IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

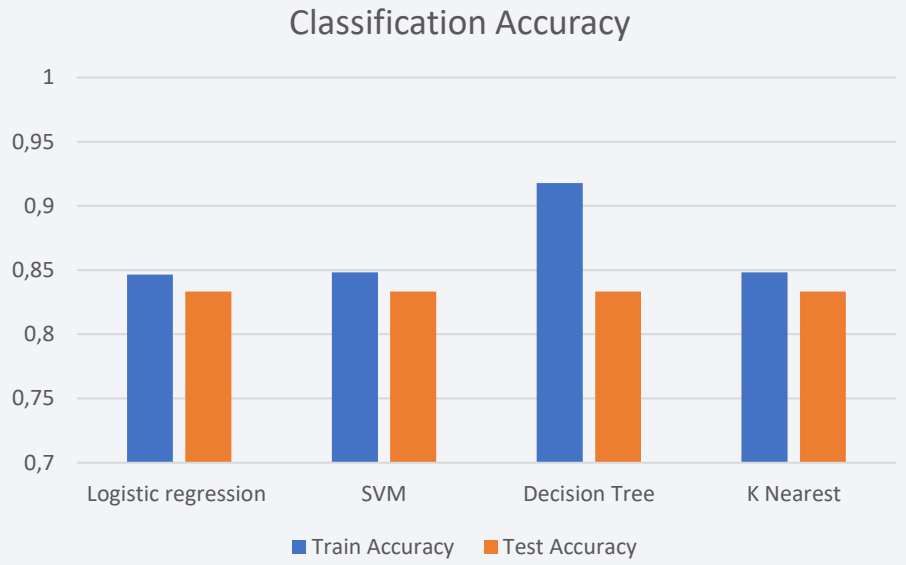
Juan Felipe Monsalvo Salazar
September 2023

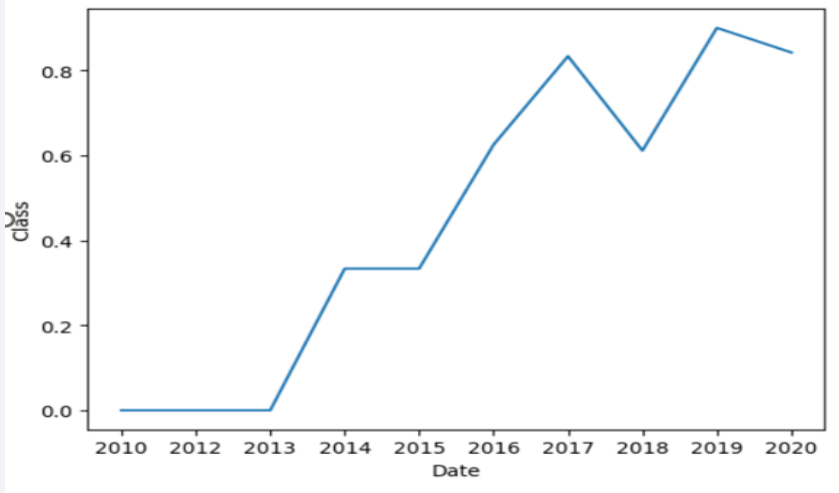
Outline

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

Executive Summary

- Summary of methodologies
  - Data collection from API and Wikipedia
  - Exploratory Data Analysis
    - SQL query
    - Relation between variables (scatter plots)
    - Proximities analysis using folium
  - Predictive analysis
- Summary of all results
  - Predictive accuracy of 83,3%.
  - Certain booster version and orbit type are more likely to have a successful launch





Introduction

- Project background and context
  - SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore if

- Problems you want to find answers
  - We want to determine if the first stage of the Falcon 9 rocket will land, if we are able to do that, we can determine the cost of a launch and this information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

