IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<MARCOS JONES DE OLIVEIRA>
<17/06/2023>

Outline

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

Executive Summary

- Summary of methodologies

Collection data from API and web page → Data Wrangling: transforming and mapping data → Exploratory data analysis (EDA) by using SQL → EDA by visualization → Interative visual analytics and dashboards → Predective analysis by machine learning

- Summary of all results

Summary: Select the best model for future prediction.

Introduction

- Project background and context
- **SpaceX's goal:**
- Sending spacecraft to the International Space Station.
- Starlink, a satellite internet constellation providing satellite Internet access.
- Sending manned missions to Space.
- Problems you want to find answers
- To determine the price of each launch.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Require the data from Apace API
  - Collect data from a Wikipedia page
- Perform data wrangling
  - Perform EDA to find some patterns
  - Determine what would be the label for training supervised model
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Compare logistic regression model,support vector machine learning tree decision classifier,KNN by using Grid SearchCV to select the best model

Data Collection

API

- spacex_url=https://api.spacexdata.com/v4/launches/past

    Required the data from Space API

    Clean the data

WEB

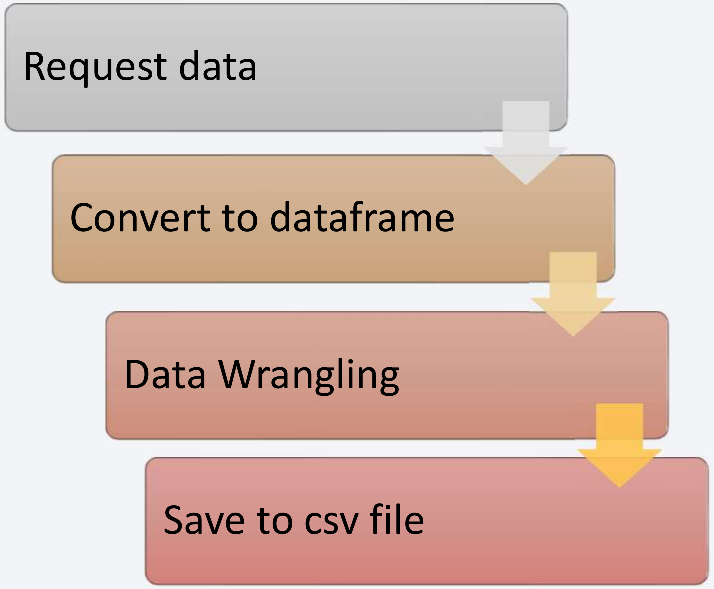
https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

    Extract a Falcon 9 launch records HTML table from Wikipedia

    Parse the table and convert it into a Pandas data frame

Data Collection – SpaceX API

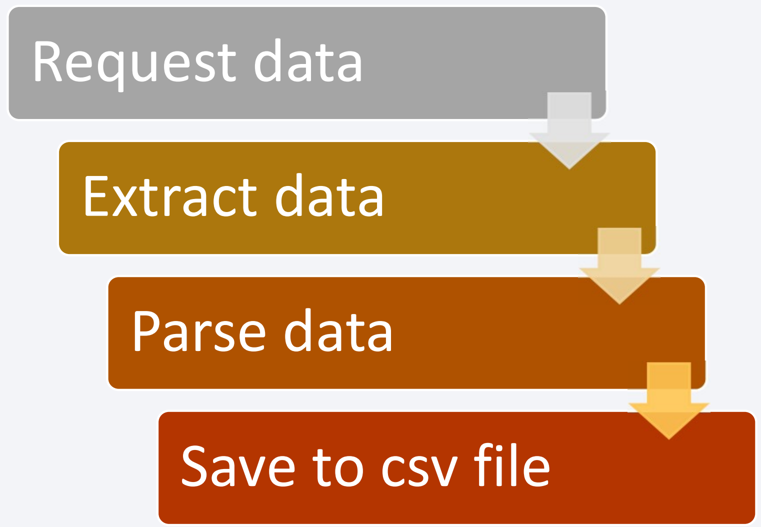
- Request data from SpaceX API
- Convert the json result into a dataframe
- Filter data frame to only Falcon 9` launches and data wrangling
- Export to csv
- Link :
- https://github.com/Jones021/mj/blob/main/spacex-data-collection-api.ipynb



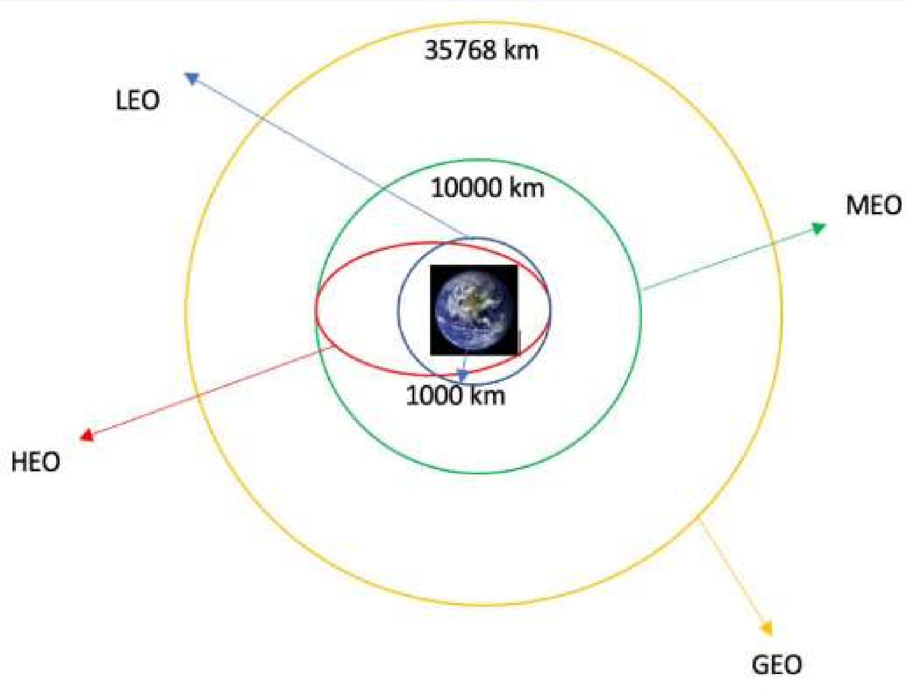
Data Collection - Scraping

- Request the Falcon9 Launch Wiki page from its URL
- Extract all column/variable names from the HTML table header
- Create a data frame by parsing the launch HTML tables
- Export to csv
- Link :

https://github.com/Jones021/jmo/blob/main/1.2%20SpaceX_Complete%20the%20Data%20Collection%20with%20Web%20Scraping%20lab.ipynb



Data Wrangling



Data Wrangling is the process of cleaning and unifying messy and complex data sets for easy access and Exploratory Data Analysis (EDA).

