IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Deron Nicholson
November 5,2022

Outline

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

Executive Summary

Summary of methodologies

- Data Collection through API
- Data Collection with Web Scraping
- Data Wrangling
- Exploratory Data Analysis with SQL
- Exploratory Data Analysis with Data Visualization
- Interactive Visual Analytics with Folium
- Machine Learning Prediction

Summary of all results

- Exploratory Data Analysis result
- Interactive analytics in screenshots
- Predictive Analytics result

Introduction

- SpaceX is an American spacecraft manufacturer, launcher, and a satellite communications corporation offering a rocket launches specifically Falcon 9 as low as 62 million dollars; while other providers cost upward of 165 million dollar each. much of the savings is because Space X can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. Repeating this process will make the price down even further. As a data scientist of a startup rivaling SpaceX, the goal of this project is to create the machine learning pipeline to predict the landing outcome of the first stage in the future. The problems included:
    - ➢ Identifying all factors that influence the landing outcome.
    - ➢ The relationship between each variables and how it is affecting the outcome.
    - ➢ The best condition needed to increase the likelihood of successful landing

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scraping from Wikipedia.
- Perform data wrangling
  - The data was converted from csv field to dataframe and a column called class was created with values determined by whether or not there was a bad landing outcome
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

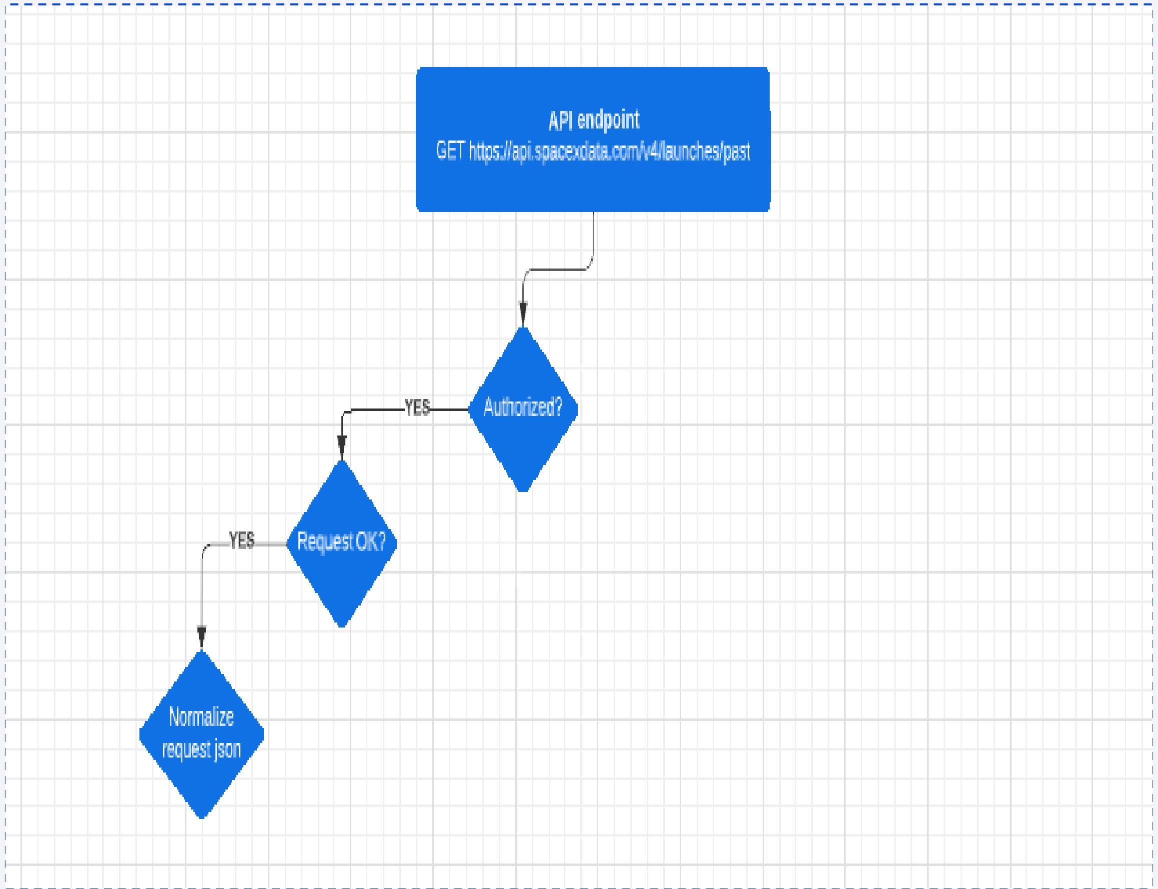
Data Collection

- Data was collected using get request to the SpaceX API.
- Then the response content was converted to json then to a dataframe where NA values were filled using domain knowledge
- Additionally, Wikipedia was scraped for Falcon9 launch records with BeautifulSoup and then subsequently converted to dataframe.

Data Collection – SpaceX API

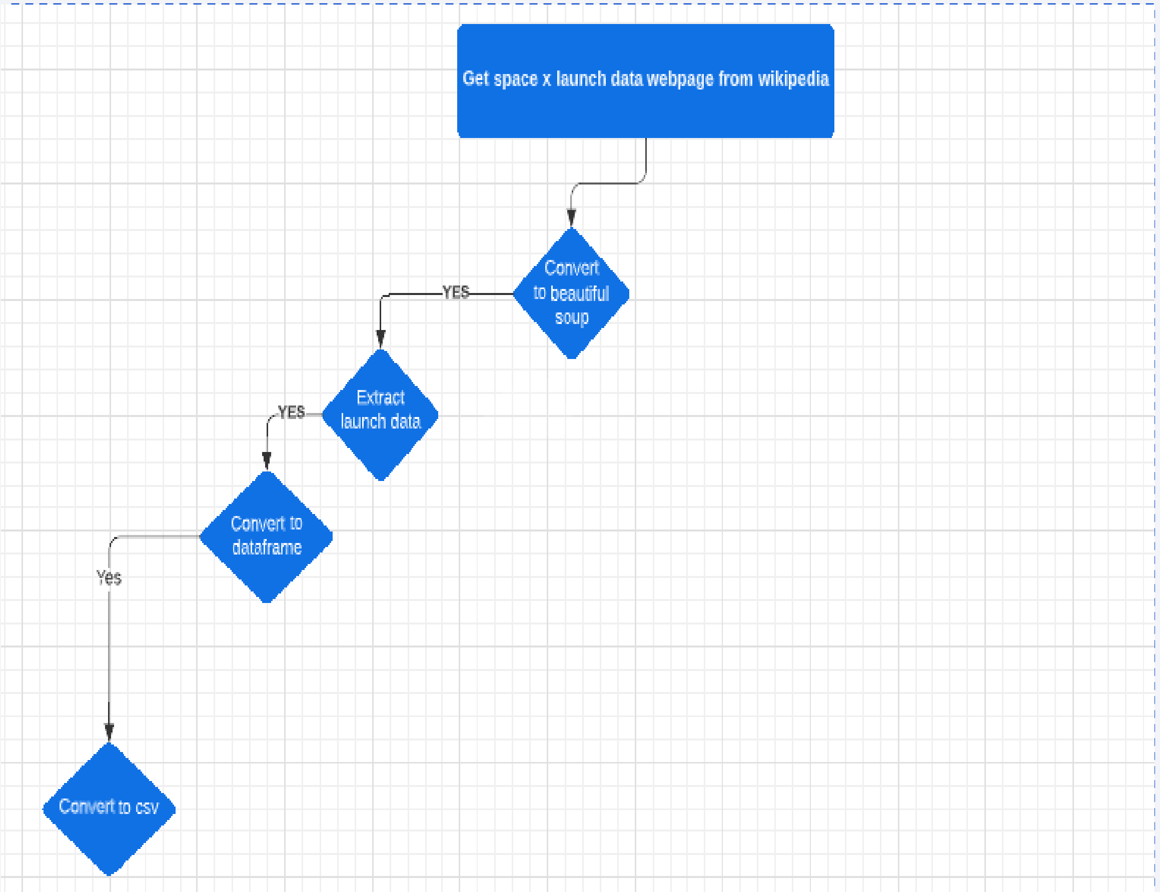
- A get request was made to space x api to retrieve rocket launch data. The data was converted to json and normalized and the data stored in a dataframe. Missing values filled using domain knowledge and then the dataframe was saved to a csv file