Section 1

Methodology

Methodology

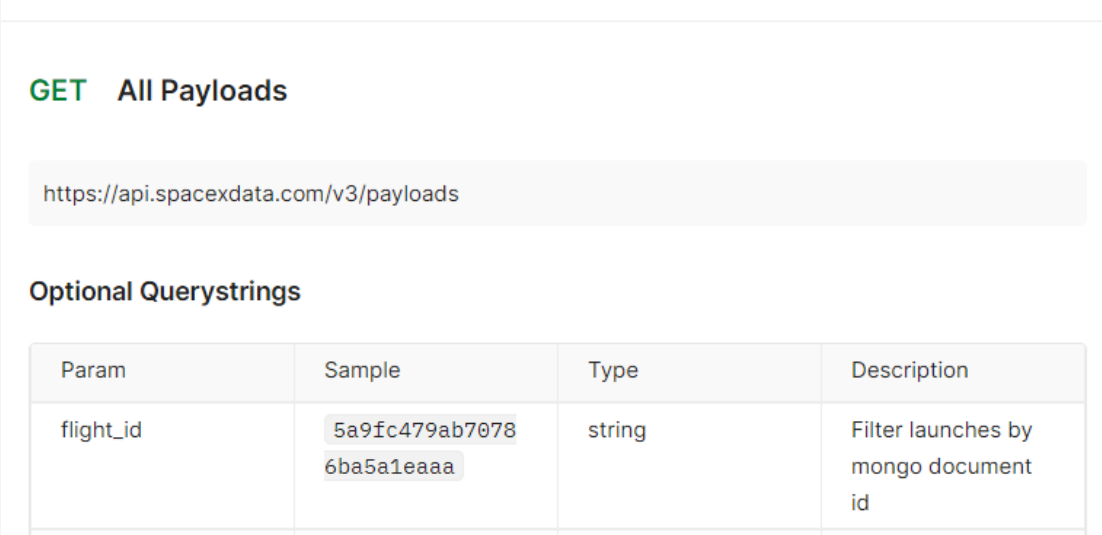
Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

The data collection process was realized in two different way:

- First, we collect information directly from the SpaceX API REST (https://docs.spacexdata.com/) using the GET method.
- Second, we use a web Scrapping method to get the information from the Wikipedia page of the Falcon 9 Rocket (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)



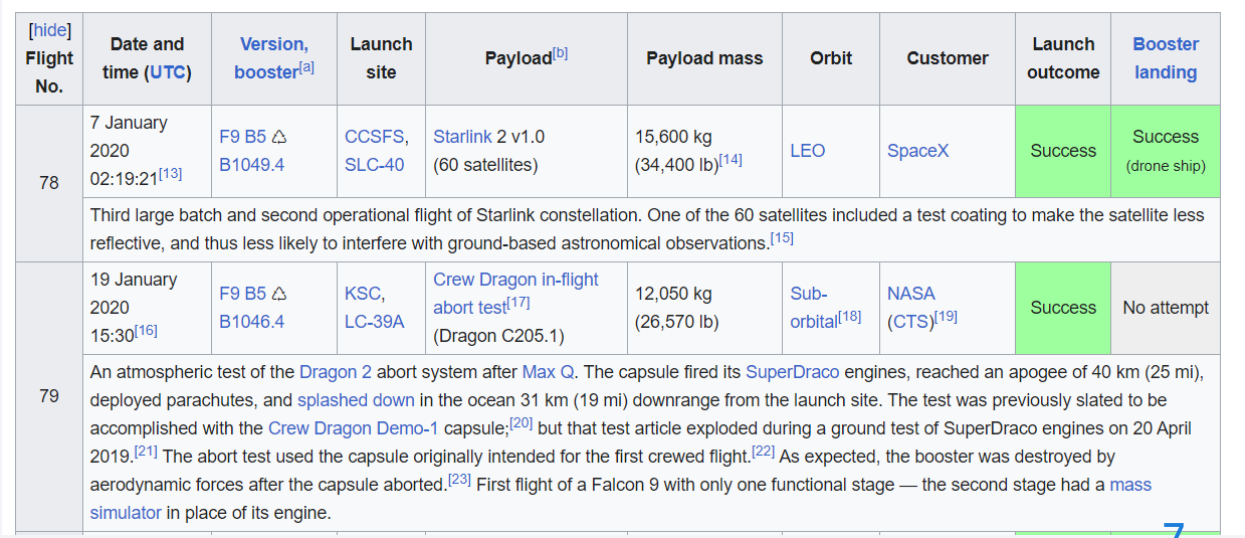
GET All Payloads

https://api.spacexdata.com/v3/payloads

Optional Querystrings

Param	Sample	Type	Description
flight_id	5a9fc479ab7078 6ba5a1eaaa	string	Filter launches by mongo document id

