IBM Developer
SKILLS NETWORK

Ashish Kumar Singh
02 May 2024

Outline

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

Executive Summary

Summary of methodologies

Our strategy involved two main methods: integrating APIs and utilizing web scraping to gather data. After collecting the data, we used various Python techniques to carefully clean and process it. We then used SQL queries to extract relevant information from the refined dataset. Initial insights were gained through systematic data visualization and trend analysis. To conclude our analytical process, we utilized supervised machine learning models to predict the success of landing events. These models were applied to forecast the likelihood of landing event success.

Summary of all results

By conducting thorough data analysis, we uncovered clear patterns and connections among variables that directly impact the outcome of landing events. Using these findings, we built and refined a predictive model that proved highly effective in accurately predicting the likelihood of a successful landing event. It's noteworthy that the model achieved an impressive accuracy rate of 83%, highlighting its reliability in making predictions within this field.

Introduction

- SpaceX's dedication to reusable rockets has substantially reduced space travel expenses by prioritizing the retrieval of the first rocket stage. Recovering this primary phase is crucial for preserving and reusing costly components, directly cutting costs. A thorough examination of the success rate of these retrieval missions acts as a key indicator for assessing efficiency and cost-effectiveness in SpaceX's innovative strategy. This specific project aims to predict the success of the first stage retrieval event, providing predictive insights to improve decision-making in the space industry.
- Our goal is to predict the success of retrieving the first phase of the rocket, with the primary aim of improving how resources are allocated. By attaining this predictive ability, we aim to boost mission success rates and generate significant cost reductions.


Methodology
Section 1

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

Describe how data sets were collected:

The initial data collection involved utilizing the SpaceX API, a RESTful API, through a process of sending a GET request to the SpaceX API. This process began by creating a series of helper functions designed to facilitate the extraction of information using identification numbers within the launch data. Subsequently, rocket launch data was requested from the SpaceX API URL.

To ensure uniformity in the obtained JSON results, the SpaceX launch data underwent parsing and decoding of the response content from the GET request, resulting in a JSON format. This JSON result was then transformed into a Pandas DataFrame for further analysis.

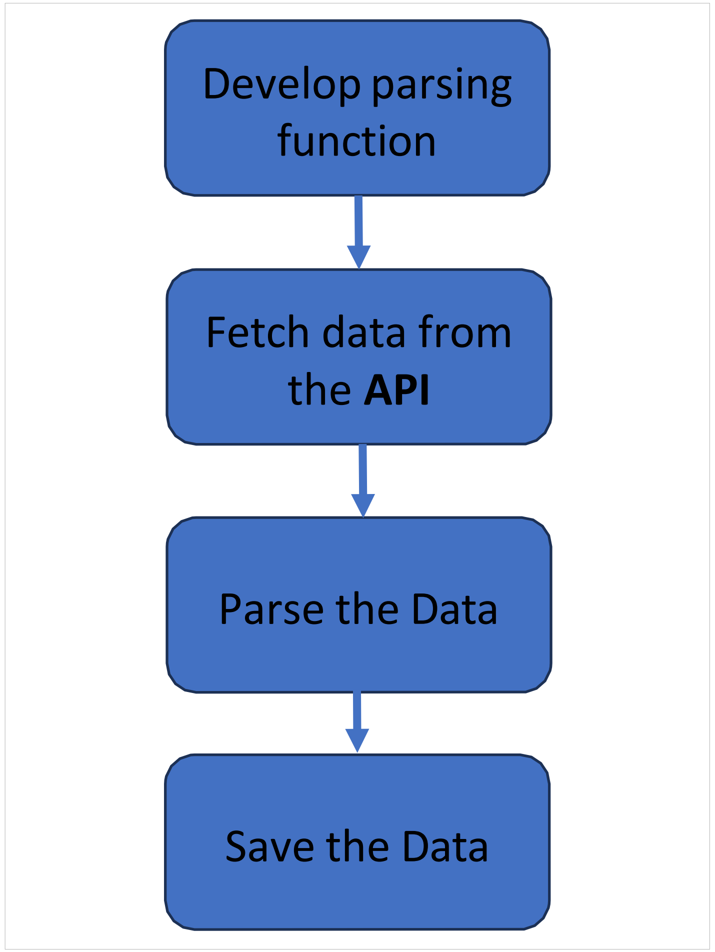
Furthermore, web scraping techniques were employed to gather historical Falcon 9 launch records from a Wikipedia page. This involved extracting data from an HTML table on the Wikipedia page titled **List of Falcon 9 and Falcon Heavy launches of the launch** records are stored in a HTML. Using BeautifulSoup and request Libraries, The extracted data was then parsed and converted into a Pandas DataFrame for analysis.

Data Collection – SpaceX API

1) Develop a function for parsing the data.
2) Fetch the data from the REST API using the GET method.
3) Use previously created functions to parse the data.
4) Save the data in **DataFrame**.

GitHub URL of the completed SpaceX API calls notebook:

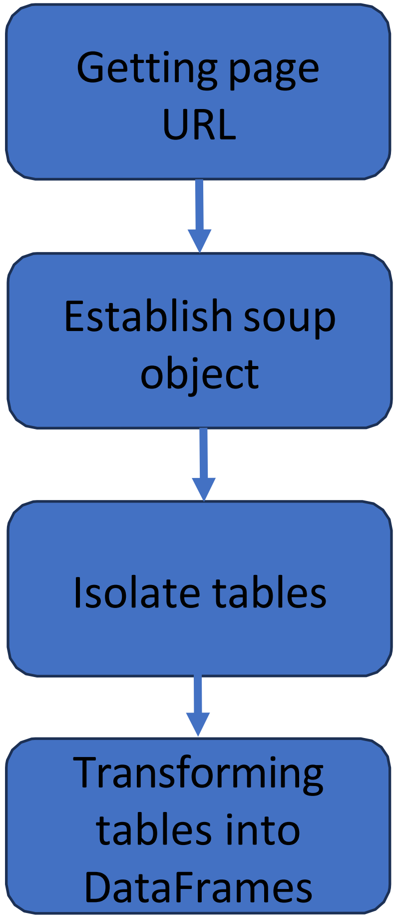
(Jupyter Labs SpaceX Data Collection API)



