IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<NG KIM FONG>
<23 DECEMBER 2023>

Outline

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

Executive Summary

In this capstone project, I employed a comprehensive approach to predict the successful landing of the Falcon 9 first stage. Utilizing a combination of API and Web Scraping techniques, I gathered pertinent data, subsequently enhancing its quality through rigorous Exploratory Data Analysis (EDA). Employing SQL, I meticulously processed and explored the dataset, extracting valuable insights through basic statistical analysis and Feature Engineering.

For a visually insightful presentation, I leveraged Python packages including matplotlib, seaborn, folium, and plotly dash to conduct Data Visualization. In the culmination of the project, I executed Machine Learning predictions employing Logistic Regression, SVM, Decision Tree, and KNN algorithms. The best model is Decision Tree with the highest accuracy score and work well to distinguish the classes as shown in the confusion matrix.

Introduction

The commercial space age is upon us, with companies making space travel more accessible for everyone. SpaceX, in particular, has achieved remarkable feats, including spacecraft missions to the International Space Station, the establishment of the Starlink satellite internet constellation, and successful manned space missions. The key to SpaceX's success lies in its cost-effective approach, prominently showcased in the Falcon 9 rocket launches, priced at $62 million, merely a fraction of competitors' costs, owing to the revolutionary reusability of the first stage.

SpaceX possesses the capability to recover and reuse this substantial component, contributing significantly to cost savings. The challenge at hand is to determine if the first stage will land successfully, directly impacting the overall launch cost.

In this capstone project, I assume the role of a data scientist for Space Y, a new rocket company aspiring to rival SpaceX. The outcomes of this project will not only furnish Space Y with precise launch cost estimations but also offer strategic insights into the reusability of SpaceX's first stage.

Section 1

Methodology

Methodology

Executive Summary

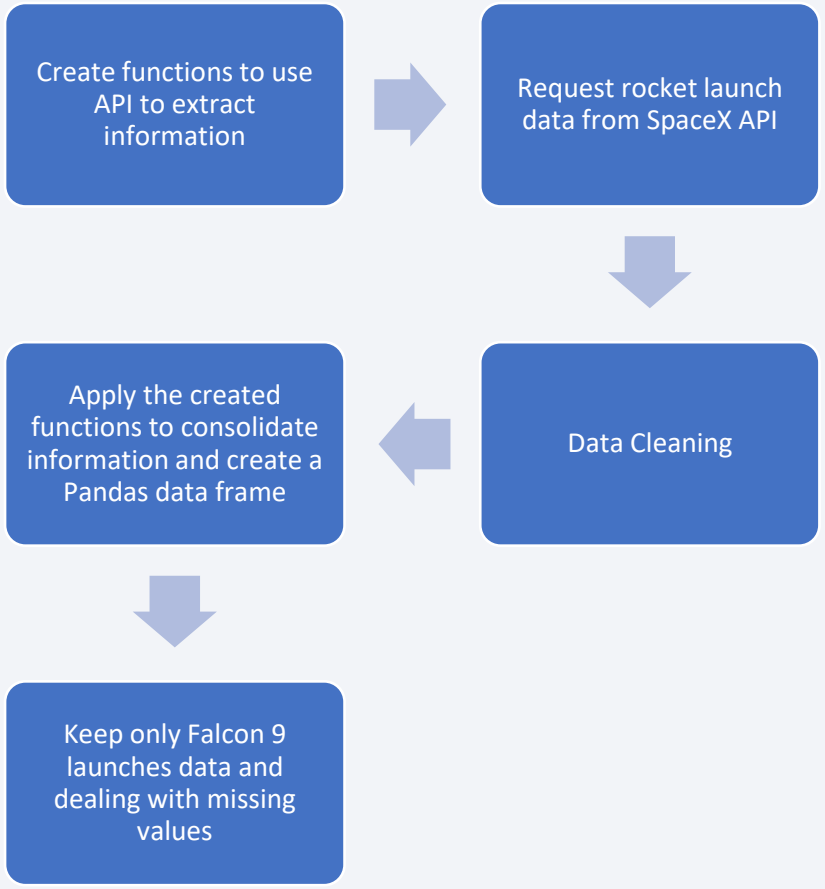
- Data collection methodology:
  - Data was collected using API and Web Scraping techniques.
- Perform data wrangling
  - Identified the variables to be used and converted the outcomes into Training Label.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardize the data.
  - Split into training data and test data.
  - Find the best hyperparameters for classification models.
  - Find the models perform best using test data.

Data Collection

- To request and parse SpaceX launch data from various API sources using the GET request.
- Keep only the selected features and perform data cleaning.
- Filter the dataframe to only include Falcon 9 launches.
- Dealing with missing values by replacing mean value for Pay Load.
- Web Scraping from Wikipedia for Falcon 9 launch records using BeautifulSoup.

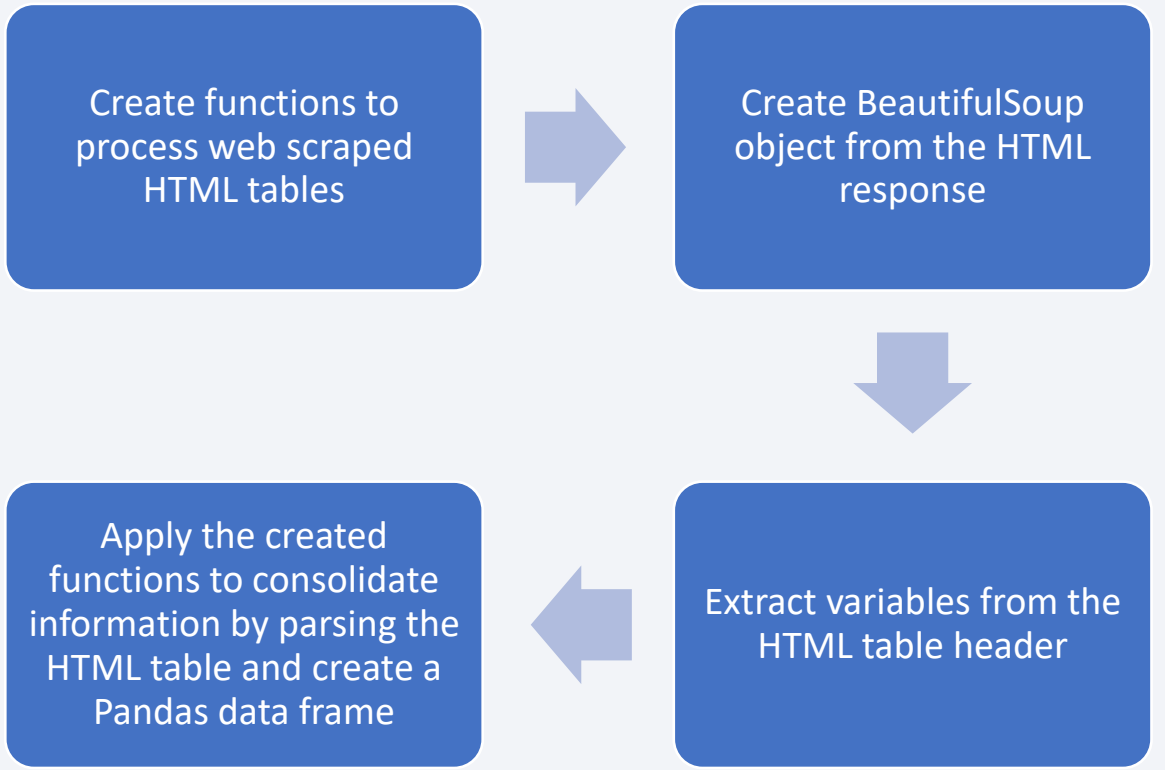
Data Collection – SpaceX API

- The data collection via API as per flowcharts
- The GitHub URL of the completed SpaceX API calls notebook as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



