IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

1. Summary of methodologies

Web Scraping – Gathered Falcon 9 launch data from online sources.

EDA (Exploratory Data Analysis) – Explored patterns, trends, and relationships in the dataset.

Data Wrangling – Handled missing values, ensured consistency, and engineered new features.

Visualization – Created charts to present key insights from the data.

Predictive Modeling – Built and evaluated a model to predict Falcon 9 launch outcomes.

2. Summary of all results

Most Falcon 9 rockets launched successfully when near or on the coastline. Some launches failed, and while rockets with launch power around 1000 succeeded, some with power around 1600 did not land successfully.

Introduction

- **Project background and context**

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

- **Problems you want to find answers**
- How can we determine if the first stage will land successfully?
- Determining the cost of the launch?
- How many flights are required to launch the rocket?
- Which launch site has the most successful landing history?
- Maximum and minimum payload needed for a Falcon 9 launch?

Section 1

Methodology

Methodology

Executive Summary

- **Web Scraping** – Gathered Falcon 9 launch data from online sources.
- **EDA (Exploratory Data Analysis)** – Explored patterns, trends, and relationships in the dataset.
- **Data Wrangling** – Handled missing values, ensured consistency, and engineered new features.
- **Visualization** – Created charts to present key insights from the data.
- **Predictive Modeling** – Built and evaluated a model to predict Falcon 9 launch outcomes.

Data Collection

- Describe how data sets were collected.

The data sets are collected from Space X website using Rest API and additional tabular data is scraped from Wikipedia using Beautiful Soup with Python.

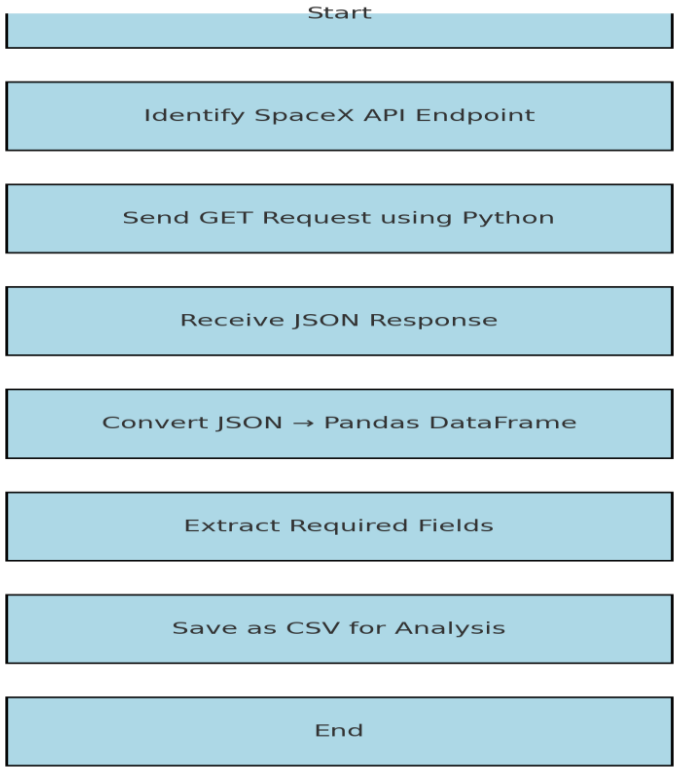
The API provides detailed JSON responses containing information on launch dates, launch sites, payloads, rocket details, and landing outcomes.

- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose
- Web Scraping- Click link

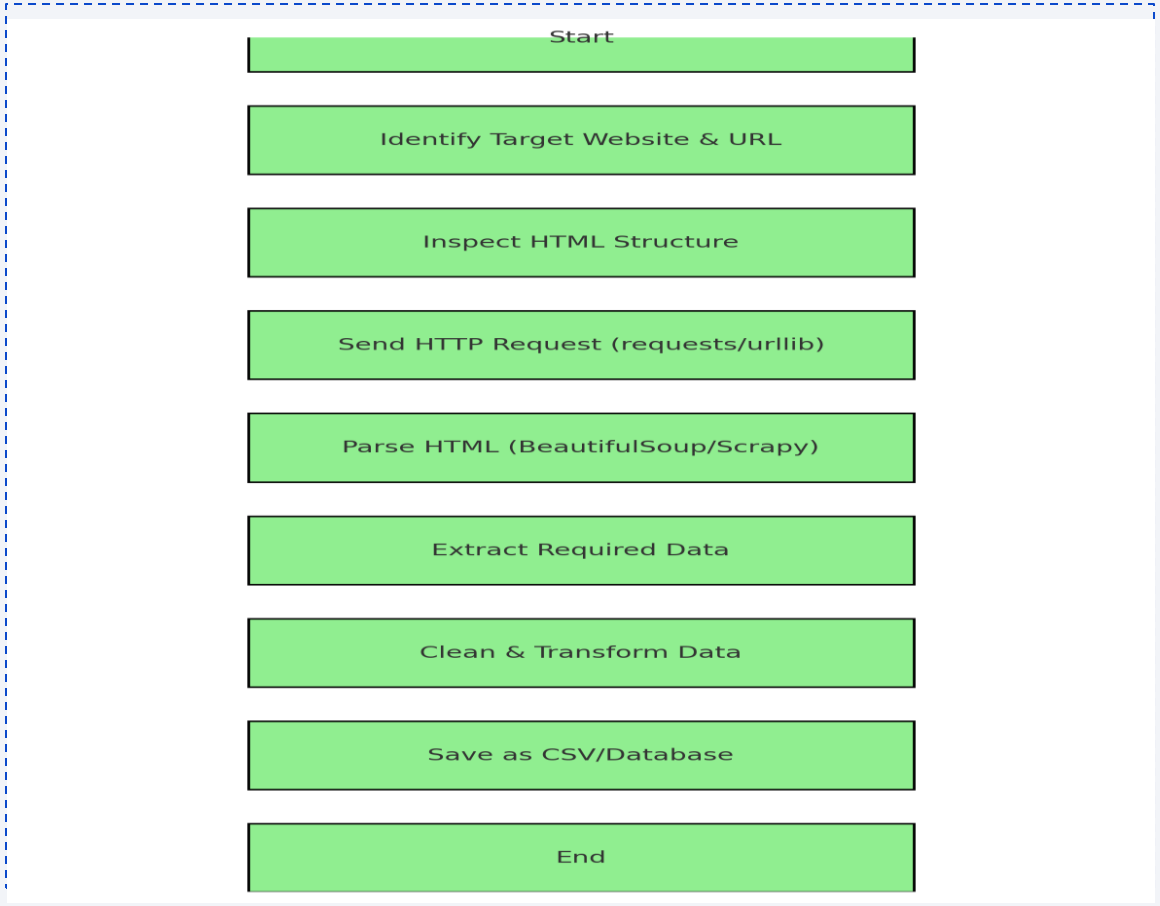


Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

- Data Collection - Click Link



Data Wrangling

- **Describe how data were processed**

The data is processed using Pandas and NumPy. We performed and answered key questions with data wrangling, such as. calculating the Number of launches on each site:

- **You need to present your data wrangling process using key phrases and flowcharts**
- Calculating the Number of launches on each site:

```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

```
LaunchSite
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: count, dtype: int64
```

Data Wrangling

- Creating the number and Occurrence of each orbit

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
Orbit
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
ES-L1     1
HEO       1
SO        1
GEO       1
Name: count, dtype: int64
```

- We engineered a new column, '**class**',

  to classify successful and unsuccessful landings as 1 and 0,

  respectively.

```
# landing_class = 0 if bad_outcome
landing_class=[
    0 if outcome in bad_outcomes else 1
    for outcome in df['Outcome']
]
# landing_class = 1 otherwise
```

This variable will represent the classification v
means the first stage landed Successfully

```
df['Class']=landing_class
df[['Class']].head(8)
```

	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1

Data Wrangling

- We also determined the unique launch sites and the number of successful landings from each launch Site.

```
Outcome
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: count, dtype: int64
```

