IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Flynn Fayman
09/13/2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data was Collected and cleaned. Multiple data visualization tools were used to see what Features Collared with each other. Different Learning models were Train and tested. The Model which performed the best in accuracy were chosen.
- Summary of all results
  - The Decision Tree Preformed Highest in accuracy and was chosen as model for successful re-landing. Payload Mass, Orbit type, and Flight number all collared with Class/re-landing success.

Introduction

- Project background and context

The market for private space expeditions is growing and to pull ahead of the pack one must lower their cost per flight. A way of lowering one’s Total flight cost is to reuse parts of the spaceship such as the first stage tank. Companies like SpaceX are reusing First stage tanks by re-landing it, after launch, at different bases enabling them save a significant amount of money.

- Problems you want to find answers

Space Y(the company that I’m working for) would also Like to land its first stage tank for reuse and by using data from space x we are able to determine what affects a Landing's success rate. In this example we examine the Falcon 9 spaceship exclusively.

Section 1

Methodology

Methodology

Executive Summary

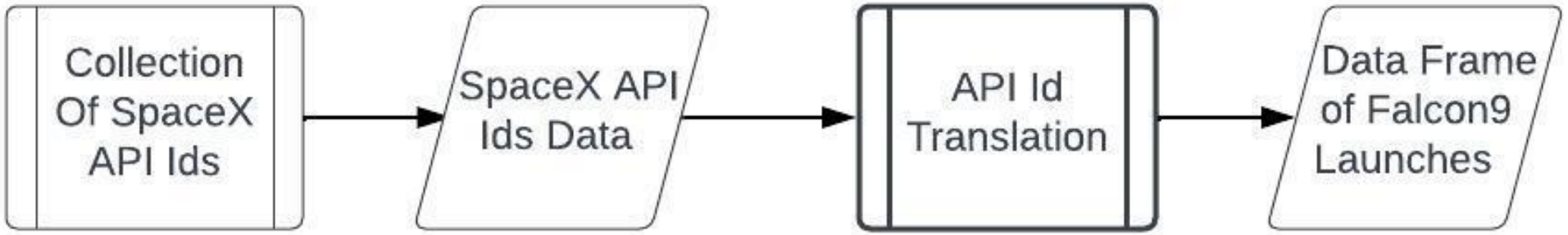
- Data collection methodology:
  - Data was Collected using APIs and wiki pages.
- Perform data wrangling
  - A Column was added called Class
- Perform interactive visual analytics using Folium and Plotly Dash
  - Folium help gain insights into how geography effected re-landing Success. Plotly Dash showed collection between Payload Mass, launch Site and Re-landing Success
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Matplotlib and SQL lite was used to explore the data and discover insights that would be helpful
- Perform predictive analysis using classification models
  - Data is Split into Features and Solution(X and Y) and Data is Further Split into test and Training Data. The Model witch most accurate is chosen

Data Collection

- Describe how data sets were collected.
  - Datasets were Collected using Webcasting and Public APIs

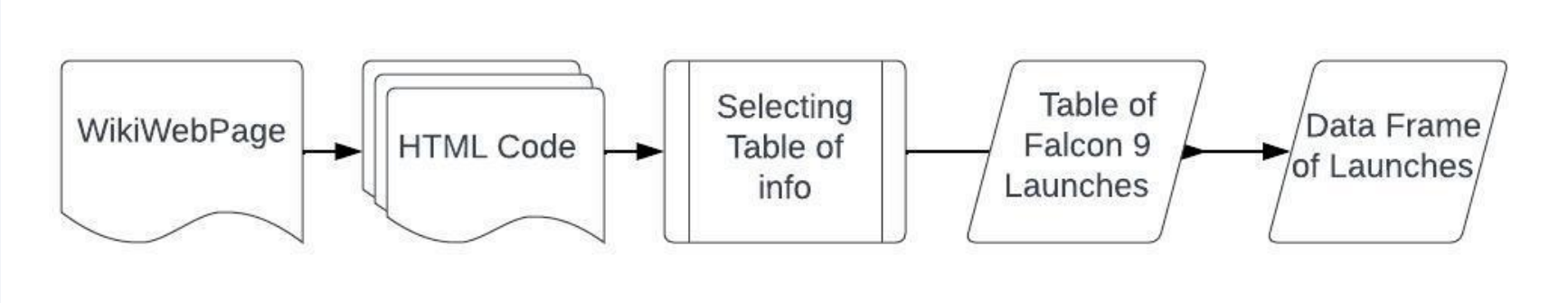
Data Collection – SpaceX API

- For The API method we Requested Data from SpaceX API. From that data we created a data frame of the requested Data. However, the requested data was in the form of Ids requiring another SpaceX Api to translate the ids into readable values for there respected features. After requesting the SpaceX Api to translate ids into values, launches that were not Falcon 9 where remove from the created data set.



Data Collection - Scraping

- Websracping involves the collection of Data through public sites like Wikipedia. We used the Wiki page, List of Falcon 9 and Falcon Heavy launches, to collect data about Falcon 9 ship launches. We collect the raw html code from the Wiki page then select the table sections of the Html code. After collecting the table code/information, we select launch tables that contain Falcon 9 Launches and turn the table code into a Data frame of Falcon 9 launches.



Data Wrangling

- In the further processing of the collected data a Columns was added. The added Column was Labeled as Class and was used to describe whether the re-landing of the First Stage of Booster was Successful. Values for Class where 1 for Successes and 0 Failure.

EDA with Data Visualization

- Bar Charts
  - Show relationship between a Categorical Variables
    - Orbit Vs Average Class. Orbit is Categorical While Average Class is Continuous. Allows us to view how Orbit effects Class
- Scatter plots
  - Show a relationship between Variables that are Categorical or Continuous.
    - Flight Number vs Payload Mass (kg) and color for points is Blue for Class 0 and Orange for Class 1. Flight Number and Payload both Continuous
    - Flight Number Vs Launch Site same Color for class. Flight Number is Continuous and Launch Site Categorial
    - Flight Number Vs Payload Mass (kg) same Color for Class. Payload Mass is Continuous for each Launch Site we can see cutoff points for mass where Class is more likely to be 1.
    - Flight Number Vs Orbit with color as Class. Orbit Categorical variable.
    - Orbit vs Payload mass color as Class
- Line plots
  - The same as Scatter plots but there is a line connecting the points
    - Class vs Date. We can see that as Date increase Average Class Increases

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
  - Queries
    - Queried Values that where Distinct in categorical columns
    - Found Max, Min, and Average for Continuous Columns
    - Found Booster Version Values that had a Payload Mass greater than 40000 and less than 6000 with a Mission Outcome of Success
    - Found the Counts of Mission Success and Mission Failure for all examples
    - Listed information based on Landing outcome and Date
    - Group by Landing Outcome and Rank query by Count of Landing Outcome

- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
  - Each Launch Site was Marked and Given A Label
  - For Each Launch Site a Group Markers are made each Marker is labeled Red or Green for Class 0 and 1 Respectively
  - The Distances Between Each Launch Site and the nearest Railroad Track, Shoreline, City and Freeway is marked with a Blue Line and a label with the distance.
- Explain why you added those objects
  - The reason for all these Makings Is to try to discover any Relationships between the Geographic location of Launch Site and Class.

Build a Dashboard with Plotly Dash Graph 1

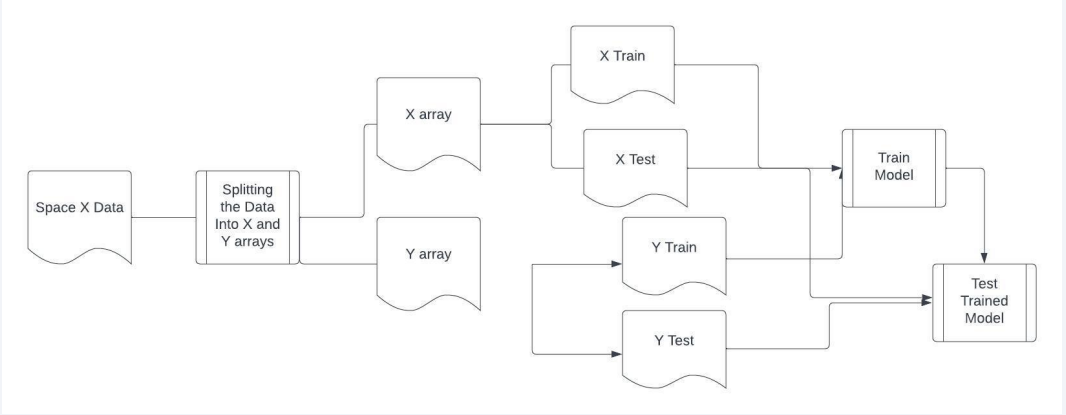
- There are two interactive graphs in total.
  - First Graph is a Pie chart where the individual can select All Launch Sites or select a pacific Launch Site.
    - If All Launch Sites are Selected a Pie Chart is created where values are values of Class and names are of each Launch. Successes of each Site Ratio
    - If an Individual Launch Site is Selected, then Count of 1 and 0 are values of the pie Chart and names are 1 and 0. Successes Ratio of selected Site