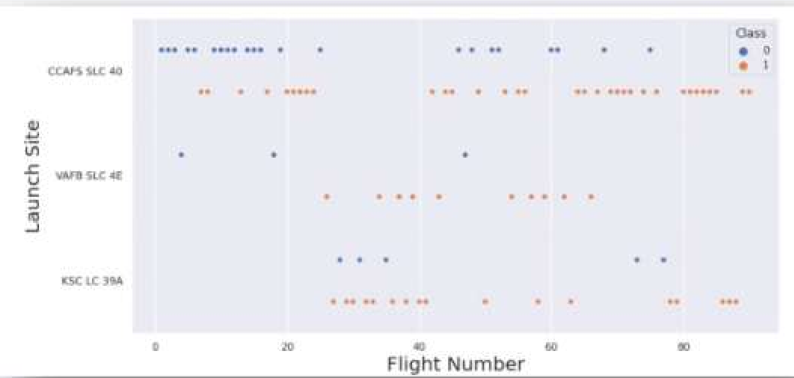
We will first calculate the number of launches on each site, then calculate the number and occurrence of mission outcome per orbit type.

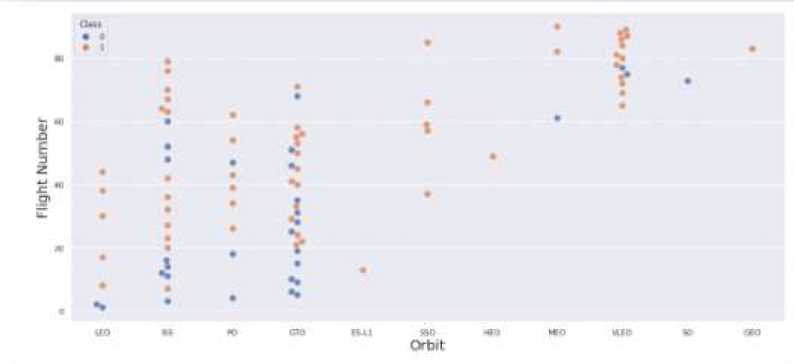
We then create a landing outcome label from the outcome column. This will make it easier for further analysis, visualization, and ML. Lastly, we will export

the result to a CSV.

Link: https://github.com/Jones021/jmo/blob/main/1.3%20SpaceX_Data%20Wrangling.ipynb

EDA with Data Visualization





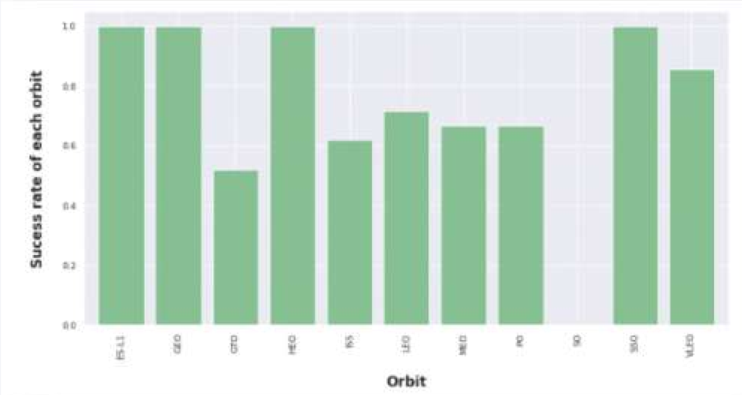
We first started by using scatter graph to find the relationship between the attributes such as between:

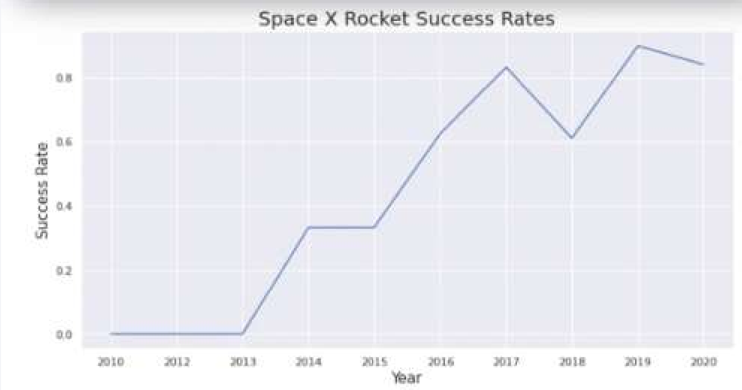
- Payload and Flight Number.
- Flight Number and Launch Site.
- Payload and Launch Site.
- Flight Number and Orbit Type.
- Payload and Orbit Type.

Scatter plots show dependency of attributes on each other. Once a pattern is determined from the graphs. It's very easy to see which factors affecting the most to the success of the landing outcomes.

https://github.com/Jones021/mj/blob/main/Exploringand%20preparingData-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with Data Visualization





Once we get a hint of the relationships using scatter plot. We will then use further visualization tools such as bar graph and line plots graph for further analysis.

Bar graphs is one of the easiest way to interpret the relationship between the attributes. In this case, we will use the bar graph to determine which orbits have the highest probability of success.

We then use the line graph to show a trends or pattern of the attribute over time which in this case, is used for see the launch success yearly trend.

We then use Feature Engineering to be used in success prediction in the future module by created the dummy variables to categorical columns.

https://github.com/Jones021/mj/blob/main/Exploringand%20preparingData-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

Using SQL, we had performed many queries to get better understanding of the dataset, Ex:

- Displaying the names of the launch sites.

- Displaying 5 records where launch sites begin with the string 'CCA'.

- Displaying the total payload mass carried by booster launched by NASA (CRS).

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

- Listing the date when the first successful landing outcome in ground pad was achieved.

- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.

- Listing the total number of successful and failure mission outcomes.

