IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Leslie Luo
22 Feb 2024

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

- Project background and context
- Question: Can We Predict the Successful Landing of Falcon 9's First Stage to Estimate Launch Costs and Inform Competitive Bidding Strategies?

Section 1

Methodology

Methodology

Executive Summary

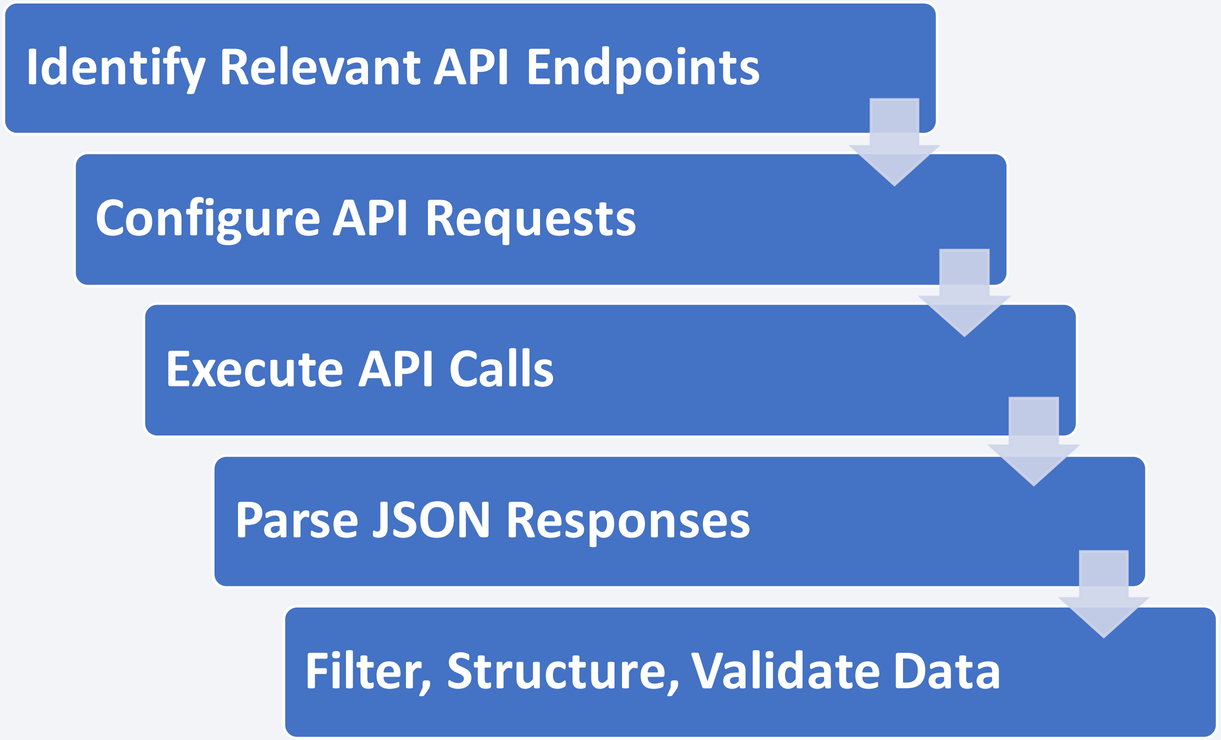
- Data collection methodology:
  - Use API and web-scrape
- Perform data wrangling
  - Standardise numeric variables and convert categories into dummies
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Use best score and confusion matrix to assess the accuracy of the models

Data Collection

- How data sets were collected.