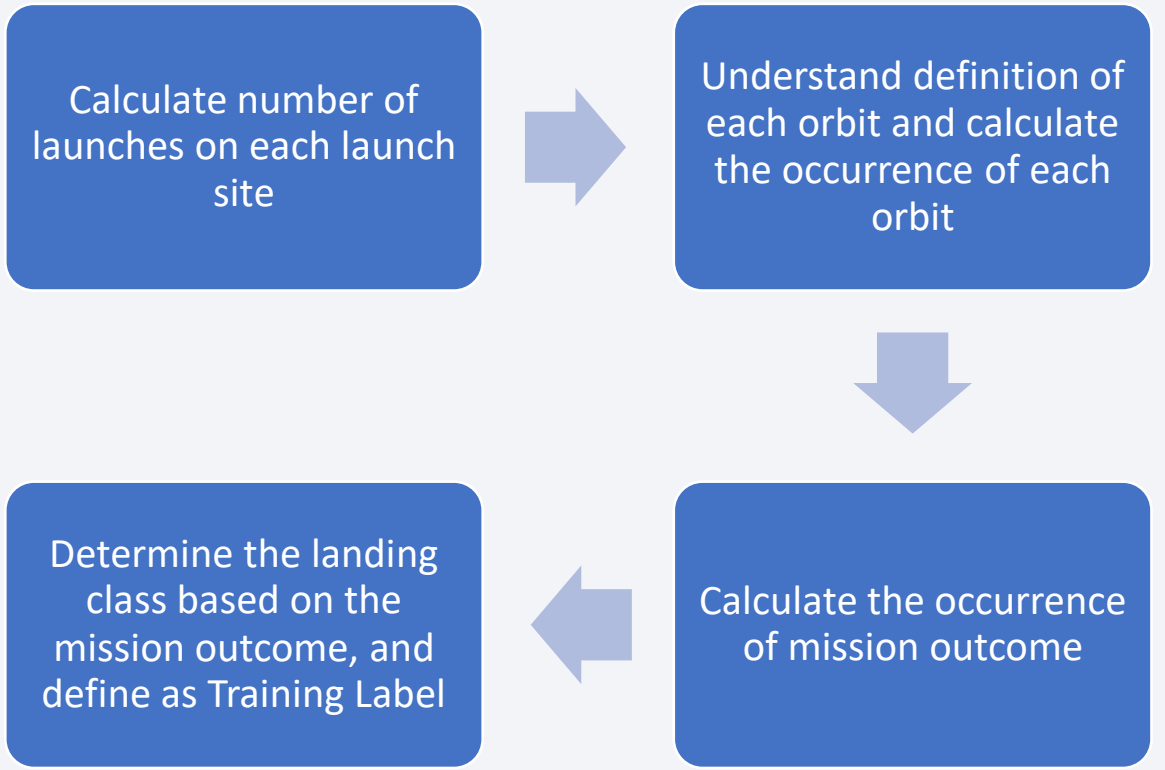
Data Collection - Scraping

- The web scraping process as per flowcharts
- The GitHub URL of the completed web scraping notebook as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Perform Exploratory Data Analysis (EDA) to find patterns in the data and determine what would be the label for training supervised models.
- The data wrangling process as per flowcharts.
- The GitHub URL of completed data wrangling notebook as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- The GitHub URL of completed EDA with data visualization notebook as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb
- Explore the data by visualizing the relationship between flight number vs payload, flight number vs launch site, and payload vs launch site by using seaborn categorical plot, success rate of each orbit type by using bar chart, relationship between flight number vs orbit type, flight number vs orbit type by using scatter plot, and the launch success yearly trend by using line chart.
- Select the important variables as Features Engineering, and use One Hot Encoder technique to create dummy variables to categorial columns.

EDA with SQL

- Perform various SQL queries for Exploratory Data Analysis, including:
  - The unique name of launch site
  - Total payload mass carried by boosters launched by NASA (CRS)
  - Average payload mass carried by booster version F9 v1.1
  - The total number of successful and failure mission outcomes
  - The count of landing outcomes
- The GitHub URL of completed EDA with SQL notebook as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

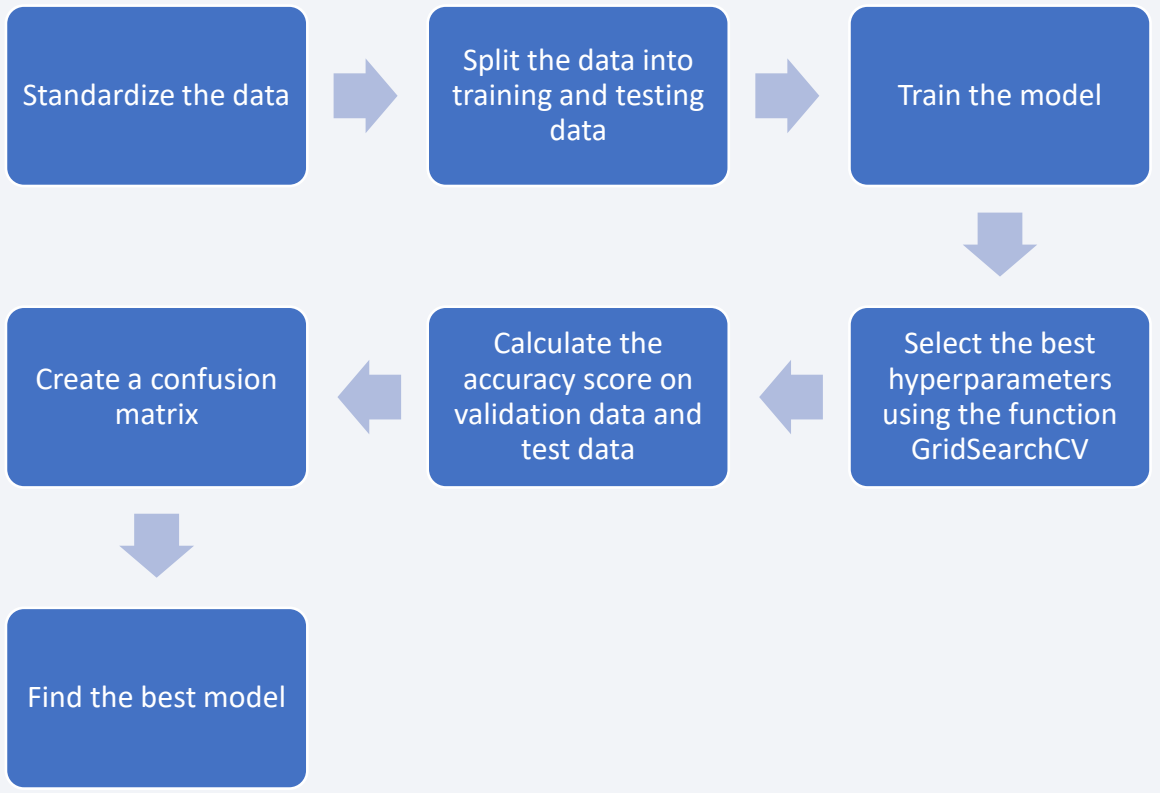
- Launch sites are marked on the Folium map using the latitude and longitude coordinates.
- Mark the success / failed launches for each site on the map.
- Calculate the distances between a launch site to its proximities. Some questions were answered:
  - Are the launch site near to railways, highways and coastlines.
  - Do launch site keep certain distance away from cities.
- The GitHub URL of completed interactive map with Folium map as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- Create the interactive dashboard using Plotly Dash.
- Pie chart is showing the total launch by sites and the success/failed launch by selected site.
- Scatter plot is showing the relationship between the payload mass and the success launch for sites in different booster version.
- The GitHub URL of completed Plotly Dash lab as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/Plotly%20Dash.ipynb