Build a Dashboard with Plotly Dash Graph 2

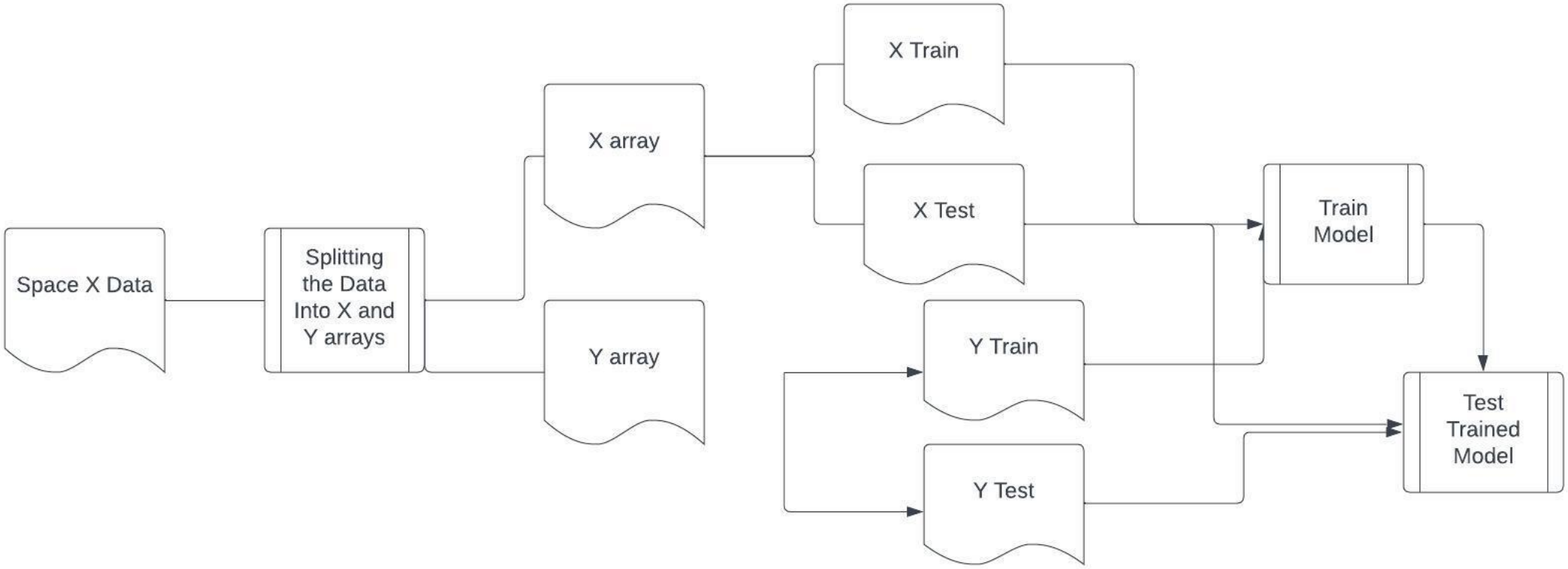
- Second Graph is created based both Launch Site that Selected and the max a min for Payload Mass.
  - If the Individual Selects All Sites a Scatter Plot is returned with its x variable as Payload Mass(kg) of the span selected and y as Class with color as Launch Site
  - If the Individual Selects a pacific Launch Site, Then Scatter plot has x as Payload Mass Span and Y as Class with no Color
- Explain why you added those plots and interactions
  - The reason for the Pie Chart is so that individual can compare each Launch site Success ratio with other launch sites. Or examine the Launch sites Success ratio
  - The Scatter Plot allows the individual to see the how the Payload mass effects Class for different Launch sites.

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
  - Frist the data is split into two different arrays. Array one or X are all the all columns in the data set expect for the Class Column. Second Array or Y is Class column from the data Set
  - The X array is then Standardized
  - Both the X and Y arrays are split into training sets and testing Sets to train the model and test out of set accuracy.
  - A Series of learning Models were Train using the training sets and an accuracy Score was taking using test set for each model. The model which scored the highest for acracy was chosen.

