IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Alejandro Achkienasi
October 2023

Outline

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

Executive Summary

- Summary of methodologies
  - ❖ Data collection trough API
  - ❖ Webscraping
  - ❖ Data wrangling
  - ❖ EDA with SQL
  - ❖ EDA with Visualization
  - ❖ Interactive Visual Analytics with Folium lab
  - ❖ Interactive Dashboard with Ploty Dash
  - ❖ Machine Learning Prediction
- Summary of all results
  - ❖ EDA conclusions.
  - ❖ Interactive Visual Analytics and Dashboard with Folium and Ploty Dash.
  - ❖ Prediction analysis and Performance Comparison between different learning algorithms

Introduction

- Project background and context

  The commercial space age is here, companies are making space travel affordable for everyone. Perhaps the most successful is SpaceX. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards 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.

- Problems you want to find answers

  - ❖ Determine the price of each if each launch, by gathering information about Space X and creating dashboards for your team.
  - ❖ Determine if SpaceX will reuse the first stage.
  - ❖ Train a machine learning model and use public information to predict if SpaceX will reuse the first stage.

Section 1

Methodology

Methodology

Executive Summary

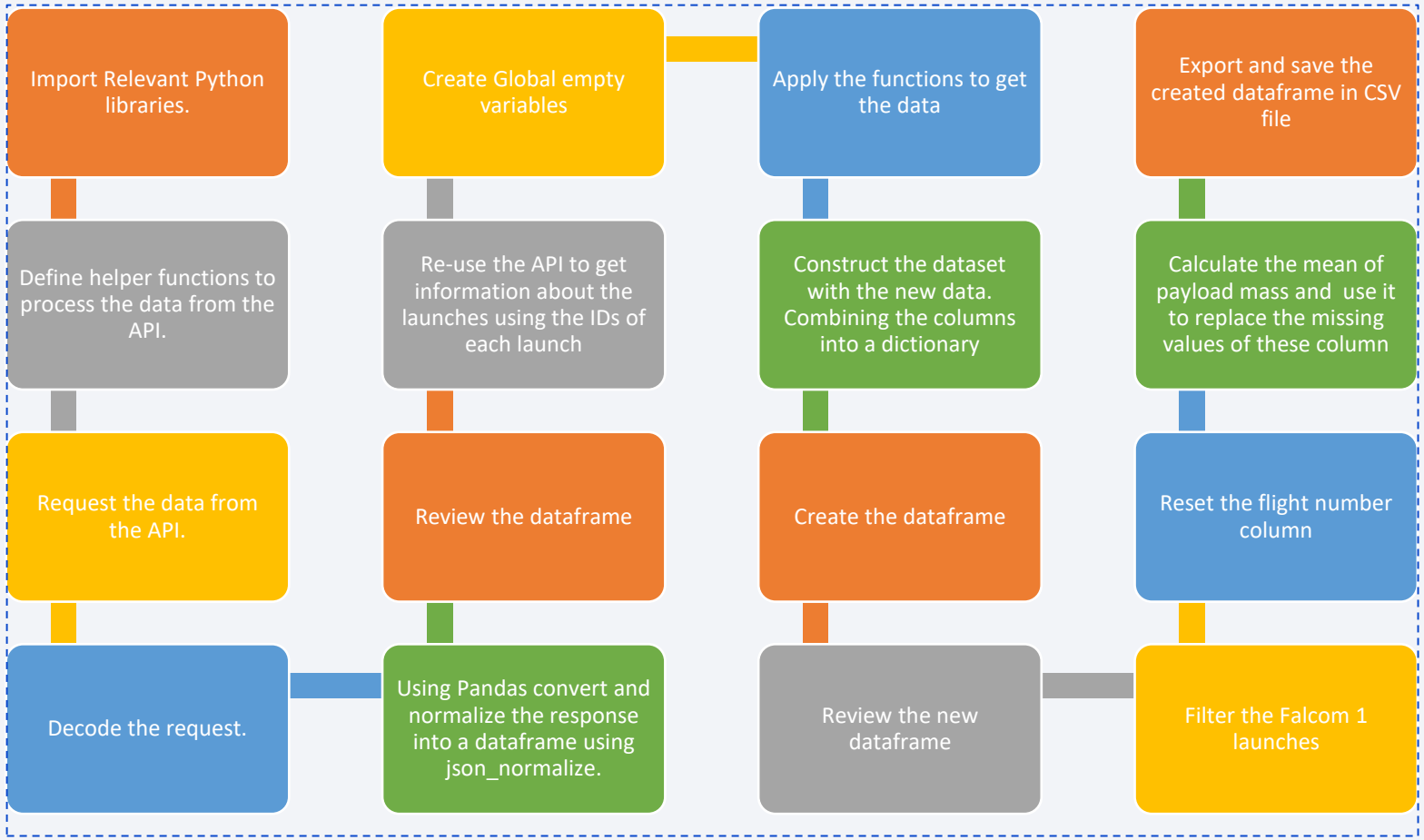
- Data collection methodology:
  - Data was collected using Python request to the SpaceX API and Web scrap with BeatifulSoup from Wikipedia.
- Perform data wrangling
  - Data was processed with EDA determining the training labels.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

  Create a column for the class, standardize the data, Split into training data and test data.

  Find best Hyperparameter for SVM, Classification Trees and Logistic Regression. Find the method performs best using test data

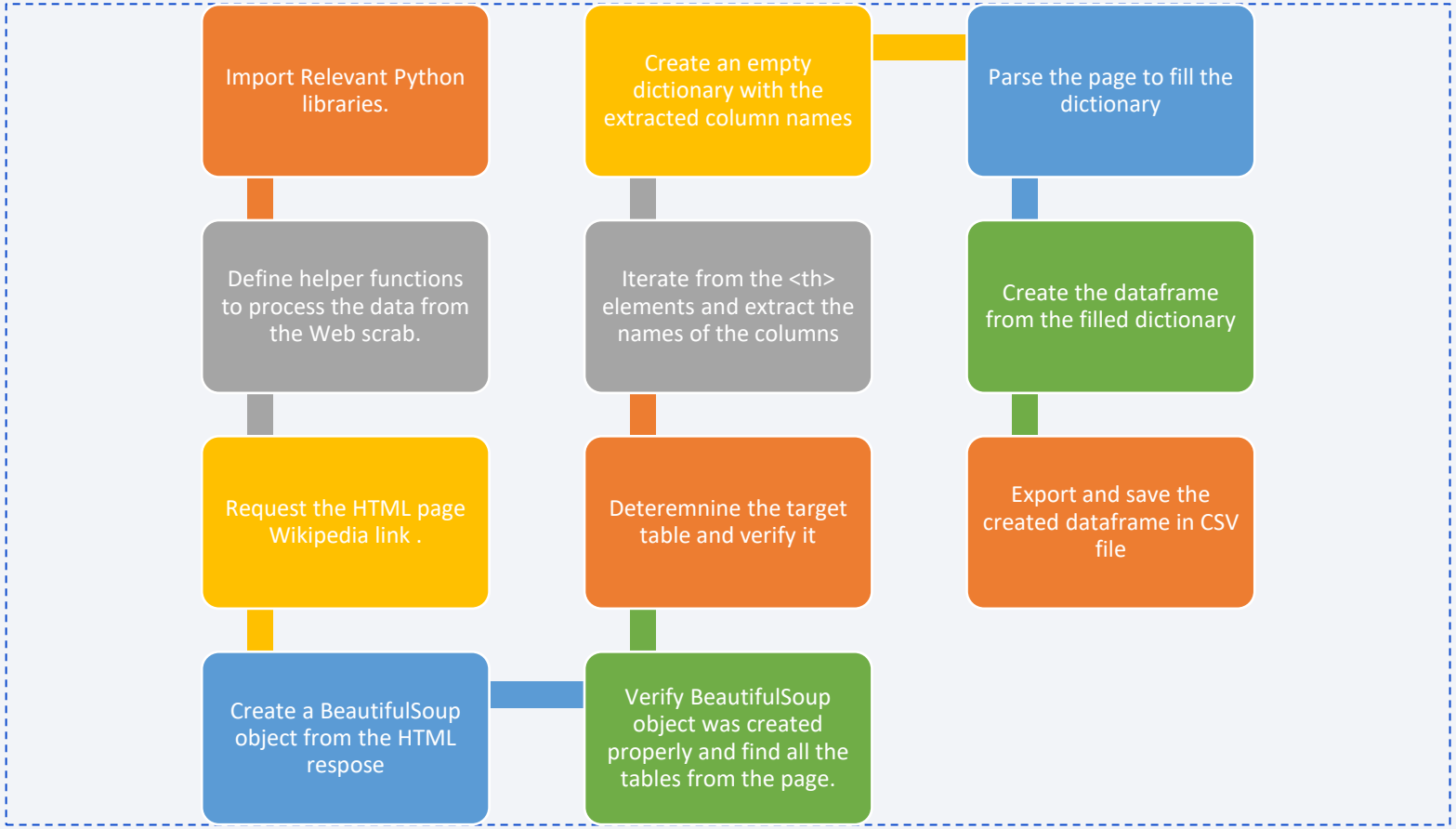
Data Collection – SpaceX API

- https://github.com/Alejandro9108/data_science_ibm/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