- GitHub URL :[data-collection-api.ipynb]



Data Collection - Scraping

- A beautiful soup object was created from the webpage. The table with launch data was extracted and the contents stored in dataframe which was subsequently cleaned and converted to csv.
- GitHub URL of the completed web scraping notebook: https://github.com/Deronic246/SpaceX_Final_Project/blob/main/webscraping.ipynb



Data Wrangling

Exploratory data analysis was performed, and the training labels were determined. Number of launches at each site, and the number and occurrence of each orbits was calculated. A landing outcome label was created from the outcome column and the results was exported to csv.

- GitHub URL of the completed data wrangling notebook: https://github.com/Deronic246/SpaceX_Final_Project/blob/main/spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- The data wad explored by visualizing the relationship between flight number and launch Site, payload and launch site, success rate of each orbit type, flight number and orbit type, the launch success yearly trend.
- GitHub URL of completed EDA with data visualization notebook: https://github.com/Deronic246/SpaceX_Final_Project/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

The following queries were performed during EDA:

- The names of unique launch sites in the space mission.
- The total payload mass carried by boosters launched by NASA (CRS)
- The average payload mass carried by booster version F9 v1.1
- The total number of successful and failure mission outcomes
- The failed landing outcomes in drone ship, their booster version and launch site names.
- GitHub URL of completed EDA with SQL notebook: https://github.com/Deronic246/SpaceX_Final_Project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- A circle, marker and polyline objects were used on the folium map. The circle was used to circle the launch sites while the marker was used to denote launch sites by their successes and failures.
- A polyline was used to show the proximity between a launch site and a selected coastline point.
- GitHub URL of completed interactive map with Folium map: https://github.com/Deronic246/SpaceX_Final_Project/blob/main/lab_jupyter_launch_site_location%20(2).ipynb

Build a Dashboard with Plotly Dash

- An interactive dashboard was created using plotly and consisted of pie charts and scatter graphs.
- Those plots and interactions were added to tell a story about the total launches by certain sites and the relationship with Outcome and Payload Mass (Kg) for the different booster versions.
- GitHub URL of completed Plotly Dash lab:
- https://github.com/Deronic246/SpaceX_Final_Project/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

The process was as follows:

- A create a column for the class
- The data was standardized and then split into training data and test data.
- Various classification algorithms were tried with the best hyperparamters and the best model was selected by its accuracy

- GitHub URL of your completed predictive analysis lab: https://github.com/Deronic246/SpaceX_Final_Project/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

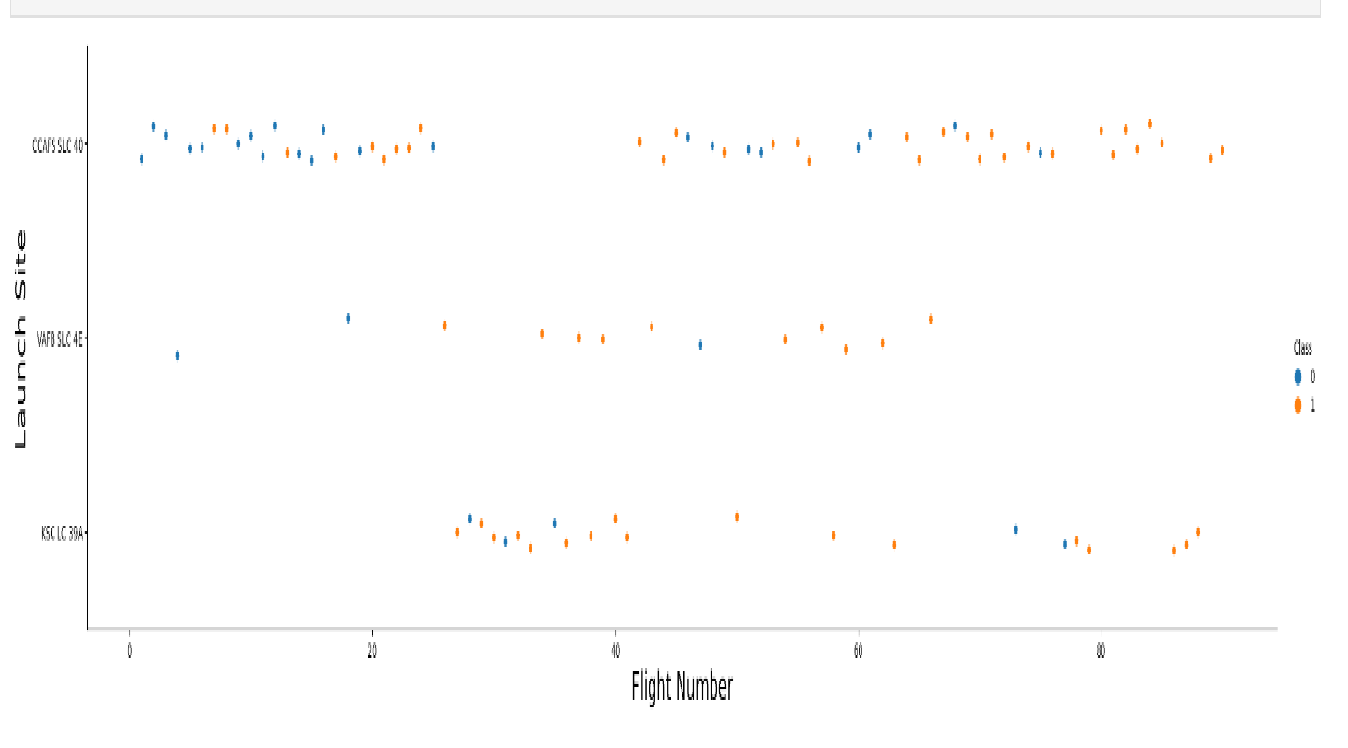
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

