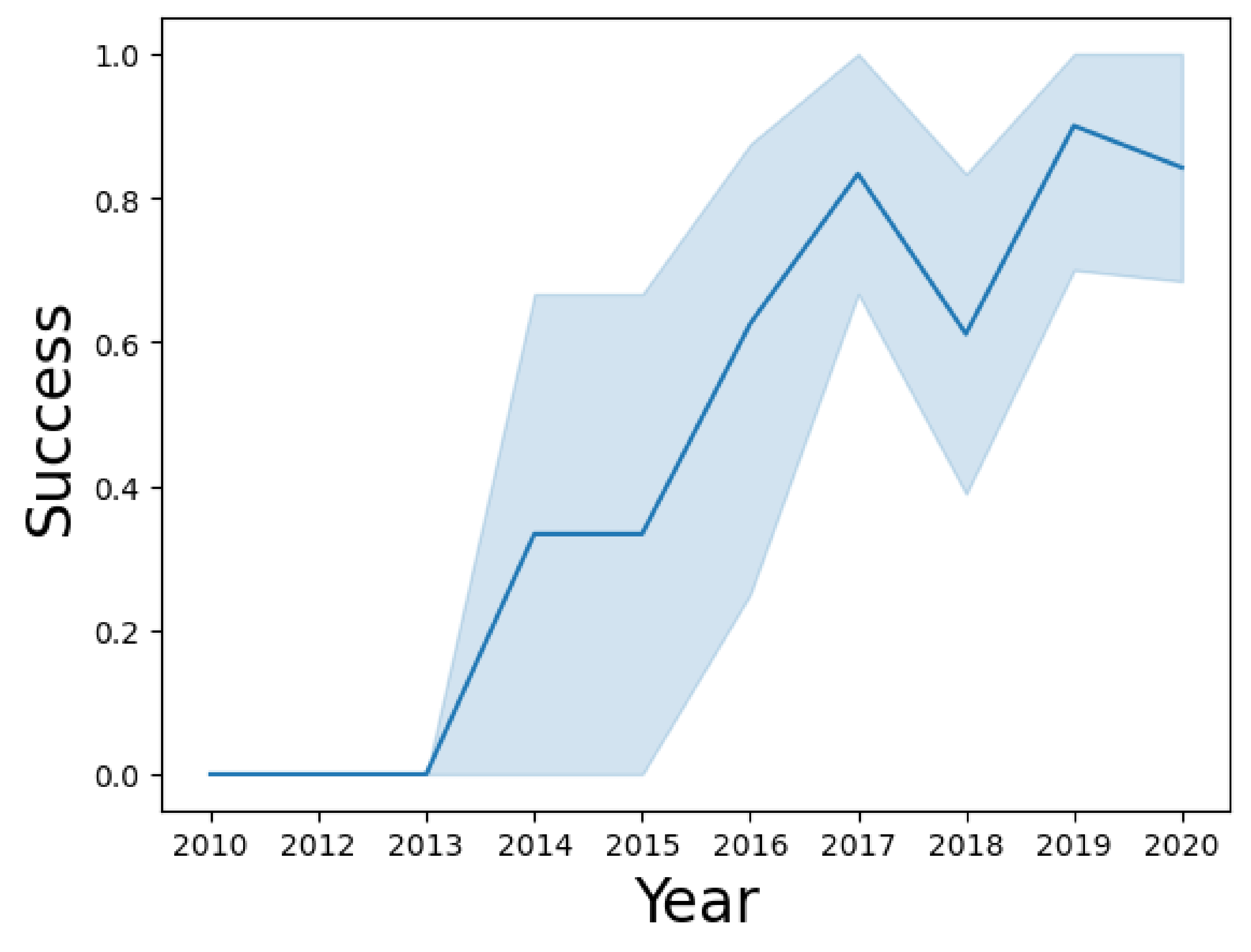
- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.



