IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dhruv Mahajan
28-10-2024

Outline

Executive Summary

Introduction

Methodology

Results

Conclusion

Appendix

Executive Summary

DATA COLLECTION

DATA WRANGLING

EXPLORATORY DATA ANALYSIS

INTERACTIVE DASHBOARDS

PREDICTIVE ANALYSIS

Introduction

- Space Y founded by Allon Mask, has tasked us to determine the price of each launch by gathering data from Space X and building dashboards
- We would like to know the cost of the launch?
- We would see if we can reuse the first stage?

Section 1

Methodology

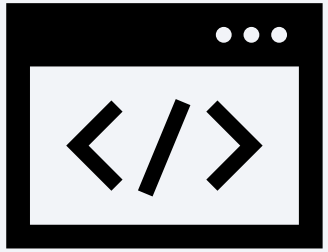
Methodology

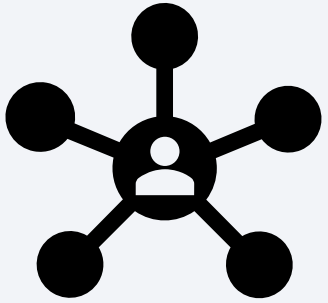
Executive Summary

- Data collection methodology:
  - I collected the data via SpaceX REST API and Web Scrapping
- Perform data wrangling
  - Used functions to get the JSON data into a DataFrame, removed null values
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Built a ML pipeline to predict if first stage will land, trained with 4 different models to see which is the best one.

Data Collection

I collected the data using REST API and web scrapping. I got the data from API in JSON, after using pandas to normalize and remove the null. Made functions that could extract the data from each and put it into a list, using a dictionary I made a Data Frame with only with the useful information. Now I saved the Data Frame using Pandas. This will allow us to do more operations of the data later.



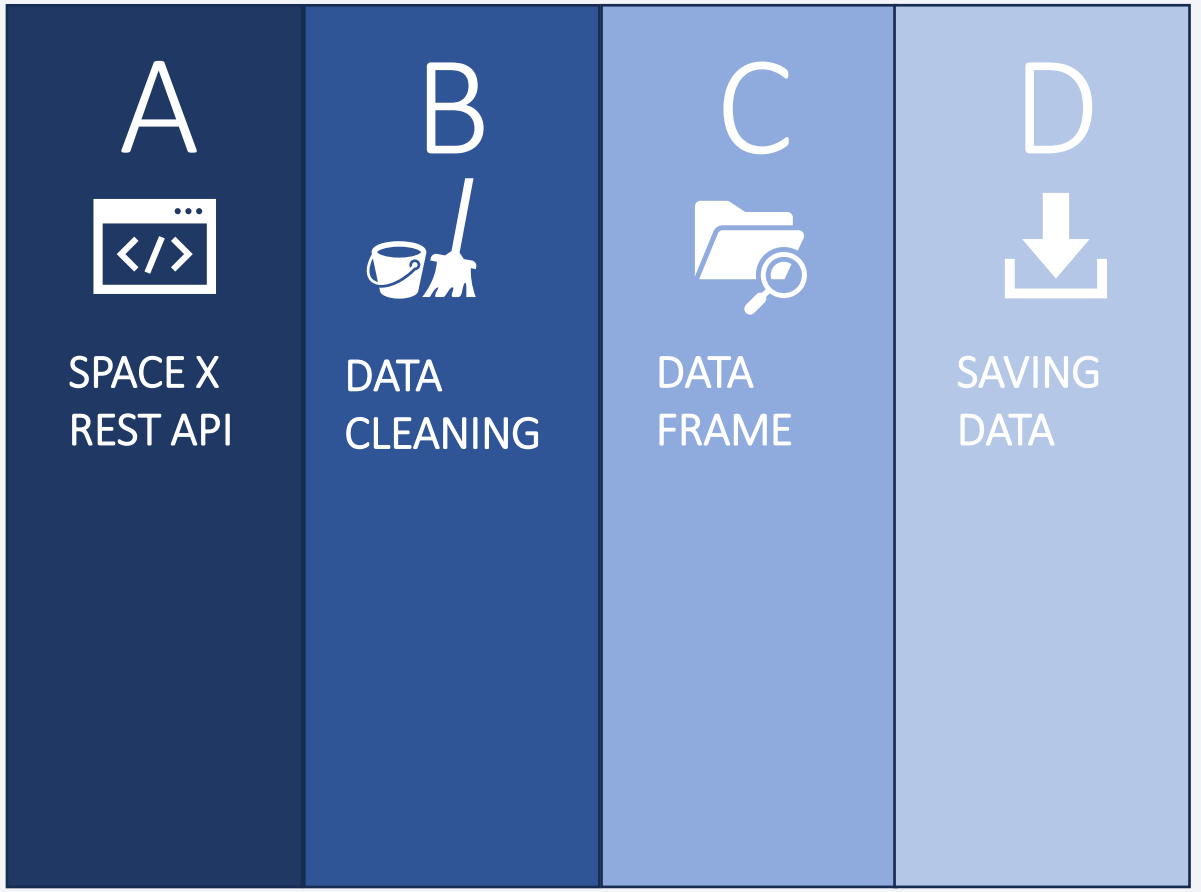


I used web scrapping to collect the data of the Falcon 9 launches using Wikipediae , I used requests and get to get the data. Then I used beautiful soup to parse through the object and got the useful data for the launches. Now I saved the data using Pandas so I can do more analysis of the data later.

Data Collection – SpaceX API

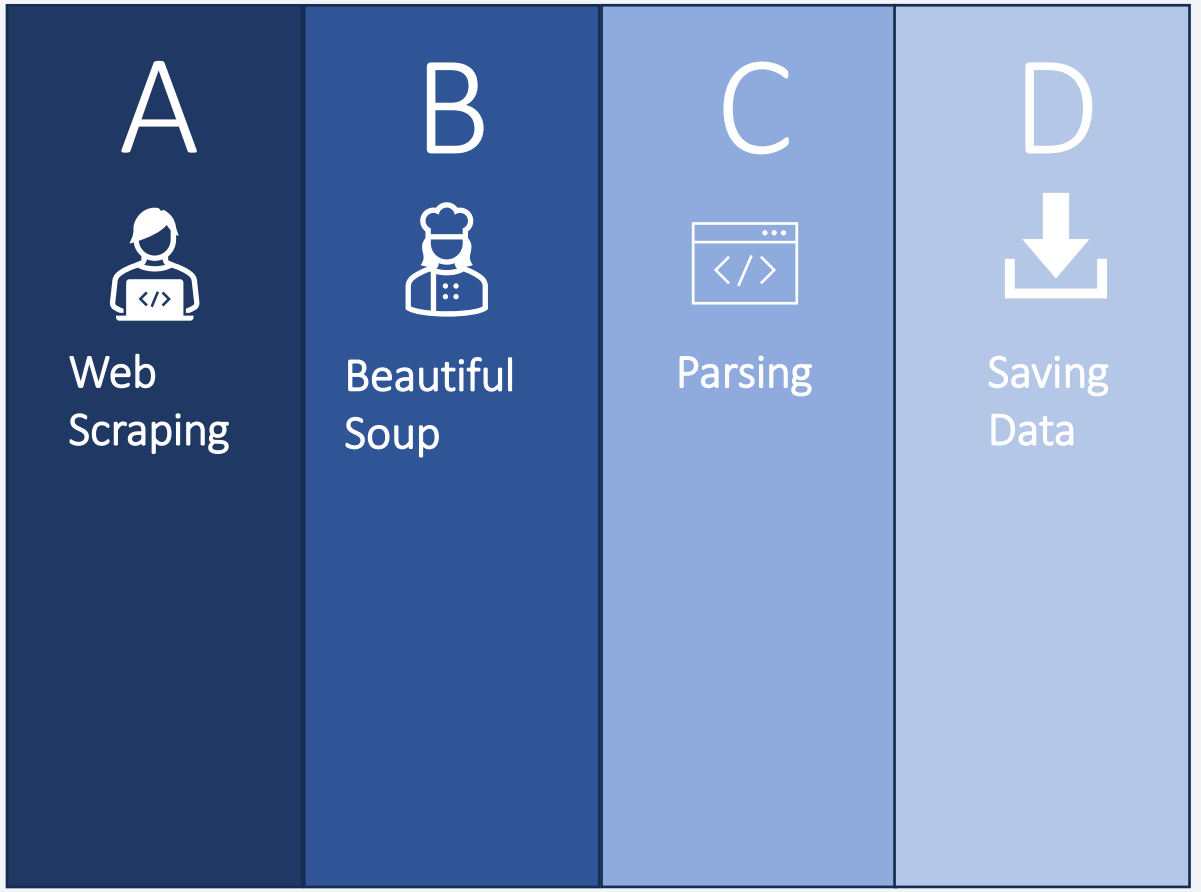
Link to GitHub

- GitHub API calls
- As we can see the API calls were made using which I got the JSON data. The JSON data was normalized and made into a Data Frame. Then using a dictionary with all the useful data, I made a Data Frame and saved this Data Frame.