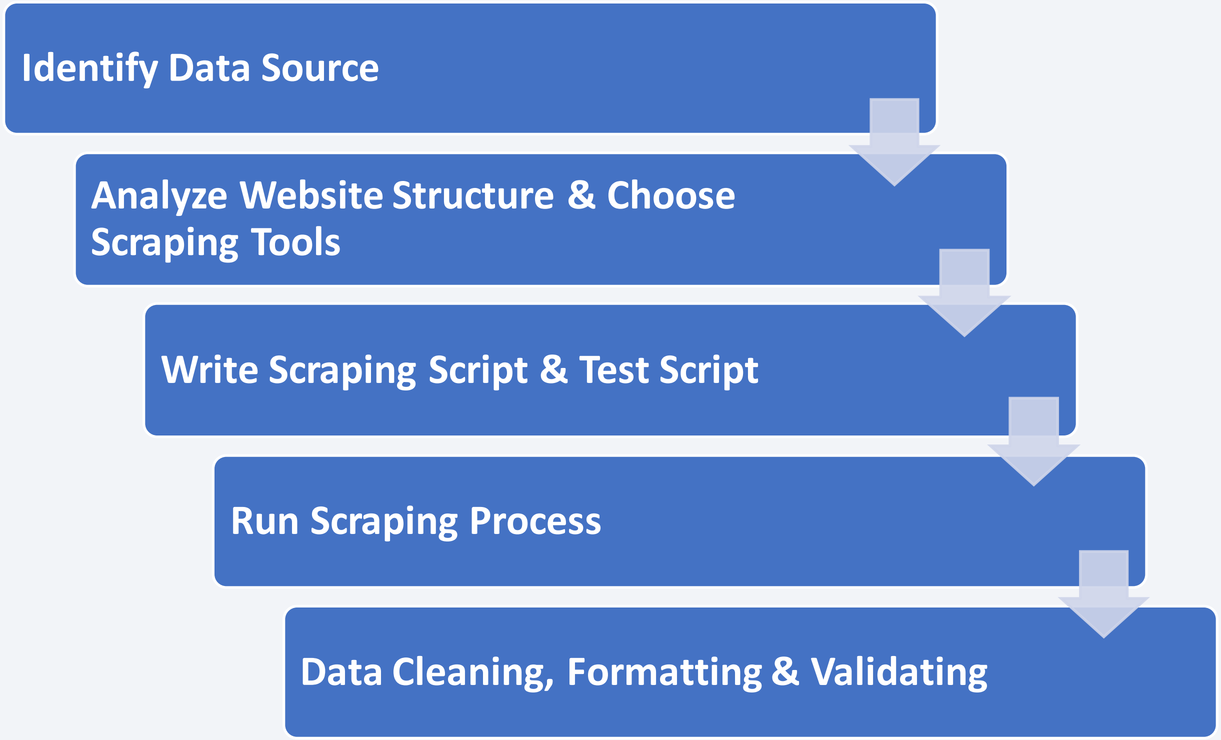
Data Collection – SpaceX API

- The GitHub URL of the completed SpaceX API calls notebook



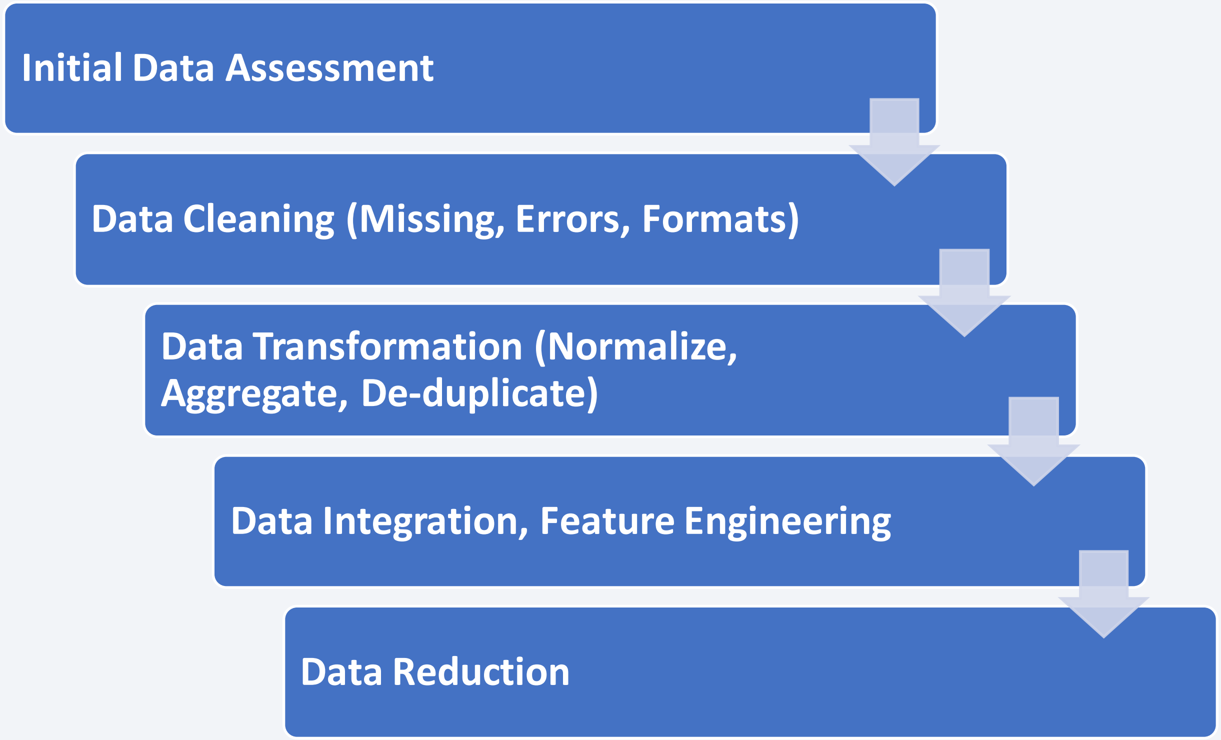
Data Collection - Scraping

- The GitHub URL of the completed web scraping notebook



Data Wrangling

- The GitHub URL of completed data wrangling related notebooks



EDA with Data Visualization

- The GitHub URL of completed EDA with data visualization notebook
- Chart Summaries & Purposes
- Scatterplots:
    - Flight Number vs. Launch Site: Identifies launch distribution over time and by site.
    - Payload vs. Launch Site & Orbit Type: Shows payload preferences for sites and orbits.
    - Flight Number vs. Orbit Type: Reveals mission focus changes over time.
- Barchart:
    - Success Rate by Orbit Type: Compares success rates across orbits, highlighting reliability.
- Line Chart:
    - Yearly Launch Success Trend: Traces success trends over years, indicating performance changes.

Why These Charts?

- Scatterplots: Best for visualizing relationships between two variables, highlighting patterns or clusters.
- Barchart: Ideal for categorical comparison, showing differences in success rates by orbit.
- Line Chart: Effective for trend analysis over time, revealing changes in launch success.

EDA with SQL

- The GitHub URL of completed EDA with SQL notebook

SQL queries summary

- Table Creation:
  - Created SPACEXTABLE from SPACEXTBL where dates exist.
- Launch Site Analysis:
  - Queried first 5 launches from 'CCA' sites.
- Payload Analysis:
  - Total mass for NASA (CRS).
  - Average mass for 'F9 v1.1' booster.
- Landing Outcome Insights:
  - Earliest successful ground pad landing.
