IBM Developer
SKILLS NETWORK

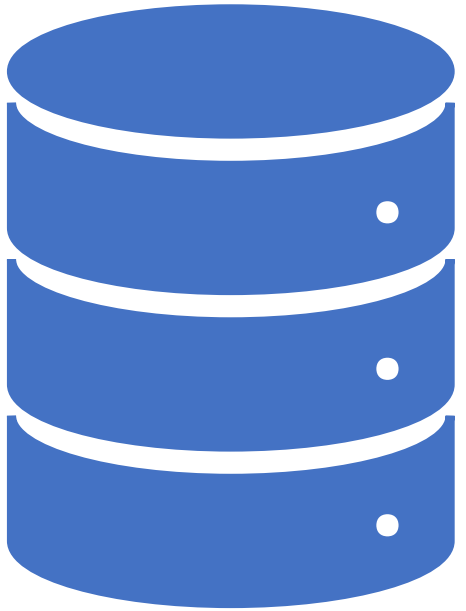
Winning Space Race with Data Science

RAGHAV AGARWAL
19/08/23

OUTLINE

EXECUTIVE SUMMARY

- We first performed data collection using both APIs and web scraping. We then performed data wrangling and cleaning of the data. Then we performed exploratory data analysis (EDA) using visualization and SQL. Next, we created interactive visual analytics using Folium and Plotly Dash. Finally, we performed predictive analysis using classification models.
- Through these techniques, we were able to draw many conclusions by different comparisons that we present later in this report. We also compared the accuracy and performance of the different classification models.



INTRODUCTION

- In this Applied Data Science Capstone course, I applied my data science skills as a Data scientist for a private space launch company.
- As a starting point of almost all data science projects, I needed to collect data, as much and as relevant as possible.
- I collected data from various sources. After my raw data had been collected, I improved the quality by performing data wrangling.
- Then I started exploring the processed data and explored some interesting real-world datasets. I used my SQL skills to query the data and gather insights.
- I gained further insights into the data by applying some basic statistical analysis and data visualization and observed directly how variables might be related to each other.
- I also split the data into groups defined by categorical variables or factors in my data.
- I built, evaluated and refined predictive models for discovering more exciting insights.
- Finally, I combined all my efforts into this report and displayed all my findings.

Section 1

Methodology

METHODOLOGY

Data collection methodology:

- Data was collected using APIs and Web Scraping.

Perform data wrangling

- We removed null values and created a new column to determine success of landing by the outcome.

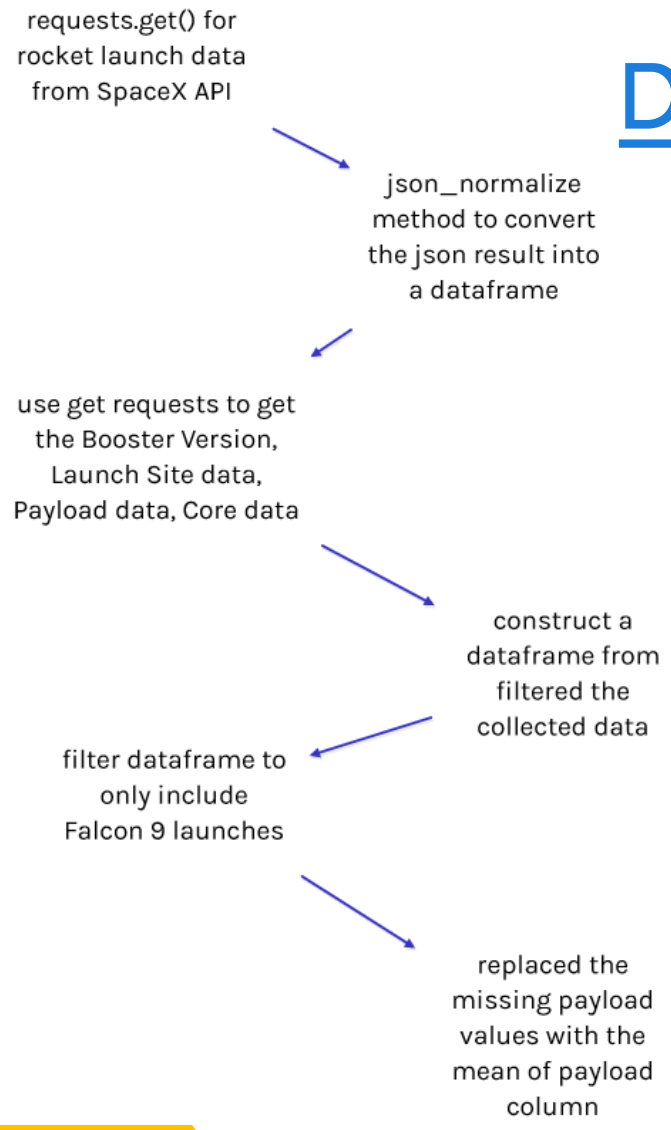
Perform exploratory data analysis (EDA) using visualization and SQL

Perform interactive visual analytics using Folium and Plotly Dash

Perform predictive analysis using classification models

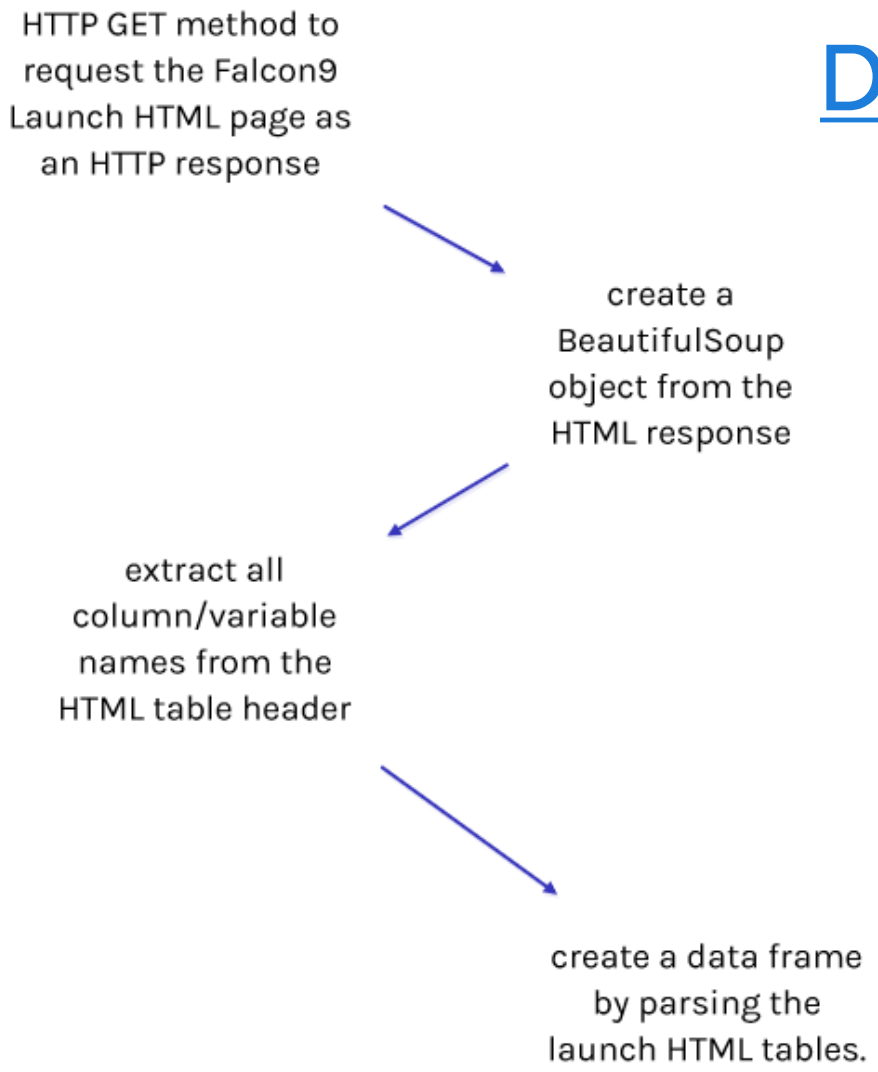
- We built Logistic Regression, SVM, Decision Tree and k-Nearest Neighbours models.