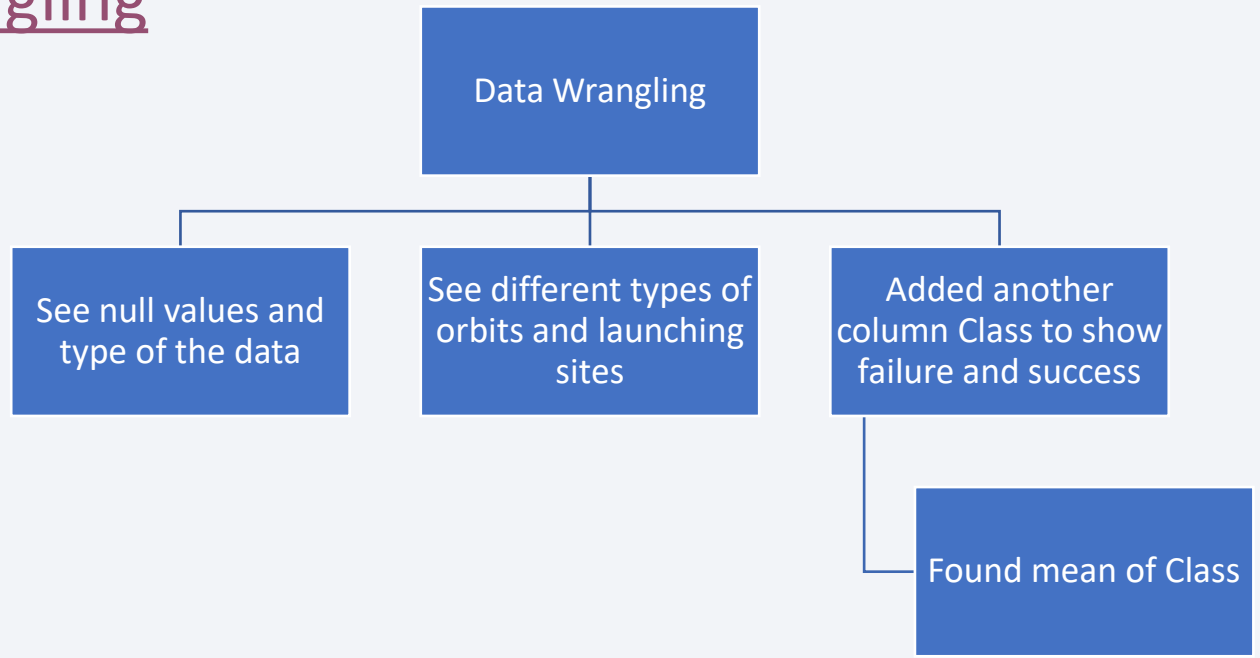
Data Collection - Scraping

- We first used the URL of the Wikipedia page to get the data using Requests. We then used Beautiful Soup to parse the data and find relevant data to Falcon 9 Launch. After this we saved the data into a Data Frame.
- Web Scrapping



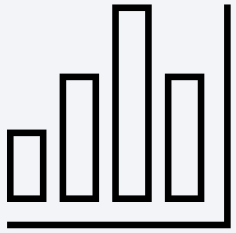
Data Wrangling

- First I saw the null values in the data, saw the different launching sites and the different types of orbits. I also added another column class based on the landing if it was a success or failure. This column is called Class and the mean came up as 66.66%
- Data Wrangling



EDA with Data Visualization

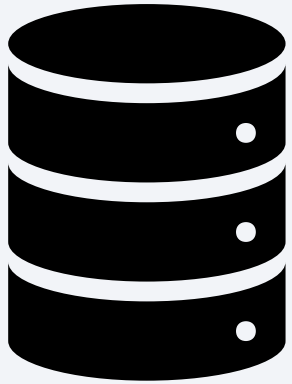
- I plotted and analyzed the data using Matplotlib and Seaborn.
- The link to the notebook is here.



Visualized the different parameters and their effects on the result, saw how the difference on launch sites and payload and other parameters had differing results, this allowed to make hypothesis on how to approach. One major update was that at first the technology didn't work and wasn't reliable only after 2016, does the success rate go up.

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- Link for EDA with SQL is here.
- Used Magic SQL to run queries for different weights and launch sites. Saw the average payload mass using SQL, using Min saw the first ground landing.

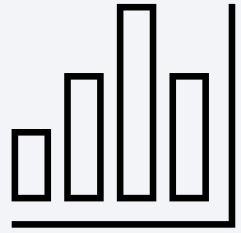


Build an Interactive Map with Folium

- Using Folium, framework I added markers to the map of the launch sites. Using our data I added markers to identify whether it was a success by green and failure by red,
- Wanted to see what are the reasons for choosing that as a launch site, like the railways, roads and waterways. Noticed the nearest city.
- The notebook for the interactive map is here.

Build a Dashboard with Plotly Dash

- Using plotly and dash built interactive dashboard, this allows to see the data and interact and understand it pretty well.
- Built a slider for payload mass, and callback for it to be interactive and dropdowns to view.
- The GitHub link for the Plotly Dashboard is here.



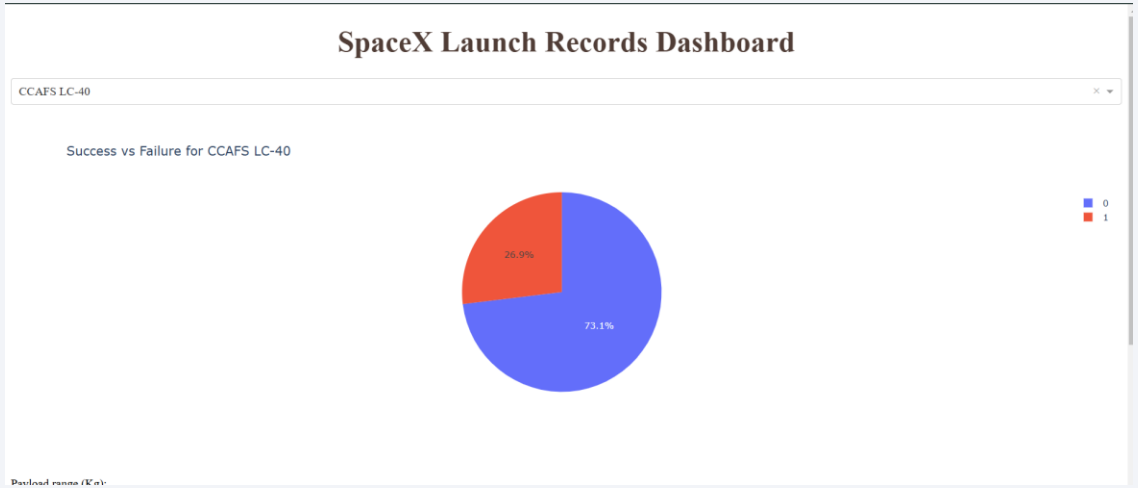
Predictive Analysis (Classification)

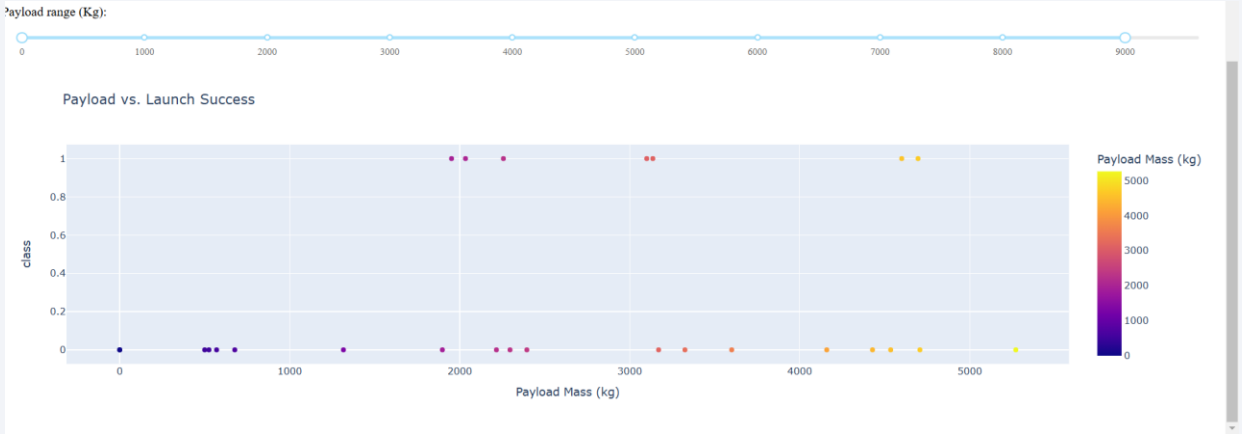
- Got the data and split the data into train and test, with random state 2. Made the object of the model and set the parameters based on grid search. Fit the data into model and tested how the result is going to be.
- Plotted the confusion matrix for the models to see the results and understand the results better.
- All the models had the same accuracy which was approx. 83% except Decision Tree which gave 94% accuracy and was the best model.
- The pipeline worked pretty well and we used four different models:
  - SVM
  - Decision Tree
  - Logistic Regression
  - KNN
- GitHub link for the notebook is here.