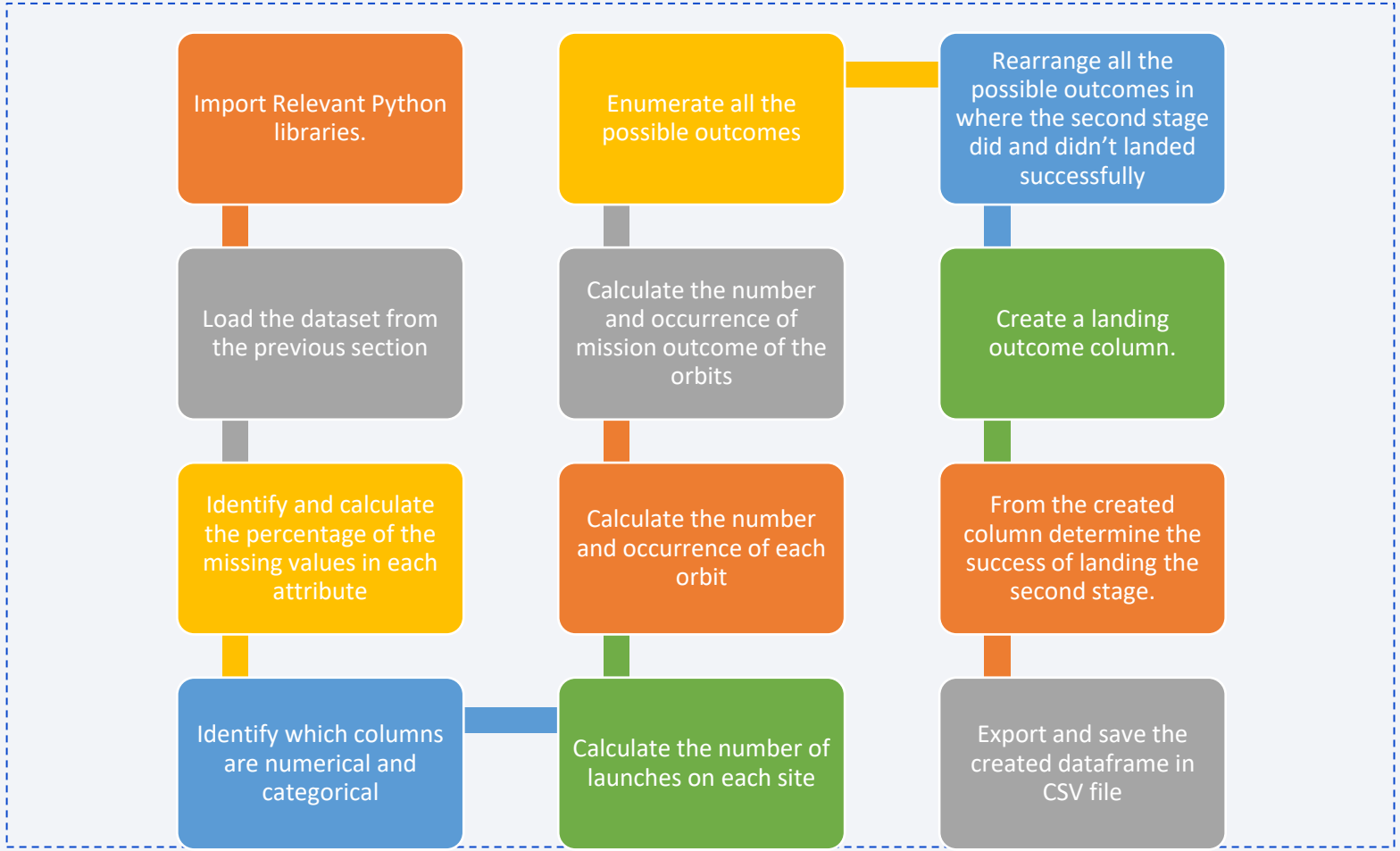
Data Collection - Scraping

- https://github.com/Alejandro9108/data_science_ibm/blob/main/jupyter-labs-webscraping.ipynb



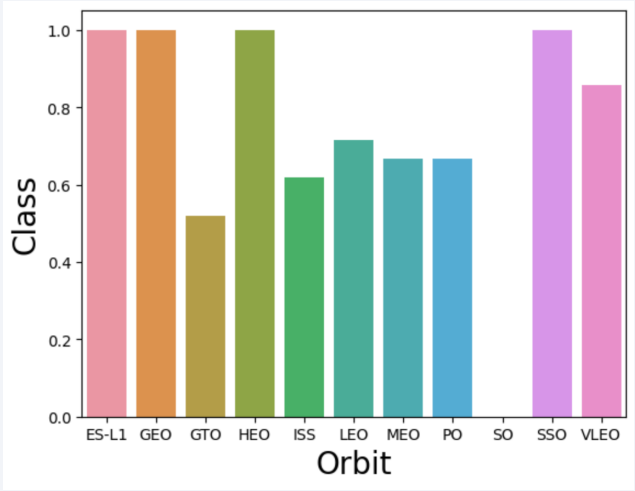
Data Wrangling

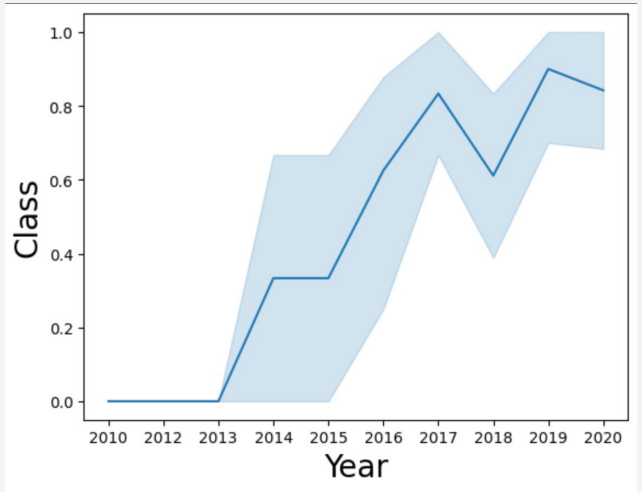
- https://github.com/Alejandro9108/data_science_ibm/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- Used Charts
  - Scatter Plots: For **exploring relationship, patterns and trends** between two variables (FlightNumber vs PayloadMass, FlightNumber vs LaunchSite, PayloadMass vs LaunchSite, FlightNumber vs Orbit, Payloadmass vs Orbit)
  - BarPlot: For visualizing the relashionship between success rate between orbit type, comparing different **classes.** (Orbit vs Class)
  - LineChart: For visualizing the Launch Success **trend** trough the years (Class vs Date)
- https://github.com/Alejandro9108/data_science_ibm/blob/main/jupyter-labs-eda-dataviz.ipynb





EDA with SQL

- SQL queries performed:
  - `%sql SELECT DISTINCT(Launch_Site) from SPACEXTABLE`
  - `%sql SELECT * from SPACEXTABLE where Launch_Site='CCAFS LC-40' LIMIT 5;`
  - `%sql SELECT sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where PAYLOAD_MASS__KG_ > 0;`
  - `sql SELECT AVG(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1' AND PAYLOAD_MASS__KG_ > 0;`
  - `%sql SELECT MIN(Date) from SPACEXTABLE where Landing_Outcome='Success (ground pad)';`
  - `%sql SELECT DISTINCT(Booster_Version) from SPACEXTABLE where Landing_Outcome='Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;`
  - `%sql select MISSION_OUTCOME, count(MISSION_OUTCOME) from SPACEXTBL GROUP BY MISSION_OUTCOME;`
  - `%sql SELECT Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTBL);`
  - `%sql SELECT substr(DATE, 6,2) as Month,MISSION_OUTCOME,BOOSTER_VERSION,LAUNCH_SITE from SPACEXTBL where substr(Date, 0,5)='2015';`
  - `%sql SELECT Landing_Outcome from SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' ORDER BY DATE DESC;`
- https://github.com/Alejandro9108/data_science_ibm/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Map objects added to folium map
- Markers and Cicles: Launch Sites (CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E). To represent the location and are of the launch sites
- Markers: Success and Fail launches (green and red respectively). To differentiate between success and fail launches and to mark where happen each of them.
- Lines: To the closest city, railway and highway. To represent the distance between two points.
- https://github.com/Alejandro9108/data_science_ibm/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Was added a pie chart to easily visualize what launch site has the largest successful launches and which site has the highest launch success rate.
- A scatter plot of the payload mass vs the class with a range slider for the payload mass was added for easily visualization of what payload range has the highest and the lowest launch success rate
- https://github.com/Alejandro9108/data_science_ibm/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- https://github.com/Alejandro9108/data_science_ibm/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