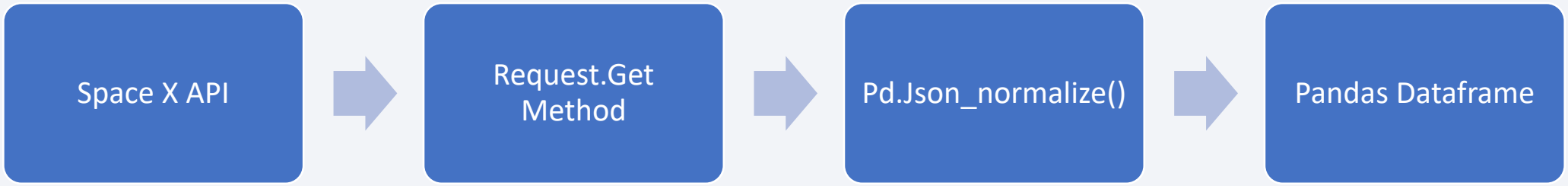
Space API



[hide] Flight No.	Date and time (UTC)	Version, booster[a]	Launch site	Payload[b]	Payload mass	Orbit	Customer	Launch outcome	Booster landing
78	7 January 2020 02:19:21[13]	F9 B5 △ B1049.4	CCSFS, SLC-40	Starlink 2 v1.0 (60 satellites)	15,600 kg (34,400 lb)[14]	LEO	SpaceX	Success	Success (drone ship)
	Third large batch and second operational flight of Starlink constellation. One of the 60 satellites included a test coating to make the satellite less reflective, and thus less likely to interfere with ground-based astronomical observations.[15]								
79	19 January 2020 15:30[16]	F9 B5 △ B1046.4	KSC, LC-39A	Crew Dragon in-flight abort test[17] (Dragon C205.1)	12,050 kg (26,570 lb)	Sub-orbital[18]	NASA (CTS)[19]	Success	No attempt
	An atmospheric test of the Dragon 2 abort system after Max Q. The capsule fired its SuperDraco engines, reached an apogee of 40 km (25 mi), deployed parachutes, and splashed down in the ocean 31 km (19 mi) downrange from the launch site. The test was previously slated to be accomplished with the Crew Dragon Demo-1 capsule;[20] but that test article exploded during a ground test of SuperDraco engines on 20 April 2019.[21] The abort test used the capsule originally intended for the first crewed flight.[22] As expected, the booster was destroyed by aerodynamic forces after the capsule aborted.[23] First flight of a Falcon 9 with only one functional stage — the second stage had a mass simulator in place of its engine.								

Wikipedia Page

Data Collection – SpaceX API



FlowChart

Different request were made to the Space X API in order to collect the necessary data to do the analysis. The methodology that was used was the one indicated on the flowchart.

- https://github.com/jmonsa13/coursera_ibm/blob/main/jupyter-labs-spacex-data-collection-api_P1.ipynb

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None

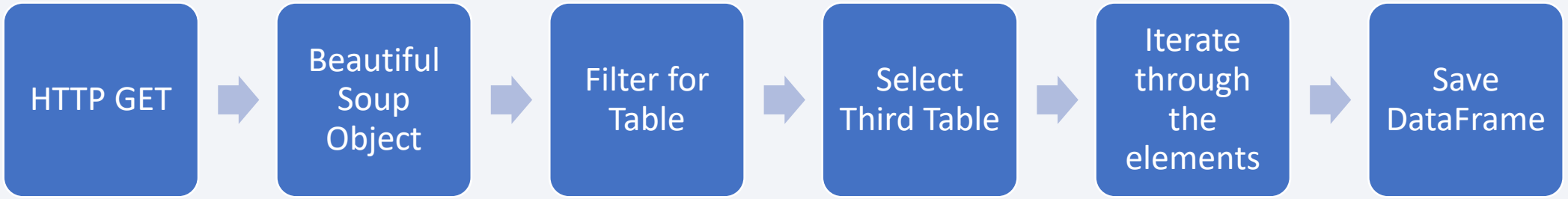
Pandas Dataframe

Data Collection - Scraping

The process for the scrapping of the table of falcon 9 launch was the following:

1. Save the HTML content of the URL using beautiful soup after a HTTP GET methods.
2. Filter the content saving just the table class.
3. Iterate through the <th> elements of the table and extracting the information of each columns
4. Saving the information as a Pandas DataFrame.

https://github.com/jmonsa13/coursera_ibm/blob/main/jupyter-labs-webscraping_P2.ipynb



Data Wrangling

We start importing the data from the data collection process and proceed to perform an EDA.

