

# Winning Space Race with Data Science

Aruna S P  
05.11.2024



# Outline

---

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

# Executive Summary

---

- This project addresses the challenge of optimizing payload pricing for rocket launches, a crucial factor in reducing overall mission costs and enhancing the viability of space missions. By examining SpaceY's Falcon 9 launch data, we aim to develop a data-driven model that predicts optimal payload pricing while minimizing financial losses associated with launch risks and failures.
- Using advanced data science techniques, including machine learning and statistical analysis, our approach analyzes historical launch data to identify patterns that impact launch success and pricing efficiency. Key features such as payload mass, launch location, and mission type are considered to predict the price point at which profitability and success likelihood are maximized.
- The outcome of this analysis provides an optimized pricing strategy that aligns with risk mitigation, supporting more cost-effective and reliable space missions. This model has potential applications across the aerospace industry, contributing to more sustainable practices in mission planning and budgeting.

# Introduction

---

- In this Data Science project, **Space Y** aims to determine the optimal launch site for the Falcon 9 rocket to enhance operational efficiency and reduce costs. By analyzing launch data collected from SpaceX's websites, we created a comprehensive dataset, which was meticulously cleaned and preprocessed, including the removal of null values.
- Through data exploration and model optimization, we refined key metrics to meet the project's objectives, enabling more informed decision-making for future launches and improved cost-effectiveness for Space Y's Falcon 9 operations.

Section 1

# Methodology



# Methodology

---

## Executive Summary

- Data collection methodology:
  - The data is collected from their official website using API and converted it into a dataframe to proceed further.
- Perform data wrangling
  - Removed the null values in the payload and converted the type's of the column from categorical to numerical to train the model
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Using various classification algorithms, we build a model to predict the results based on the requirements.

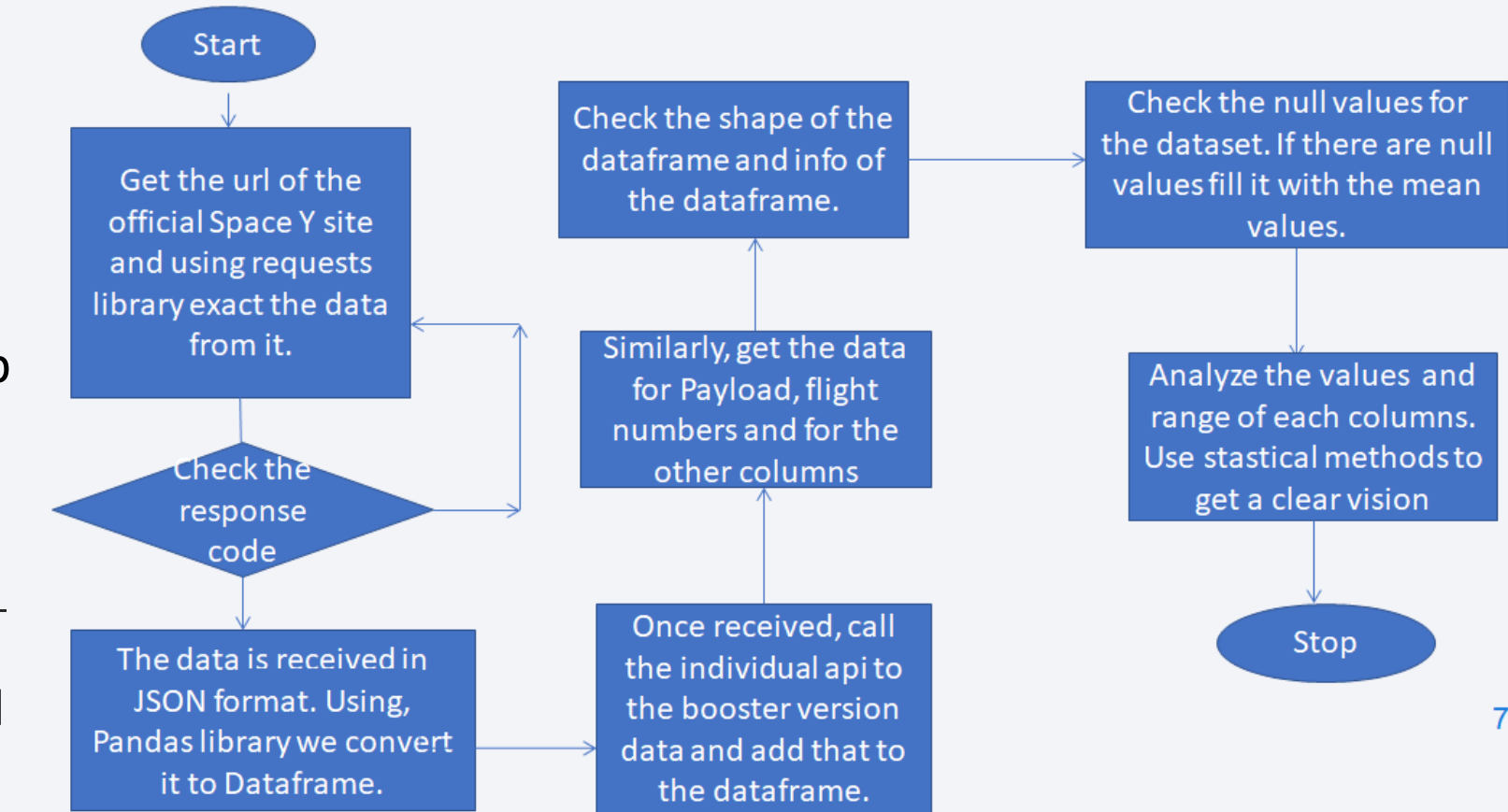
# Data Collection

---

In the following slides, we will see how we collect the data for this project and perform EDA based on the requirements.