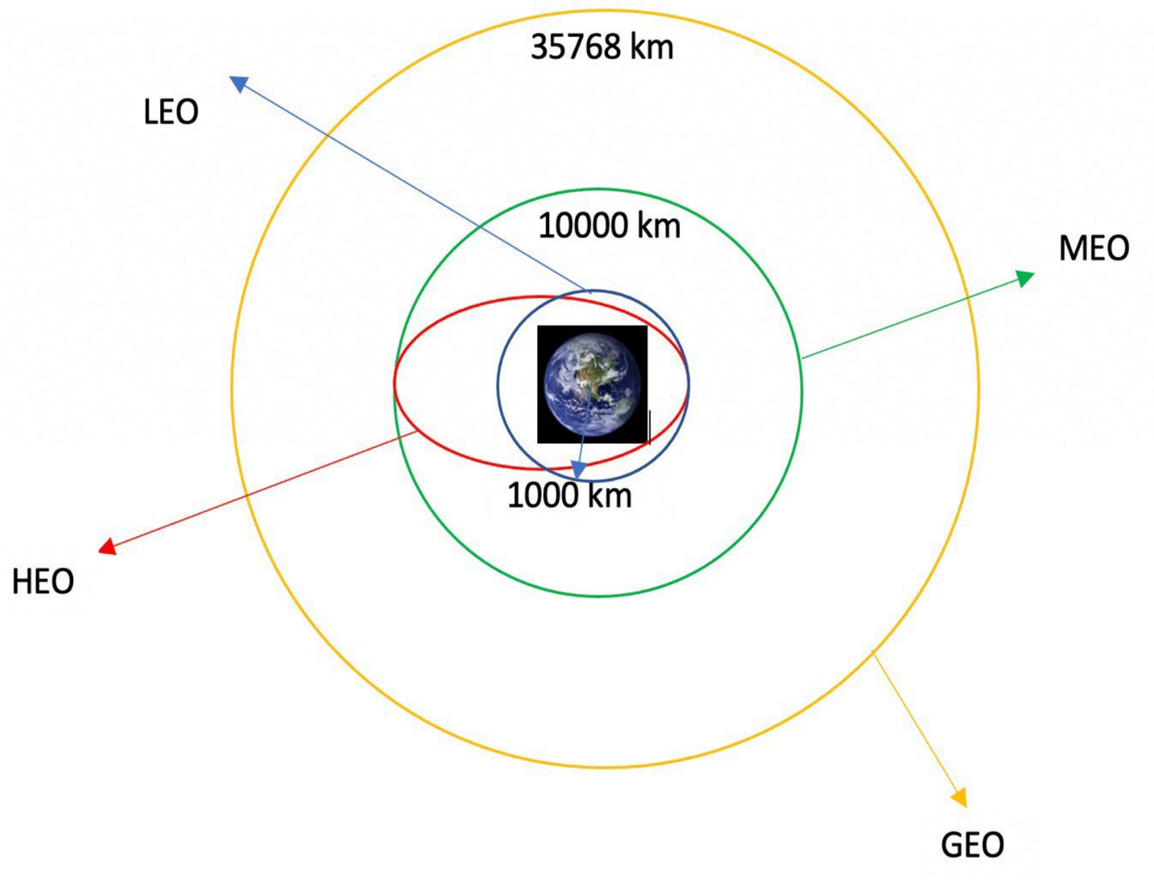
Data Collection - Scraping

1) Employed request.get method to retrieve the webpage.
2) Established a BeautifulSoup object for manipulating the HTML text.
3) Utilized soup manipulation methods to isolate the relevant tables.
4) Transformed the HTML data into a Pandas DataFrame format.

GitHub URL of the completed web scraping notebook:
[Jupyter Labs Webscraping]



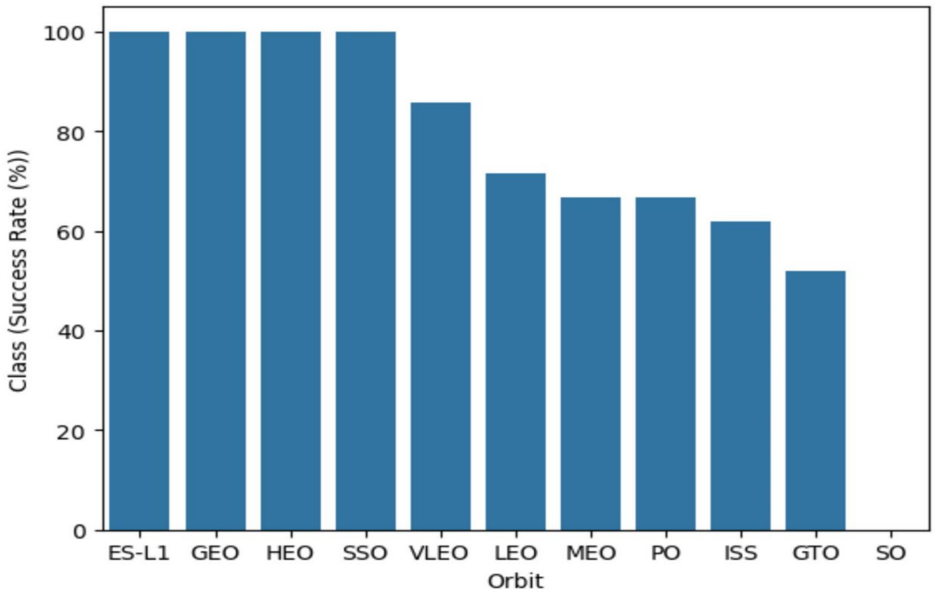
Data Wrangling



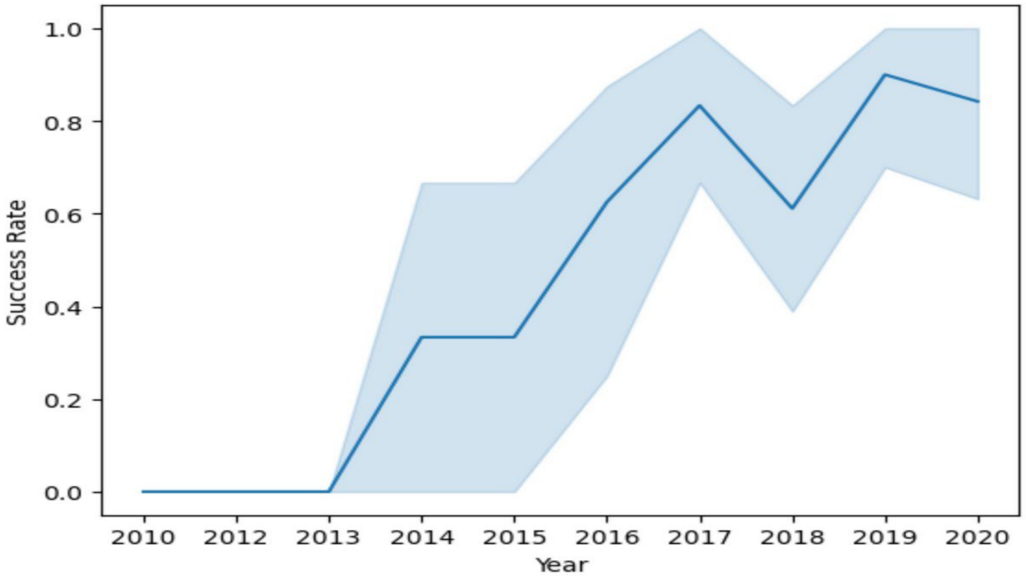
We conducted exploratory data analysis to establish the training labels.
This involved calculating the frequency of launches at each site, as well as the occurrence of orbits.
we generated a landing outcome label based on the outcome column and exported the findings to a **CSV** file.

- For Girhub Notebook File [Click Here]

EDA with Data Visualization



We investigated the data through visualizations, examining the correlation between flight number and launch site, payload and launch site, success rates for each orbit type, flight number and orbit type, as well as the annual trend in launch success.



- For Github Notebook FilClick Here

EDA with SQL

We utilized SQL to execute numerous queries, enhancing our comprehension of the dataset

- Displaying the launch site names.
- Showing 5 records where launch sites start with 'CCA'.
- Presenting the total payload mass carried by boosters launched by NASA (CRS).
- Displaying the average payload mass carried by booster version F9 v1.1.
- Listing the date of the first successful landing outcome on a ground pad.
- Listing the names of boosters that successfully landed on a drone ship with a payload mass greater than 4000 but less than 6000.
- Listing the total number of successful and failed mission outcomes.
- Listing the names of booster versions that carried the maximum payload mass.
- Listing failed landing outcomes on a drone ship, along with their booster versions and launch site names for the year 2015.
- Ranking the count of landing outcomes or successes between June 4, 2010, and March 20, 2017, in descending order.

For Github Notebook File Click Here

Build an Interactive Map with Folium

- We labeled all launch sites and incorporated map elements like markers, circles, and lines to denote the success or failure of launches for each site on the Folium map.
- We assigned launch outcomes (failure or success) to classes 0 and 1, where 0 represents failure and 1 represents success.
- By utilizing color-coded marker clusters, we identified launch sites with relatively high success rates.

Furthermore, we computed the distances between each launch site and its surroundings to address specific inquiries, such as:

  - Are launch sites situated near railways, highways, or coastlines?
  - Do launch sites maintain a certain distance from cities?

For Github Notebook File Click Here

Build a Dashboard with Plotly Dash

- Using Plotly Dash, we created an interactive dashboard to visualize launch data.
- Our interactive dashboard, built with Plotly Dash, features pie charts for launch distribution by launch site
- Scatter plots exploring the correlation between booster version, payload mass, and launch outcome.

For Github Notebook File Click Here

Predictive Analysis (Classification)

After loading and manipulating the data with NumPy and Pandas, we split it for training and testing.

We then built and optimized various machine learning models using GridSearchCV to find the best performing classification model.

This process involved tuning hyperparameters and applying feature engineering, all while evaluating models based on accuracy.