Results

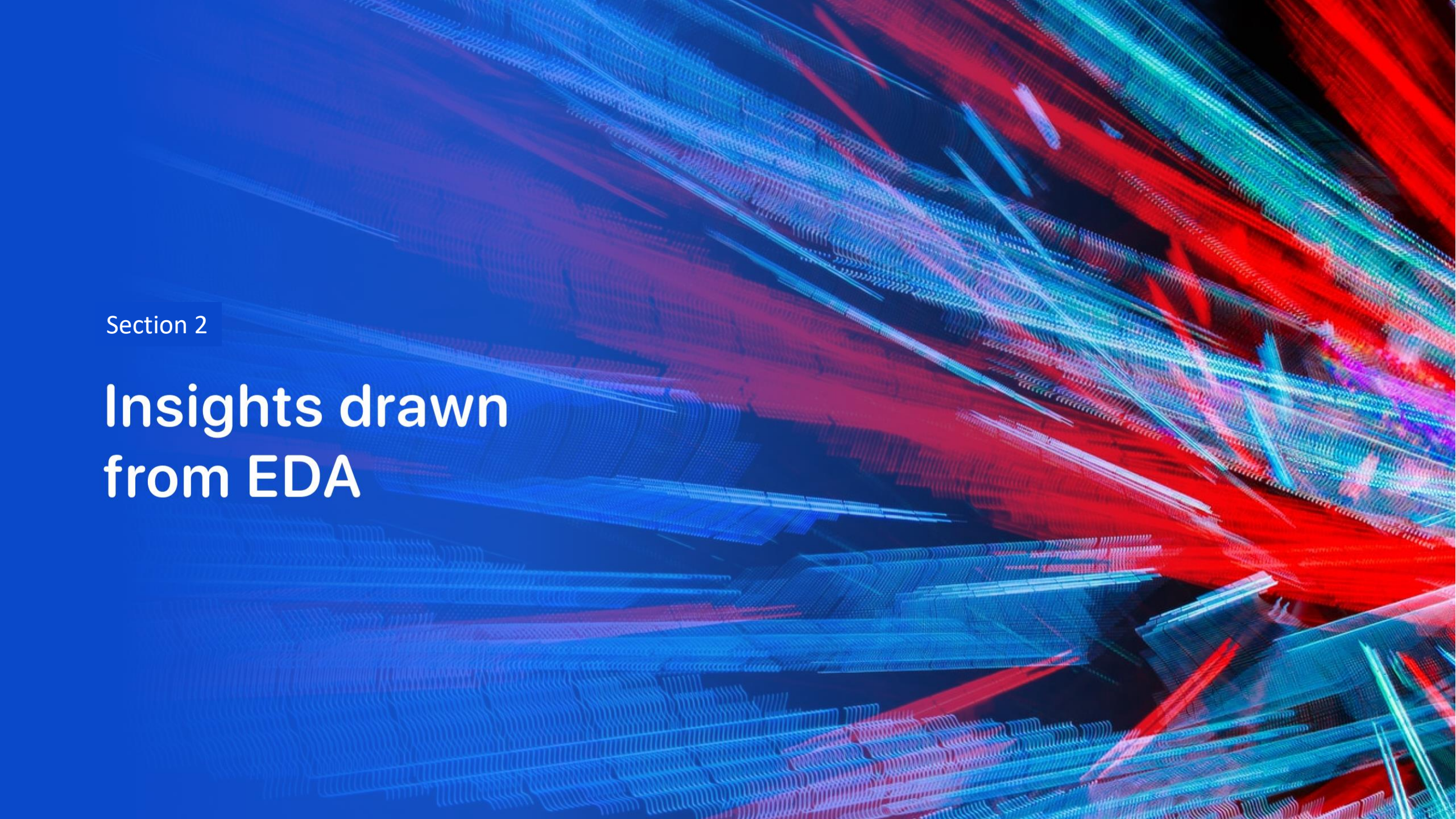
Out of the 4 model 3 of the models gave the same result, while the decision tree gave the best result of 94%

KNN	SVM	Decision Tree	Logistic Regression
• 83.3%	• 83.3%	• 94%	• 83.3%





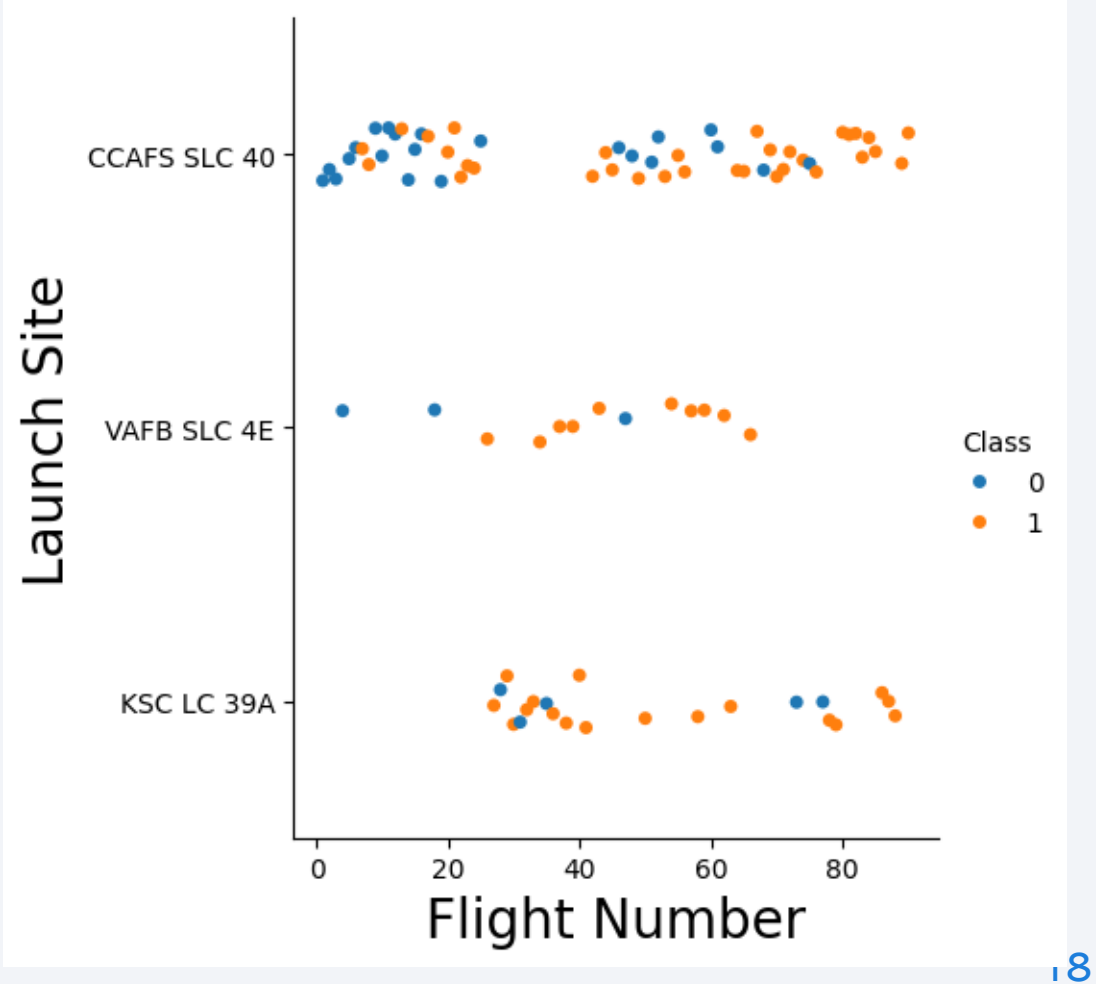
- We see that as flight number increases that is indicating continuous launch attempts, shows that more chance of success.
- We also see that some launch sites give more chance of success.
- Some launch sites can't really handle huge amount of payload.
- Orbits also are a huge factor as some give huge amount of success.
- Another important finding was that after 2013 success rate increases and increases rapidly after 2015.


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

This plot shows the number of times the flight has taken place and in which Launch Site. We can see the lower flight in VAFB SLC 4E is unsuccessful but later the higher flight numbers are majority of them are success.