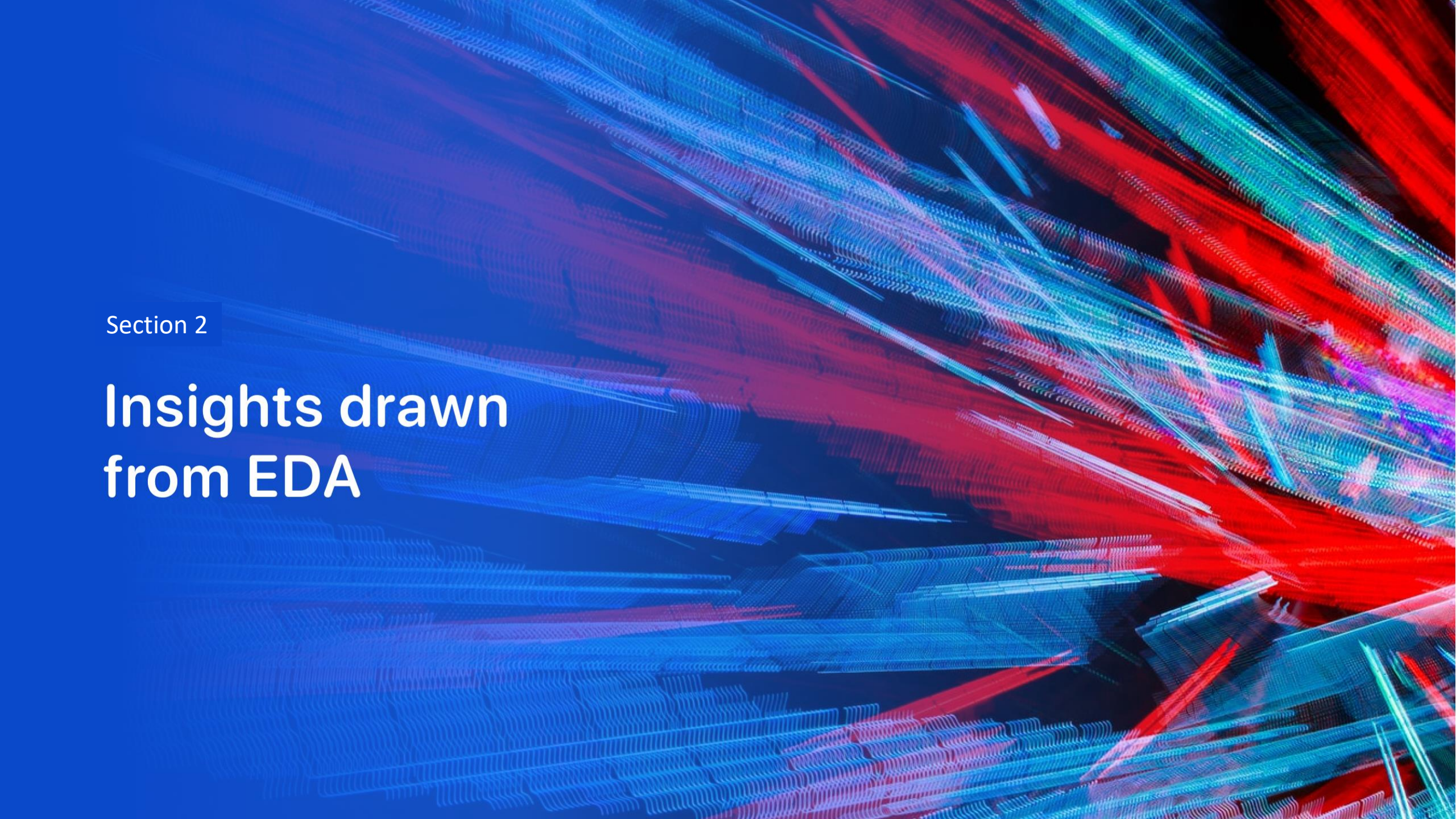


Predictive Analysis (Classification) Flow Chart



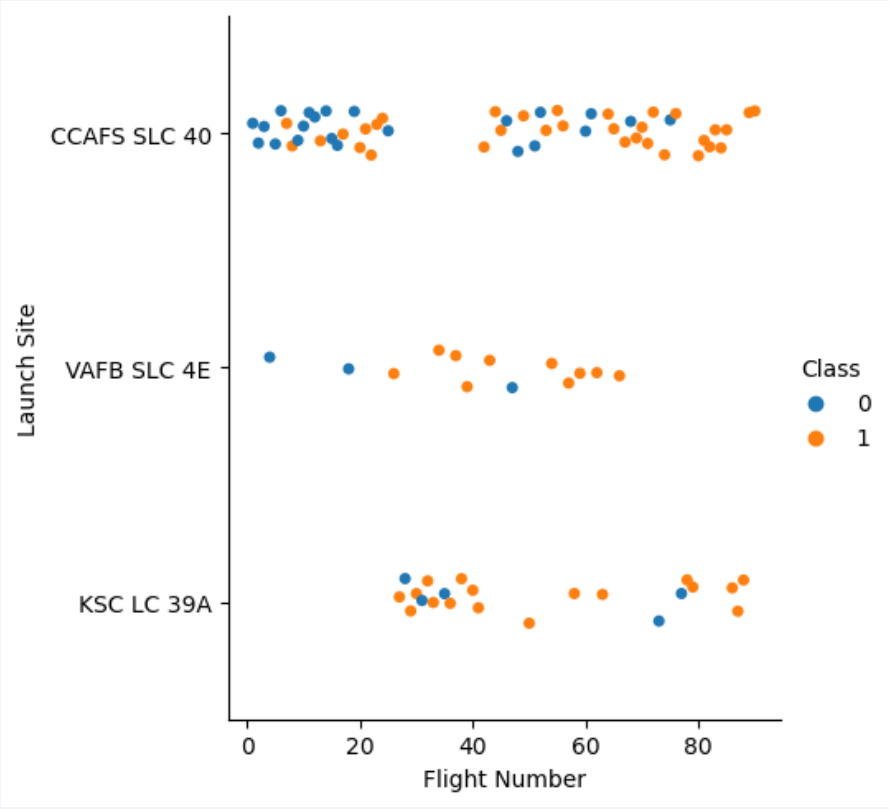
Results

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


Section 2
Insights drawn from EDA

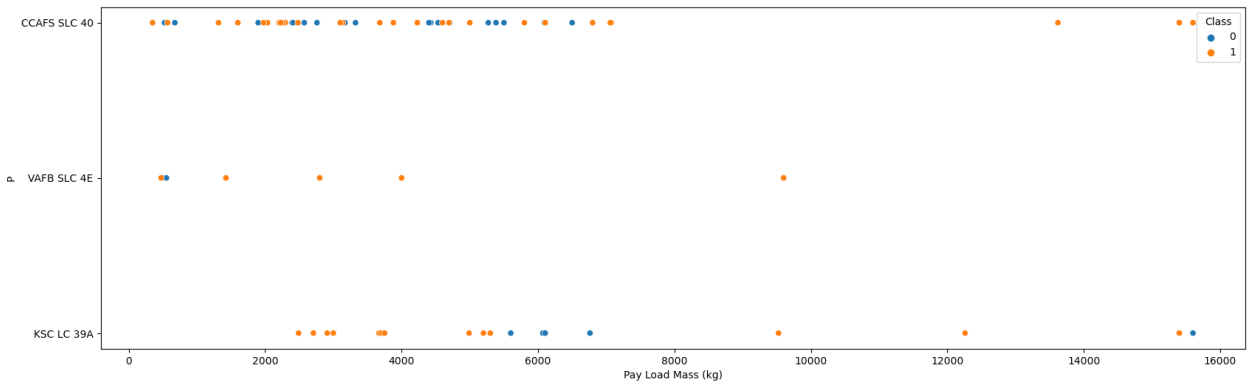
Flight Number vs. Launch Site

- The scatter plot shows a relationship between Flight Number and Class. As Flight number increases for each Launch site the probability of Class being one increases.



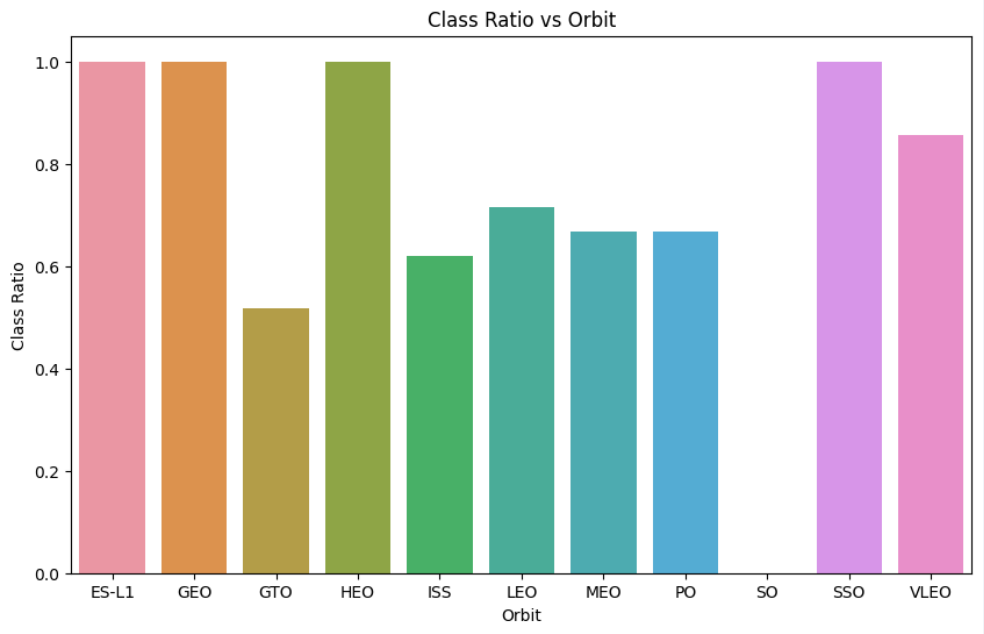
Payload vs. Launch Site

- In the Payload Vs Launch Site scatter plot it is apparent that each Launch Site has a certain range of Pay Load Mass where the Class is more Likely to be 1 or more likely to be 0



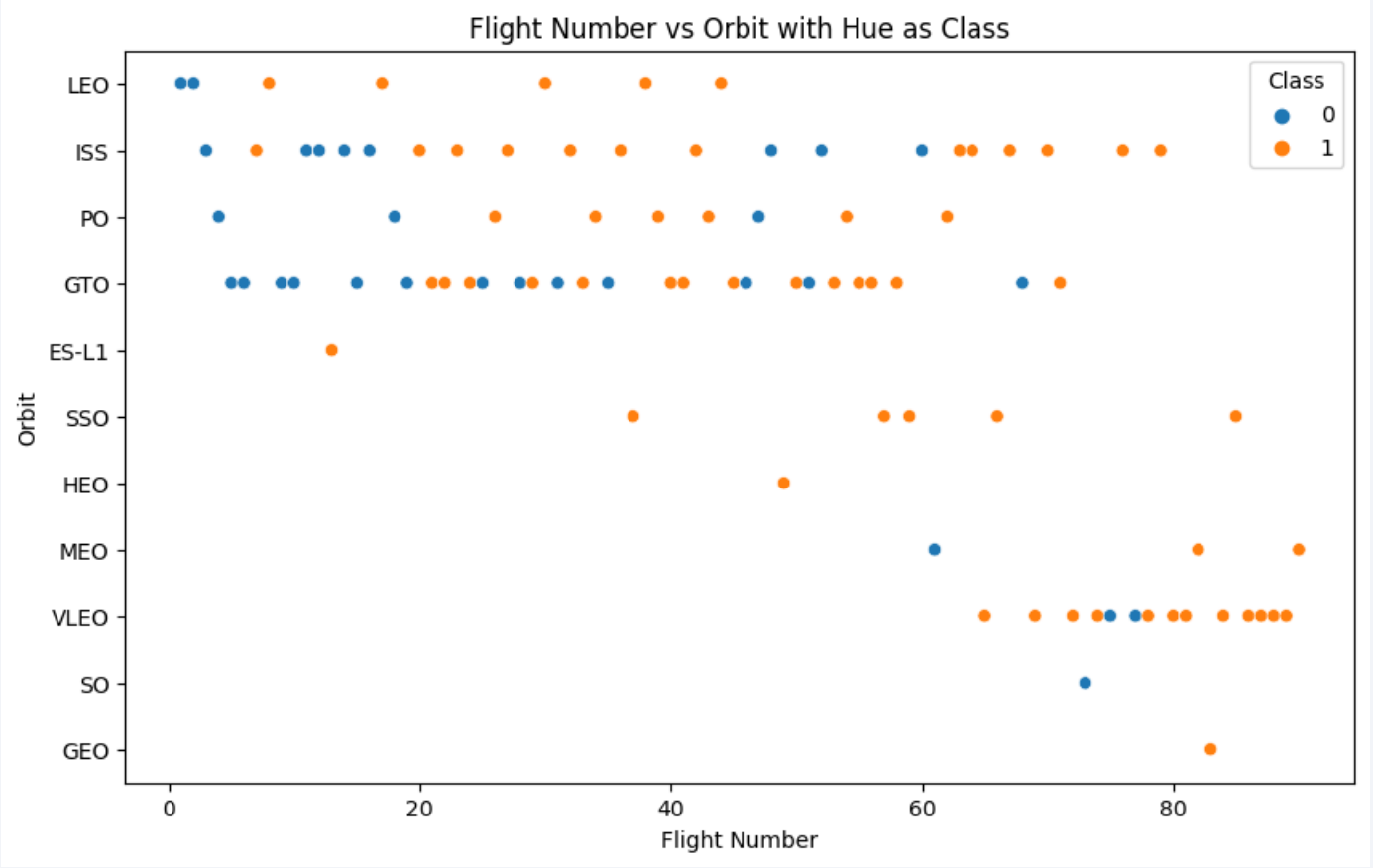
Success Rate vs. Orbit Type

- With the Class Ratio Vs Orbit the observer can see that Certain Orbits have a Higher average Class. For instances Es-L,GEO,HEO and SSO orbits have an average Class of 1 while Orbits Like ISS,GTI and SO have average Class closer to .4 or even 0