For Github Notebook File Click Here

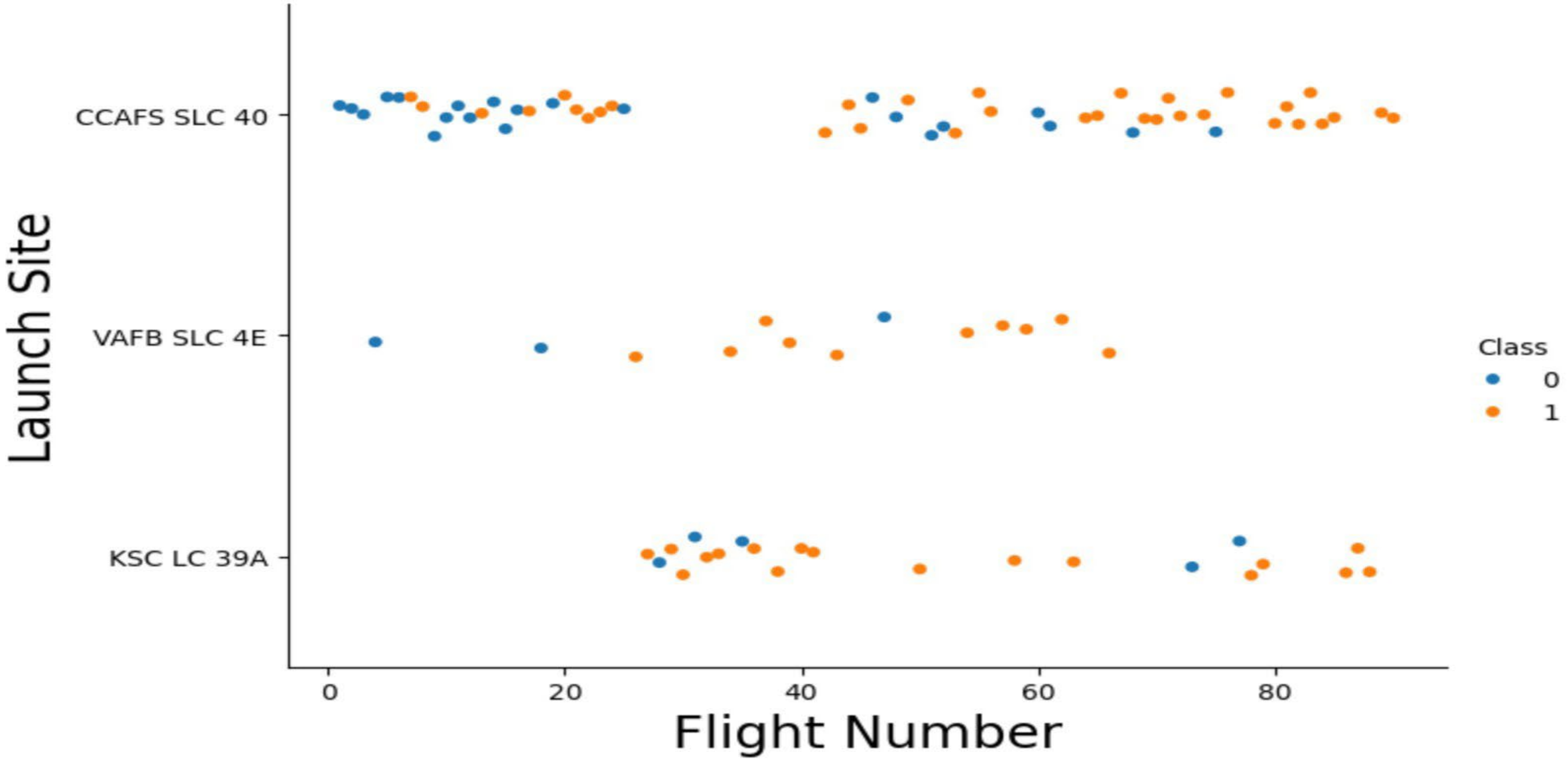
Results

- Findings from exploratory data analysis
- Screenshots of the interactive analytics demonstration
- Outcomes of predictive analysis

Section 2

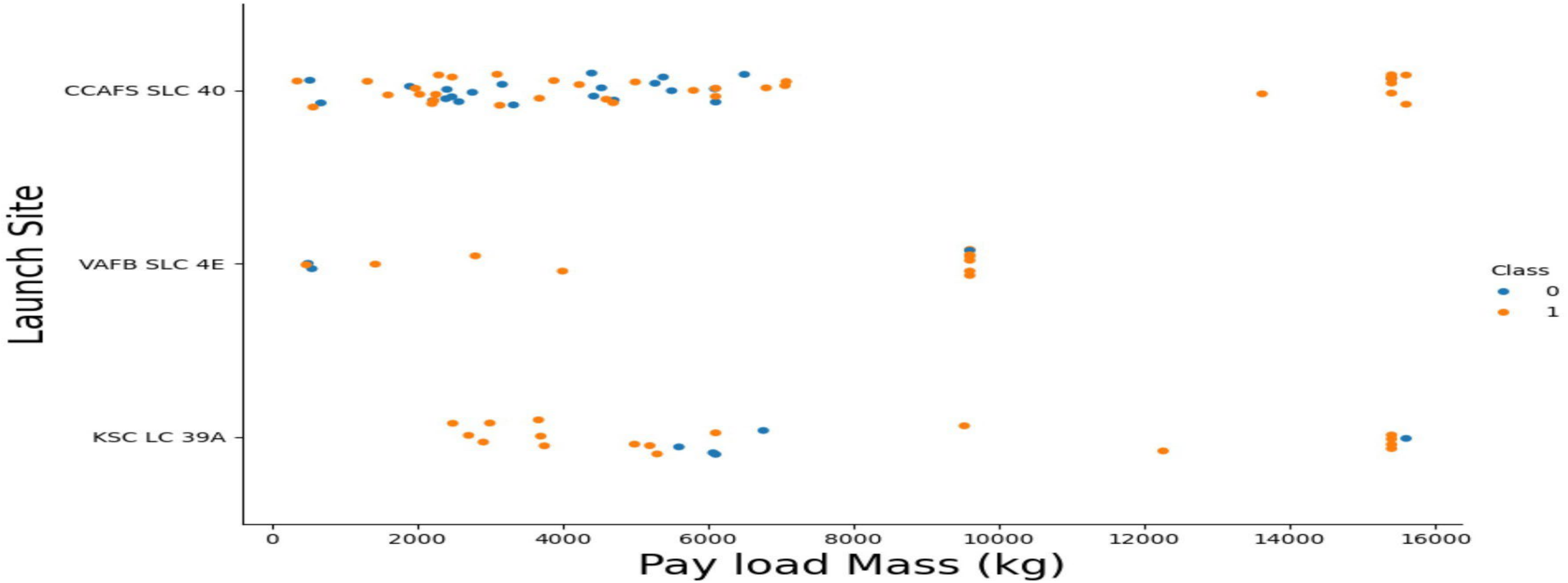
Insights from EDA

Flight Number vs. Launch Site



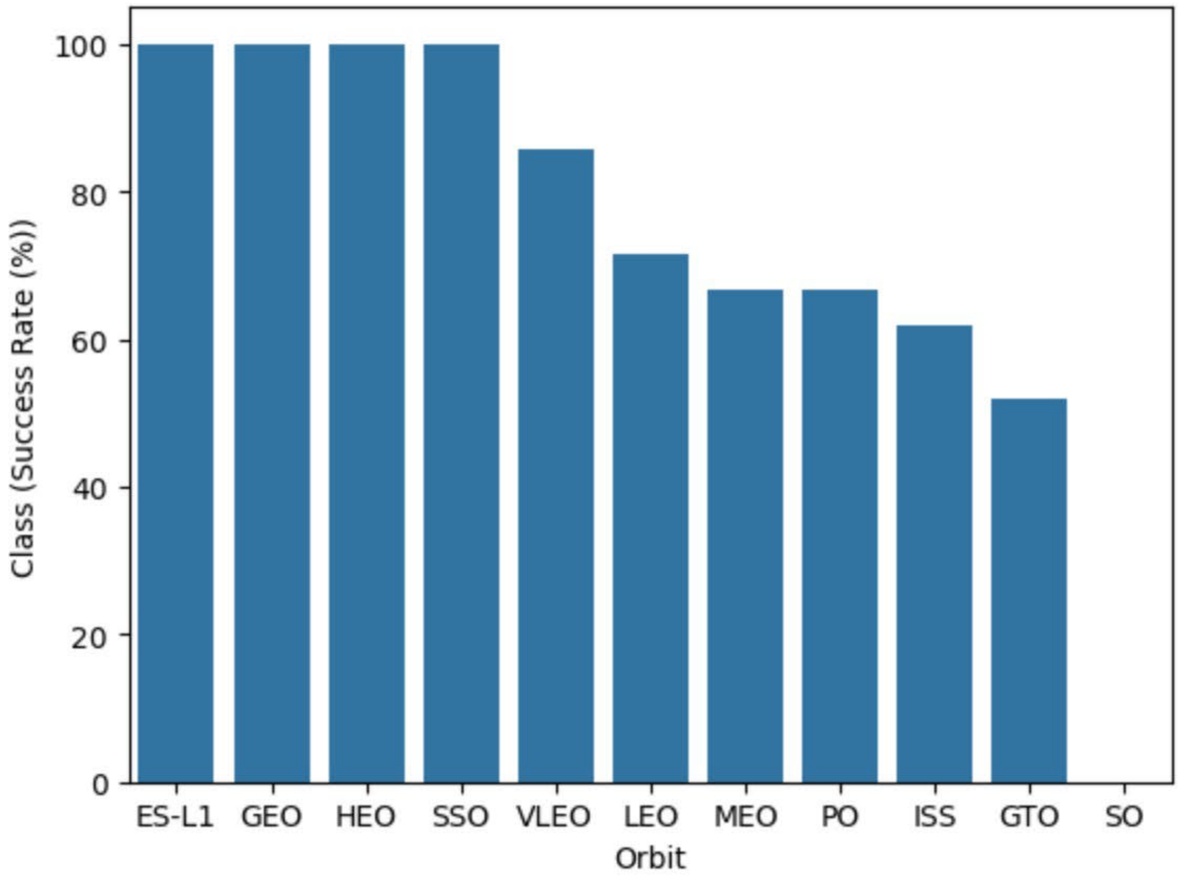
From the plot, we found that the larger the flight amount at a launch site, the greater the success rate at a launch site.