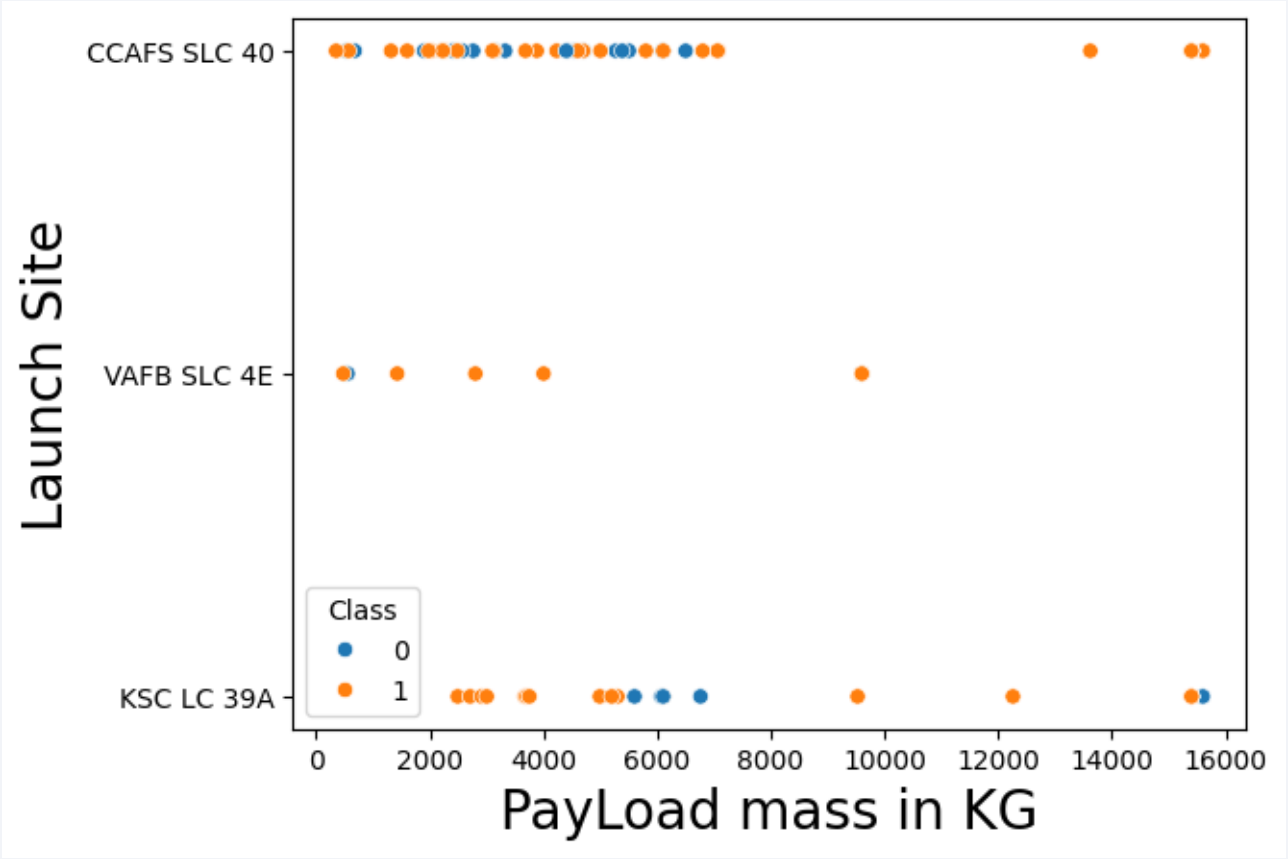
This shows flight number and launch pad are important factors.



Payload vs. Launch Site

In this plot we can see the different launch sites and their payload with the success. We can see that the KSC LC 39A is successful with lighter and heavier payloads.

This allows us to better understand the correlation between payload and launch site.



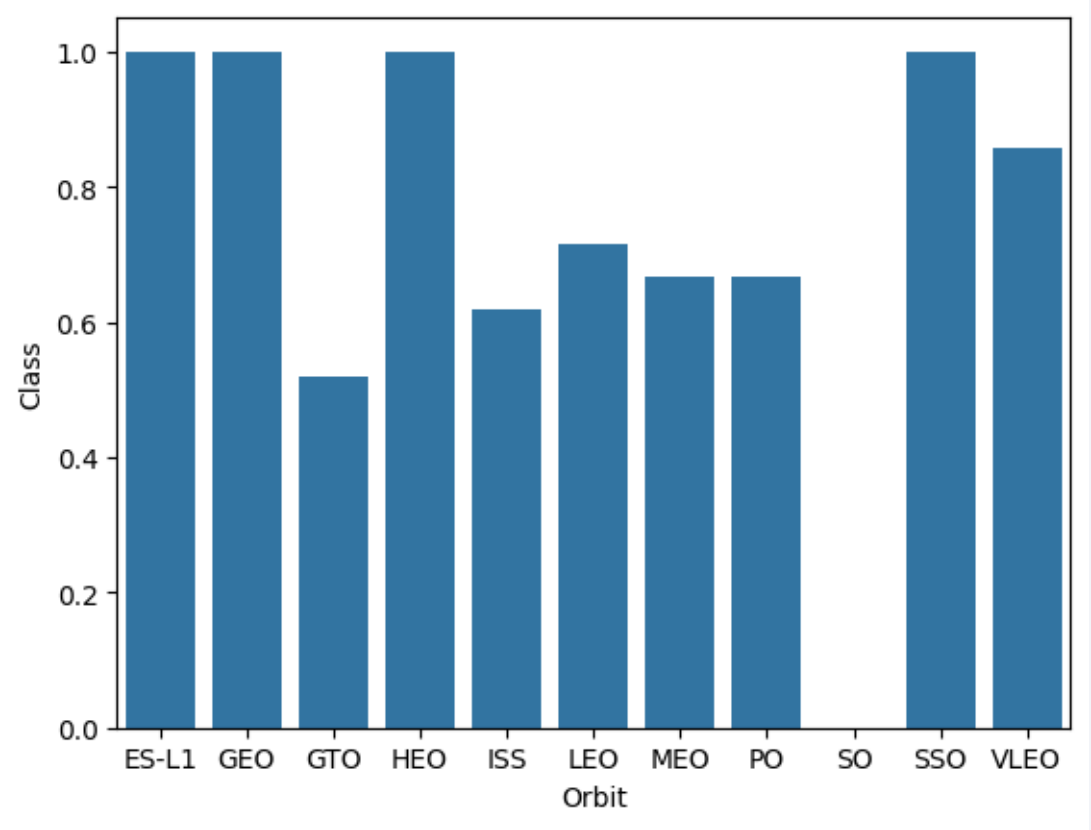
Success Rate vs. Orbit Type

The success rate of different orbits, is on the plot.

As we can see the SO orbit has a 0 chance of success

While orbit like GTO are always successful.

This leads to orbit type being a huge factor on prediction of success or failure.

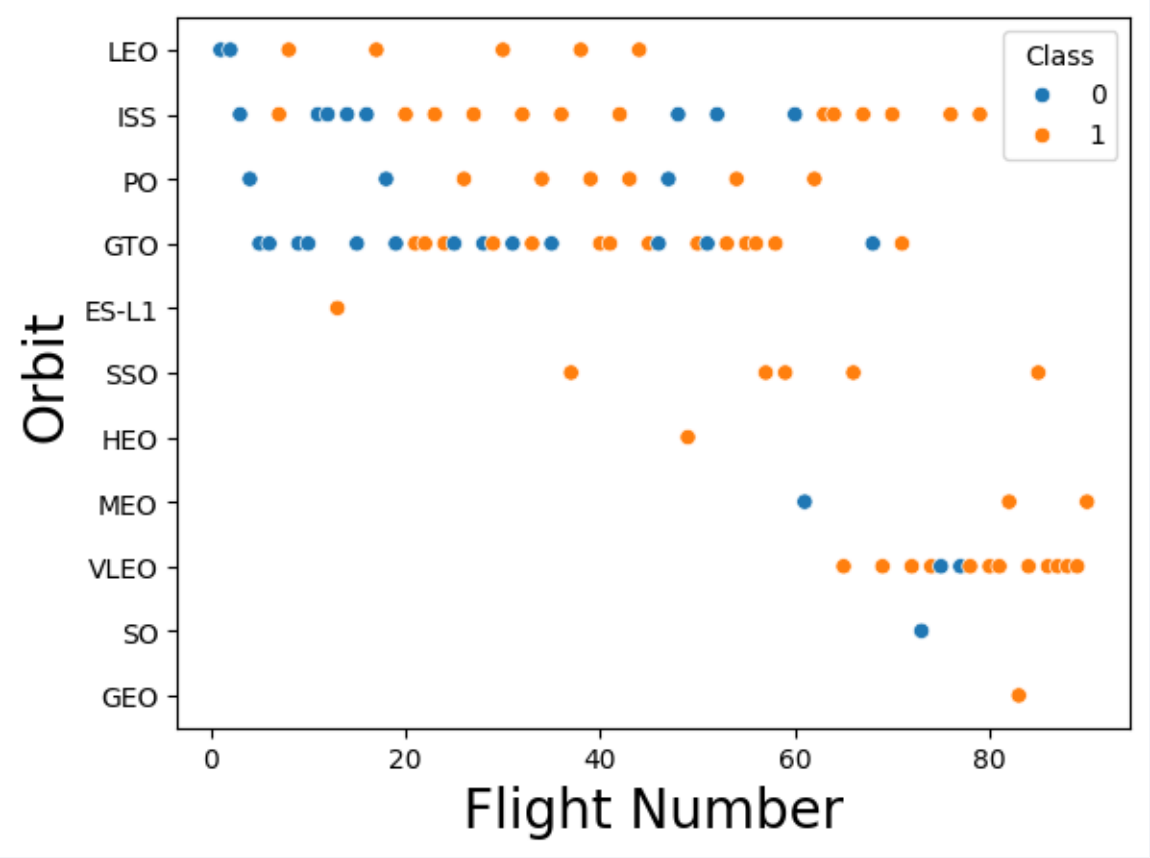


Flight Number vs. Orbit Type

As flight number increases the chances of success is more as that many times the flight is being done again.

This allows us also to see which flights are done multiple times based on the orbits and their success.