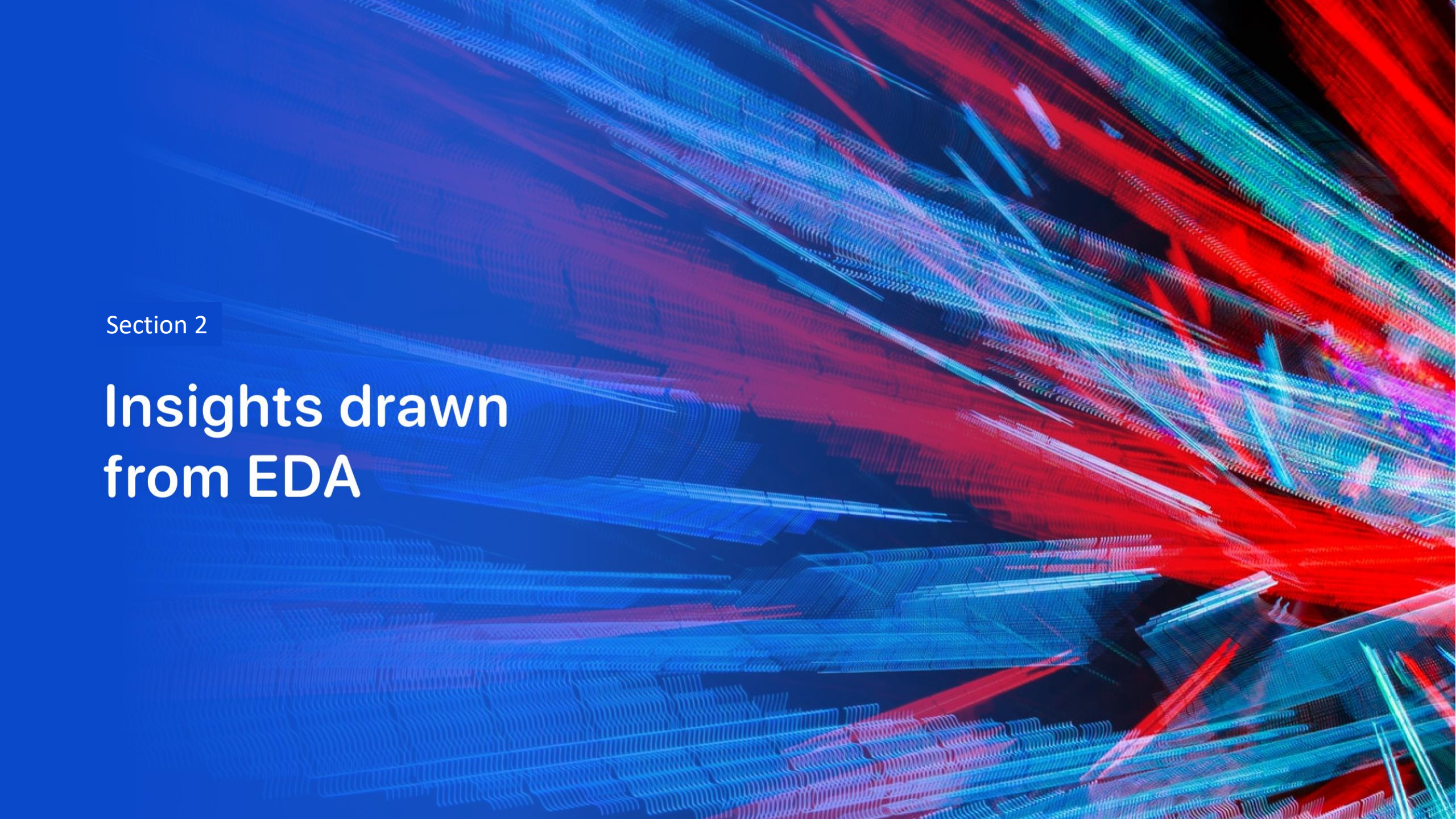
Predictive Analysis (Classification)

- Build the Machine Learning models: Logistic Regression, Support Vector Machine (SVM), Decision Tree, and K Nearest Neighbours (KNN).
- The predictive analysis process as per flowcharts.
- The GitHub URL of completed predictive analysis lab as below: https://github.com/ngkimfong/IBM_DS_Capstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb



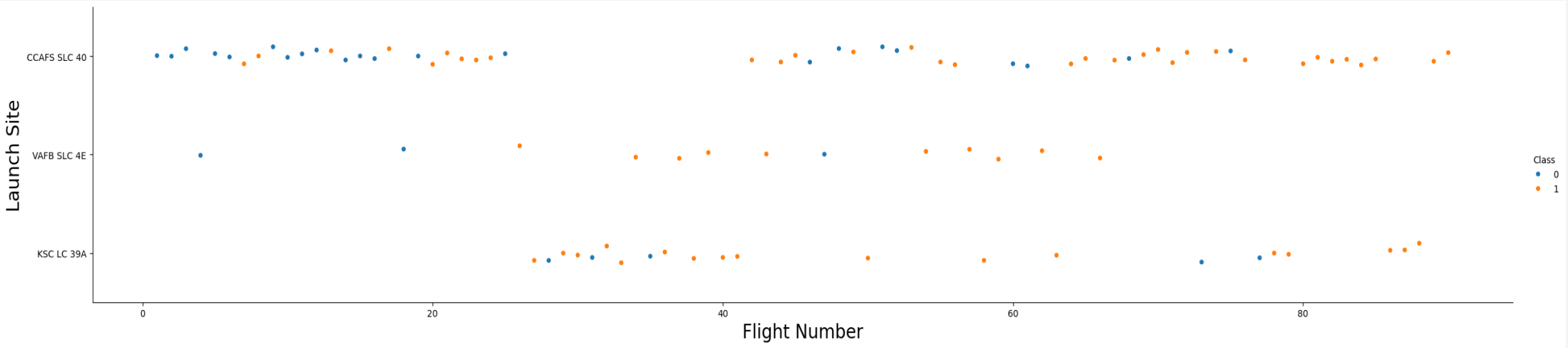
Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

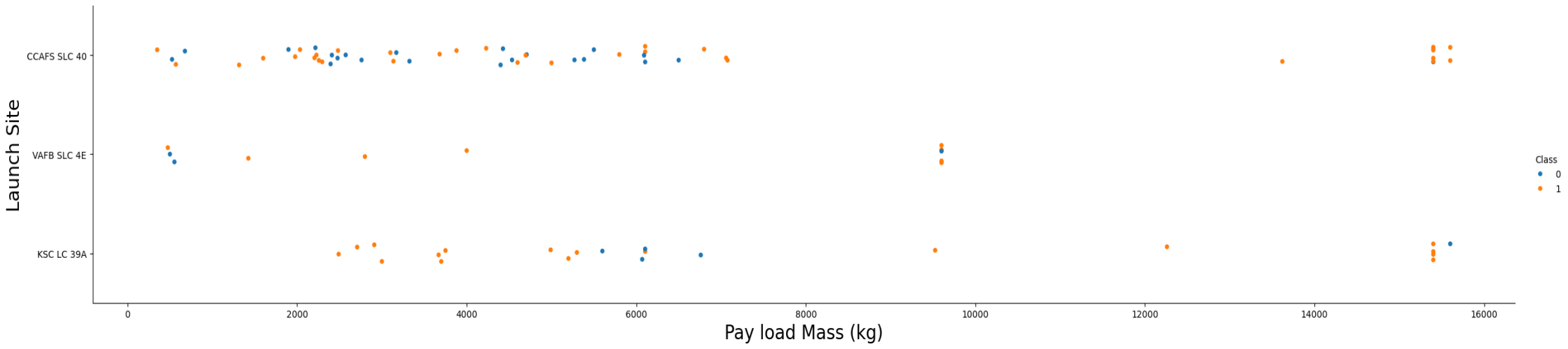
Flight Number vs. Launch Site

- The scatter plot shows that the larger the flight amount at a launch site, the greater the success rate at a launch site.