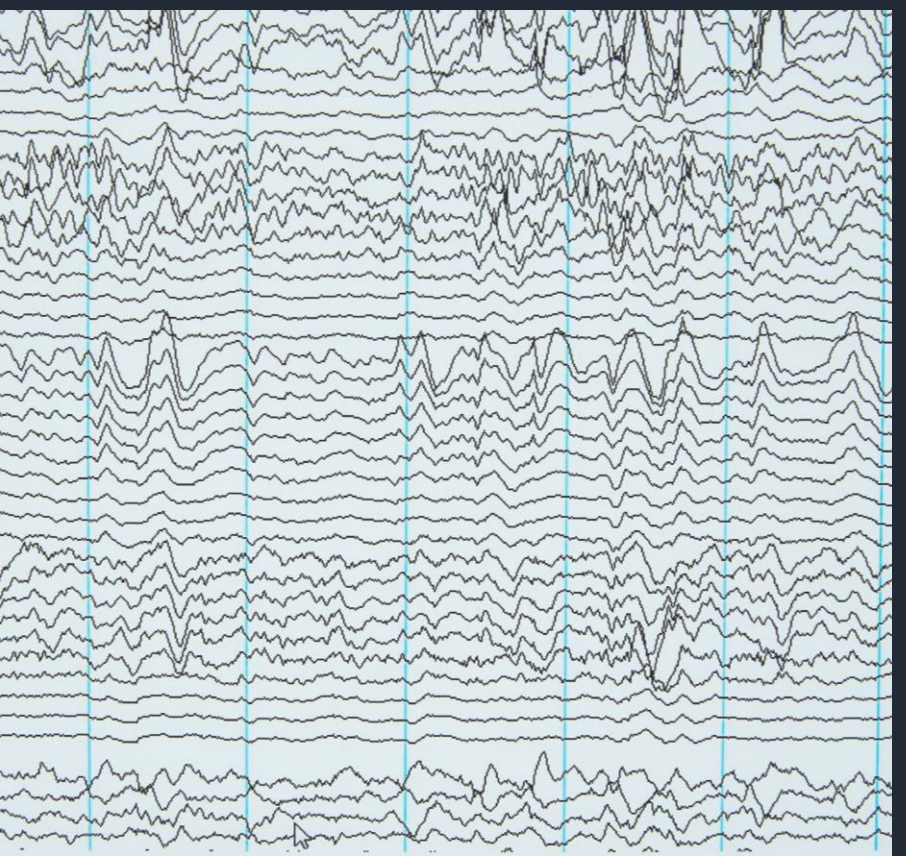
```
a=df['LaunchSite'].unique()
print(a)
```

```
['CCAFS SLC 40' 'VAFB SLC 4E' 'KSC LC 39A']
```

- Add the GitHub URL of your completed Data Wrangling ,as an external reference and peer-review purpose

[Data Wrangling- Click link]

EDA with Data Visualization



In data visualization, I used scatter plots, cat plots, and line plots to visualize and gain insights on launch sites, payload masses, and the class (whether the launch was successful or unsuccessful)

- **Charts Plotted and Purpose**
- **Scatter Plots** – Used to show the relationship between Flight number and launch success across different launch sites.
- **Cat Plots** – Helped compare categorical variables, such as launch site and success rate, in a clear and grouped format.
- **Line Plots** – Illustrated trends over time, such as changes in success rates or payload capacities across launches.
- **Reason for Using These Charts**
- **Scatter Plots** → Identify correlations and patterns between numerical and categorical data.
- **Cat Plots** → Compare performance metrics across categories for better grouping insights.
- **Line Plots** → Observe trends and changes over time for time-series analysis.

- GITHUB URL: Data Visualization - Link

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- **Unique Launch Sites** – Identifies all distinct Falcon 9 launch locations.
- **"CCA" Sites** – Shows launches from Cape Canaveral facilities.
- **NASA (CRS) Payload** – Total cargo mass carried for NASA CRS missions.
- **F9 v1.1 Average Payload** – Typical payload for Falcon 9 v1.1 booster.
- **First Ground Pad Landing** – Date of first successful ground pad recovery.
- **Successful Drone Ship (4–6t)** – Boosters landing offshore with medium-heavy payloads.
- **Max Payload Boosters** – Booster versions with the highest payload capacity.
- **2015 Drone Ship Failures** – Offshore landing failures by month in 2015.
- **Landing Outcome Rank** – Most common landing results (2010–2017).
- **Success vs Failure** – Overall mission reliability rate.

- **Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose**

SQL EDA- LINK

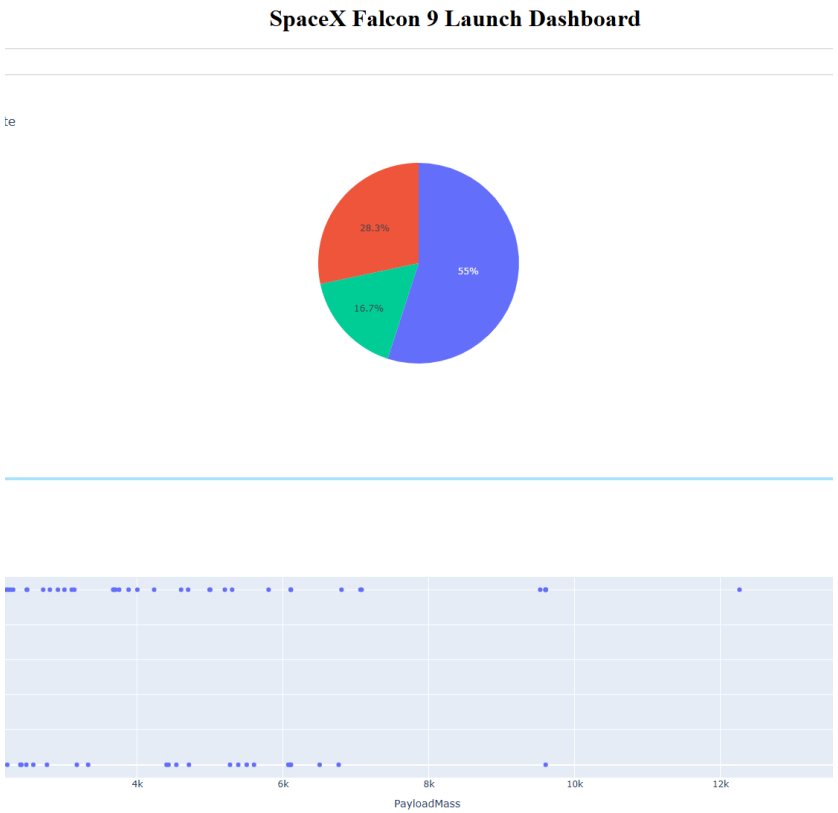
Build an Interactive Map with Folium

- **Map Objects Created in Folium**
- **Markers** – Placed at each unique launch site to show exact geographic locations.
- **Circles** – Added around launch sites to visually highlight surrounding areas and make them easier to spot on the map.
- **Lines** – Drew paths between launch sites and relevant points (e.g., Coastline) to visualize distances and connections.
- **Reason for Adding These Objects**
- **Markers** → Provide quick, clickable points for site identification and details.
- **Circles** → Emphasize and visually separate each site from the background map.
- **Lines** → Help understand spatial relationships and travel paths between important locations.