Launch Success Yearly Trend

- Show a line chart of yearly average success rate

- The success rate since 2013 kept increasing till 2020



All Launch Site Names

- Find the names of the unique launch sites
- SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;

```
8]: %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;

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

8]:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE "CCA%" LIMIT 5;

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

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

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE;

```
20]: %sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE;

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

20]:

SUM("PAYLOAD_MASS__KG_")
619967

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" LIKE "F9 v1.1%";

```
]: %sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" LIKE "F9 v1.1%";

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

]:

AVG("PAYLOAD_MASS__KG_")
2534.6666666666665

First Successful Ground Landing Date

- **Find the dates of the first successful landing outcome on ground pad**
- SELECT MIN(DATE) FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE "Success (ground pad)";

```
]: %sql SELECT MIN(DATE) FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE "Success (ground pad)";
 * sqlite:///my_data1.db
Done.
```

]:

MIN(DATE)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- **List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000**
- SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' and "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;

```
[32]: #%sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;
%sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' and "PA
#%sql SELECT DISTINCT("Landing_Outcome") FROM SPACEXTABLE;

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

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

Total Number of Successful and Failure Mission Outcomes

- **Calculate the total number of successful and failure mission outcomes**
- SELECT "Mission_Outcome", COUNT("Mission_outcome") FROM SPACEXTABLE GROUP BY "Mission_Outcome";

```
%sql SELECT "Mission_Outcome", COUNT("Mission_outcome") FROM SPACEXTABLE GROUP BY "Mission_Outcome";
```

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

Mission_Outcome	COUNT("Mission_outcome")
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- **List the names of the booster which have carried the maximum payload mass**
- SELECT "Booster_Version", "PAYLOAD_MASS__KG_" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE);

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- SELECT "Date", "Booster_Version", "Launch_Site", "Landing_Outcome" FROM SPACEXTABLE WHERE substr(Date, 0, 5) = '2015' AND substr(Date, 6, 2) IN ('01', '02', '03', '04', '05', '06', '07', '08', '09', '10', '11', '12');

Date	Booster_Version	Launch_Site	Landing_Outcome
2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
2015-02-11	F9 v1.1 B1013	CCAFS LC-40	Controlled (ocean)
2015-03-02	F9 v1.1 B1014	CCAFS LC-40	No attempt
2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)
2015-04-27	F9 v1.1 B1016	CCAFS LC-40	No attempt
2015-06-28	F9 v1.1 B1018	CCAFS LC-40	Precluded (drone ship)
2015-12-22	F9 FT B1019	CCAFS LC-40	Success (ground pad)

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

- Rank 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

- 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;