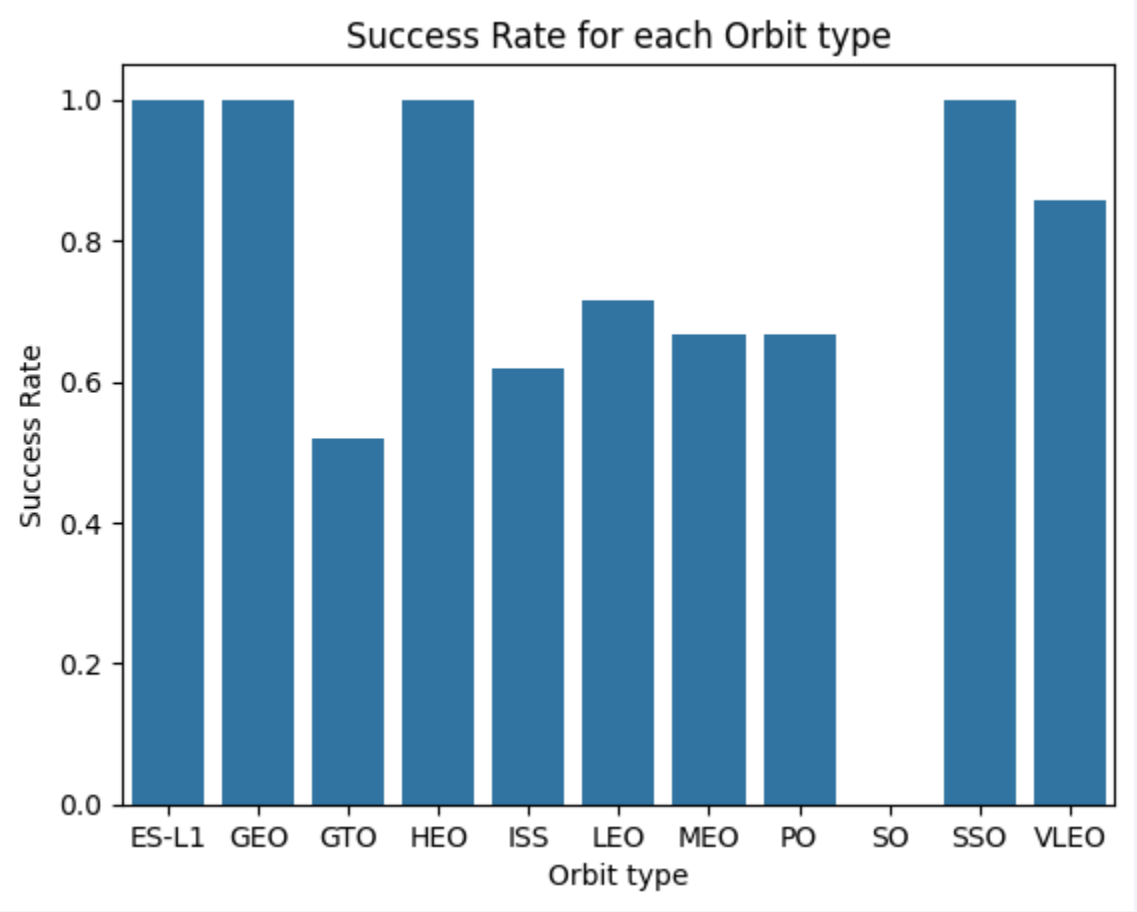
Payload vs. Launch Site

- The scatter plot shows that VAFB-SLC launch site has no rockets launched for heavy payload mass (greater than 10000kg).
- The heavy payload mass has higher success rate for the launch site CCAFS SLC 40.



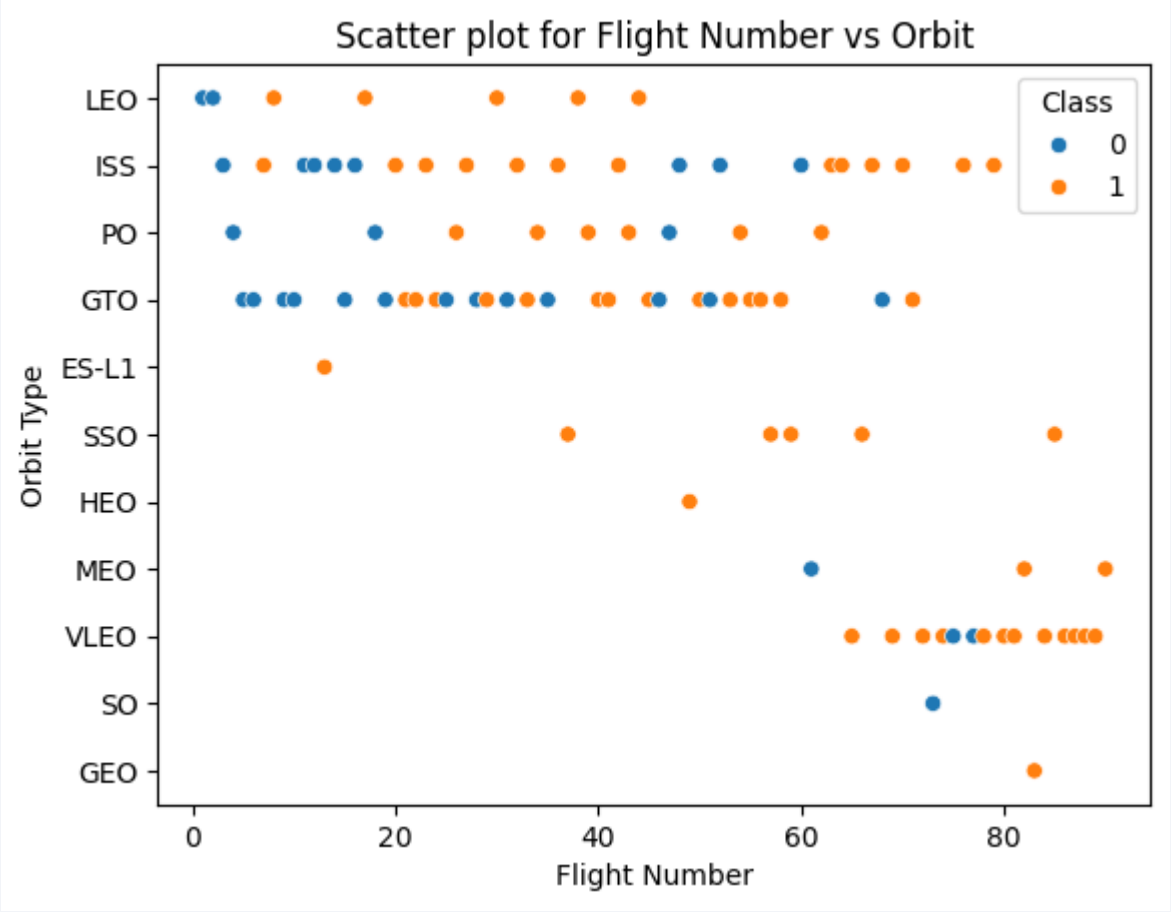
Success Rate vs. Orbit Type

- The bar chart shows that ES-L1, GEO, HEO and SSO orbit type have the highest success rate.



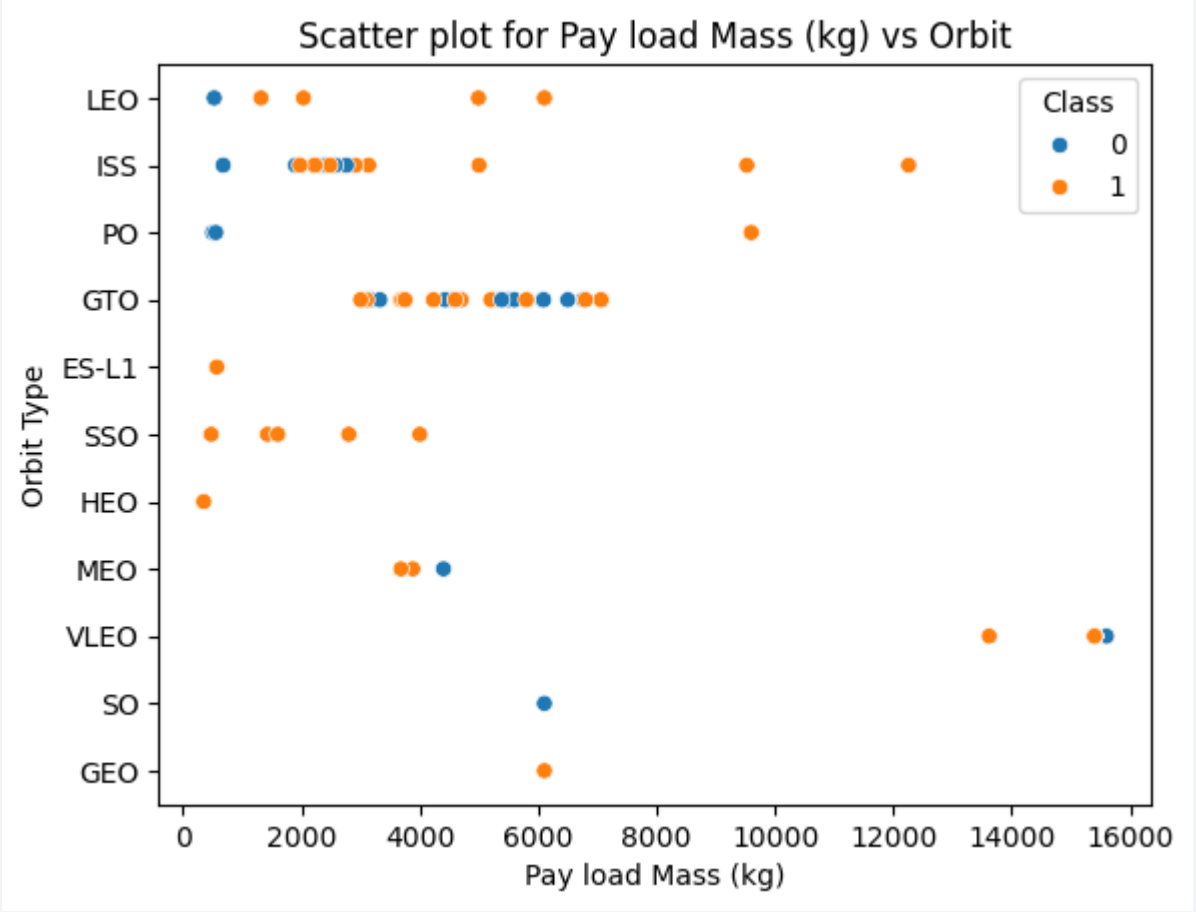
Flight Number vs. Orbit Type

- The scatter plot shows that success is related to the number of flight in LEO orbit.
- However, there seems to be no relationship between flight number and success rate in GTO orbit.