Github Link: https://github.com/shomaielkhan/Data-Science-Capstone/blob/main/lab-jupyter-launch-site-location-v2%20(1).ipynb

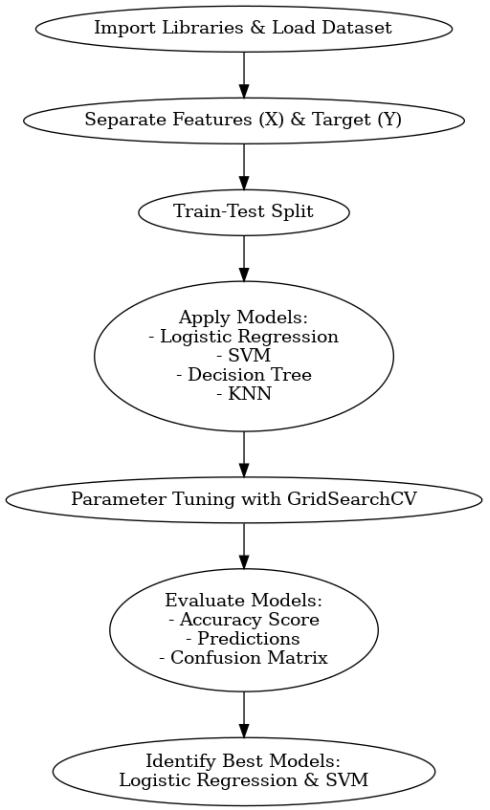
Build a Dashboard with Plotly Dash

- Explain why you added those plots and interactions
- PieCharts: Quickly tells you which sites are most successful or how reliable a specific site is.
- RangeSlider: Helps understand how payload mass affects launch success.
- Scatter: You can compare performance of different booster versions.
- Filtering by site or payload makes it easier to analyze patterns.
- https://github.com/shomaielkhan/Data-Science-Capstone/blob/main/dashboard.py



Predictive Analysis (Classification)

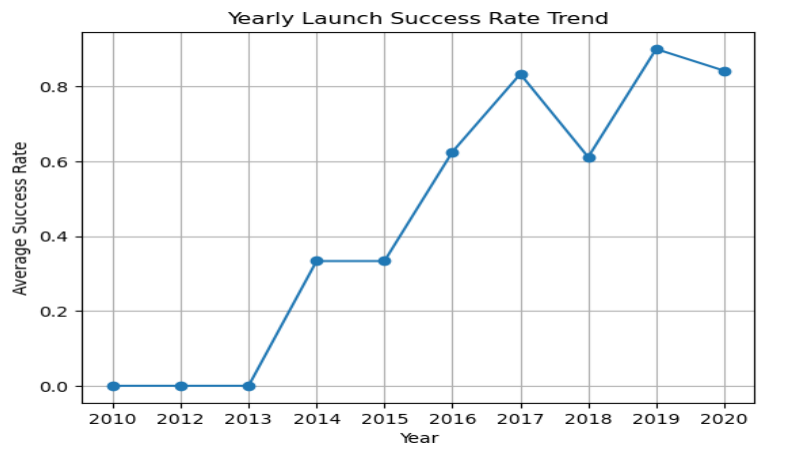
- First, I imported the required libraries and loaded the dataset in the notebook. After that, I stored the **Y** column for prediction in the y variable and the other data in the x variable for transformation. I used **train-test split** to separate the data into training and testing sets.
- Then, I applied **Logistic Regression**, **SVM**, **Decision Tree**, and **KNN** to perform testing and scoring. I also evaluated and tuned the models using **GridSearchCV** with the required parameters.
- Additionally, I calculated the **accuracy score**, made predictions for all algorithms, and created a **confusion matrix**. The final results showed that **Logistic Regression** and **SVM** were the best-performing models.
- Github URL: [Predictive Model - Link]

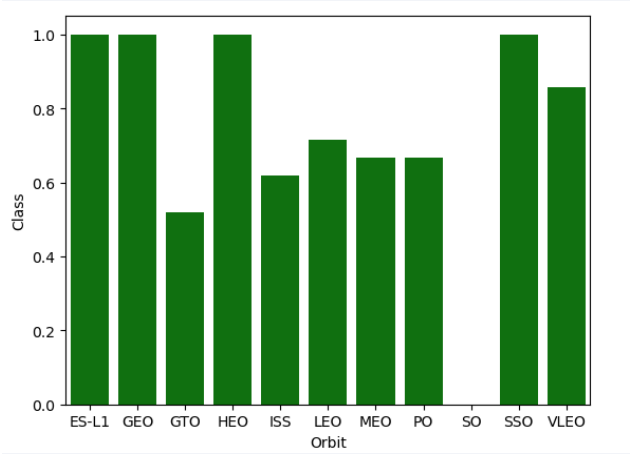


Results

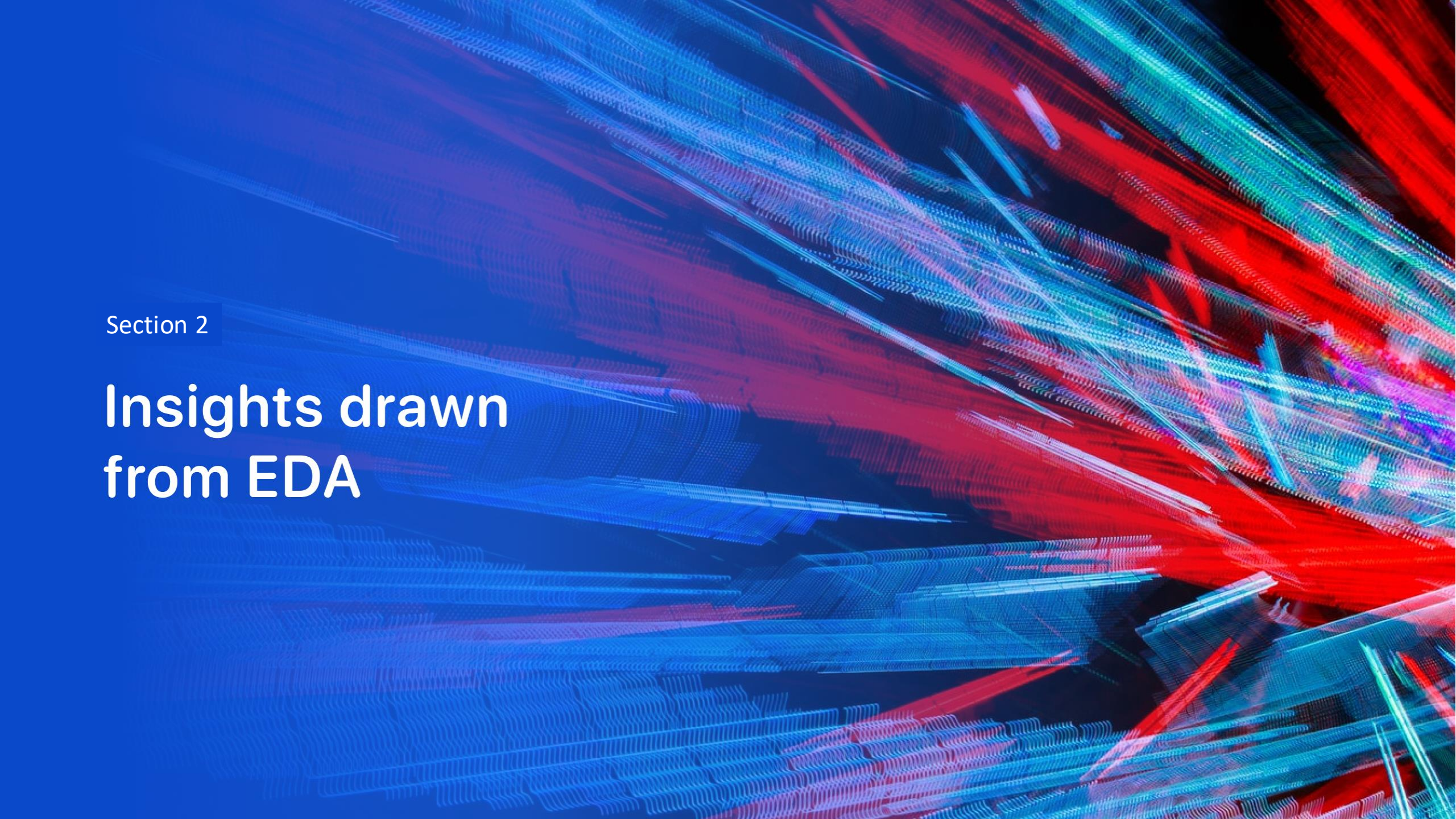
- **Exploratory data analysis results**
  I performed EDA to identify relationships between variables, applied feature engineering, and cleaned the dataset through data wrangling. This helped in better understanding the data and preparing it for predictive modeling.