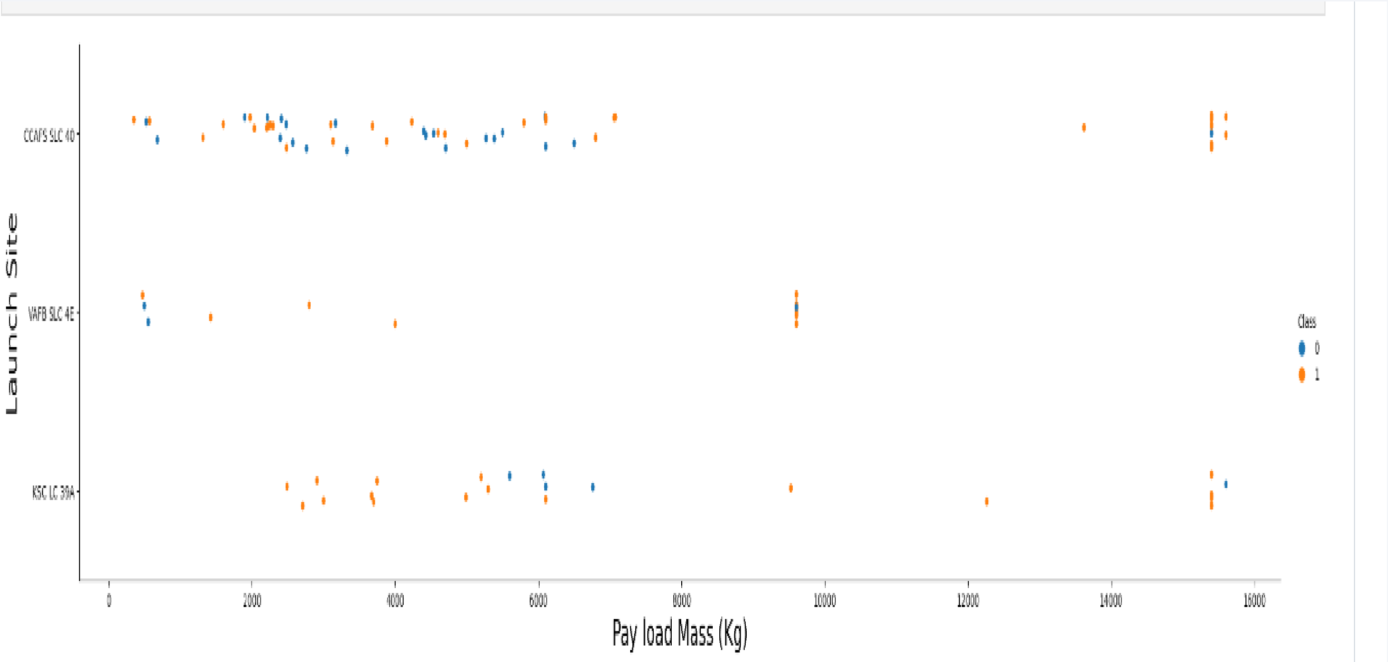
Flight Number vs. Launch Site

- The scatter plot shows that the closer the launches are , the more failures there are.



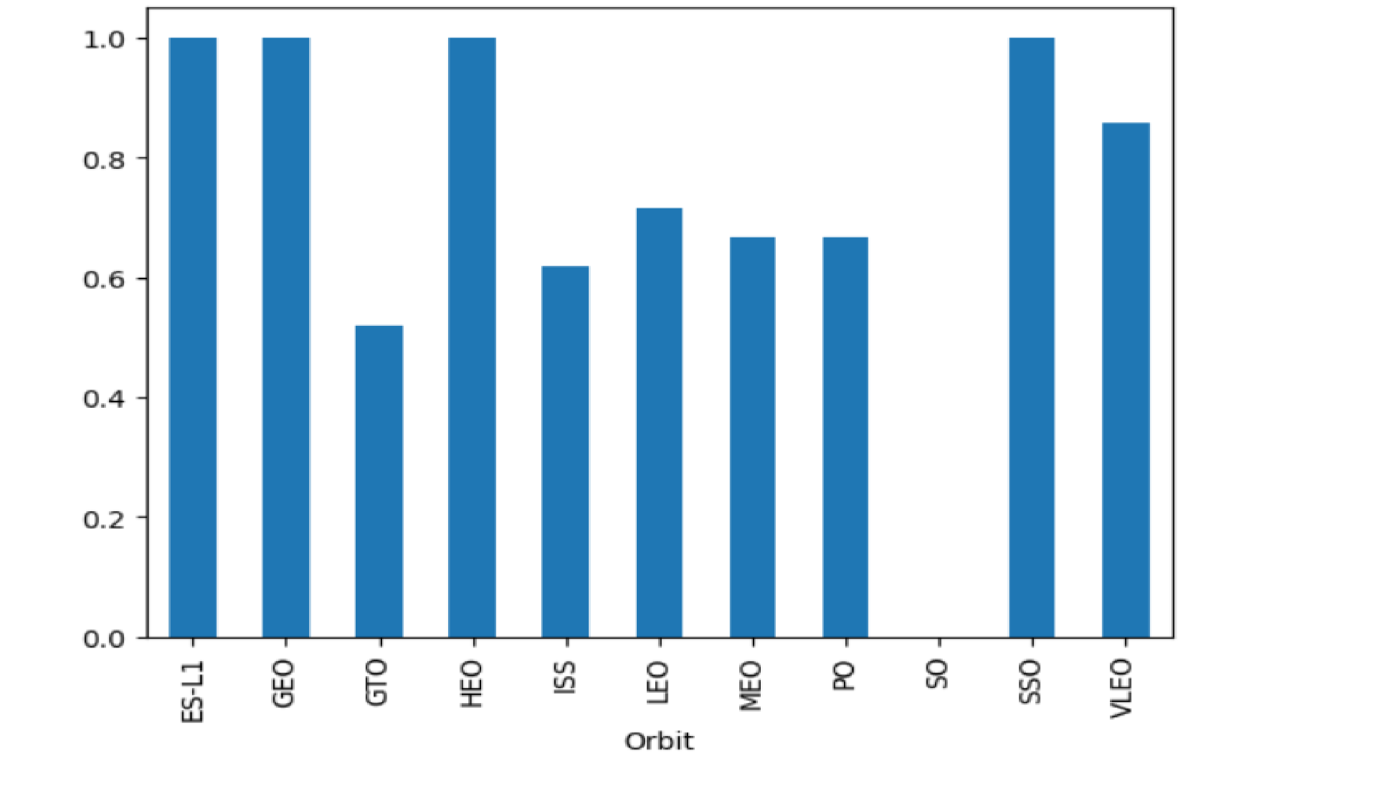
Payload vs. Launch Site

- The scatter plot shows that KSC LC 39A is more favorable to heavy loads.



