IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Nur Ibrahim
05/01/2025

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection with API and Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL and Visualization
  - Interactive Map with Folium
  - Interactive Dashboards with Plotly Dash
  - Model Prediction using Machine Learning
- Summary of all results
  - Exploratory Data Analysis Results
  - Interactive Analytics Results
  - Predictive Modeling Results

Introduction

- Project background and context

  The cost of rocket launches varies significantly across providers, with SpaceX offering Falcon 9 launches at approximately $62 million compared to upwards of $165 million for other providers. A key factor in SpaceX's cost efficiency is the reusability of the Falcon 9's first stage. Understanding whether the first stage will successfully land is critical to estimating launch costs. This insight can be valuable for alternate companies aiming to compete with SpaceX in the rocket launch market.

- Problems you want to find answers

  - What technical parameters impact the success of a Falcon 9 first-stage landing (e.g., rocket speed, fuel levels, payload weight)?
  - How do weather conditions, such as wind speed and sea state, influence landing success?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using web scraping
- Perform data wrangling
  - Exploratory Data Analysis to find some patterns in the data and determine what would be the label for training supervised models.
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Exploratory Data Analysis using python library (Matplotlib, Seaborn) and using SQL , Feature Engineering
- Perform interactive visual analytics using Folium and Plotly Dash
  - Use Folium to view previously observed correlations
- Perform predictive analysis using classification models
  - Create a machine learning pipeline to predict if the first stage will land given the data from the preceding observations

Data Collection

Import Libraries and Define Auxiliary Functions

Request and parse the SpaceX launch data using the GET request

Filter the dataframe to only include Falcon 9 launches

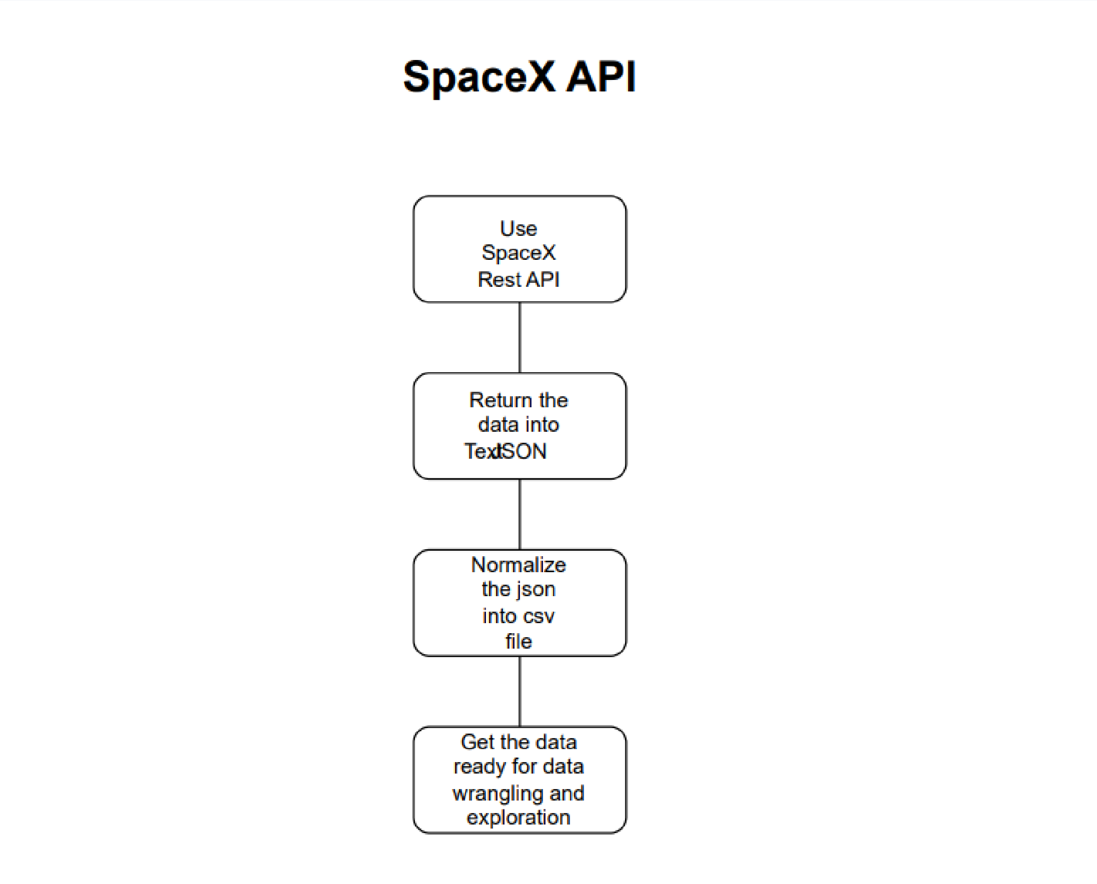
Libraries:	**Functions:**
Request	getBoosterVersion
Pandas	getLaunchSite
Numpy	getPayloadData
Datetime	getCoreData

Decode the response content as a Json using and turn it into a Pandas dataframe

Remove the Falcon 1 launches keeping only the Falcon 9 launches.