- We explore the number of launches from each launch site, identifying that Cape Canaveral Space Launch is the site with more launches.

```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

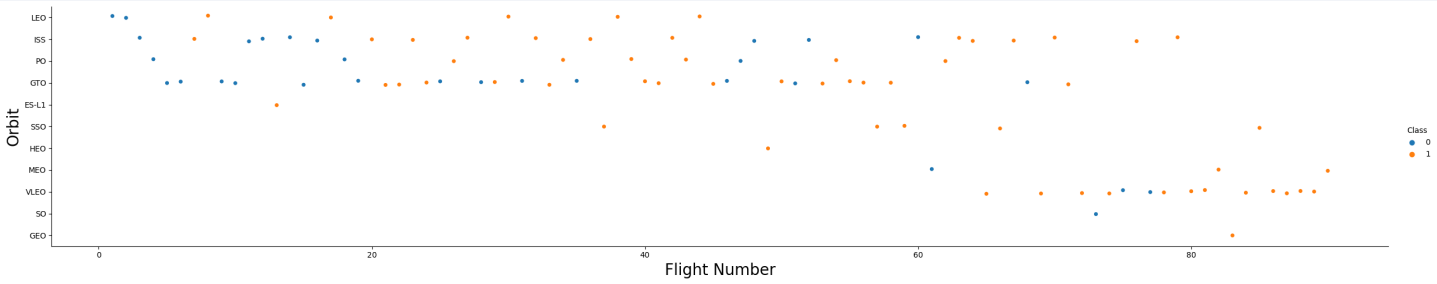
```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

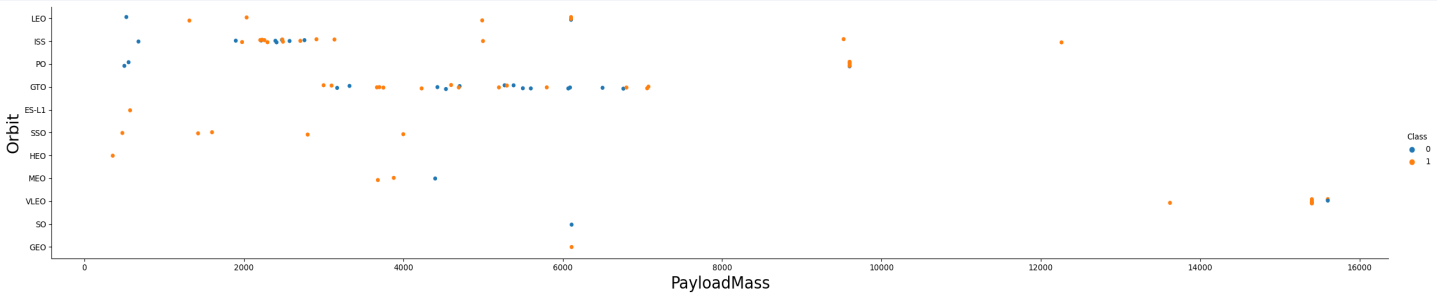
- We also explore the number of occurrence of each orbit and occurrence of mission output.
- Finally, we create the landing outcome label where 1 mean a successful launch and 0 mean otherw

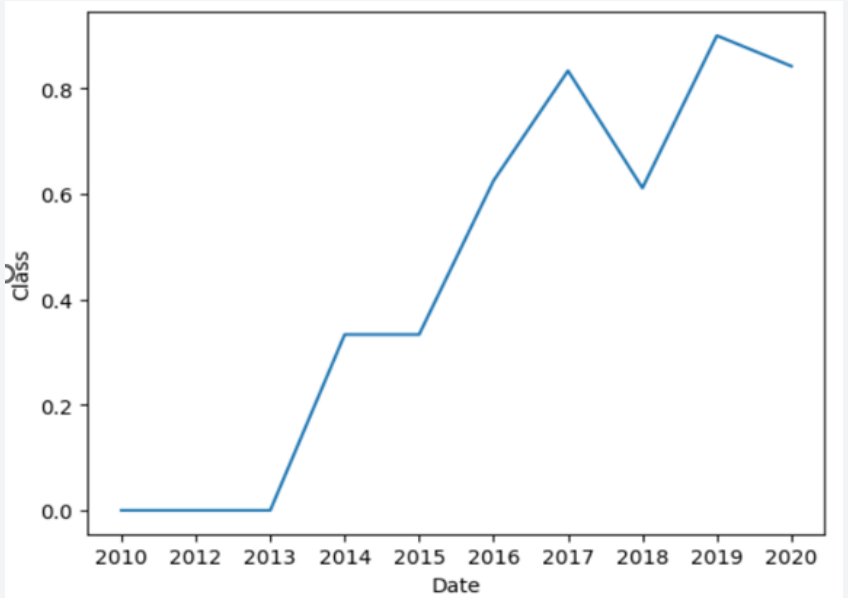
https://github.com/jmonsa13/coursera_ibm/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite_P3.ipynb

