IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<SUDARSHAN>
<APRIL 2025>

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

Summary of methodologies

- Data collection
- Data wrangling
- EDA with SQL
- Building an interactive map with Folium
- EDA with visualization
- Predictive analysis using classification models
- Building a Dashboard with Plotly Dash

Summary of all results •

- Data analysis (EDA) using SQL
- Interactive visual analytics using Folium
- Predictive analysis using classification models
- EDA with Data Visualization.
- Build a Dashboard with Plotly Dash

INTRODUCTION

Project Background and Context

- The commercial space industry is rapidly expanding, driven by companies like SpaceX.
- SpaceX's Falcon 9, with a reusable first stage, has significantly lowered launch costs.
- Traditional rocket launches cost around $165 million, while Falcon 9 missions cost about $62 million due to reusability.
- Success of landings depends on multiple factors such as payload mass, orbit, and mission objectives.
- This project takes the role of SpaceY data scientists to analyze historical launch data and predict first stage recovery using machine learning.

Problems Statment

- Predict the success of Falcon 9 first-stage landings using historical launch data.
- Identify key influencing factors such as payload mass, launch site, and orbit type.
- Estimate overall launch costs based on predicted outcomes.
- Develop an interactive dashboard for mission planning and data-driven decision-making.

Methodology

Data Collection Methodology

- Collected data using the SpaceX REST API and web scraping from Wikipedia.

Data Wrangling

- Applied One Hot Encoding to prepare data for binary classification tasks.

Exploratory Data Analysis (EDA)

- Conducted EDA using visualizations and SQL queries to uncover patterns and insights.

Interactive Visual Analytics

- Created interactive visualizations using **Folium** and **Plotly Dash** to explorel and launch-related data.

Predictive Analysis

- Built and evaluated multiple classification models.
- **Perform predictive analysis using classification models Logistic Regression, K-Nearest Neighbors, Support Vector Machines, and Decision Tree models.**

Data Collection

The data had been collected using two methods. The first method involved using the SpaceX API to gather information about past launches. A request was sent to the API, and the response data was filtered to include only Falcon 9 launches. It was also noted that for some secret missions with missing payload weights, the average payload weight from other missions was used to fill in the gaps.

The second method described was web scraping. That data had been taken from the Wikipedia page listing Falcon 9 and Falcon Heavy launches. The table on that page was accessed to extract column names, and the data was then converted into a Pandas DataFrame to prepare it for analysis.

Data Collection – SpaceX API

The data was requested and parsed from the SpaceX API using a GET request. The DataFrame was then filtered to include only Falcon 9 launches needed for the project analysis. After that, the data was cleaned by removing missing values in the Payload Mass column. Finally, the cleaned data was stored and organized into a new dictionary containing only the relevant columns.

GIT HUB LINK--https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/Data%20Collection%20SPACEX%20API.ipynb

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	Reused
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	

Data Collection – Web Scraping

The data was requested from Wikipedia using an HTTP GET request along withBeautifulSoup for parsing. All columnor variable names were then extracted from the HTML table header. An empty dictionary was created using these extracted column names, and the column data was cleaned and added to the dictionary. Finally, a DataFrame was created by parsing the launch HTML tables into a structured format suitable for analysis

GIT HUB LINK https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/Data%20Collectioon%20With%20Web%20Scrapping.ipynb

- .

[36]: df.head()

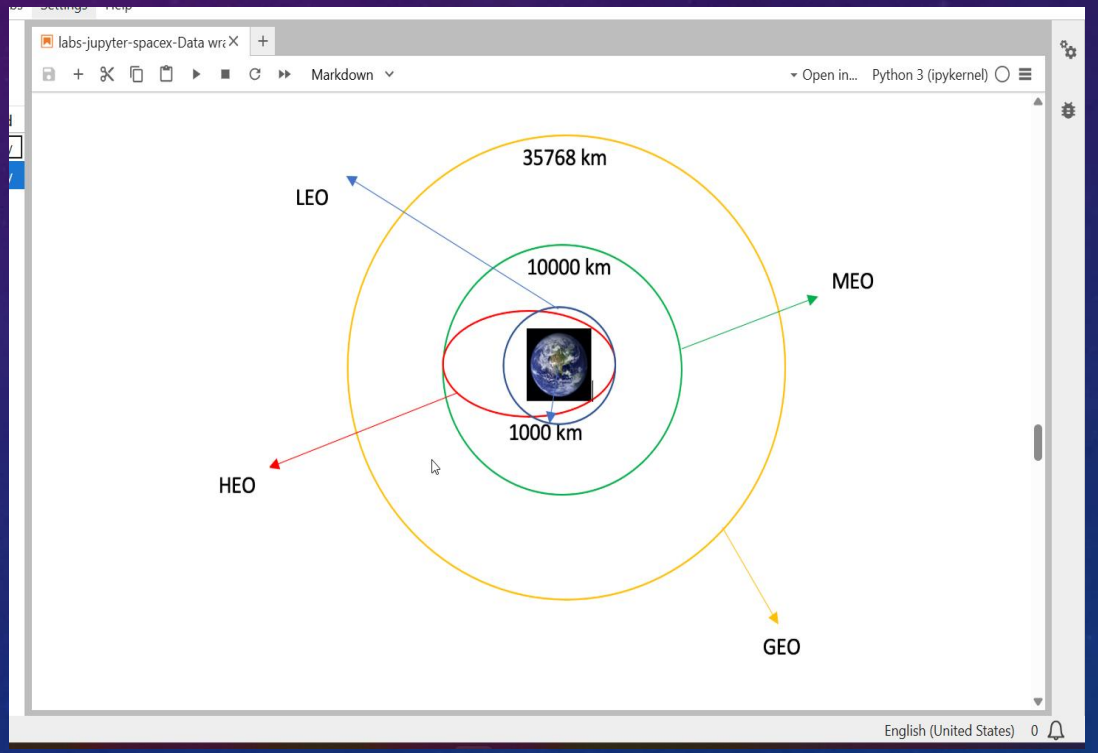
[36]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.07B0003.18	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.07B0004.18	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.07B0005.18	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.07B0006.18	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.07B0007.18	No attempt\n	1 March 2013	15:10

Data Wrangling

In this lab, the first step was to load the SpaceX launch data and understand the structure of various columns, particularly the landing outcomes. The second step involved filtering and cleaning the data, including handling missing values and irrelevant columns to improve data quality. Then, a new landing outcome label was created where 1 indicates a successful landing and 0 indicates failure, by analyzing the Landing_Outcome field. Next, the dataset was normalized and converted into a numerical format to be suitable for machine learning models. Finally, exploratory data analysis (EDA) was conducted to identify useful patterns and features that can help predict future landings.

GIT HUB LINK- https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/Data%20Wrangling.ipynb



EDA with SQL

SQL queries are used to display unique launch sites, filter records based on conditions like launch sites starting with "CCA," and calculate total or average payload mass for specific mission types or booster versions.

Queries identify the first successful landing outcome on the ground pad and list boosters with successful drone ship landings and payload masses within a specific range.

They also count the total number of successful and failed mission outcomes to assess mission performance.

Subqueries are utilized to find booster versions that carried the maximum payload mass across missions.

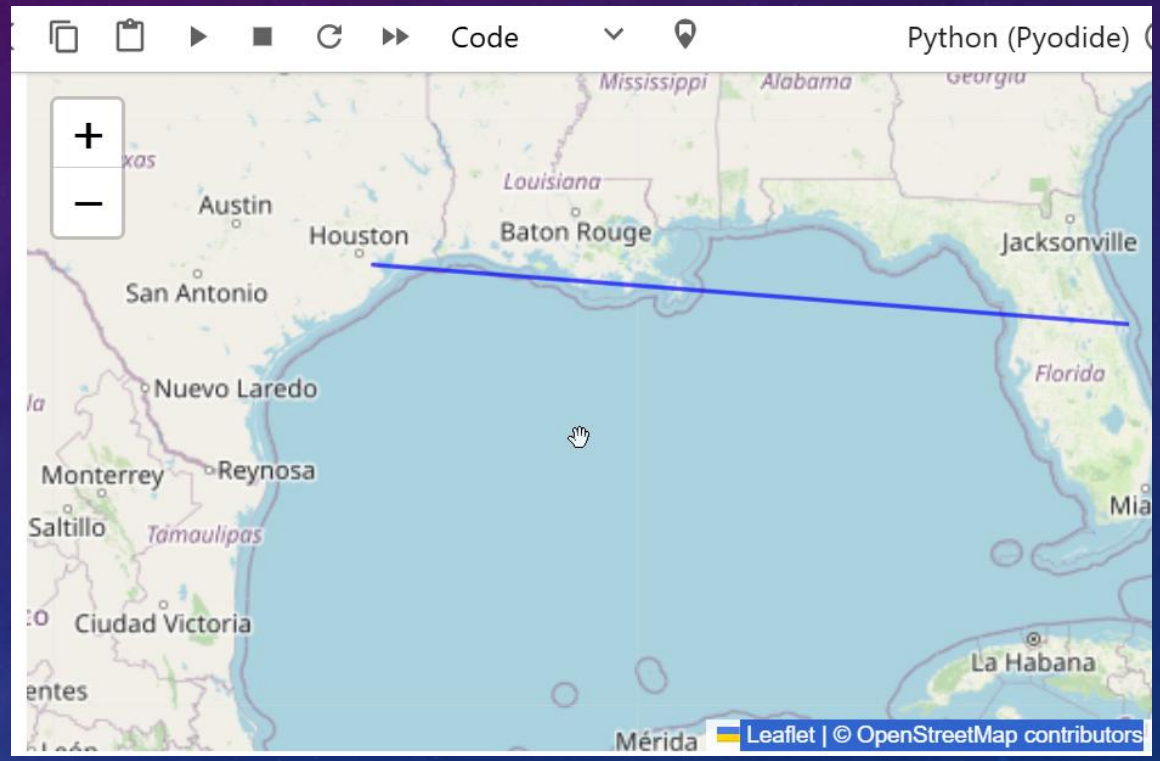
Additionally, queries filter data for the year 2015 to display monthly failure outcomes, booster versions, and launch sites, and rank landing outcomes by success or failure between specified dates.