# Data Collection – SpaceX API

- The data is obtained from the following url,” [https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API\\_call\\_spacex\\_api.json](https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json)”
- The GitHub Url for this section, [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/jupyter\\_labs\\_spacex\\_data\\_collection\\_api.ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/jupyter_labs_spacex_data_collection_api.ipynb)

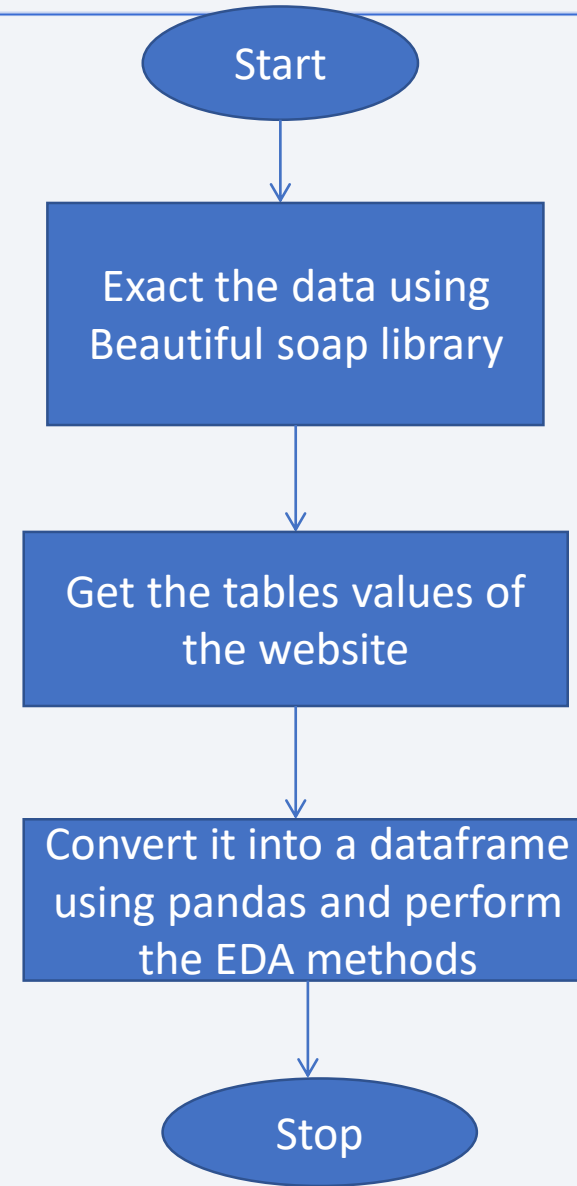




# Data Collection - Scraping

---

- The webscraping method is done using beautiful soap library.
- The GitHub URL of the completed web scraping notebook,  
[https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/jupyter-labs-webscraping%20\(1\).ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/jupyter-labs-webscraping%20(1).ipynb)



# Data Wrangling

---

- Once the raw data obtained, we have to clean, transform and organize it for further process.
- Removed the null and duplicate values.
- Standardize or normalize the numerical data.
- Convert the categorical data using one hot encoding.
- [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb)

# EDA with Data Visualization

---

- In the exploratory data analysis (EDA) phase, data visualization plays a crucial role in understanding the distribution, relationships, and patterns within the dataset.
- Generally for univariate variables, we apply bar chart, histogram charts,etc., For bivariate variables, we apply line charts, scatter plots,etc.,
- [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/edadataviz.ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/edadataviz.ipynb)

# EDA with SQL

---

- Using SQL queries, we got the distinct launch sites. Top 5 records that has cca in its launch site name.
- Obtained the success and failure rates and its counts. Landing Outcomes Between 2010-06-04 and 2017-03-20.
- Got the max, average, sum of payloads using Sql built-in functions.
- First landing date and success rate for the payload between 4000 and 6000 mass/kg , etc.,
- [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/jupyter\\_labs\\_eda\\_sql\\_coursera\\_sqlite.ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/jupyter_labs_eda_sql_coursera_sqlite.ipynb)

# Build an Interactive Map with Folium

---

- Using Folium built the interactive maps which are discussed more detailed in the upcoming slides.
- Generated the global map with the launch sites on it.
- Differentiated success and failure launch sites with the built-in color marker.
- Calculated the distance between the coordinates.
- [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/lab\\_jupyter\\_launch\\_site\\_location.ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/lab_jupyter_launch_site_location.ipynb)

# Build a Dashboard with Plotly Dash

---

- Further analyzed using plotly dash libraries.
- Created a dynamic map for the better understanding.
- Generated a pie chart for each launch sites with it success and failure counts.
- Created a scatter plot to understand which launch sites had been used for which amount of payloads.
- [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/spacex\\_launch\\_dash.py](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/spacex_launch_dash.py)



# Predictive Analysis (Classification)

---

- We trained and tested our model using various classification techniques.
- To evaluate the model's performance, we calculated several metrics, including accuracy, F1 score, recall, and precision, providing a comprehensive assessment of the model's effectiveness.
- Additionally, a confusion matrix was generated to give a clearer visualization of true and false predictions, further enhancing our understanding of the model's classification capabilities.
- [https://github.com/Aruna2897/IBM\\_Data\\_science\\_course\\_assignments/blob/main/SpaceX\\_Machine%20Learning%20Prediction\\_Part\\_5.ipynb](https://github.com/Aruna2897/IBM_Data_science_course_assignments/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb)

# Results

---

- **Data Collection and wrangling:** The real time data is obtained from its source. Then, it is cleaned, transformed and organized based on the requirements. Here, we separated the falcon 9 data alone from the rest of the data.
- **Using EDA methods:** The variation between the variables and its flows has been observed. Removed the null and duplicate values for better model efficiency.
- **Using Folium Maps** – The following insights are observed
- **Proximity to Transportation Infrastructure:** Launch sites are located close to NASA railroads, providing cost-effective transportation of rocket parts to the launch area, enhancing logistical efficiency.
- **Strategic Coastal Positioning Near the Equator:** Sites are positioned near the coastline close to the equator. This location is ideal for launching rockets due to the Earth's rotational speed, which provides an additional velocity boost, reducing the fuel required to reach orbit. In case of a malfunction, rockets can be safely diverted to the ocean, minimizing risks to populated areas.