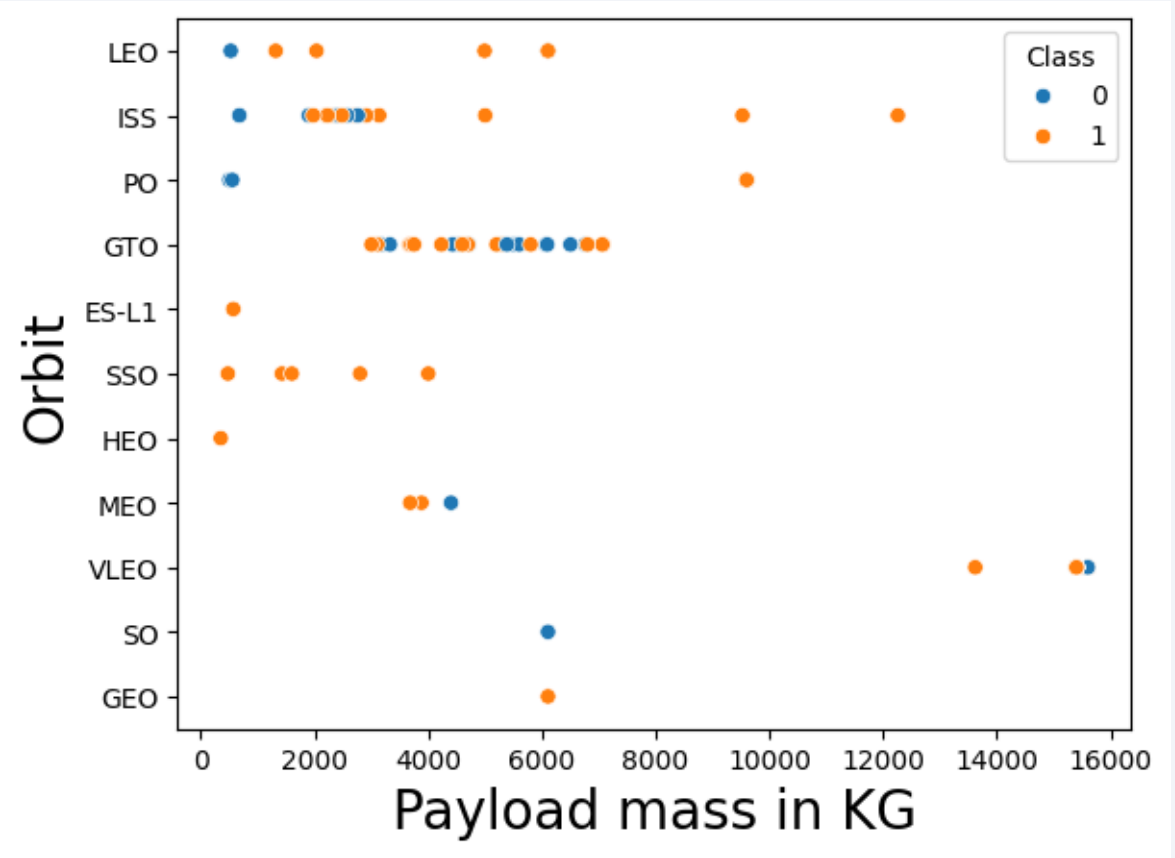
For example, VLEO orbit is successful and has high flight number



Payload vs. Orbit Type

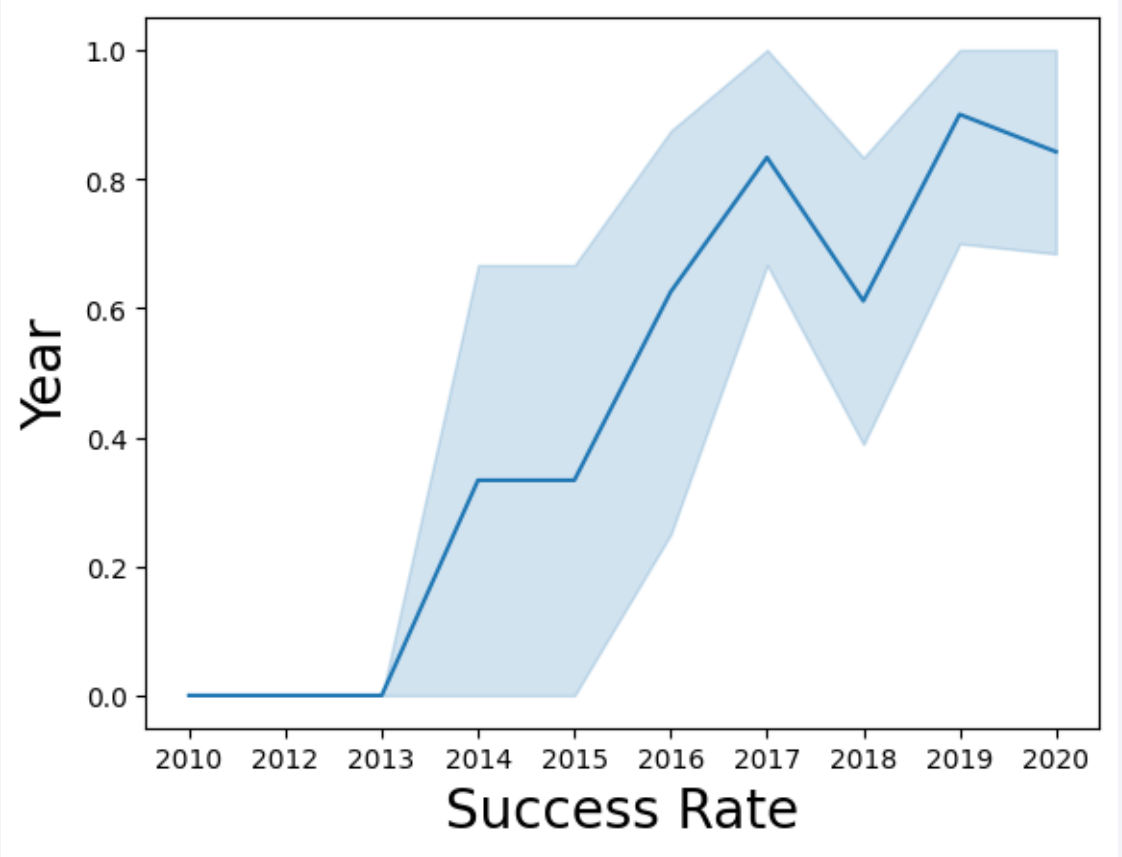
We can see the payload differs in some orbits and so does the success, for example SSO orbit is always a success and payload is less, while in some orbits the payload is not a huge factor like GTO.

Thus, visualizing it helps us understand how to go ahead with our data.



Launch Success Yearly Trend

This line plot shows us the Success rate over the years. As we can see the success rate increases from 2013 and is constant till 2014 to 2015, but rapidly increases later and reaches new heights never before.



All Launch Site Names

This query gives us all the distinct Launch Sites which later will allow us to see if the launch sites make a difference in the result of the flight or successful landing.

Task 1

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

[11]:
```
%sql select DISTINCT "Launch_Site" from SPACEXTABLE
```

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

[11]:

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

Launch Site Names Begin with 'CCA'

This shows us the data of Launch sites with name beginning from CCA, there are 2 different sites with the name like starting from that as we can see later on the map.

Task 2

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

[20]:
```
%sql select * from SPACEXTABLE where "Launch_Site" like 'CCA%' limit 5
```

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

[20]:

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

This is total sum of payload caried by NASA CRS as customer

Task 3

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

[22]:
```
%sql select sum("PAYLOAD_MASS__KG_") from SPACEXTABLE where Customer="NASA (CRS)"
```

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

[22]:

sum("PAYLOAD_MASS__KG_")
45596

Average Payload Mass by F9 v1.1

The average payload mass is 2900 approximately which shows us how to go on later and see the heavier payloads and lighter ones and being able to distinguish the payloads.

Task 4

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

```
[28]: %sql select avg("PAYLOAD_MASS__KG_")from SPACEXTABLE where Booster_Version='F9 v1.1'
```

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

[28]:

avg("PAYLOAD_MASS__KG_")
2928.4

First Successful Ground Landing Date

We see from this query that any data from before this date will always be a failure as the technology to do the ground landing wasn't proper to ensure success. As the technology was better after 2015, we can expect to land successfully

Task 5

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

Hint:Use min function

[26]:
```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome="Success (ground pad)"
```

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

[26]:

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

These are the boosters which have Success in drone ship at higher payload range than normally , I believe boosters are a huge factor as the payload increases and these are detrimental in the result of Success and Failure.

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

[29]:
```
%sql select Booster_Version from SPACEXTABLE where ("PAYLOAD_MASS__KG_">4000 and "PAYLOAD_MASS__KG_"<6000 and Landing_Outcome="Success (drone ship)")
```

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

[29]:

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

This query gave us the different Mission Outcomes and their counts which in turn gives us a good reason to believe which ours will be as per probability.