DATA COLLECTION – SPACEX API



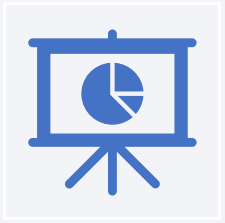
- We used requests.get() for rocket launch data from SpaceX API. We used json_normalize method to convert the json result into a dataframe. We then used get requests to get the booster version, Launch Site data, Payload data, Core data. Then we constructed a dataframe from the collected data which we further filtered to only include Falcon 9 launches. Finally, we replaced the missing payload values with the mean of payload column.
- Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/1.%20jupyter-labs-spacex-data-collection-api.ipynb

DATA COLLECTION - SCRAPING



- We used HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response. Then we created a BeautifulSoup object from the HTML response. Then we extracted all column/variable names from the HTML table header. Finally, we created a data frame by parsing the launch HTML tables.
- Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/2.%20jupyter-labs-webscraping.ipynb

DATA WRANGLING



First, we calculated the percentage of missing values in each attribute. Then we used the method value_counts() on the column LaunchSite to determine the number of launches on each site. We then did the same for columns Orbit and Outcome. Next, we created a set of outcomes for which the second stage did not land successfully. This included the outcomes 'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS' and 'None None'. Finally, we created a landing outcome label from Outcome column. If the value is zero, it did not land successfully; one means it landed successfully.

Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/3.%20spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA WITH DATA VISUALIZATION

- First, we plotted out the FlightNumber vs. PayloadMass and overlayed the outcome of the launch.
- Next, we visualized the relationship between Flight Number and Launch Site as a catplot.
- Then we visualized the relationship between Payload and Launch Site as a catplot.
- Next, we visualized the relationship between success rate of each orbit type as a bar chart.
- Then we visualized the relationship between FlightNumber and Orbit type as a scatter plot
- Then we visualized the relationship between Payload and Orbit type as a scatter plot.
- Finally, we plotted the launch success yearly trend as a line chart.
- Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/5.%20eda-dataviz.ipynb.jupyterlite.ipynb

EDA WITH SQL

- First, we created a table SPACEXTABLE using CREATE TABLE
- Displayed the names of the unique launch sites in the space mission using SELECT DISTINCT
- Displayed 5 records where launch sites begin with the string 'CCA’
- Displayed the total payload mass carried by boosters launched by NASA (CRS) using SUM()
- Displayed average payload mass carried by booster version F9 v1.1 using AVG()
- Listed the date when the first successful landing outcome in ground pad was achieved
- Listed the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Listed the total number of successful and failure mission outcomes
- Listed the names of the booster_versions which have carried the maximum payload mass using a subquery
- Listed the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
- Ranked 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
- Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/4.%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

BUILD AN INTERACTIVE MAP WITH FOLIUM

- First, we created a folium Map object, with an initial center location to be NASA Johnson Space Center at Houston, Texas.
- We used folium.Circle to add a highlighted circle area with a text label on each launch site in data frame launch_sites and we also added folium markers.
- Next, we created markers for all launch records. If a launch was successful (class=1), then we used a green marker and if a launch was failed, we used a red marker (class=0). Since a launch only happens in one of the four launch sites, many launch records will have the exact same coordinate. So, we used marker clusters to simplify the map.
- We calculated the distances between a launch site to its proximities by adding a MousePosition on the map to get coordinate for a mouse over a point on the map.
- We also drew a PolyLine between a launch site to a selected coastline point.
- Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/6.%20folium_launch_site_location.jupyterlite.ipynb

BUILD A DASHBOARD WITH PLOTLY DASH

- We added a launch site drop-down input component.
- We added a callback function to render success-pie-chart based on selected site dropdown.
- We added a range slider to select payload.
- We added a callback function to render the success-payload-scatter-chart scatter plot.
- Github link for relevant Python code: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/7.%20spacex_dash_app.py

PREDICTIVE ANALYSIS (CLASSIFICATION)

- We created a NumPy array from the column Class in data, by applying the method to_numpy() then assign it to the variable Y. We also standardized the data in X.
- We used the function train_test_split() to split the data X and Y into training and test data.
- We created a Logistic Regression object then created a GridSearchCV object logreg_cv with cv = 10. Then we fit the object to find the best parameters from the dictionary parameters. We then outputted the best parameters and accuracy and calculated the accuracy on the test data using the method score(). We also plotted the confusion matrix.
- We then did the same for Support Vector Machine, Decision Tree Classifier and k-Nearest Neighbor objects.
- We then found which method performs the best by comparing the accuracy on the test data using the method score().
- Github link for relevant notebook: https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/8.%20SpaceX_Machine_Learning_Prediction.jupyterlite.ipynb

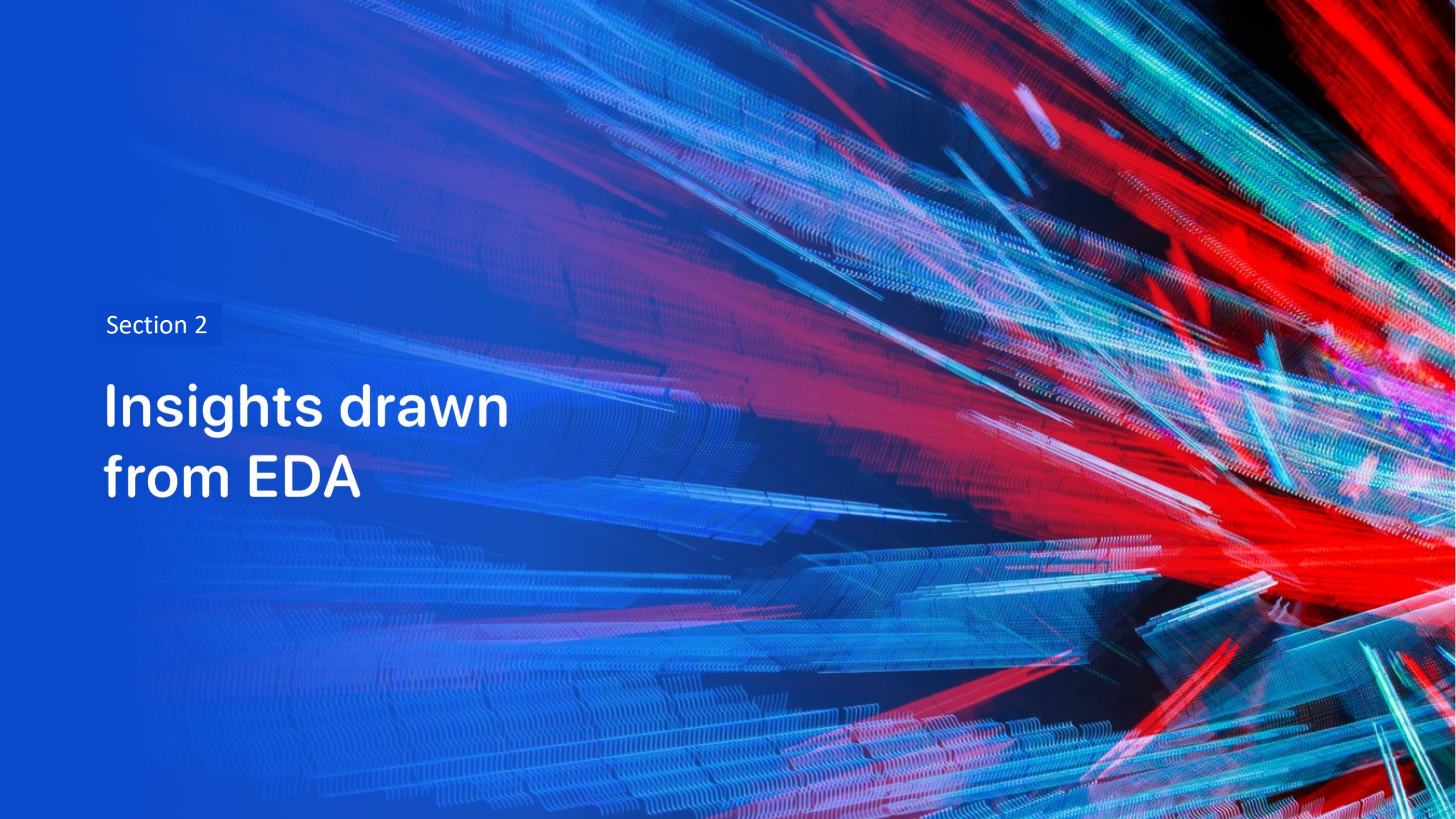
RESULTS

Exploratory Data Analysis Results:

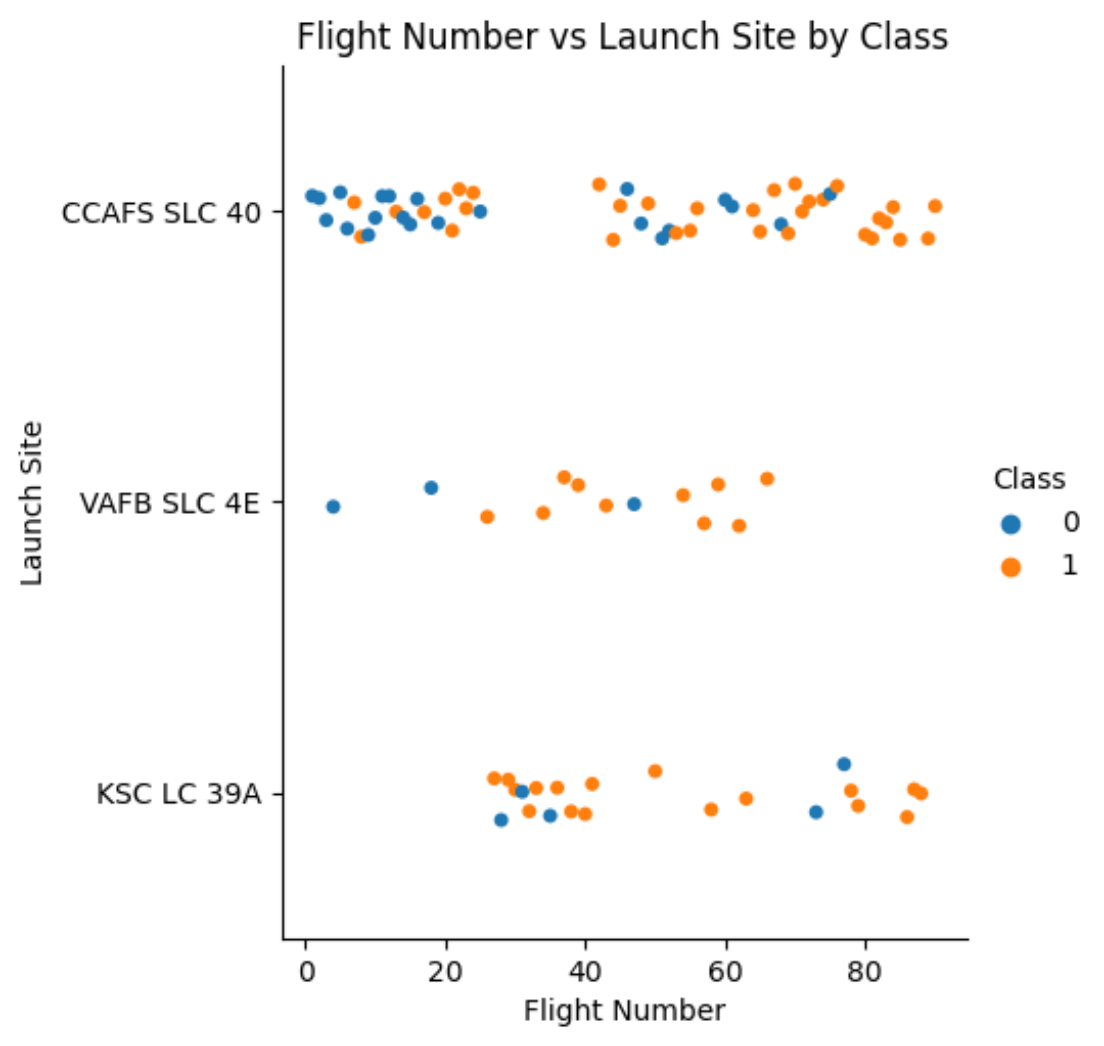
- We saw that as the flight number increases, the first stage is more likely to land successfully. The payload mass is also important; the more massive the payload, the less likely the first stage will return. We saw that different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.
- We observed the Payload Vs. Launch Site scatter point chart to find that for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).
- The bar chart shows that the ES-L1, GEO, HEO and SSO orbits have the highest success rate of 100% while the GTO orbit has the lowest success rate.
- We see that 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.
- We observe that with heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However, for GTO we cannot distinguish this well as both positive landing rate and negative landing are both present.
- Finally, we observe that the success rate since 2013 kept increasing till 2020.

Predictive analysis results:

- We observed that all models gave the same accuracy of 83.33% on the test data except decision tree which gave 66.67%.
- However, Decision Tree classifier had the best score of 87.31% after fitting with training data.