Payload vs. Launch Site

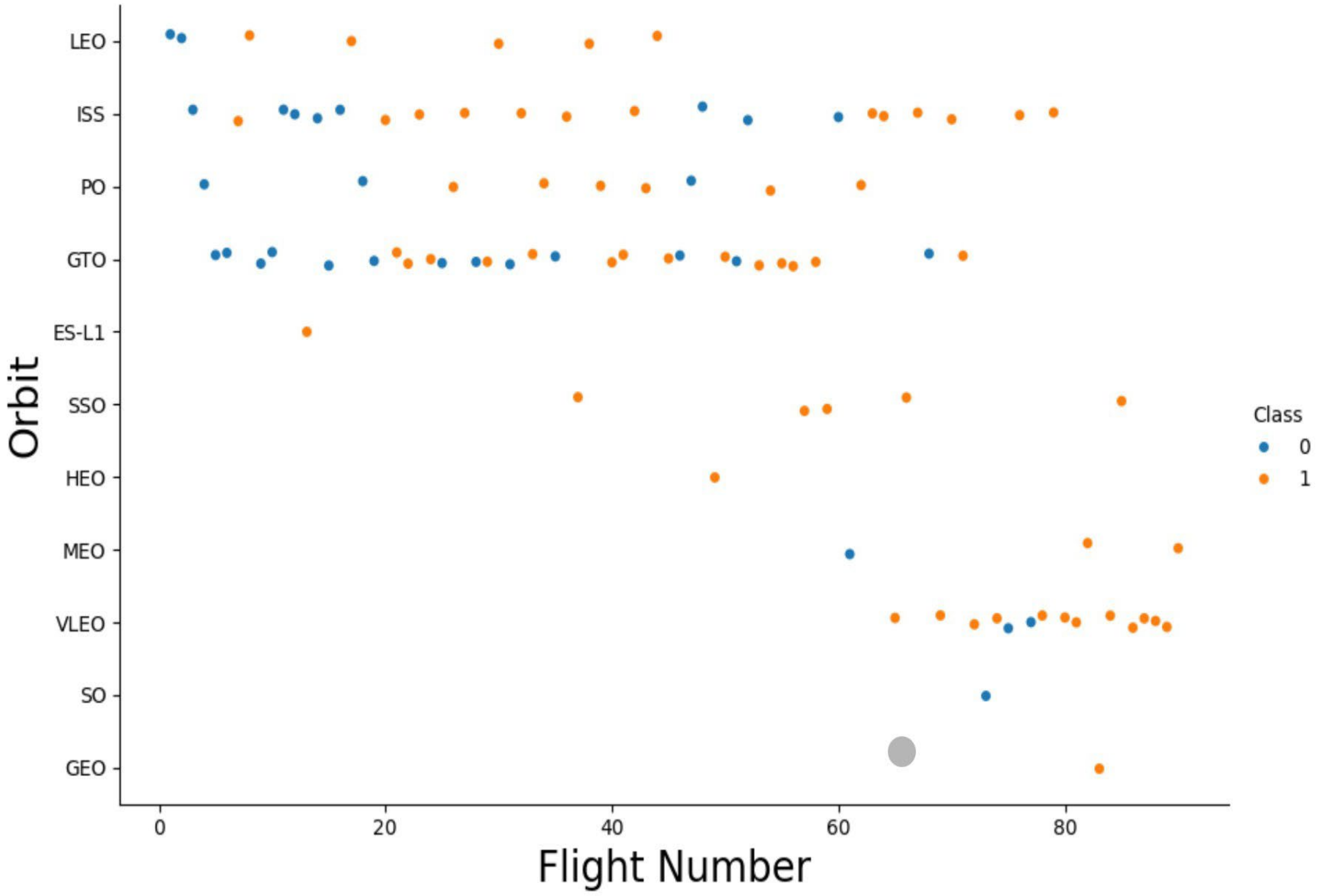


The greater the payload mass for launch site CCAFS SLC 40 the higher the success rate for the rocket.

Success Rate vs. Orbit Type

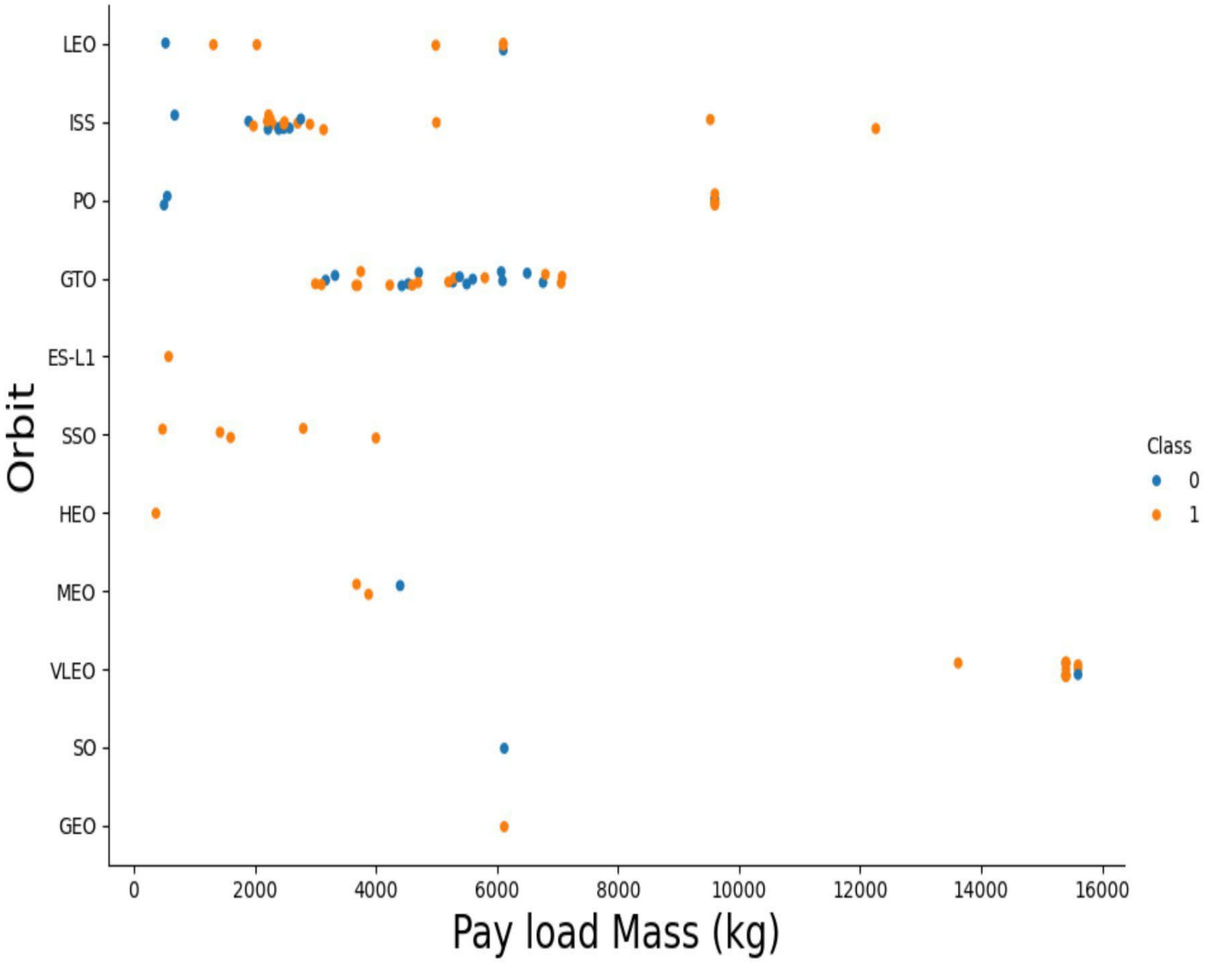


Flight Number vs. Orbit Type



- The graph depicted illustrates the relationship between Flight Number and Orbit Type. It is noticeable that in the LEO orbit, success correlates with the number of flights, whereas in the GTO orbit, there is no discernible connection between flight number and orbit success.