  - Booster versions for successful drone ship landings with payloads between 4000-6000kg.
- Landing Success & Failures:
  - Counted successes and failures overall.
  - Specifically counted successes and failures.
- Heaviest Payload Booster:
  - Identified booster for the maximum payload mass.
- 2015 Drone Ship Failures:
  - Details of failures including month, booster, and site.
- Landing Outcome Trends (2010-2017):
  - Grouped and ordered landing outcomes by count.

Build an Interactive Map with Folium

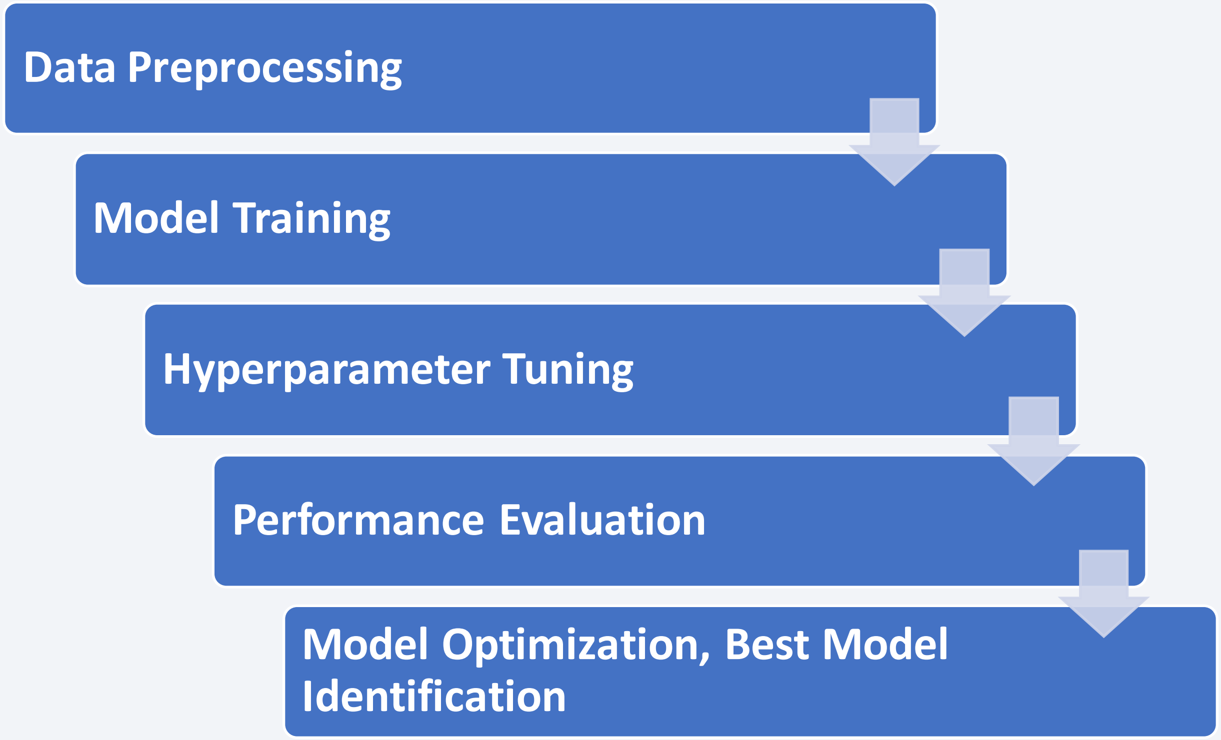
- The GitHub URL of completed interactive map with Folium map
- Summarize map objects & Why added those objects
  - Marker: Marks specific locations with customizable icons.
  - Circle: Draws circles with defined radii to highlight areas.
  - Mouse Position: Shows real-time latitude and longitude under the mouse cursor.
  - Distance Marker: Marks distances along paths or lines.
  - Lines: Connects multiple points with lines to depict routes or boundaries.

Build a Dashboard with Plotly Dash

- The GitHub URL of completed Plotly Dash lab
- Summarize plots/graphs and interactions & Why
  - Success Pie Chart with Site Dropdown
    - Visualizes launch success rates by site.
    - Enables comparison and performance assessment.
  - Payload Range Slider
    - Allows filtering of launches by payload weight.
    - Facilitates tailored analysis on selected payload ranges.
  - Success-Payload Scatter Chart
    - Plots relationship between payload weight and launch success.
    - Identifies trends, supporting data-driven insights.

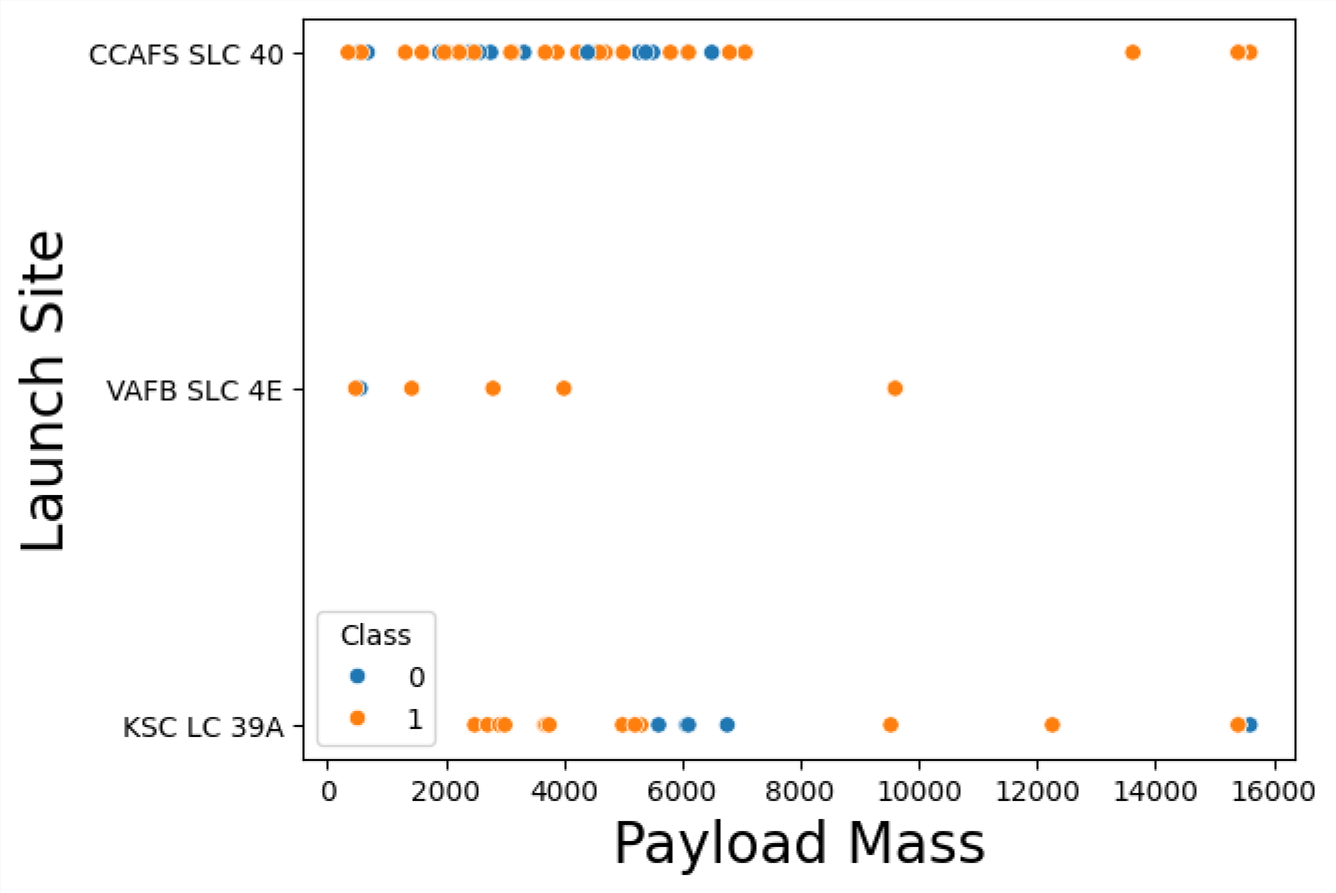
Predictive Analysis (Classification)