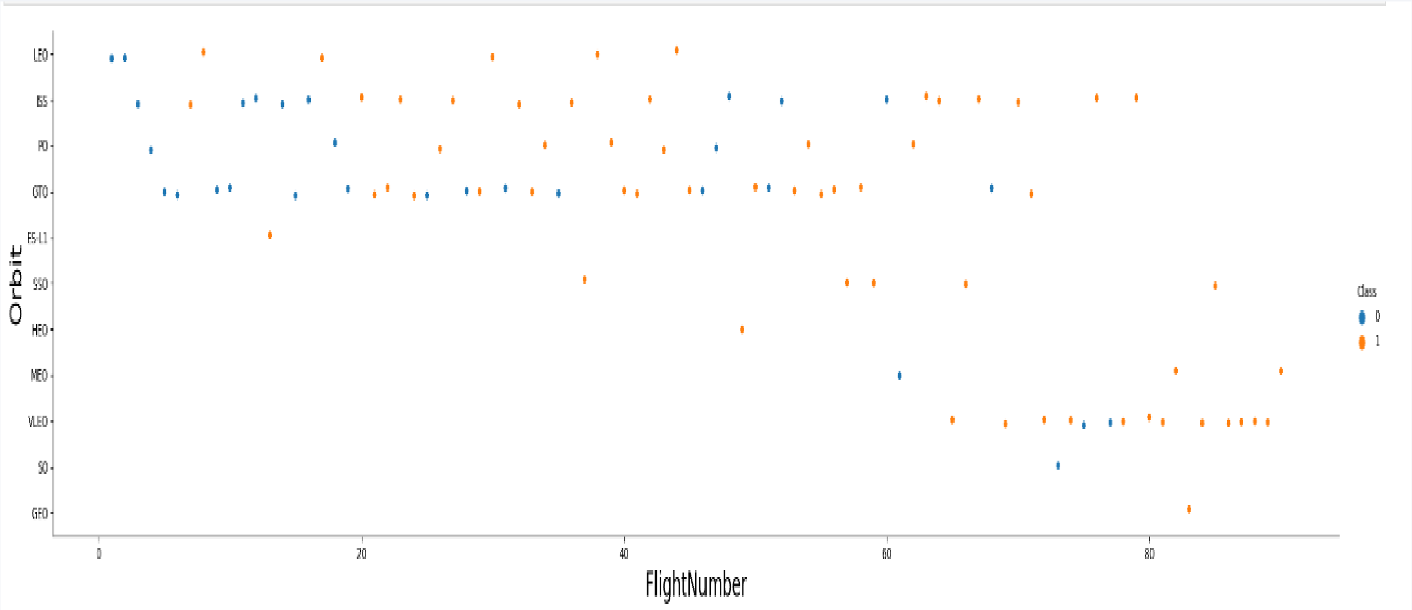
Success Rate vs. Orbit Type

- The bar chart depicts the fact that certain orbits improves the success rate of the landing outcome with ES-L1, GEO,HEO and SSO having a 100% success rate.



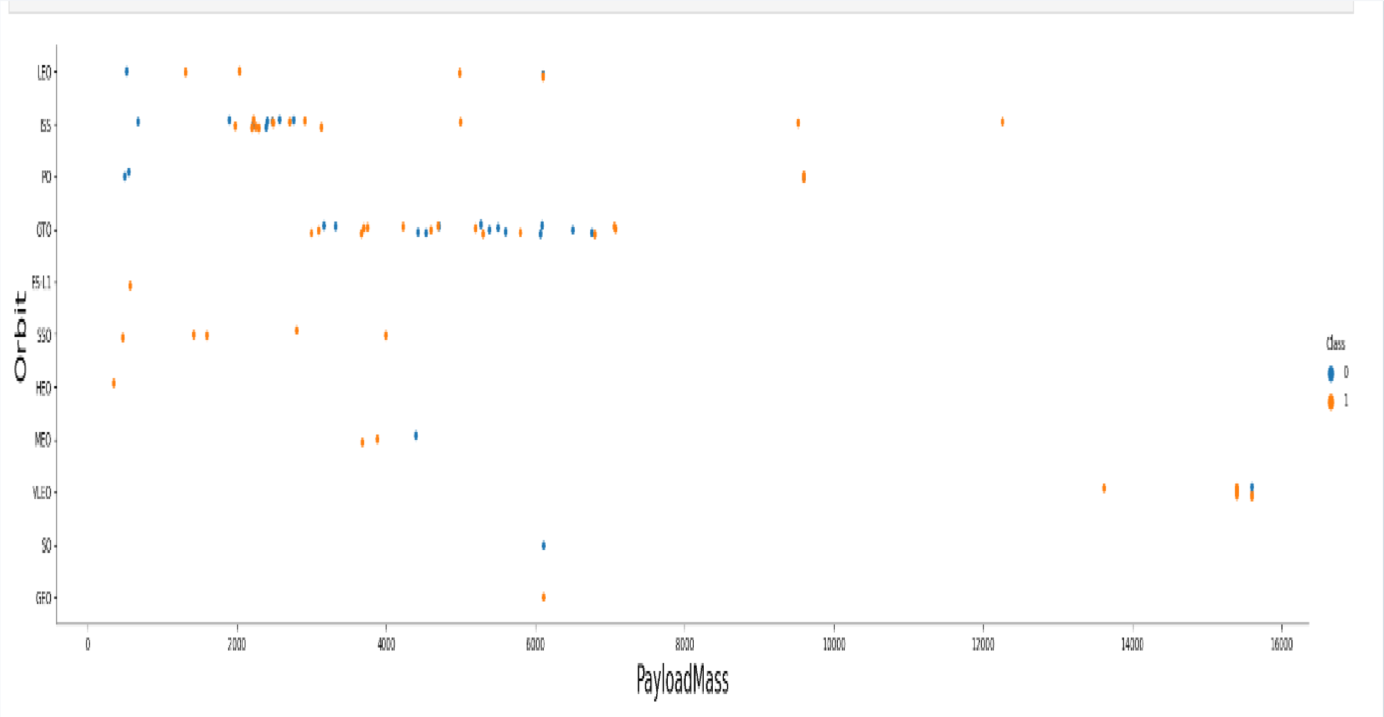
Flight Number vs. Orbit Type

We see that the success of the flight increases at each orbit increases with the Flight Number.



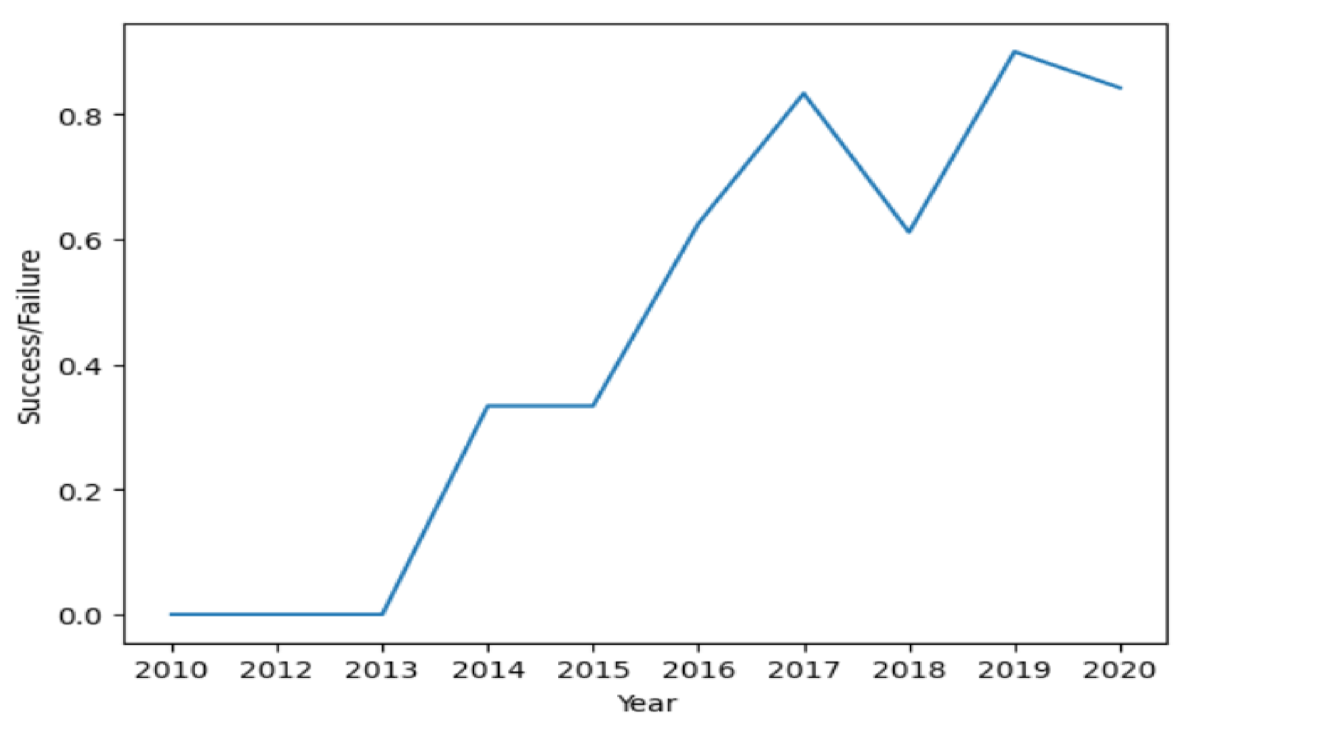
Payload vs. Orbit Type

- The Scatter plot shows that 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