The background of the slide is an abstract composition. It features a solid blue area on the left side, which transitions into a dynamic pattern of diagonal streaks in shades of blue, red, and cyan on the right. A fine, light-colored grid or mesh pattern is overlaid across the entire image, particularly visible in the blue and cyan areas.

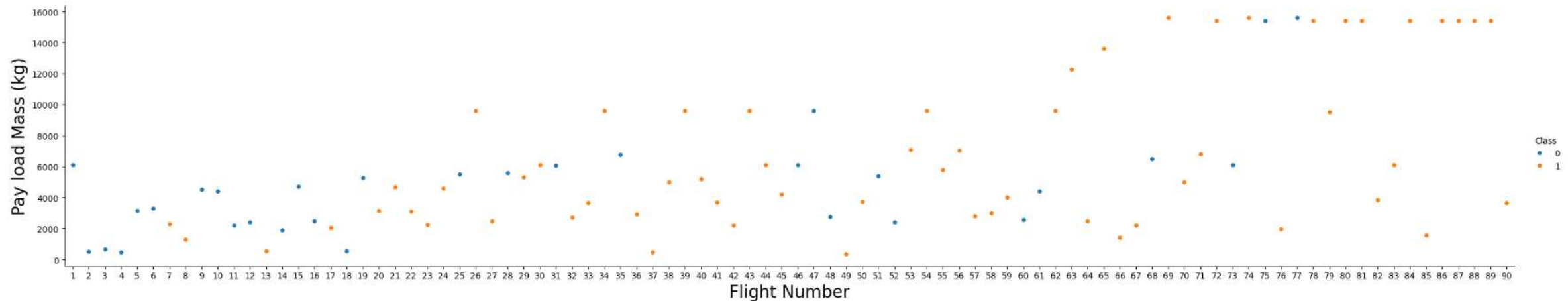
Section 2

# Insights drawn from EDA

# Flight Number vs. PayloadMass

- We see that as the flight number increases, the first stage is more likely to land successfully. The payload mass also appears to be a factor; even with more massive payloads, the first stage often returns successfully.

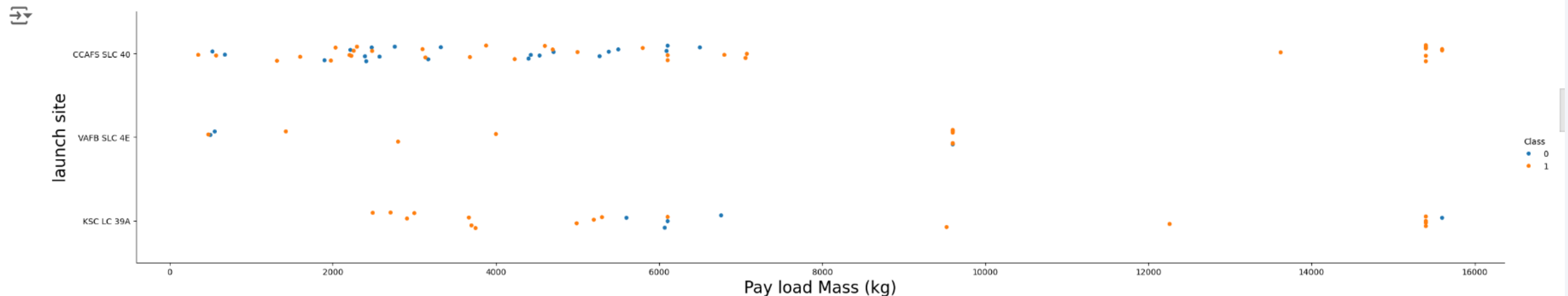
```
sns.catplot(y="PayloadMass", x="FlightNumber", hue="Class", data=df, aspect = 5)  
plt.xlabel("Flight Number",fontsize=20)  
plt.ylabel("Pay load Mass (kg)",fontsize=20)  
plt.show()
```



# Payload vs. Launch Site

- As we can see, The launch site CCAFS SLC 40 has highest number of payloads.