Payload vs. Orbit Type



- It can be noted that for heavy payloads, there is a higher frequency of successful landings observed for the PO, LEO, and ISS orbits.

All Launch Site Names

Display the names of the unique launch sites in the space mission

```
In [31]: %sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTBL;

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

Out[31]:

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

We used the key word DISTINCT to show only unique launch sites from the SpaceX data.

Launch Site Names Begin with 'CCA'

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

```
%sql SELECT * FROM 'SPACEXTBL' 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	Landin _Outcom
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failu (parachut
08-12-2010	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	Failu (parachut
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attem
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attem
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attem

Used the query here to display 5 records with launch sites begin with CCA.

Total Payload Mass

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

In [17]:
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) as "Total Payload Mass(Kgs)", Customer FROM 'SPACEXTBL' WHERE Customer = 'NASA
```

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

Out[17]:

Total Payload Mass(Kgs)	Customer
45596	NASA (CRS)

Calculated the total payload carried by boosters from NASA as 45596 using the query below

Average Payload Mass by F9 v1.1

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

In [19]:
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) as "Payload Mass Kgs", Customer, Booster_Version FROM 'SPACEXTBL' WHERE Booste
```

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

Out[19]:

Payload Mass Kgs	Customer	Booster_Version
2534.6666666666665	MDA	F9 v1.1 B1003

Calculated the average payload mass carried by booster version F9 v1.1 B1003 as 2534.66666666666.

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

In [21]:
```
%sql SELECT MIN(DATE) FROM 'SPACEXTBL' WHERE "Landing _Outcome" = "Success (ground pad)";
```

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

Out[21]:

MIN(DATE)
01-05-2017

We noted that the initial successful landing on a ground pad occurred on May 1st, 2017.

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

In [26]:
```
# %sql SELECT * FROM 'SPACEXTBL'
```

In [27]:
```
%sql SELECT DISTINCT Booster_Version, Payload FROM SPACEXTBL WHERE "Landing _Outcome" = "Success (drone ship)" AN
```

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

Out[27]:

Booster_Version	Payload
F9 FT B1022	JCSAT-14
F9 FT B1026	JCSAT-16
F9 FT B1021.2	SES-10
F9 FT B1031.2	SES-11 / EchoStar 105

We employed the WHERE clause to filter boosters that have successfully landed on a drone ship, and then used the AND condition to identify successful landings with a payload mass ranging from **greater than 4000 to less than 6000**.

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

In [28]:
```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") as Total FROM SPACEXTBL GROUP BY "Mission_Outcome";
```

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

Out[28]:

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

Total number of Mission outcome – success/failure.

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [30]:
```
%sql SELECT "Booster_Version",Payload, "PAYLOAD_MASS__KG_" FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX
```

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

Out[30]:

Booster_Version	Payload	PAYLOAD_MASS__KG_
F9 B5 B1048.4	Starlink 1 v1.0, SpaceX CRS-19	15600
F9 B5 B1049.4	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600
F9 B5 B1051.3	Starlink 3 v1.0, Starlink 4 v1.0	15600
F9 B5 B1056.4	Starlink 4 v1.0, SpaceX CRS-20	15600
F9 B5 B1048.5	Starlink 5 v1.0, Starlink 6 v1.0	15600
F9 B5 B1051.4	Starlink 6 v1.0, Crew Dragon Demo-2	15600
F9 B5 B1049.5	Starlink 7 v1.0, Starlink 8 v1.0	15600
F9 B5 B1060.2	Starlink 11 v1.0, Starlink 12 v1.0	15600
F9 B5 B1058.3	Starlink 12 v1.0, Starlink 13 v1.0	15600
F9 B5 B1051.6	Starlink 13 v1.0, Starlink 14 v1.0	15600
F9 B5 B1060.3	Starlink 14 v1.0, GPS III-04	15600
F9 B5 B1049.7	Starlink 15 v1.0, SpaceX CRS-21	15600

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.

In [68]:
```
%sql SELECT substr(Date,7,4), substr(Date, 4, 2),"Booster_Version", "Launch_Site", Payload, "PAYLOAD_MASS__KG_",
```

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

Out[68]:

substr(Date,7,4)	substr(Date, 4, 2)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Mission_Outcome	Landing _Outcome
2015	01	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	Success	Failure (drone ship)
2015	04	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898	Success	Failure (drone ship)

Utilized a combinations of the WHERE clause, LIKE, AND, and Between conditions to filter for failed landing outcomes in drone ship, their booster versions, and launch site names for 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.

In [45]:

```
%sql SELECT LANDING_OUTCOME, COUNT(*) AS COUNT_LAUNCHES FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-
```

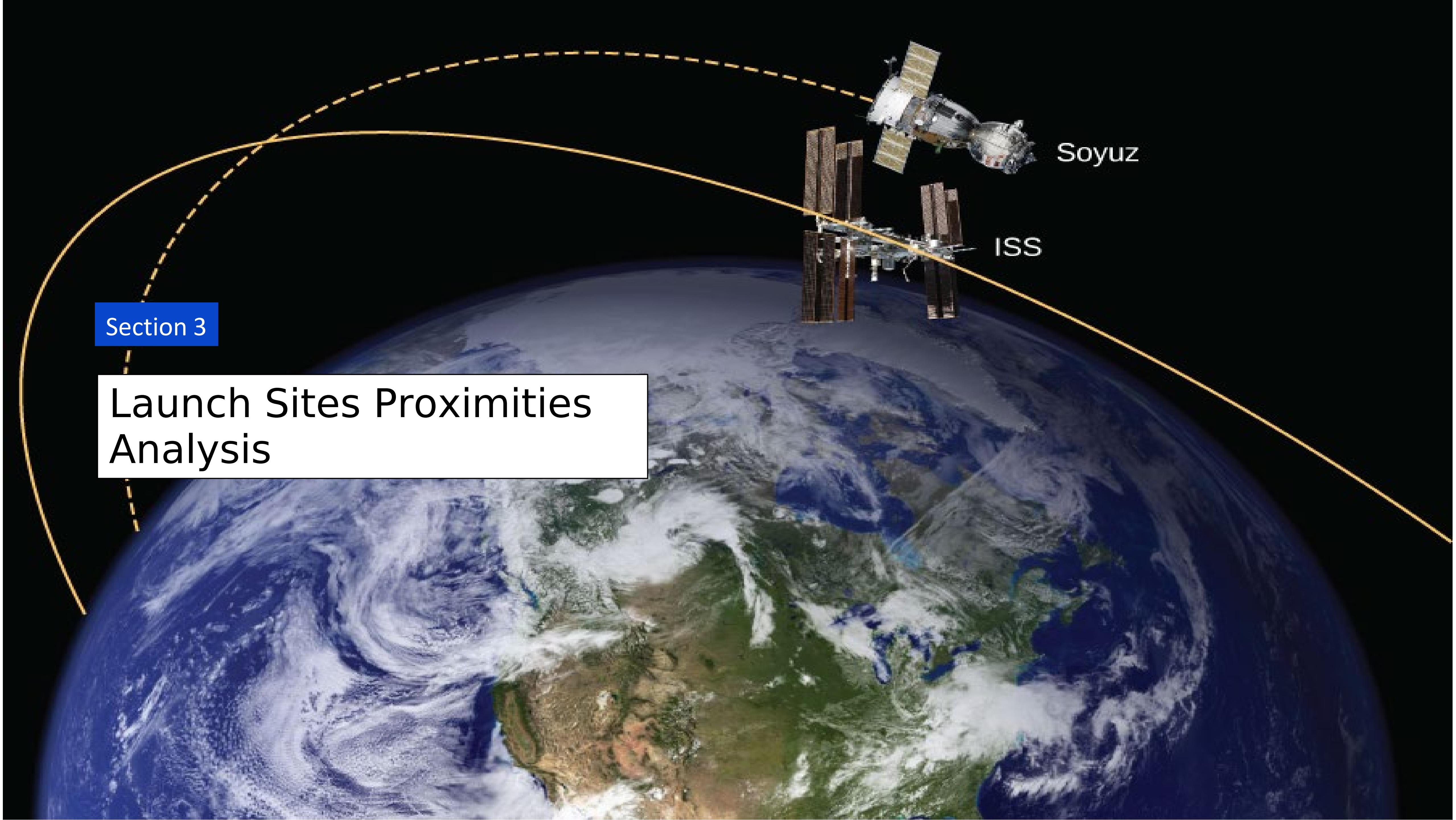
```
 * sqlite:///my_data1.db
Done.
```

Out[45]:

Landing_Outcome	COUNT_LAUNCHES
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

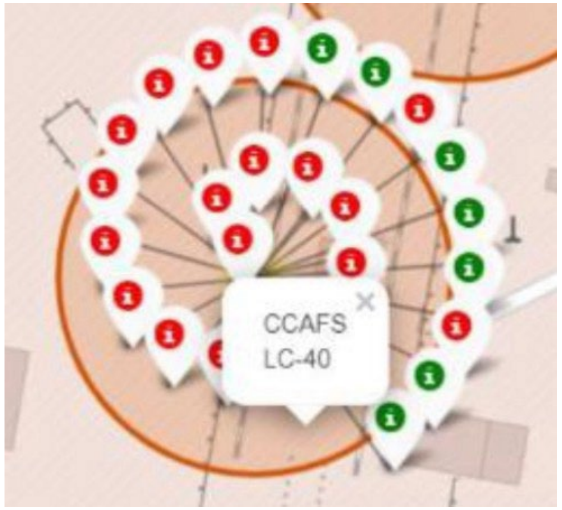
All launch sites global map markers

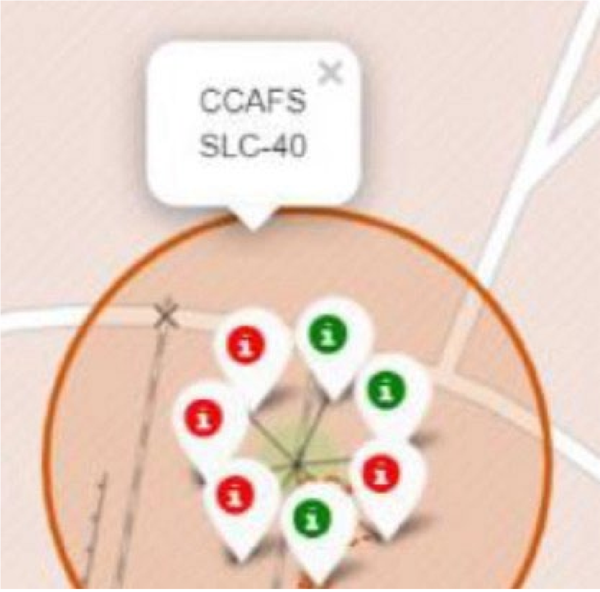
We can note that the Space launch sites are in the United States of America coasts. Florida and California

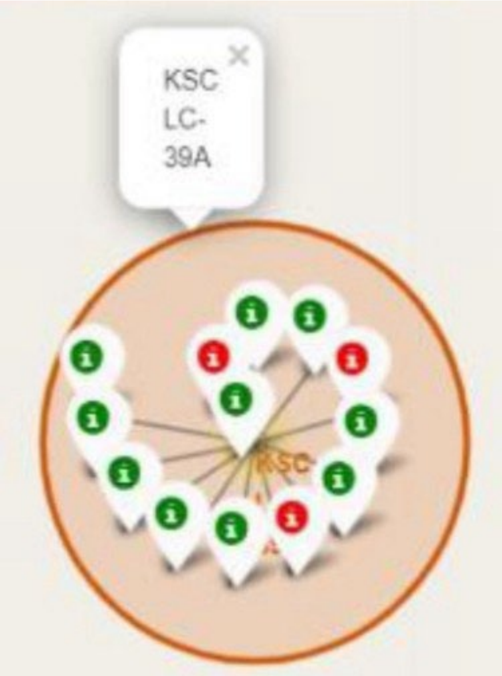
Markers showing launch sites with color labels

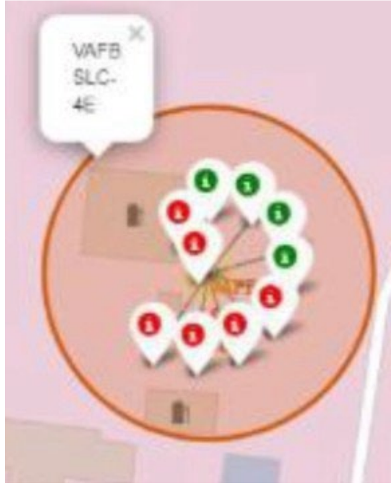
Green markers show **Successful** Launches.

Red markers show **Unsuccessful** Launches.









Launch Site distance to landmarks

Are lunch sites is close proximity to railways?

No

Are lunch sites is close proximity to highway?

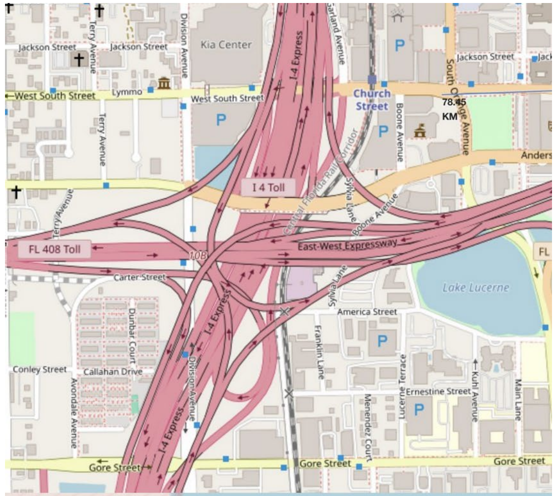
No

Are lunch sites is close proximity to coastline?

Yes

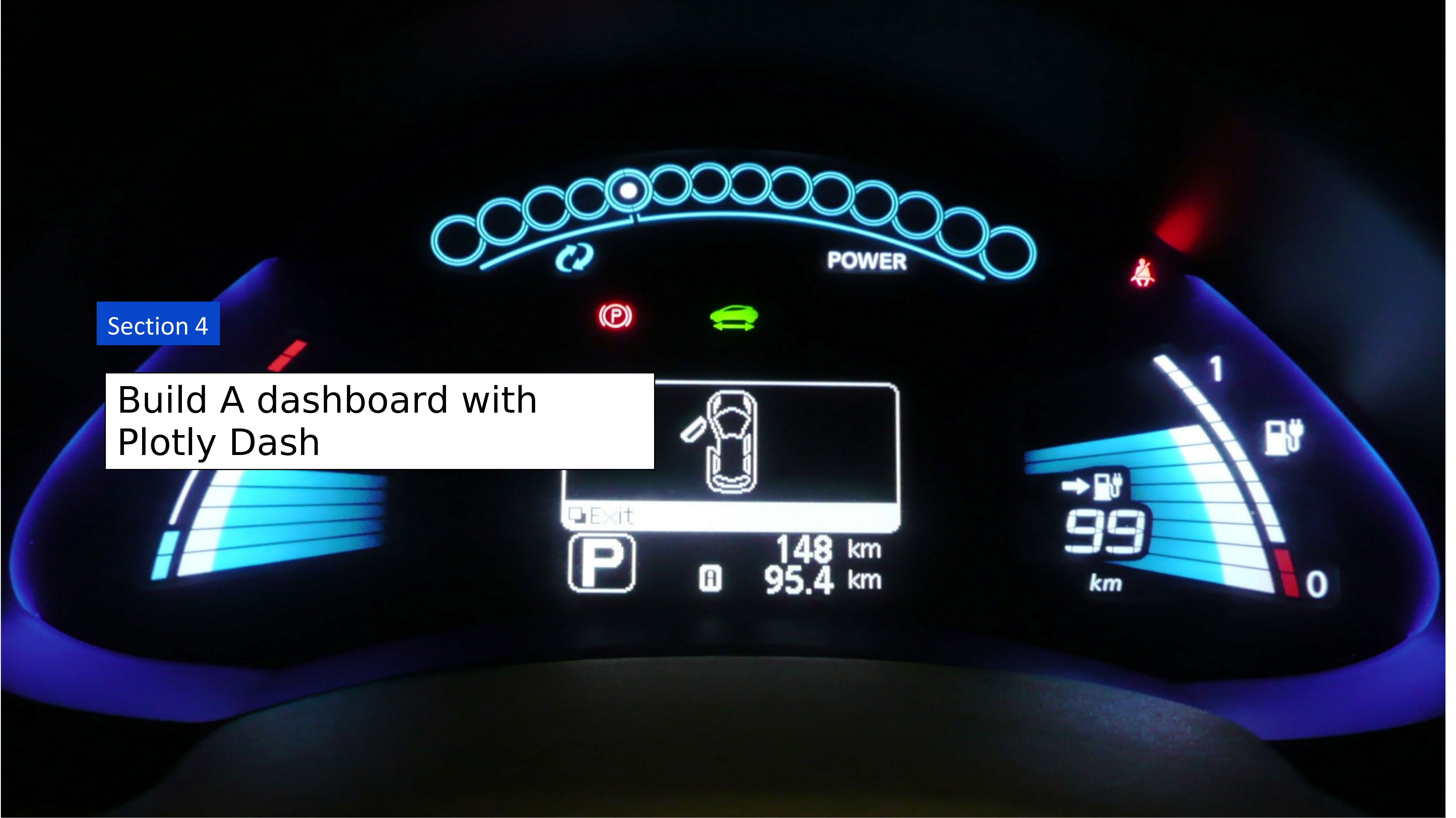
Do launch sites keep certain distance away from cities?