GIT HUB LINK- https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/EDA%20With%20SQL.ipynb

Build an Interactive Map with Folium

An interactive map with Folium was created to visualize SpaceX launch sites. Markers pinpoint the exact geographical locations of the launch sites, providing spatial reference. Circles were added around each site to represent proximity zones, visually indicating safety or operational boundaries. Lines connect launch sites to relevant locations, highlighting spatial relationships and dependencies. This setup helps users understand the operational context and geographic connections of SpaceX launches.

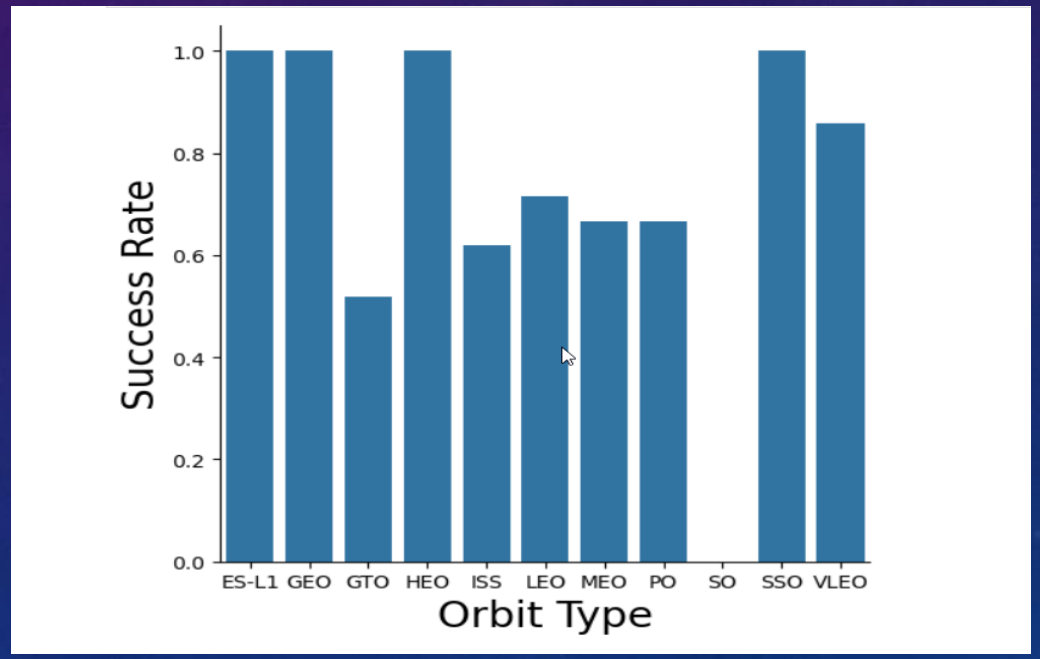
GIT HUB LINK ---https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/Interactive%20Visual%20Analytics%20With%20Folium.ipynb



EDA with data visualization

The charts plotted include 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, and Success Rate Yearly Trend. Scatter plots are used to show the relationship between variables, and if a connection exists, these could potentially be utilized in machine learning models. Bar charts are designed to compare discrete categories, aiming to highlight the relationship between specific categories and their associated measured values. Line charts, on the other hand, illustrate trends over time, making them ideal for time series data analysis. Each chart type serves a unique purpose depending on the data's nature and the insights you aim to extract from it.

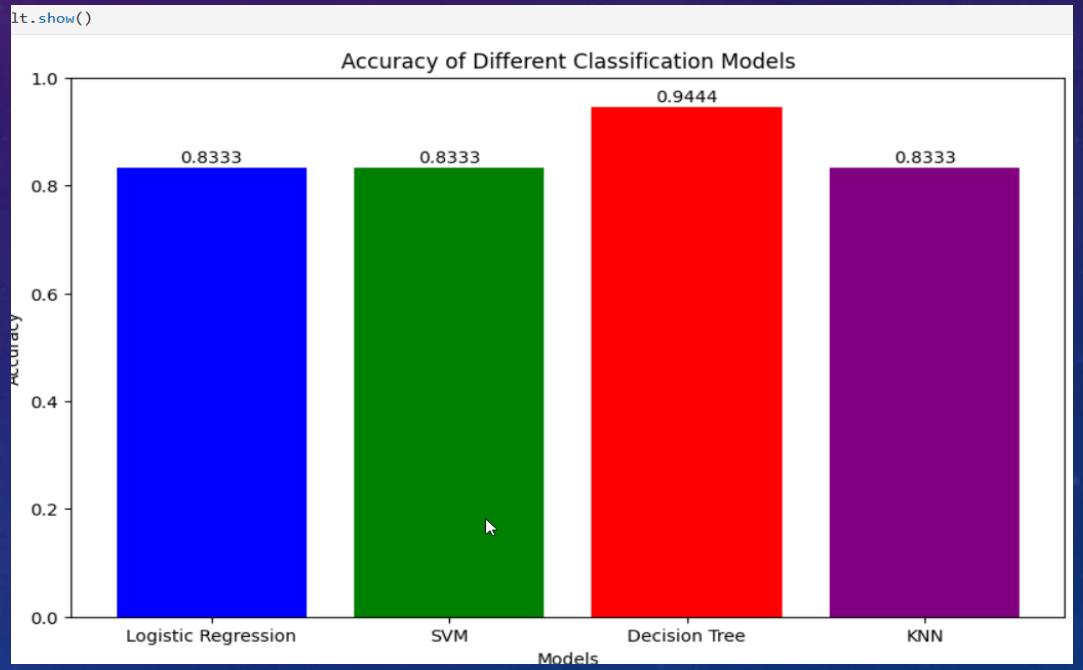
GIT HUB LINK-https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/EDA%20WITH%20DATA%20VISUALISATION.ipynb



Predictive analysis using classification models

The “Class” column from the dataset was converted into a NumPy array for model training. The data was standardized using StandardScaler, followed by fitting and transforming to ensure consistent scaling across features. The dataset was then split into training and testing sets using the train_test_split function. Multiple models, including Logistic Regression, SVM, Decision Tree, and KNN, were evaluated using Jaccard Score, F1 Score, and confusion matrices to identify the best-performing method. A GridSearchCV object with cv=10 was applied to each model to optimize hyperparameters, and accuracy on the test data was calculated using the .score() method.