- **Interactive analytics demo in screenshots**





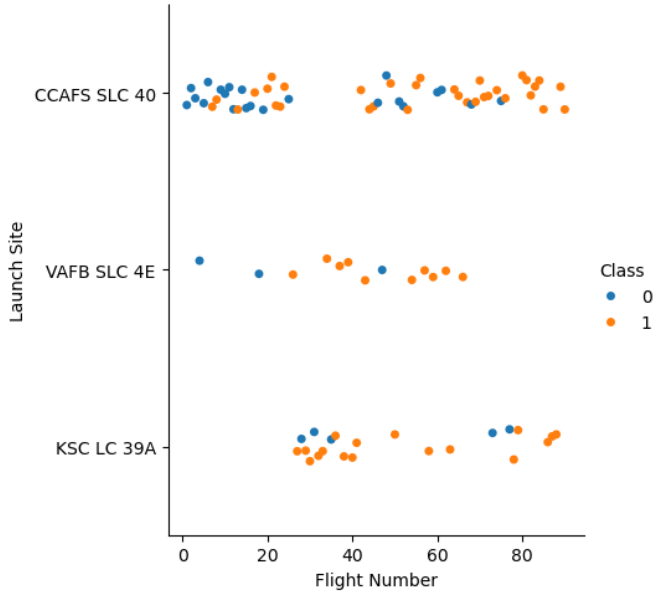
- **Predictive Analysis Results**
- Implemented **Logistic Regression, SVM, Decision Tree, and KNN** models.
- Used **GridSearchCV** for hyperparameter tuning.
- Accuracy scores: Logistic Regression `0.9444`**%**, SVM – `0.8889`**Y%**, Decision Tree –`0.9444`**Z%**, KNN `0.9444`**W%**.
- Best performing models: **Logistic Regression** and **Decision Tree** based on accuracy and confusion matrix results.


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site

- Show the screenshot of the scatter plot with explanations



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

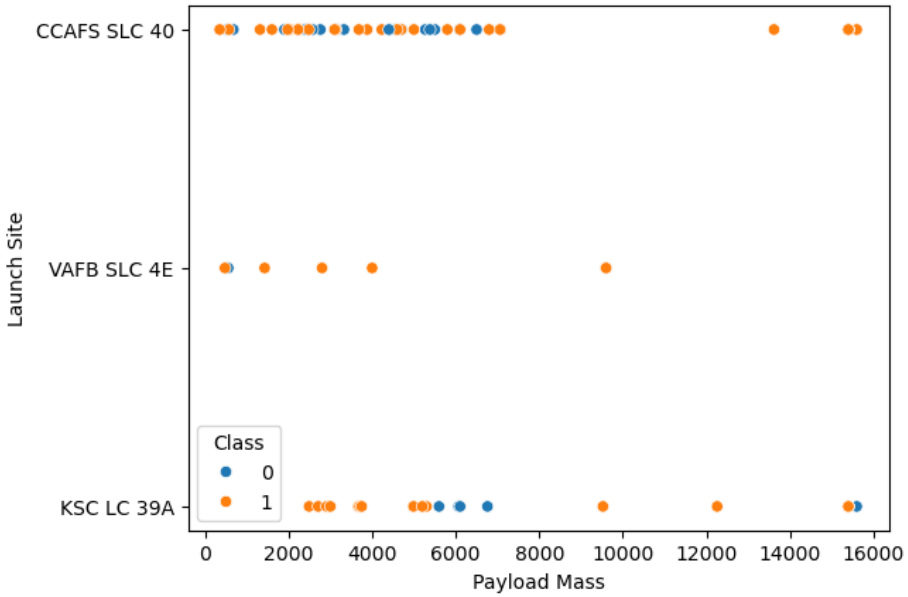
The CCAFS SLC-40 launch site shows that the flight number ranges are very high, and most of the Falcon rockets were successfully launched. At VAFB SLC-42, the range is much lower, but most launches were still successful. At KSC LC-39A, the flight numbers range from 20–90, with most landings being successful and some unsuccessful.

Unsuccessful landings are more frequent in the 0–20 flight numbers at CCAFS SLC-40.

Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site

- Show the screenshot of the scatter plot with explanations

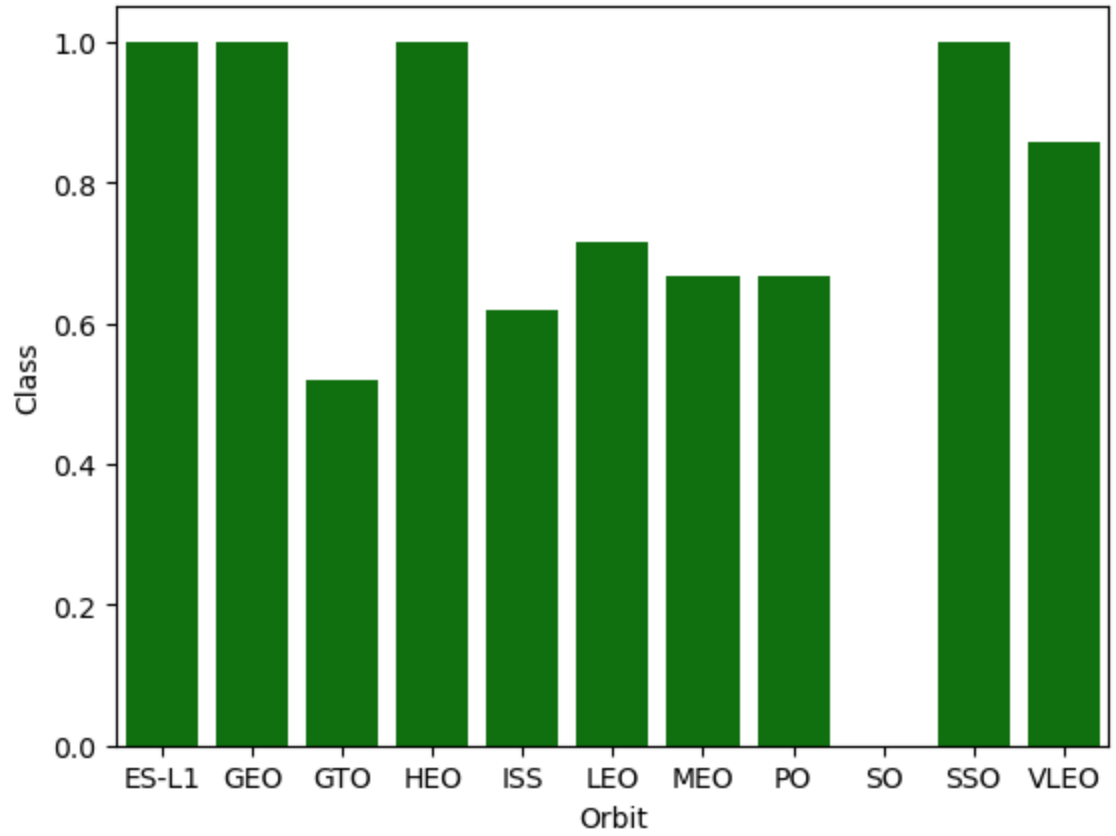


Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

- Show the screenshot of the scatter plot with explanations

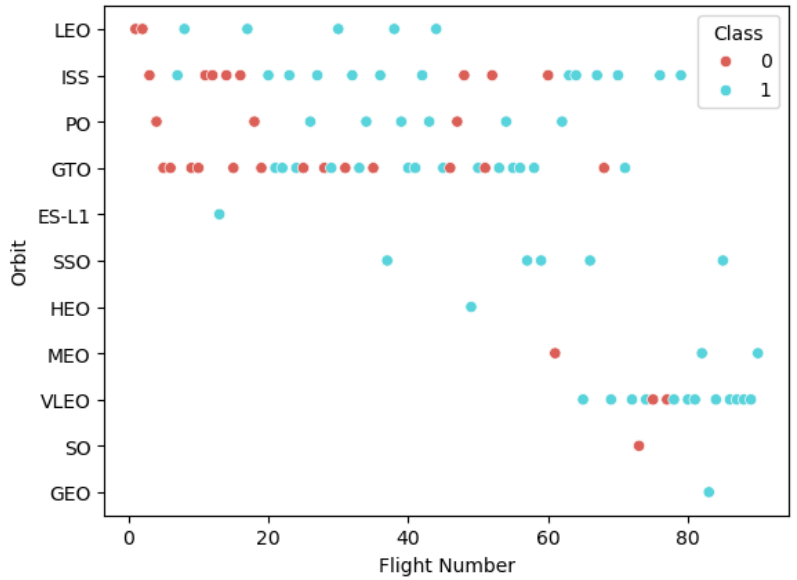


In this bar chat, GE0, HE0, SSO have most succesful landing

Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type

- Show the screenshot of the scatter plot with explanations

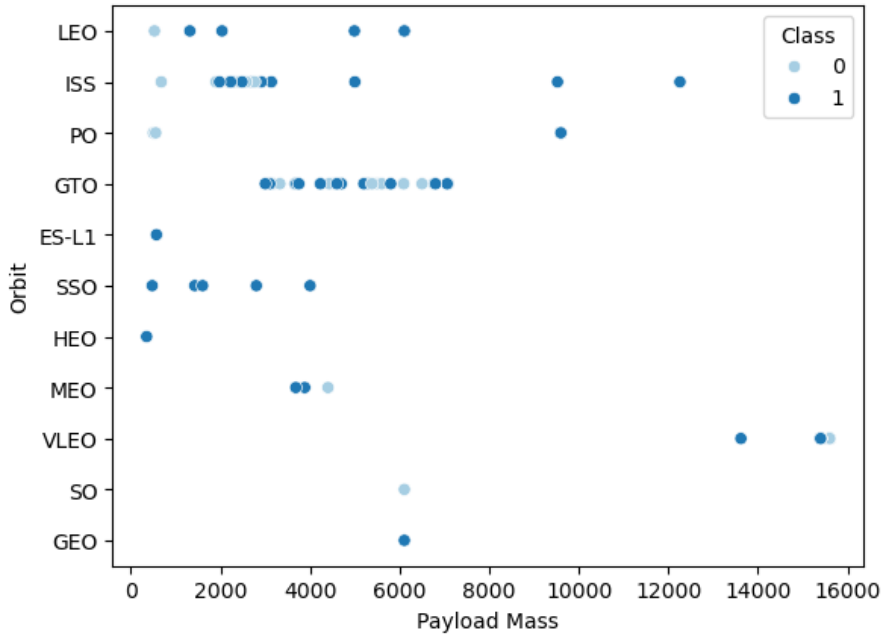


You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type

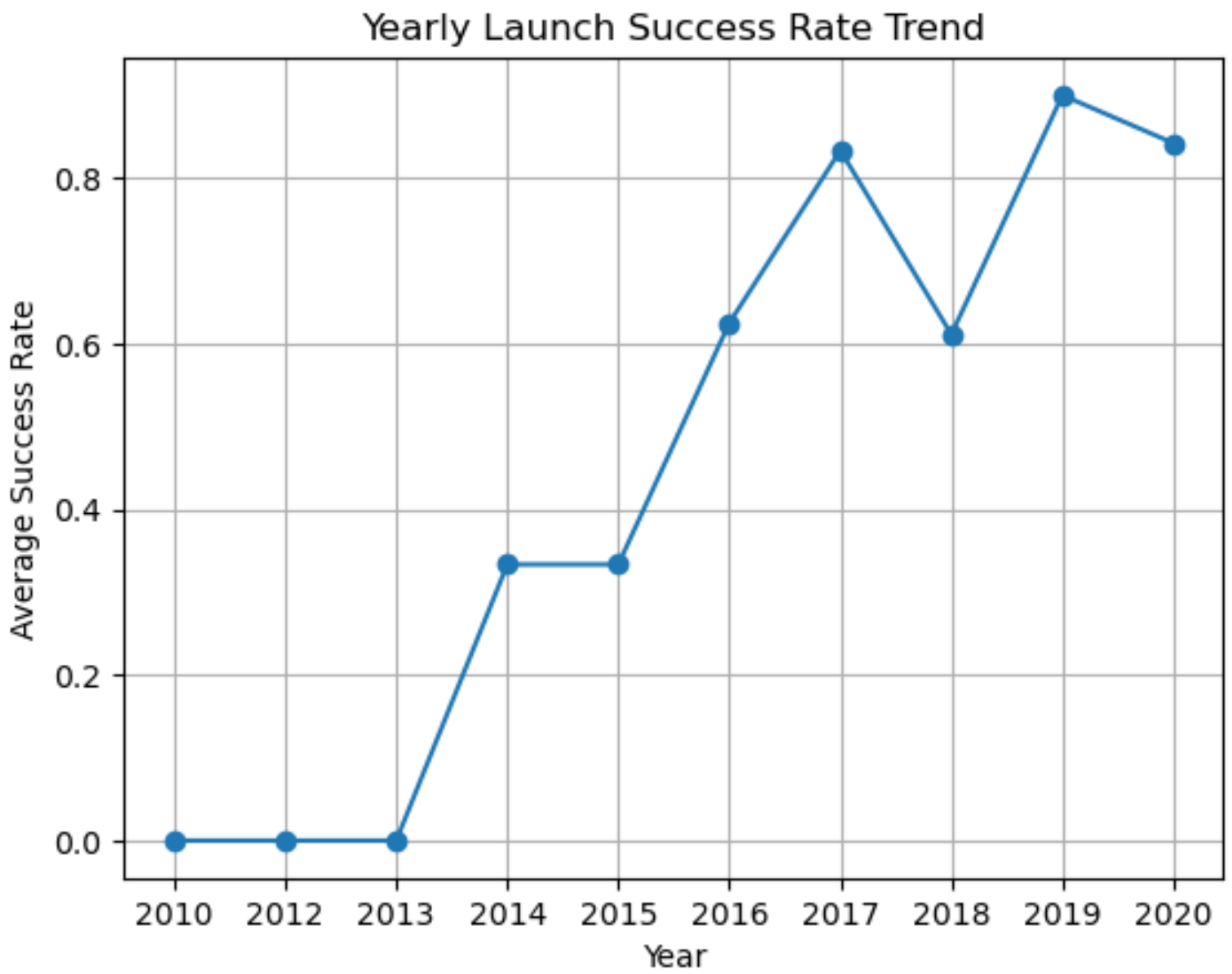
- Show a scatter point of payload vs. orbit type

- Show the screenshot of the scatter plot with explanations



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(unsuccessful mission) are both there here.



Launch Success Yearly Trend

- Show a line chart of yearly average success rate
- Show the screenshot of the scatter plot with explanations

You can observe that the success rate since 2013 kept increasing till 2017 (stable in 2014) and

All Launch Site Names

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

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

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

- These are 4 unique Launch Sites.