- Listing the names of the booster_versions which have carried the maximum payload mass.

- Listing the failed landing_outcomes in drone ship, their booster versions, and launch sites names for in year 2015.

- Rank the count of landing outcomes or success between the date 2010-06-04 and 2017-03-20, in descending order.

https://github.com/Jones021/mj/blob/main/eda-sql.ipynb

Build an Interactive Map with Folium

- To visualize the launch data into an interactive map. We took the latitude and longitude
- coordinates at each launch site and added a circle marker around each launch site with a
- label of the name of the launch site.
- We then assigned the dataframe launch_outcomes(failure,success) to classes 0 and 1 with
- Red and Green markers on the map in MarkerCluster().
- We then used the Haversine's formula to calculated the distance of the launch sites to
- various landmark to find answer to the questions of:
- • How close the launch sites with railways, highways and coastlines?
- • How close the launch sites with nearby cities?
- From: https://github.com/Jones021/mj/blob/main/launch_site_location.ipynb

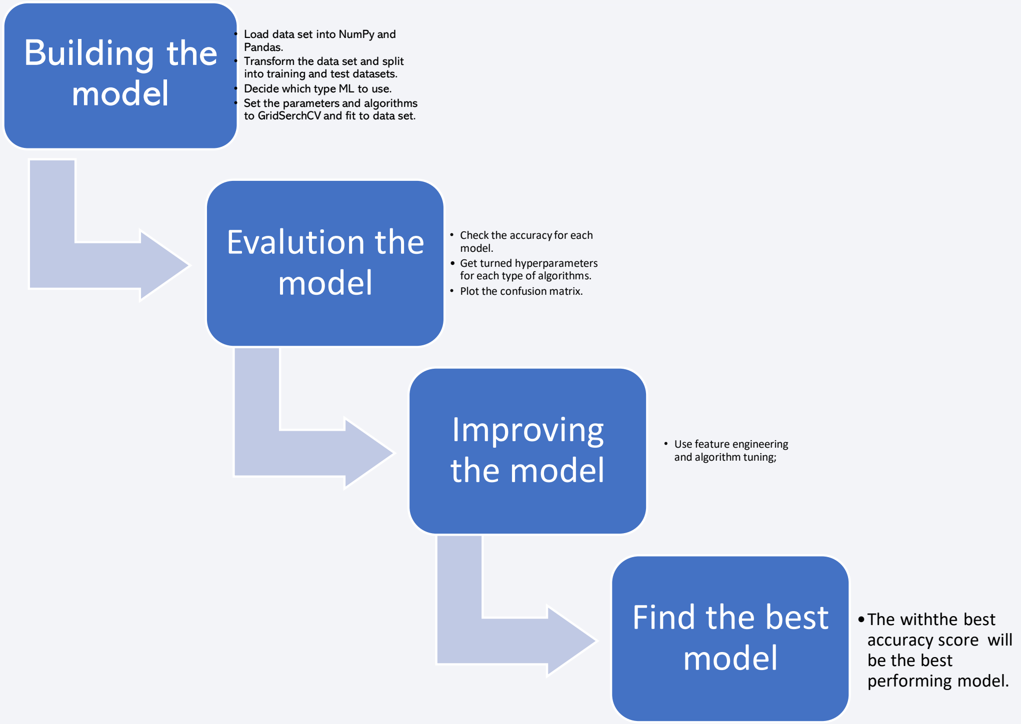
Build a Dashboard with Plotly Dash

We built an interactive dashboard with Plotly dash which allowing the user to play around with the data as they need.

- We plotted pie charts showing the total launches by a certain sites.
- We then plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version.

The link of the app.py:: https://github.com/Jones021/mj/blob/main/App.py

Predictive Analysis (Classification)



- https://github.com/Jones021/mj/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

The results will be categorized to 3 main results which is:

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

Section 2

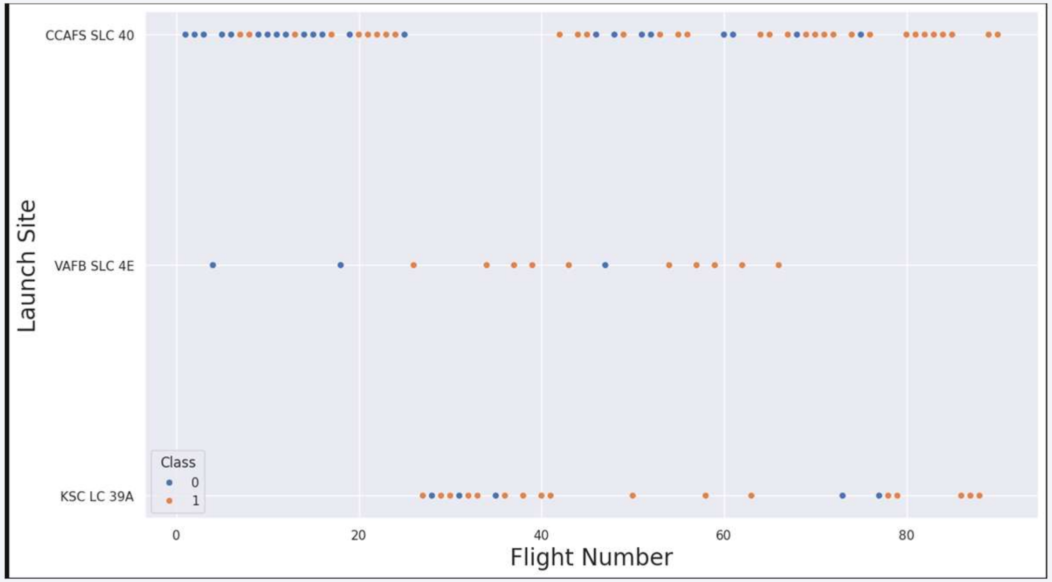
Insights drawn from EDA

Flight Number vs. Launch Site

This scatter plot shows that the larger the flights amount of the launch site, the greater the success rate will be.

However, site CCAFSSLC40 shows the least pattern of this.