Yes



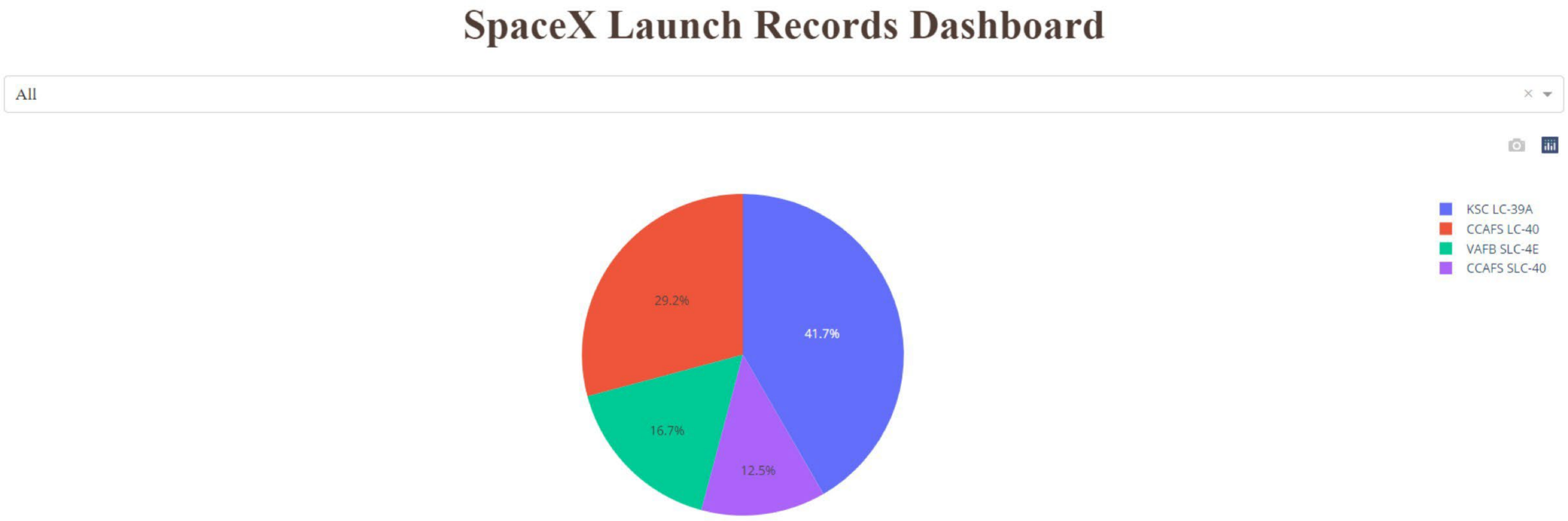
Section 4

Build A dashboard with Plotly Dash



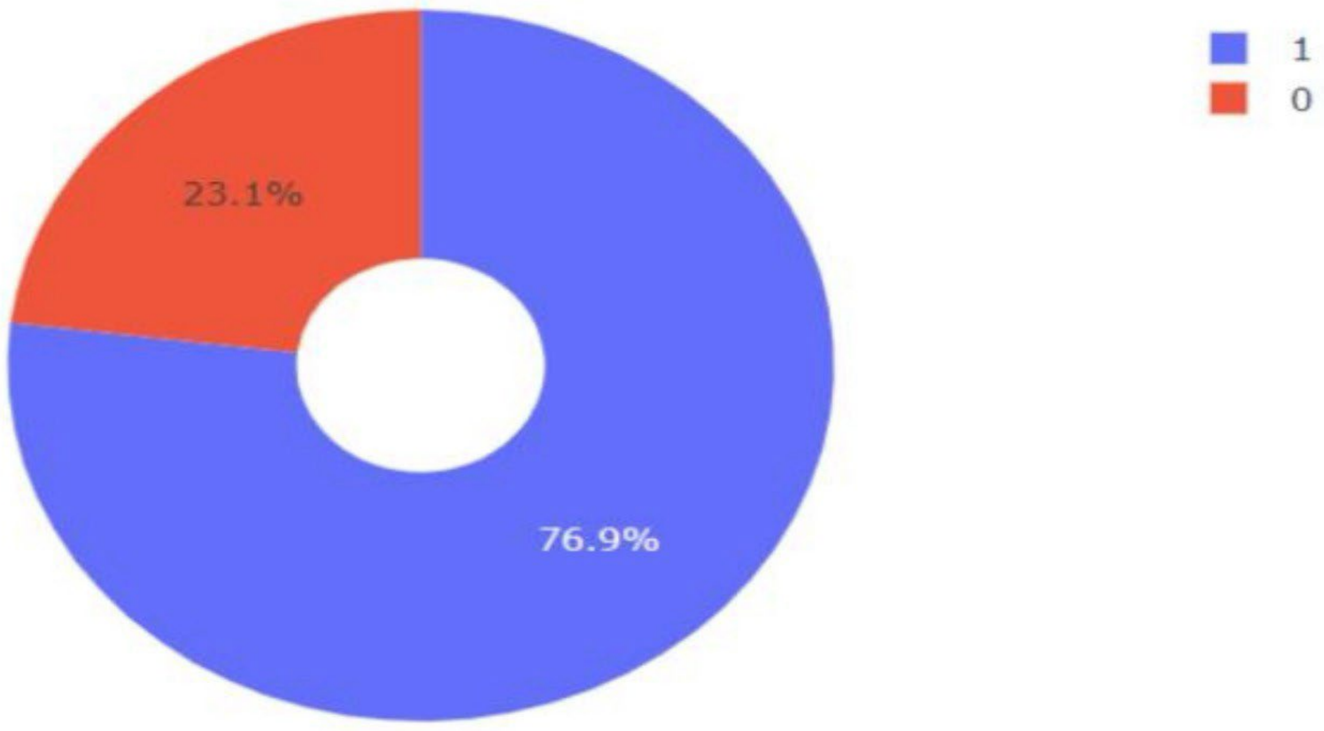
Pie chart showing the success percentage achieved by each launch site

Total success launches by all sites



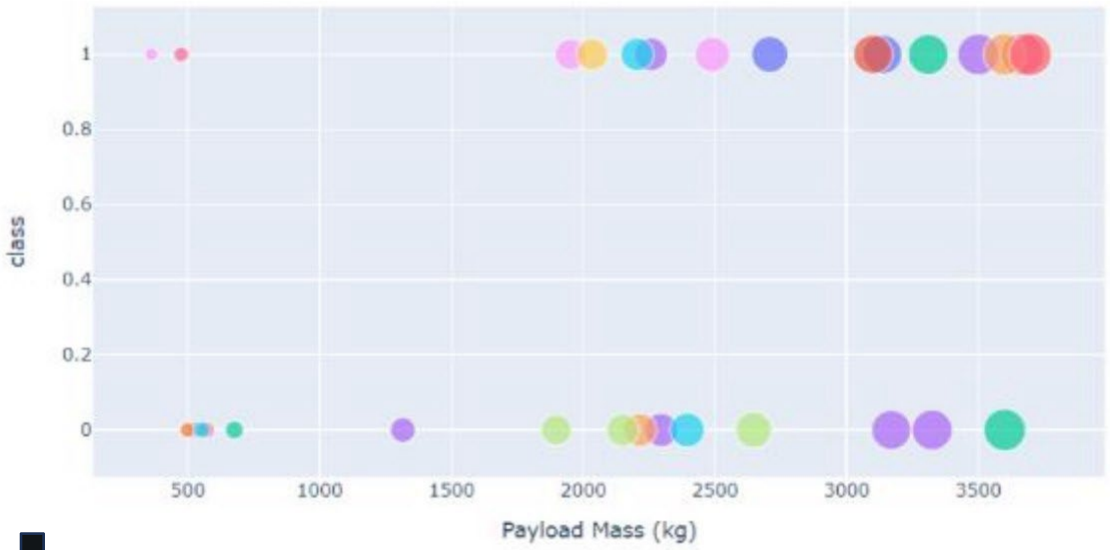
Lets note that KSC LC-39A had the most successful launches from all the sites.