EDA with Data Visualization

We plot some scatter plot in order to visualize the relation between different variables as you can see below:







https://github.com/jmonsa13/coursera_ibm/blob/main/jupyter-labs-eda-dataviz_P5.ipynb

EDA with SQL

- Display the names of the unique launch sites using “Distinct” command.
- Get the 5 records where the launch site begin with “CCA” using “like” command.
- Sum the payload of all launches where customer was “NASA (CRS)” using “sum”
- Find the first successful landing in ground pad using the command “min”
- Use subquery as the one shows below to filter the data.

```
%sql select distinct("Booster_Version") from SPACEXTBL where "Landing_Outcome" = "Success (drone ship)" and
"PAYLOAD_MASS__KG_" between 4000 and 6000
```

https://github.com/jmonsa13/coursera_ibm/blob/main/jupyter-labs-eda-sql-coursera_sqllite_P4.ipynb

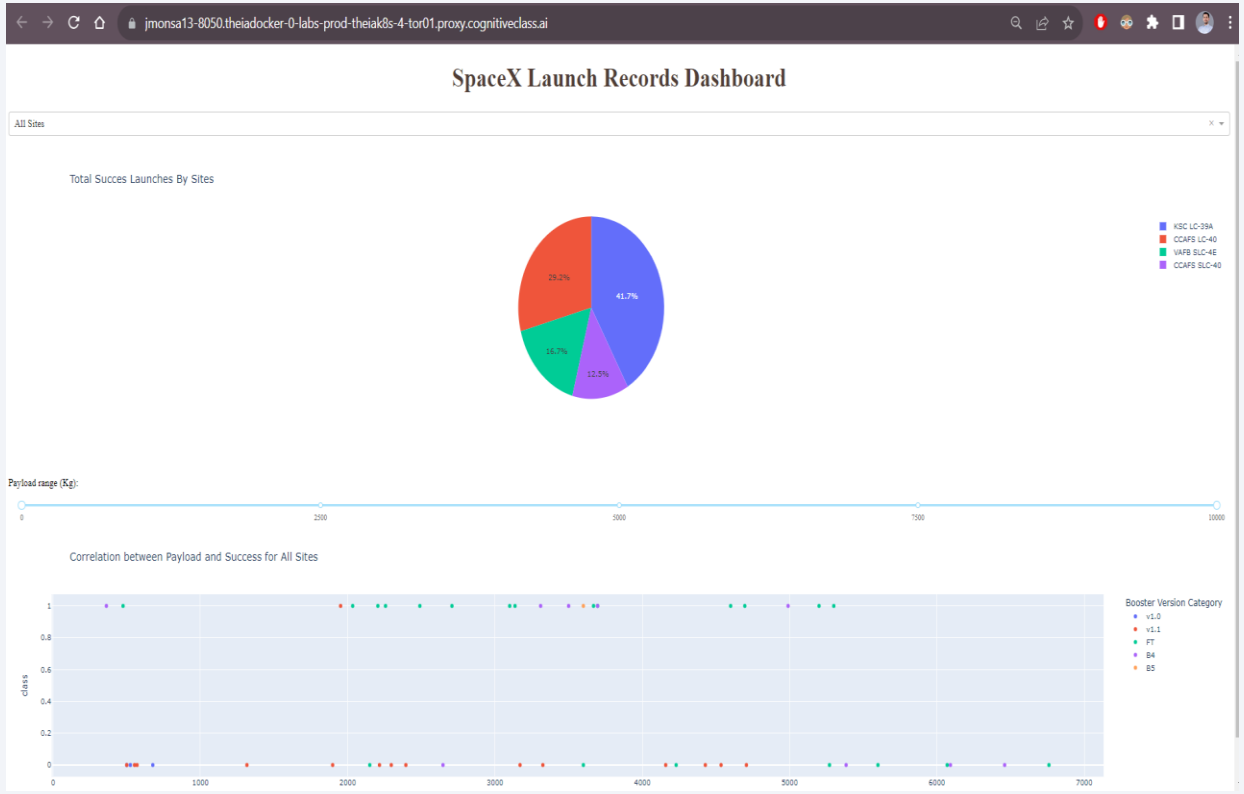
Build an Interactive Map with Folium

- We use markers in order to shows the successful and failed launches in each location.
- We also use lines to shows the distance from the launch site to highway, railway and coastline.
- The reason to do that was to show in a more convenient way information relative to the location of the launch sites.