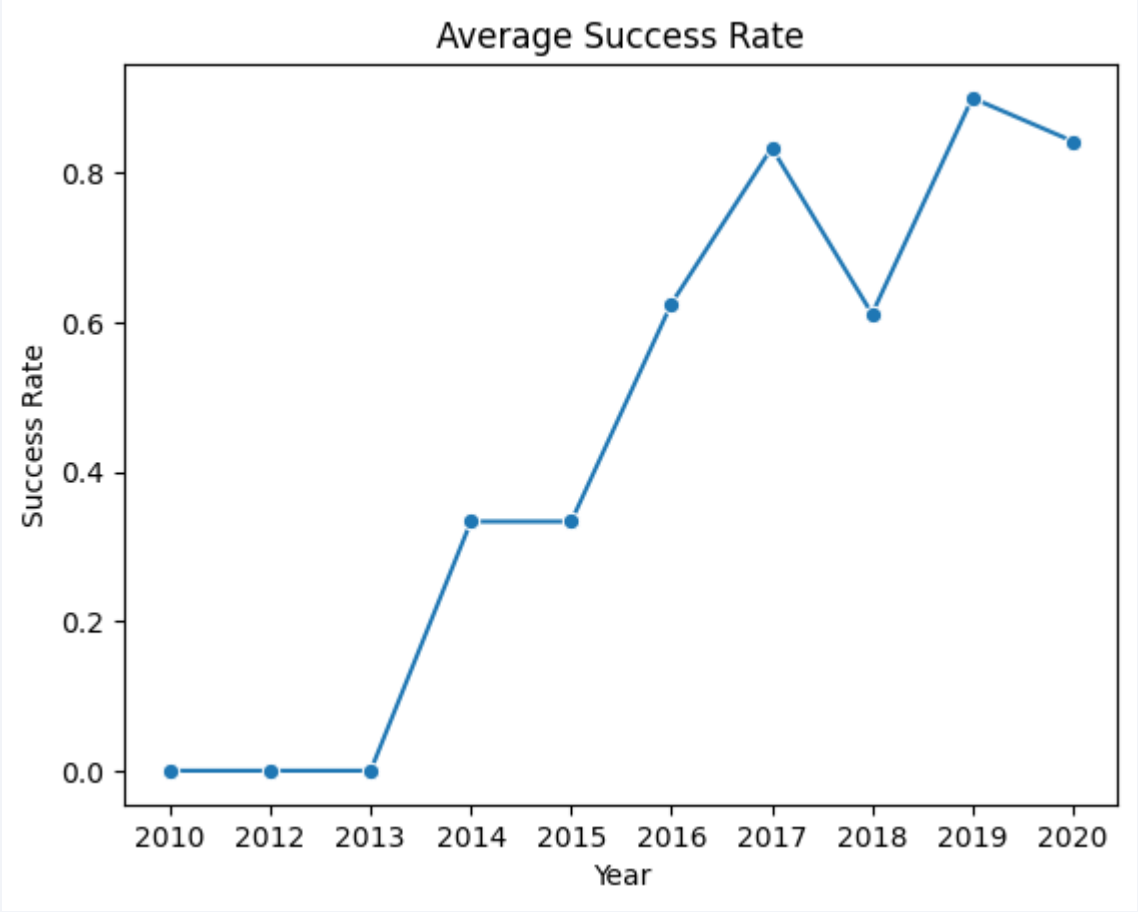
Payload vs. Orbit Type

- The scatter plot shows that with the heavy payload the successful landing are more for LEO, ISS and Polar orbit.
- However for GTO, there seems to be no relationship between payload mass and success rate.



Launch Success Yearly Trend

- The line chart shows that the success rate increase since Year 2013 to 2020.



All Launch Site Names

- Use the keyword DISTINCT to show only unique launch sites from the SpaceX data.

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'

- Use the key word LIKE to show launch sites begin with ‘CCA’.
- Use the keyword LIMIT to show the first 5 records.

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	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

- Use the SUM function to calculate the total payload carried by boosters from NASA.

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

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE Customer LIKE 'NASA (CRS)%';
```

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

SUM("PAYLOAD_MASS__KG_")
48213

Average Payload Mass by F9 v1.1

- Use the AVERAGE function to calculate the 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" LIKE 'F9 v1.1%';
```

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

AVG("PAYLOAD_MASS__KG_")
2534.6666666666665

First Successful Ground Landing Date

- Use the MINIMUM function to find the date of the first successful landing outcome on ground pad.

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

Hint:Use min function

```
%sql SELECT MIN(Date) FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE 'Success%' AND "Landing_Outcome" LIKE '%ground pad%';
```

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

MIN(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Use the combination of keywords LIKE, AND, BETWWEN to list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000.

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" LIKE 'Success%' AND "Landing_Outcome" LIKE '%drone ship%' AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;
```

```
 * 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

- Use the COUNT function to calculate the total number of successful and failure mission outcomes.

List the total number of successful and failure mission outcomes

```
%sql SELECT "Mission_Outcome", COUNT(*) AS COUNT FROM SPACEXTABLE GROUP BY "Mission_Outcome";
```

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

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

Boosters Carried Maximum Payload

- Use the SQL Subquery method to 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

```
%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

- Use the SUBSTRING function to 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.