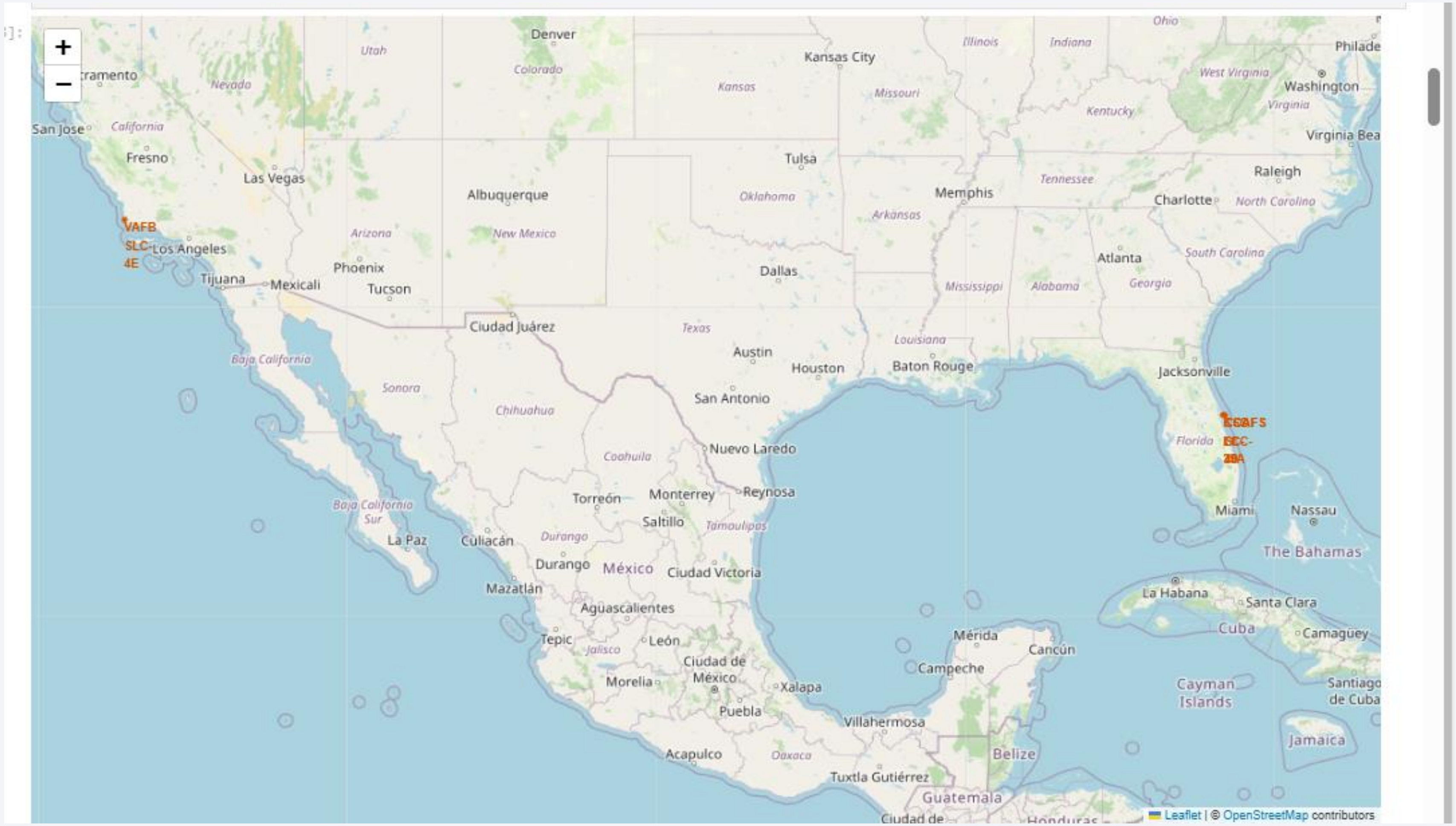
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

Map with launch sites location markers

- Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map
- Explain the important elements and findings on the screenshot



Color-labeled launch outcomes map

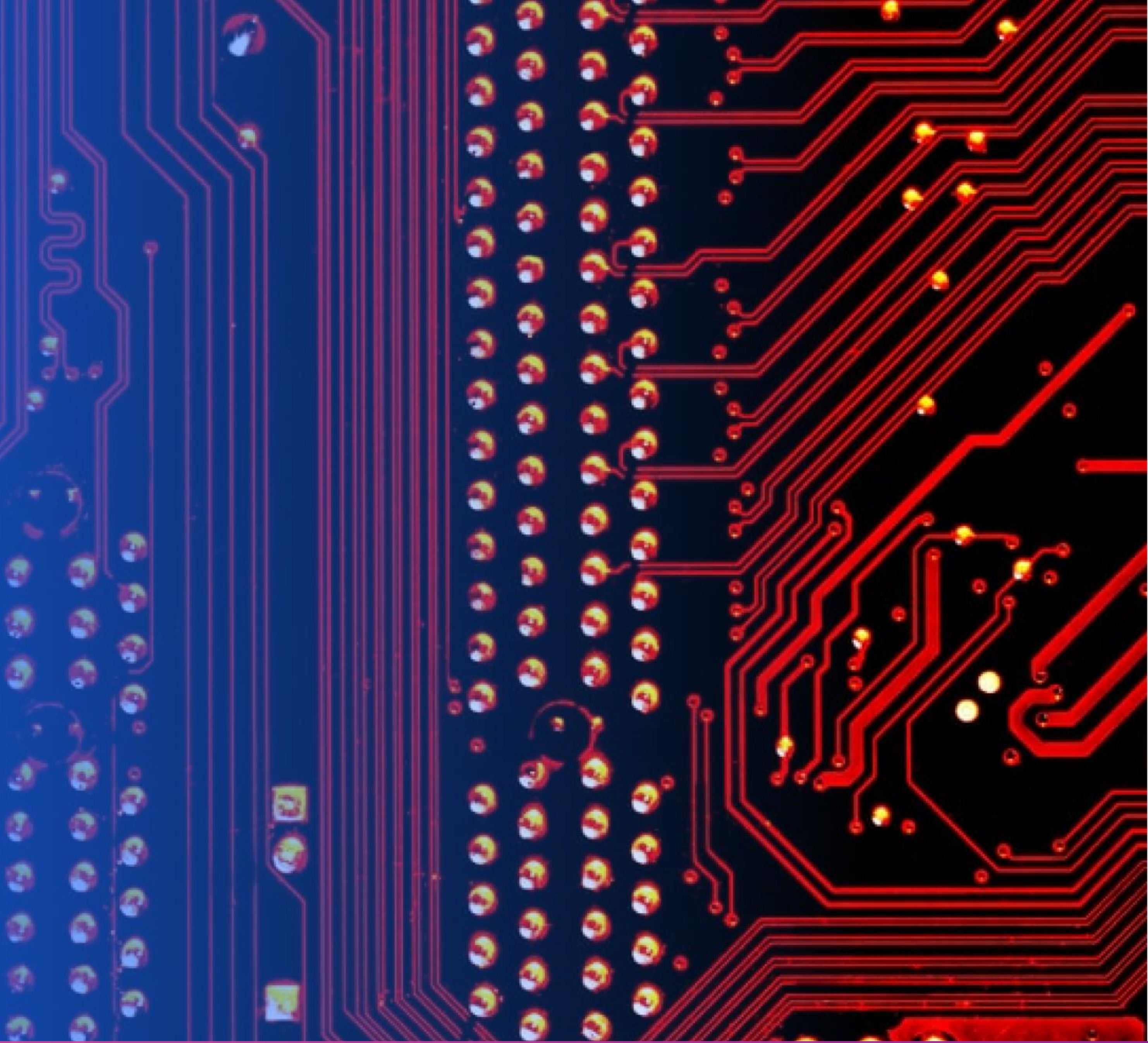
- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map