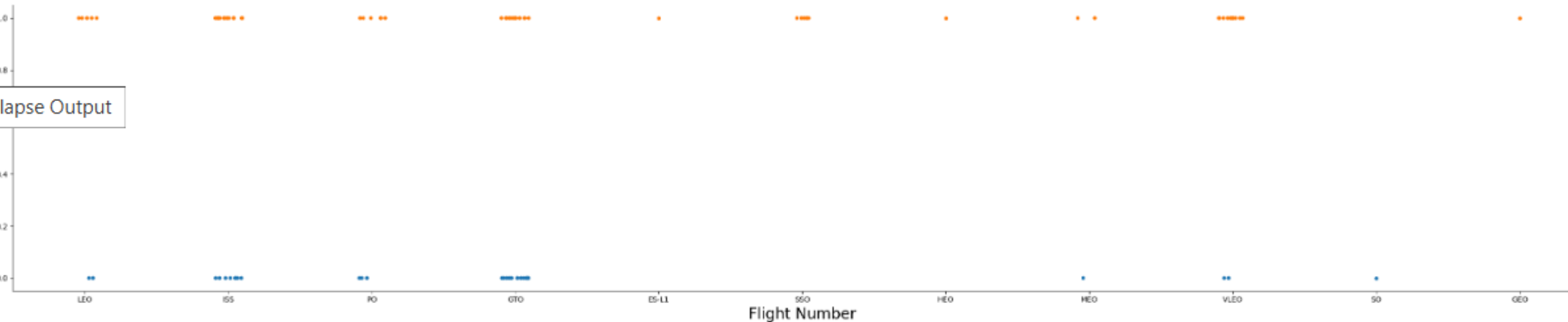
```
[ ] # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("Pay load Mass (kg)",fontsize=20)
plt.ylabel("launch site",fontsize=20)
plt.show()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





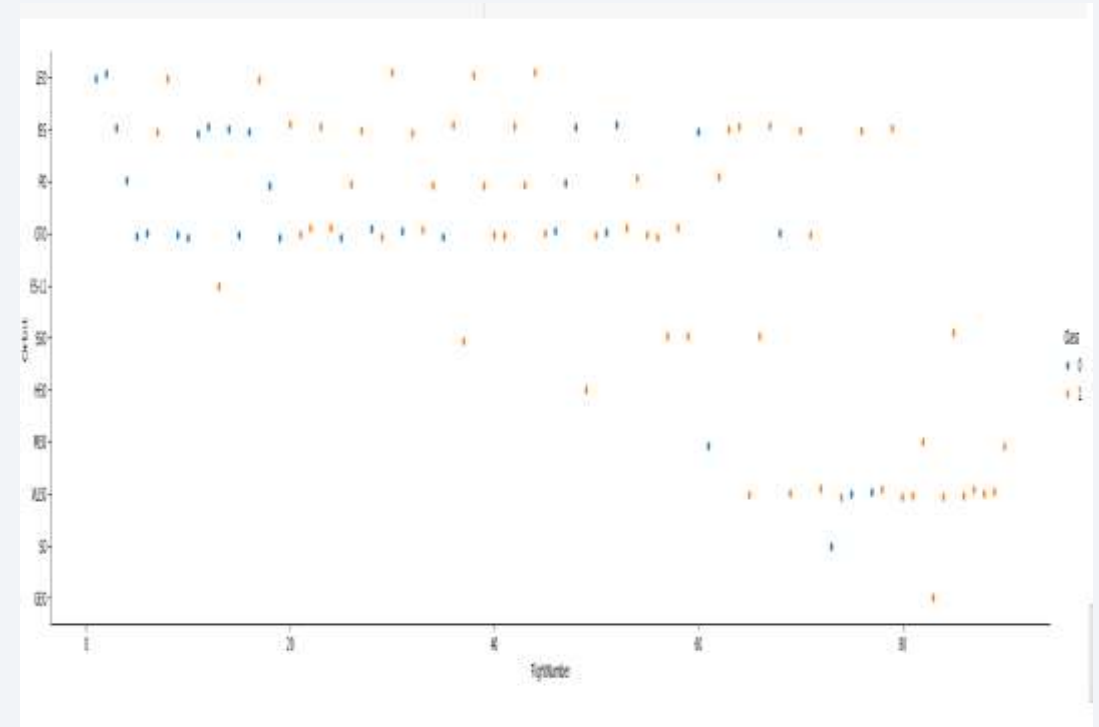
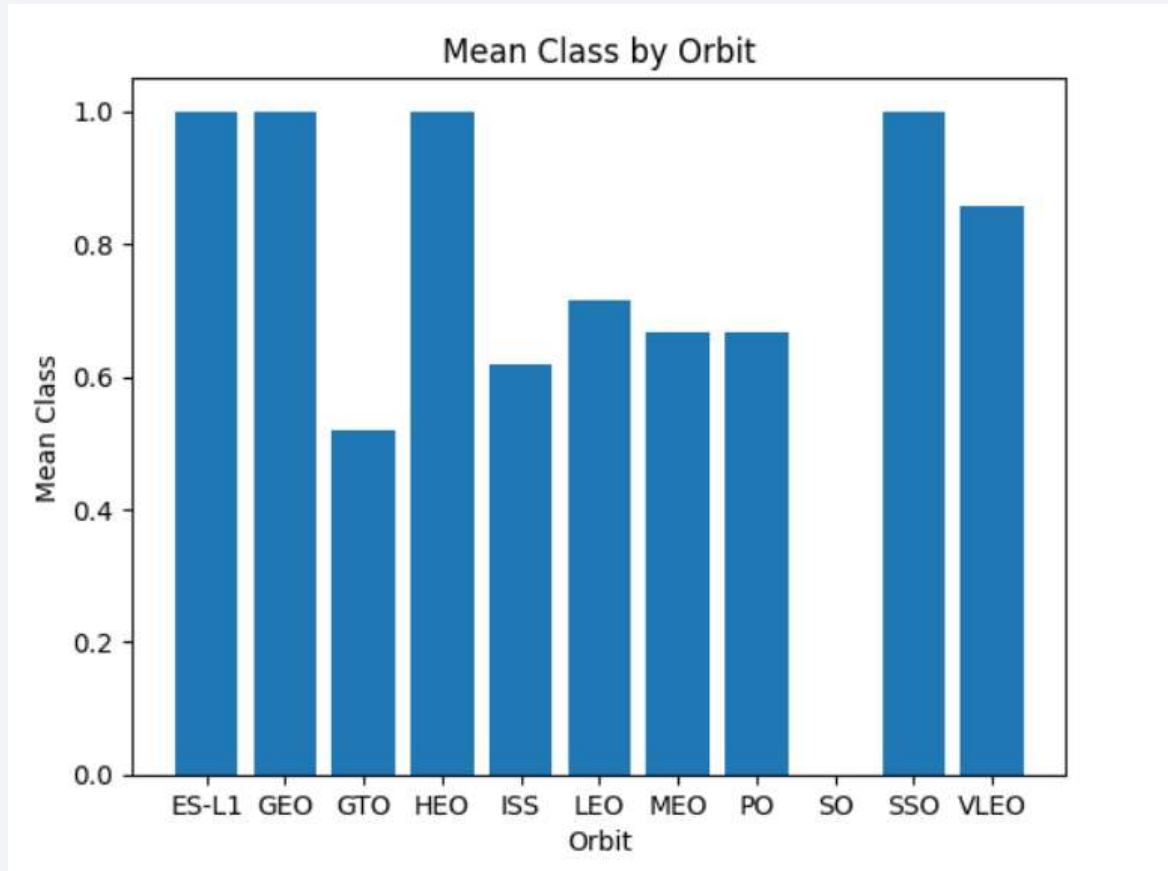
# Success Rate vs. Orbit Type