```
%sql SELECT SUBSTR(Date,6,2) AS MONTH, "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTABLE
WHERE SUBSTR(Date,0,5) = '2015' AND "Landing_Outcome" LIKE 'Failure%' AND "Landing_Outcome" LIKE '%drone ship%'
```

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

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

- Use the keywords GROUP BY and ORDER BY to rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

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

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

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

Section 3

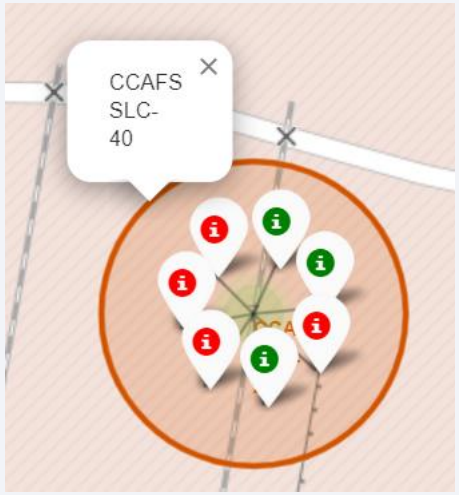
Launch Sites Proximities Analysis

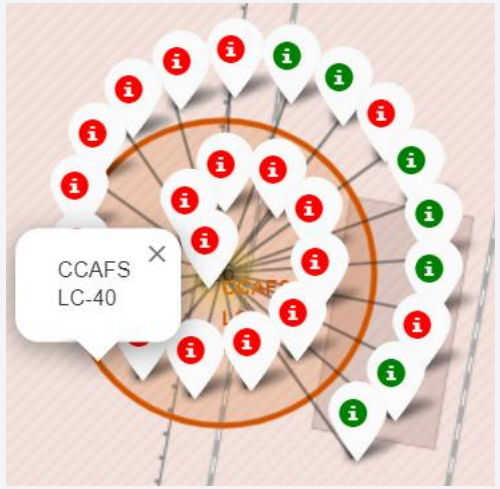
Launch Sites Location Marker

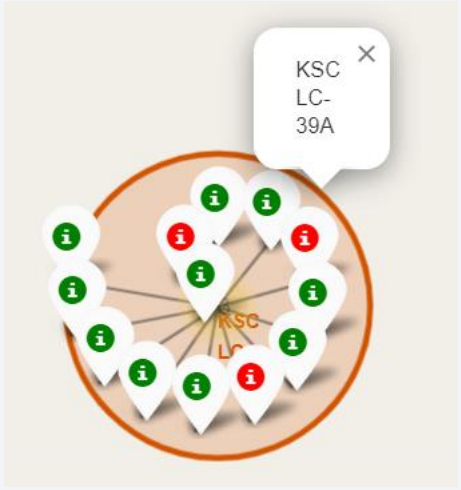
- The screenshot from the Folium map above shows the marker of all the launch sites.
- It shows that all the launch sites are located closed to the coasts in Florida and California, in USA.
- The locations explained most of the successful landing happened on the drone ship.

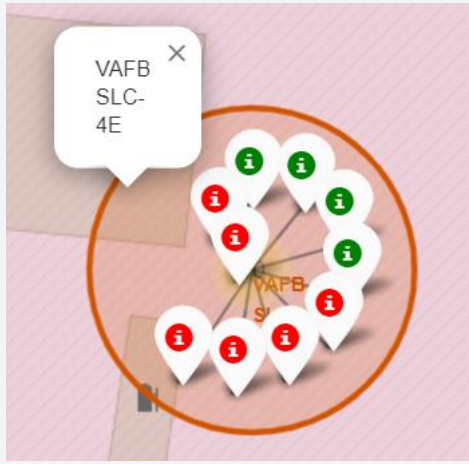
Color-labeled Landing Outcomes

- The screenshots show the color-labeled landing outcomes on the Folium map.