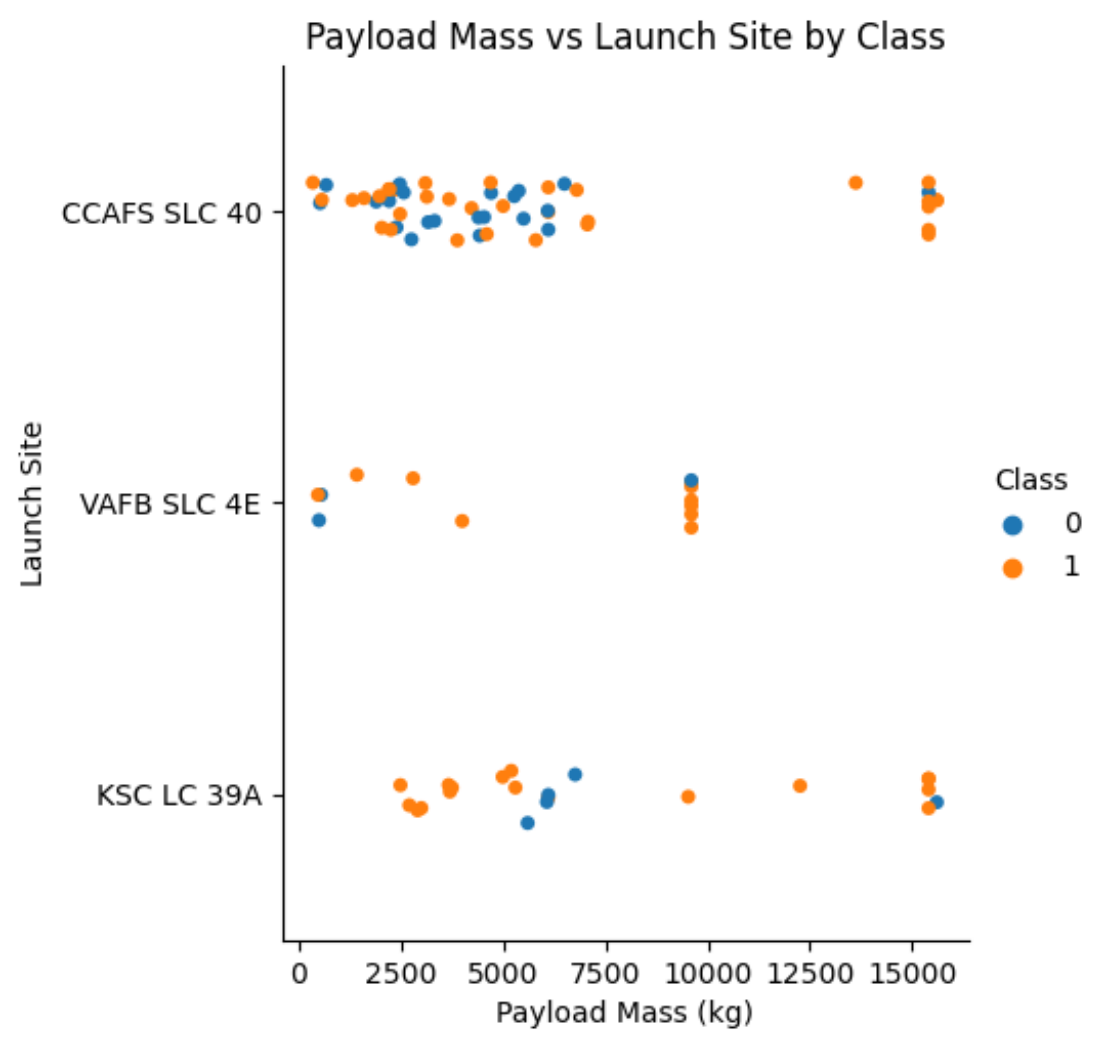

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



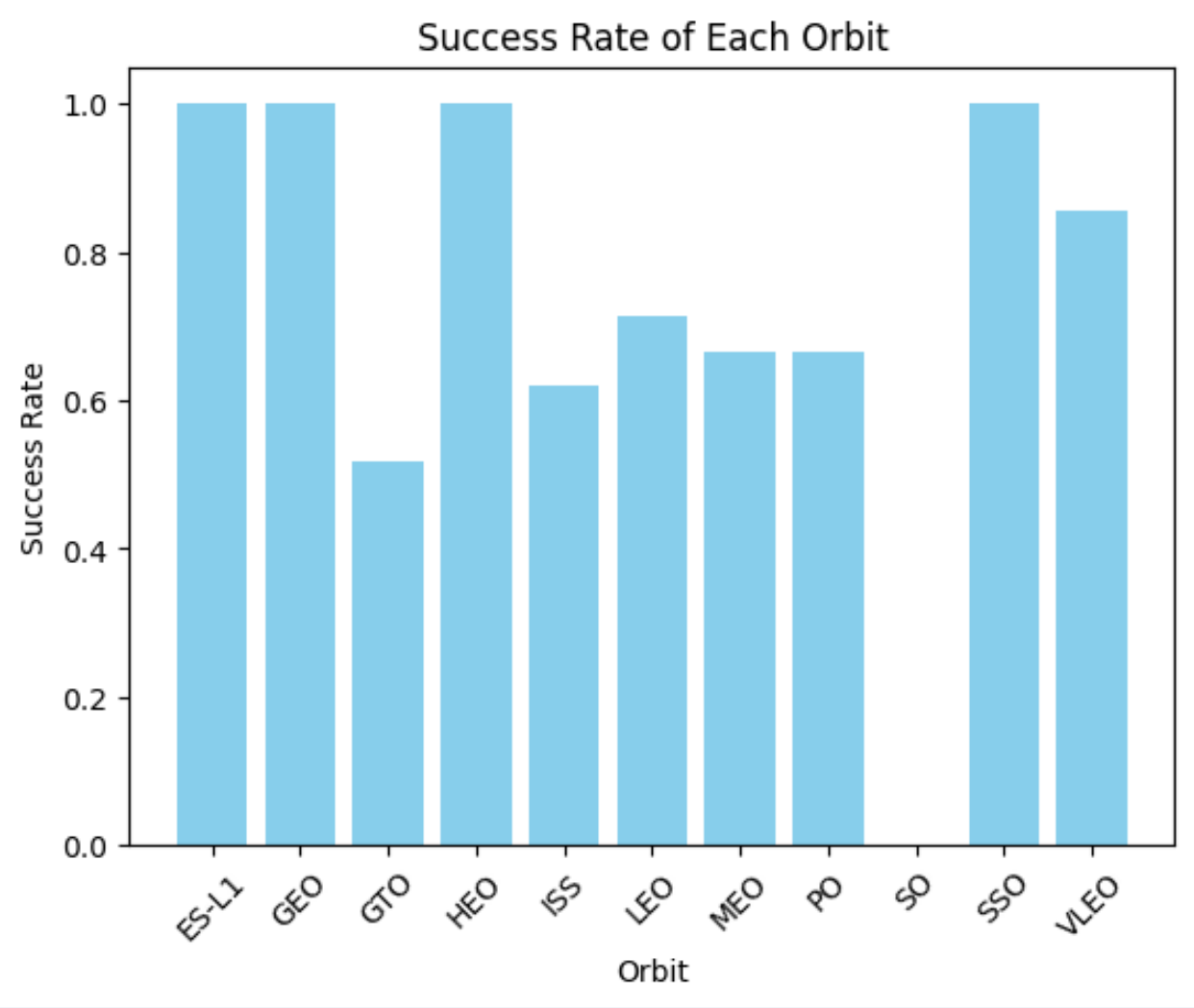
- As the flight number increases, the success rate increases for all launch sites.

Payload vs. Launch Site



- For the VAFB-SLC launch site there are no rockets launched for heavy payload mass (greater than 10000).

Success Rate vs. Orbit Type

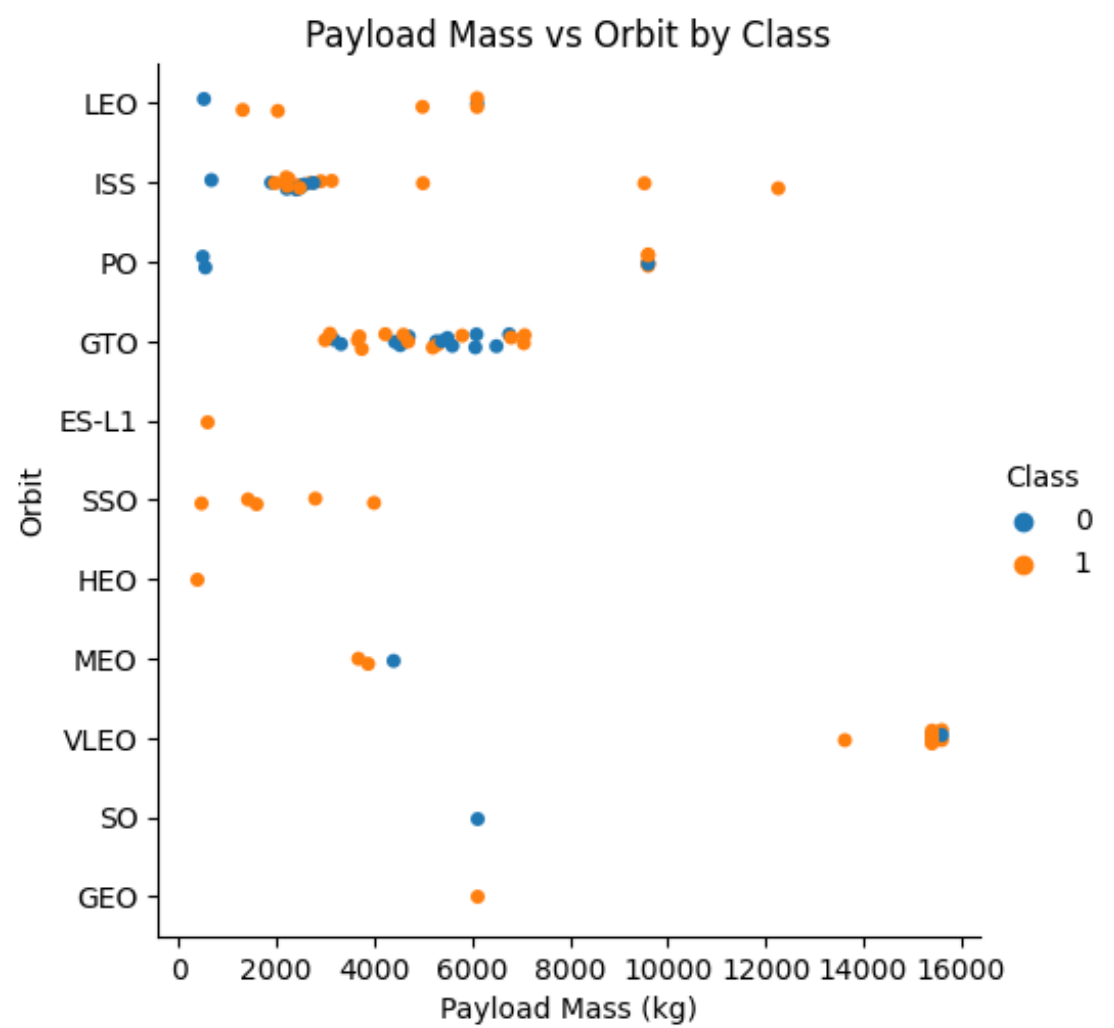


- The ES-L1, GEO, HEO and SSO orbits have the highest success rate of 100% while the GTO orbit has the lowest success rate.