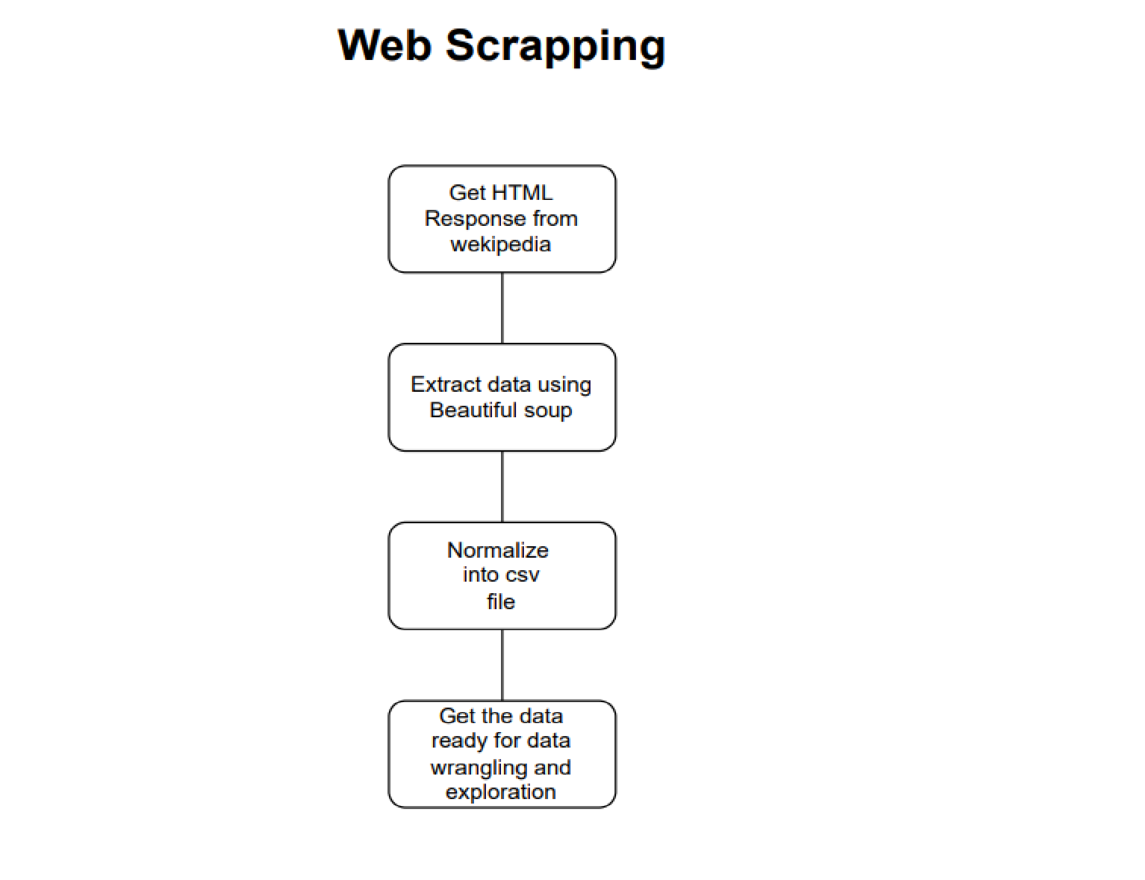
Data Collection – SpaceX API

Github URL : https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

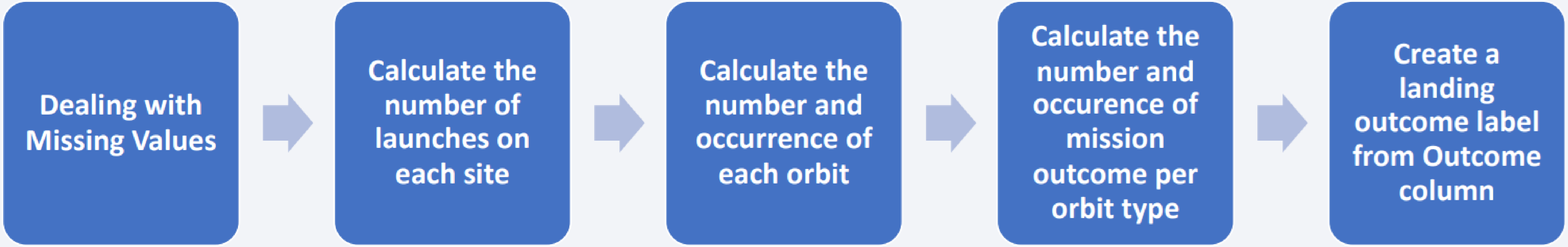


Data Collection - Scraping

Github URL : https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling



Github URL : https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- The graphs used are, Scatter Plot, Bar chart, and Line chart, because they are the ones that best highlight the relationships between the variables considered
- Github URL: https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/edadataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
  - Displaying the names of the unique launch sites in the space mission
  - Displaying 5 records where launch sites begin with the string ‘KSC’
  - Displaying the total payload mass carried by boosters launched by NASA (CRS)
  - Displaying average payload mass carried by booster version F9 v1.1
  - Listing the data where the successful landing outcome in drone ship was achieved
  - Listing the data where the successful landing outcomes in drone ship was achieved
  - Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
  - Listing the total number of successful and failure mission outcomes
  - Listing the names of the booster version which have carried maximum payload mass.
  - Listing the records which will display the month names, successful landing. Outcomes in ground pad booster
  - Various launch site for the months in year 2017
  - Ranking the count of successful landing outcomes between dates 2010 to 2017
- Github URL: https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summary of map objects
    - Markers: Show a geo location from latitude and longitude data
    - Cluster: Show a group of markers
    - Circles: Show a single location
    - Lines: Show distance between two

      I have added object to find some geographical patterns about launch site

- Github URL: https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
  - Bar: Show categories differences
  - Line: Reports time series changes
  - Pie: Shows the percentage of events
  - Tree: Shows complex relationship of variables in interactive way
  - Map: Shows variables of states on a map
- Github URL: https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/Interactive_Visual_Analytics_with_Plotly.py

Predictive Analysis (Classification)

Create a variable Y and standarize the variable assigned to X

Split in train and test database

Calculate logistic regression – support vector machine – decison tree and knn

Calculate the number and occurence of mission outcome per orbit type

Compare score accurancy to chose the best model

Github URL: https://github.com/Nur-Telu/IBM-Data-Science-Capstone-Project-SpaceX/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

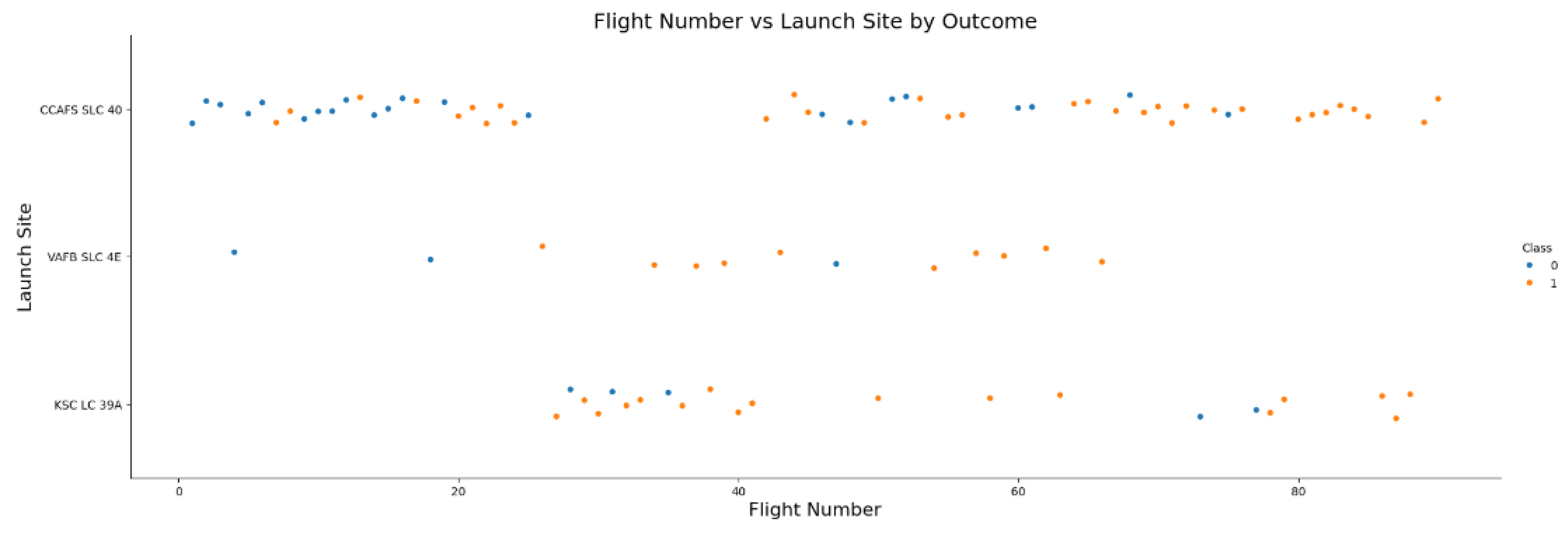
Results

- Exploratory data analysis results
  - Both API and web scraping are capable to collect Xspace data
- Interactive analytics demo in screenshots
  - EDA with SQL is effective for data filtering
  - EDA with interactive visualization provides informative information
  - Plotly Dash is powerful to show instant data change
- Predictive analysis results
  - Decision Tree Classifier Algorithm has the best accuracy of predicting