https://github.com/jmonsa13/coursera_ibm/blob/main/lab_jupyter_launch_site_location_P6.ipynb

Build a Dashboard with Plotly Dash

- There is a pie chart that show the success ratio of the different locations.
- There is also a scatter plot that show the Payload Mass versus the Booster Version category. This scatter plot can be filtered using the dropdown menu and the range slider.
- We added some dropdown menu to interact with the location, we also added a range slider of the payload mass that is used to filter the plots.



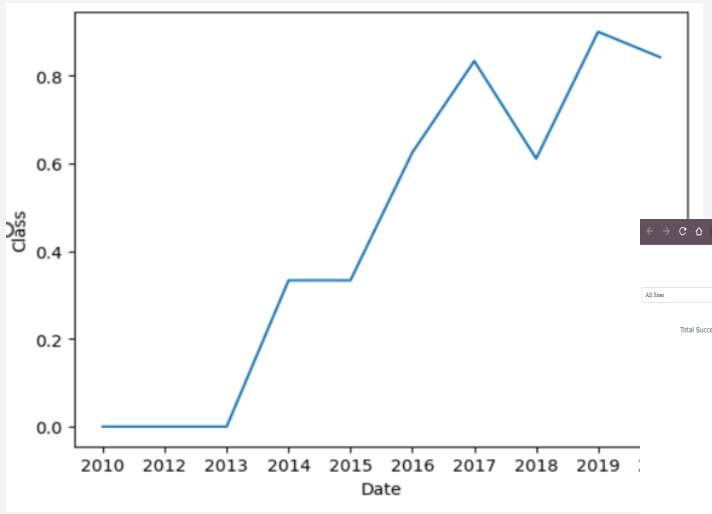
https://github.com/jmonsa13/coursera_ibm/blob/main/spacex_dash_app_P7.py

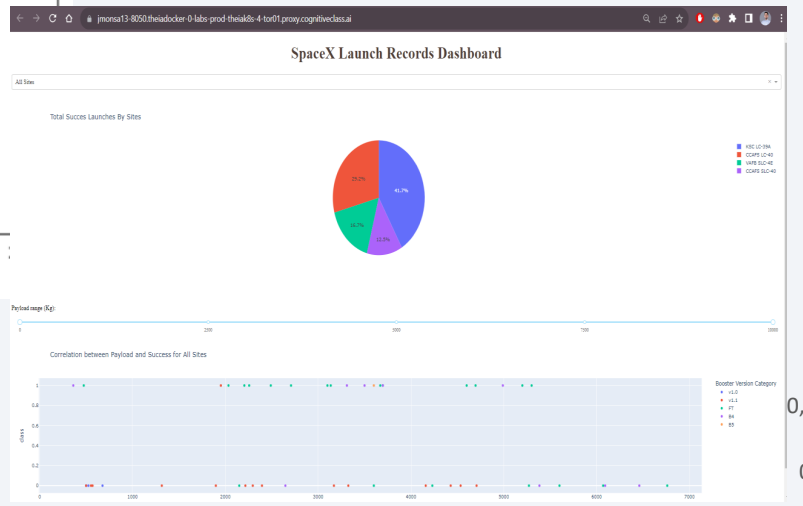
Predictive Analysis (Classification)