- Green marker shows the successful launches while Red marker shows failures.
- CCAFS LC-40 has the most launch count.
- KSC LC-39A has the highest success rate for the landing outcome.



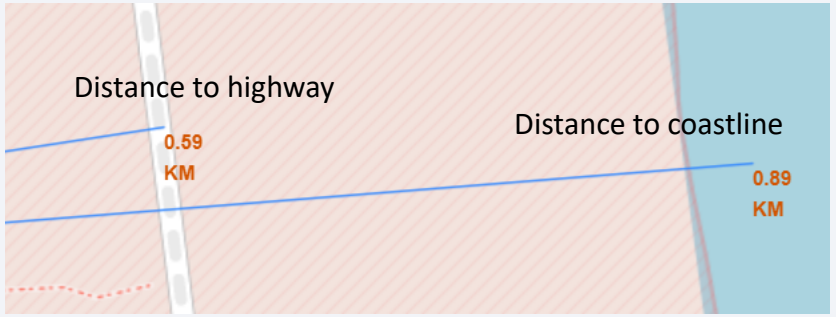


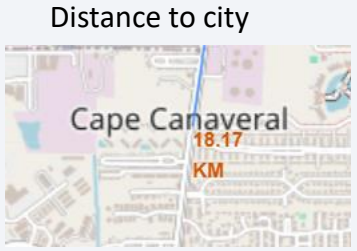


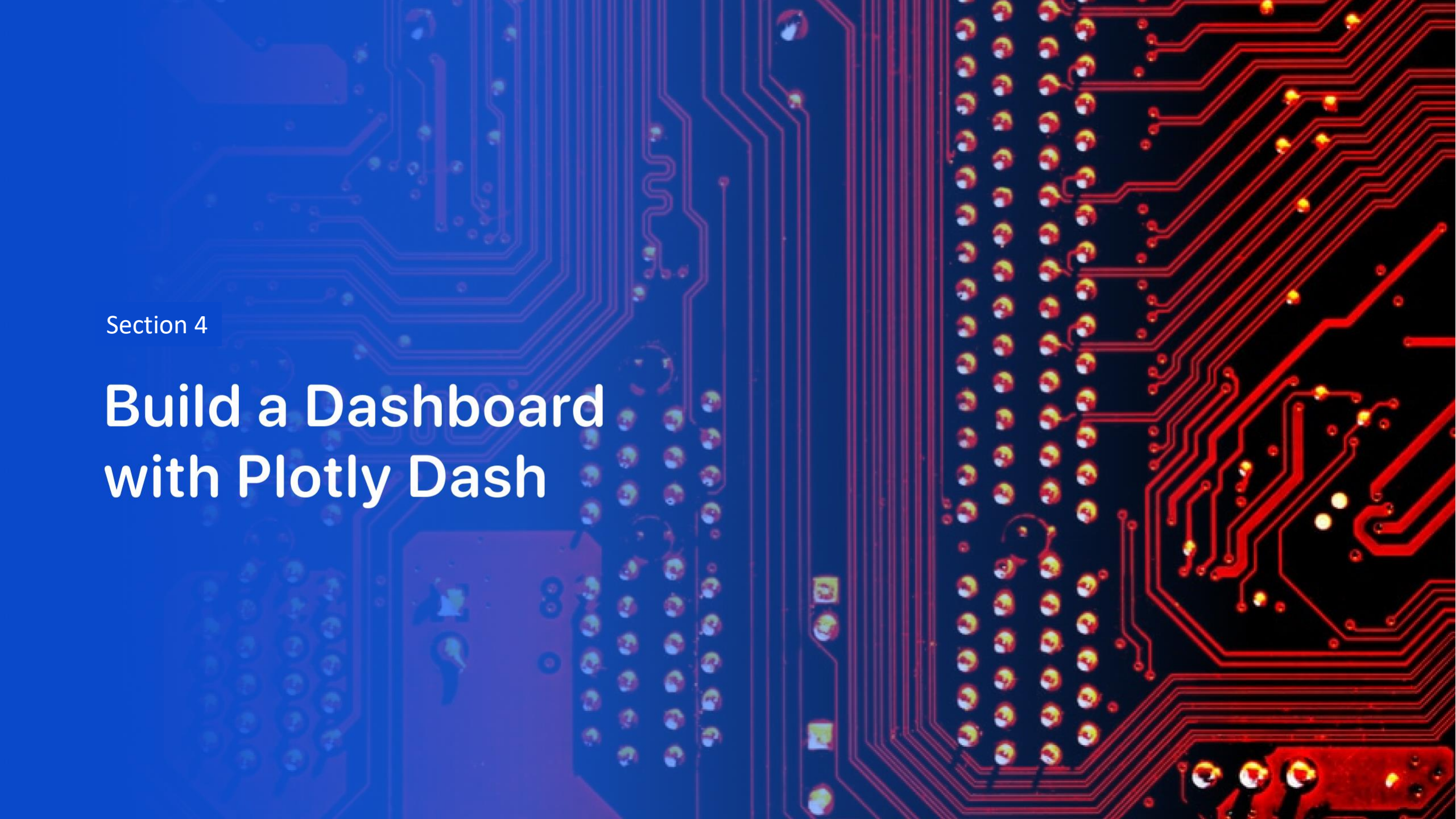


Launch Site distance to its proximities

- The screenshots show the distance of launch site CCAFS SLC-40 to its proximities such as railway, highway and coastline.
- The launch site is less than 2km to the closest highway, railway and coastline.
- The launch site is about 20km to the closest city which keep certain away from the city.

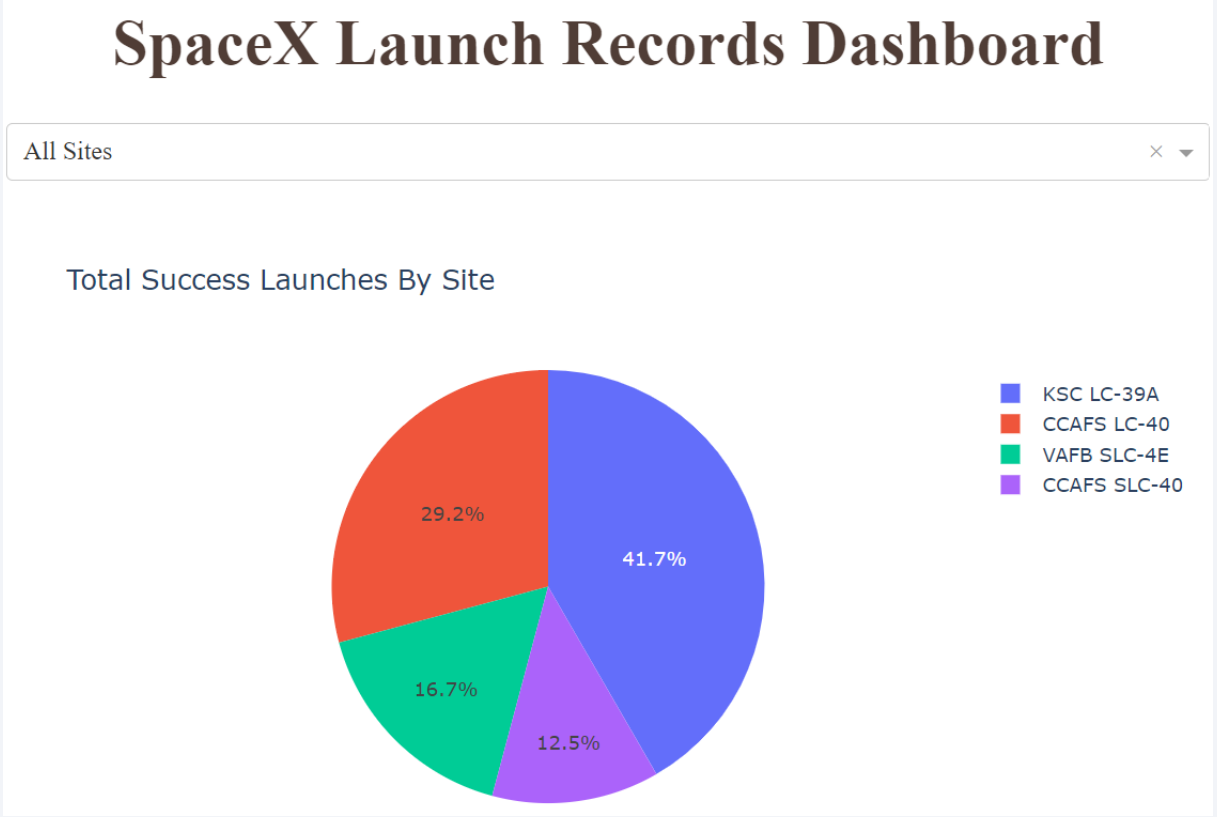





Section 4
Build a Dashboard with Plotly Dash

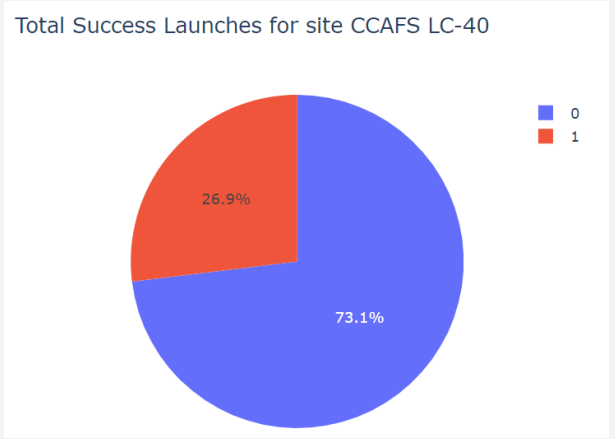
Pie Chart of Success Percentage

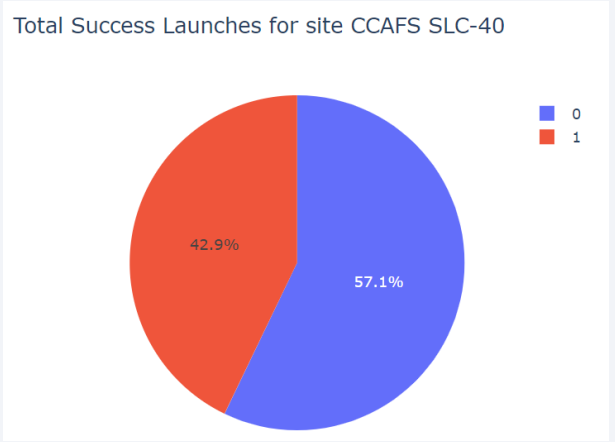
- The screenshot shows the launch success percentage for all sites, in a pie chart.
- KSC LC-39A has the most successful launch.

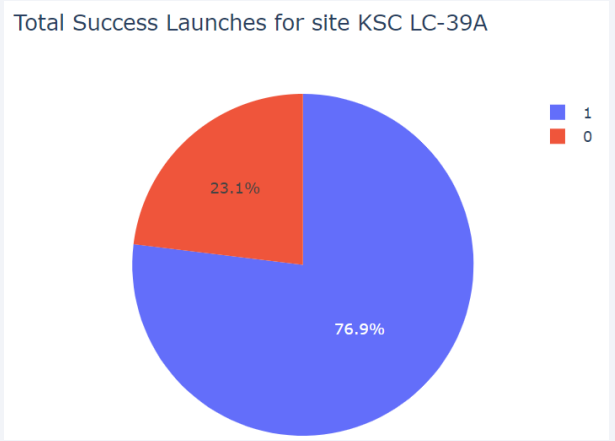


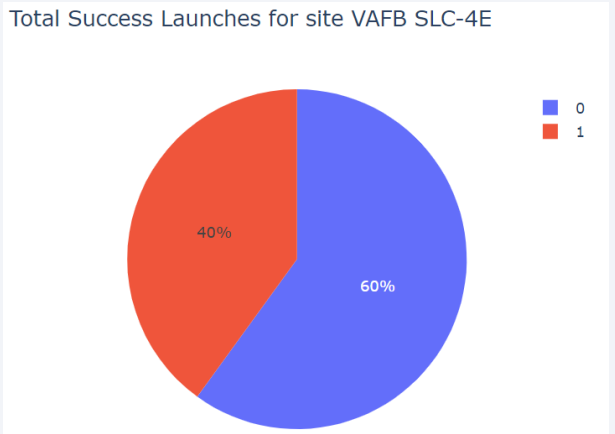
Pie Chart of Success Rate

- Screenshots show the launch success rate of each sites.
- KSC LC-39A has the highest success rate, whereas CCAFS SLC-40 the lowest.