- The line chart shows that there was a plateau with success rate from 2010 to 2013, then a gradual increase in the success rate onwards until 2020.



All Launch Site Names

- A simple select query was used to return the launch site and the distinct keyword was used to prevent duplicates

Task 1

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

```
%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

Launch Site Names Begin with 'CCA'

- A select query was executed and five records starting with ‘CCA’ were returned

Task 2

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

```
%sql select * from SPACEXTBL 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
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- A sum of all the payload mass results was taken where customer was equal to ‘NASA (CRS)’ and the result was aliased as total payload mass.

```
%sql select sum(payload_mass__kg_) as " total payload mass" from SPACEXTBL where customer='NASA (CRS)'
```

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

total payload mass
45596

Average Payload Mass by F9 v1.1

- An average of all the payload mass results was taken where booster version started with ‘F9 v1.1’ and the result was aliased as average payload mass.

Task 4

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

```
%sql select avg(payload_mass__kg_) as "average payload mass" from SPACEXTBL where booster_version like 'F9 v1.1%'
```

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

average payload mass
2534.6666666666665

First Successful Ground Landing Date

- The date of the first successful landing outcome on ground pad was found by filtering landing outcome by ‘Success (ground pad)’ and then the minimum date was found. The result was aliased as the first successful landing outcome in ground pad.

Task 5

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

Hint:Use min function

```
%sql select min(Date)as "first successful landing outcome in ground pad" from SPACEXTBL where ("Landing _Outcome")= 'Success (ground pad)'
```

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

first successful landing outcome in ground pad
01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- A select query was executed where results were filtered by landing outcome where “Success (drone ship)” was present and by payload mass between 4000 and 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

```
%sql select booster_version, payload_mass__kg_,("Landing _Outcome") from SPACEXTBL where ("Landing _Outcome") = 'Success (drone ship)' and payload_ma
```

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

Booster_Version	PAYLOAD_MASS__KG_	Landing _Outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

Total Number of Successful and Failure Mission Outcomes

- A select query was executed where records were aggregated by mission outcome and then counted.

Task 7

List the total number of successful and failure mission outcomes

```
%sql select count(*)as "total number" , mission_outcome from SPACEXTBL group by mission_outcome
```

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

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

Boosters Carried Maximum Payload

- A select query which relied on sub query was executed to get the names of the booster_versions 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 booster_version, max(payload_mass__kg_) as"maximum payload mass" from (select booster_version , payload_mass__kg_ from SPACEXTBL) group b
```

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

booster_version	maximum payload mass
F9 B4 B1039.2	2647
F9 B4 B1040.2	5384

2015 Launch Records

- The approach was incorrect, and an error occurred.

```
%sql SELECT MONTH(DATE),MISSION_OUTCOME,BOOSTER_VERSION,LAUNCH_SITE FROM SPACEXTBL WHERE EXTRACT(YEAR FROM DA
```

```
 * sqlite:///my_data1.db
(sqlite3.OperationalError) near "FROM": syntax error
[SQL: SELECT MONTH(DATE),MISSION_OUTCOME,BOOSTER_VERSION,LAUNCH_SITE FROM SPACEXTBL WHERE EXTRACT(YEAR FROM DA
TE)='2015']
(Background on this error at: http://sqlalche.me/e/e3q8)
```

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

- A select query was executed 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.

Task 10

Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