Section 2

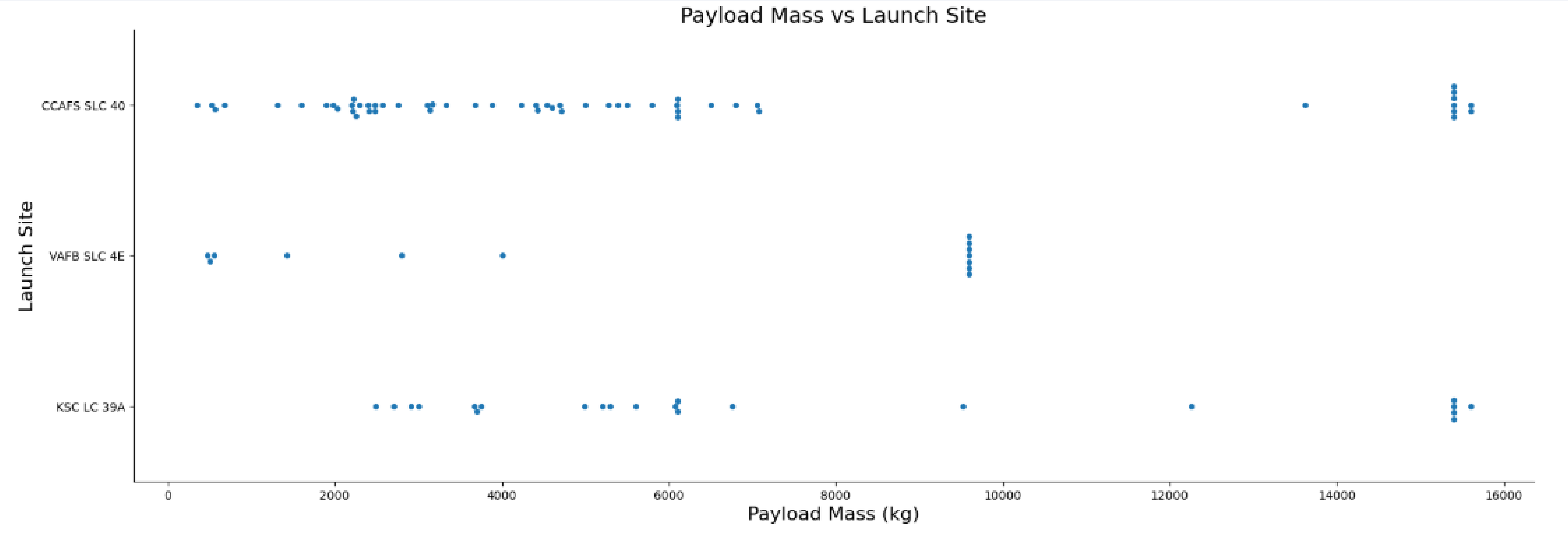
Insights drawn from EDA

Flight Number vs. Launch Site



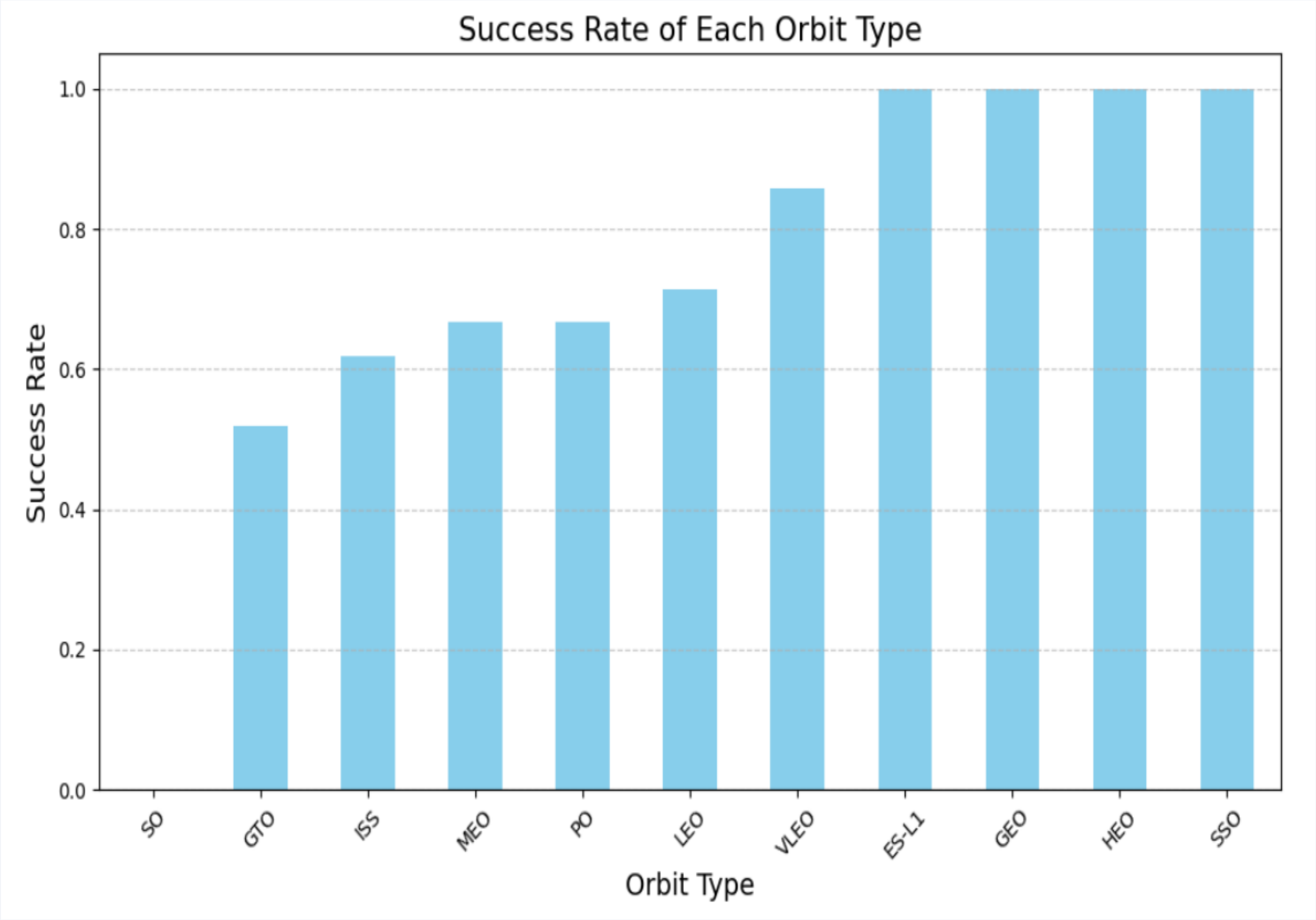
From the plot, we found that the larger the flight amount at a launch site, the greater the success rate at a launch site