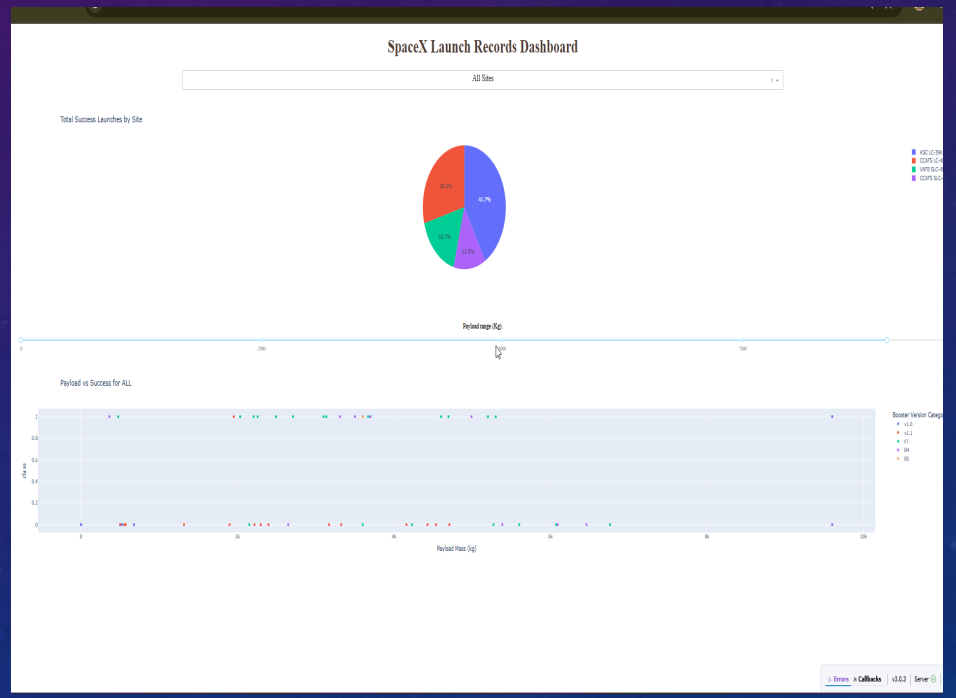
Git hub link -https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/Machine%20Learning%20Prediction.ipynb



Build a Dashboard with Plotly Dash

A dropdown menu was added to allow users to select a specific launch site for analysis. A pie chart was included to display the total number of successful launches across all sites, or the success versus failure count when a particular site is selected. A payload mass slider was implemented to let users filter data based on a selected payload range. A scatter chart was created to show the relationship between payload mass and launch success rate. This chart also helps compare different booster versions and their performance based on payload.

GIT HUB LINK- https://github.com/soun369/Applied-Data-Science-Capstone-Project/blob/main/SPACEDASH.py



Results

- **Data analysis (EDA) using SQL**
- **Interactive visual analytics using Folium**
- **Predictive analysis using classification models**
- **EDA with Data Visualization.**
- Build a Dashboard with Plotly Dash

DATA ANALYSIS USING EDA WITH SQL

All Launch Site Names

Task 1

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

[33]:
```
sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL ORDER BY 1;
```

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

[33]:

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

Launch Site Names Begin with 'CCA'

```
[34]: sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

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

[34]:

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

Task 3

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

[44]:
```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Customer" = 'NASA (CRS)';
```

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

[44]:

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1 AND First Successful Ground Landing Date

45596

Task 4

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

[46]:
```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

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

[46]:

AVG(PAYLOAD_MASS__KG_)
2928.4

Task 5

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

Hint:Use min function

[48]:
```
%sql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

[48]:

MIN(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Total Number of Successful and Failure Mission Outcomes

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

[50]:
```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS_
```

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

[50]:

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

Task 7

List the total number of successful and failure mission outcomes

[52]:
```
%sql SELECT "Mission_Outcome", COUNT(*) AS "Total" FROM SPACEXTABLE WHERE "Mission_Outcome" IN ('Success', 'Failure') GROUP BY "Mission_Outcome";
```

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

[52]:

Mission_Outcome	Total
Success	98

Total Number of Successful and Failure Mission Outcomes

Task 8

List all the booster_versions that have carried the maximum payload mass. Use a subquery.

[54]:
```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE);
```

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

[54]:

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

Boosters Carried Maximum Payload

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

[56]:

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

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

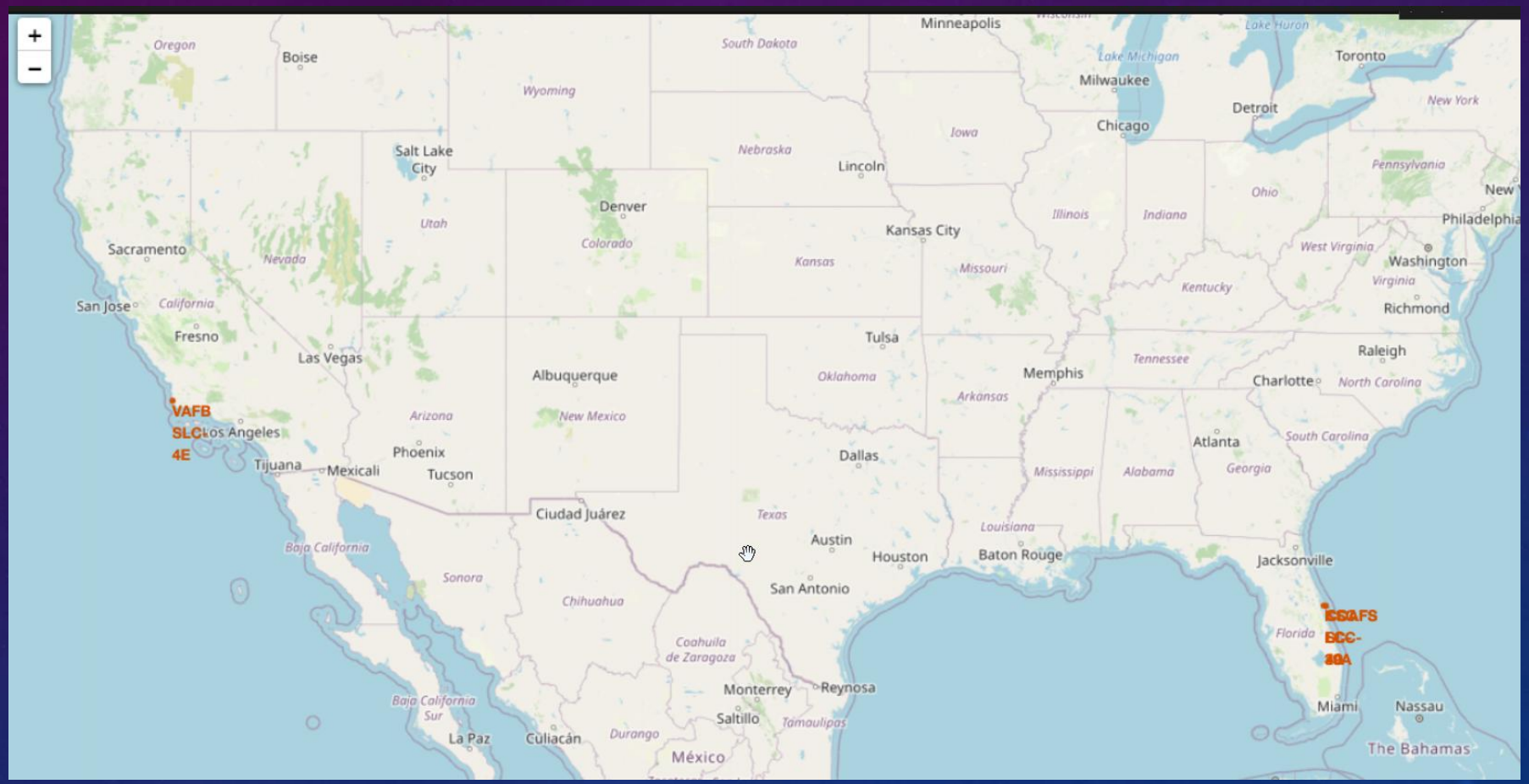
```
 * sqlite:///my_data1.db
Done.
```

[58]:

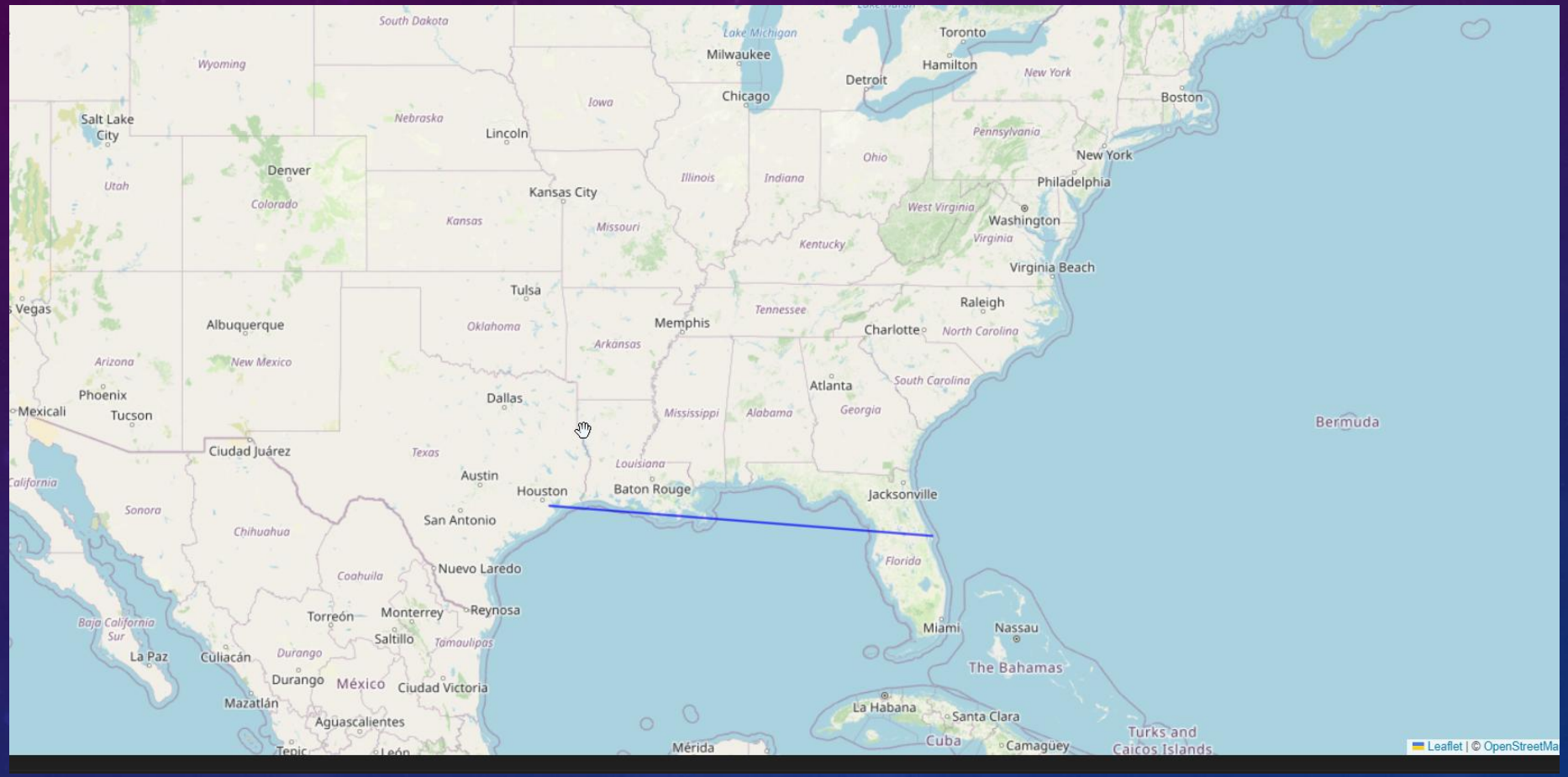
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Interactive visual analytics using Folium

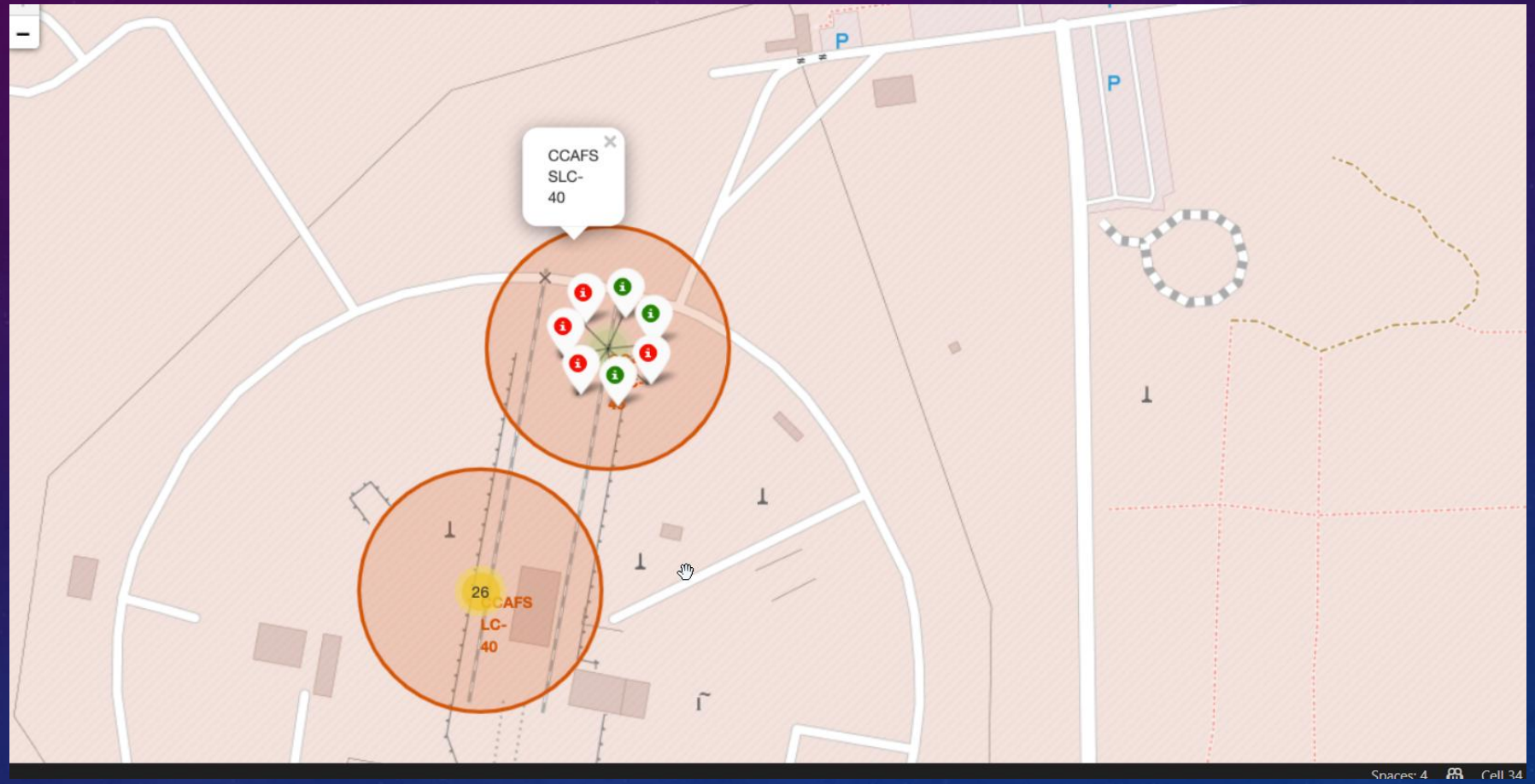
All launch sites on a map



Distances between a launch site to its proximities

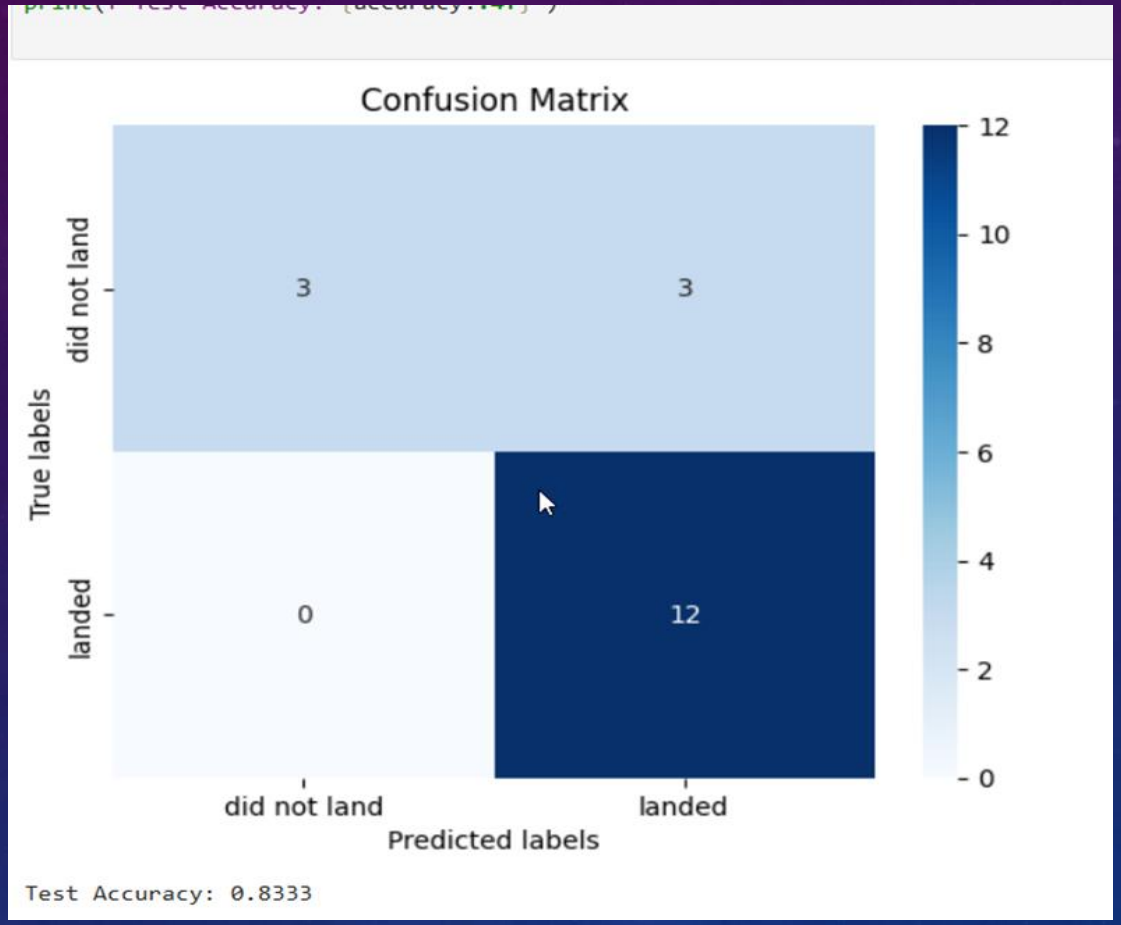


All success/failed launches for each site on the map

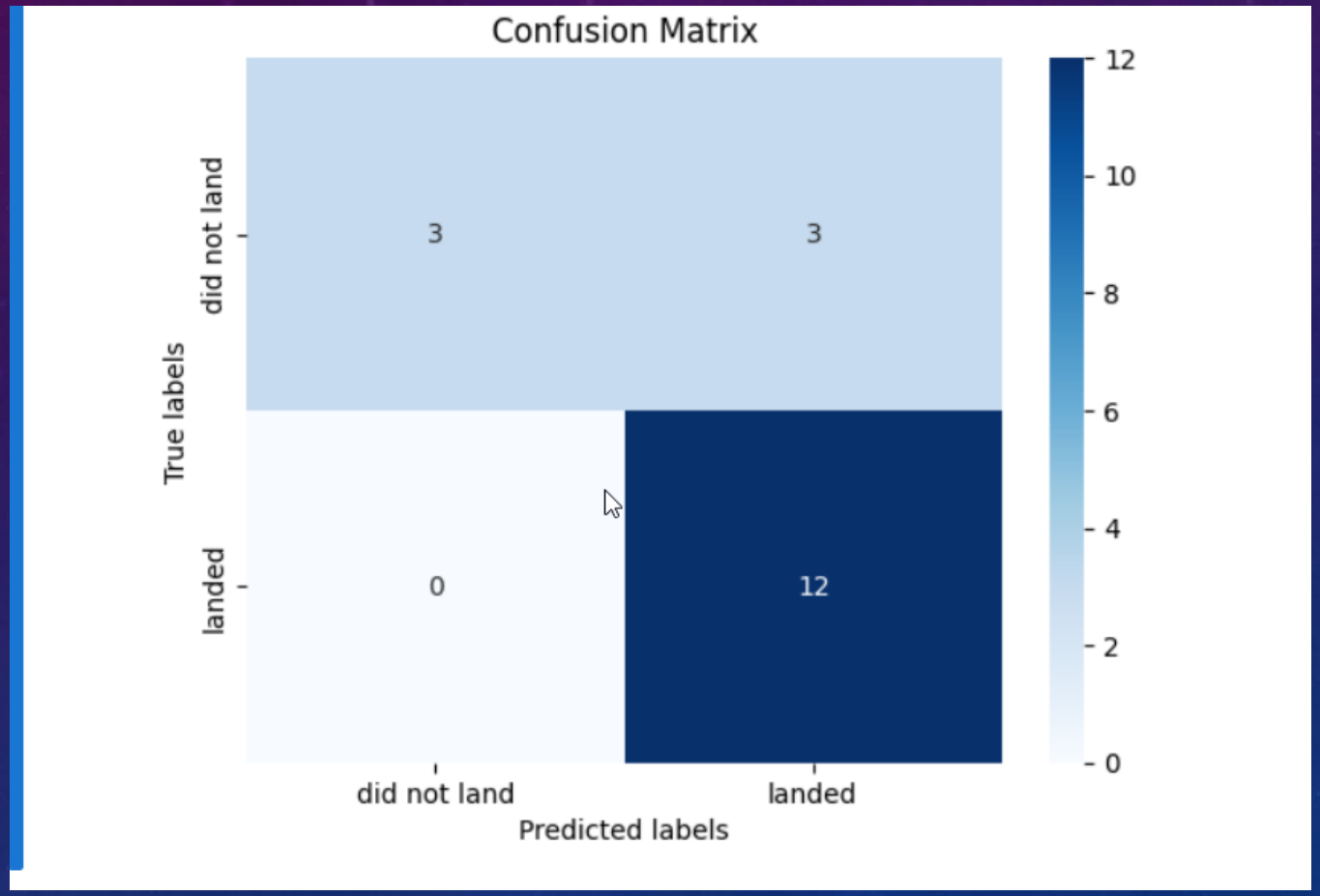


Predictive analysis using classification models