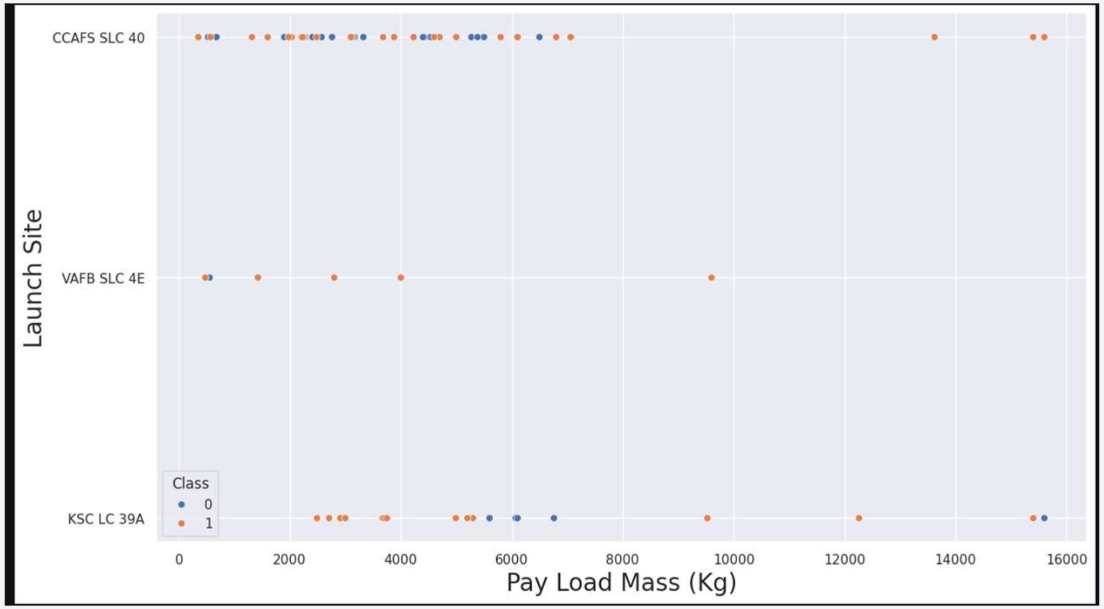
https://github.com/Jones021/mj/blob/main/Exploringand%20preparingData-eda-dataviz.ipynb.jupyterlite.ipynb



Payload vs. Launch Site

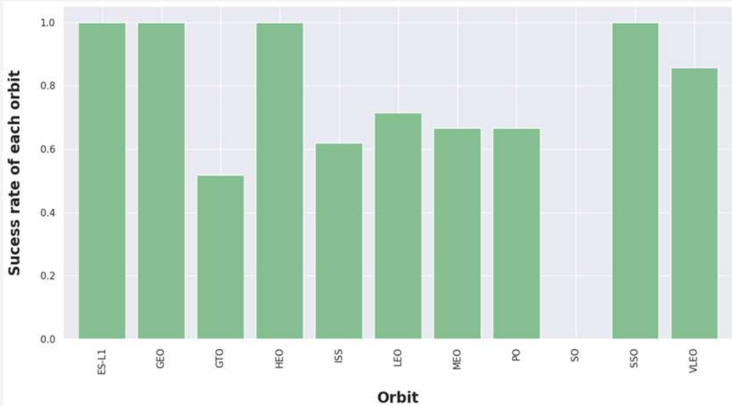
This scatter plot shows once the pay load mass is greater than 7000 Kg, the probability of success rate will be heightly increased.

However there is no clear pattern to say the launch site is depemdemt to the pay load mass for the success rate.



Success Rate vs. Orbit Type

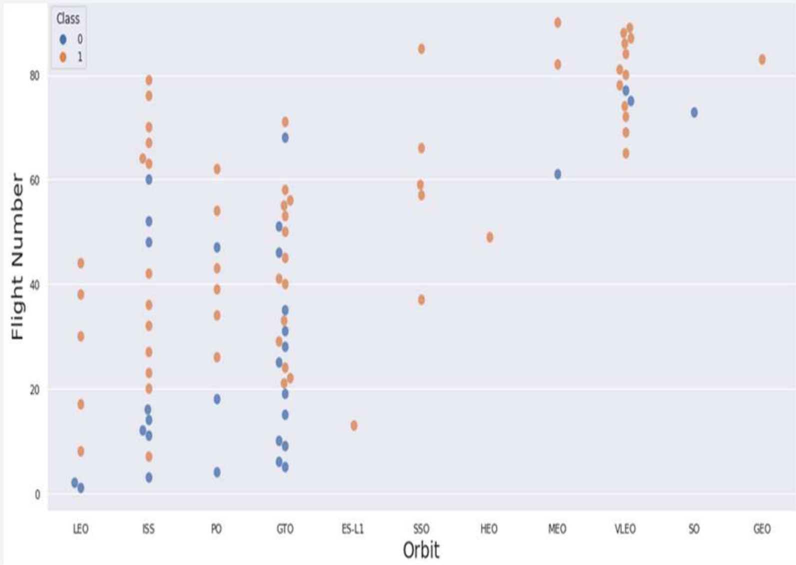
- Show a bar chart for the success rate of each orbit type