```
ns.catplot(y="Class", x="Orbit", hue="Class", data=df, aspect=5, height=6)
plt.xlabel("Flight Number", fontsize=20)
plt.ylabel("Payload Mass (kg)", fontsize=20)
plt.show()
```



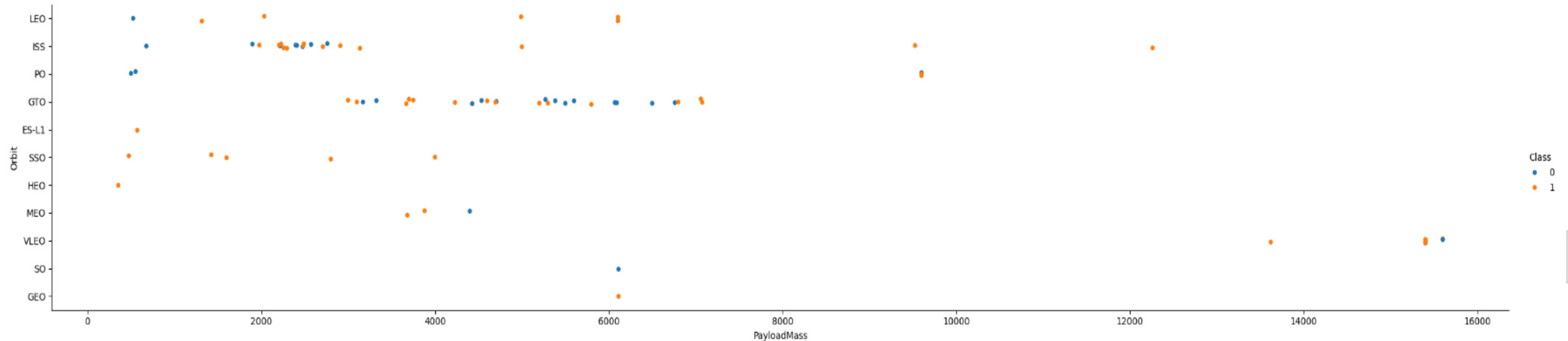


# Flight Number vs. Orbit Type



# Payload vs. Orbit Type

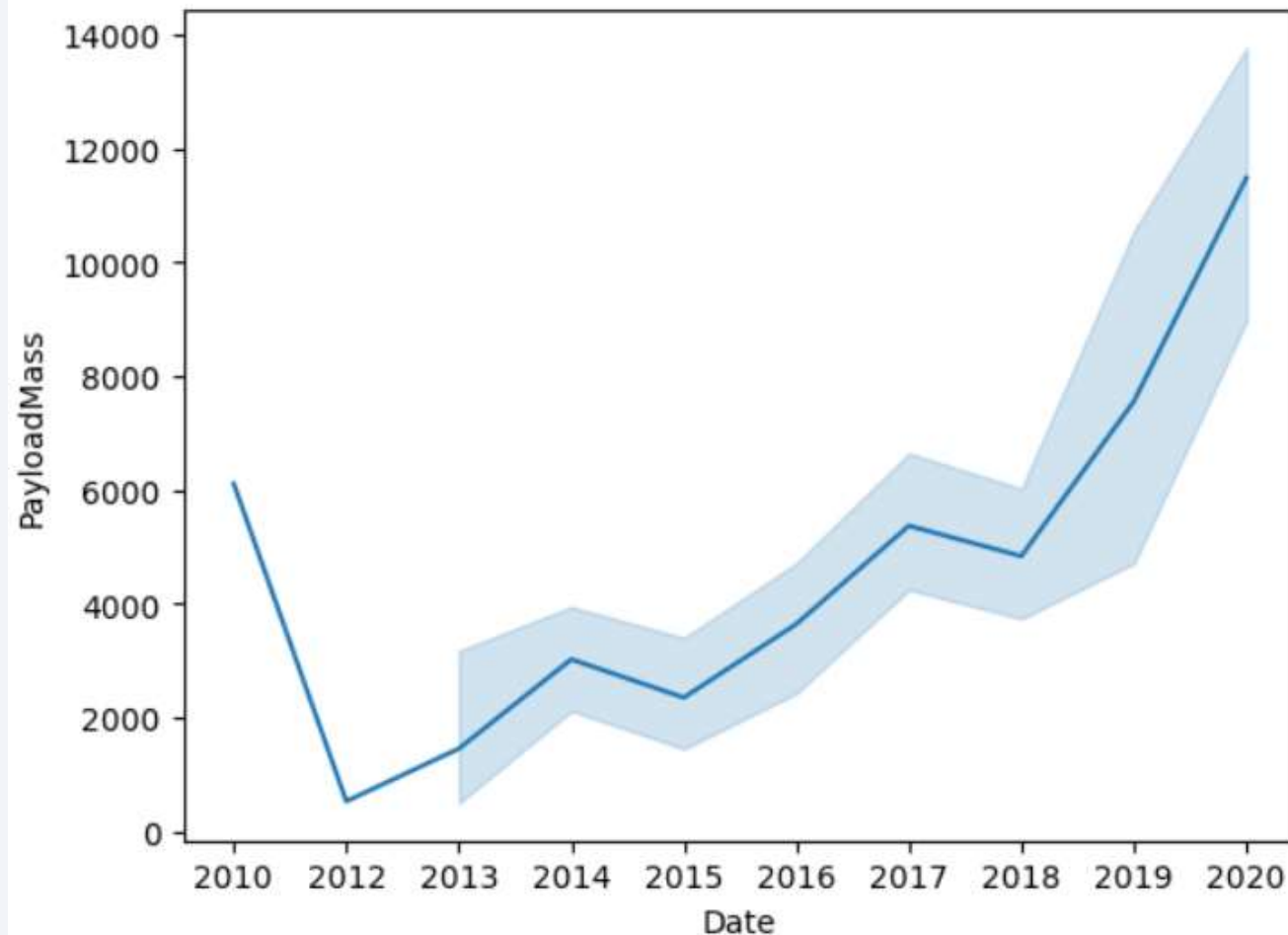
```
# Plot a scatter point chart with x axis to be Payload Mass and y axis to be the Orbit, and hue to be the class value
sns.catplot(x="PayloadMass",y="Orbit",hue="Class",data=df,aspect=5)
plt.xlabel("PayloadMass")
plt.ylabel("Orbit")
plt.show()
```



# Launch Success Yearly Trend

---

- We can observe that the success rate since 2013 kept increasing till 2020.



# All Launch Site Names

---

- The query to find the names of all launch site is “%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'

- The below image displays the five 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
2010-06-04	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

Task 2

# Total Payload Mass

---

- The total payload carried by boosters from NASA is calculated using sum built in function of sql.