Task 7

List the total number of successful and failure mission outcomes

[42]:
```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") FROM SPACEXTABLE GROUP BY "Mission_Outcome";
```

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

[42]:

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

Boosters Carried Maximum Payload

These are all the boosters that carry the maximum payload, from the overall data we have. Used a subquery to get the result

Task 8

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

In the year 2015 the months were January and May when there were failures using drone ship.

Task 9

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),"Landing_Outcome","Booster_Version","Launch_Site" from SPACEXTABLE where substr(Date,0,5)='2015' and
"Landing_Outcome"="Failure (drone ship)"
```

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

substr(Date,6,2)	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

As we can see from the data later on the landings are success as we can see the count is also pretty high.

Task 10

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 count("Landing_Outcome") as count,* from SPACEXTBL where (DATE between '2010-06-04' and '2017-03-20') group by "Landing_Outcome"
order by date desc
```

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

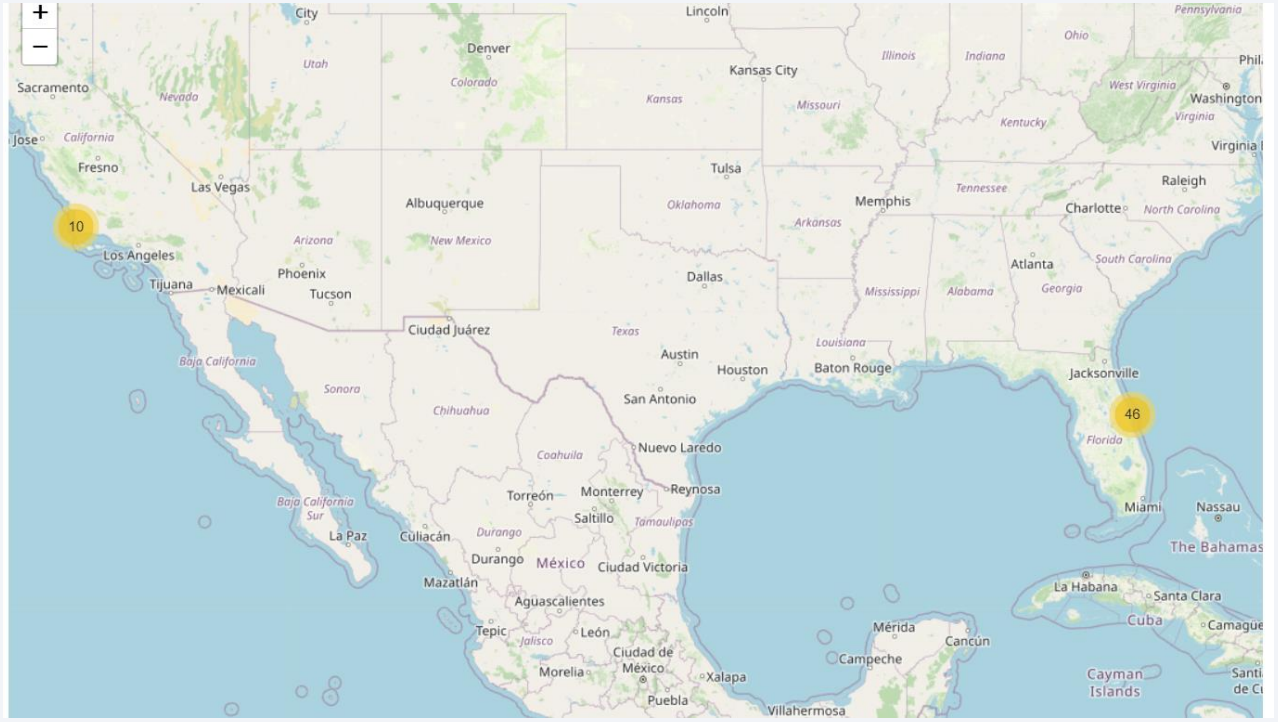
count	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
5	2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)
3	2015-12-22	1:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)
1	2015-06-28	14:21:00	F9 v1.1 B1018	CCAFS LC-40	SpaceX CRS-7	1952	LEO (ISS)	NASA (CRS)	Failure (in flight)	Precluded (drone ship)
5	2015-01-10	9:47:00	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)
3	2014-04-18	19:25:00	F9 v1.1	CCAFS LC-40	SpaceX CRS-3	2296	LEO (ISS)	NASA (CRS)	Success	Controlled (ocean)
2	2013-09-29	16:00:00	F9 v1.1 B1003	VAFB SLC-4E	CASSIOPE	500	Polar LEO	MDA	Success	Uncontrolled (ocean)
10	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
2	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)

Section 3

Launch Sites Proximities Analysis

Launch Sites Map

This map shows the different launch sites, we can see that all the launch sites are near a city but not to near, and always near the ocean.



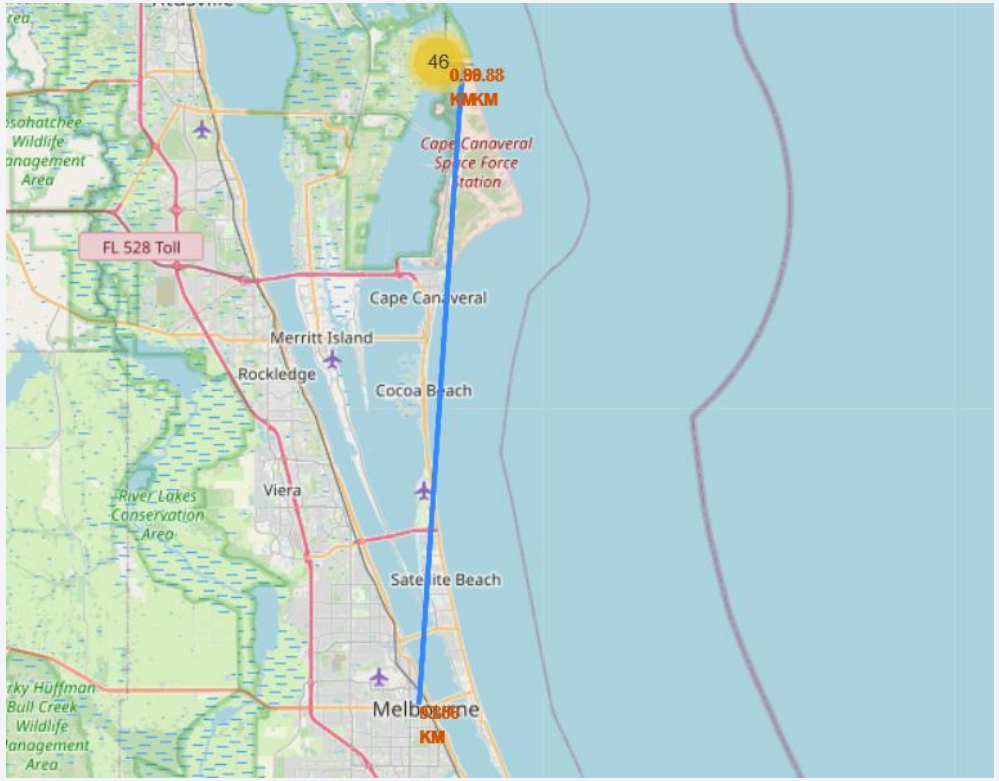
Adding Markers

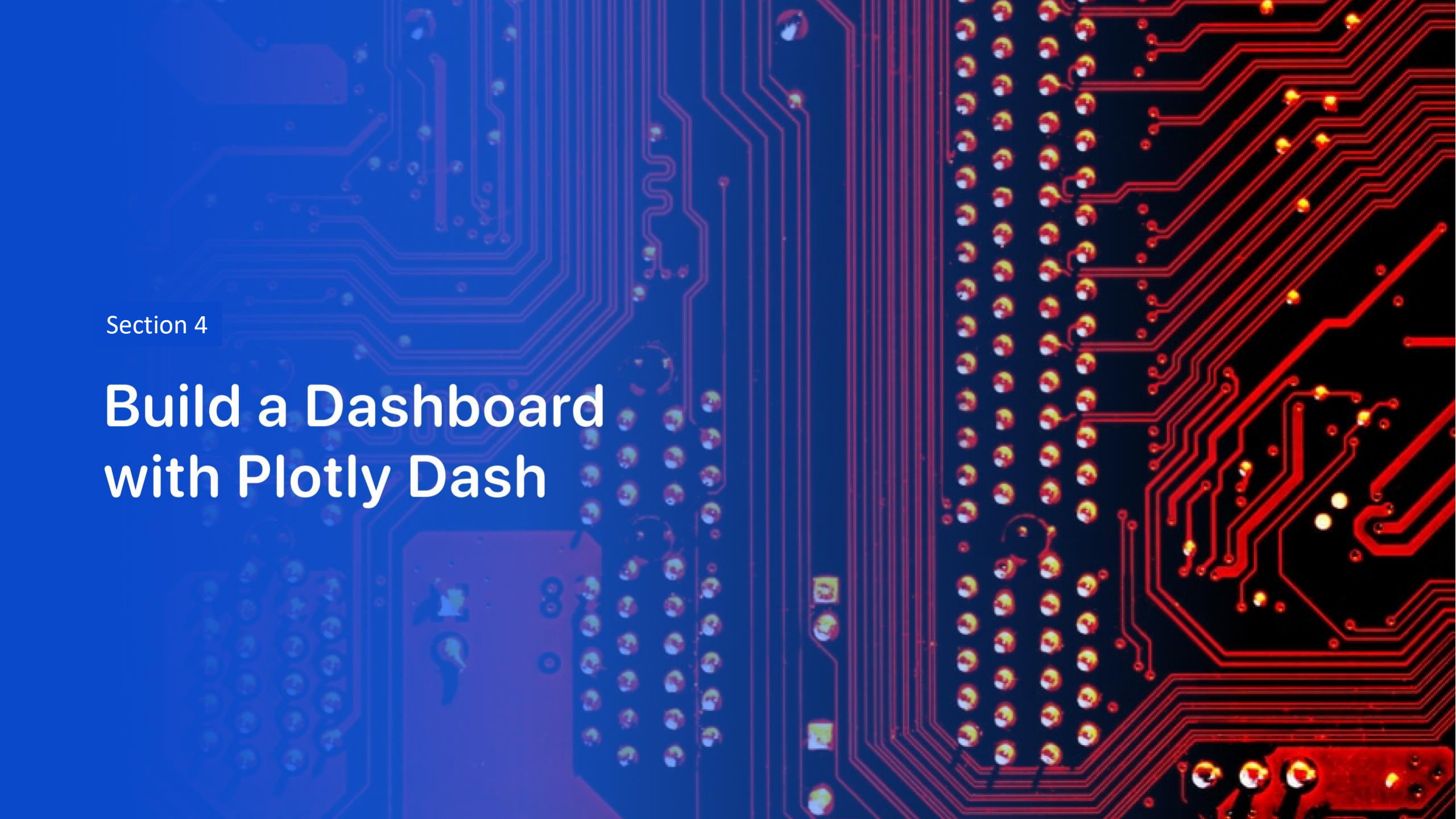
In this image we can see the launches and the number of failures and their locations. These markers help understand the data in more visual and easy manner to understand