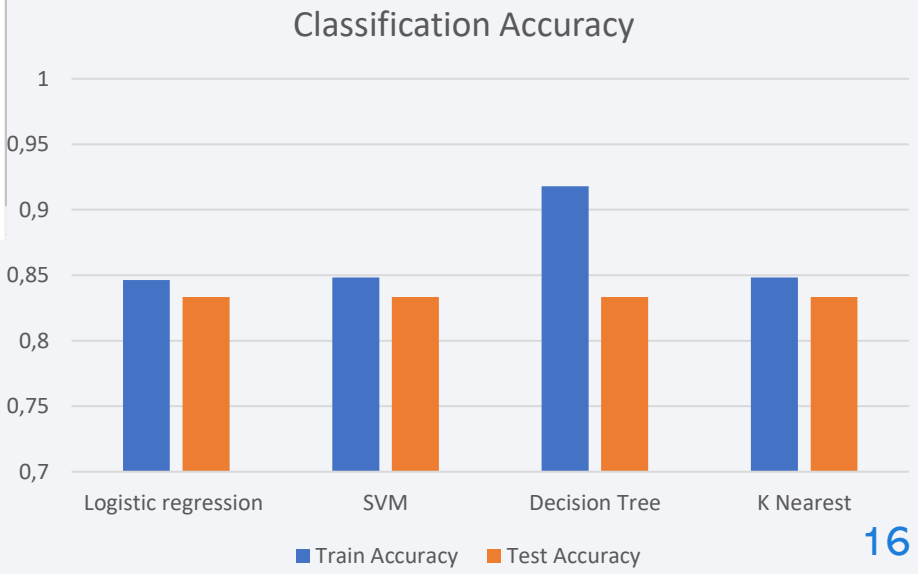
- First, we load the data and separate the target value (Y) from the features (x).
- After that we split the data into train and test data set using a split test ratio of 20%
- We create gridsearch object for each model and train the models.
- We analyses the performance of each model using the testing data set, reviewing the confusion matrix and the accuracy.
- We select the best model comparing the testing accuracy.

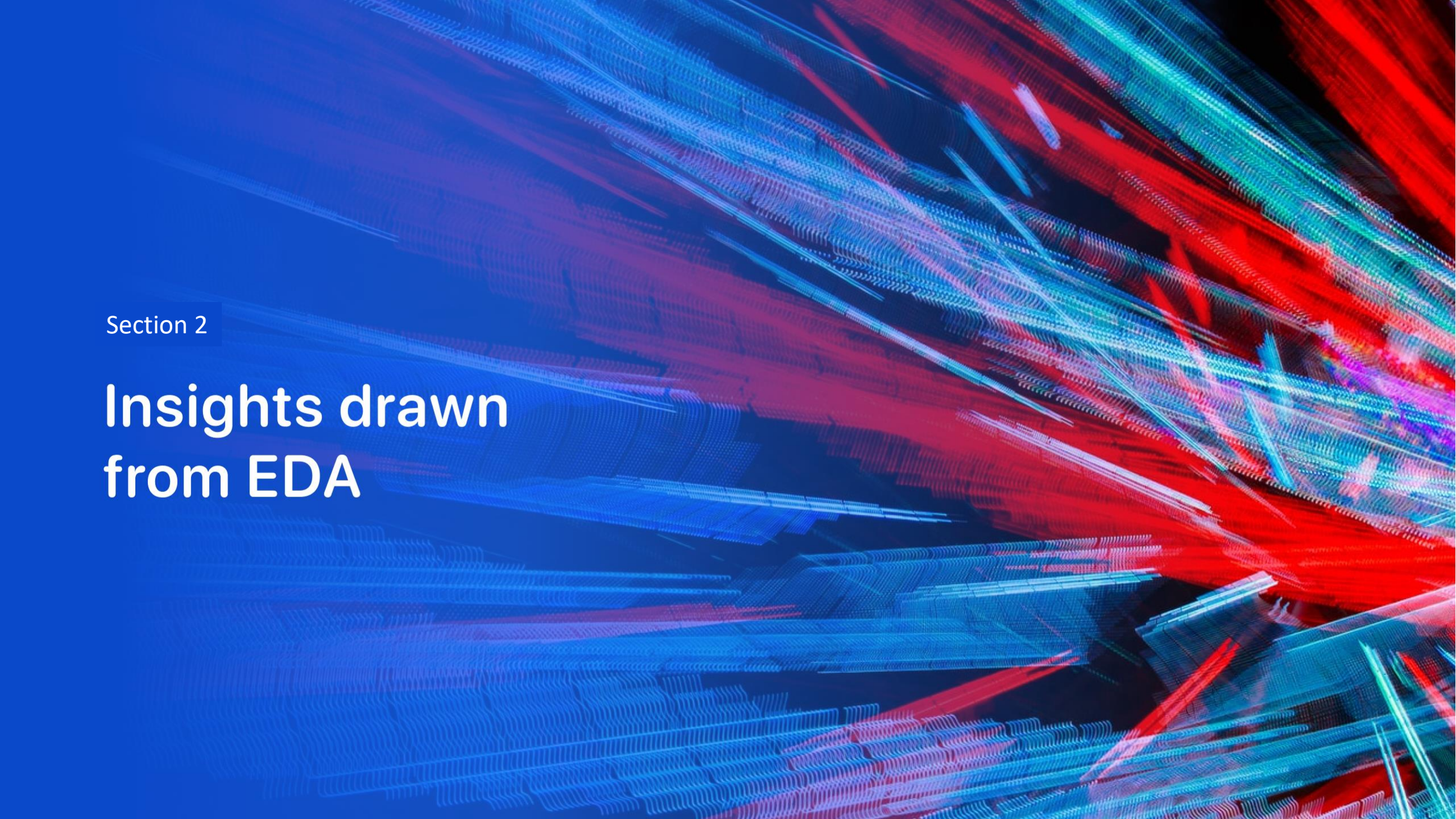
https://github.com/jmonsa13/coursera_ibm/blob/main/SpaceX_Machine_Learning_Prediction_Part_5_P8.ipynb