```
[ ] %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer like 'NASA%';
```



```
* sqlite:///my_data1.db  
Done.  
sum(PAYLOAD_MASS__KG_)  
99980
```



# Average Payload Mass by F9 v1.1

---

- The average payload Mass by F9 v1.1 is calculated as below,

```
[ ] %sql select avg(PAYLOAD_MASS_KG_) from SPACEXTABLE where Booster_Version like 'F9 V1.1%';
```

```
↳ * sqlite:///my_data1.db  
Done.  
avg(PAYLOAD_MASS_KG_)  
2534.6666666666665
```

# First Successful Ground Landing Date

---

- As we can see, the first successful ground landing date is 2010-06-04.

```
%sql select Date from SPACEXTABLE where Mission_Outcome='Success' order by Date Limit 1;
```

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

```
Done.
```

```
   Date
```

```
2010-06-04
```

Task 6

# Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
%sql select Booster_version from SPACEXTABLE where Mission_Outcome = 'Success' and PAYLOAD_MASS__KG_ between 4000 and 6000;
```



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


```
Done.
```

```
Booster_Version
```

```
F9 v1.1  
F9 v1.1 B1011  
F9 v1.1 B1014  
F9 v1.1 B1016  
F9 FT B1020  
F9 FT B1022  
F9 FT B1026  
F9 FT B1030  
F9 FT B1021.2  
F9 FT B1032.1  
F9 B4 B1040.1  
F9 FT B1031.2  
F9 FT B1032.2  
F9 B4 B1040.2  
F9 B5 B1046.2
```

# Total Number of Successful and Failure Mission Outcomes

```
[ ] %sql SELECT Mission_Outcome, COUNT(Mission_Outcome) AS Total_Count FROM SPACEXTABLE GROUP BY Mission_Outcome;
```

 \* sqlite:///my\_data1.db  
Done.

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

# Boosters Carried Maximum Payload

---

- The boosters carried Maximum Payload is obtained by using sub query methods.

```
%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

---

- The results of 2015 failed launch records as shown below. The query is in the sql notebook.

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

<b>Month_Name</b>	<b>Landing_Outcome</b>	<b>Booster_Version</b>	<b>Launch_Site</b>
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40



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

The results of landing Outcomes Between 2010-06-04 and 2017-03-20. The query used for this is %%sql

```
SELECT
    Landing_Outcome,
    COUNT(*) AS Outcome_Count,
    RANK() OVER (ORDER BY COUNT(*) DESC) AS Rank
FROM
    SPACEXTABLE
WHERE
    Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY
    Landing_Outcome
ORDER BY
    Rank;
```