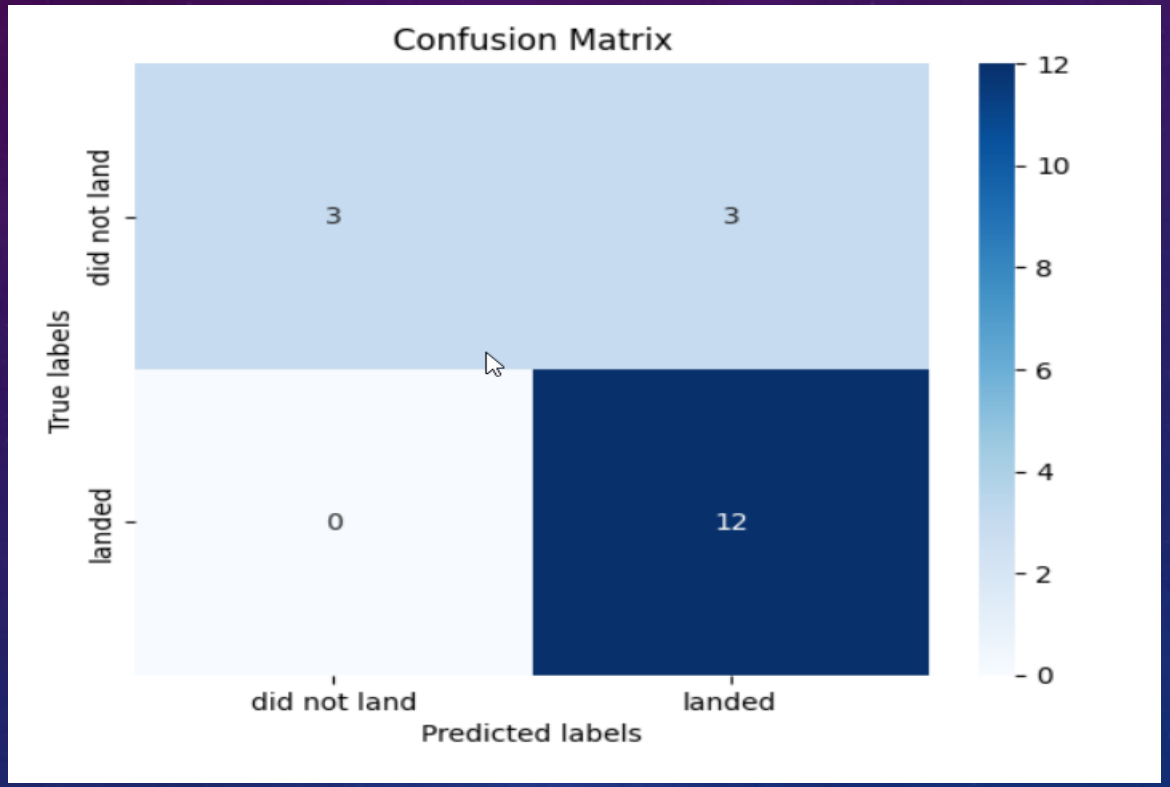
LOGICAL REGRESSION



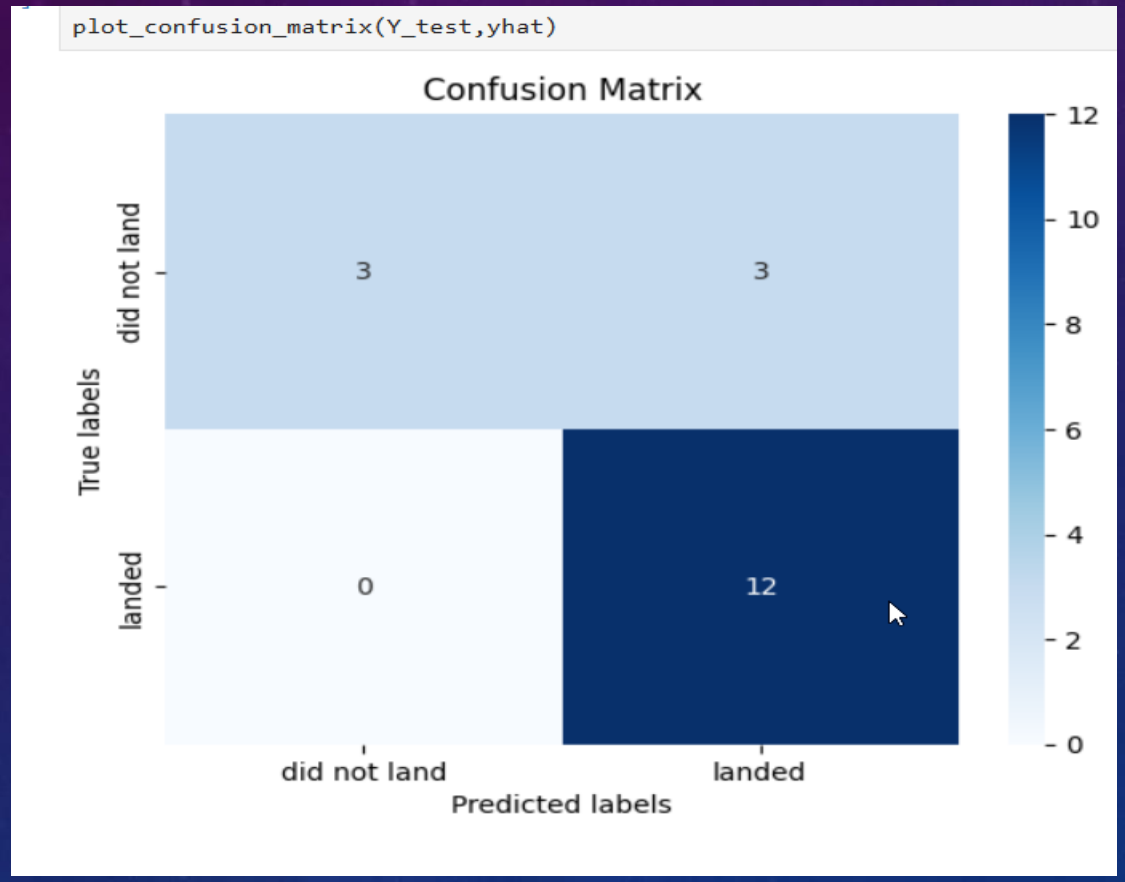
Confusion Matrix of SVM



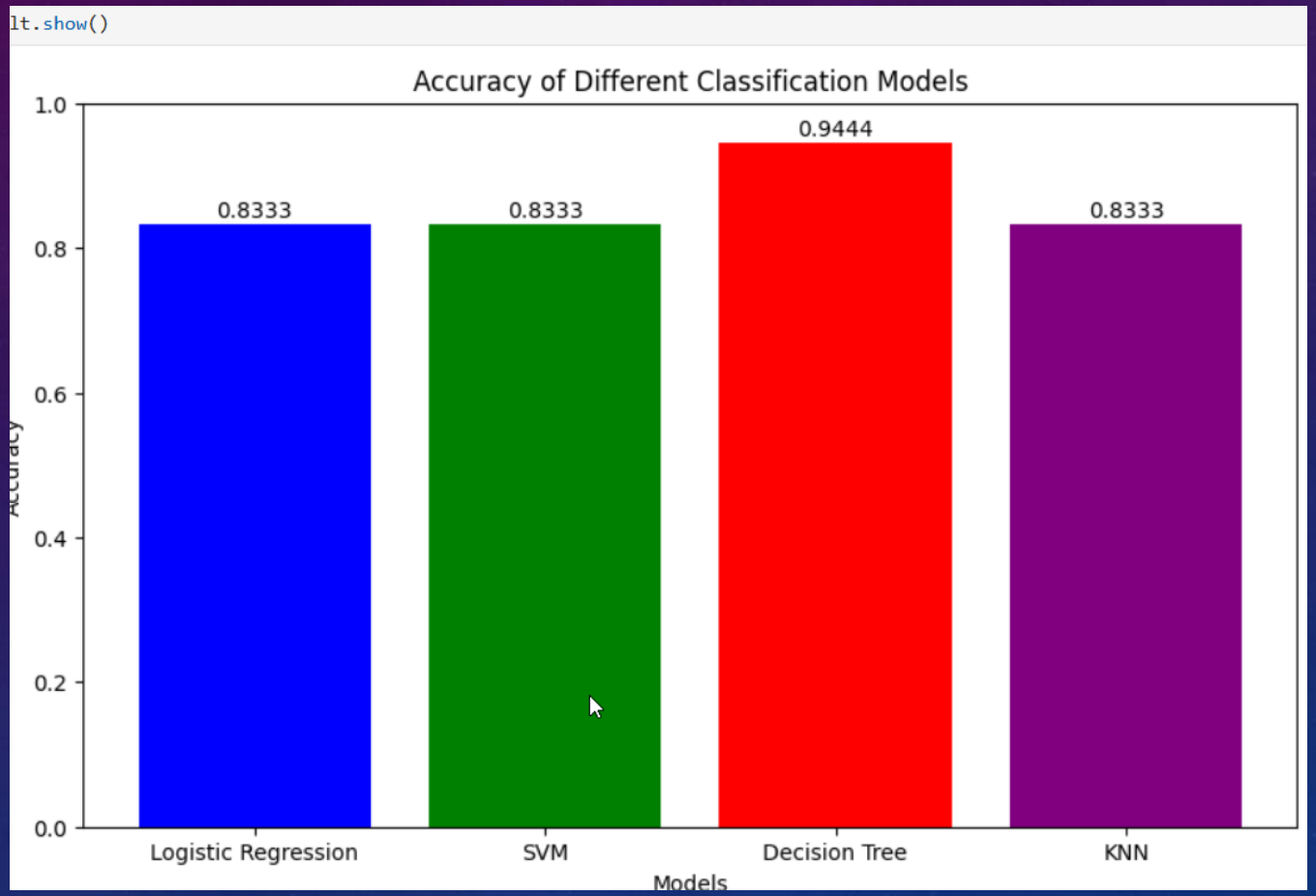
Confusion Matrix of Decision Tree



Confession Matrix of Kneighbors Classifer



ACCURACY OF DIFFFERENCE CLASSIFICATON MODELS



EDA with Data Visualization.

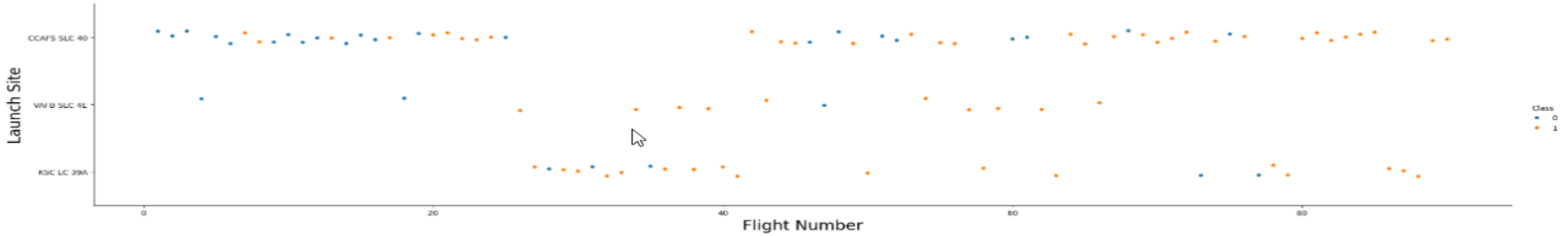
Flight Number vs. Launch Site

Next, let's drill down to each site visualize its detailed launch records.

TASK 1: Visualize the relationship between Flight Number and Launch Site

Use the function `catplot` to plot `FlightNumber` vs `LaunchSite`, set the parameter `x` parameter to `FlightNumber`,set the `y` to `Launch Site` and set the parameter `hue` to `'class'`

[7]:
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(x='FlightNumber', y='LaunchSite', hue='Class', data=df, aspect=5)
plt.xlabel('Flight Number', fontsize=20)
plt.ylabel('Launch Site', fontsize=20)
plt.show()
```