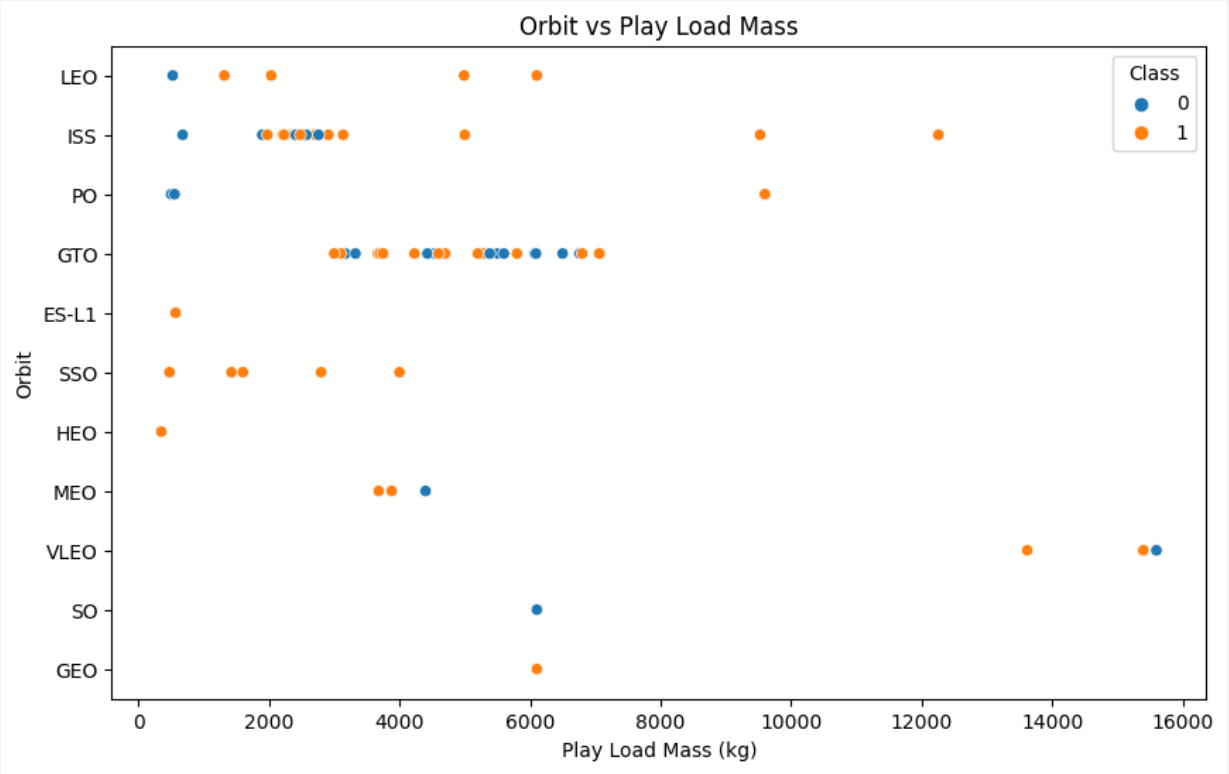
Flight Number vs. Orbit Type

- The Flight Number Vs Orbit Type Scatter plot shows the observer that for each Orbit there exist a different range for Flight Number where the Class is more likely to 1 then 0



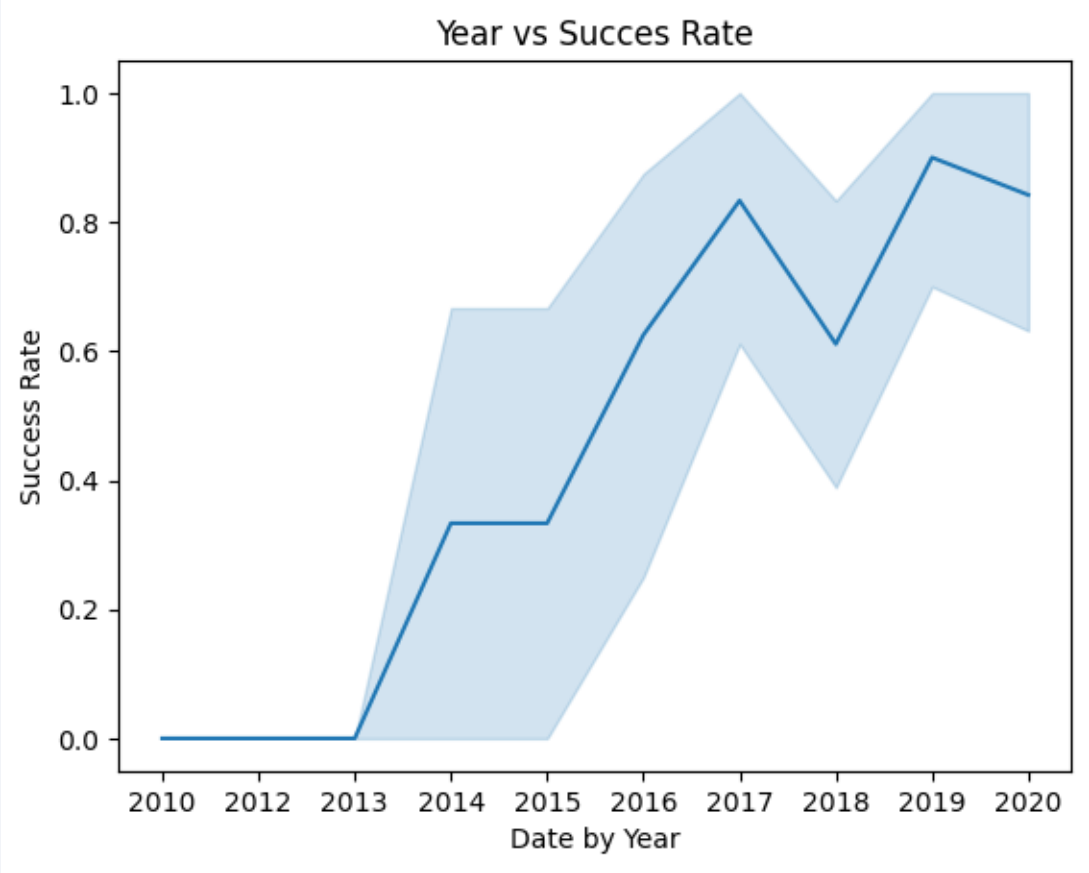
Payload vs. Orbit Type

- The Payload Vs Orbit Type Scatter plot shows the observer that for each Orbit there exist a different range for Payload where the Class is more likely to 1 then 0



Launch Success Yearly Trend

- Launch Success Yearly Trend line plot shows a positive correlation between Year and Success Rate



All Launch Site Names

- The query shows the viewer the distinct Launch Site values/names for are data

Task 1

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`
- The query below gives the 5 rows where the column Launch Site name begins with CCA

Task 2

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

```
[1]: %sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5
```

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

[1]:

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

Total Payload Mass

- The query below shows the sum of the mass of all Launches Payload

Task 3

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

```
3]: %sql Select SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer Like '%NASA%CRS%
 * sqlite:///my_data1.db
Done.
```

3]:

SUM(PAYLOAD_MASS__KG_)
48213

Average Payload Mass by F9 v1.1

- The query below shows the average payload mass carried by booster version F9 v1.1. The average payload mass carried by booster version is 2534.67(kg)

Task 4

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

- The query below show the dates of the first successful landing outcome on ground pad. The dates are 12/22/2015

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

Hint:Use min function

```
[13]: %sql Select min(Date) from SPACEXTABLE where Landing_Outcome Like '%Success%ground pad%'
```