Map with proximity to railway, highway and coastline

- Explore the generated folium map of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed.

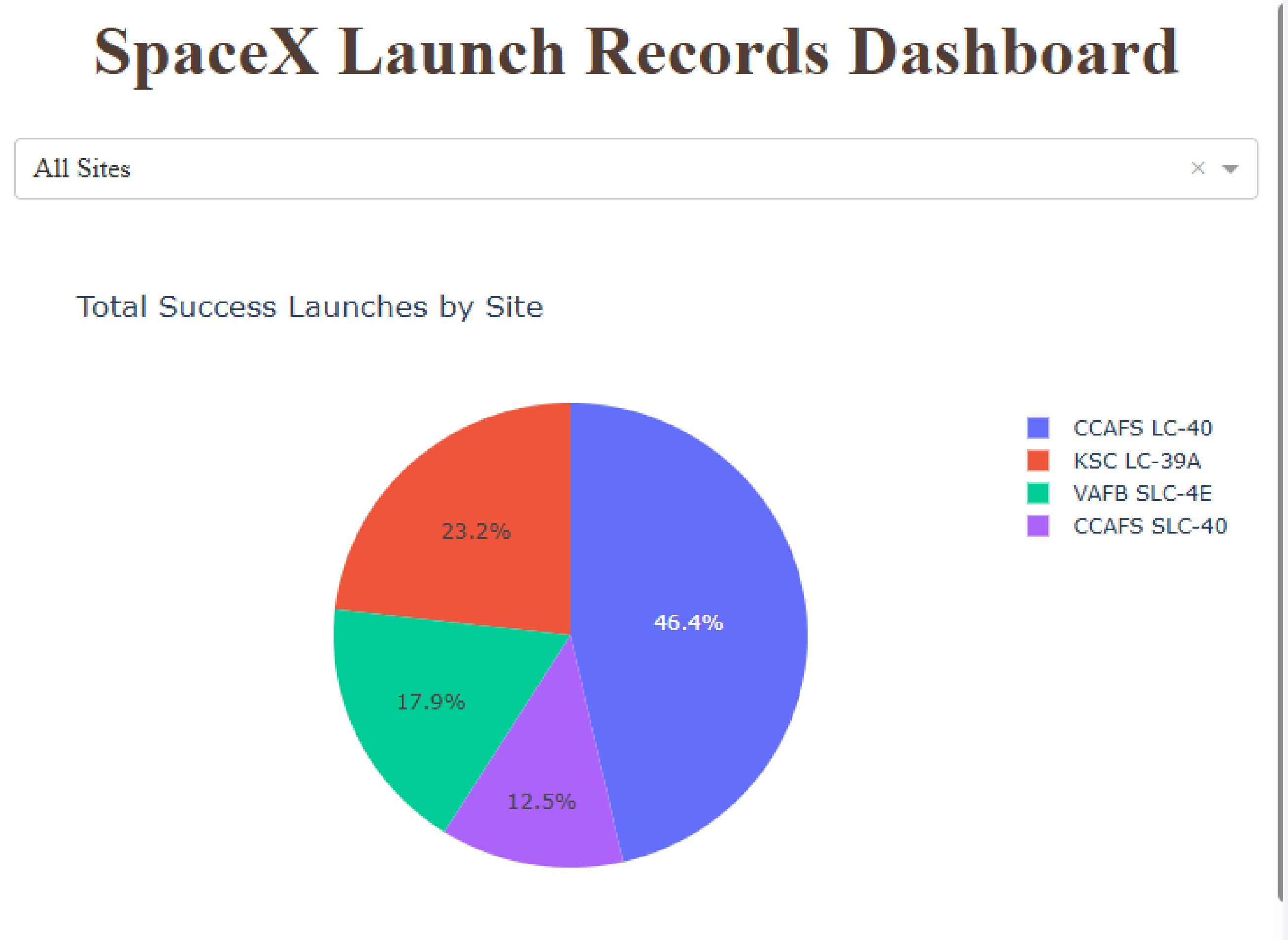
Section 4

Build a Dashboard with Plotly Dash



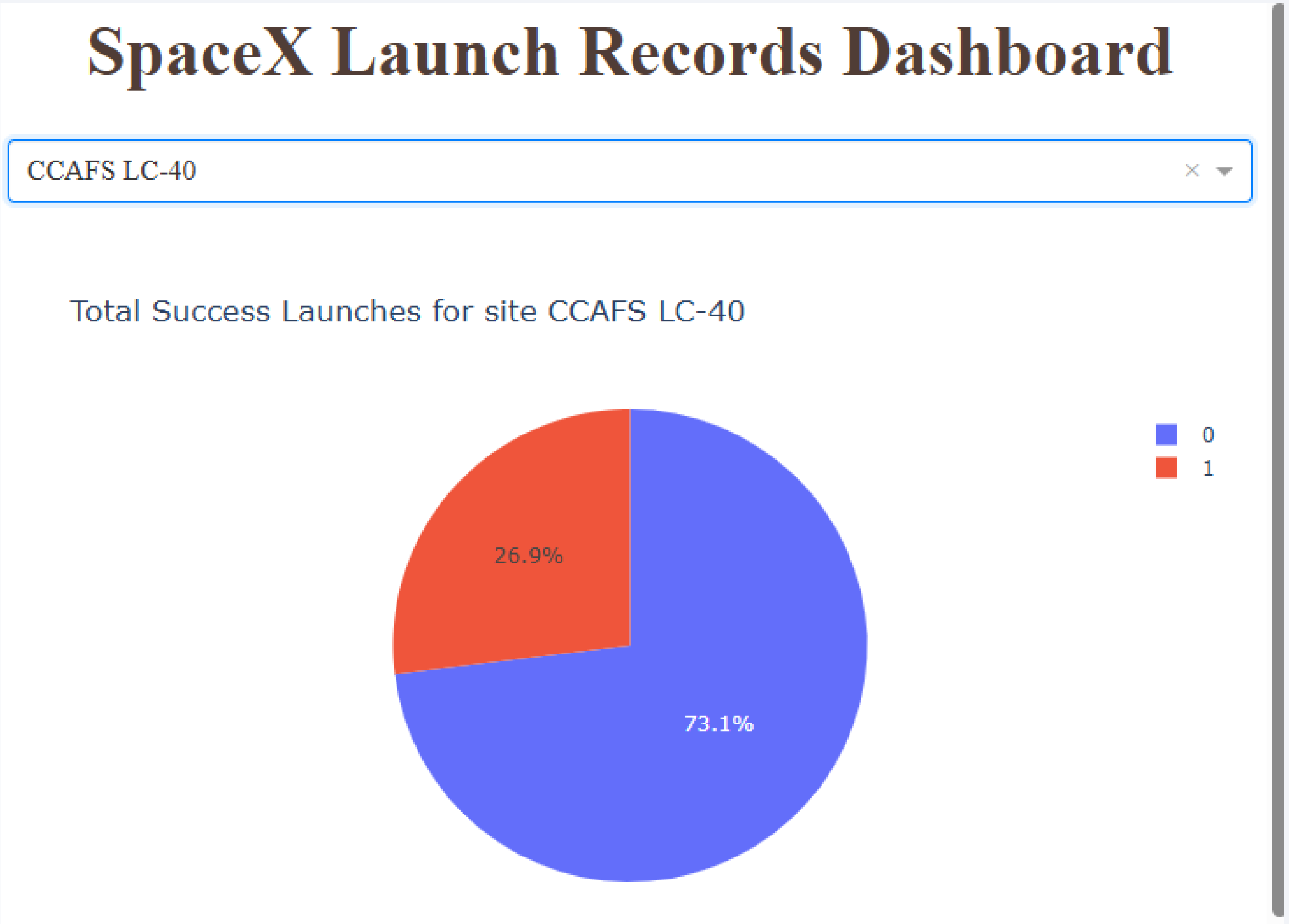
Piechart of Total Success Launches by Site