Now try to explain the patterns you found in the Flight Number vs. Launch Site scatter point plots.

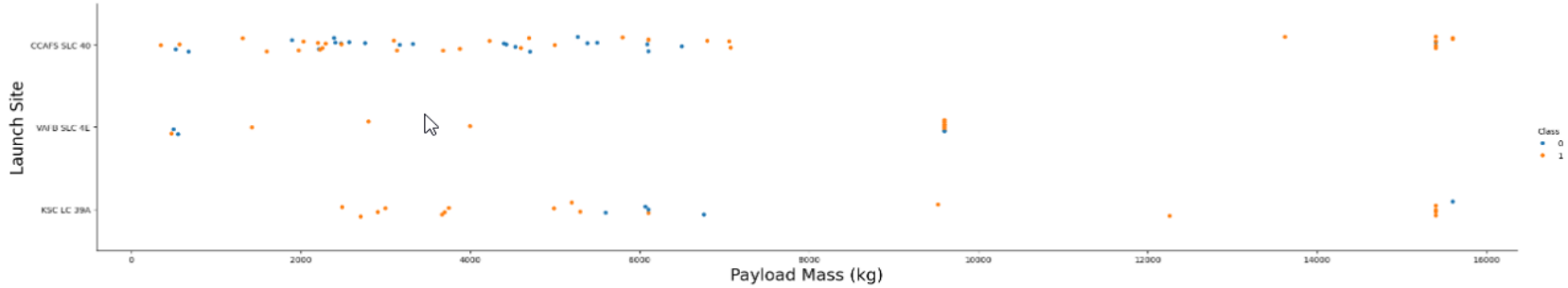
TASK 2: Visualize the relationship between Payload Mass and Launch Site

Payload vs. Launch Site

TASK 2: Visualize the relationship between Payload Mass and Launch Site

We also want to observe if there is any relationship between launch sites and their payload mass.

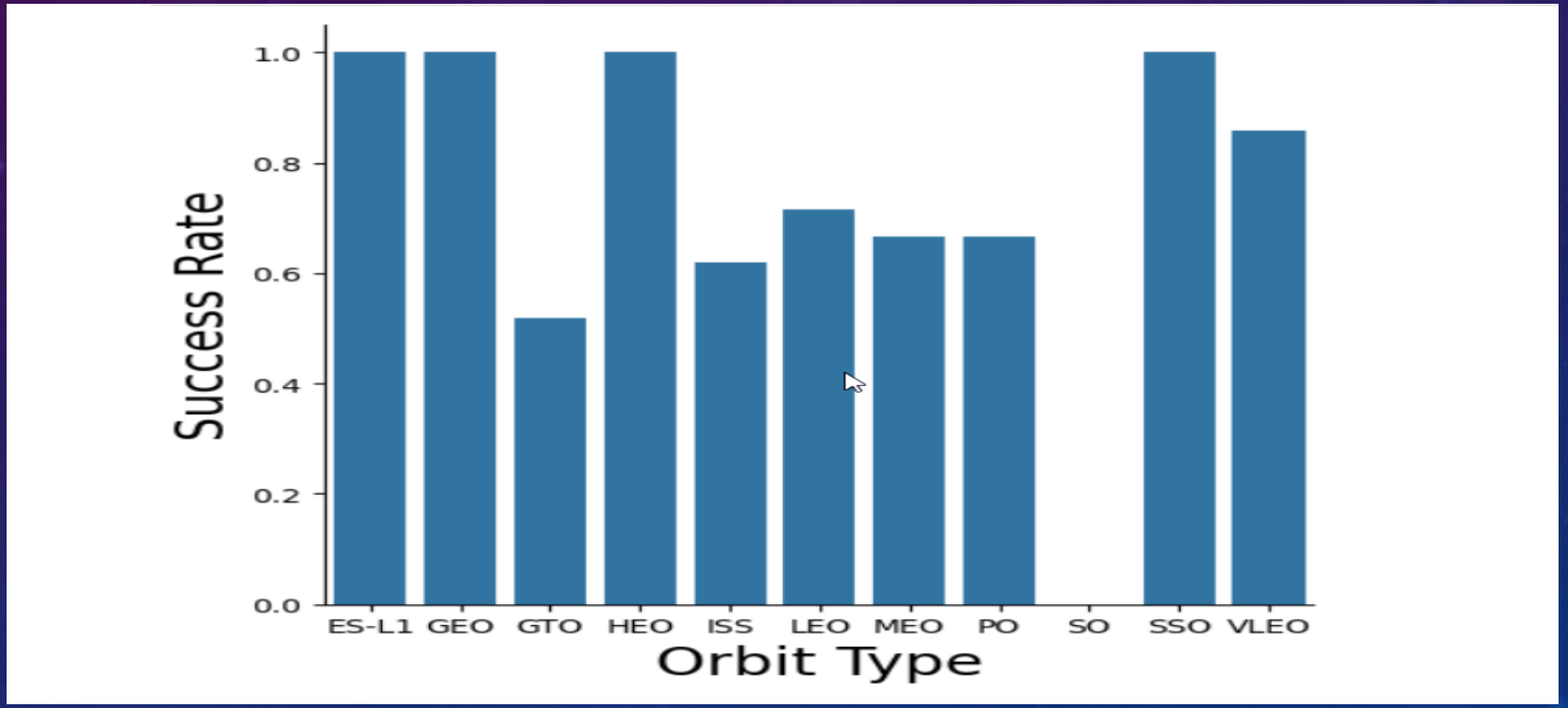
[8]:
```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
sns.catplot(x='PayloadMass', y='LaunchSite', hue='Class', data=df, aspect = 5)
plt.xlabel('Payload Mass (kg)',fontsize=20)
plt.ylabel('Launch Site',fontsize=20)
plt.show()# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
```



Now if you observe Payload Mass Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

TASK 3: Visualize the relationship between success rate of each orbit type

SUCCESS RATE VS ORBIT TYPE

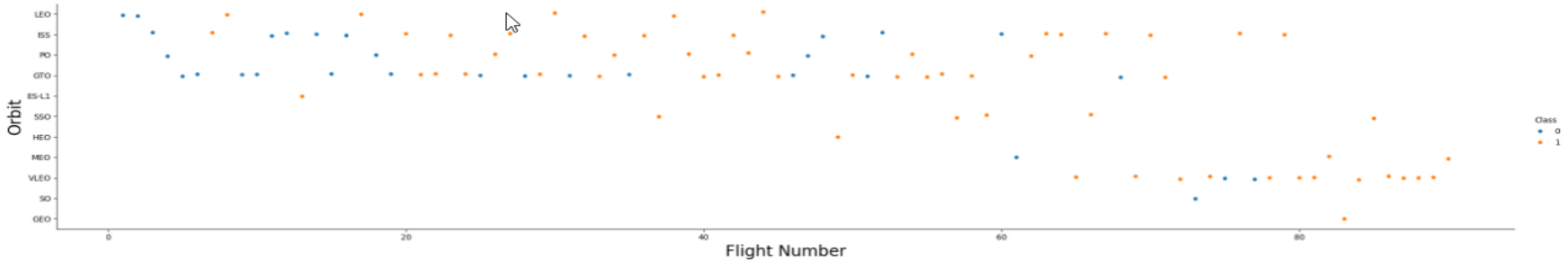


Flight Number vs. Orbit Type

TASK 4: Visualize the relationship between FlightNumber and Orbit type

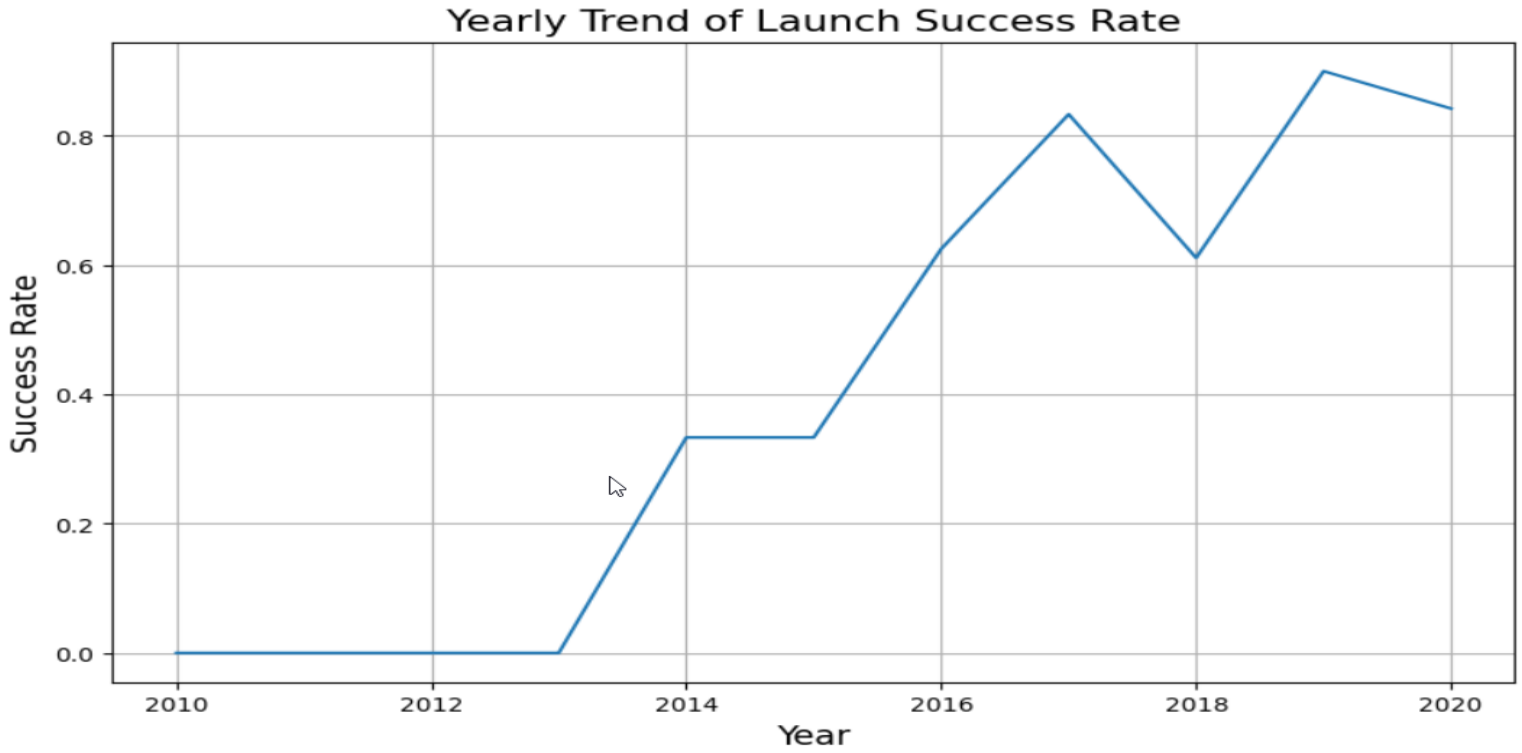
For each orbit, we want to see if there is any relationship between FlightNumber and Orbit type.