Lines And Distance

Using the distance formula and line and markers, we added lines from the nearest city and sea.

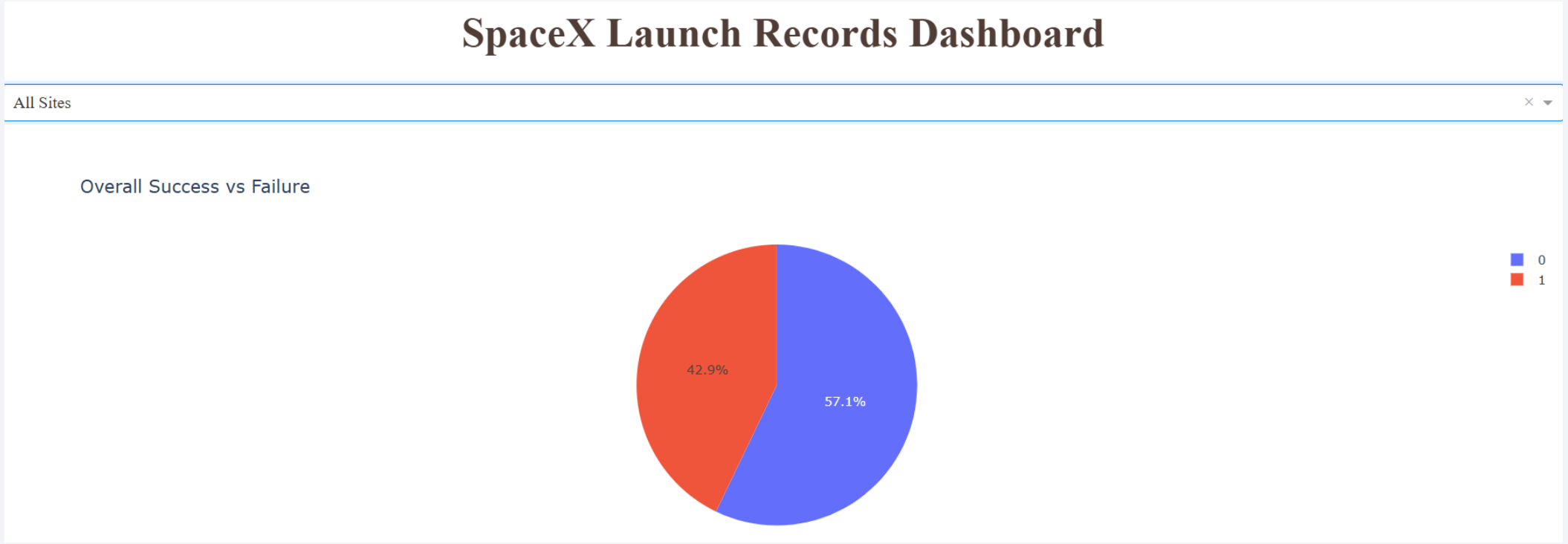
The Launch site being close to the sea makes sense as we do have landing on water and allows for safety, the site is close but not too close to the city which is for safety reasons I believe.




Section 4
Build a Dashboard with Plotly Dash

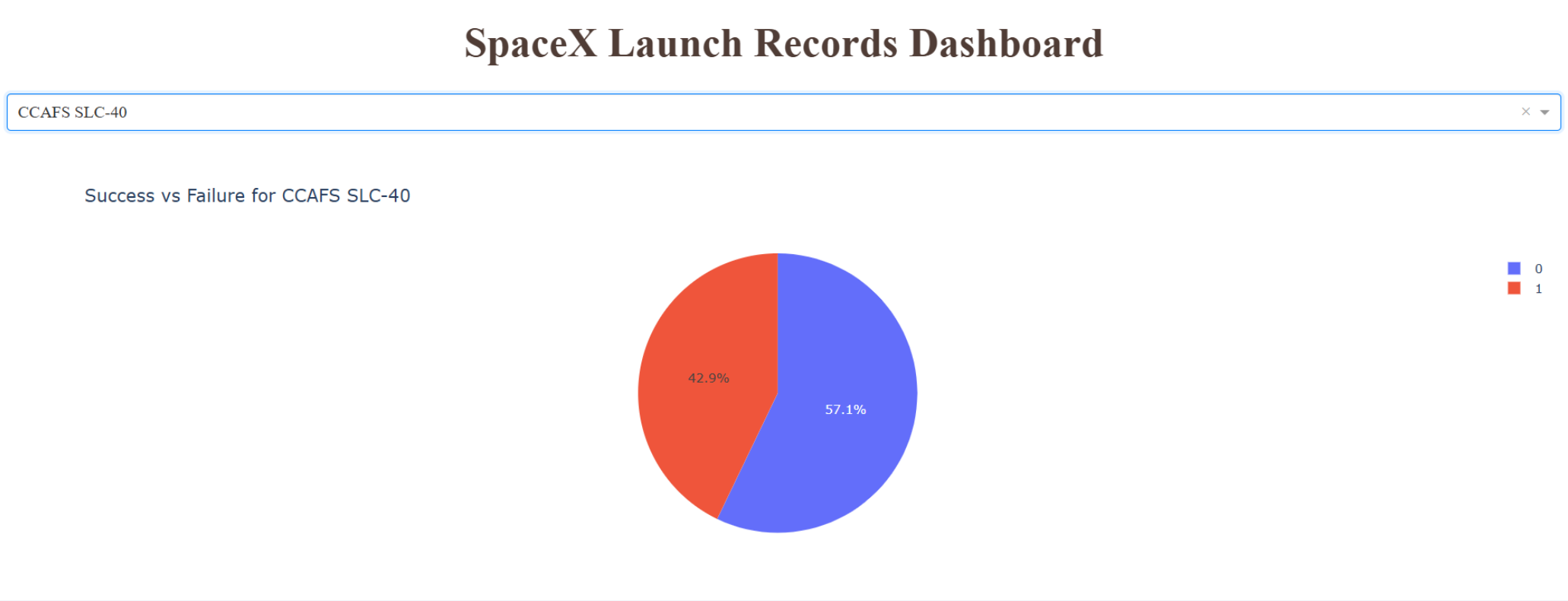
Overall Success

This pie chart shows the overall success rate of our launch despite any site, which shows its hard to predict but when we see the other sites, it becomes more easier as there is a huge variance