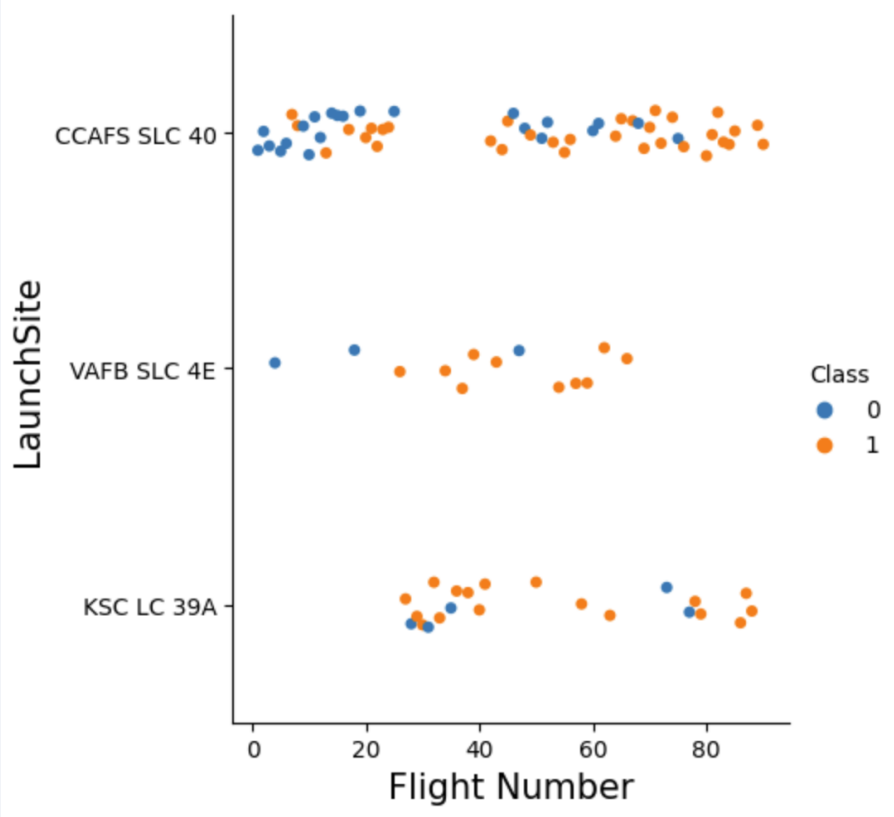
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

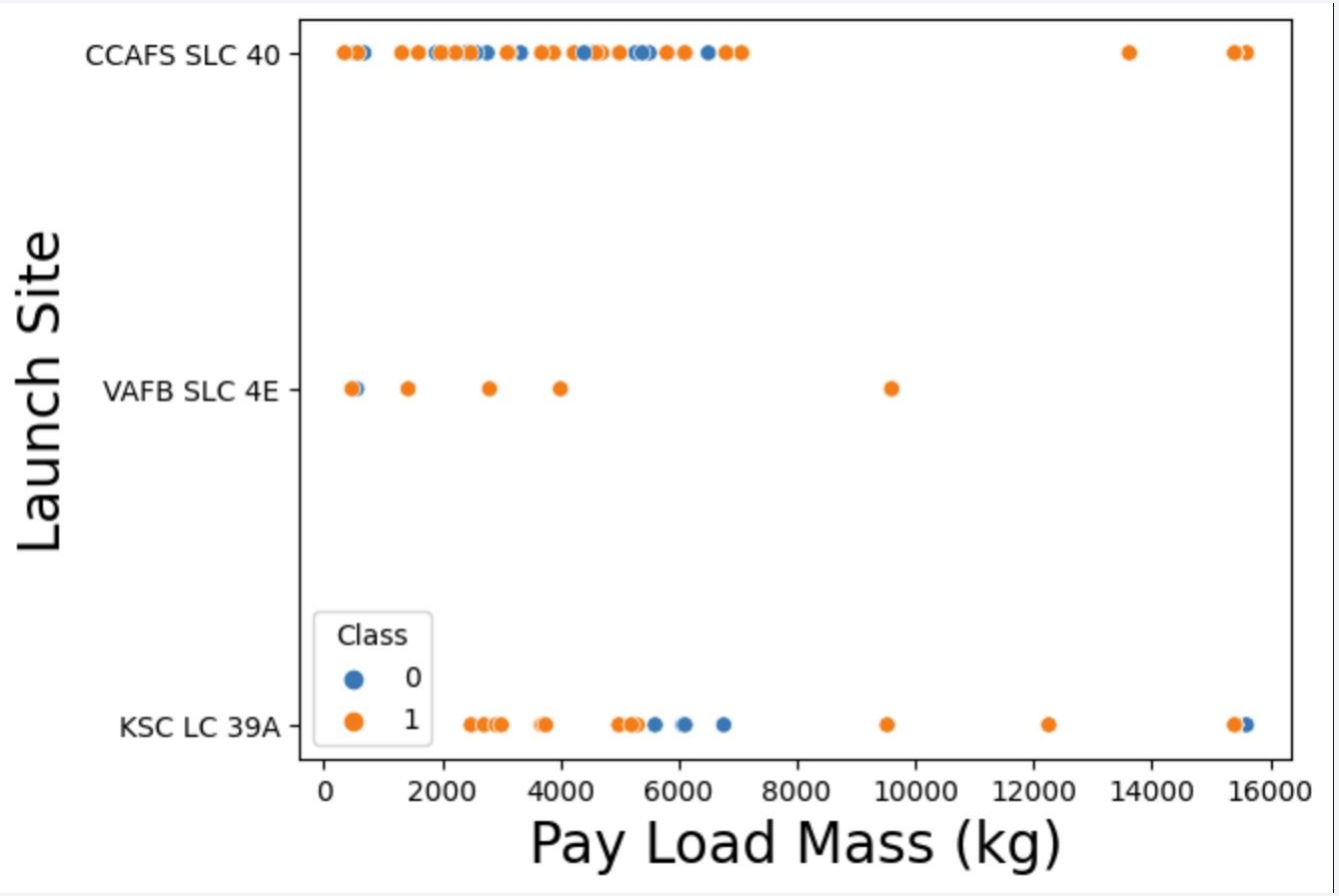
Flight Number vs. Launch Site

- No real correlation for success launch is found between the launch site and the flight number



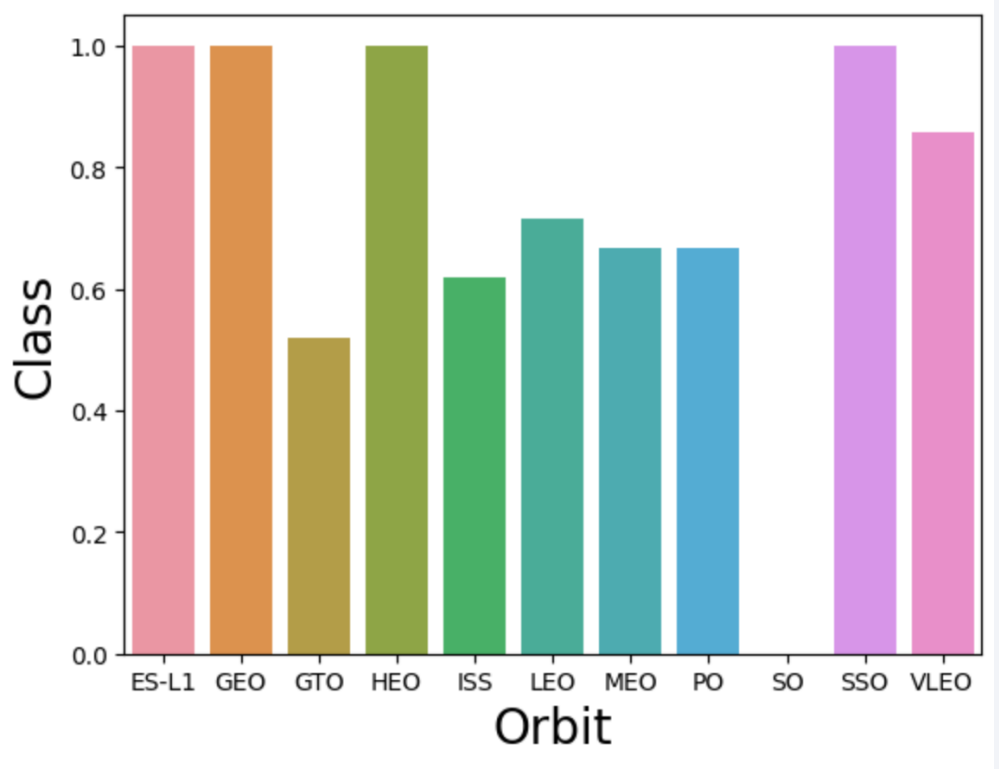
Payload vs. Launch Site

- For Payload over 8000 Kg the the launches are usually successful in all the launch sites.