Payload vs. Launch Site



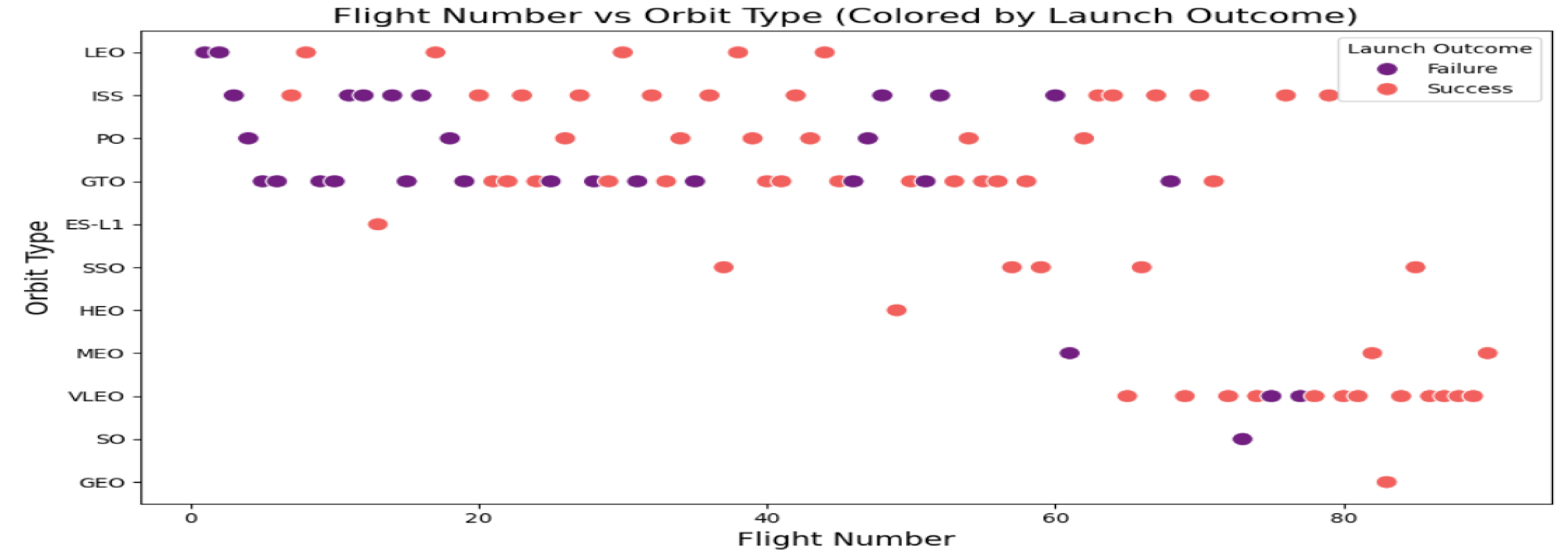
From the plot, we found that thePayloads with lower mass are have more launches compared to those with higher mass across all three launch sites

Success Rate vs. Orbit Type



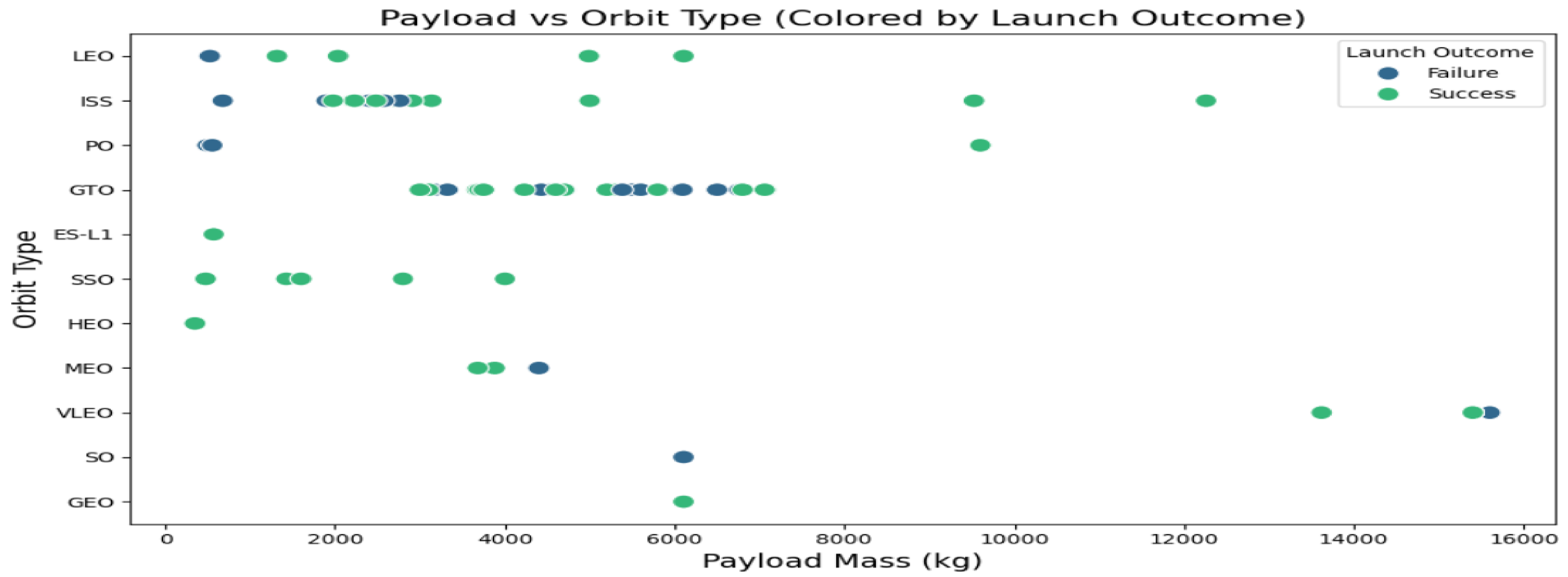
We can see that the orbits with the highest success rate are:SSO, HEO, GEO, ES-L1, while the SO Orbit it is the one with lowest rate