[10]:
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(x = 'FlightNumber', y = 'Orbit', hue = 'Class', data = df, aspect = 5)
plt.xlabel('Flight Number', fontsize = 20)
plt.ylabel('Orbit', fontsize = 20)
plt.show()
```



You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

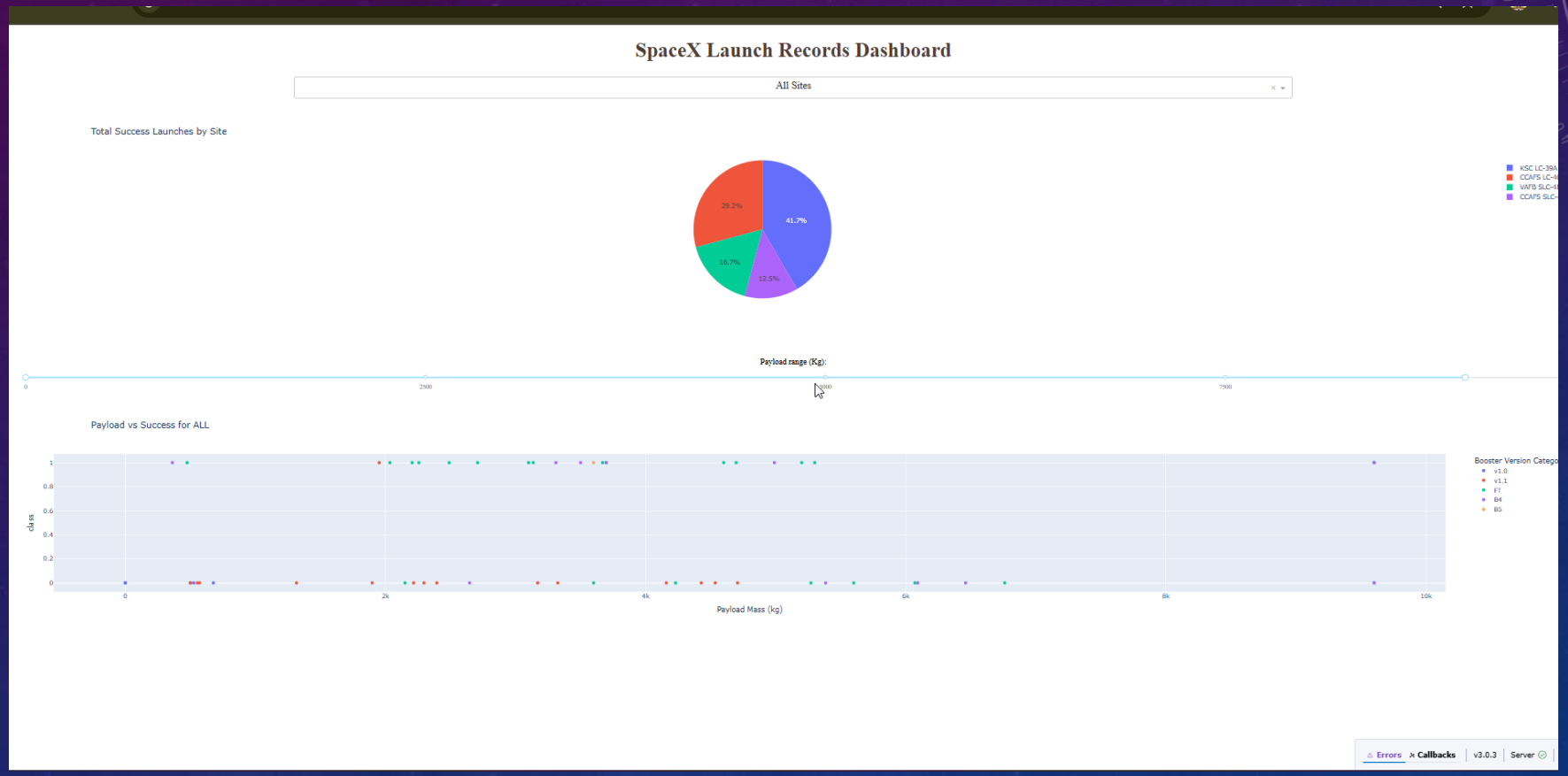
Launch Success Yearly Trend



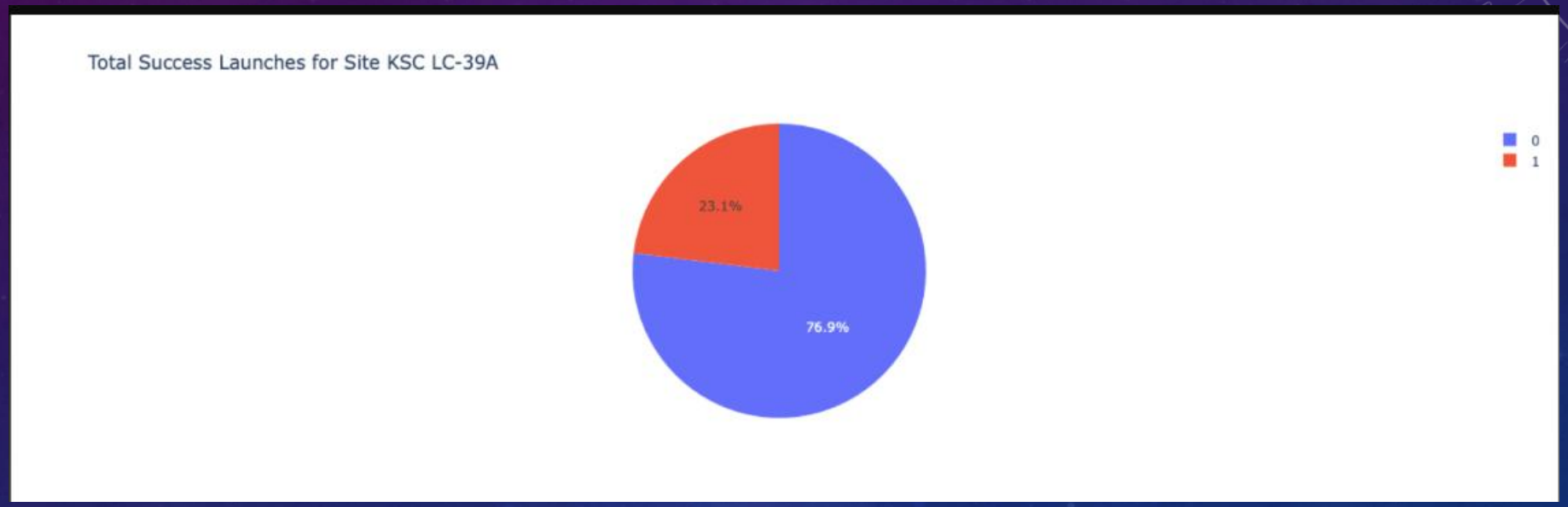
you can observe that the sucess rate since 2013 kept increasing till 2020

Build a Dashboard with Plotly Dash

Launch success count for all site



Success ratio of the launch site with the highest success launch



Conclusions

This dataset revealed that Logistic Regression (LR), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) are highly effective machine learning models for predictive analysis, as they consistently delivered accurate results

Launch performance improves when the payload mass is low, compared to missions with high payload mass.

The majority of launch sites are located near the Equator and all are situated close to the coastline.

Among all launch sites, KSC LC-39A records the highest success rate.

The highest success rates have been observed in missions to GEO, HEO, SSO, and ES-L1 orbits.

THANK YOU