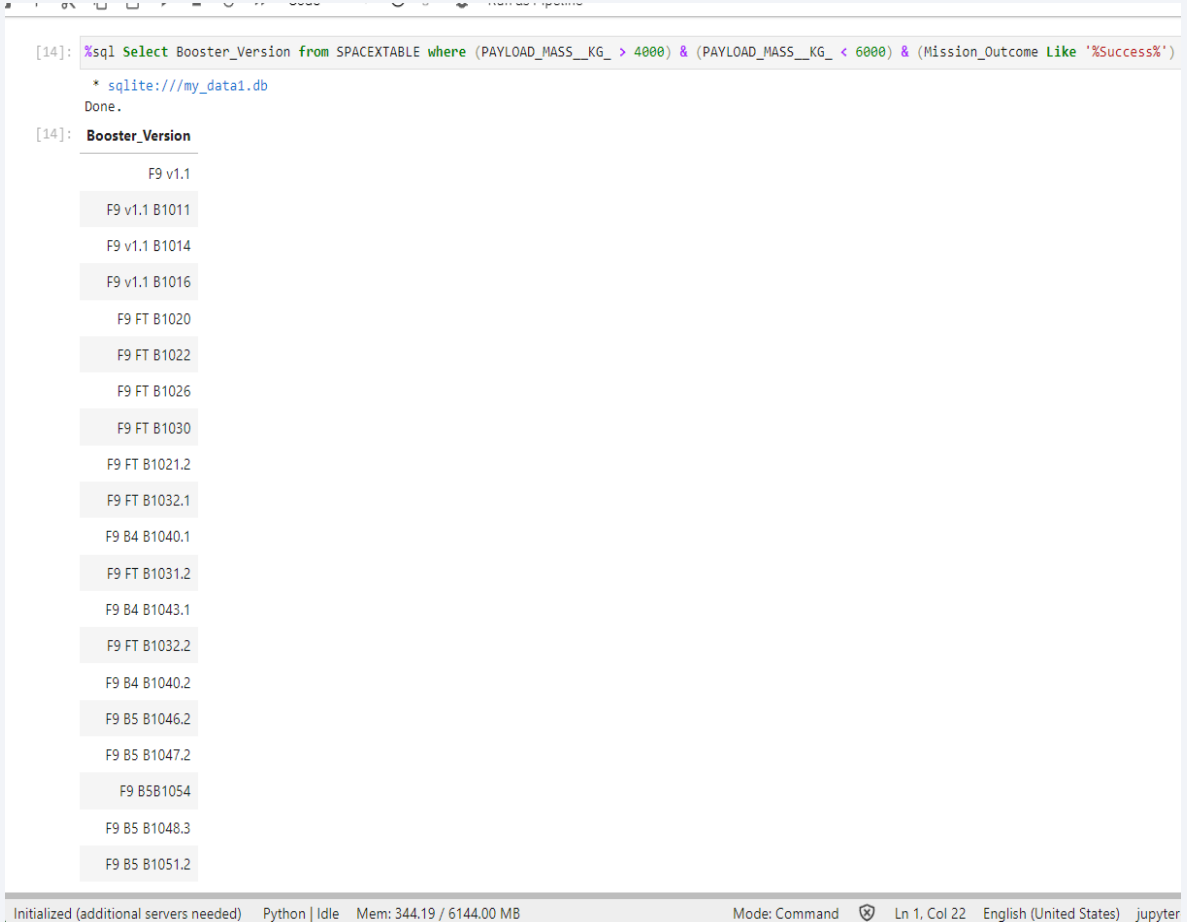
```
 * sqlite:///my_data1.db
Done.
```

[13]:

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- The query the right shows the list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000.



```
[14]: %sql Select Booster_Version from SPACEXTABLE where (PAYLOAD_MASS__KG_ > 4000) & (PAYLOAD_MASS__KG_ < 6000) & (Mission_Outcome Like '%Success%')
 * sqlite:///my_data1.db
Done.
```

[14]:

Booster_Version
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1014
F9 v1.1 B1016
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1030
F9 FT B1021.2
F9 FT B1032.1
F9 B4 B1040.1
F9 FT B1031.2
F9 B4 B1043.1
F9 FT B1032.2
F9 B4 B1040.2
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5B1054
F9 B5 B1048.3
F9 B5 B1051.2

Total Number of Successful and Failure Mission Outcomes

- The query below calculates the total number of successful and failure mission outcomes. There are 100 Successful missions and 1 Mission Failure Sum

Task 7

List the total number of successful and failure mission outcomes

[15]:
```
%%sql Select distinct (Select Count(Mission_Outcome) from SPACEXTABLE Where Mission_Outcome Like '%Success%') AS 'Successful missions Sum' ,
(Select Count(Mission_Outcome) from SPACEXTABLE Where Mission_Outcome Like '%Failure%') as 'Mission Failure Sum'
from SPACEXTABLE
```

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

[15]:

Successful missions Sum	Mission Failure Sum
100	1

Boosters Carried Maximum Payload

- The query below the 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

```
6]: %sql Select distinct Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ in (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

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

6]:

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

- The query below shows List the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015. There are two rows that match the conditions set upon the query

```
%sql select substr(Date,6,2) as Month,Landing_Outcome,Booster_Version,Launch_Site from SPACEXTABLE where (Landing_Outcome like '%ailure%drone ship%') & (substr(Date,0,5) =
```

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

Month	Landing_Outcome	Booster_Version	Launch_Site
10	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

- The query to the right shows the 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 landing outcomes (such as Failure (drone ship) or Success (ground