Flight Number vs. Orbit Type



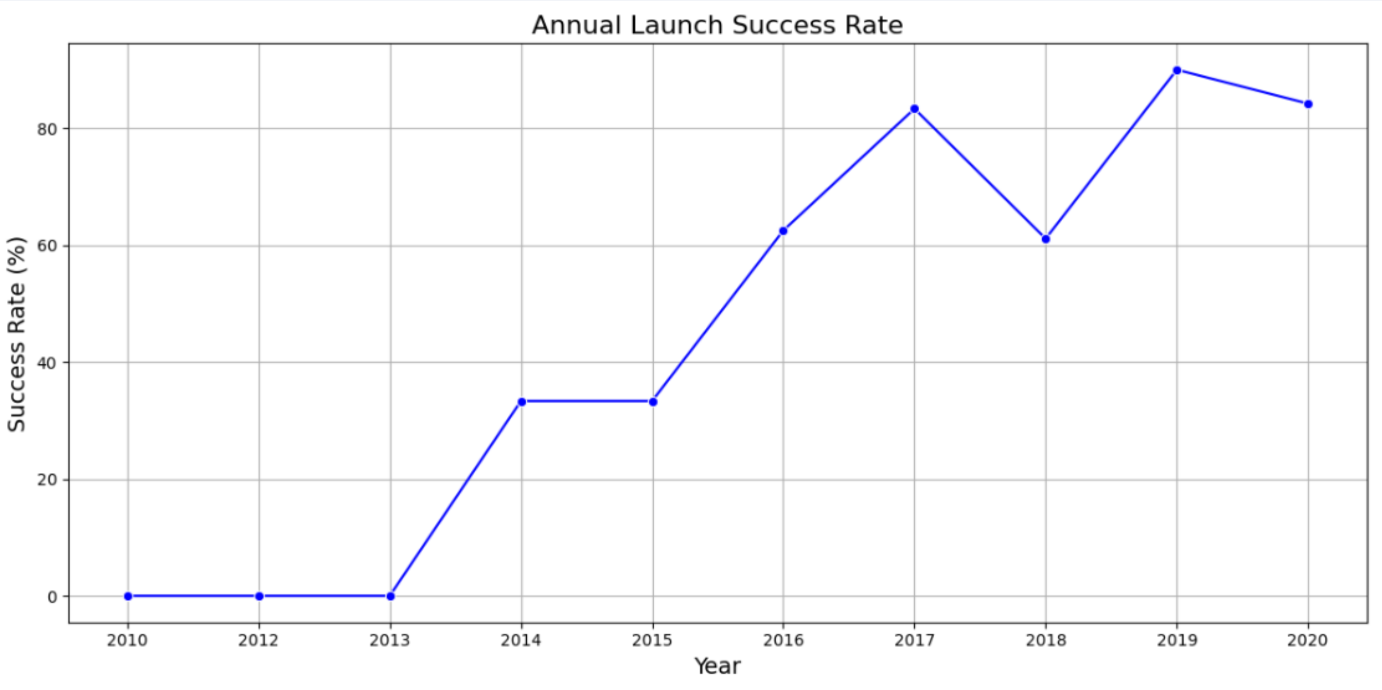
From the plot, we found that in the LEO orbit the Success appears related to the number of flights; on the other hand , there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



From the plot, we found that the Heavy payloads tend to have higher successful landing rates for PO, LEO, and ISS orbits, but for GTO orbit, success is less predictable with an almost equal mix of successes and failures.

Launch Success Yearly Trend



From the plot, we can observe that success rate since 2013 kept on increasing till 2020.

All Launch Site Names

Task 1

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

In [10]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

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

Out[10]:

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

Performed an SQL query to obtain all launch site names

Launch Site Names Begin with 'CCA'

Task 2

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

In [11]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

Out[11]:

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

Performed an SQL query to obtain 5 launch site names that begin with ‘CCA’

Total Payload Mass

Task 3

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

In [12]:
```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Customer" = 'NASA (CRS)';
```

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

Out[12]:

SUM("PAYLOAD_MASS__KG_")
45596

Performed an SQL query to obtain the total payload mass carried by boosters launched by NASA (CRS)

Average Payload Mass by F9 v1.1

Task 4

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

In [13]:
```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

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

Out[13]:

AVG("PAYLOAD_MASS__KG_")
2928.4

Performed an SQL query to calculate the average payload mass carried by booster version F9 v1.1

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 [14]:
```
%sql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

Out[14]:

MIN("Date")
2015-12-22

Performed an SQL query to find the dates of the first successful landing outcome on ground pad

Successful Drone Ship Landing with Payload between 4000 and 6000

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [15]:

```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS_
```

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

Out[15]:

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

Performed an SQL query to list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [16]:

```
%sql SELECT "Mission_Outcome", COUNT(*) AS "Total" FROM SPACEXTABLE WHERE "Mission_Outcome" IN ('Success', 'Failure') GROUP
```

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

Out[16]:

Mission_Outcome	Total
Success	98

Performed an SQL query to calculate the total number of successful and failure mission outcomes

Boosters Carried Maximum Payload

Task 8

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

In [17]:
```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SP
```

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

Out[17]:

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

Performed an SQL query to list the names of the booster which have carried the maximum payload mass

2015 Launch Records

Task 9

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, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

In [18]:

```
%%sql

SELECT
    CASE
        WHEN substr("Date", 6, 2) = '01' THEN 'January'
        WHEN substr("Date", 6, 2) = '02' THEN 'February'
        WHEN substr("Date", 6, 2) = '03' THEN 'March'
        WHEN substr("Date", 6, 2) = '04' THEN 'April'
        WHEN substr("Date", 6, 2) = '05' THEN 'May'
        WHEN substr("Date", 6, 2) = '06' THEN 'June'
        WHEN substr("Date", 6, 2) = '07' THEN 'July'
        WHEN substr("Date", 6, 2) = '08' THEN 'August'
        WHEN substr("Date", 6, 2) = '09' THEN 'September'
        WHEN substr("Date", 6, 2) = '10' THEN 'October'
        WHEN substr("Date", 6, 2) = '11' THEN 'November'
        WHEN substr("Date", 6, 2) = '12' THEN 'December'
        ELSE 'Unknown'
    END AS "Month_Name",
    "Mission_Outcome",
    "Booster_Version",
    "Launch_Site"
FROM
    SPACEXTABLE
WHERE
    substr("Date", 0, 5) = '2015';
```

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

Out[18]:

Month_Name	Mission_Outcome	Booster_Version	Launch_Site
January	Success	F9 v1.1 B1012	CCAFS LC-40
February	Success	F9 v1.1 B1013	CCAFS LC-40
March	Success	F9 v1.1 B1014	CCAFS LC-40
April	Success	F9 v1.1 B1015	CCAFS LC-40
April	Success	F9 v1.1 B1016	CCAFS LC-40
June	Failure (in flight)	F9 v1.1 B1018	CCAFS LC-40
December	Success	F9 FT B1019	CCAFS LC-40

Performed an SQL query to list the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015

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

Task 10

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 [19]:

```
%%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.
```

Out[19]:

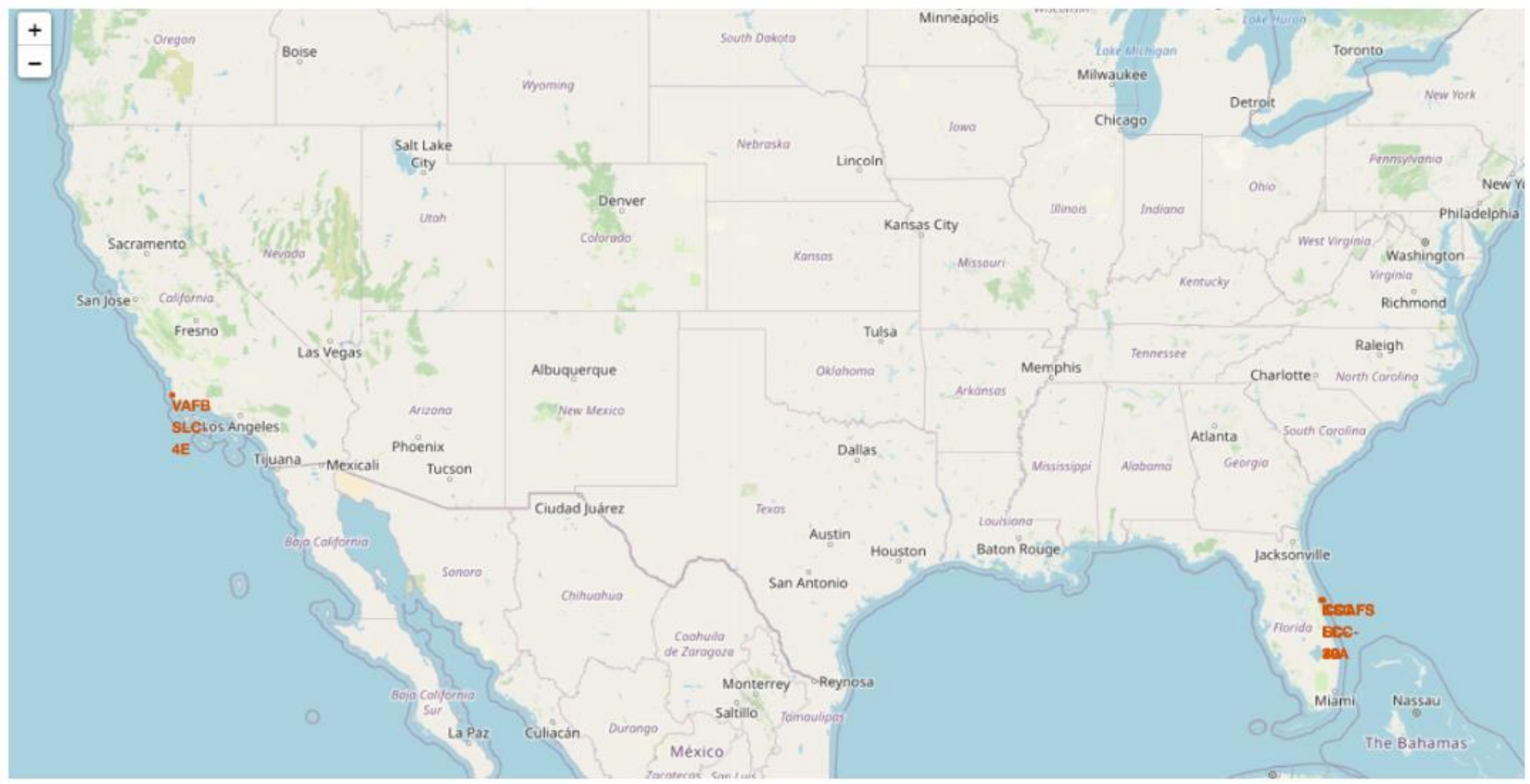
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Performed an SQL query to 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

Section 3

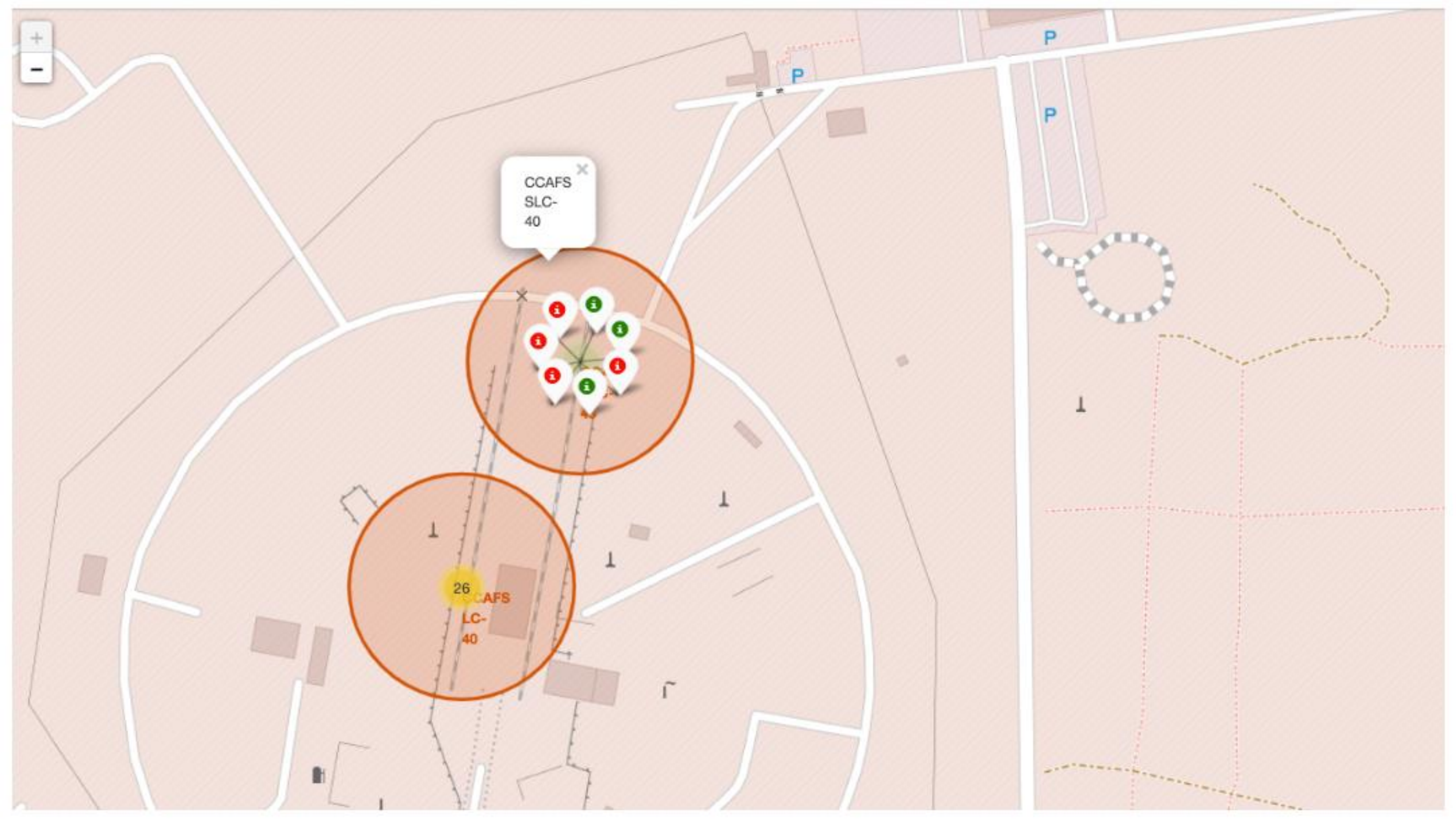
Launch Sites Proximities Analysis

All launch sites' location markers on global map



The launch sites are labelled by a marker with their names on the map.