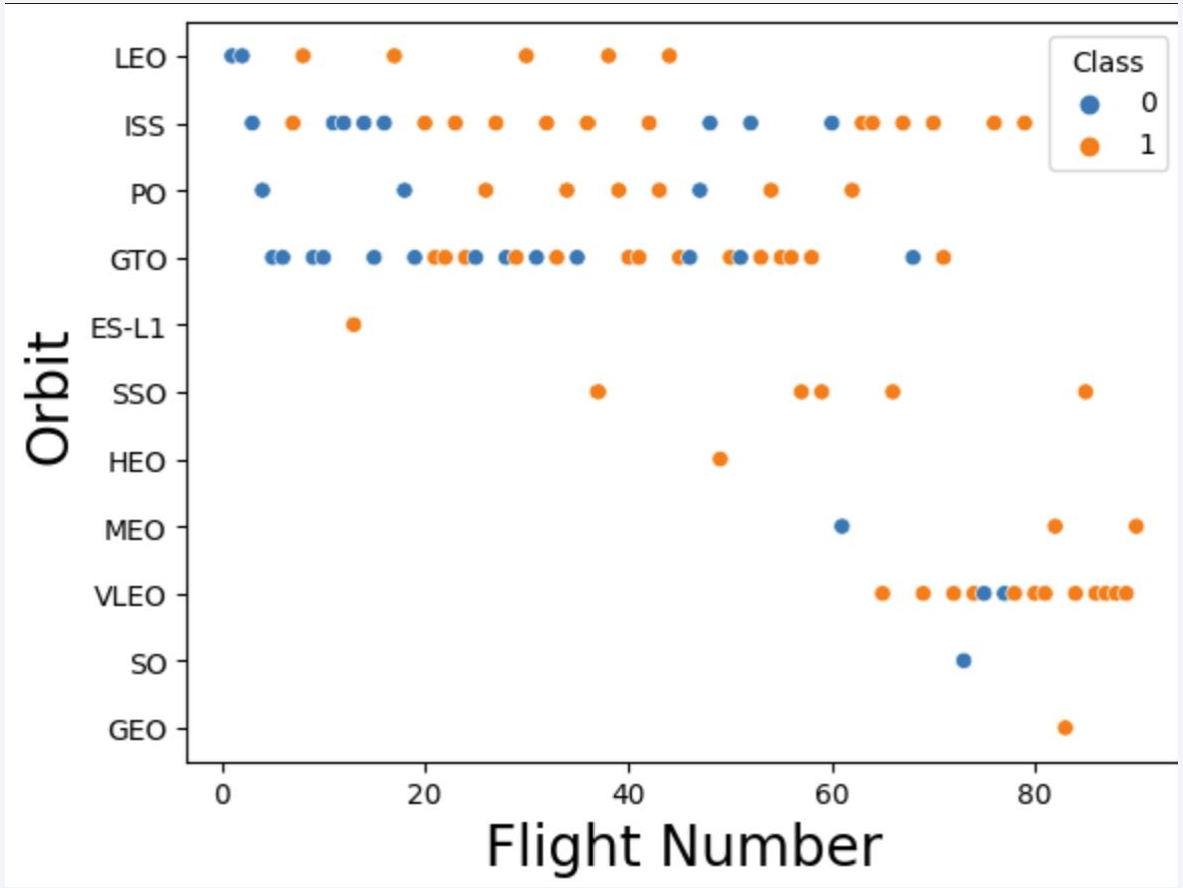
Success Rate vs. Orbit Type

- For the Orbits ES-L1, GEO, HEO and SSO always the first stage was recover.
- For SO was never succefully recover



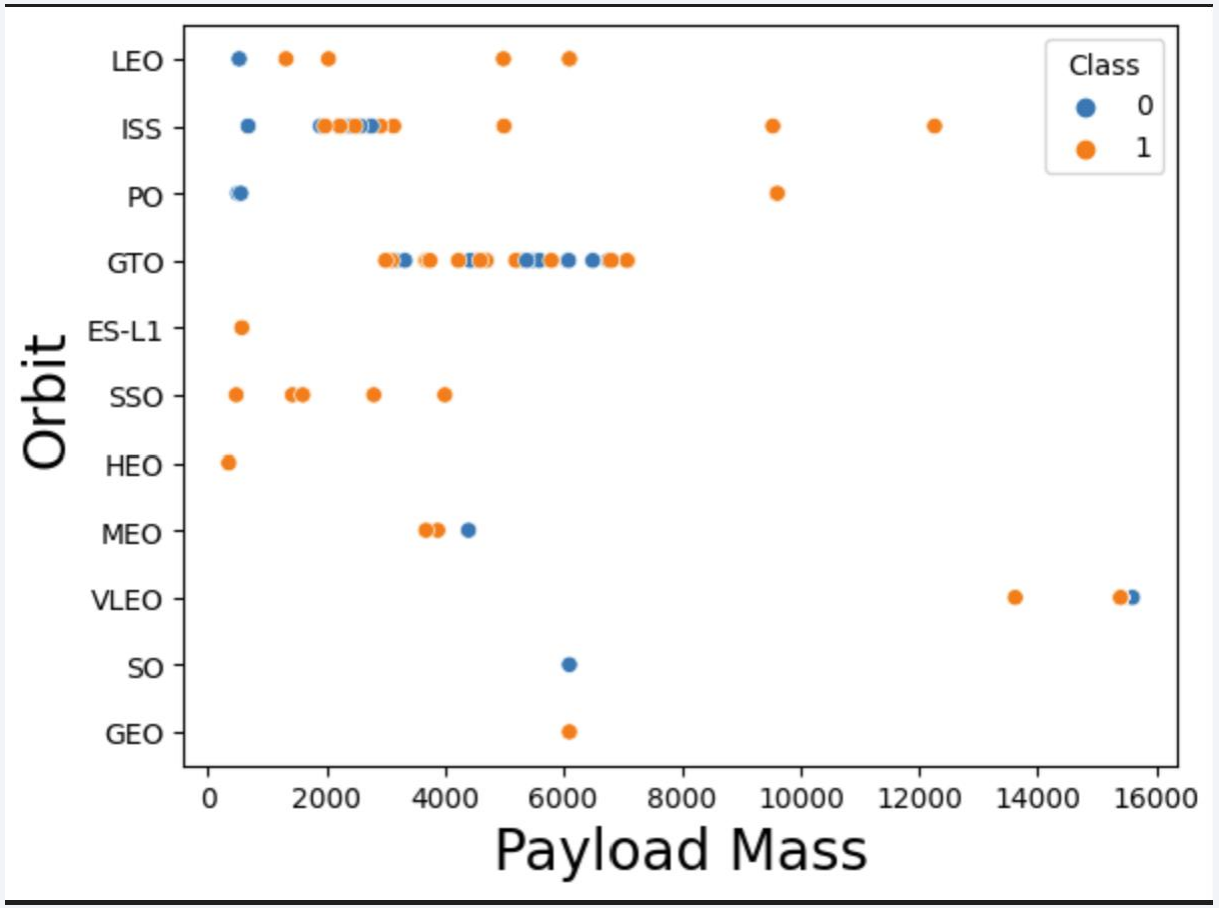
Flight Number vs. Orbit Type

- 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.