```
%%sql select Landing_Outcome, Count(Landing_Outcome) from SPACEXTABLE
where (Date between '2010-06-04' and '2017-03-20')
group by Landing_Outcome
order by Count(Landing_Outcome) DESC
```

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

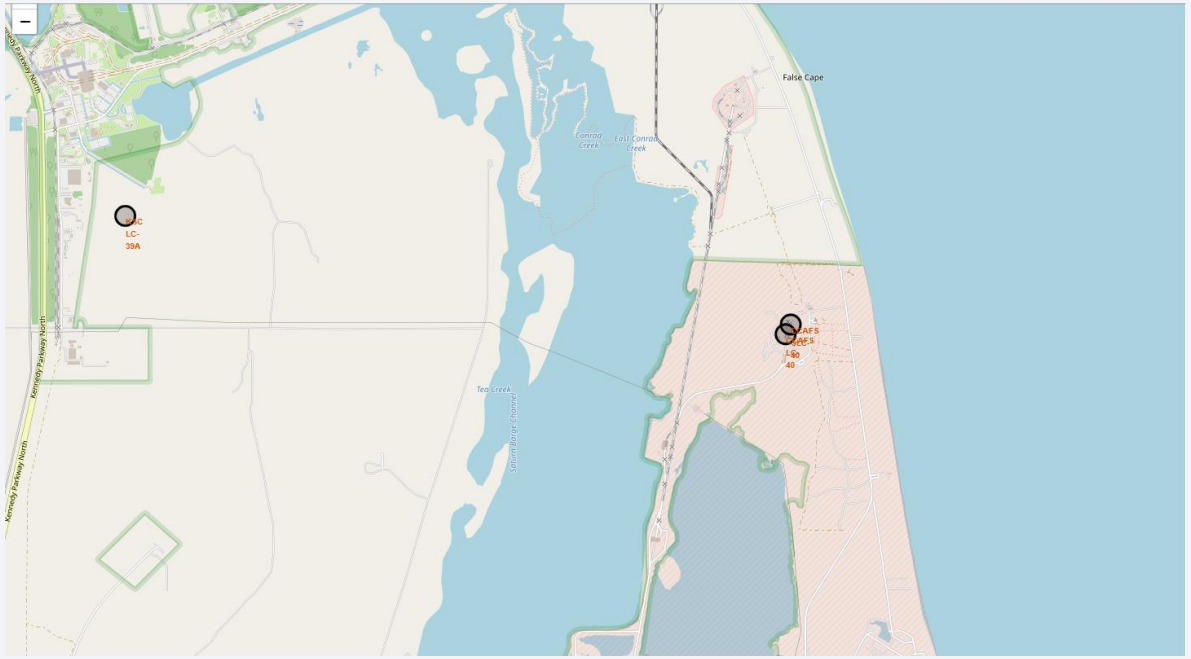
Landing_Outcome	Count(Landing_Outcome)
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

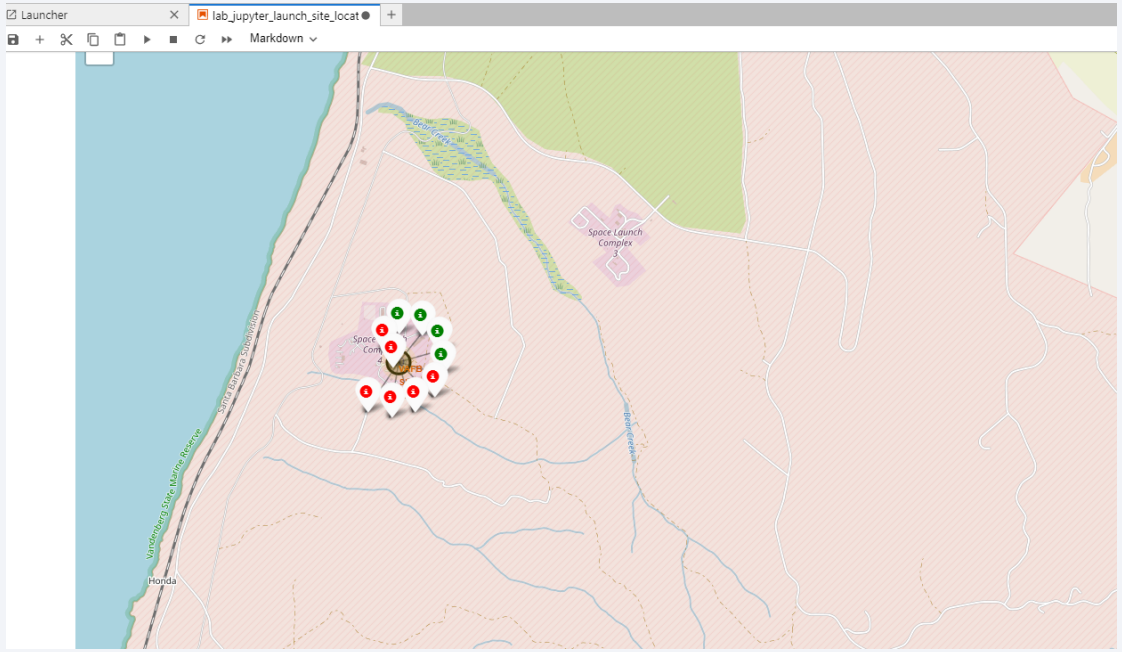
Map of Launch Locations

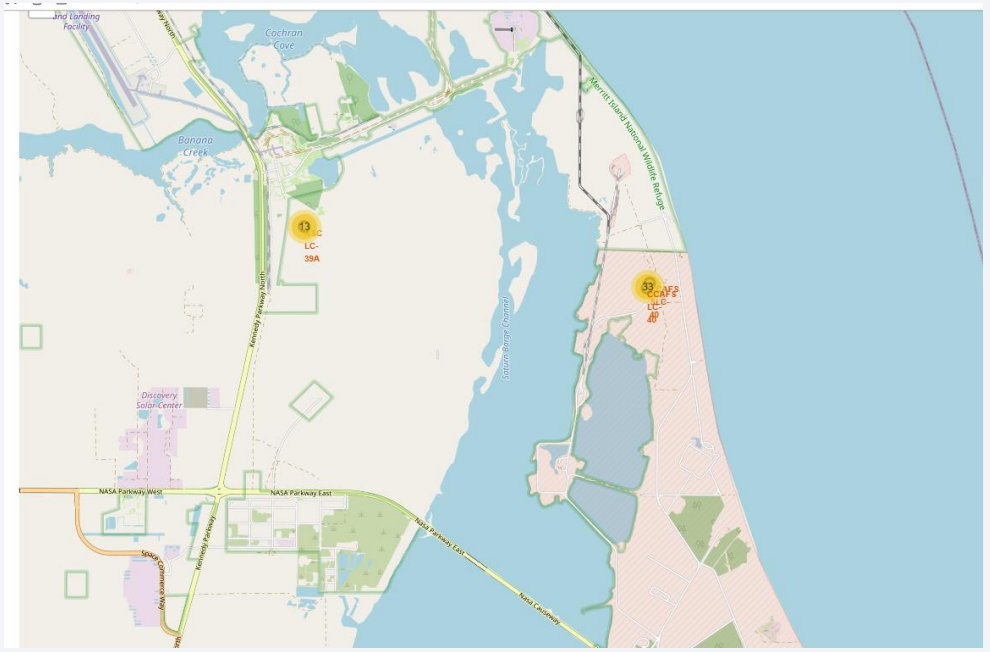
- The Map below show the location of the three launch sites. The Name of the launch sites are label on .the map markers



Launch Sites with Class Group markers

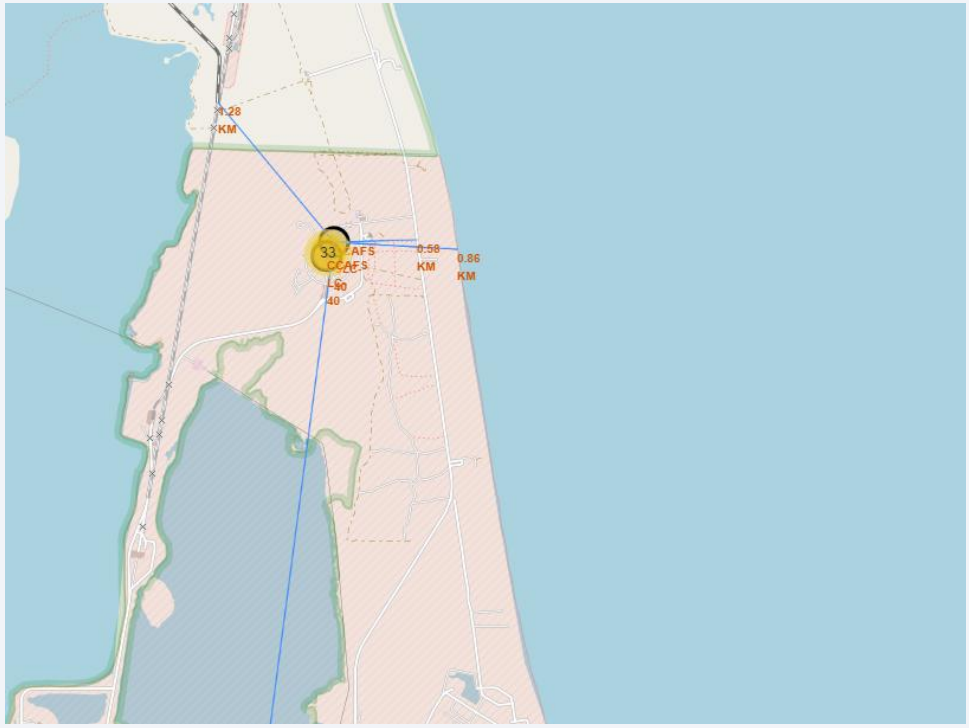
- For each Launch Site location there exist a group of markers. When the induvial clicks he can see number of launches that accrued at that site. The markers are Green and red for Successful re-landing/Class = 1 and Red/Class = 0

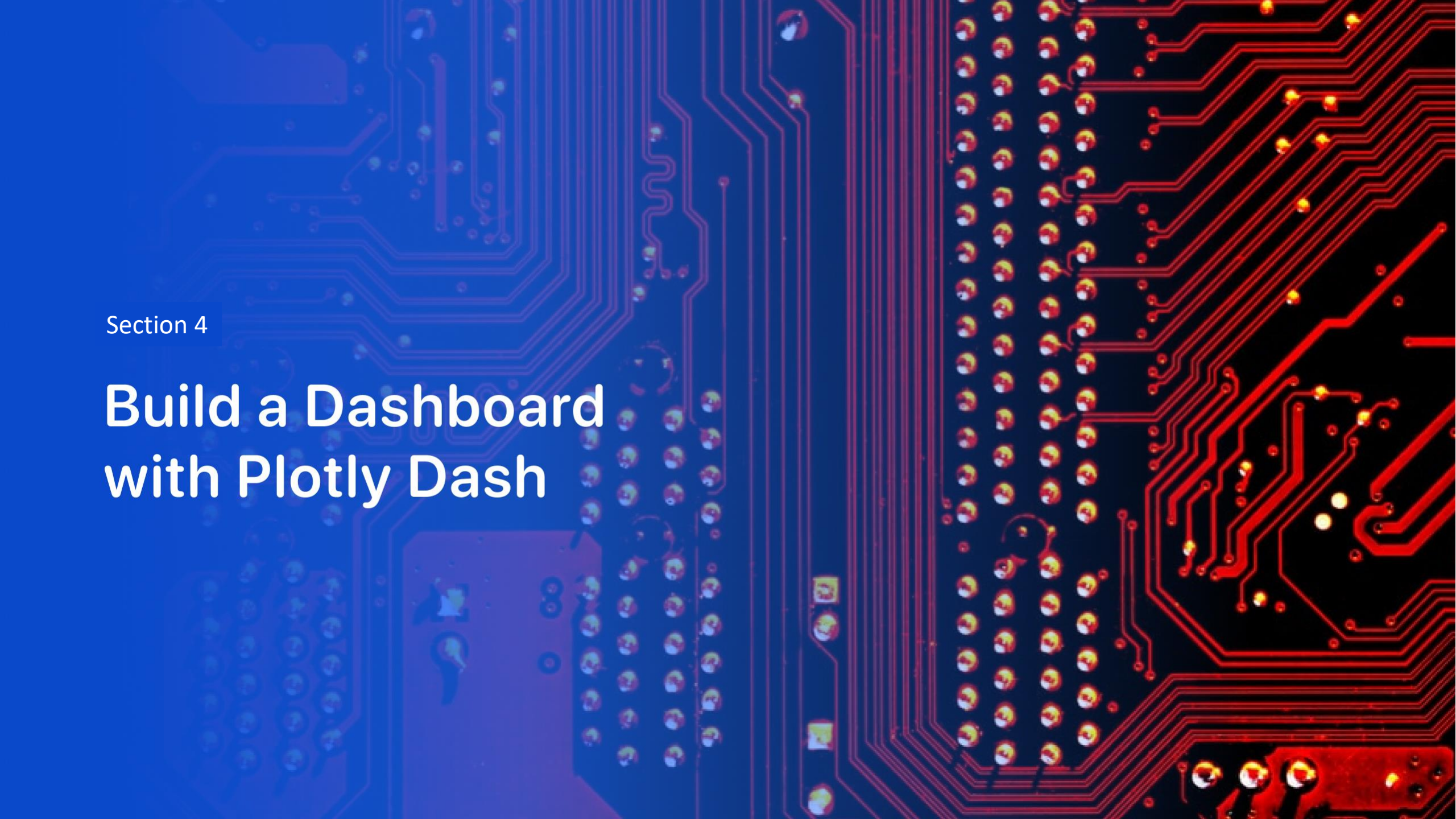




Lines to the cost, tracks, road, major city

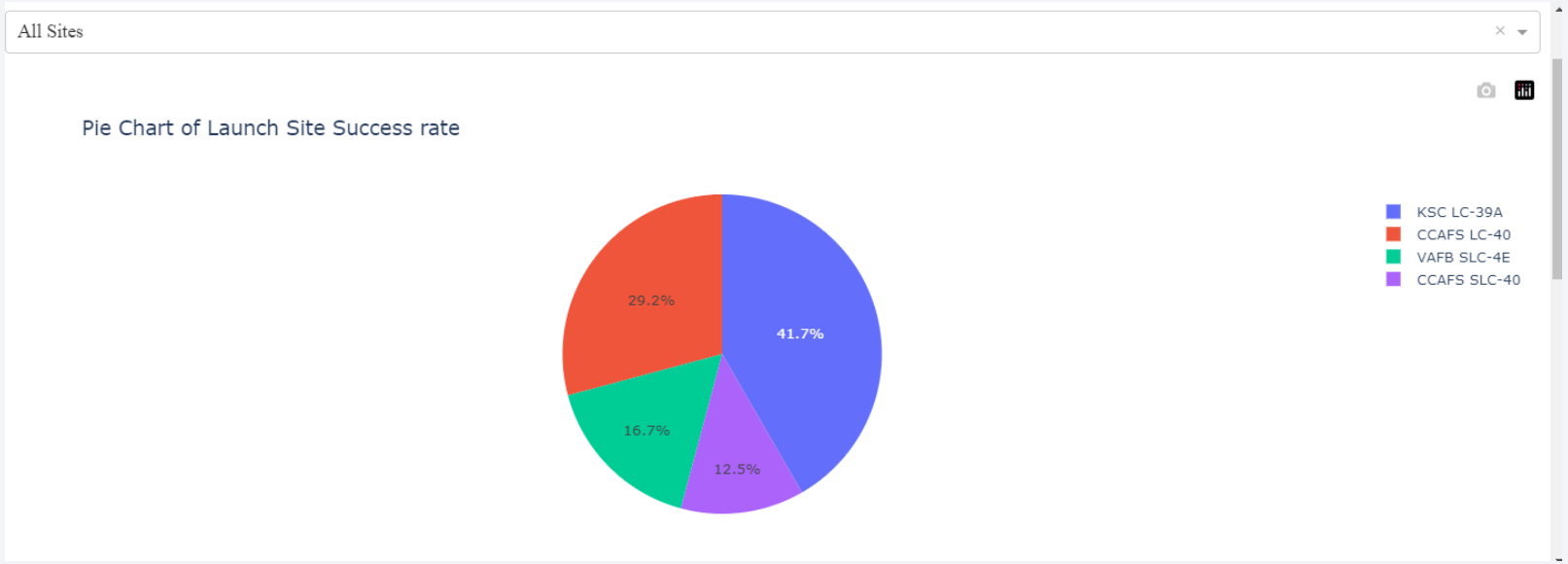
- The Blue lines on the map is the distance between the nearest cost line, road, train tracks, and Major city. From visualization it is easy to observer that Launch site are relatively close to the cost line, roads and Train track and far away from major city.