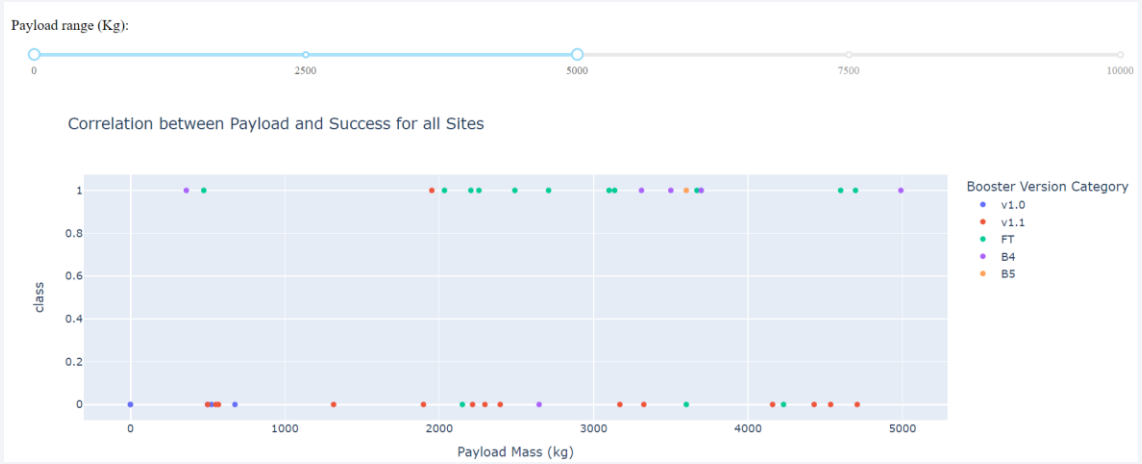


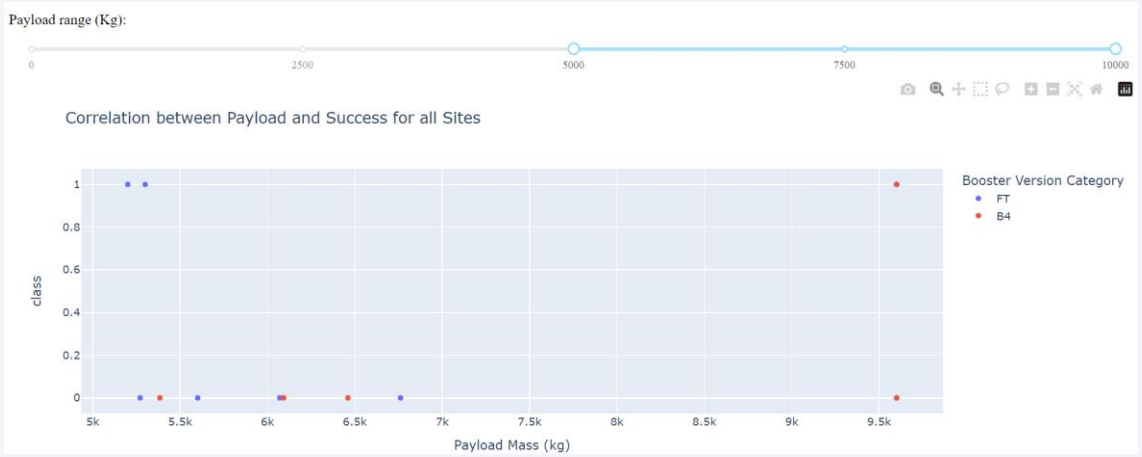




Correlation between Payload and Success Launch

- Screenshots show the scatter plot of Payload vs Launch Outcome for all sites.
- Lighter payload (< 5000kg) has the better success rate than the heavy payload.
- The FT booster has the higher success rate compared to others.



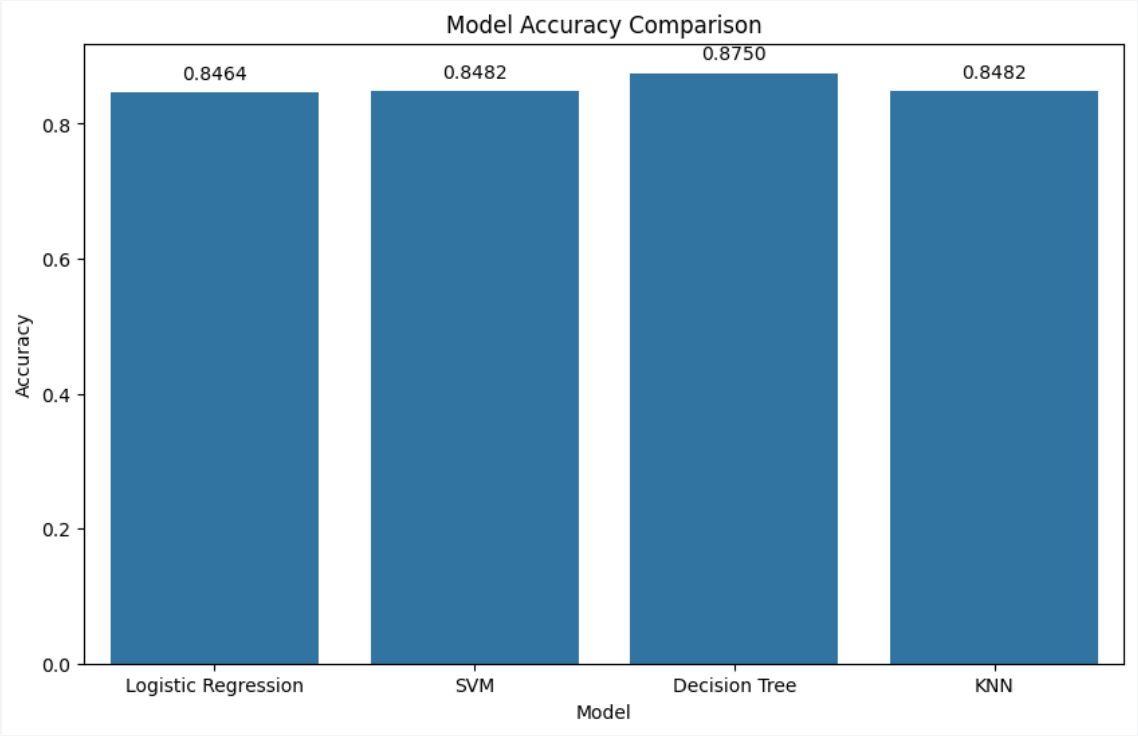


Section 5

Predictive Analysis (Classification)

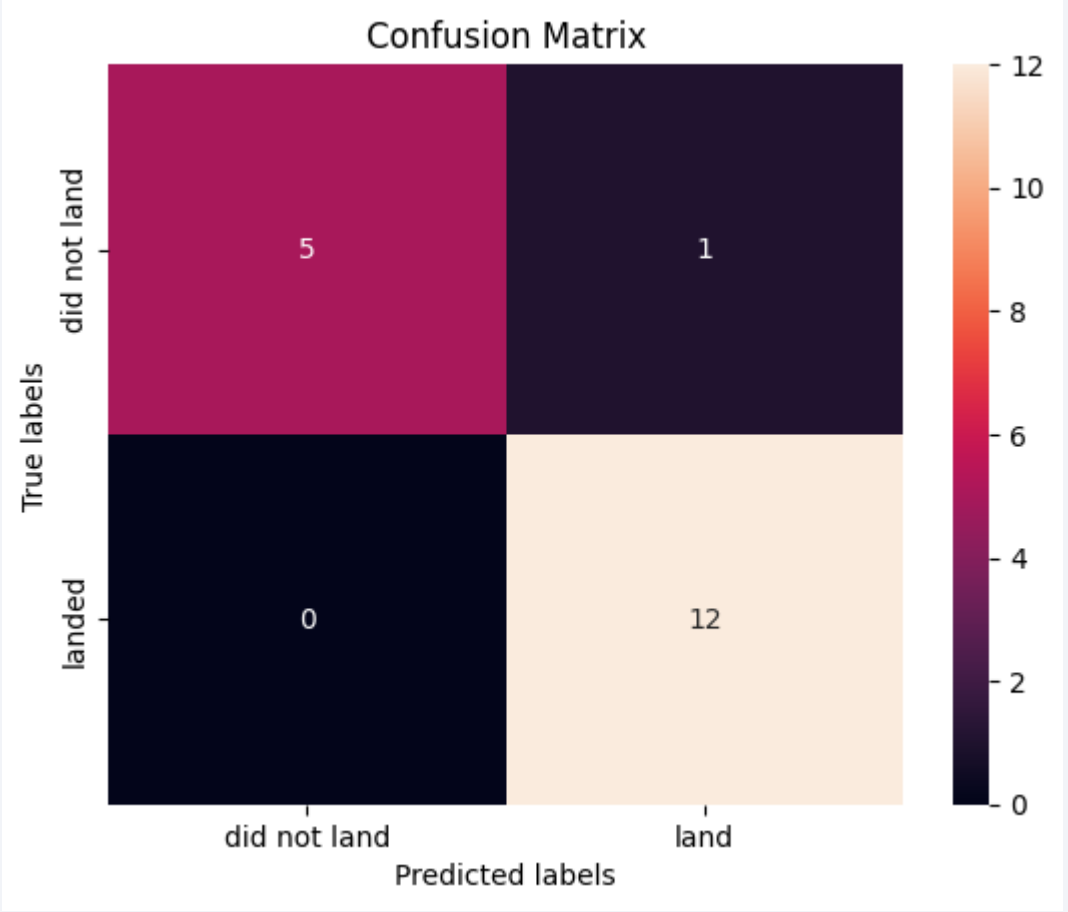
Classification Accuracy

- Decision Tree model has the highest classification accuracy.



Confusion Matrix

- The confusion matrix shows higher number of true positive and true negative and lower number of false positive and false negative.
- The confusion matrix for the Decision Tree model reveals that the classification effectively distinguishes between various classes.



Conclusions

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- The success rate increase since Year 2013 to 2020.
- KSC LC-39A has the highest success rate, whereas CCAFS SLC-40 the lowest.
- Lighter payload (< 5000kg) has the better success rate than the heavy payload.
- Decision Tree is the best classification model algorithm for this capstone project.

Thank you!