```
%sql SELECT "Landing _Outcome" FROM SPACEXTBL WHERE DATE BETWEEN '04-06-2010' AND '20-03-2017' ORDER BY DATE
```

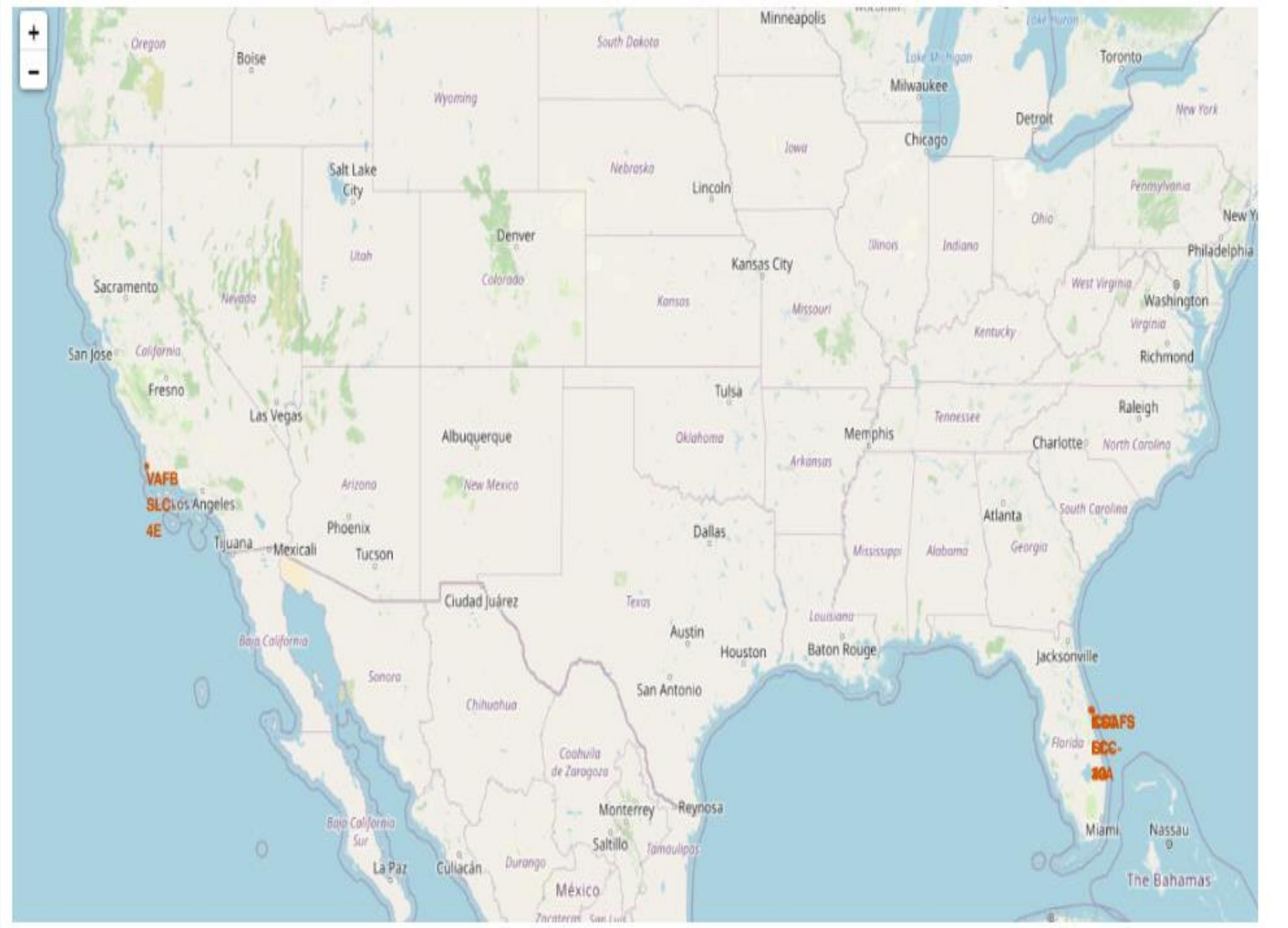
```
 * sqlite:///my_data1.db
Done.
```

Section 3

Launch Sites Proximities Analysis

Folium Map Showing Launch Sites

- Upon analyzing the folium map, I noticed that the space x launch sites appear to be along the coastline within the United States of America and were denoted with a circle marker.

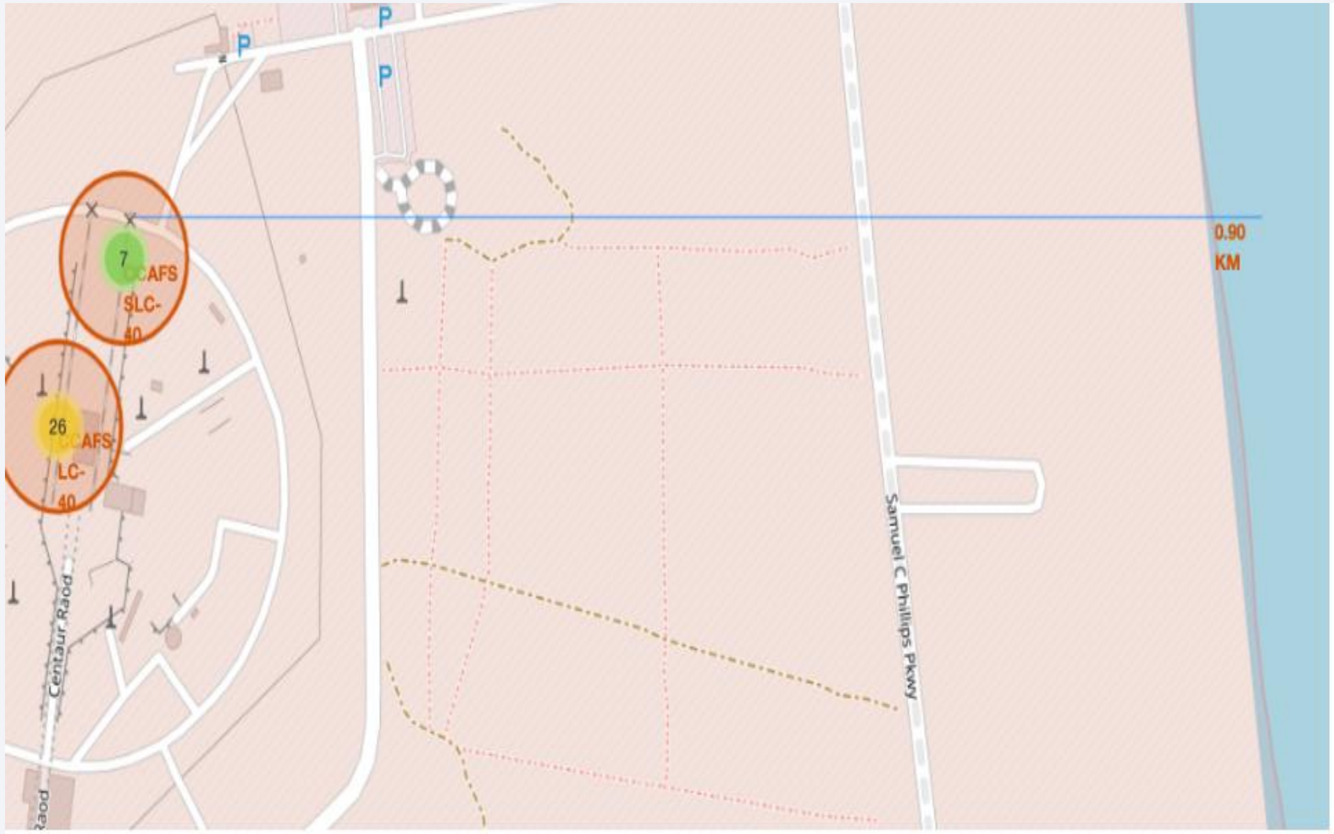


Folium Map Showing success/failed launches for Site

- The green marker shows successful launches while the red marker shows failed launches for the site.

Folium Map showing distance from coastline to site

- The folium map depicts the distance from a launch site to the coastline.

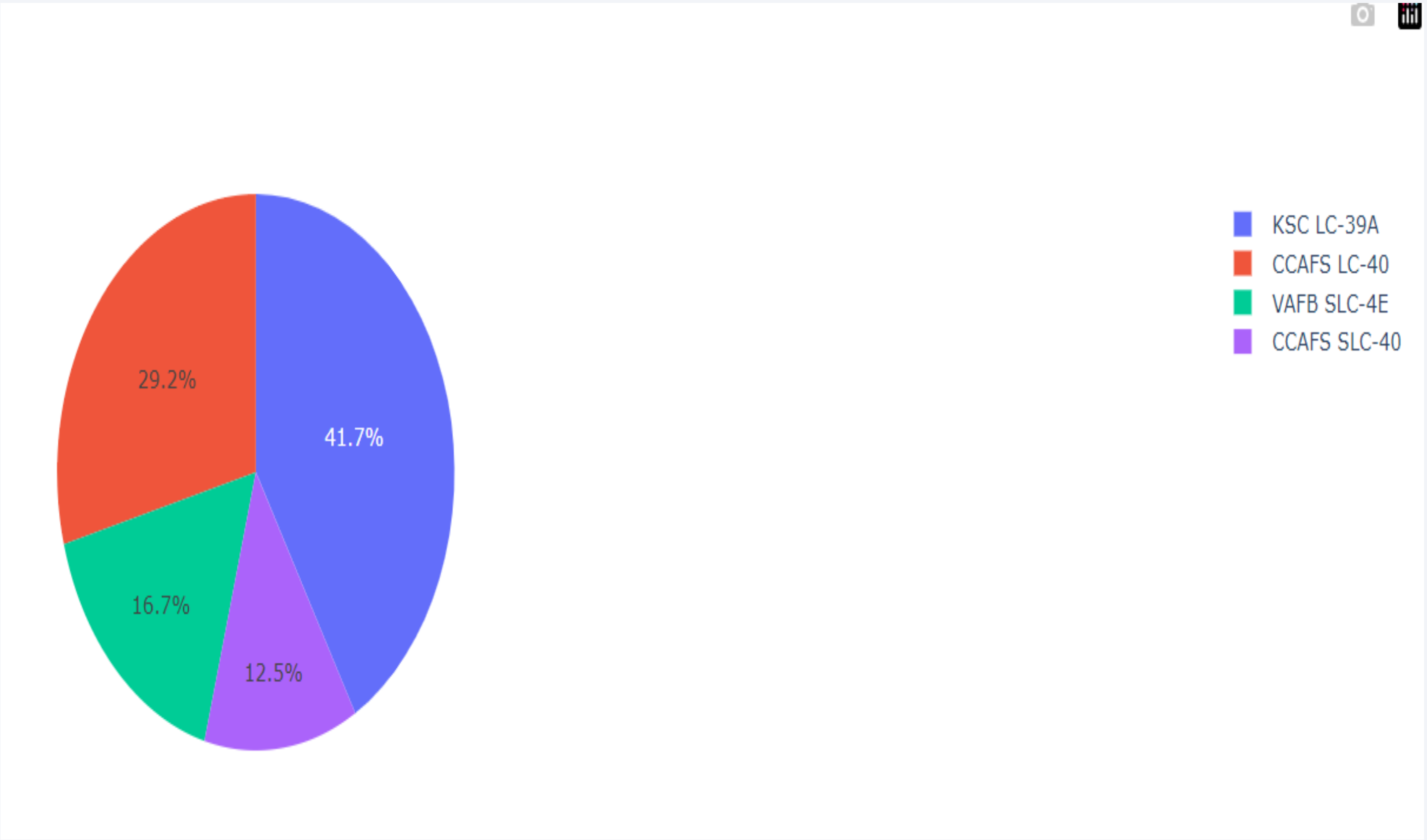