```
FROM
    SPACEXTABLE
WHERE
    Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY
    Landing_Outcome
ORDER BY
    Rank;
```



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

Landing_Outcome	Outcome_Count	Rank
No attempt	10	1
Success (drone ship)	5	2
Failure (drone ship)	5	2
Success (ground pad)	3	4
Controlled (ocean)	3	4
Uncontrolled (ocean)	2	6
Failure (parachute)	2	6
Precluded (drone ship)	1	8

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The image is dark blue with a thin white line representing the horizon. The city lights are visible as bright yellow and orange spots against the dark blue background of the night sky.

Section 3

# Launch Sites Proximities Analysis

# Folium Map to locate the launch sites

- This map illustrates the launch sites in various location.



# Folium Map to show launch site success Rate

- Here, we can see the success and failure launch counts in different colors.





# Folium Map to calculate the nearest distance

---

- From this Map, we can infer that the launch site is located **Proximity to Transportation Infrastructure and nearer the Equator.**





Section 4

# Build a Dashboard with Plotly Dash

# Pie chart of launch success count for all sites

---

- The screenshot of launch success count for all sites, in a pie chart. Here, we can compare which launch site has the highest.

Launch Success Rate for All Sites



# Pie chart of the most successful launch site

---

- The most successful launch site is KSC-LC-39A