Flight Number vs. Orbit Type

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

- Show the screenshot of the scatter plot with explanations

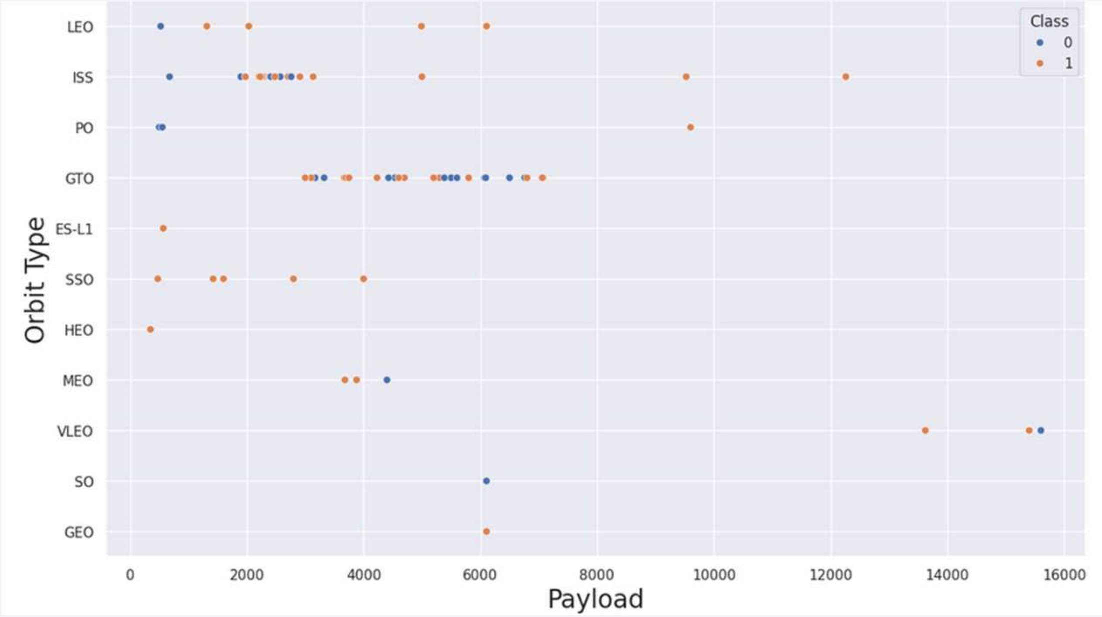


Payload vs. Orbit Type

Heavier payload impact on ÇEO,ISS and PO orbit. However , it has negative impact on MEO and VLEQ .

GTO orbit seem to depect no relation between the attributes.

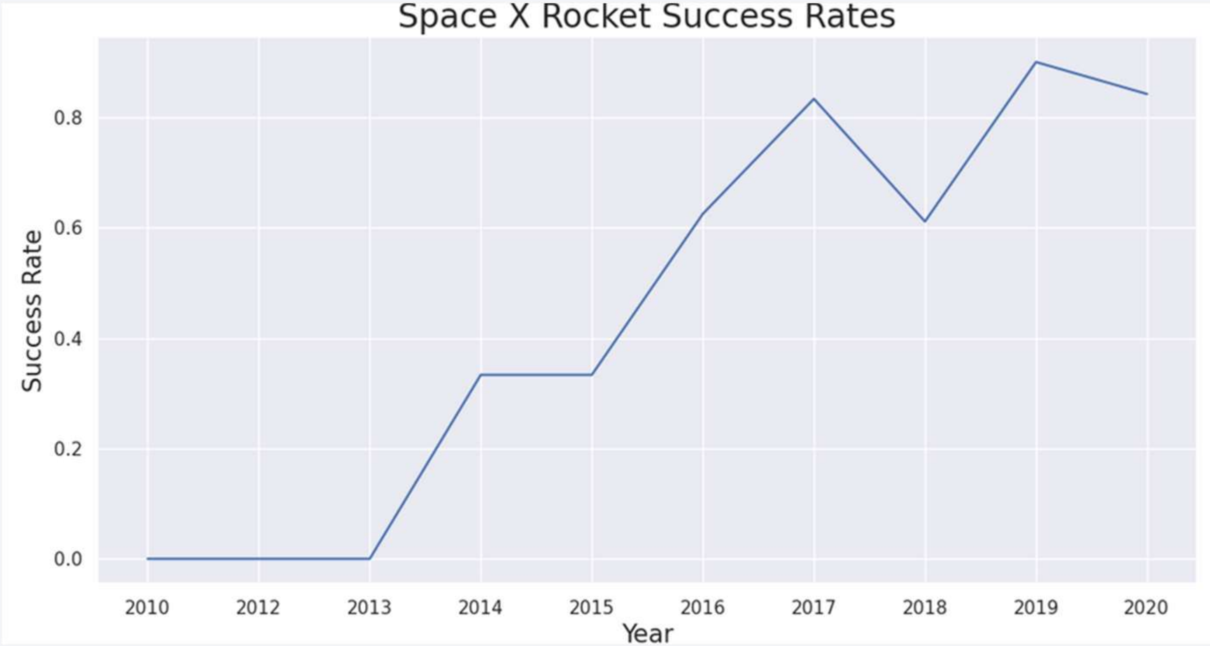
Meanwhile, again,So,GEO abd HEO orbit nedd moore data set to see any pattern or trend.



Launch Success Yearly Trend

The figure cleary depicted and increasing trend from the year 2013 until 2020.

If this trend continue for the next year onward. The success rate will steadily increase until reaching 1/100% success rate.



All Launch Site Names

We use the key word DISTNCT to show only unique launch sites from the SpaceX data

In [5]:
```
%sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEX;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Out[5]:

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

Launch Site Names Begin with 'CCA'

We used the query above to display 5 records where launch sites begin with `CCA`

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

In [11]:
```
task_2 = '''
        SELECT *
        FROM SpaceX
        WHERE LaunchSite LIKE 'CCA%'
        LIMIT 5
        '''
create_pandas_df(task_2, database=conn)
```

Out[11]:

	date	time	boosterversion	launchsite	payload	payloadmasskg	orbit	customer	missionoutcome	landingoutcome
0	2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	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
3	2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

We calculated the total payload carried by boosters from NASA

as 45596 using the query below

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS "Total Payload Mass by NASA (CRS)
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3
sd0tgtu01qde00.databases.appdomain.cloud:32731/bludb
Done.
```

Total Payload Mass by NASA (CRS)
45596

Average Payload Mass by F9 v1.1

We calculated the average payload mass carried by booster version

F9 v1.1 as 2928.4

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

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS "Average Payload Mass by Booster
WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3
sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Average Payload Mass by Booster Version F9 v1.1
2928

First Successful Ground Landing Date

We use the min() function to find the result

We observed that the dates of the first successful landing outcome on ground

pad was 22nd December 2015

```
%sql SELECT MIN(DATE) AS "First Succesful Landing Outcome in Ground Pac
WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3
sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

First Succesful Landing Outcome in Ground Pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

We used the WHERE clause to filter for boosters which have successfully landed on drone ship and applied the AND condition to determine successful landing with payload mass greater than 4000 but less than 6000

```
%sql SELECT BOOSTER_VERSION FROM SPACEX WHERE LANDING__OUTCOME = 'Success (drone ship)' \
AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.datab
ases.appdomain.cloud:32731/bludb
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

We used wildcard like '%' to filter for WHERE MissionOutcome was a success or a failure.