Section 4

Build a Dashboard with Plotly Dash

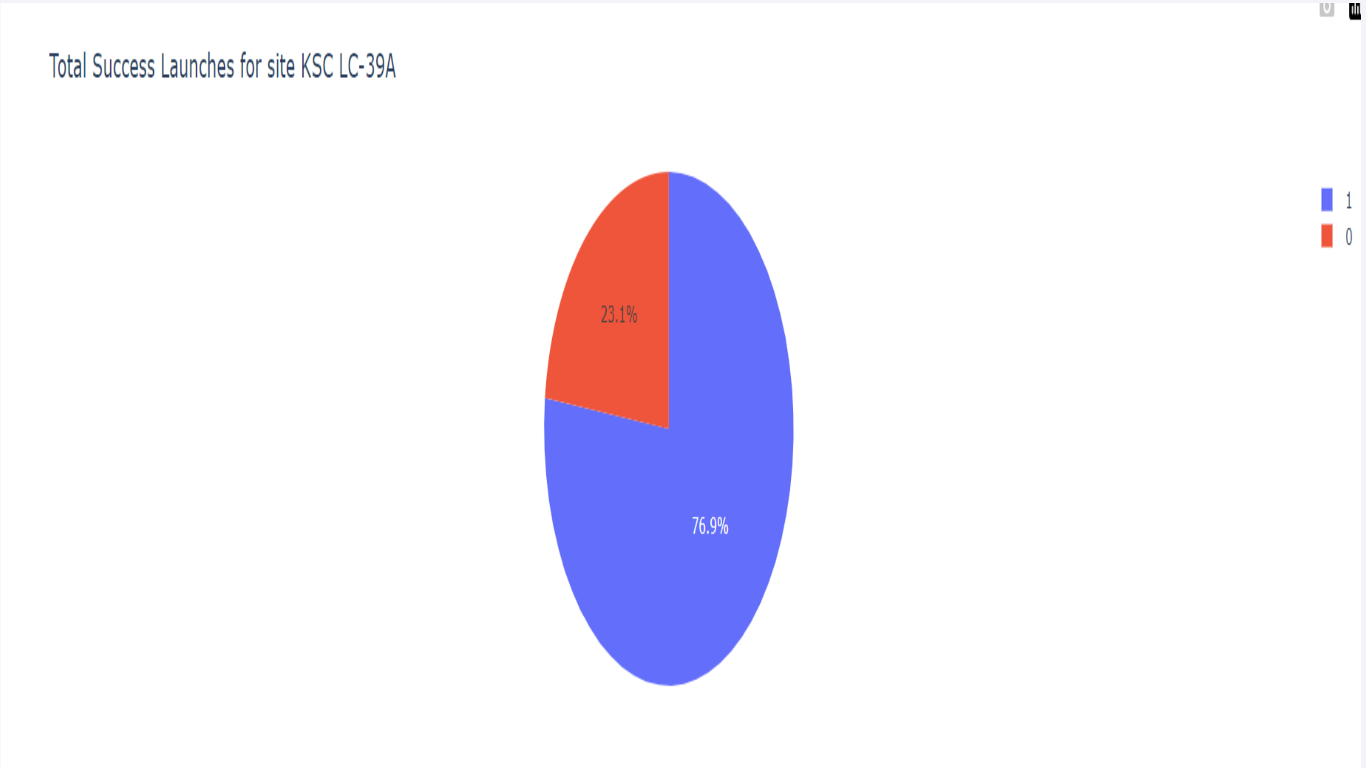
Pie Chart showing of launch success count for all sites

- On the Pie chart KSC LC-39A appears to has largest launch success.

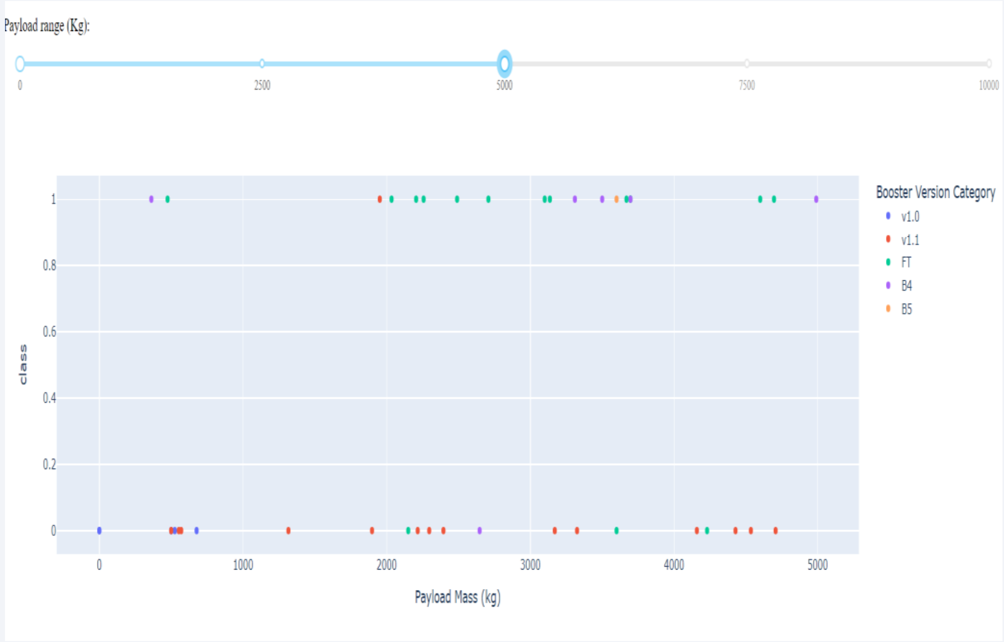


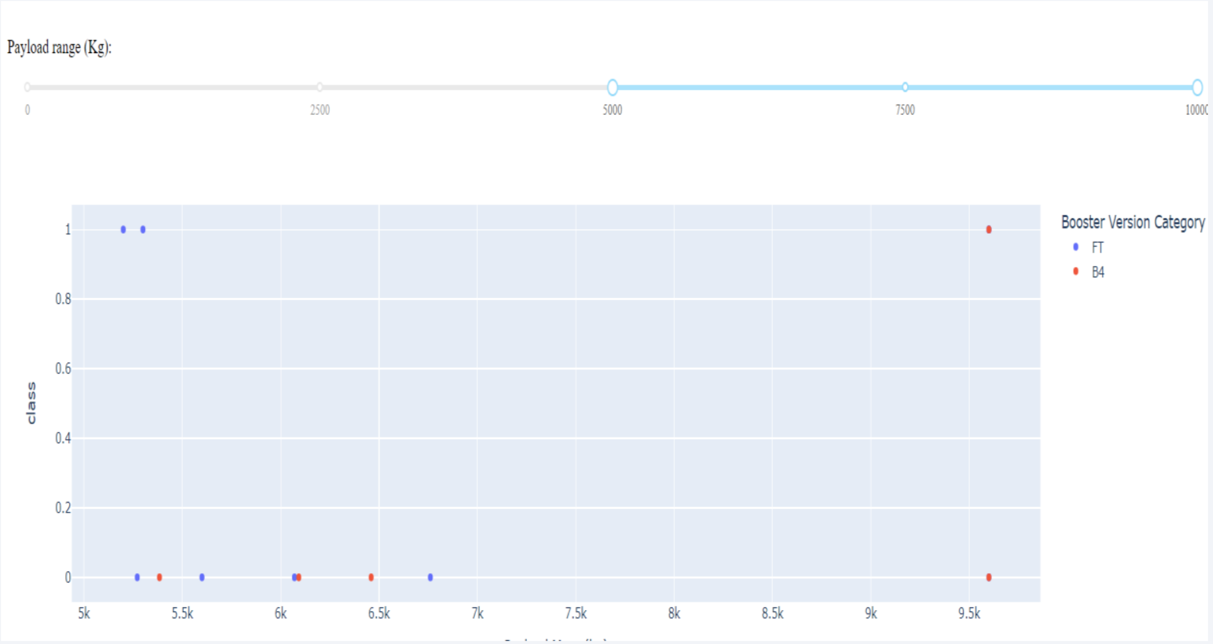
Pie Chart for launch site with highest launch success ratio

- The pie chart clearly shows that KSC LC-39A has the highest success ratio



Payload vs Launch Outcome Scatter Plot





Between Payload 0-5000 kg rate was higher compared to payload greater than 5000 kg

Section 5

Predictive Analysis (Classification)

Classification Accuracy

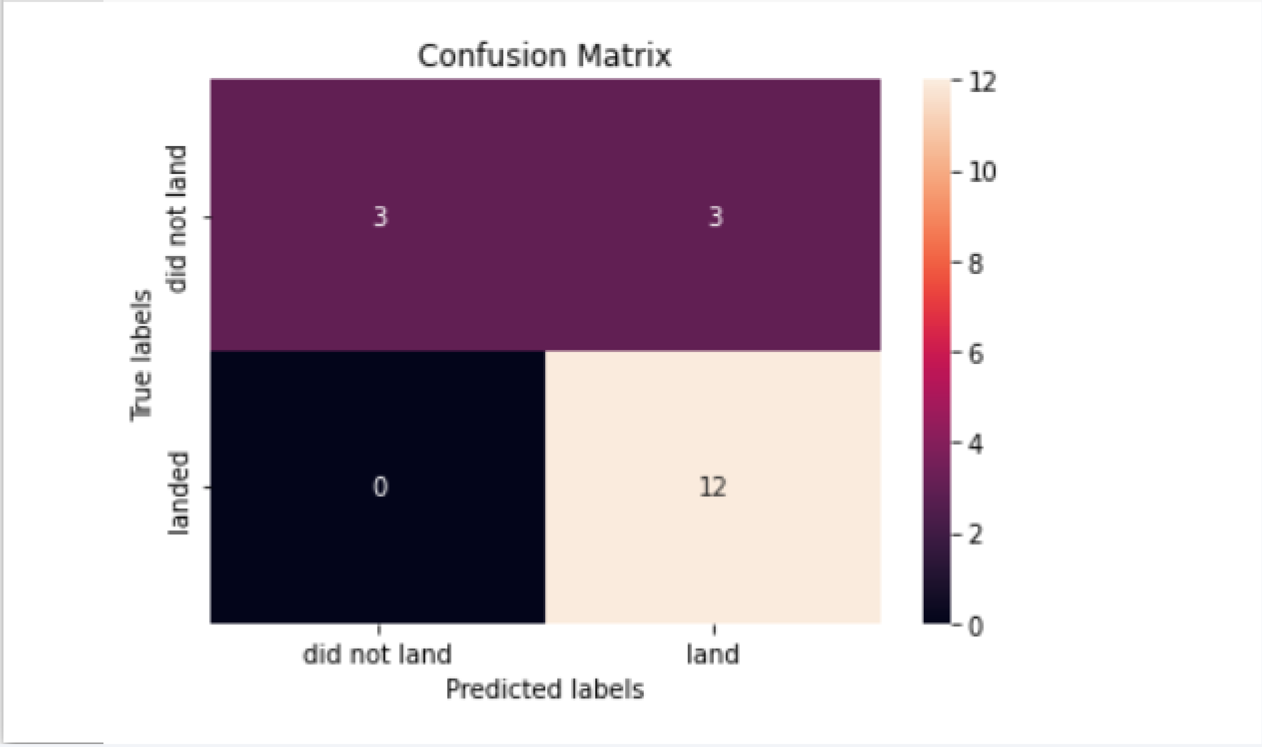
```
[60] methods = ['Logistic Regression', 'Support Vector Machine','Decisionn Tree', 'K Nearest Neighbor']