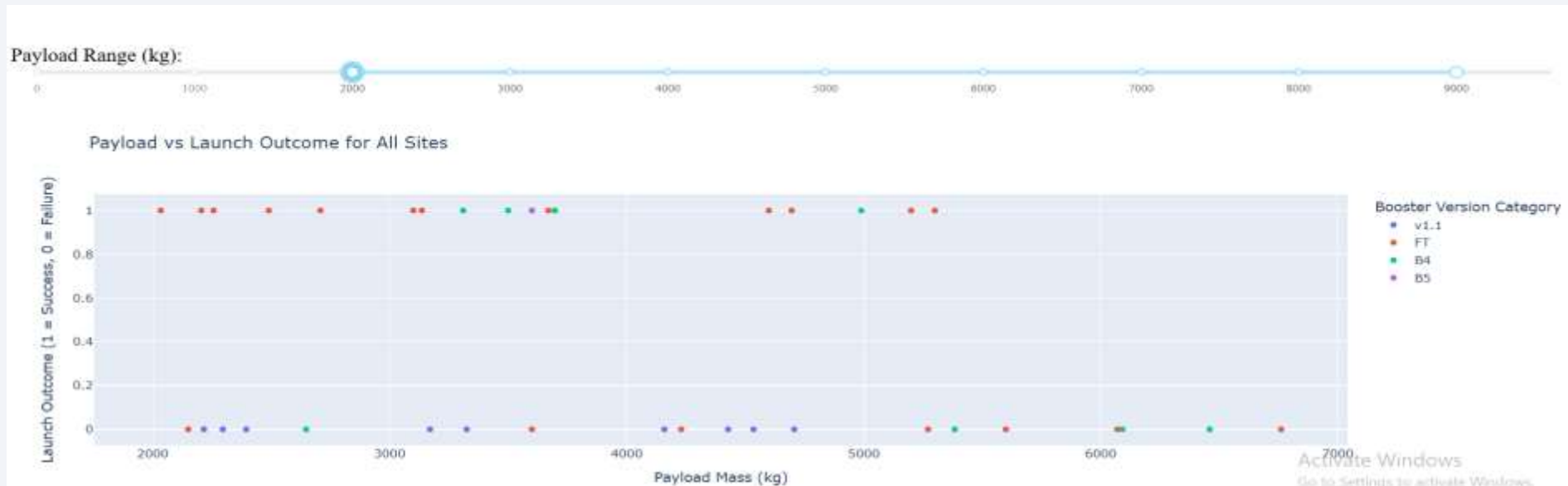
Launch Success Rate for KSC LC-39A





# Scatter Plot of Payload vs. Launch Outcome

- The importance of the payload and booster version lies in maintaining an optimal weight to transport the satellite to space. As the payload size increases, accommodating it becomes more challenging



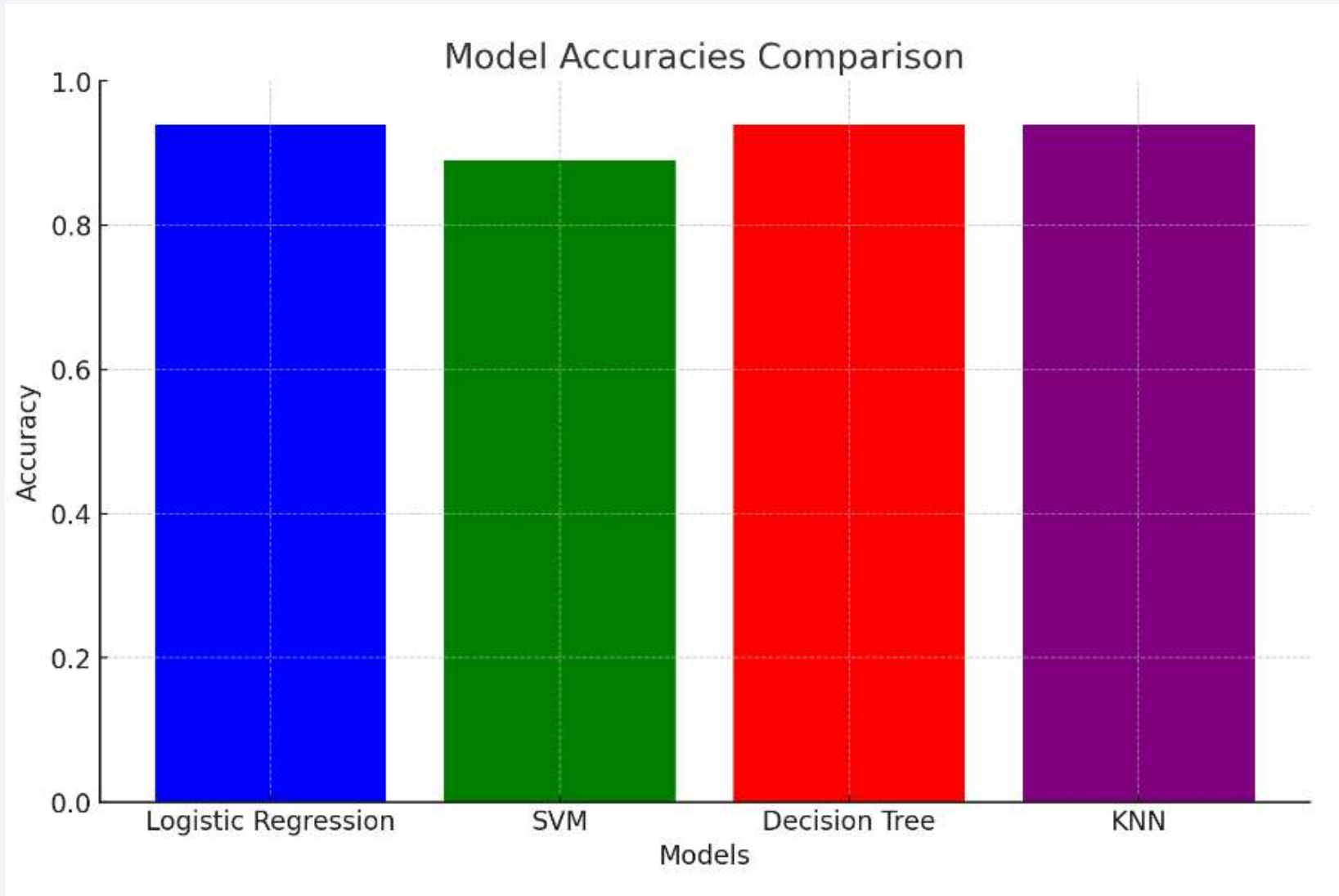


Section 5

# Predictive Analysis (Classification)

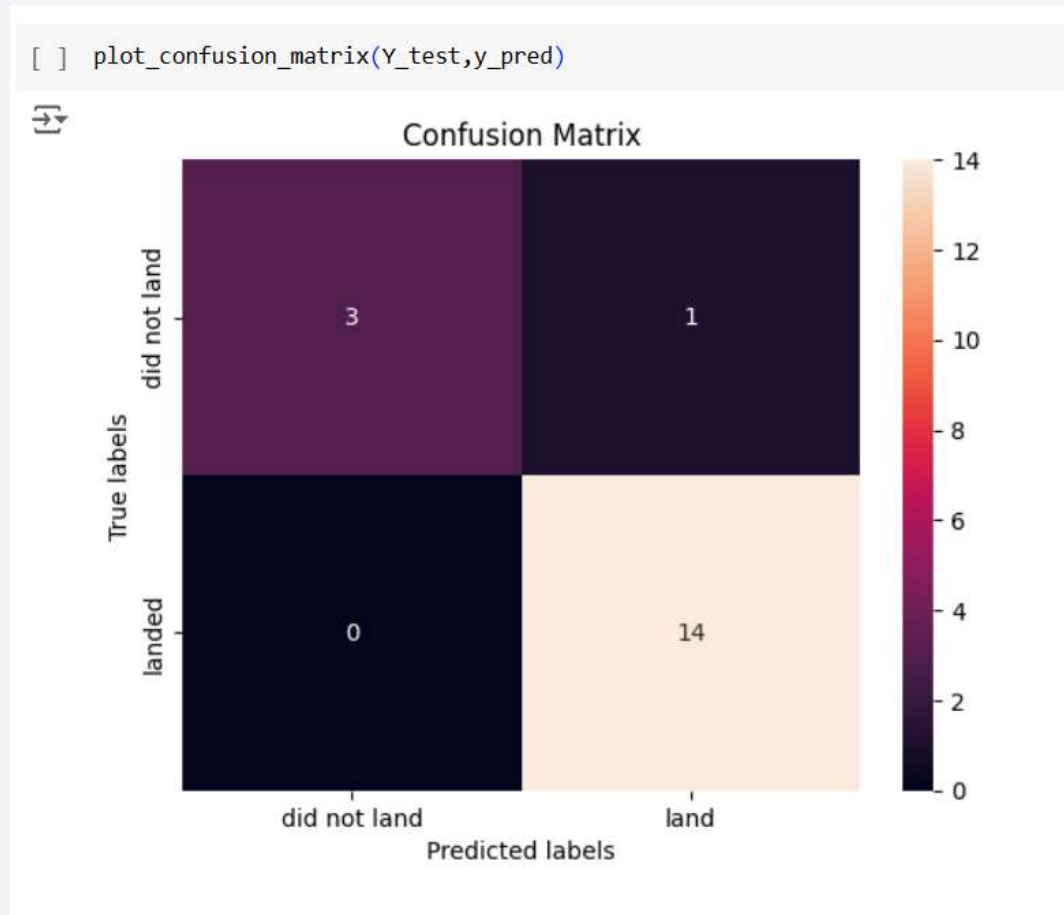
# Classification Accuracy

---



# Confusion Matrix

The Confusion Matrix of KNN model is shown as below,



# Conclusions

---

- The best model for predicting a successful launch site should incorporate these discussed factors: proximity to reliable transportation infrastructure, coastal equatorial positioning for efficiency and safety, and distance from populated areas.
- These factors collectively enhance logistical, operational, and safety outcomes for rocket launches.
- In this project, we determined that the best model for predicting the success rate of rocket launches is based on the K-Nearest Neighbors (KNN) classification algorithm

# Appendix

---

- [https://github.com/Aruna2897/IBM Data science course assignments](https://github.com/Aruna2897/IBM_Data_science_course_assignments)
- [https://github.com/Aruna2897/IBM Data science course assignments/commit/a386e53069810f912ec99c1542aff617f109c9c8](https://github.com/Aruna2897/IBM_Data_science_course_assignments/commit/a386e53069810f912ec99c1542aff617f109c9c8)

Thank you!