All success/failed launches for each site on the map



The launch records are grouped in clusters on the map, then labelled by green markers for successful launches, and red markers for unsuccessful ones.

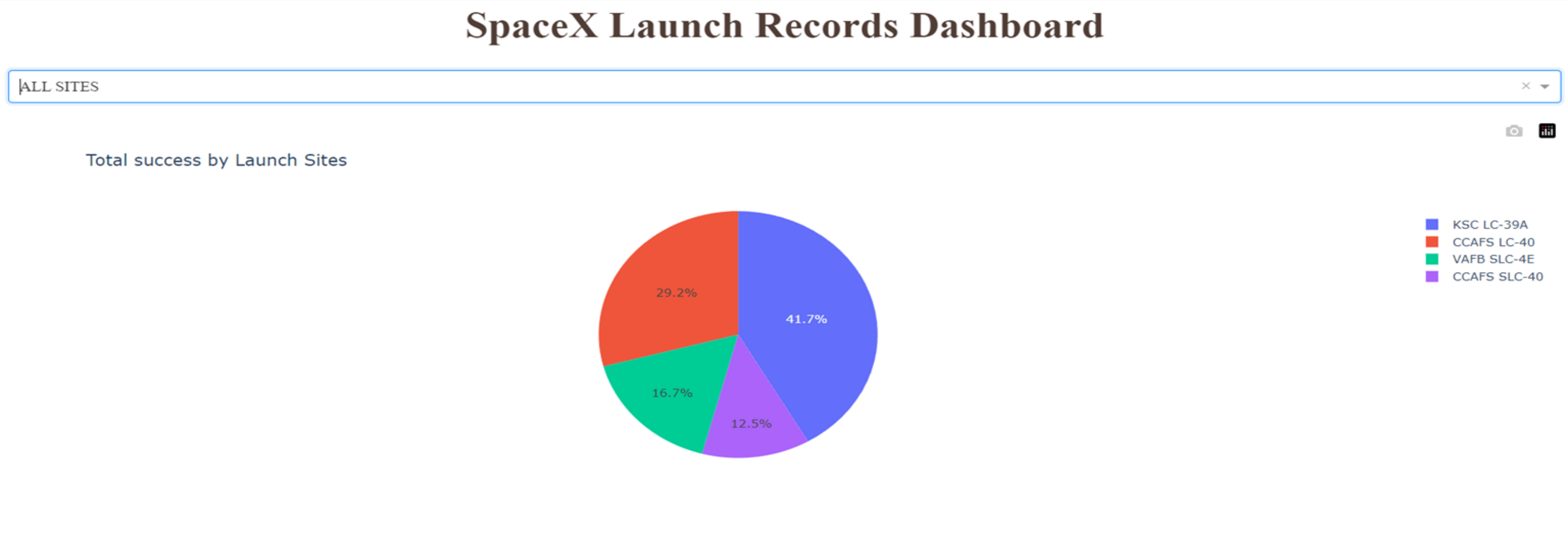
Launch Site Proximity Mapping: Railway, Highway, Coast, City

The closest coastline from CCAFS SLC-40 is marked as a point using MousePosition and the distance between the coastline point and the launch site, which is approximately 0.9 km.

Section 4

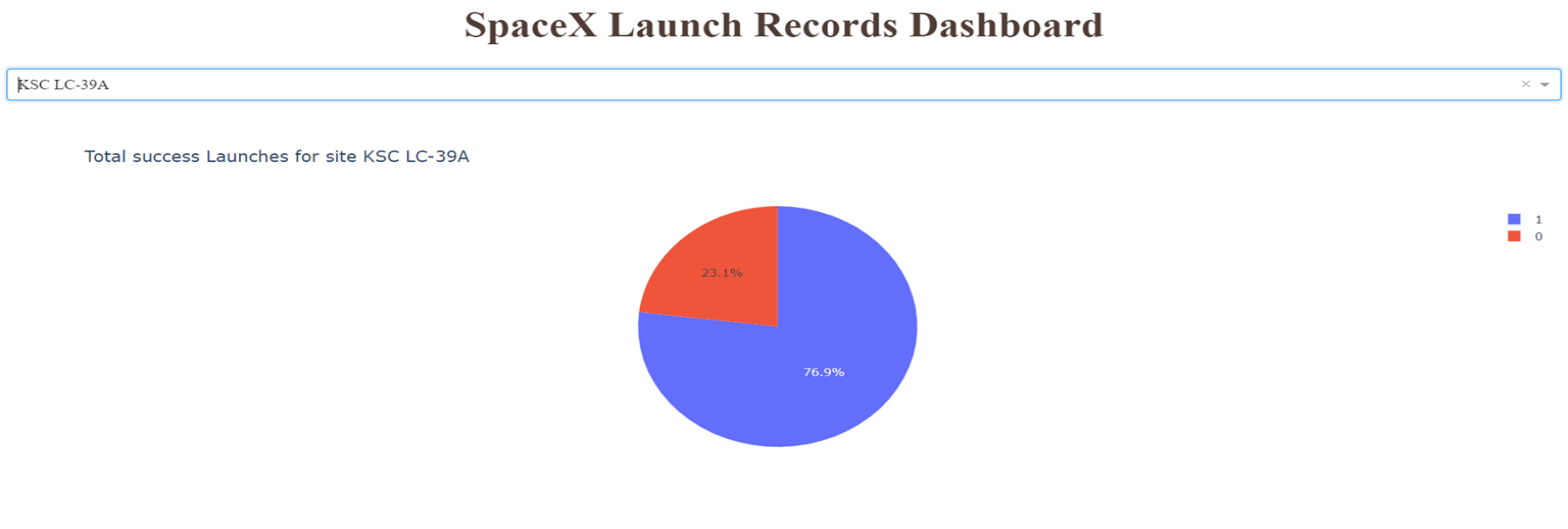
Build a Dashboard with Plotly Dash

Most successful launches for all sites



KSC LC-39A has the highest amount of success launches with 41.7% from the entire record, whereas CCAFS SLC-40 has the lowest amount of success launches with only 12.5%.

Highest success ratio of the launch site



KSC LC-39A which is the launch site with highest amount of success, has a 76.9% success rate for the launches from its site, and 23.1% failure rate.