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`

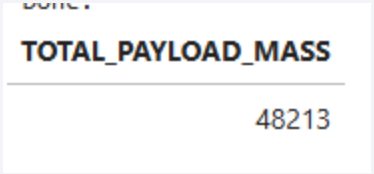
%sql SELECT * from SPACEXTBL where Launch_Site LIKE 'CCA%' LIMIT 5;

- Present your query result with a short explanation here

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

- Calculate the total payload carried by boosters from NASA
- %sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL WHERE Customer LIKE 'NASA (CRS)%';
- Present your query result with a short explanation here



TOTAL_PAYLOAD_MASS
48213

- Total payload mass is 48213 carried by boosters from NASA

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

%sql SELECT AVG(PAYLOAD_MASS__KG_) AS Avg_Payload_Mass from SPACEXTBL WHERE Booster_Version='F9 v1.1';

- Present your query result with a short explanation here

Avg_Payload_Mass
2928.4

- Avg Payload Mass caried by booster version F9V1.1

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

%sql SELECT MIN(DATE) AS FIRST_SUCCESFULL_LANDING_OUTCOME from SPACEXTBL WHERE Landing_Outcome ='Success (ground pad)';

- Present your query result with a short explanation here

FIRST_SUCCESFULL_LANDING_OUTCOME
2015-12-22

- First succesfull Landing Outcome was on 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

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

- Present your query result with a short explanation here

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

```
%%sql
SELECT Landing_Outcome, COUNT(*) AS TotalCount
FROM SPACEXTBL
GROUP BY Landing_Outcome;
```

- Present your query result with a short explanation here

Landing_Outcome	TotalCount
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

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

- Present your query result with a short explanation here

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 failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- %%sql SELECT substr(Date,6,2) AS MONTH, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL
- WHERE SUBSTR(Date,1,4)='2015'
- AND Landing_Outcome LIKE 'Failure%';
- Present your query result with a short explanation here

MONTH	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

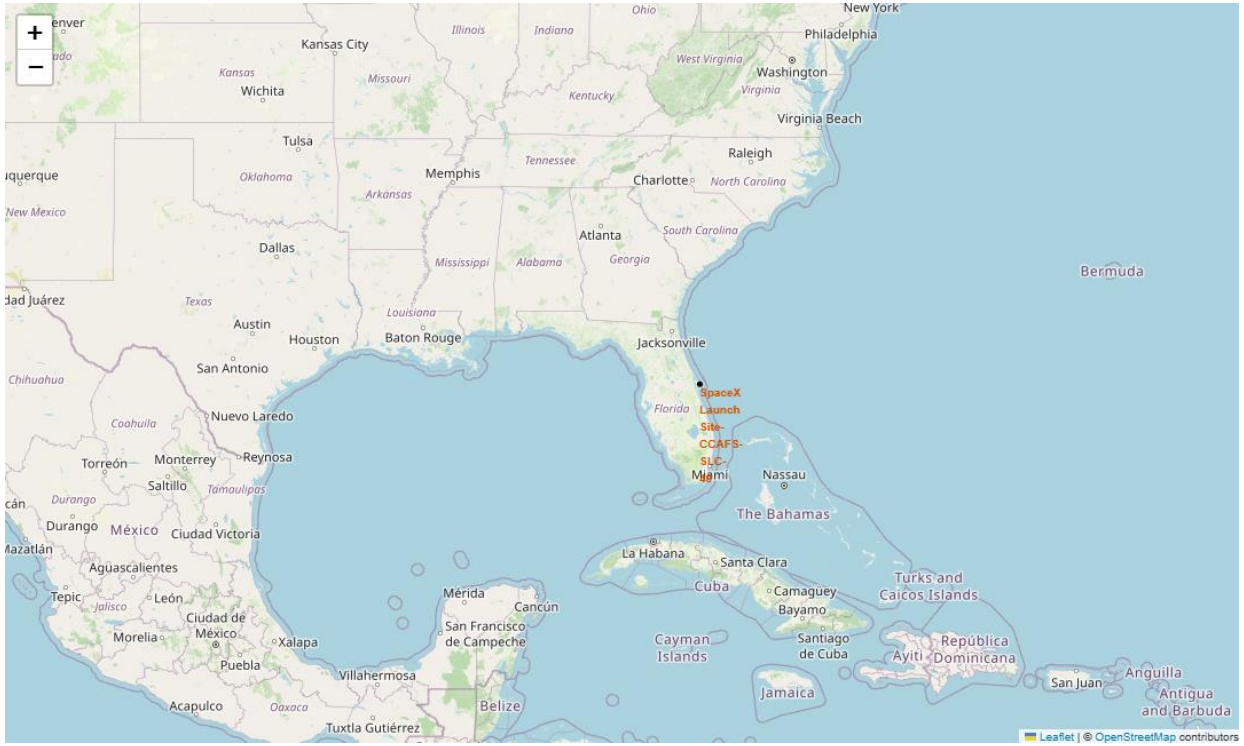
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- 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 OUTCOME_COUNT FROM SPACEXTBL
- WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
- GROUP BY Landing_Outcome
- ORDER BY OUTCOME_COUNT DESC;
- Present your query result with a short explanation here

Landing_Outcome	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

Section 3

Launch Sites Proximities Analysis



SpaceX Launch Site Map

- Space X Launch Sites Map

Important Elements and Findings

The key observation from the screenshot is that most launch sites are located near coastlines, likely to ensure safety and allow for efficient launch trajectories over open water.

Success Launch on each Sites on map

Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map

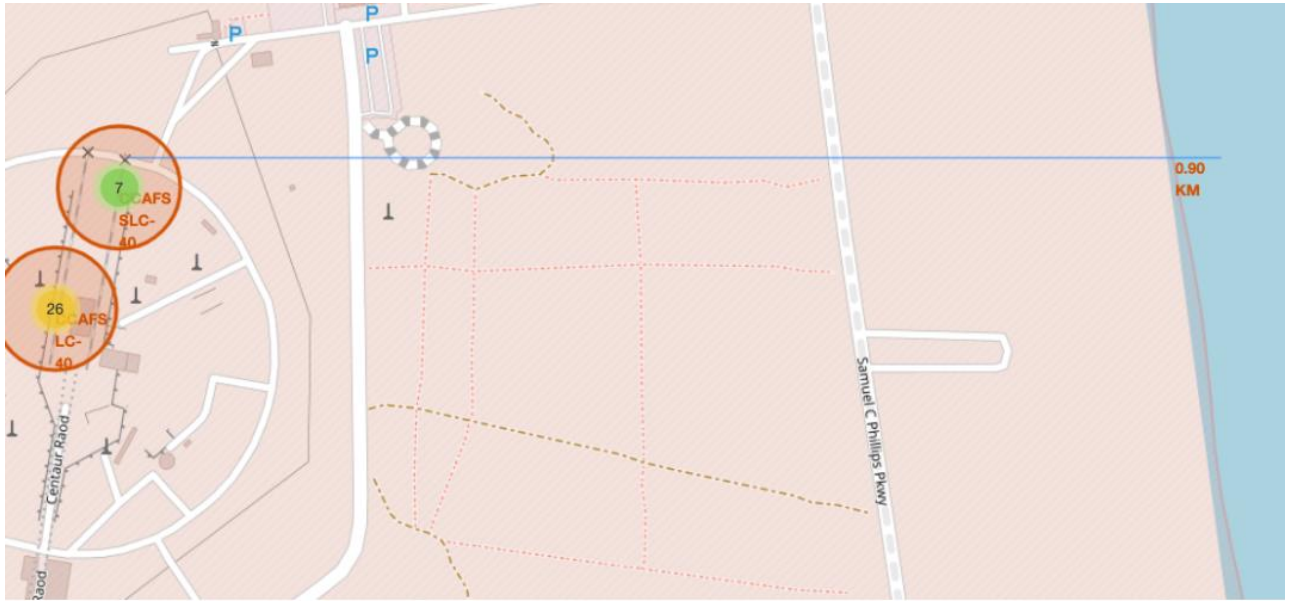
- Explain the important elements and findings on the screenshot