Pie chart showing the Launch site with the highest launch success ratio

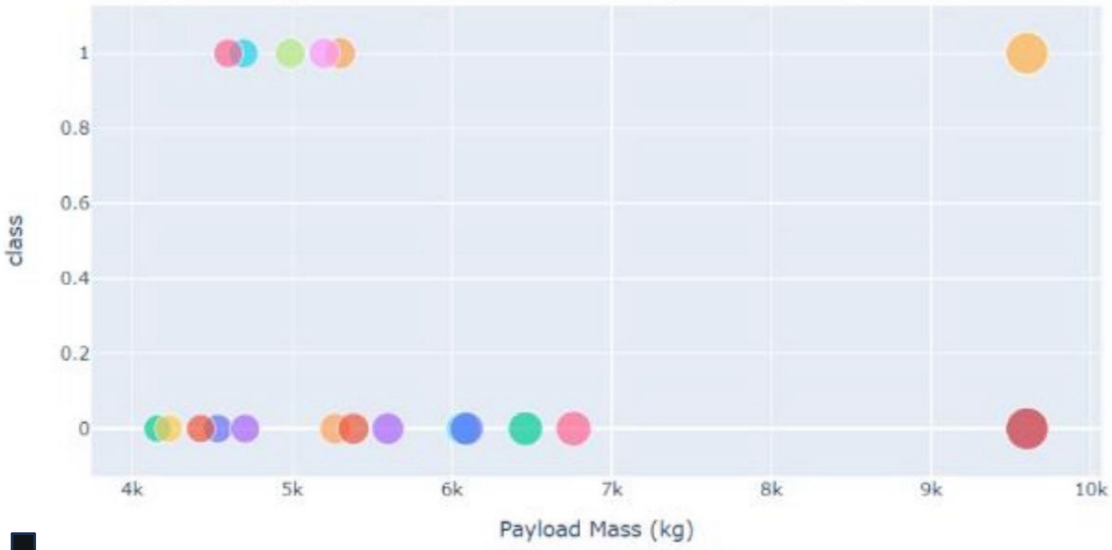


KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Scatter plot of Payload vs Launch Outcome



Low Weighted Payload 0kg - 4000kg



Heavy Weighted Payload 4000kg - 10000kg

Success rates for low weighted payloads is higher than the heavy weighted payloads.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

Among the models considered, the decision tree classifier achieved the highest accuracy in classification.

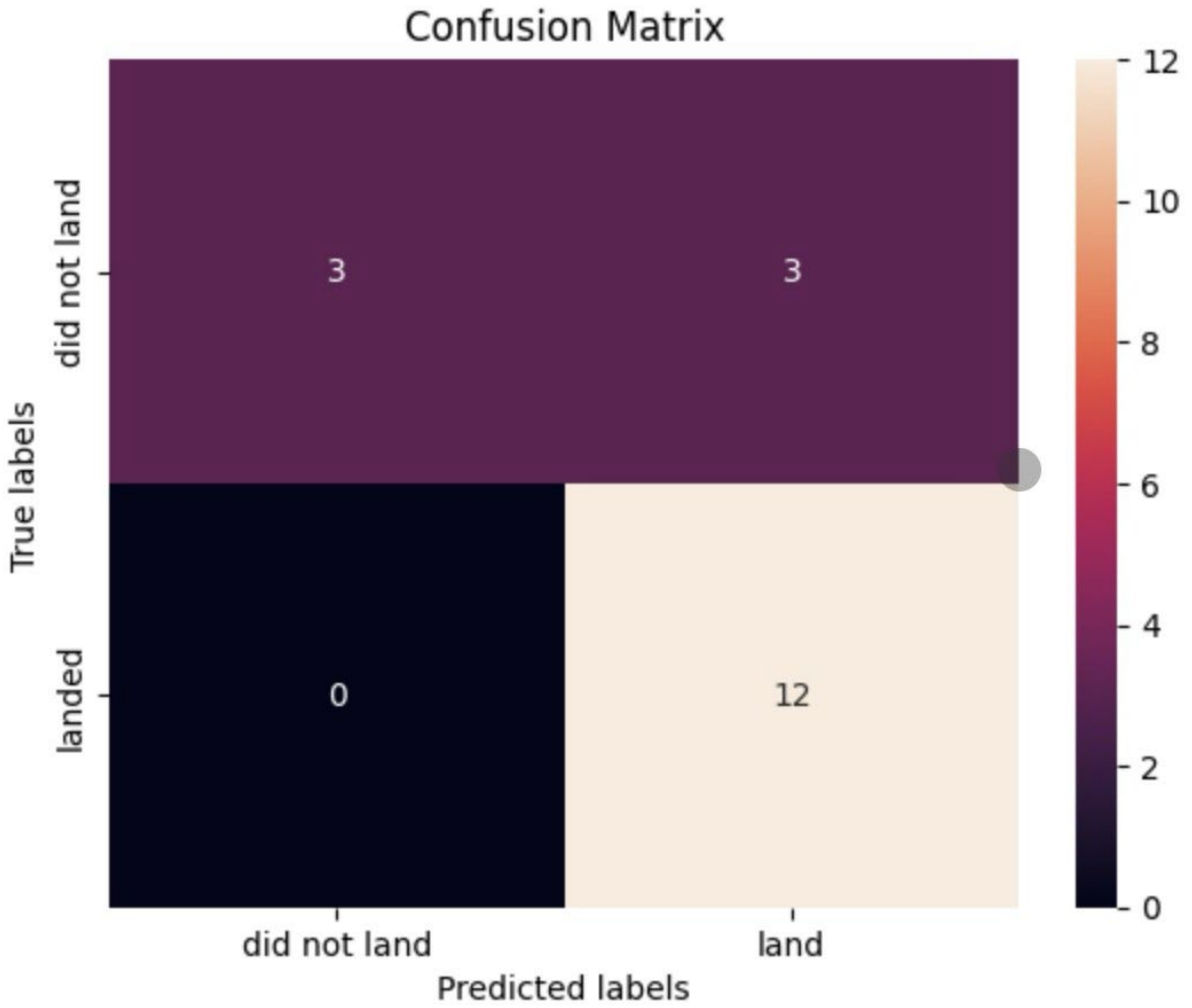
In [46]:

```
algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_}
bestalgorithm = max(algorithms, key=algorithms.get)
print('Best Algorithm is',bestalgorithm,'with a score of',algorithms[bestalgorithm])
if bestalgorithm == 'Tree':
    print('Best Params is :',tree_cv.best_params_)
if bestalgorithm == 'KNN':
    print('Best Params is :',knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best Params is :',logreg_cv.best_params_)
```

```
Best Algorithm is Tree with a score of 0.8767857142857143
Best Params is : {'criterion': 'entropy', 'max_depth': 2, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_sampl
es_split': 2, 'splitter': 'random'}
```

Confusion Matrix

The confusion matrix of the decision tree classifier indicates its ability to differentiate between various classes, with the primary issue being false positives, where unsuccessful landings are incorrectly identified as successful by the classifier.



Conclusions

From our analysis, we can infer that:

- A higher number of flights at a launch site correlates with a higher success rate.
- The success rate of launches began to rise in 2013 and continued until 2020. Notably, orbits ES-L1, GEO, HEO, SSO, and VLEO exhibited the highest success rates.
- KSC LC-39A experienced the highest number of successful launches among all sites.
- The decision tree classifier emerged as the most effective machine learning algorithm for this particular task.

Appendix

Include 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