Section 4
Build a Dashboard with Plotly Dash

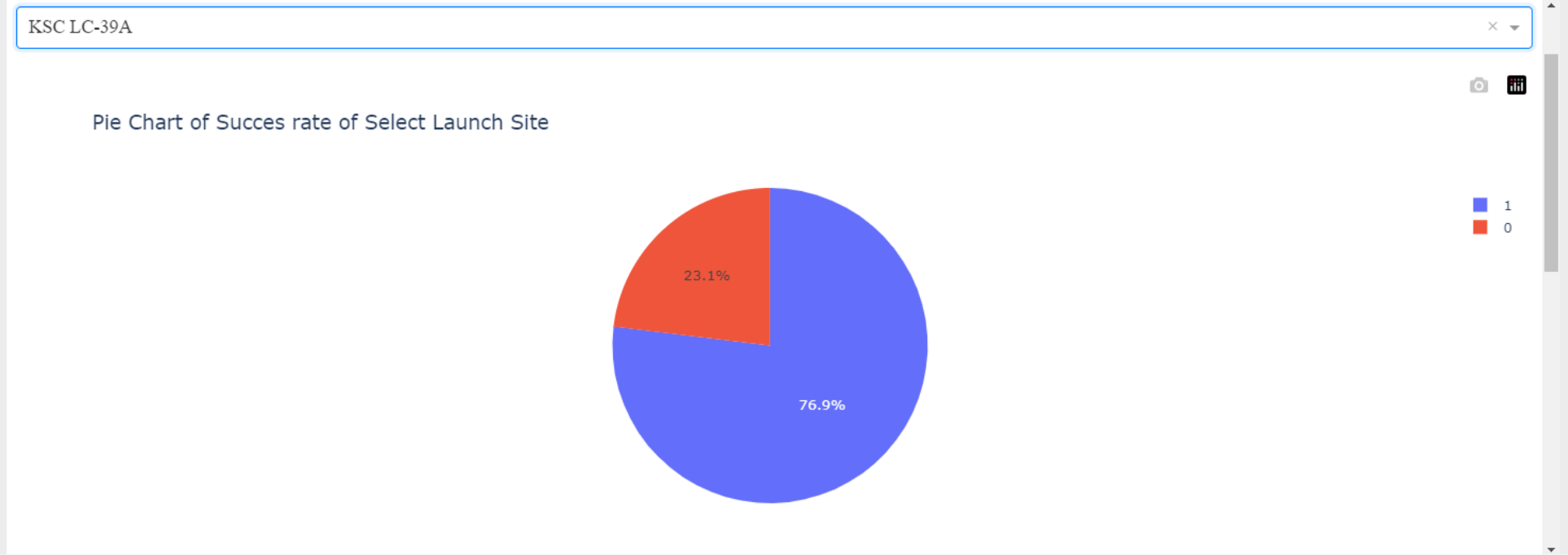
Pie Chart of Launch Sites success

- The pie chart is ratio of each Launch Sites Success rating. From the graph KSC LC – 394 dramatically out preforms the other Launch sites in term of Successful re-landing.



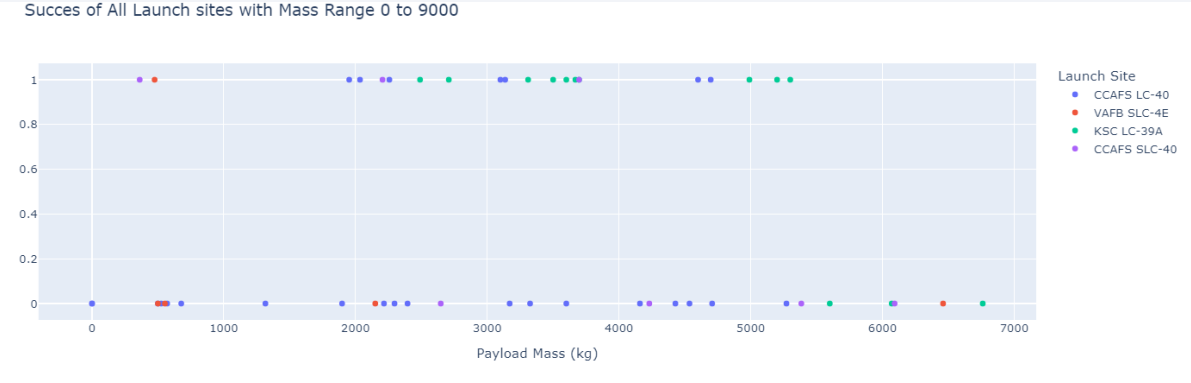
Pie Chart of the Success and Failure ratio for single Site

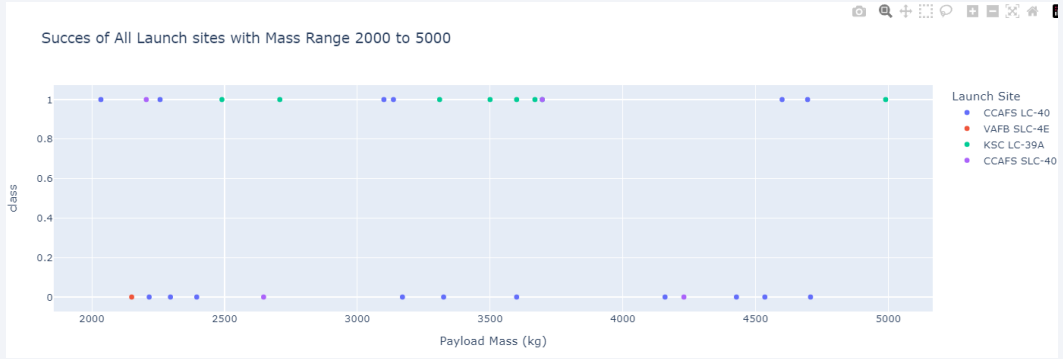
- The pie chart bellow is ratio of Successful and Failed re-landings at Site KSCL LC-39A. Around 76% of re-landings were successful.



Scatter Plots Payload Mass Vs Class with color Launch Site

- Below there are two plots. Both plots are Payload Mass Vs Class where color is Launch Site. Plots interactive feature is the range for Payload Mass or the X-axis. The user is abele to observer how mass effects Class.
  - Plot on the left has a Mass range of 0 to 9000 kg.
  - Plot on the right has Mass Range of 2000 to 5000 kg



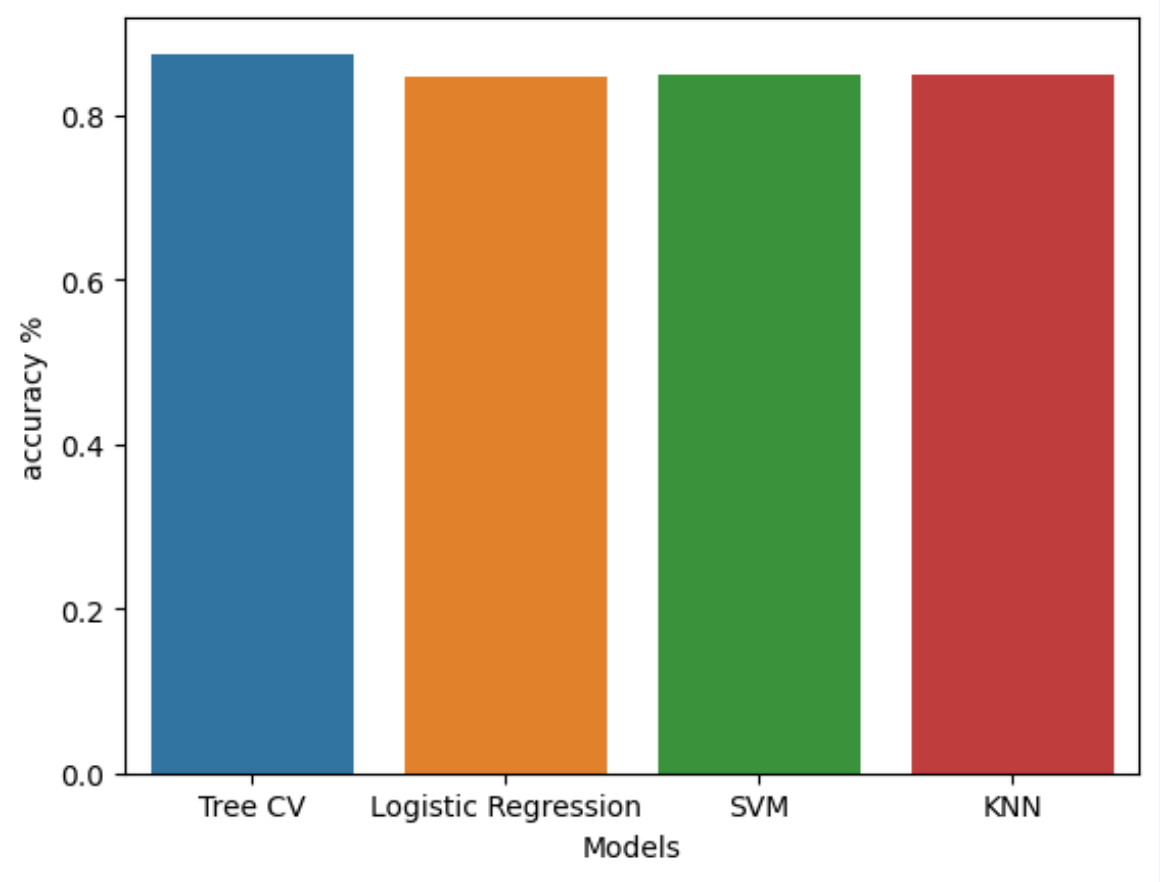


Section 5

Predictive Analysis (Classification)