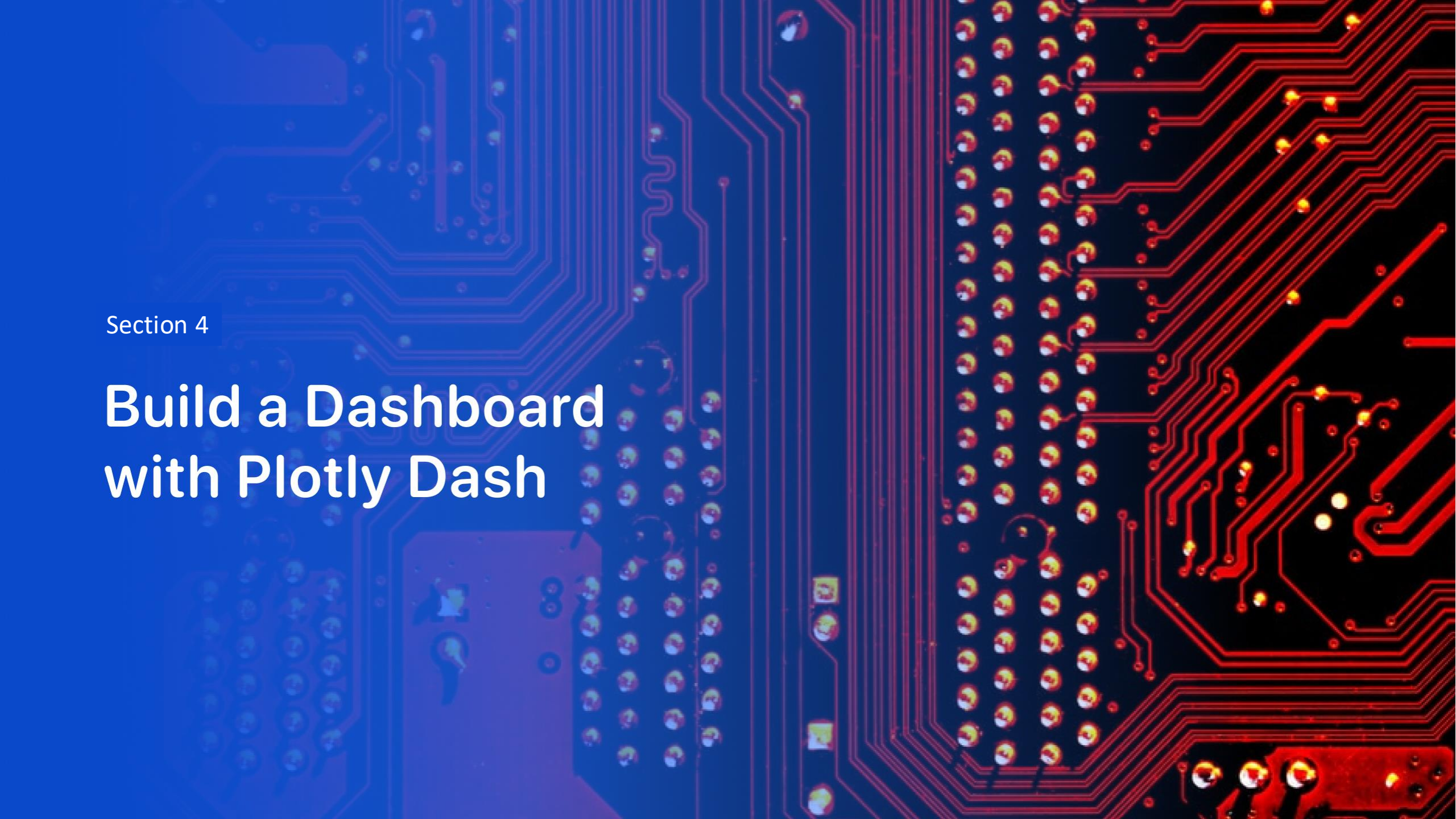
From the color-labeled markers in marker clusters, you should be able to easily identify which launch sites have relatively high success rates.

Distance between Launch Sites to its Proximities

- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

- Explain the important elements and findings on the screenshot



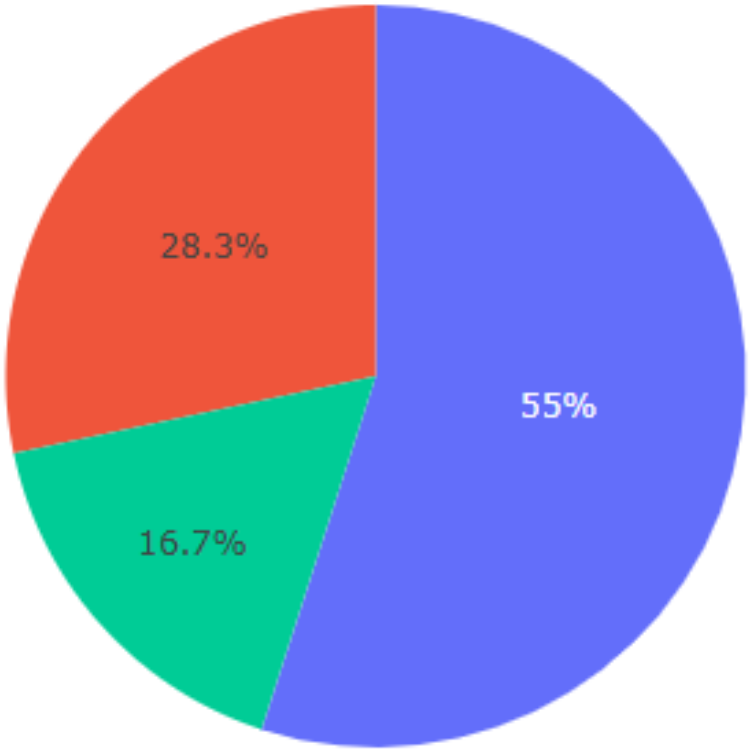

Section 4
Build a Dashboard with Plotly Dash

SpaceX Launch Site

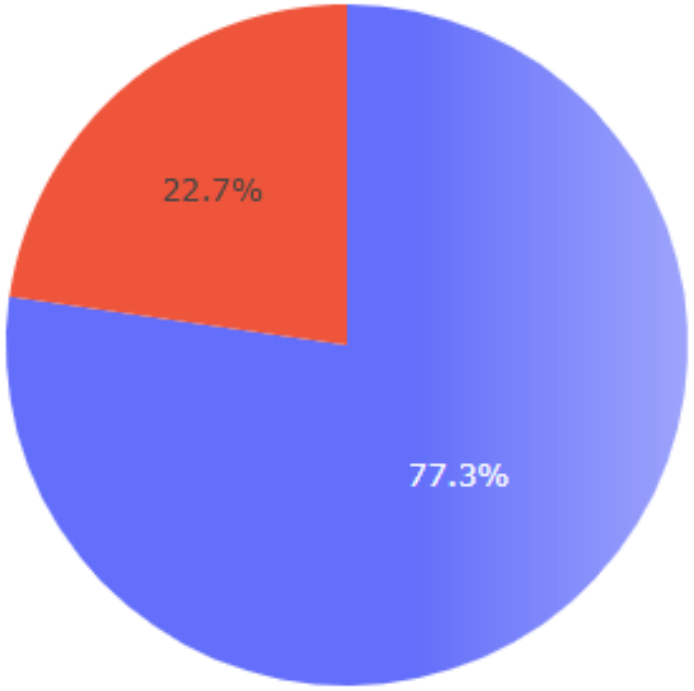
- SpaceX Launch Site

- Explain the important elements and findings on the screenshot

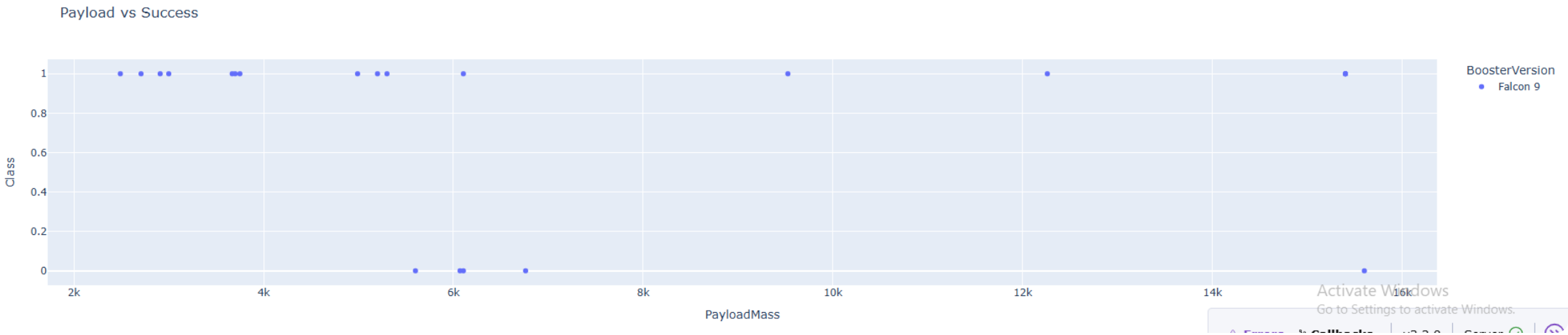
- The pie chart shows the **total successful launches per launch site** (LaunchSite).
- Each slice represents a launch site, and the size of the slice shows how many successful launches happened there.