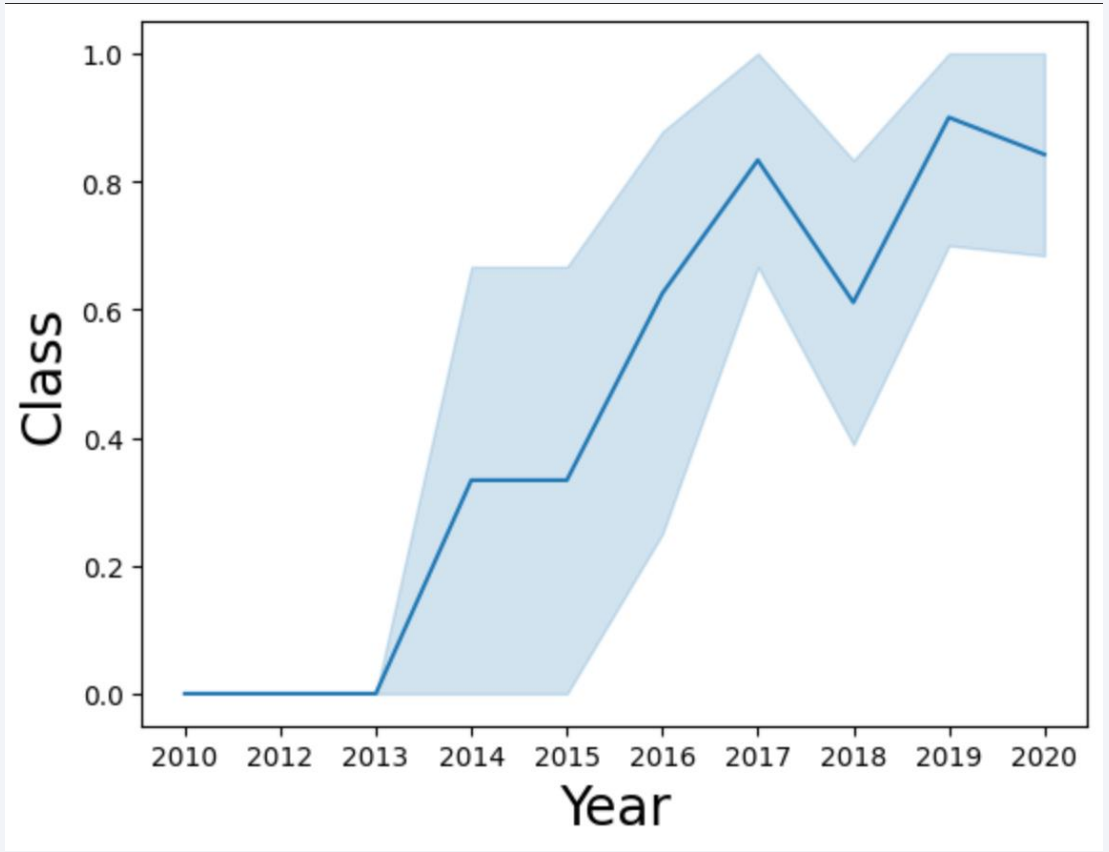
Payload vs. Orbit Type

- 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

- After 2013 the success had a general tendency to increase.
- In the years 2018 and 2020 the success rate decreased in relation to the previous year.



All Launch Site Names

- %sql SELECT DISTINCT(Launch_Site) from SPACEXTABLE
- Launch_Site
- CCAFS LC-40
- VAFB SLC-4E
- KSC LC-39A
- CCAFS SLC-40

  - The **SELECT DISTINCT** statement was used to select the unique values in the Launch_site column.

Launch Site Names Begin with 'CCA'

- %sql SELECT * from SPACEXTABLE where Launch_Site='CCAFS LC-40' LIMIT 5;

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- The **where** Launch_Site='CCAFS LC-40' to display only records of the launch site 'CCAFS LC-40 additionally the clause **LIMIT** 5 was used to display only 5 records.

Total Payload Mass

```
%sql SELECT sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where
PAYLOAD_MASS__KG_ > 0;
```

```
sum(PAYLOAD_MASS__KG_)

619967
```

- The **SELECT SUM** statement was used to calculate the total payload mass in addition with the **where** clause to select the records with positive value.

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) from SPACEXTABLE where
Booster_Version='F9 v1.1' AND PAYLOAD_MASS__KG_ > 0;
```

AVG(PAYLOAD_MASS__KG_)

2928.4

- The **SELECT AVG** statement was used to calculate the average value of Payload Mass additionally using **where** & **AND** clauses to average only the records with Booster_Version='F9 v1.1' and PAYLOAD_MASS__KG_ > 0

First Successful Ground Landing Date

%sql SELECT MIN(Date) from SPACEXTABLE where Landing_Outcome='Success (ground pad)';

MIN(Date)

2015-12-22

- The **SELECT MIN** statement was used to select the first successful landing outcome additionally the **where** clause was used to state the type of outcome searched (Success ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT DISTINCT(Booster_Version) from SPACEXTABLE where
Landing_Outcome='Success (drone ship)' AND PAYLOAD_MASS__KG_
BETWEEN 4000 AND 6000;
```

Booster_Version

F9 FT B1022

F9 FT B1026

F9 FT B1021.2

F9 FT B1031.2

- The **SELECT DISTINCT** statement was used to select the unique values in Booster_version additionally the **where** & **AND** clauses were used to state that the desire records are only where the landing outcome was success on drone ship and the payload mass was between 4000 and 6000 kg

Total Number of Successful and Failure Mission Outcomes

```
%sql select MISSION_OUTCOME, count(MISSION_OUTCOME) from SPACEXTBL
GROUP BY MISSION_OUTCOME;
```

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

- The **COUNT** statement was used to count each type of Mission outcome additionally the **GROUP BY** clause was used to concentrate the different types of outcomes.

Boosters Carried Maximum Payload

%sql SELECT Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTBL);

Booster_Version	
F9 B5 B1048.4	F9 B5 B1049.5
F9 B5 B1049.4	F9 B5 B1060.2
F9 B5 B1051.3	F9 B5 B1058.3
F9 B5 B1056.4	F9 B5 B1051.6
F9 B5 B1048.5	F9 B5 B1060.3
F9 B5 B1051.4	F9 B5 B1049.7

- The **where** clause & **SELECT MAX** sub-querie were used to state that the desire records are only where the payload mass was maximum

2015 Launch Records

%sql SELECT substr(DATE, 6,2) as Month,MISSION_OUTCOME,BOOSTER_VERSION,LAUNCH_SITE from SPACEXTBL where substr(Date, 0,5)='2015'

Month	Mission_Outcome	Booster_Version	Launch_Site
10	Success	F9 v1.1 B1012	CCAFS LC-40
11	Success	F9 v1.1 B1013	CCAFS LC-40
02	Success	F9 v1.1 B1014	CCAFS LC-40
04	Success	F9 v1.1 B1015	CCAFS LC-40
04	Success	F9 v1.1 B1016	CCAFS LC-40
06	Failure (in flight)	F9 v1.1 B1018	CCAFS LC-40
12	Success	F9 FT B1019	CCAFS LC-40

The substr(Date, 6,2) and substr(Date,0,5)='2015' were used to select the month and years

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

%sql SELECT Landing_Outcome from SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' ORDER BY DATE DESC;

Landing_Outcome				
No attempt	Success (ground pad)	Controlled (ocean)	Controlled (ocean)	No attempt
Success (ground pad)	Failure (drone ship)	Failure (drone ship)	No attempt	No attempt
Success (ground pad)	Success (drone ship)	Precluded (drone ship)	No attempt	No attempt
Success (drone ship)	Success (drone ship)	No attempt	No attempt	Failure (parachute)
Success (ground pad)	Failure (drone ship)	Failure (drone ship)	Controlled (ocean)	
Success (drone ship)	Failure (drone ship)	No attempt	Uncontrolled (ocean)	
Success (drone ship)	Success (ground pad)	Uncontrolled (ocean)	No attempt	

- The clause WHERE DATE BETWEEN was used to delimited the time period and the ORDER BY DATE to make the rank in descending order


Section 3
Launch Sites
Proximities Analysis

Launch sites' location markers on a global map

- The map displays the location of the launch sites in Florida and California.