- CCAFS LC-40 had more than double (46.4%) succe ssful launches than other sites
- KSC LC-39A was next with 23.2%
- CCAFS SLC-40 was the lowest at 12.5%



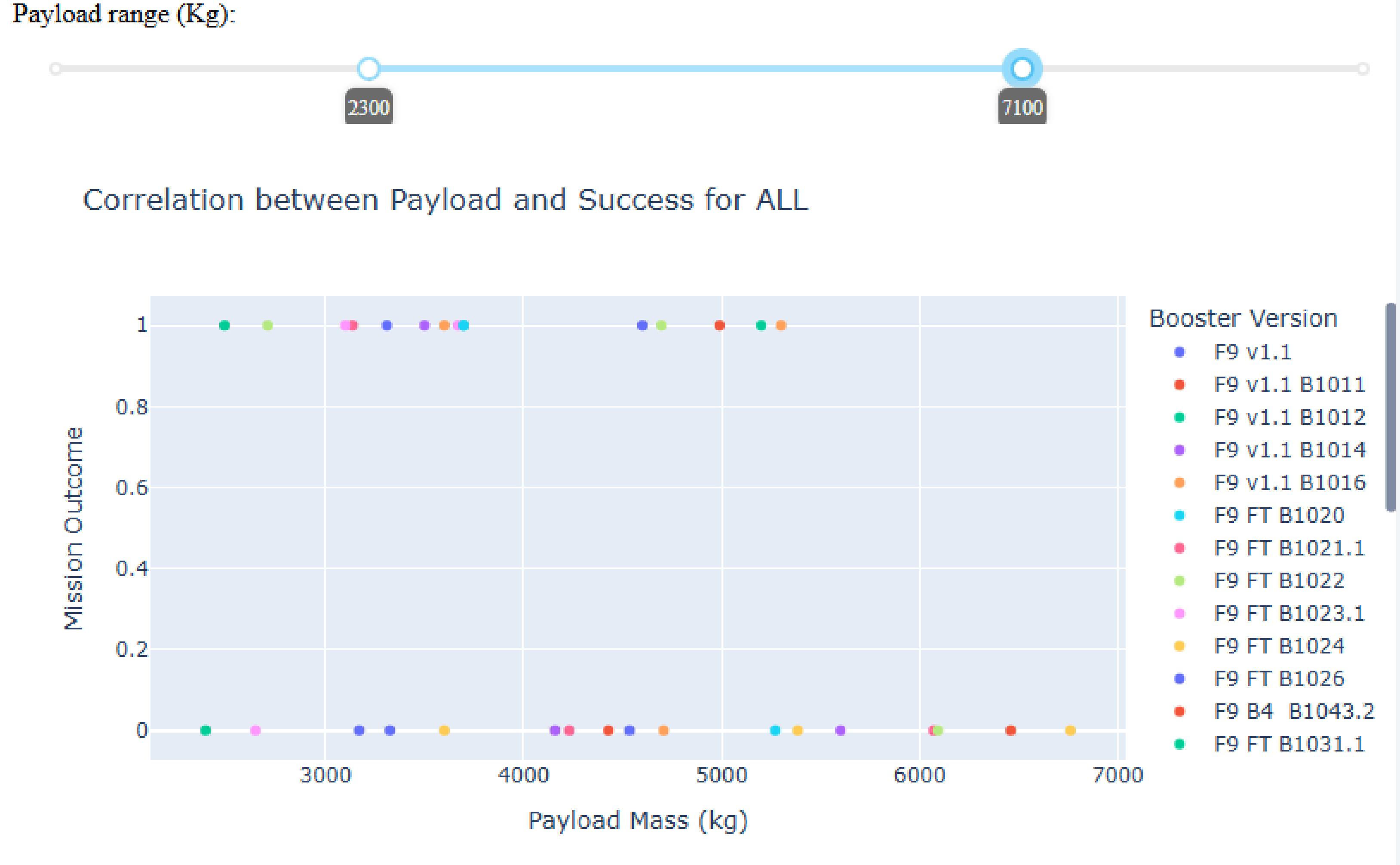
Piechart of Total Success Launches for CCAFS LC-40