Flight Number vs. Orbit Type

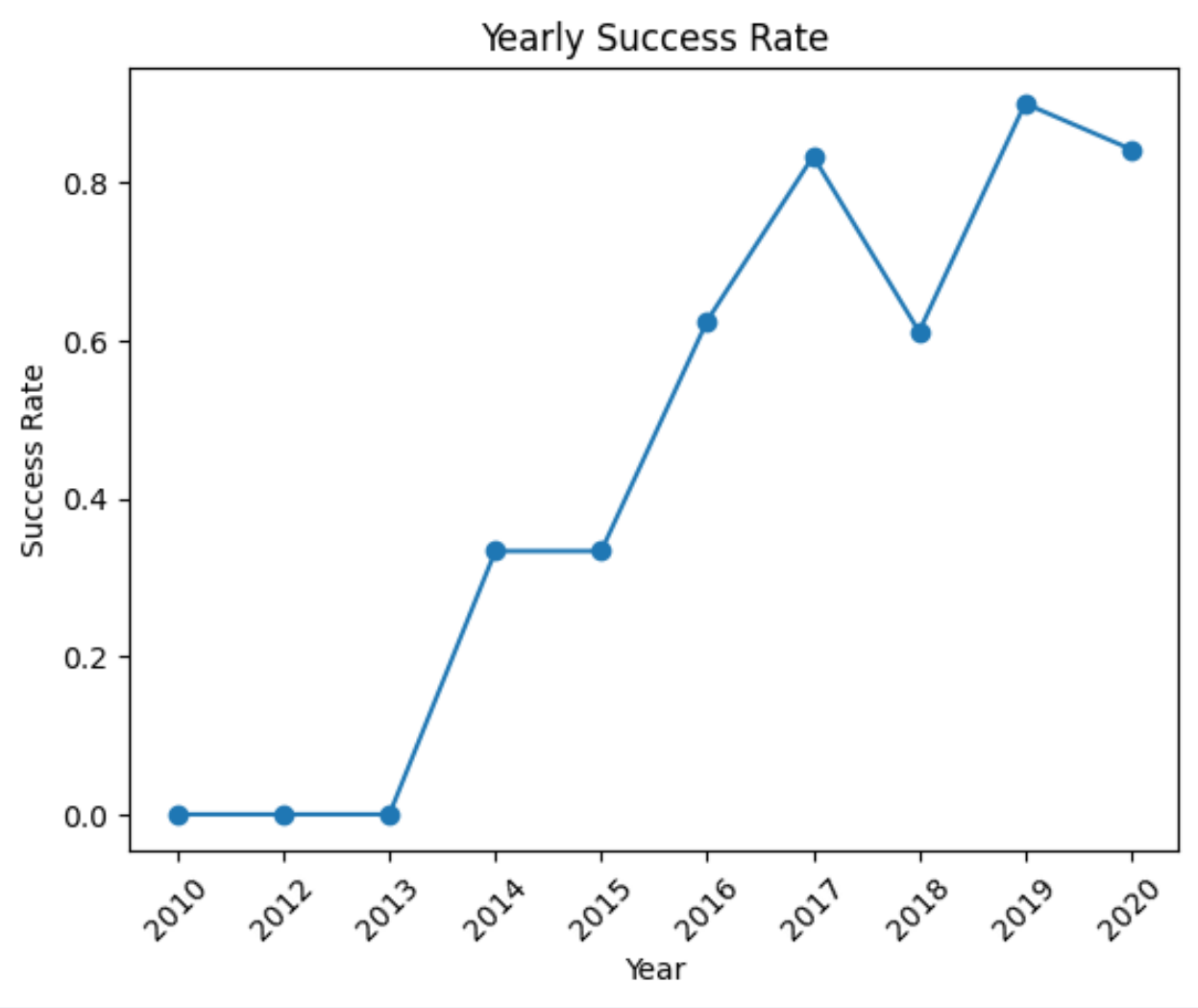
- In the LEO orbit the success appears related to the number of flights. On the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However, for GTO we cannot distinguish this well as both positive landing rate and negative landing are both present.

Launch Success Yearly Trend



- The success rate since 2013 kept increasing till 2020 where it took a slight dip.

All Launch Site Names

```
%%sql
SELECT DISTINCT Launch_Site
FROM SPACEXTABLE;
```

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

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

Launch Site Names Begin with 'CCA'

Display 5 records where launch sites begin with the string 'CCA'

```
%%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
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Customer = 'NASA (CRS)';
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

Display average payload mass carried by booster version F9 v1.1

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

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

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

```
%%sql
SELECT MIN(Date)
FROM SPACEXTABLE
WHERE Landing_Outcome = '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 the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

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

```
 * 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

List the total number of successful and failure mission outcomes

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

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

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

Boosters That Have Carried Maximum Payload

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

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

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

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

```
%%sql
SELECT SUBSTR(Date, 4, 2) AS Month,Landing_Outcome,Booster_Version,Launch_Site
FROM SPACEXTABLE
WHERE SUBSTR(Date, 7, 4) = '2015' AND Landing_Outcome = 'Failure (drone ship)';
```

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

Month	Landing_Outcome	Booster_Version	Launch_Site

- It appears that there was no landing outcome that failed by drone ship in the year 2015.

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.