List the total number of successful and failure mission outcomes

```
%sql SELECT COUNT(MISSION_OUTCOME) AS "Successful Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Success%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Successful Mission
100

```
%sql SELECT COUNT(MISSION_OUTCOME) AS "Failure Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Failure%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.clogj3sd0tgtu0lqde00.databases.appdomain.clou
d:32731/bludb
Done.
```

Failure Mission
1

Boosters Carried Maximum Payload

```
%sql SELECT DISTINCT BOOSTER_VERSION AS "Booster Versions which carried the Maximum Payload Mass" FROM SPACEX
WHERE PAYLOAD_MASS__KG_ =(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEX);
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.clogj3sd0tgtu0lqde00.databases.appdomain.clou
d:32731/bludb
Done.
```

Booster Versions which carried the Maximum Payload Mass
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

We determined the booster that have carried the maximum payload using a subquery in the WHERE clause and the MAX() function.

2015 Launch Records

We used a combinations of the WHERE clause, LIKE, AND, and BETWEEN conditions to filter for failed landing outcomes in drone ship, their booster versions, and launch site names for year 2015

```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEX WHERE DATE LIKE '2015-%' AND \
LANDING__OUTCOME = 'Failure (drone ship)';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.clogj3sd0tgtu0lqde00.
databases.appdomain.cloud:32731/bludb
Done.
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

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

```
%sql SELECT LANDING__OUTCOME as "Landing Outcome", COUNT(LANDING__OUTCOME) AS "Total Count" FROM SPACEX \
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
GROUP BY  LANDING__OUTCOME \
ORDER BY COUNT(LANDING__OUTCOME) DESC ;
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.c
loud:32731/bludb
Done.
```

Landing Outcome	Total Count
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

We selected Landing outcomes and the COUNT of landing

outcomes from the data and used the WHERE clause to

filter for landing outcomes BETWEEN 2010-06-04 to

2010-03-20. We applied the GROUP BY

clause to group the landing outcomes and the ORDER BY

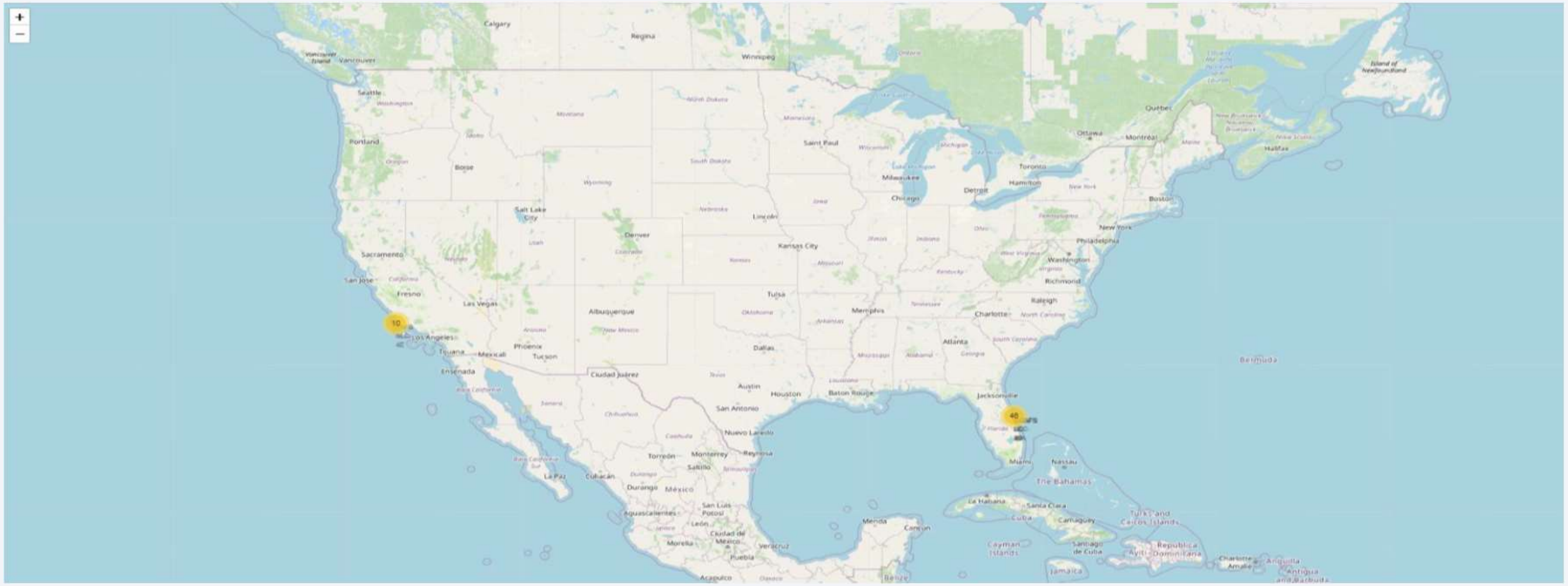
clause to order the grouped landing outcome in descending order.