     evaluations_test = [logreg_accuracy,svm_accuracy,tree_accuracy,KNN_accuracy ]
     evaluations_train = [logreg_train_accuracy,svm_train_accuracy,tree_train_accuracy,knn_train_accuracy ]

     df_eval = pd.DataFrame({'Methods' : methods, 'Test_Accuracy' : evaluations_test, 'Train_Accuracy' :
                             evaluations_train})

     df_eval['Test_Accuracy'] = df_eval['Test_Accuracy'].apply(lambda x: round(x * 100,2))
     df_eval['Train_Accuracy'] = df_eval['Train_Accuracy'].apply(lambda x: round(x * 100,2))
     df_eval.sort_values(by=['Test_Accuracy','Train_Accuracy'],ascending=False,inplace=True,
                         ignore_index=True)
     df_eval
```

	Methods	Test_Accuracy	Train_Accuracy
0	Decisionn Tree	83.33	88.93
1	Support Vector Machine	83.33	84.82
2	K Nearest Neighbor	83.33	84.82
3	Logistic Regression	83.33	84.64

- By using the code ,we identify that the best algorithm to be the Decision Tree Algorithm which has the highest accuracy.

Confusion Matrix

- The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes.



Conclusions

- The Tree Classifier Algorithm is the best Machine Learning approach for this dataset.
- The low weighted payloads 5000kg and below performed better than the heavy weighted payloads greater than 5000 kg.
- After a plateau from 2010 to 2013, the success rate for SpaceX launches increased
- KSC LC-39A has the most successful launches of any sites
- SSO orbit has the greatest success rate

Thank you!