- The GitHub URL of completed predictive analysis lab



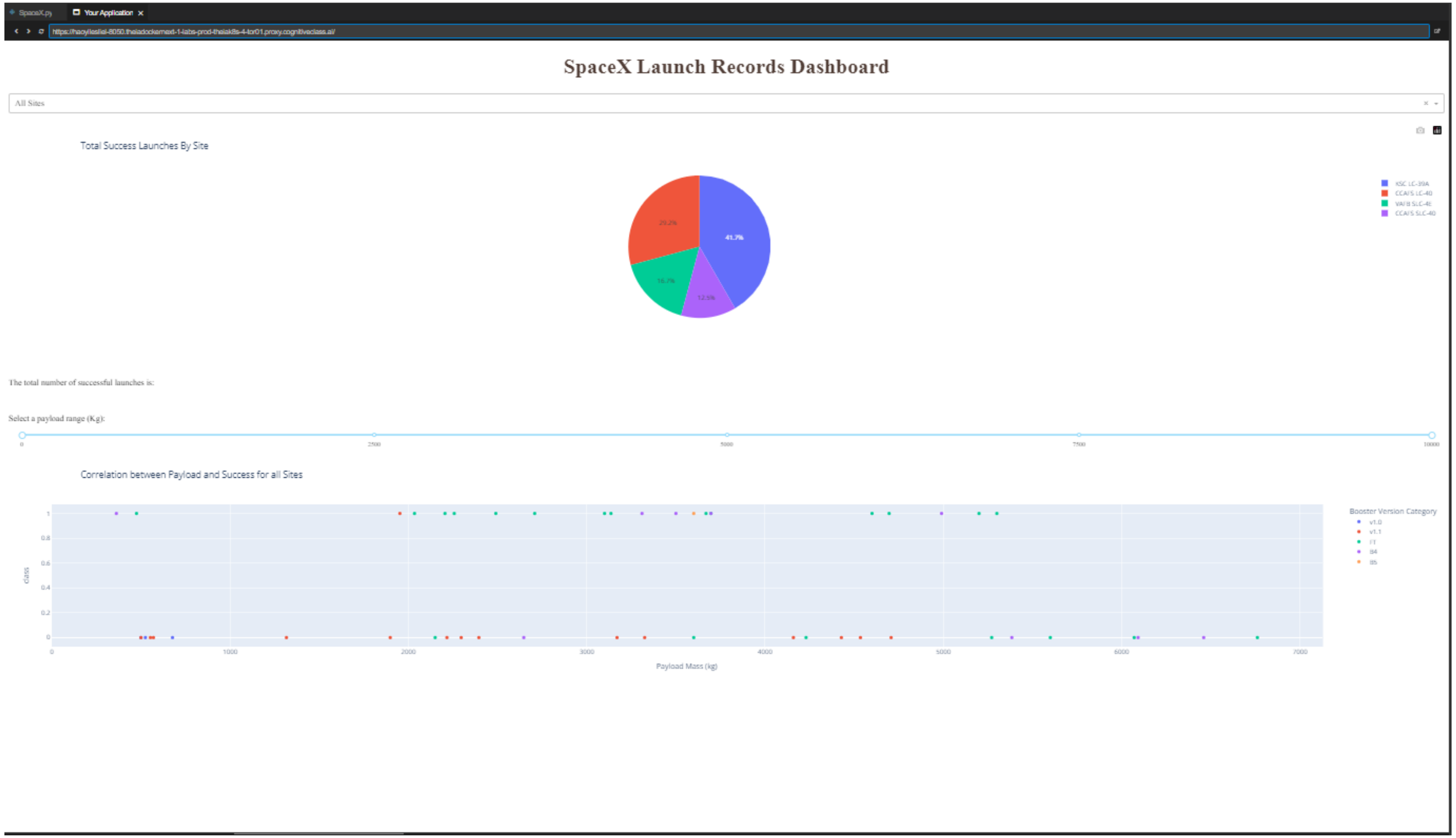
Results

- Exploratory data analysis results



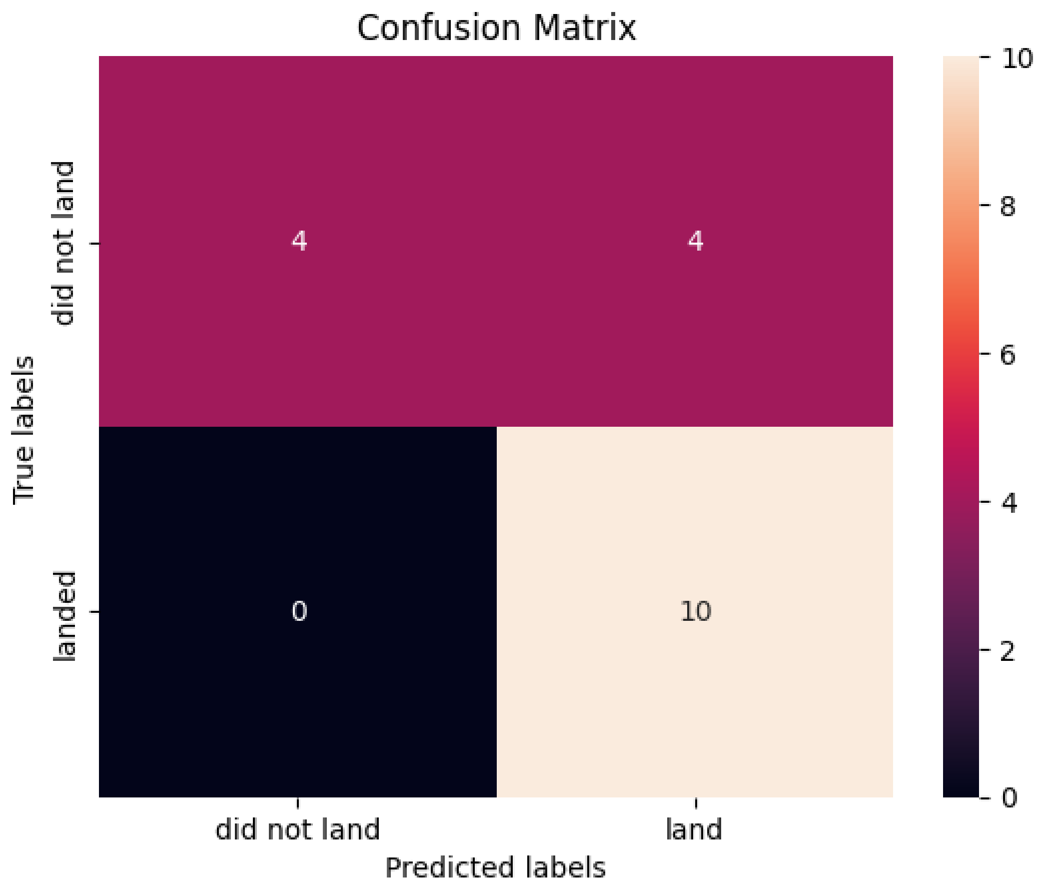
Results

- Interactive analytics demo in screenshots



Results

- Predictive analysis results

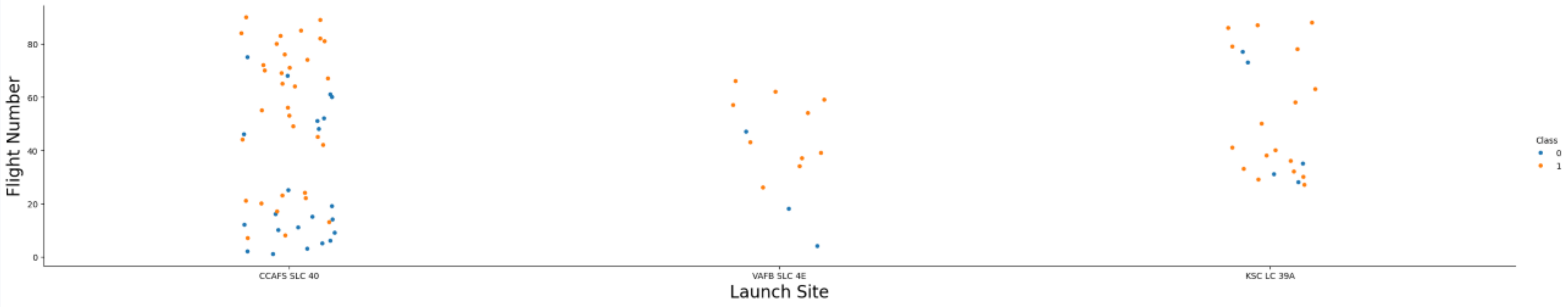


Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

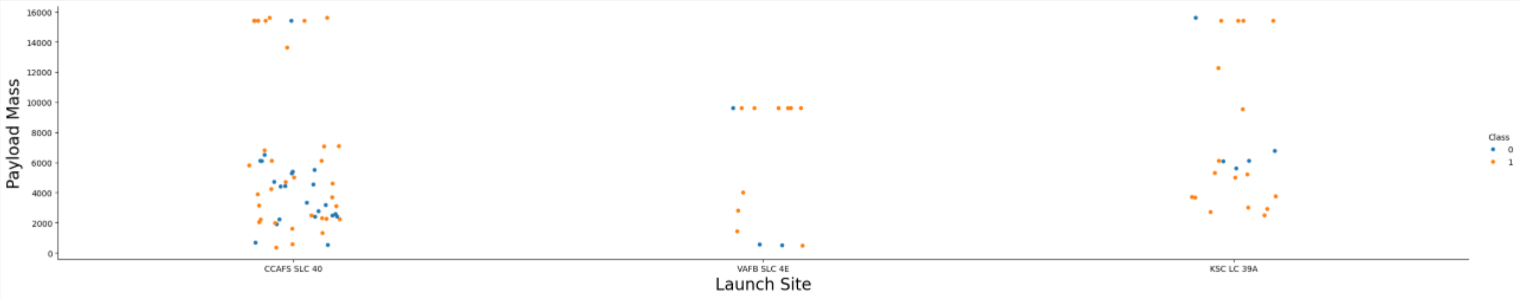
- Show a scatter plot of Flight Number vs. Launch Site



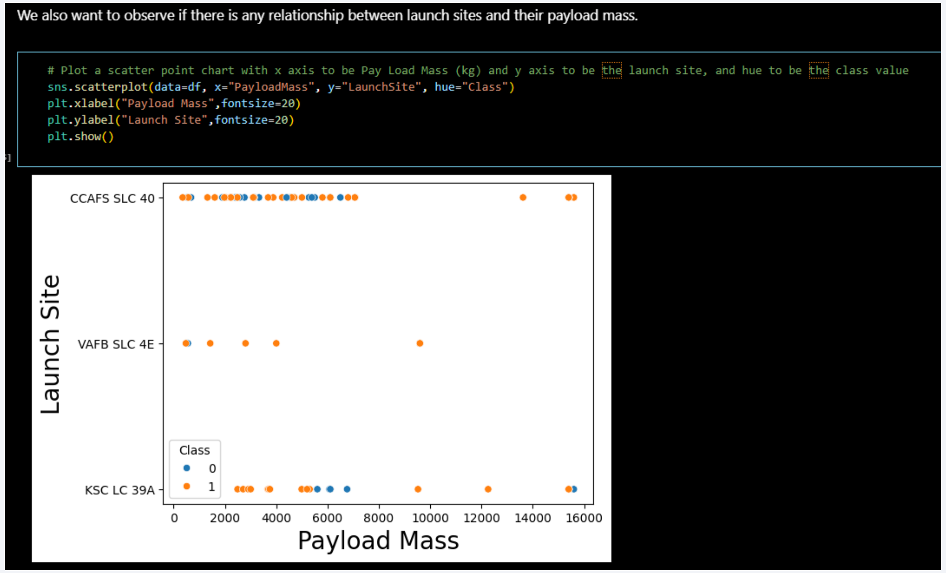
- Show the screenshot of the scatter plot with explanations

Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site

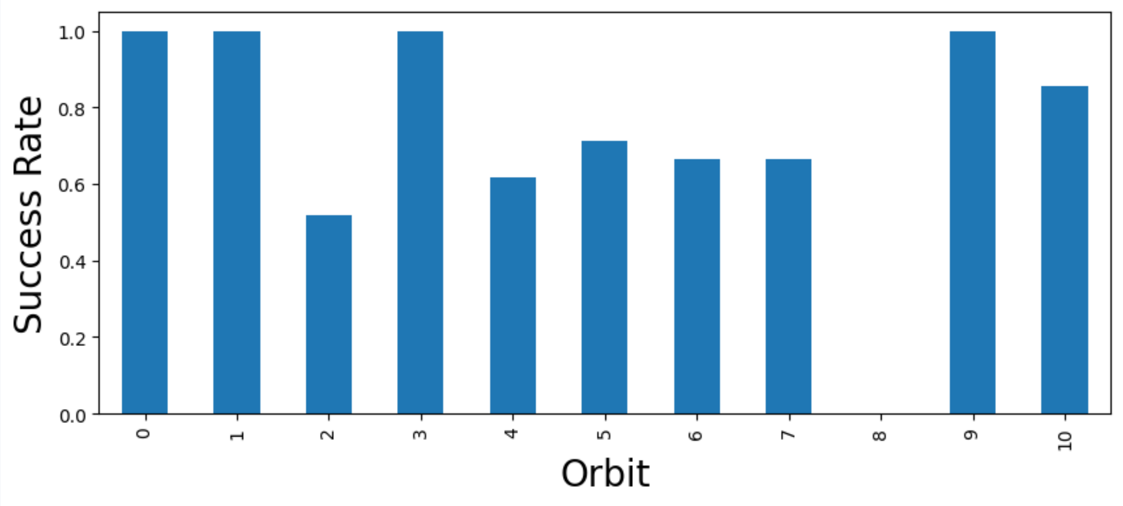


- Show the screenshot of the scatter plot with explanations

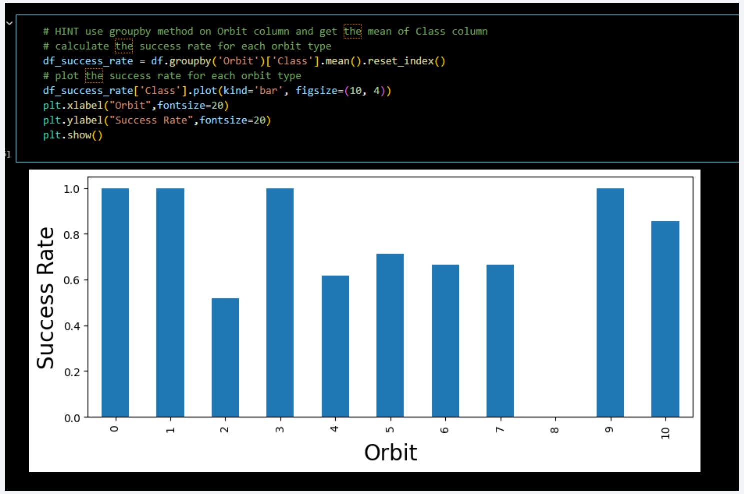


Success Rate vs. Orbit Type

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

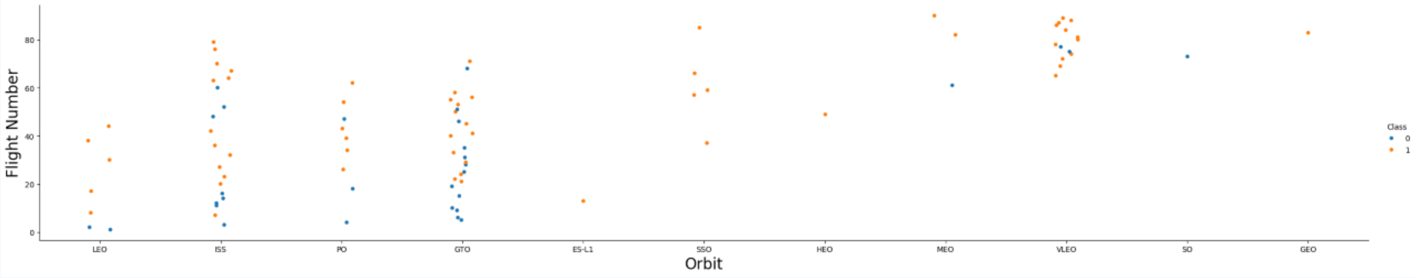


- Show the screenshot of the scatter plot with explanations

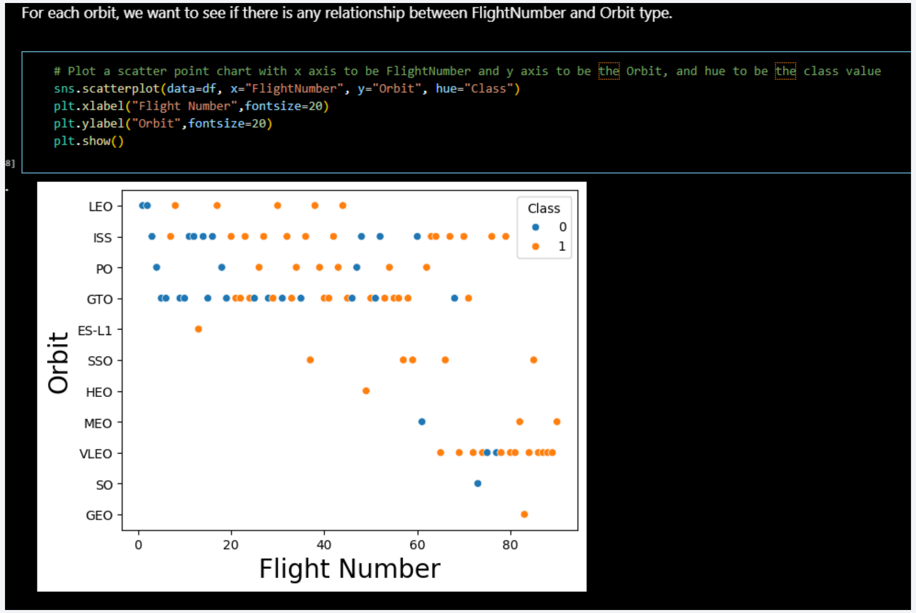


Flight Number vs. Orbit Type

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



- Show the screenshot of the scatter plot with explanations

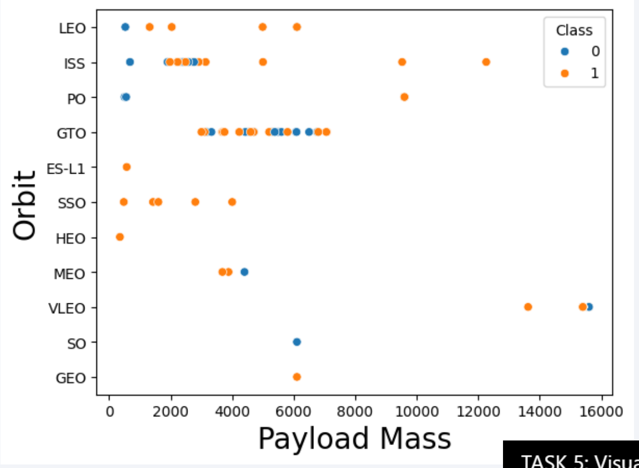


For each orbit, we want to see if there is any relationship between FlightNumber and Orbit type.

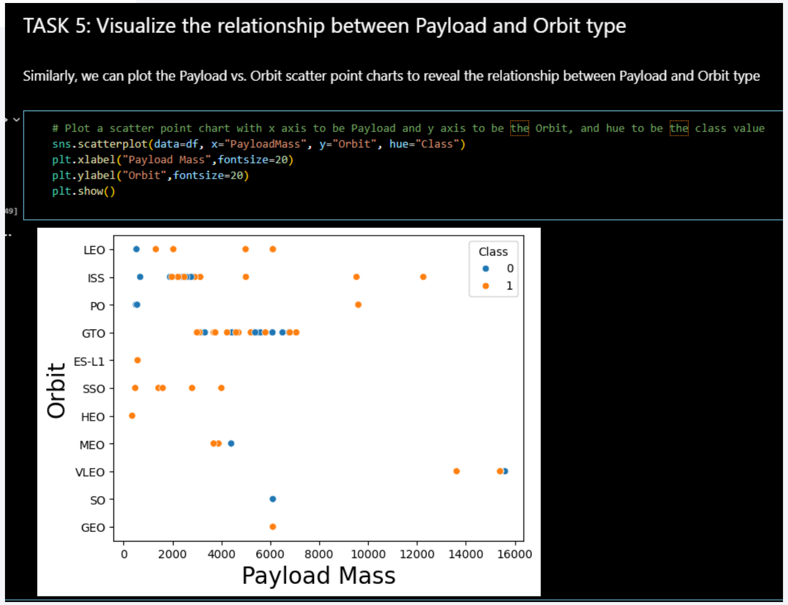
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.scatterplot(data=df, x="FlightNumber", y="Orbit", hue="Class")
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```

Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type

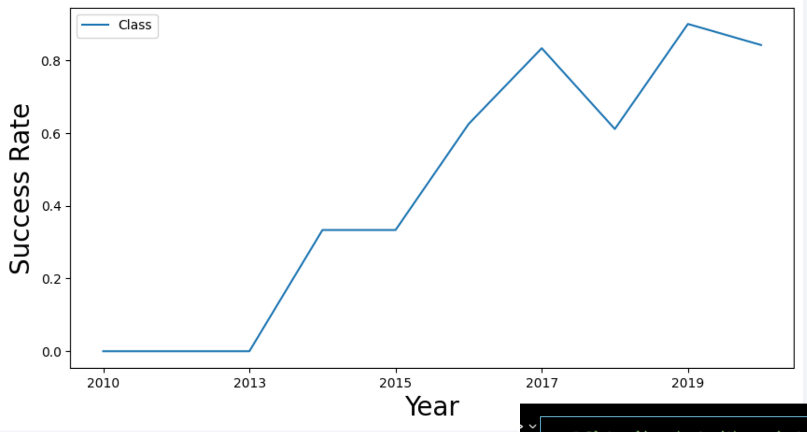


- Show the screenshot of the scatter plot with explanations

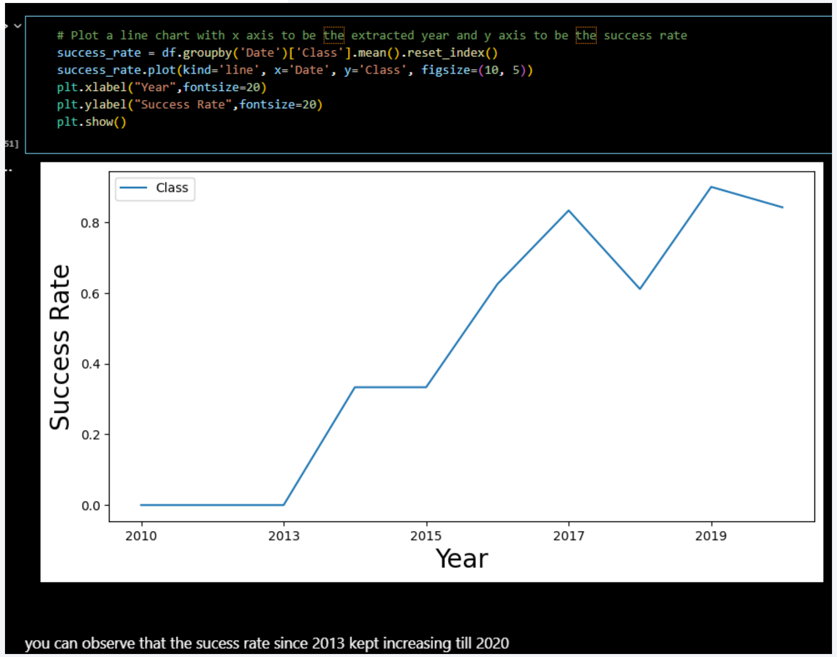


Launch Success Yearly Trend

- Show a line chart of yearly average success rate



- Show the screenshot of the scatter plot with explanations



All Launch Site Names

- Find the names of the unique launch sites
- the query below shows the unique launch sites in the space mission

```
# display the names of the unique launch sites in the space mission
%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'

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

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

```
# display 5 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
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO
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)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

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

```
# display the total payload mass carried by boosters launched by NASA (CRS)
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where CUSTOMER = 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

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

Task 4

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

```
# display average payload mass carried by booster version F9 v1.1
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where BOOSTER_VERSION = 'F9 v1.1'
```

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

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

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

Task 5

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

Hint:Use min function

+ Code + Markdown

[14]

```
# list the date when the first successful landing outcome in ground pad was acheived.
%sql select min(Date) from SPACEXTABLE where LANDING_OUTCOME = 'Success (ground pad)'
```

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

min(Date)
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

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

```python
# list the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
%sql select distinct(BOOSTER_VERSION) from SPACEXTABLE where LANDING_OUTCOME = 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

Python

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

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

Task 7

List the total number of successful and failure mission outcomes

```
[16]
# list the total number of successful and failure mission outcomes
%sql select count(LANDING_OUTCOME) from SPACEXTABLE where LANDING_OUTCOME like 'Success%' or LANDING_OUTCOME like 'Failure%'
```

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

count(LANDING_OUTCOME)
71

Boosters Carried Maximum Payload

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

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

```
# list the names of the booster_versions which have carried the maximum payload mass. Use a subquery
%sql select distinct(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

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

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.