```
%%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;
```

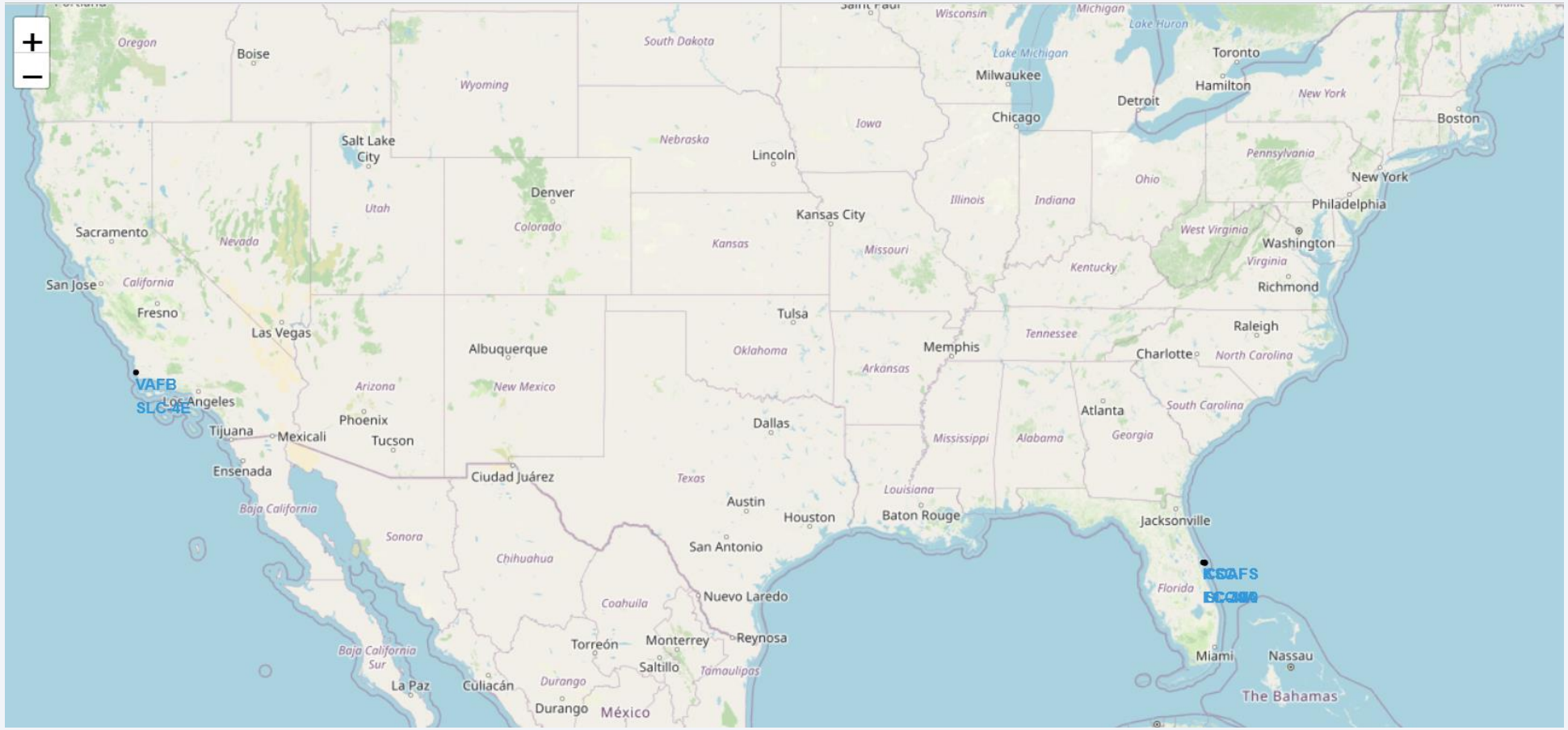
```
 * sqlite:///my_data1.db
Done.
```

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

Section 3

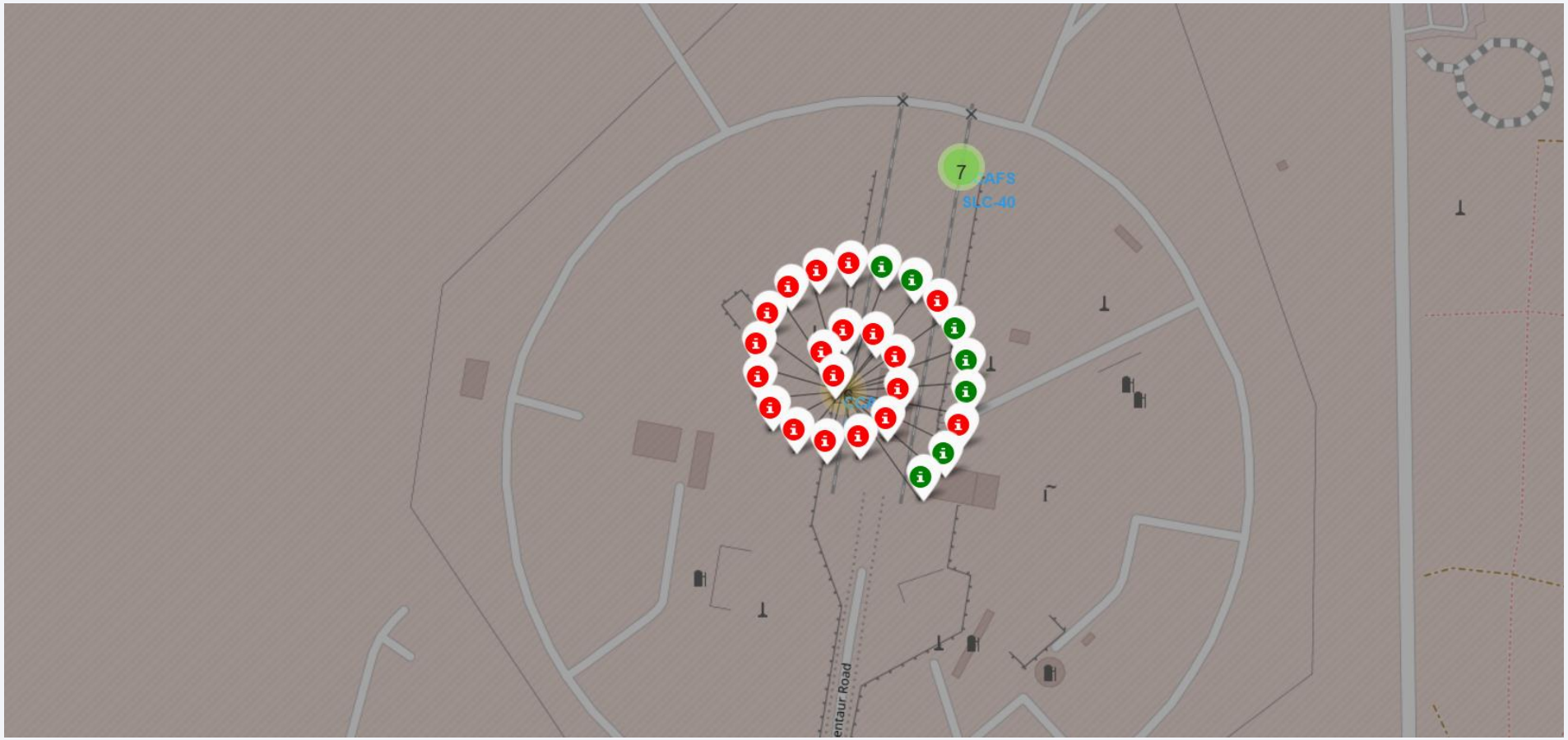
Launch Sites Proximities Analysis

Folium Map with all Marked Launch Sites



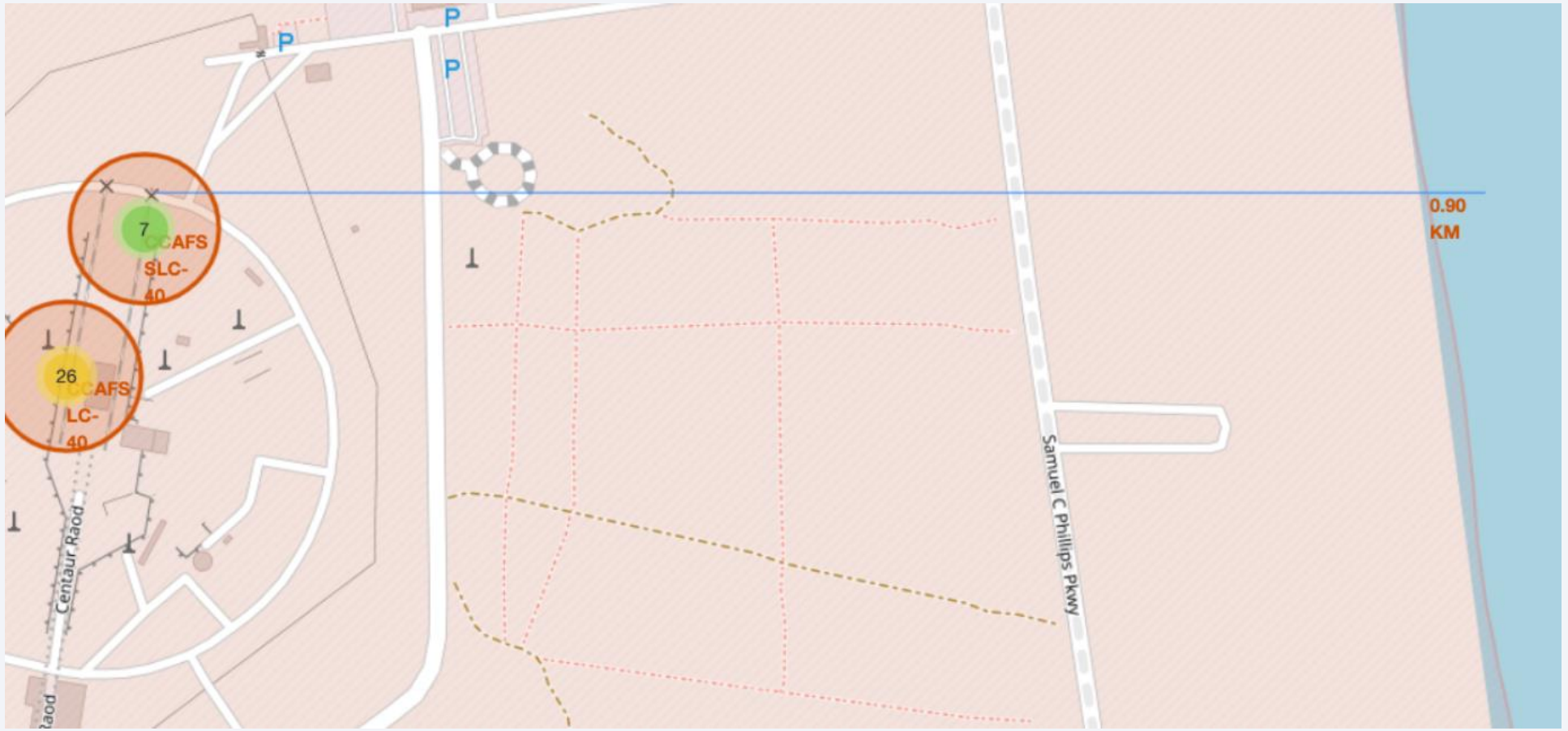
- We can see that all launch sites are in very close proximity to the coast.

Folium Map with Colour-Labeled Launch Outcomes

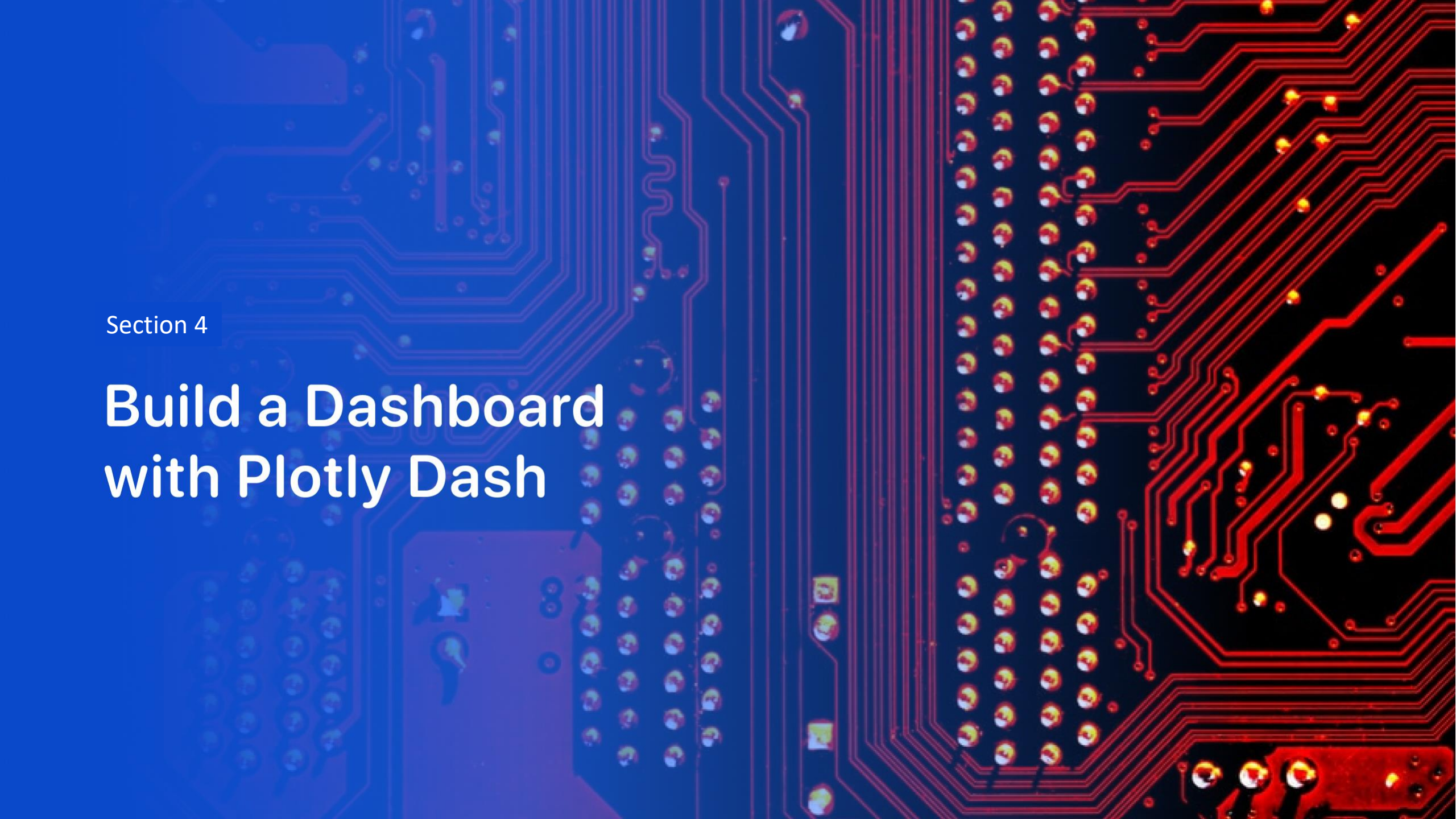


- We can use this map to identify which launch sites have high success rates.

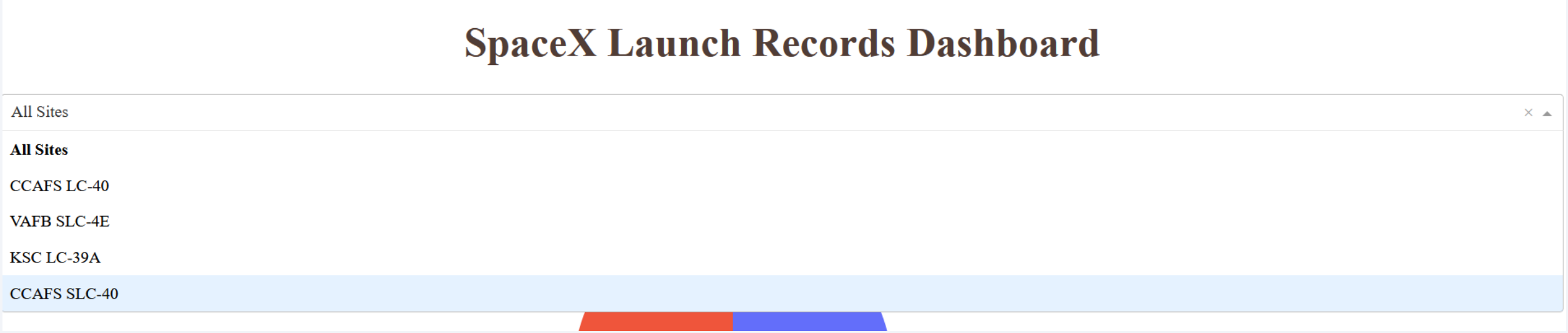
Folium Map to show Proximity to Coastlines



- We can use this map to tell the proximity of launch sites from coastlines, cities, railways etc.

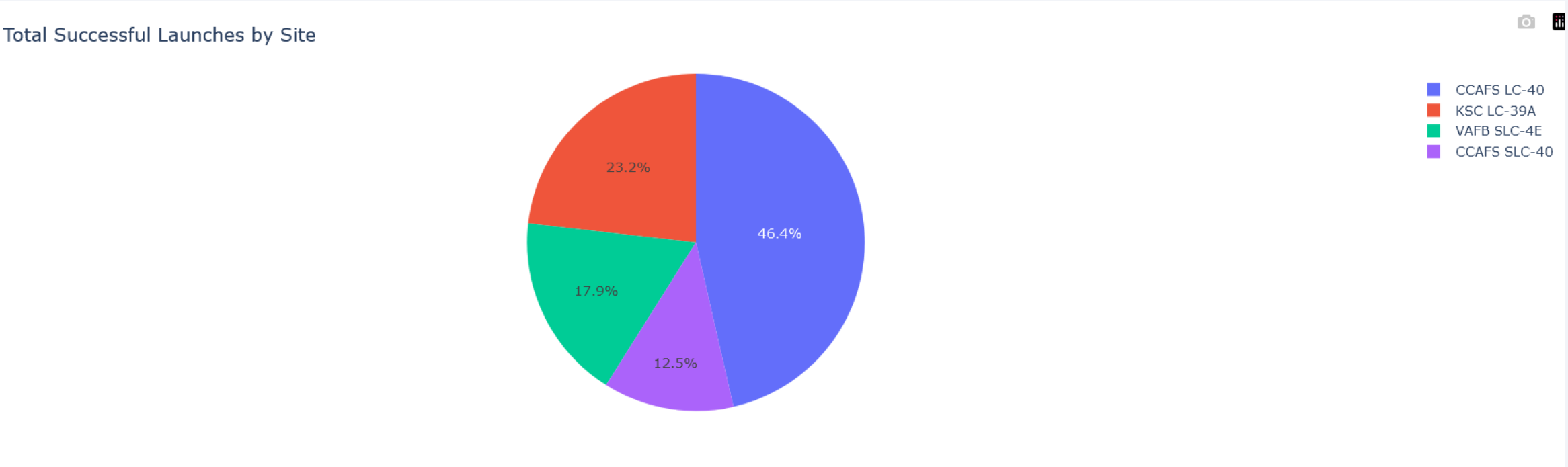

Section 4
Build a Dashboard with Plotly Dash

Dashboard Dropdown List



- The dropdown list lets us view the data for all sites or for a specific site.

Dashboard Launch Success Ratio



- The pie chart shows us the total successful launches by site.
- We can see that CCAFS LC-40 has the highest success rate while CCAFS SLC-40 has the lowest.