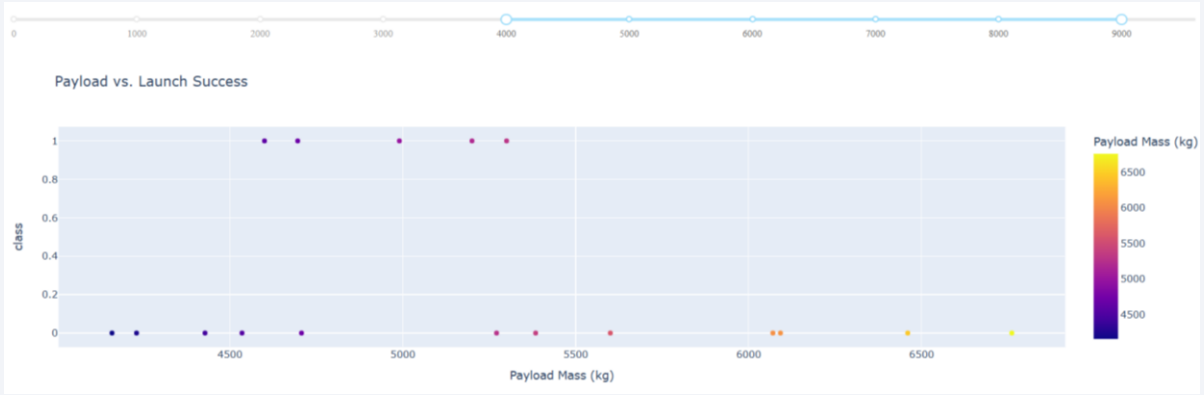
Launch Site Matters

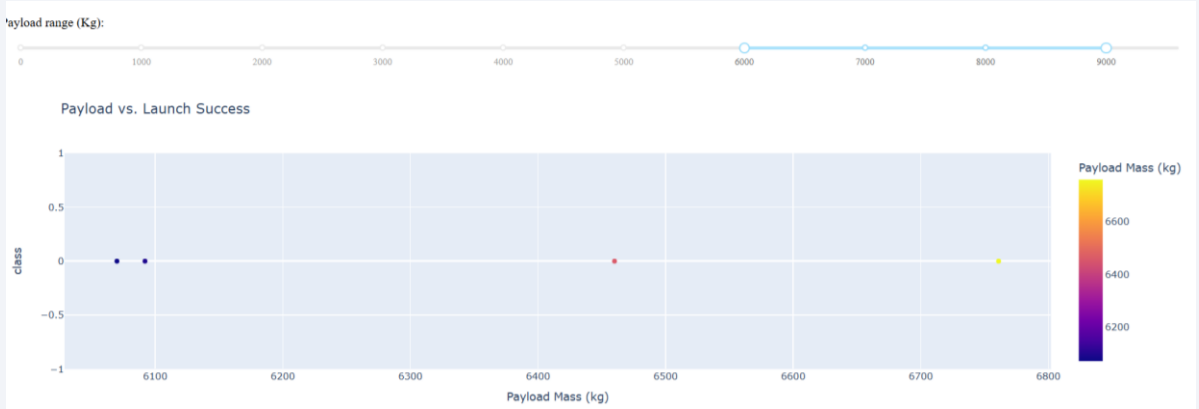
This Launch Site has the highest success rate and shows that if we launch from this site there is a higher chance of success compared to the other sites.



Payload vs Launch Success

As we slide the payload higher the success reduces as we can see from this plot.





Section 5

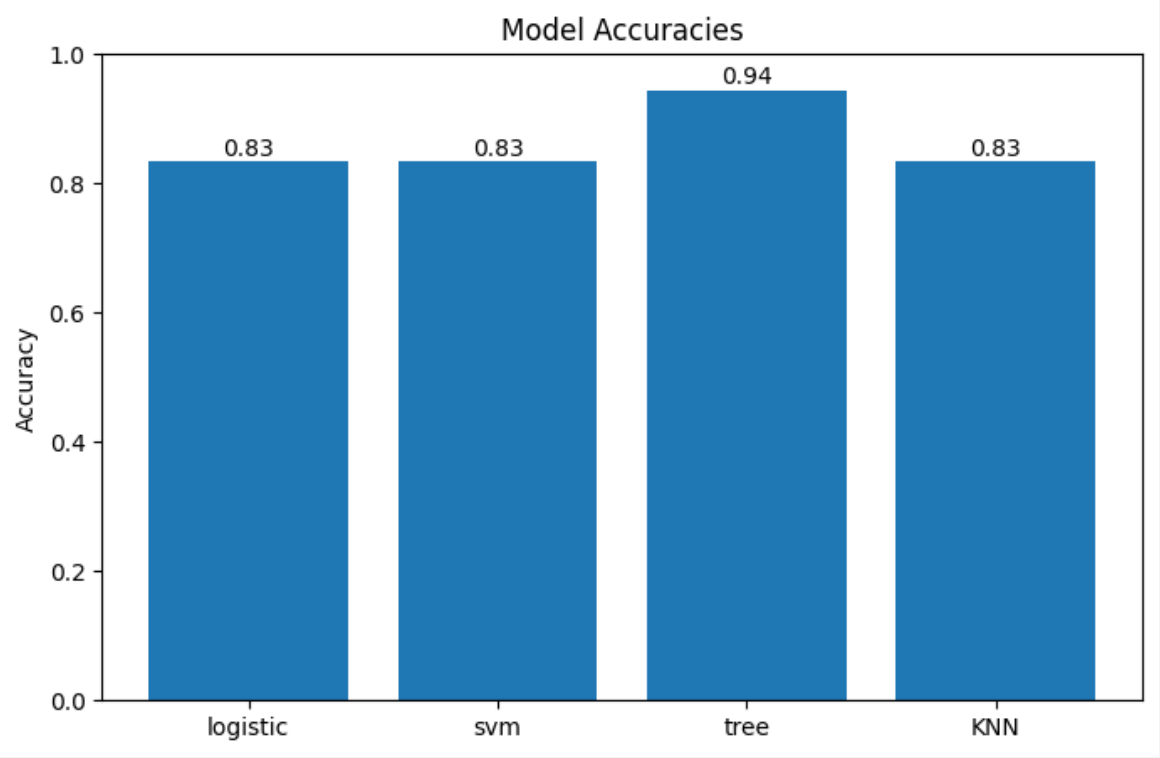
Predictive Analysis (Classification)

Classification Accuracy

Most of the models performed similar with the parameters we gave except decision tree.

The decision tree's accuracy is 94% and is far superior compared to our other models that are:

- logistic regression
- SVM
- KNN



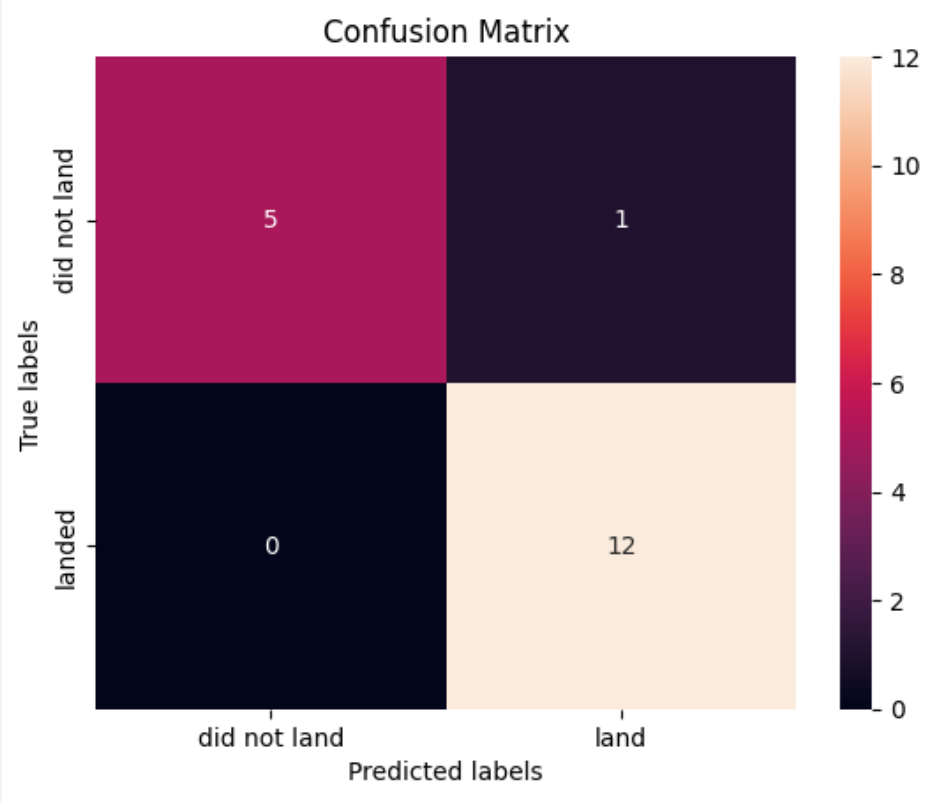
Confusion Matrix

This is the confusion matrix of Decision Tree:

The model predicted one out of the eighteen wrong the one it predicted wrong is supposed to not land thus is a false positive.

The model is accurate for all True Positive.

The accuracy of the model will be 17/18 which comes to 94% accuracy, which is good.



Conclusions

- As the flight number increases there is more success in CCAFS Launch site, which shows better progress.
- After 2015 there is a higher rate of success and better technology has been introduced probably
- Some of the orbit's have 100% success and some have 0% success which make orbits very important
- Payload is an important factor for some orbits like LEO, where success rate increases as payload increases
- Our model can predict with 94% accuracy whether the landing of stage 1 will be a success or not

Appendix

- Datasets used:
  - https://github.com/shcbswvc/SpaceX-Falcon9/tree/main/Data
- API used:
  - https://github.com/r-spacex/SpaceX-API
- Websites used for scrapping:
  - https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- Frameworks used
  - Pandas, Folium, Plotly, Dash, SQL, Sci-kit, Beautiful Soup, Matplotlib and Seaborn
- Data Base used:
  - https://github.com/shcbswvc/SpaceX-Falcon9/blob/main/Data/my_data1.db

Thank you!