Color-labeled launch outcomes on map

- Fig 1 color labeled launch outcome in CCAFS LC-40
- Fig 2 color labeled launch outcome in CCAFS SLC-40
- Fig 3 color labeled launch outcome in KSC LC-39A
- Fig 4 color labeled launch outcome in VAFB SLC-4E
- Green markers represent success
- Red markers represent failures

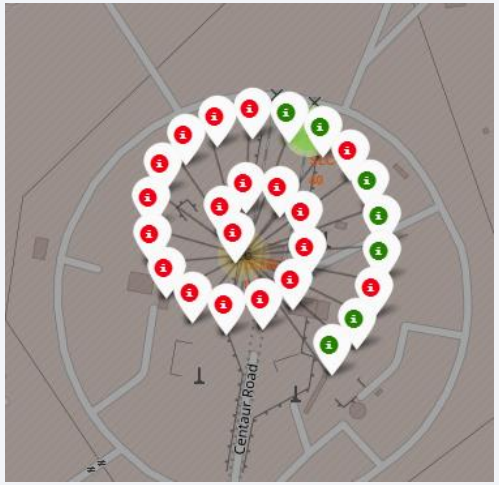


Fig. 1

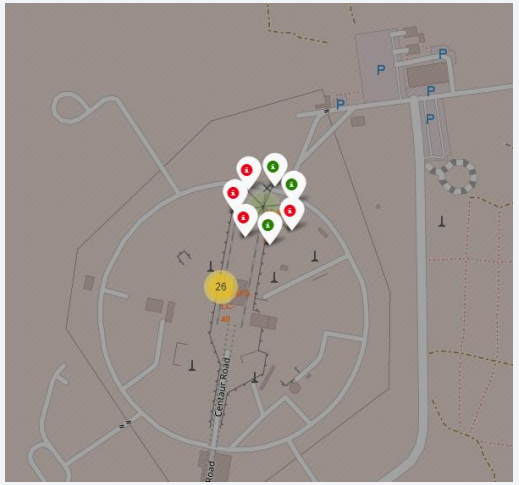


Fig. 2

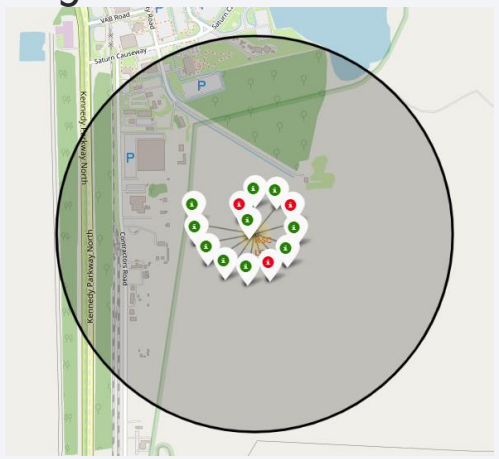


Fig. 3

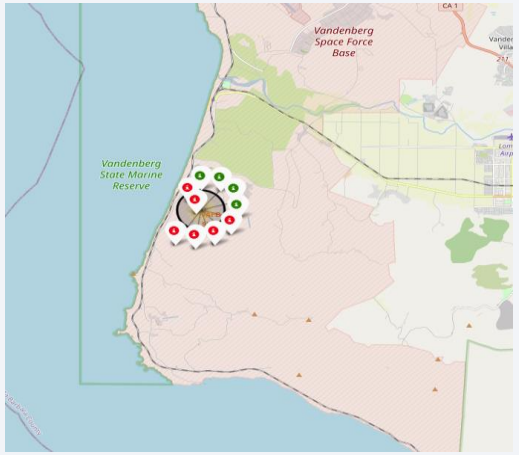
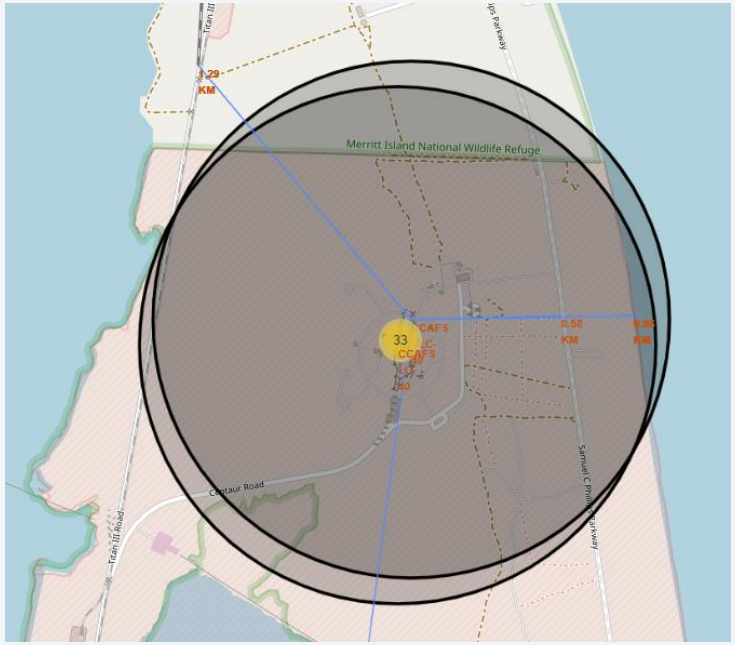


Fig. 4

Selected launch site to proximities of interest

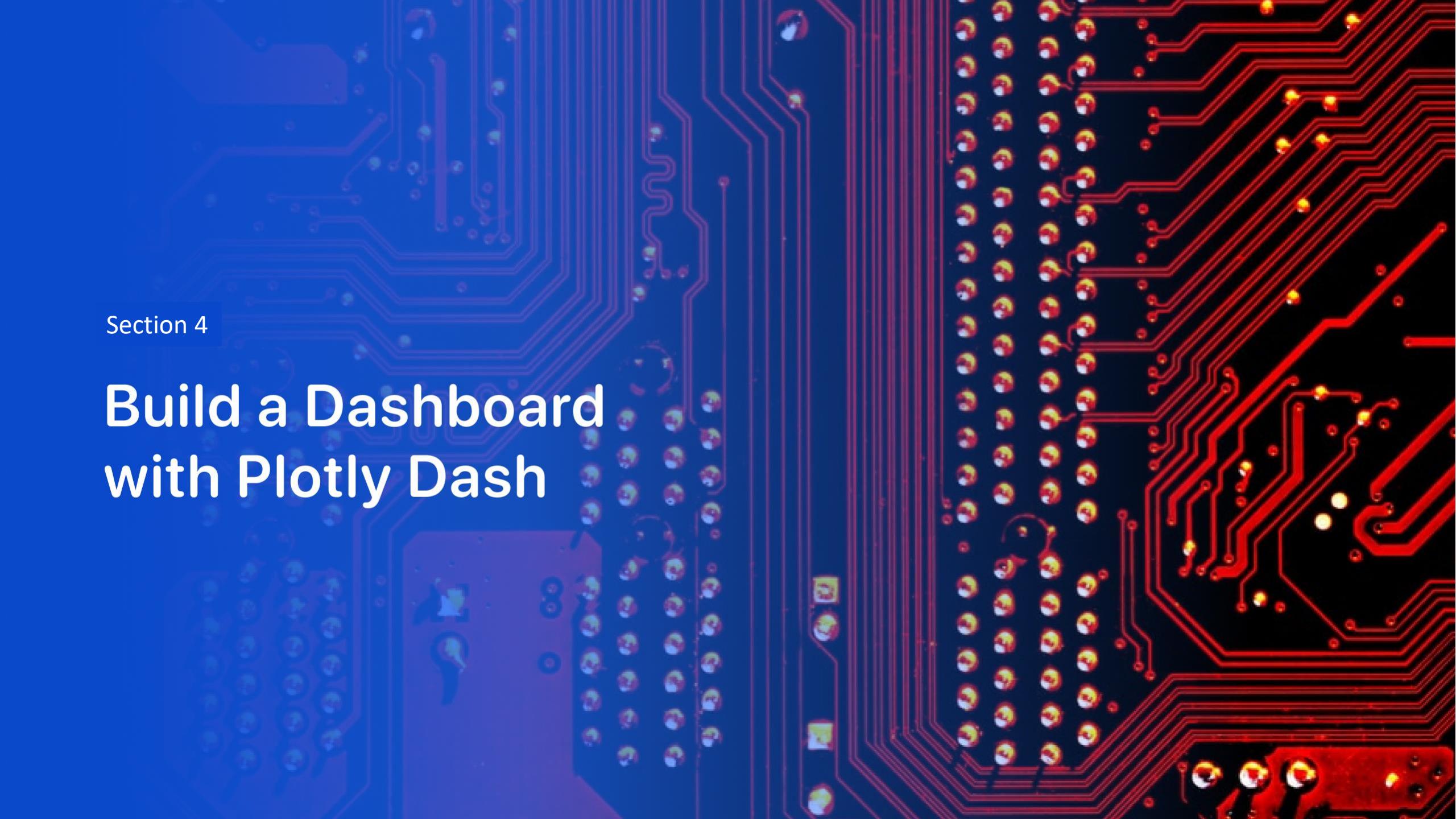
- Figure 1 shows plot of distance to railroad, city and coastline
- Figure 2 shows full view of distance plot to Melbourne City in Florida