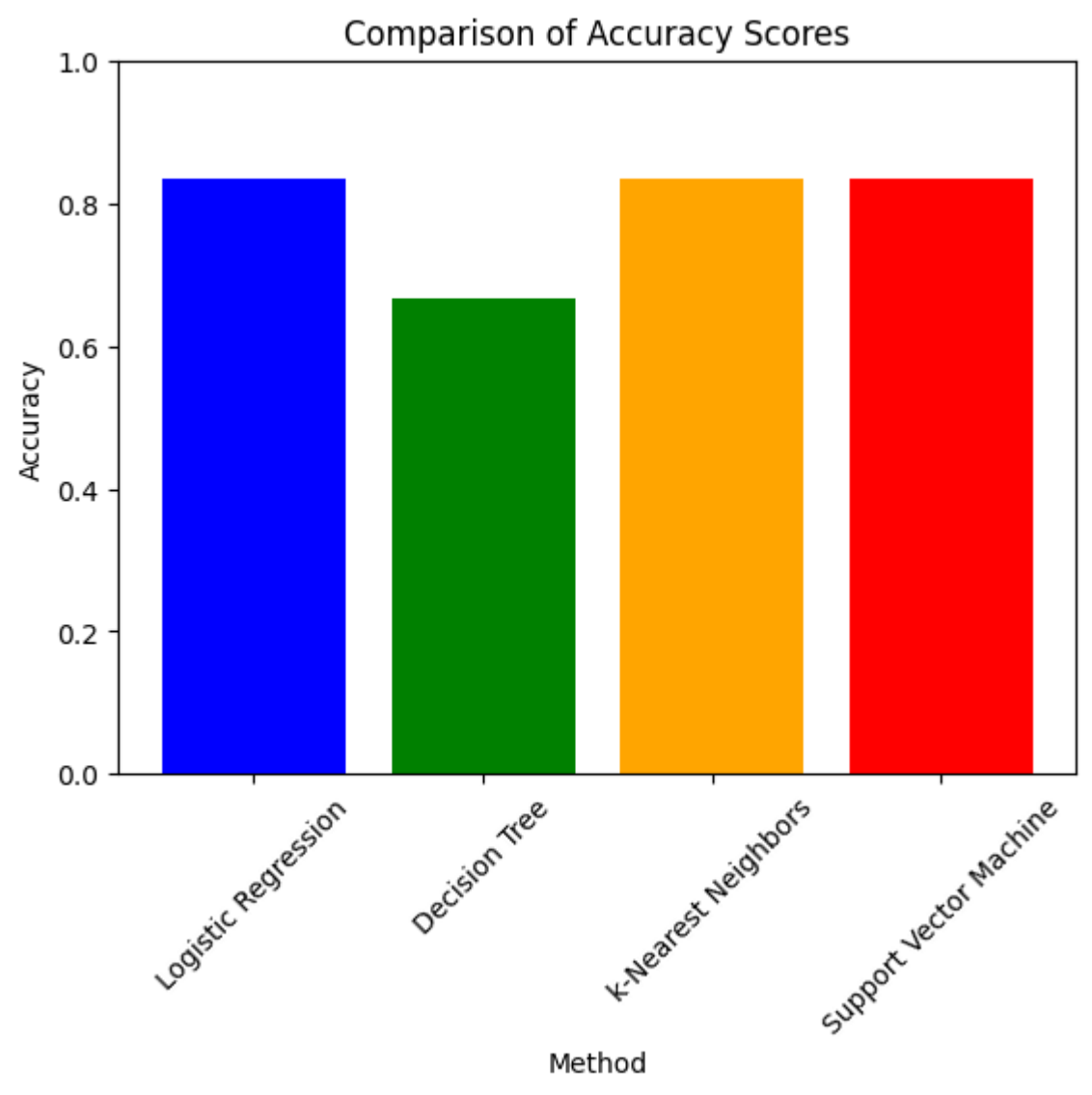
Dashboard Scatter Plot with Range Slider

- We can use the scatter plot to tell which payload mass has better launch outcome.
- We can use the range slider to change the range of payload displayed in the plot.

Section 5

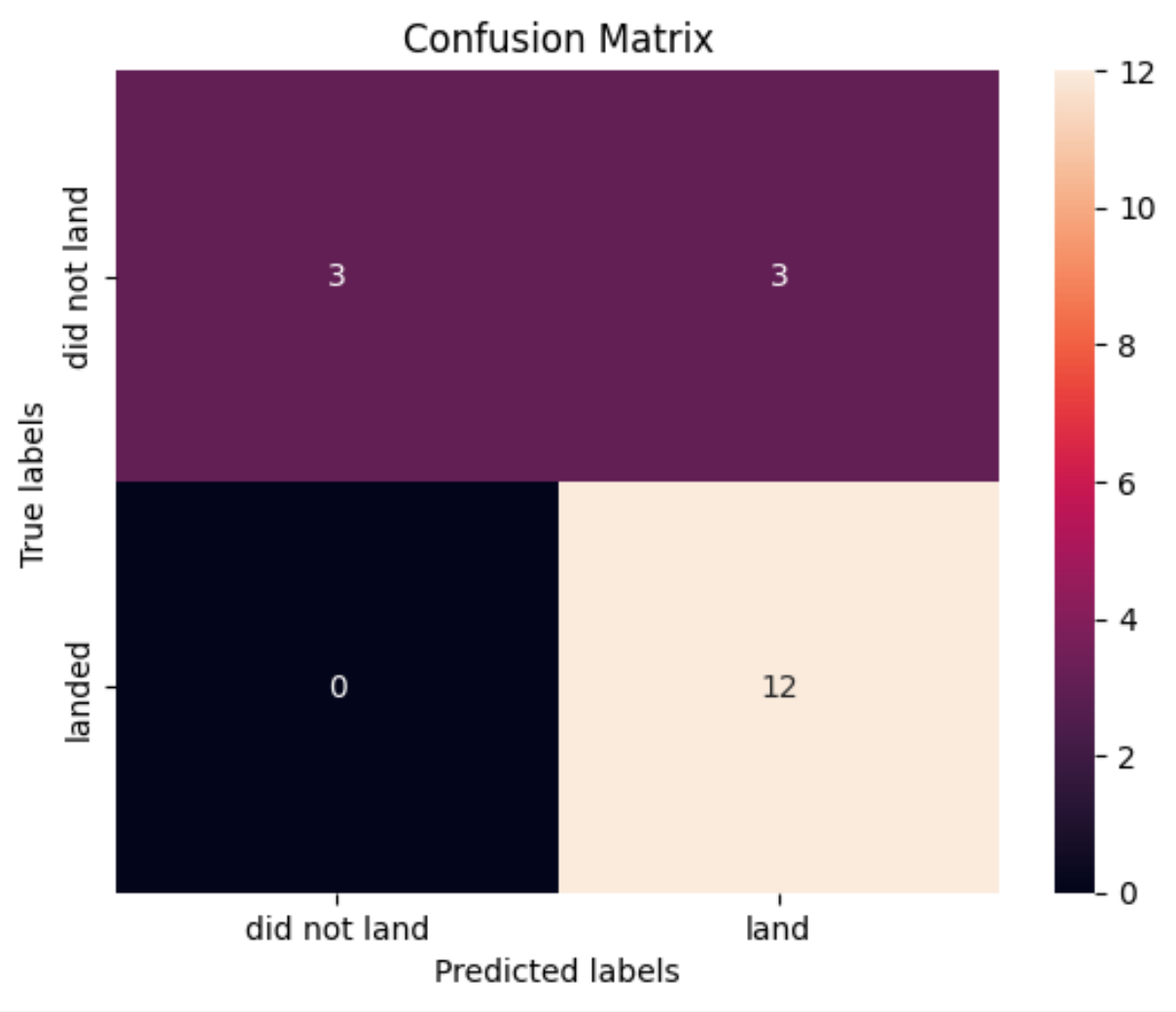
Predictive Analysis (Classification)

Classification Accuracy



- We can see that Logistic Regression, k-Nearest Neighbors and Support Vector Machine all have an accuracy of 83.33%
- Decision Tree has a lower accuracy of 66.67%

Confusion Matrix



- This is the confusion matrix of the best performing models.

CONCLUSIONS

- Hence, we have explored and visualized the SpaceX launch data using various techniques to gain insights into the data and its characteristics.
- We utilized SQL queries to extract relevant information and perform data analysis, contributing to a comprehensive understanding of the dataset.
- We created informative Folium maps to visualize launch site locations, proximity to coastlines, and other geographical features.
- We developed an interactive Plotly dashboard that enables users to explore launch site success rates and payload ranges.
- We employed classification models (Logistic Regression, Decision Tree, k-Nearest Neighbors, and Support Vector Machine) to predict launch success based on various features.
- We leveraged machine learning techniques to determine the best-performing model, evaluating their accuracy on test data.
- We have demonstrated the power of data wrangling, visualization, and machine learning in extracting valuable insights from complex datasets.
- We have also highlighted the importance of a multidisciplinary approach involving data engineering, analysis, visualization, and predictive modeling.
- The insights gained from this analysis provide valuable information for SpaceX's future launches and decision-making processes.

APPENDIX

- Github link for full repository - https://github.com/raghav8822/Data-Science-Capstone---Coursera
- Python code for Plotly Dashboard - https://github.com/raghav8822/Data-Science-Capstone---Coursera/blob/main/7.spacex_dash_app.py
- Original SpaceX API with launch data - https://api.spacexdata.com/v4/launches/past
- Original static json url - https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json

Thank you!