- 19 Successful Launches
- 7 Failure Launches



Scatter plot for Payload vs Launch Outcome

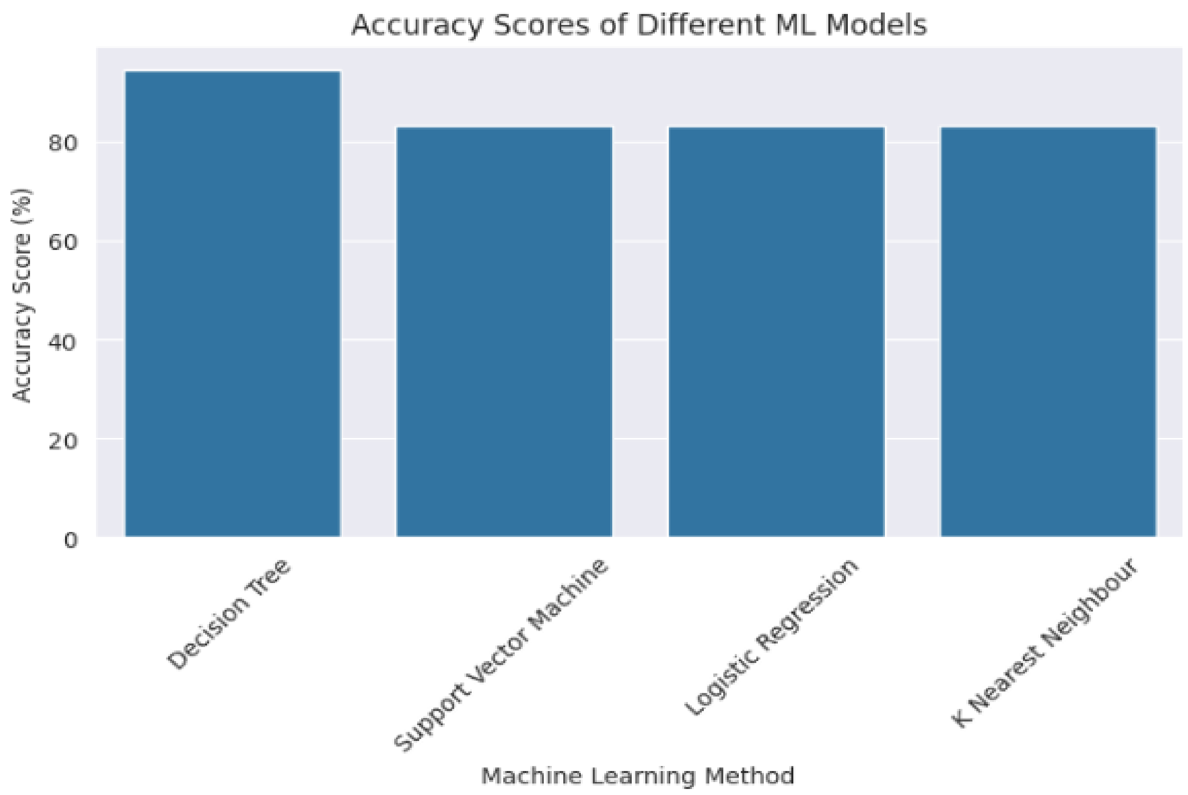
- Payload Mass: Across all launch sites, the higher the payload mass (kg), the higher the success rate