Section 3

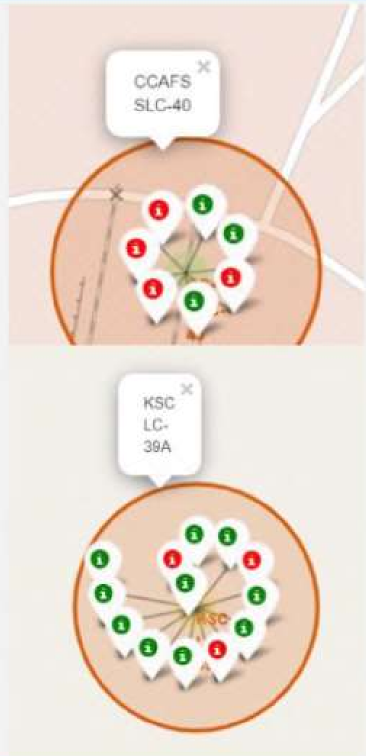
Launch Sites Proximities Analysis

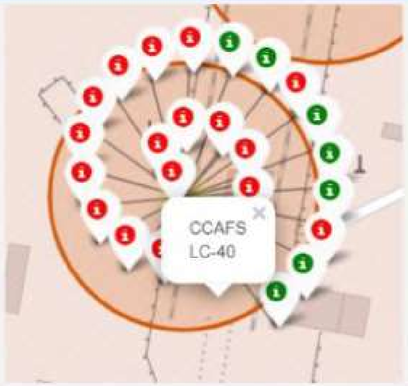
Launch Sites

We can see that all the SpaceX launch sites are located inside the United States



Markers showing launch sites with color labels





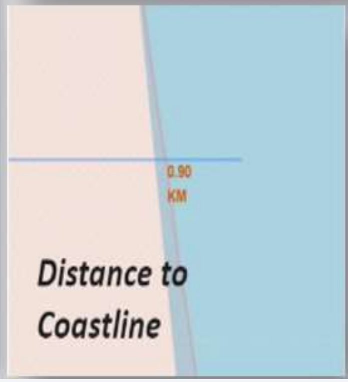
Florida Launch Sites

Green Marker shows successful Launches and Red Marker shows Failures

California Launch Site

37

Launch Sites Distance to Landmarks

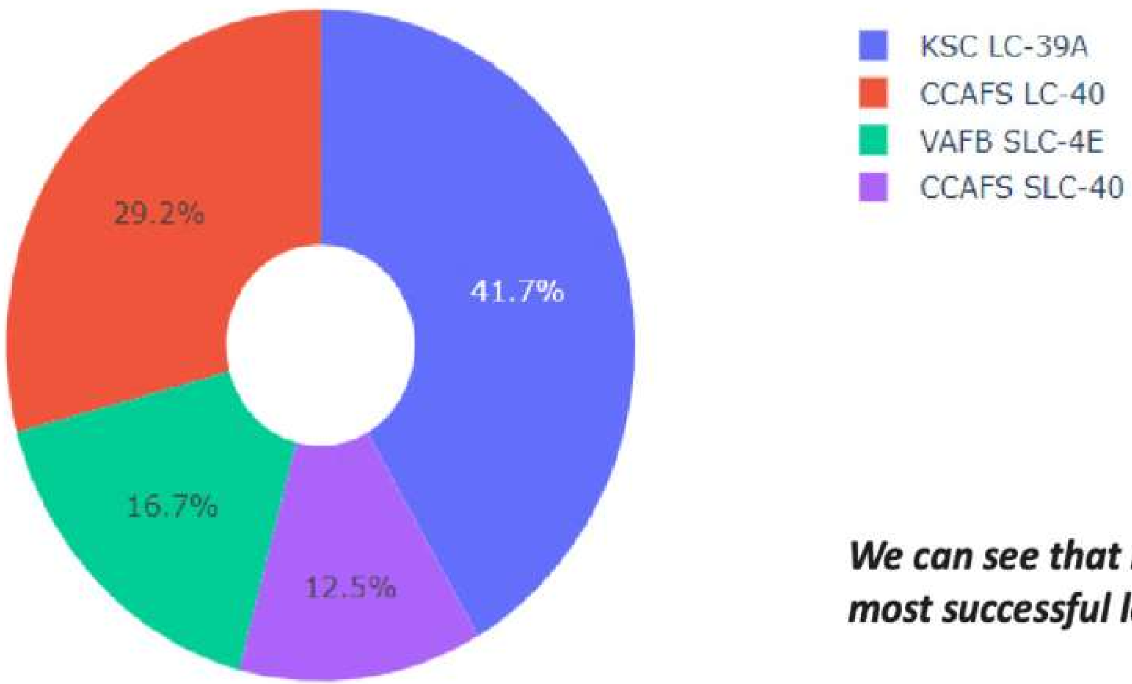


- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

Section 4

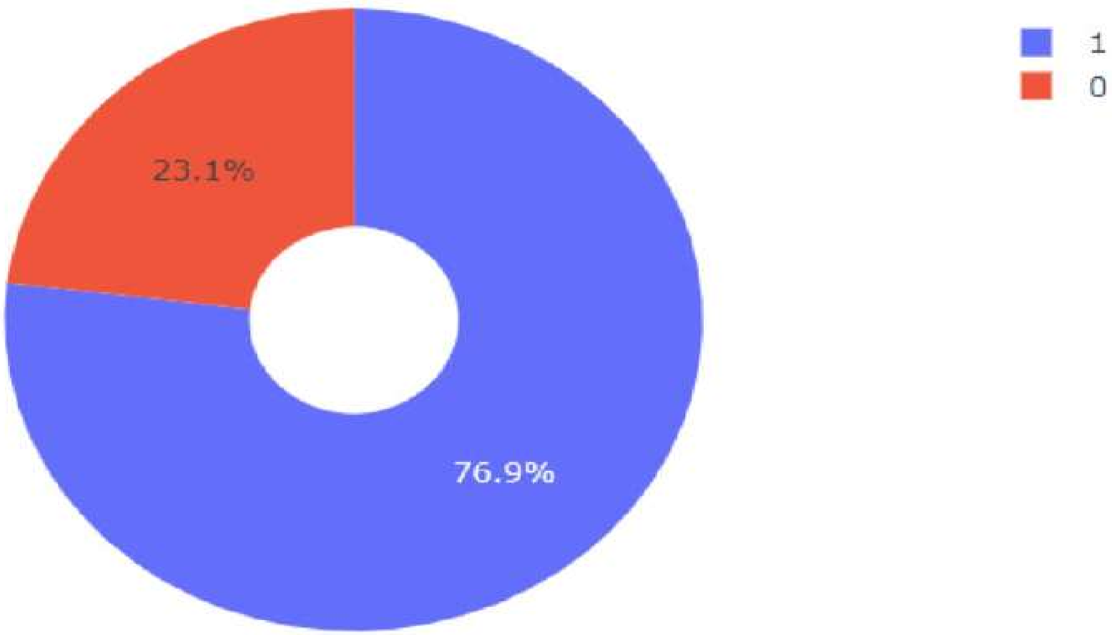
Build a Dashboard with Plotly Dash

The success percentage by each sites.