```python
# list the failed landing_outcomes in drone ship, the month names, their booster versions, and launch site names for the months in year 2015
# use the whole cell for this query rather than a single line
%sql select     LANDING_OUTCOME,     strftime('%m', Date) as Month,     BOOSTER_VERSION,     Launch_Site from SPACEXTABLE where LANDING_OUTCOME like 'Failure (dr
```

Python

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

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

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.

```python
# 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 count(LANDING_OUTCOME) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by LANDING_OUTCOME order by count(LANDING_OUTCOME) des
```

Python

* sqlite:///my_data1.db
Done.

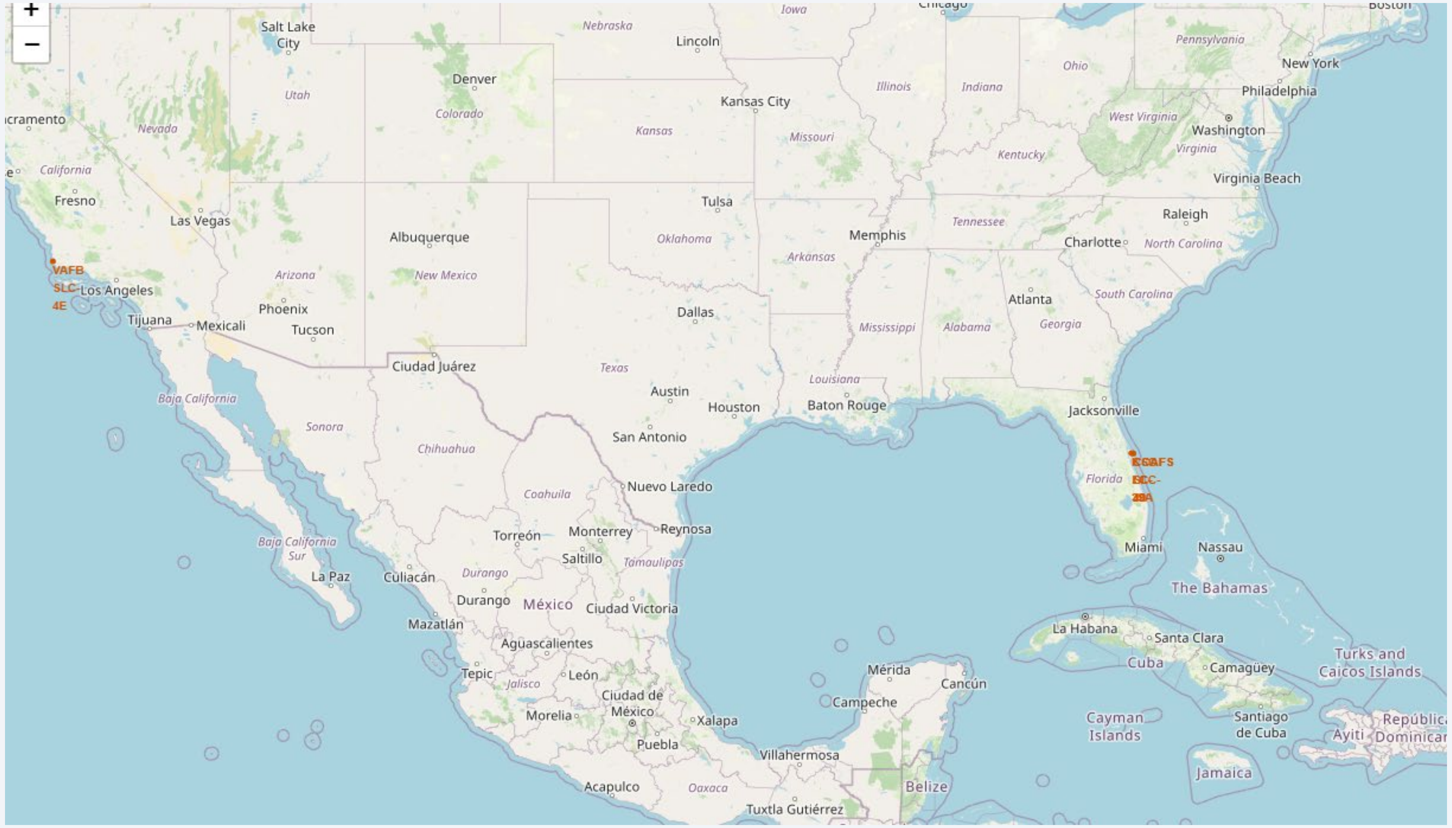
count(LANDING_OUTCOME)
10
5
5
3
3
2
2
1

Section 3

Launch Sites Proximities Analysis

Launch Sites in U.S.

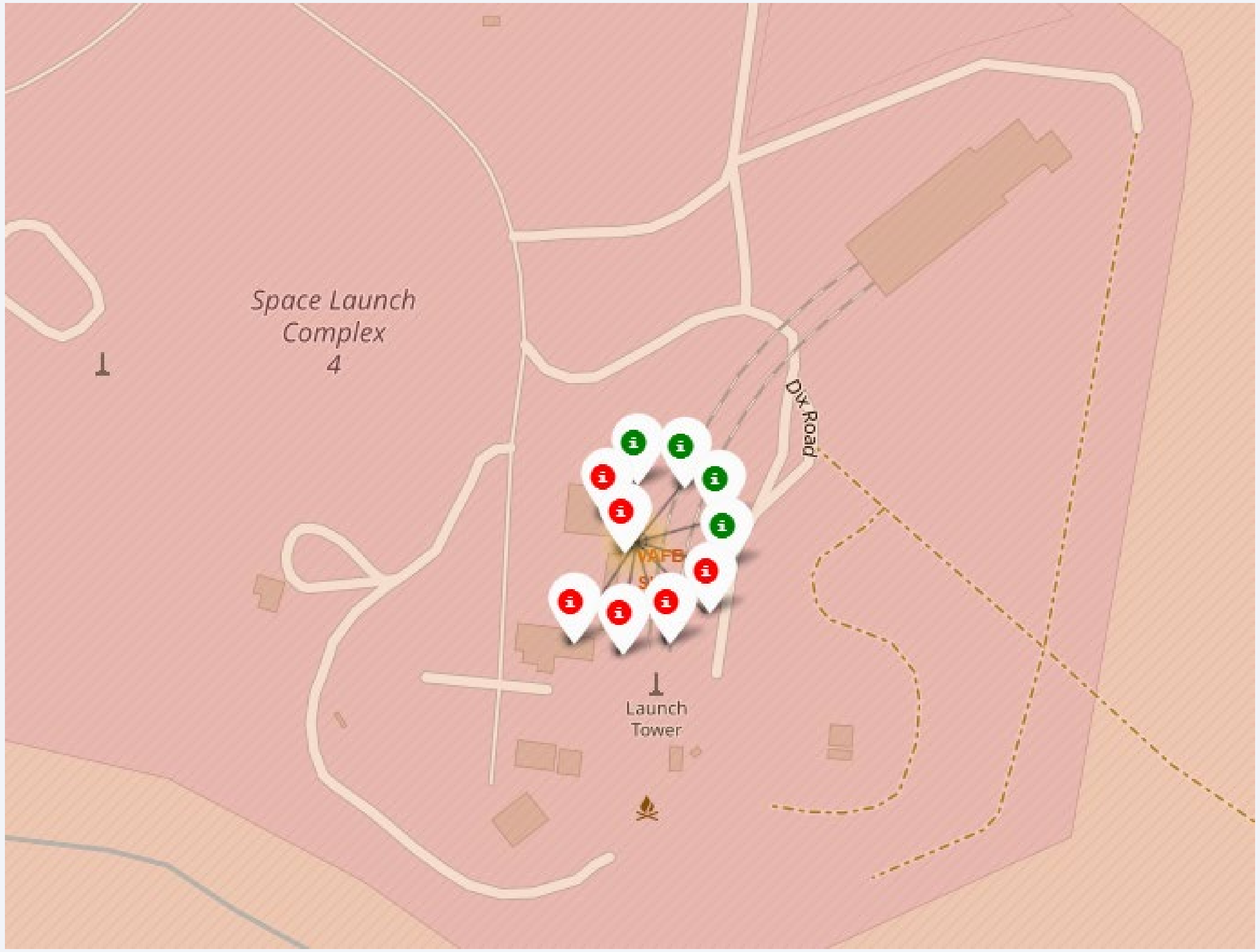
- All launch sites' location



- Findings:
  - All locations are either in California or Florida
  - All launch sites are in coastal areas

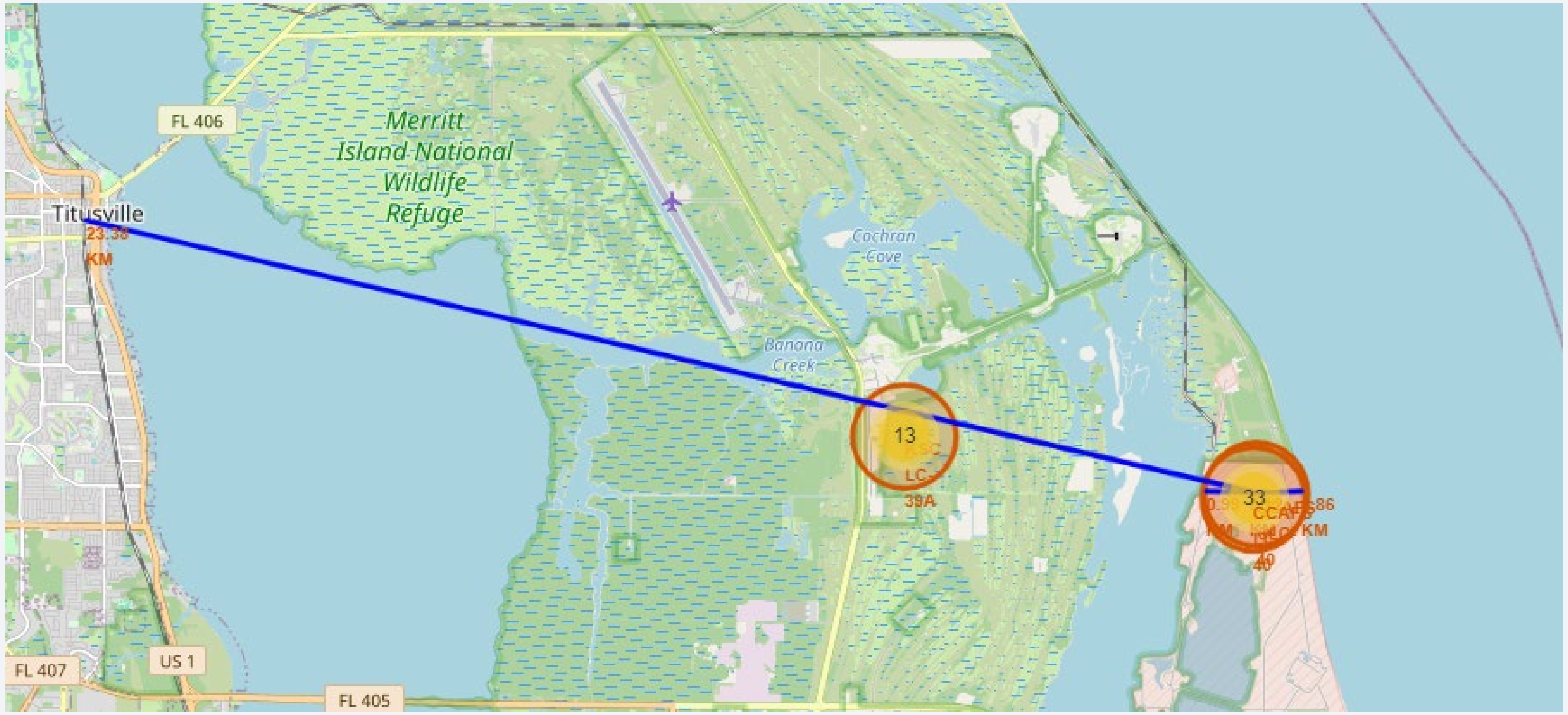