Section 5

Predictive Analysis (Classification)

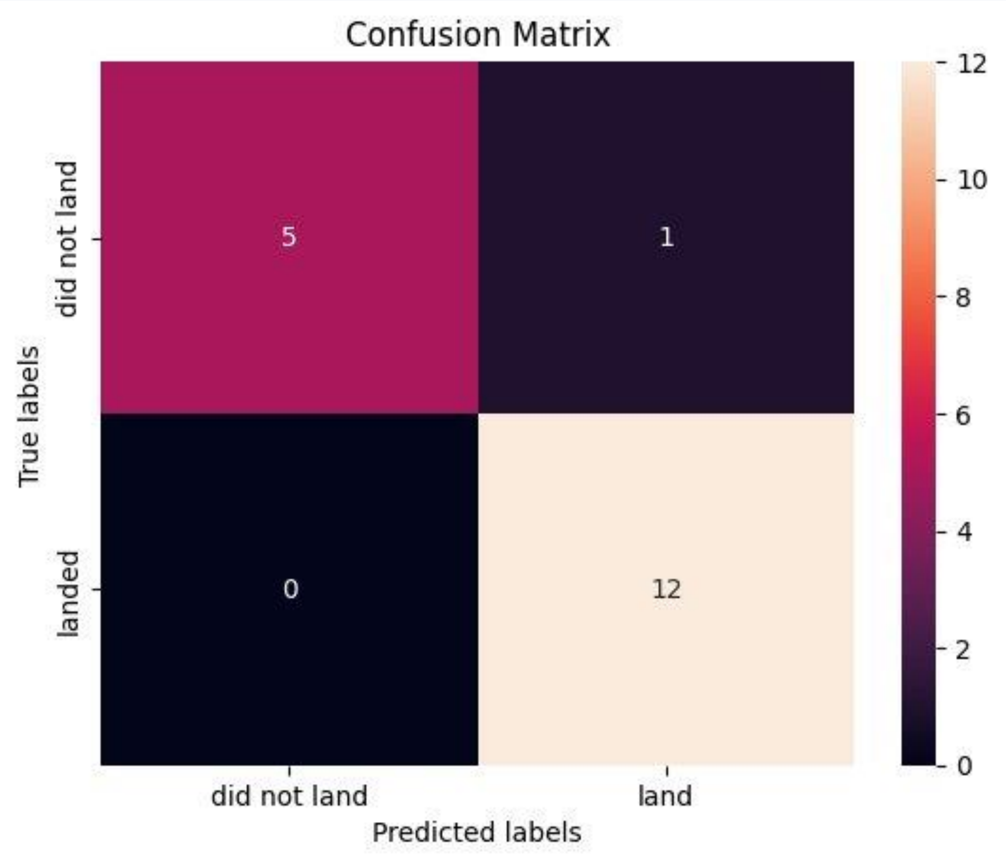
Classification Accuracy




```
                  Method   Accuracy
3          Decision Tree  94.444444
0  Support Vector Machine  83.333333
1     Logistic Regression  83.333333
2      K Nearest Neighbour  83.333333
Best params is : {'criterion': 'entropy', 'max_depth': 8, 'max_features': 'log2', 'min_samples_leaf': 1,
'min_samples_split': 5, 'splitter': 'random'}
```

Confusion Matrix of Decision Tree

- There are:
  - 12 True Positives
  - 0 True Negatives
  - 1 False Positive
  - 5 False Negatives
- Model performance has decent precision since it did miss some FN.



Conclusions

- Decision Tree model performed the best
- Launch Success has improved over time
- The higher the payload mass (kg), the higher the success rate

- We are able to predict with a fair amount of accuracy the odds of a successful or failure landing.

Appendix

- None Included (Any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project)

Thank you!