distance_city = 51.43416999517233 km

distance_coastline = 0.583469536693414 4 km

distance_railroad = 1.286415258151074 6 km

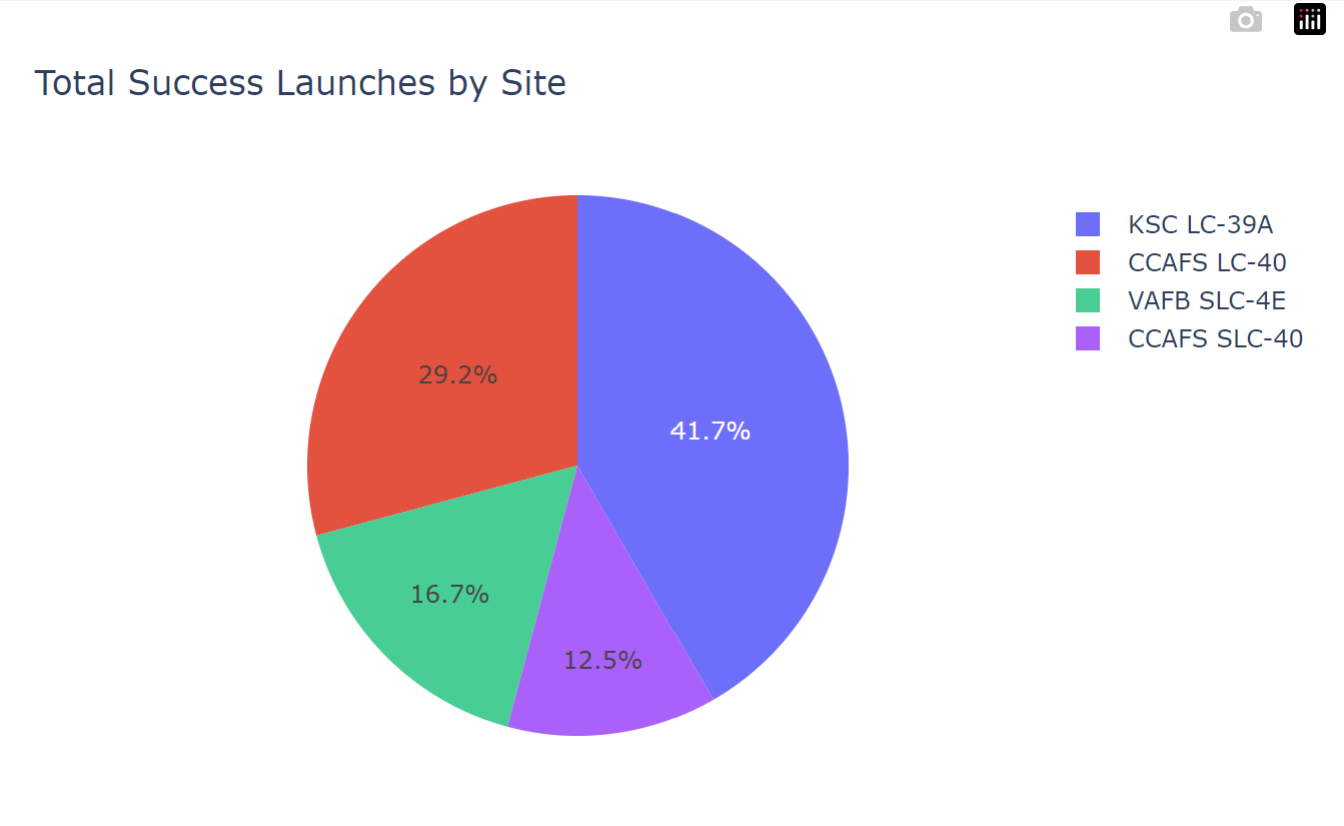

Section 4
Build a Dashboard with Plotly Dash

Launch success count for all sites

It is appreciated in the graph the ratio of success launches between all sites.

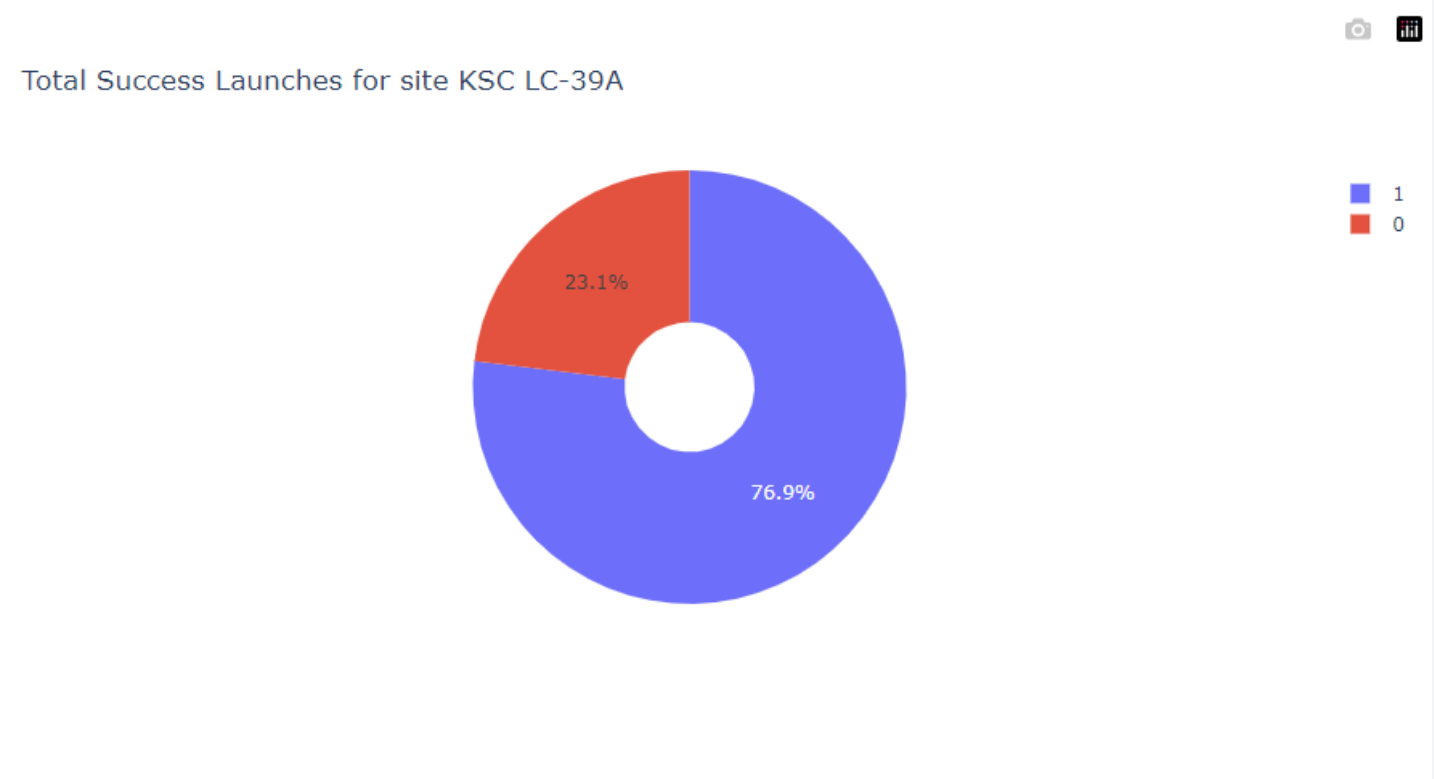
The site with higher success counts is KSC LC-39A.