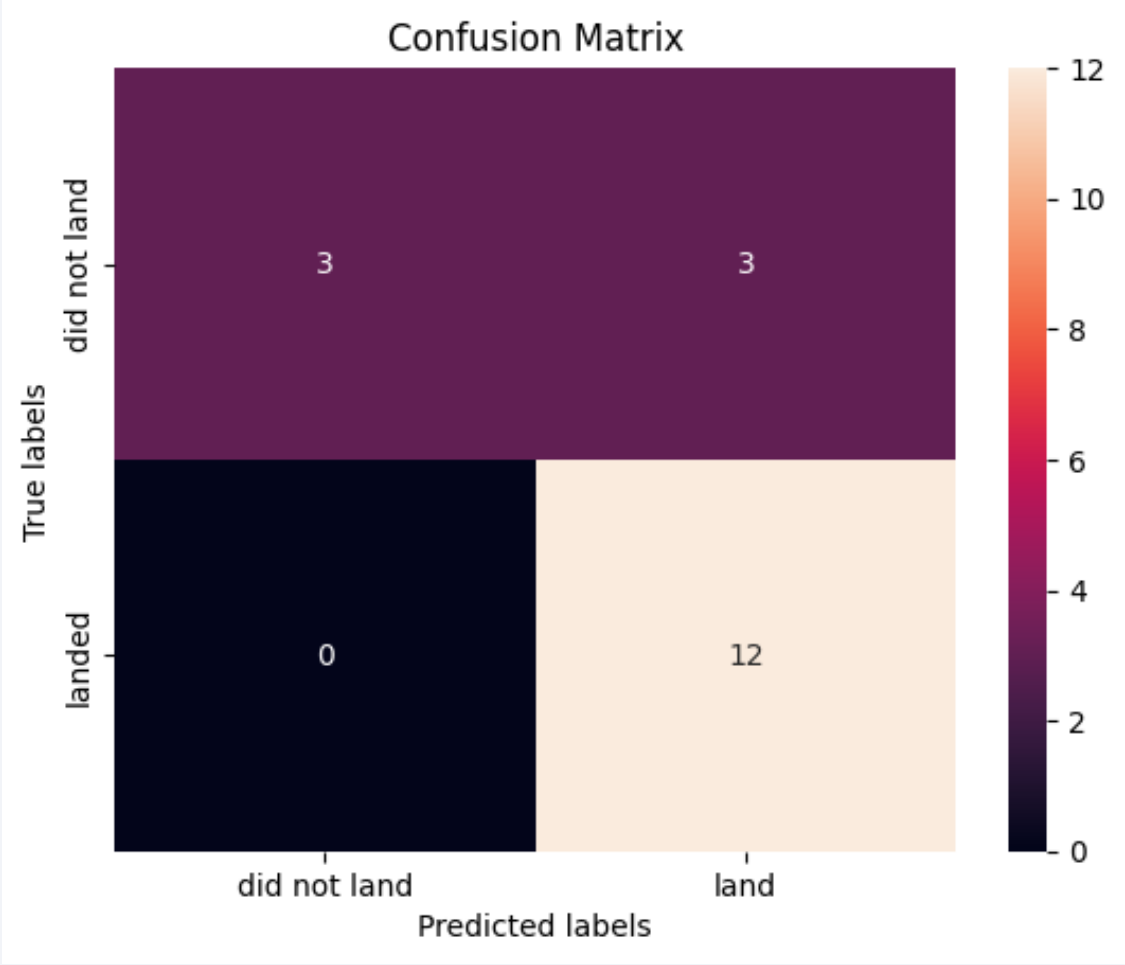
Classification Accuracy

- To the Right is a Bar Chart of Models used for are Classification Model where Y axis is Accuracy on test Data and X axis is the model used. Form the graph Tree CV is the best preforming model.



Confusion Matrix

- To the right is the Confusion Matrix for the Tree CV model. On the Y-axis are the True Class values while on the X-axis are the Tree CV predicted values. From the matrix we can see that 15 prediction were accurate while 3 were False positives.



Conclusions

- The Tree CV model should be used to predict re-landing out come of the first stage
- The Payload Mass effects the Class in different ranges depending on the Launch Site.
- The orbit of the Spaceship effects the re-landing of first Stage
- Geography and location near major sites effect Launch sites Class or re-landing success rate
- As the number of launches increases the re-landing success rate increases.

Appendix

- Project and Files location: https://github.com/FlynnFayman/IBMCAPStoneProjects/tree/main
- Programs and packages used: Python ,Pandas, Dashboard, Space X, Folium, Matplotlib, Seaborn, Skitlearn, SQLlite, Beautifulsoup, Ploty Dash

Thank you!