Payload vs launch outcome

- The payload range that has the highest success launches is between 2,000 to 4,000 kg, which can be seen by the highest number of plots in that range, followed by the payload range of 4,000 to 6,000 kg, with the second highest number of plots.
- Booster version FT (green spots) has the highest success launches, followed by B4 (purple spots) with the second highest success launches, among all booster versions

Section 5

Predictive Analysis (Classification)

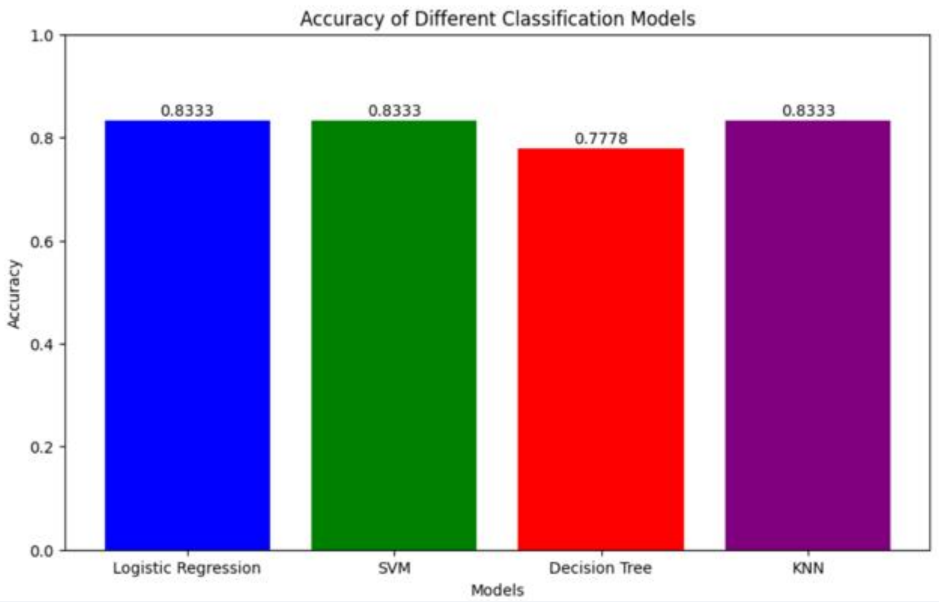
Classification Accuracy

TASK 12

Find the method performs best:

In [35]:

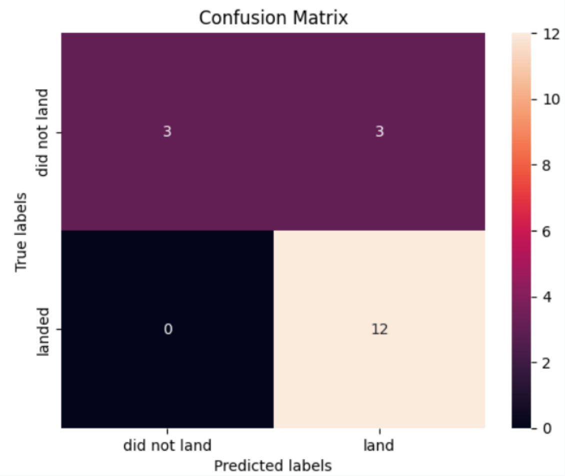
```
import matplotlib.pyplot as plt

# Model names and their corresponding accuracies
models = ['Logistic Regression', 'SVM', 'Decision Tree', 'KNN']
accuracies = [0.8333, 0.8333, 0.7778, 0.8333]

# Create a bar chart
plt.figure(figsize=(10, 6))
plt.bar(models, accuracies, color=['blue', 'green', 'red', 'purple'])

# Add title and labels
plt.title('Accuracy of Different Classification Models')
plt.xlabel('Models')
plt.ylabel('Accuracy')
plt.ylim(0, 1)  # Setting the y-axis range from 0 to 1

# Add accuracy values on top of the bars
for i, accuracy in enumerate(accuracies):
    plt.text(i, accuracy + 0.01, f'{accuracy:.4f}', ha='center')

# Show the plot
plt.show()
```



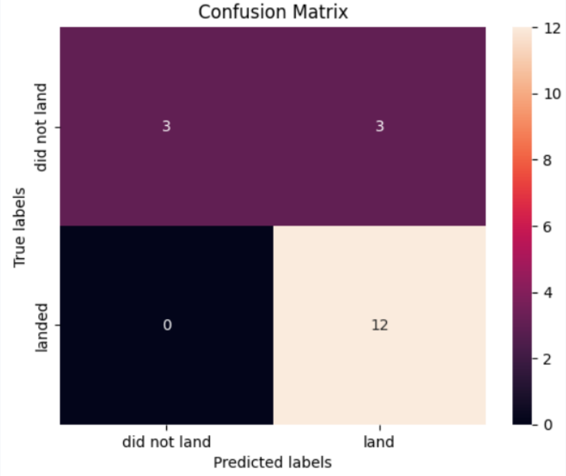
The model that performed best are Logistic Regression, SVM, KNN where all 3 achieved the highest accuracy of 83.33%.

Confusion Matrix

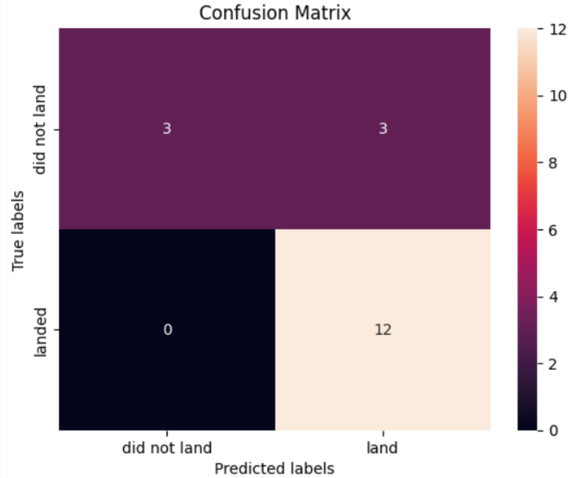
Logistic Regression



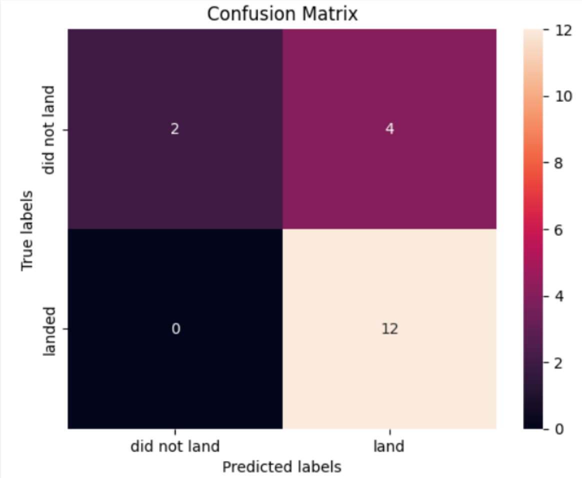
SVM



KNN



Decision Tree



- LR, SVM, KNN, and Decision Tree models are good as their confusion matrix show that they predicted all 12 successful landing correctly, with 0 error
- However, the Decision Tree model only predicted 2 out of 6 not successful landing correctly, with 4 of them wrongly predicted as successful landing.

Conclusions

- LR, SVM, KNN are top-performing models for forecasting outcomes in this data
- GEO,HEO,SSO,ES L1 orbit types exhibit the highest rates of successful launches
- Success rates for spaceX launches has been increasing relatively with time and it looks like soon they will reach the required target
- The success of a mission can be explained by several factors such as the launch site, the orbit and especially the number of previous launches. Indeed, we can assume that there has been a gain in knowledge between launches that allowed to go from a launch failure to a success
- Depending on the orbits, the payload mass can be a criterion to take into account for the success of a mission. Some orbits require a light or heavy payload mass. But generally low weighted payloads perform better than the heavy weighted payloads.

Thank you!