The site with lower success count is CCAFS SLC-40



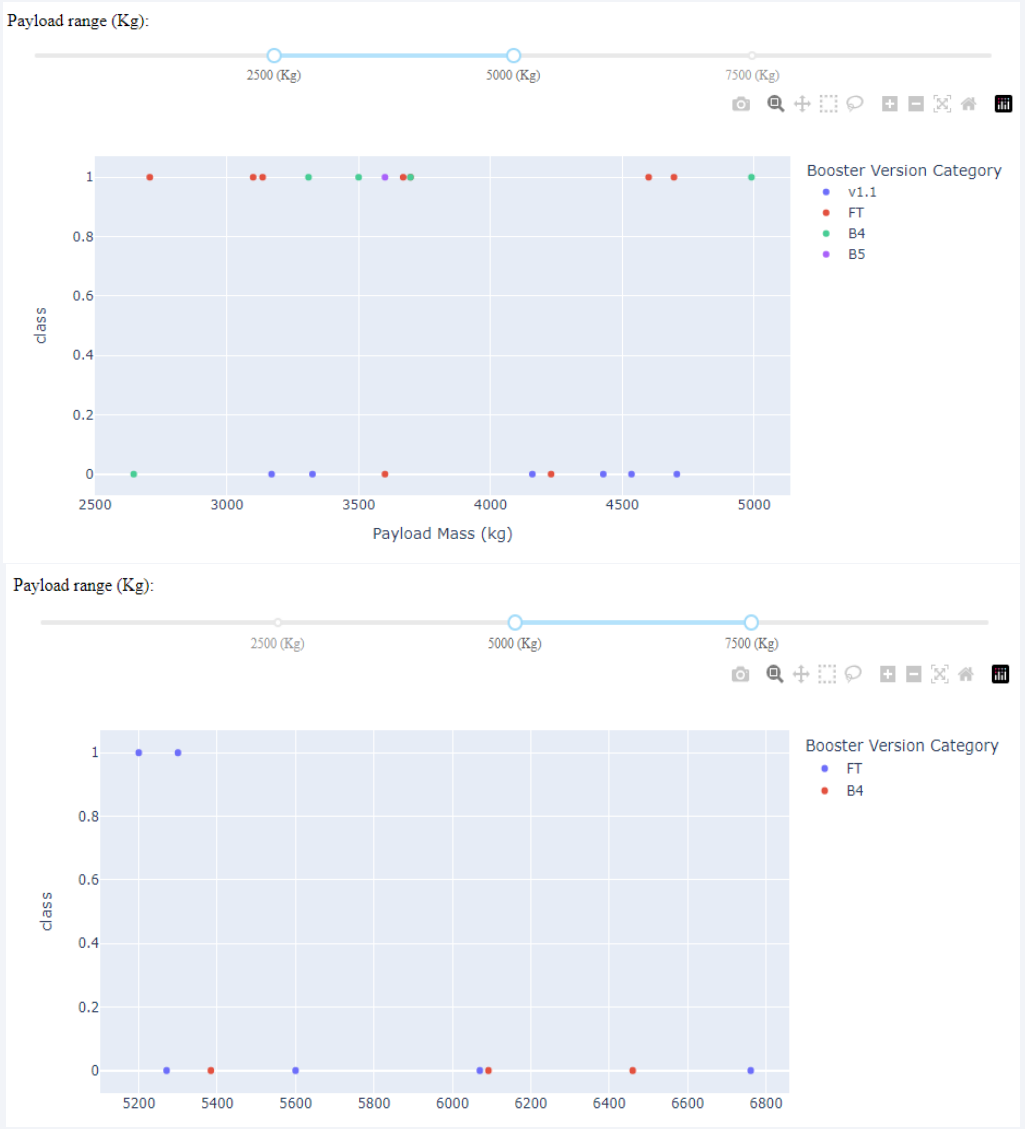
Success ratio of the launch site KSC LC-39A

- The success ratio of the launch site KSC LC-39A is 76.9% of success.



Payload vs. Launch Outcome scatter plot for all sites

- The first graph shows the relation of the launch outcome of all launch sites and payload mass range between 2500-5000kg.
- The second graph shows the relation of the launch outcome of all launch sites and payload mass range between 5000-7500kg.
- The first of the two selected ranges has a higher success rate.

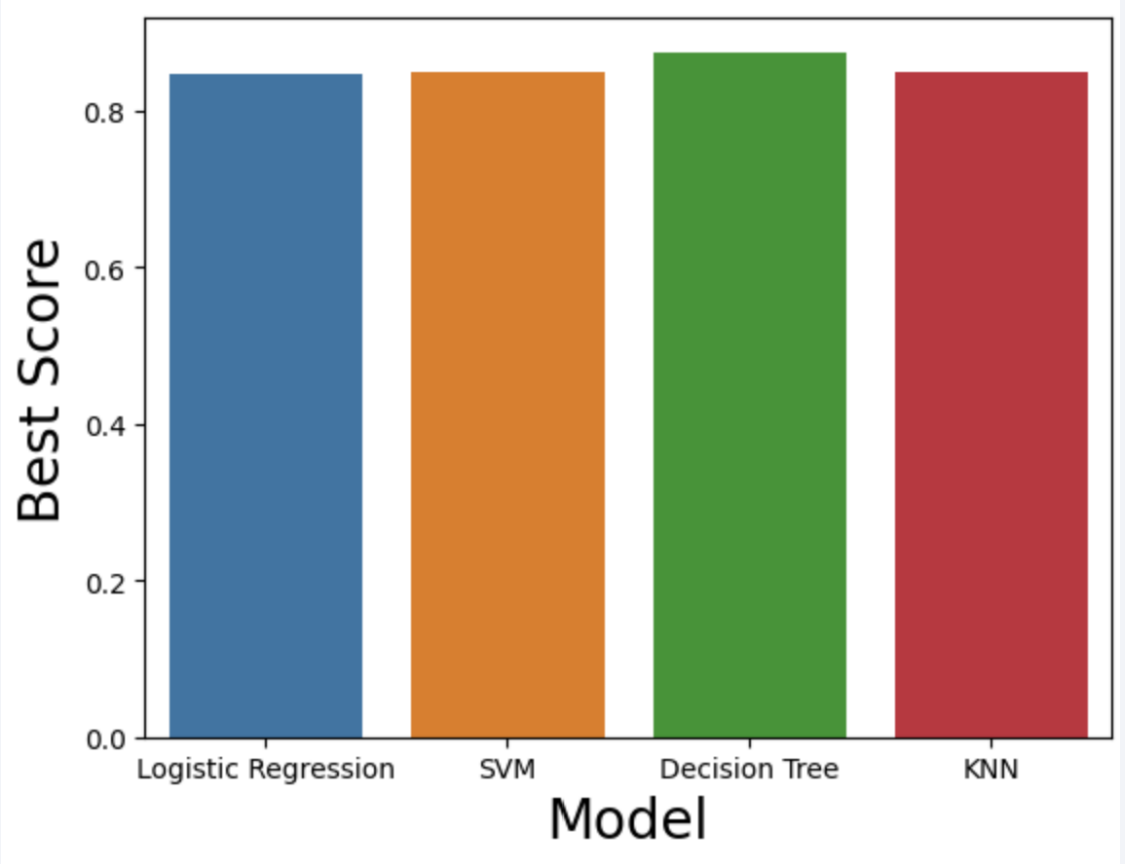


Section 5

Predictive Analysis (Classification)

Classification Accuracy

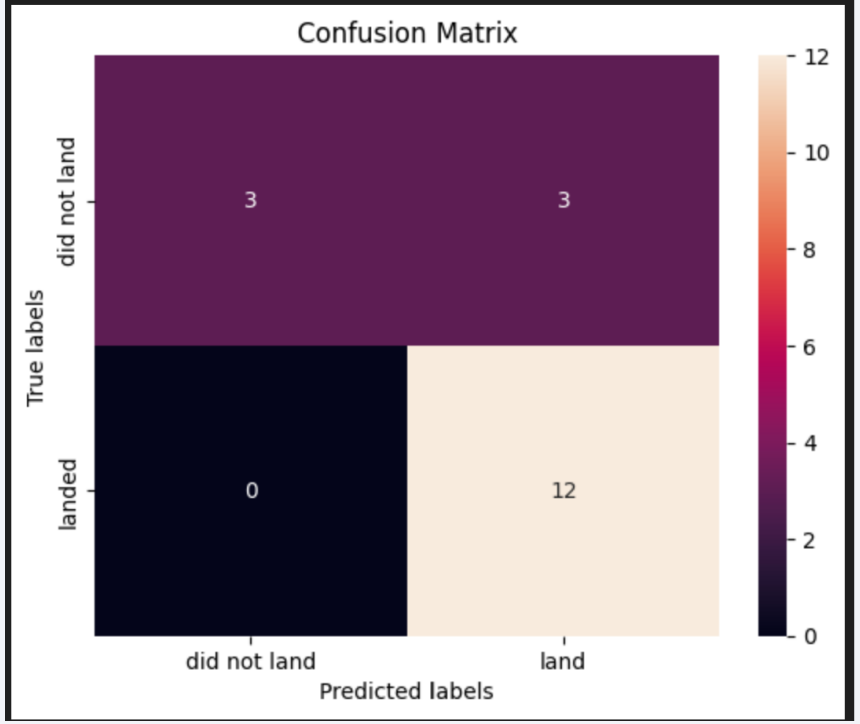
- All the models performed very similar, although the Decision tree model had a slightly better accuracy



Confusion Matrix

- The confusion matrix of the descision tree shows that the classifier can distinguish between different classes. With the biggest problem being the outliers.

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Conclusions

From EDA insights:

- Payloads over 8000 Kg the the launches are usually successful in all the launch sites.
- For the Orbits ES-L1, GEO, HEO and SSO always the first stage was recovered.
- For payloads between 4000-6000 kg the Booster_Versions that were successfully recoverd are:F9 FT B1022, F9 FT B1026, F9 FT B1021.2, F9 FT B1031.2.

From dashboard:

- The site with higher success counts is KSC LC-39A.with 76.9% of success.

From the prediction analysis:

- All the models performed very similar, although the Decision tree model had a slightly better accuracy.
- The confusion matrix shows that the classifier can distinguish between different classes. With the biggest problem being the outliers.

Thank you!