Coloured-Labelled Launch Sites

- Findings:
- Among the 10 launches at this site, 4 were successful while 6 failed.



Distance to railway, highway and coastline

- Findings:
- The launch site is close to railway, highway and coastline, but far away from cities.

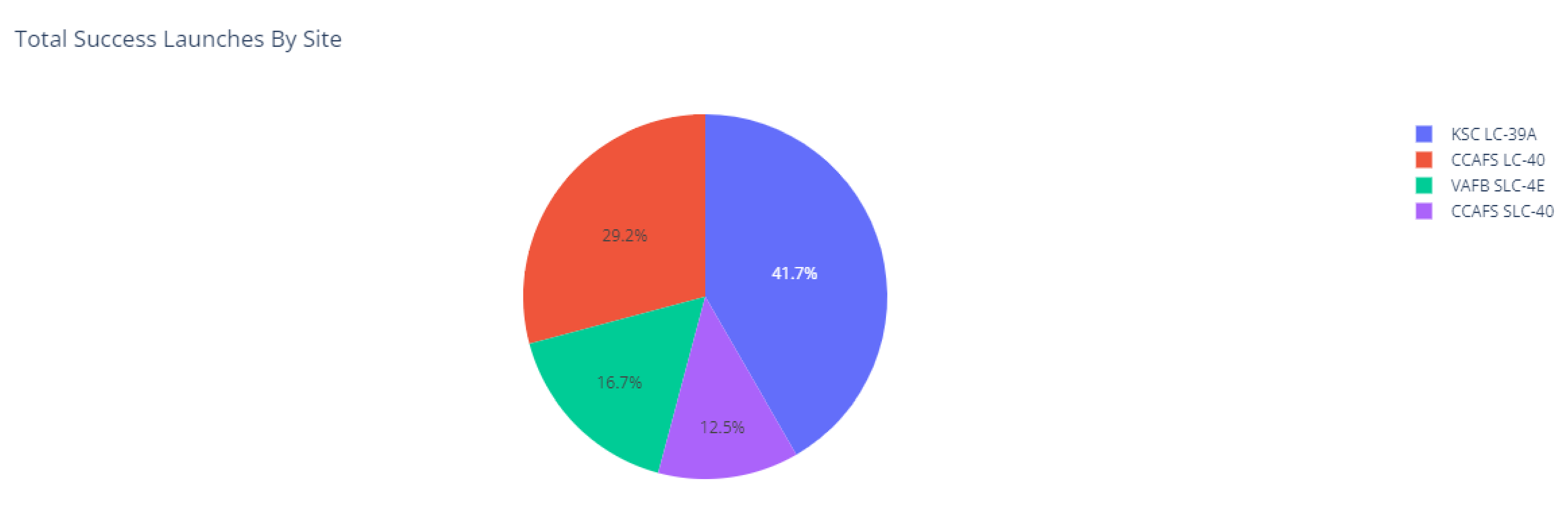


Section 4

Build a Dashboard with Plotly Dash

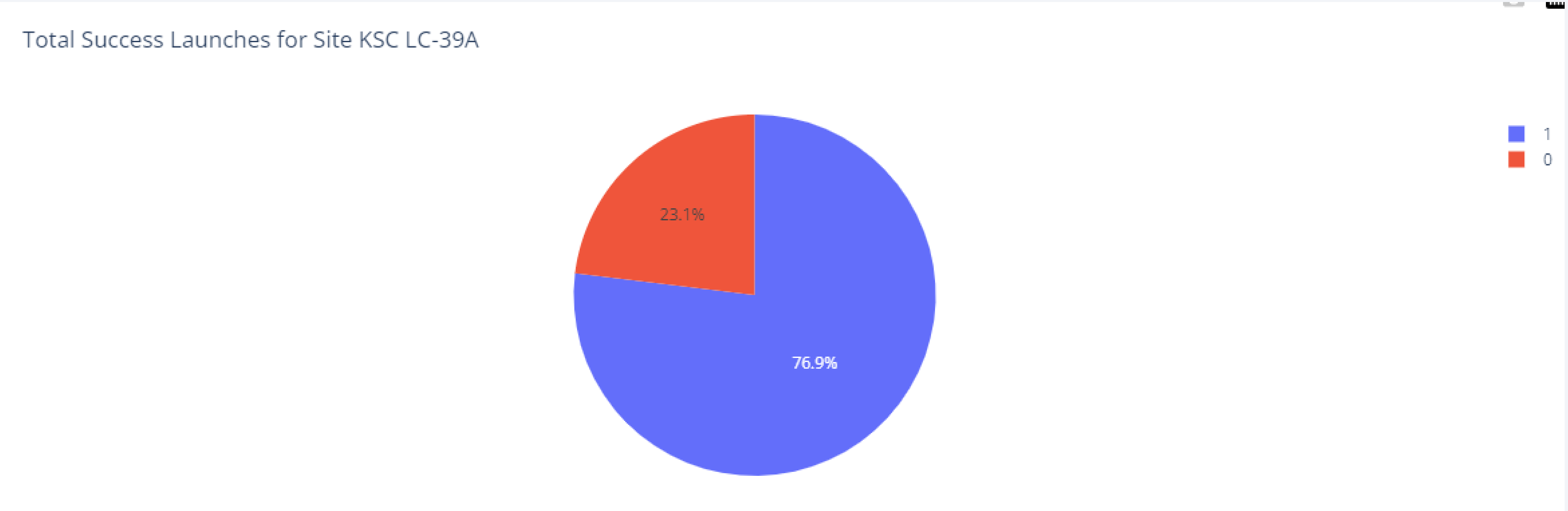
Launch Success

- KSC LC-39A has the most successful launches



Launches records in KSC LC-39A

- The success rates for the KSC LC-39A is 76.9%



Launch records by boosters and payload

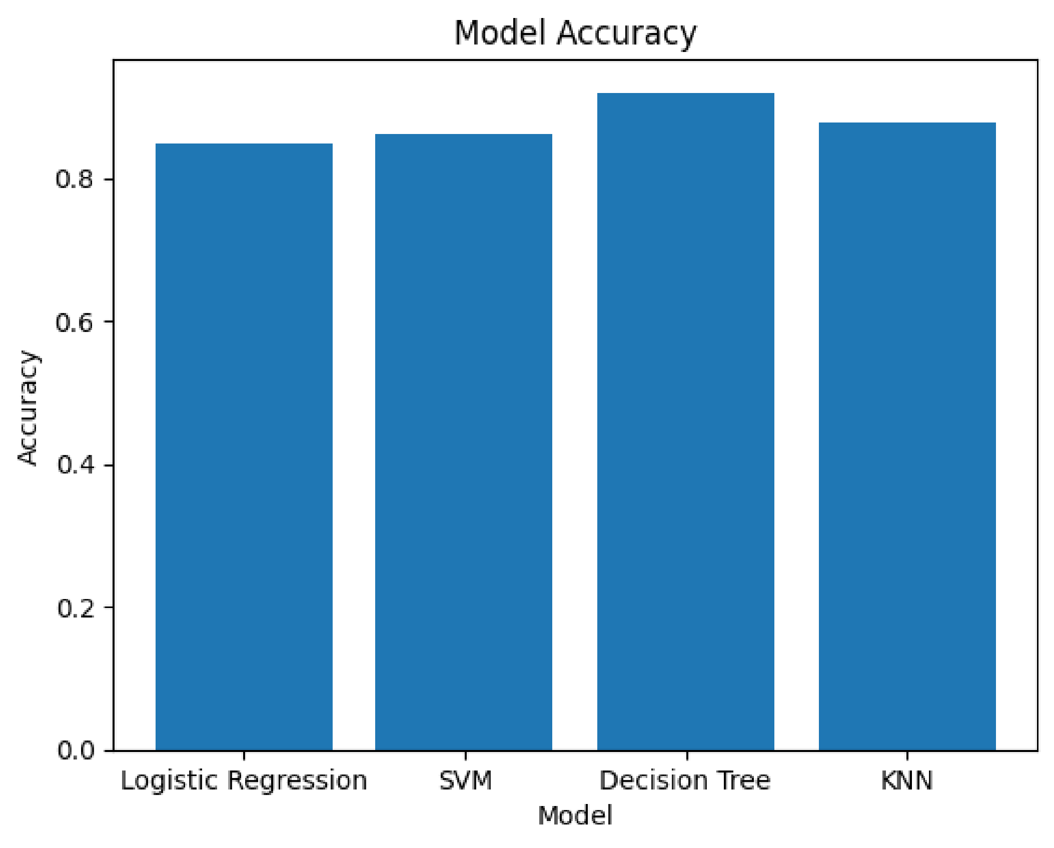
- Findings: Booster version FT has the highest success rate.

Section 5

Predictive Analysis (Classification)

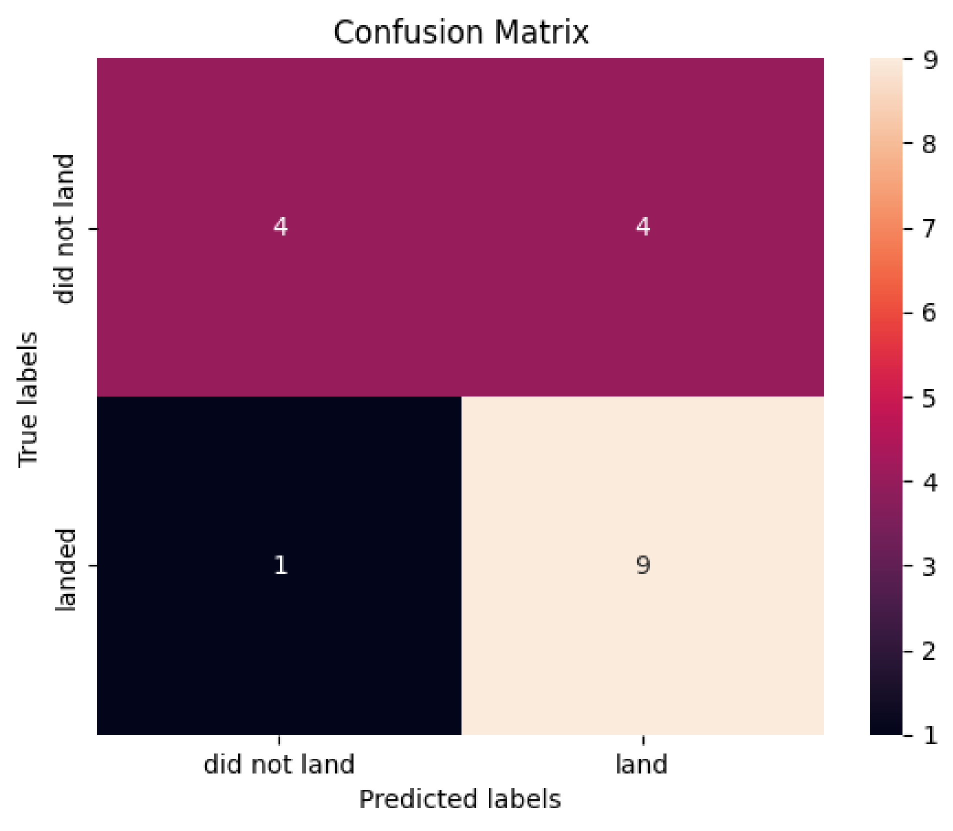
Classification Accuracy

- The Decision Tree Model the highest classification accuracy



Confusion Matrix of Decision Tree Model

- **Top-left square (True Negative)**: The model predicted 4 instances of the class 'did not land' correctly.
- **Top-right square (False Positive)**: The model incorrectly predicted 4 instances as 'land' when they actually 'did not land'.
- **Bottom-left square (False Negative)**: The model incorrectly predicted 1 instance as 'did not land' when it actually 'landed'.
- **Bottom-right square (True Positive)**: The model predicted 9 instances of the class 'land' correctly.



Conclusions

- Control the payload to a moderate size for higher success rate
- Use the FT booster for a higher success rate
- Launch in the site KSC LC-39A

Appendix

- Notebook outputs/data sets
- https://github.com/haoyileslie/Coursera/blob/main/dataset_part_1.csv
- https://github.com/haoyileslie/Coursera/blob/main/dataset_part_2.csv
- https://github.com/haoyileslie/Coursera/blob/main/dataset_part_3.csv
- https://github.com/haoyileslie/Coursera/blob/main/spacex_web_scraped2.csv

Confusion Matrix

- From the confusion matrix, we can calculate various performance metrics, such as accuracy, precision, recall, and F1 score, which provide more details on the model's predictive performance
  - Accuracy: (True Positives + True Negatives) / Total Predictions
  - Precision: True Positives / (True Positives + False Positives)
  - Recall (Sensitivity or True Positive Rate): True Positives / (True Positives + False Negatives)
  - F1 Score: Harmonic Mean of Precision and Recall

Thank you!