Launch Site with Highest ration



- Show the screenshot of the piechart for the launch site with highest launch success ratio
- In each slice, KSC-LC-39 have 77.3 have successful launches and 22.7% unsuccessful launches.



Payload vs Launch Outcome

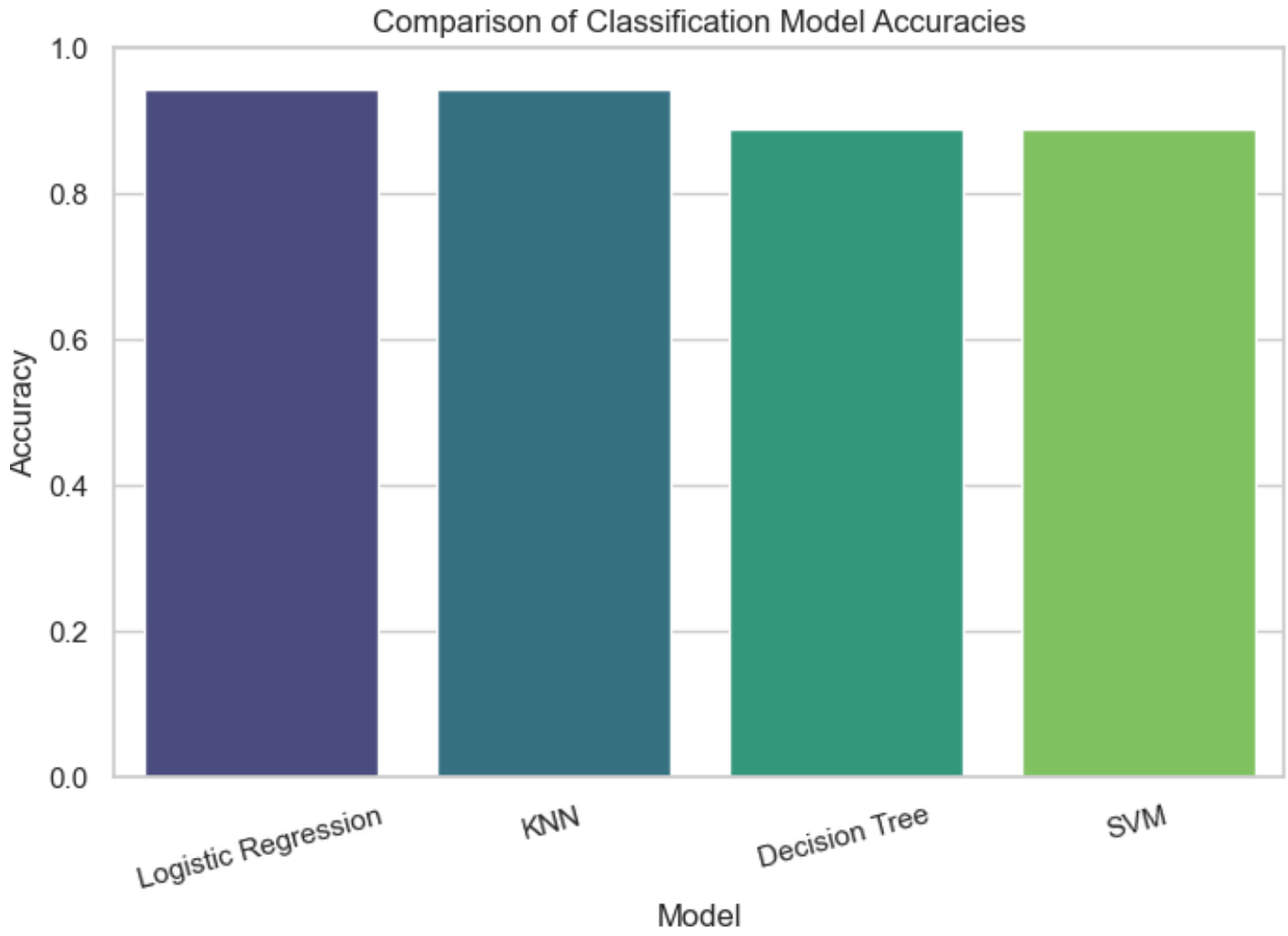
- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.
- sers can choose a payload range in kilograms.
- All other plots update dynamically according to the selected payload range.
- **Insight:** Helps understand how payload mass affects launch success.

Section 5

Predictive Analysis (Classification)

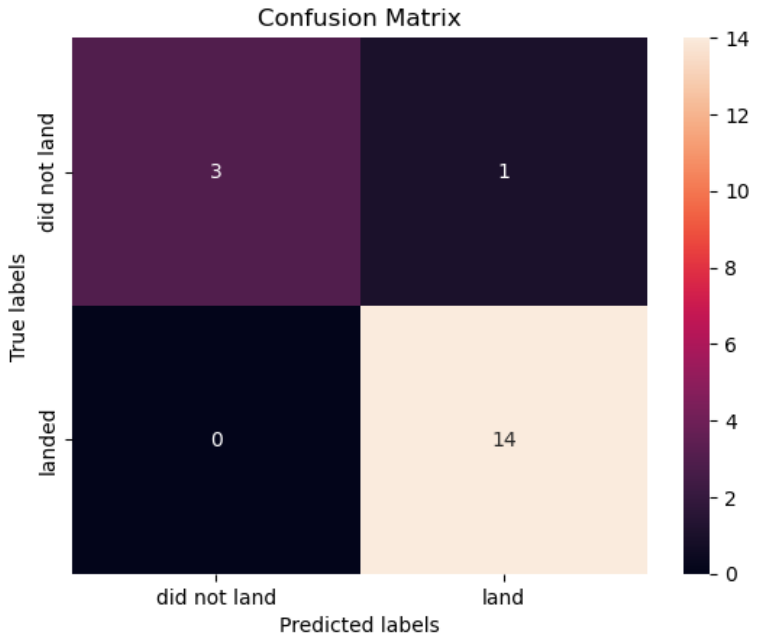
Classification Accuracy

- The Logistic Regression and KNN have the highest classification model accuracies.



Confusion Matrix

Show the confusion matrix of the best performing model with an explanation



Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the problem is false positives.

Overview:

True Postive - 12 (True label is landed, Predicted label is also landed)

False Postive - 3 (True label is not landed, Predicted label is landed)

Conclusions

- The SpaceX Falcon 9 project analyzed historical launch data to identify factors affecting mission success.
- Using data cleaning, EDA, and predictive modeling, we explored relationships between payload, booster versions, and launch sites. An interactive dashboard was created to visualize success rates and trends.
- Among the models tested, the KNN **achieved the highest accuracy** in predicting launch outcomes. T
- his project demonstrates how **data-driven insights** can support better planning and decision-making for future launches.

Appendix

- Dataset Independent Variables : FlightNumbers, BoosterVersion, PayloadMass, LaunchSite, Orbit, Outcome …)
- Dependent Variable: Class (1-Successful Landing, 0-Unsuccessful Landing)

- # Run the app
- if __name__ == '__main__':
- app.run(debug=True)

Thank you!