We can see that KSC LC-39A had the most successful launches from all the sites

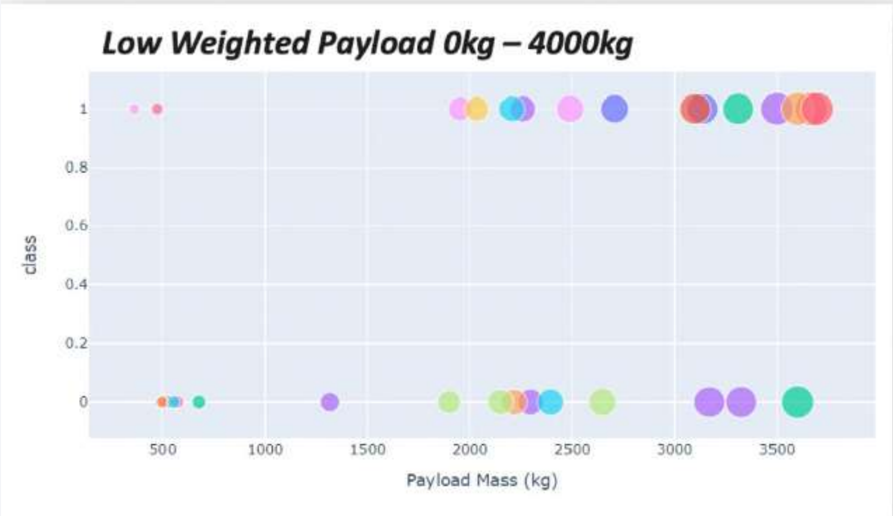
The highest launch-success ratio: KSC LC-39A

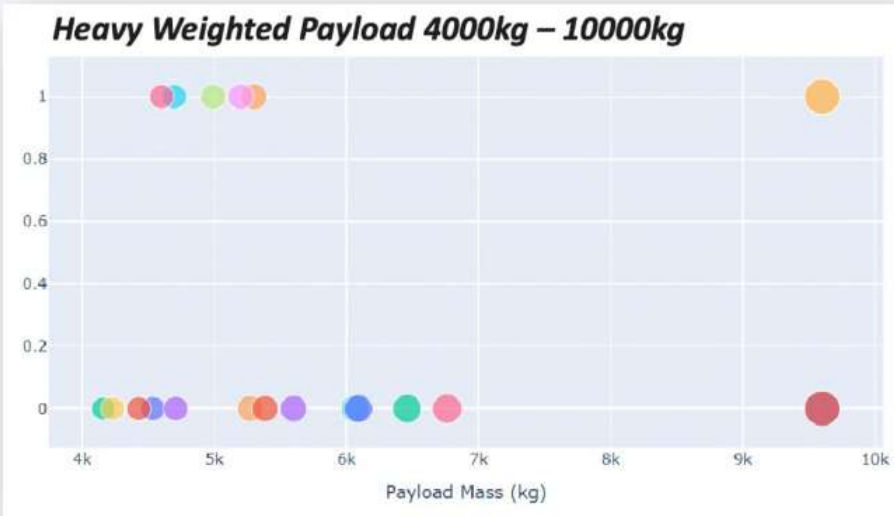


KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Payload vs Launch Outcome Scatter Plot

We can see that all the success rate for low weighted payload is higher than heavy weighted payload





Section 5

Predictive Analysis (Classification)

Classification Accuracy

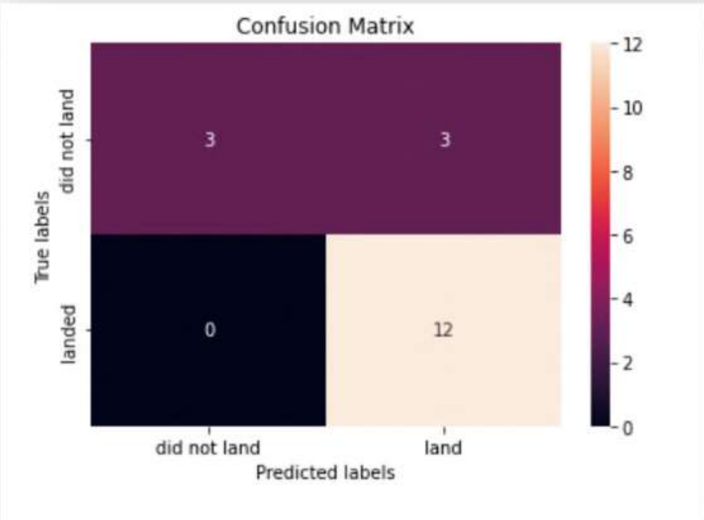
As we can see, by using the code as below: we could identify that the best algorithm to be the Tree Algorithm which have the highest classification accuracy

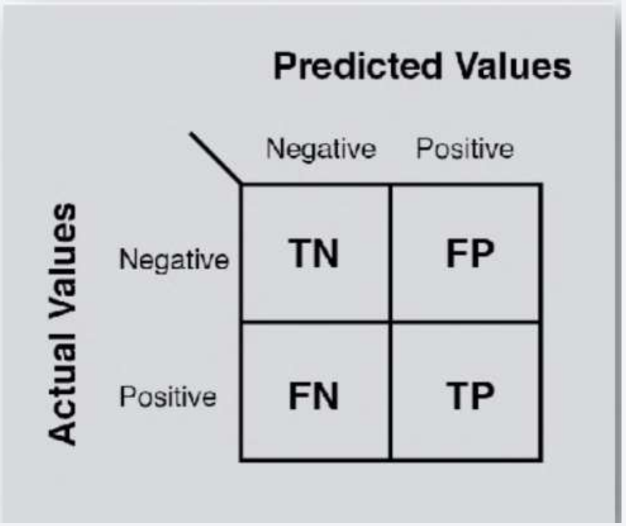
```
algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_}
bestalgorithm = max(algorithms, key=algorithms.get)
print('Best Algorithm is',bestalgorithm,'with a score of',algorithms[bestalgorithm])
if bestalgorithm == 'Tree':
    print('Best Params is :',tree_cv.best_params_)
if bestalgorithm == 'KNN':
    print('Best Params is :',knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best Params is :',logreg_cv.best_params_)
```

```
Best Algorithm is Tree with a score of 0.9017857142857142
Best Params is : {'criterion': 'entropy', 'max_depth': 10, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_sampl
es_split': 10, 'splitter': 'random'}
```

Confusion Matrix

The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.





Conclusions

We can conclude that:

• The Tree Classifier Algorithm is the best Machine Learning approach for this dataset.

• The low weighted payloads (which define as 4000kg and below) performed better than the heavy weighted payloads.

• Starting from the year 2013, the success rate for SpaceX launches is increased, directly proportional time in years to 2020, which it will eventually perfect the launches in the future.

• KSC LC-39A have the most successful launches of any sites; 76.9%

• SSO orbit have the most success rate; 100% and more than 1 occurrence.

Appendix

https://github.com/Jones021/mj/tree/main


Thank you!