Results

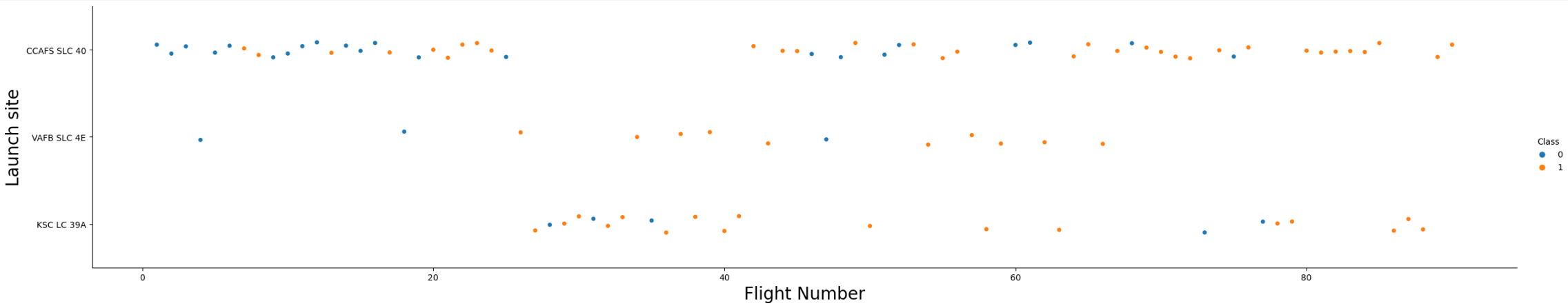

0.8
0.6
0.4
0.2
0.0
Class
2010
2012
2013
2014
2015
2016
2017
2018
2019
Date


SpaceX Launch Records Dashboard
All Sites
Total Succes Launches By Sites
Payload range (Kg):
Correlation between Payload and Success for All Sites
Booster Version Category


Classification Accuracy
1
0,95
0,9
0,85
0,8
0,75
0,7
Logistic regression
SVM
Decision Tree
K Nearest
Train Accuracy
Test Accuracy

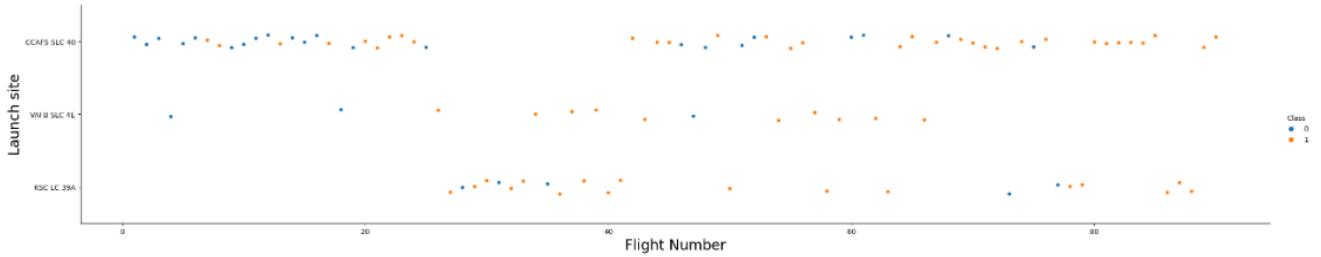

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



In [10]:

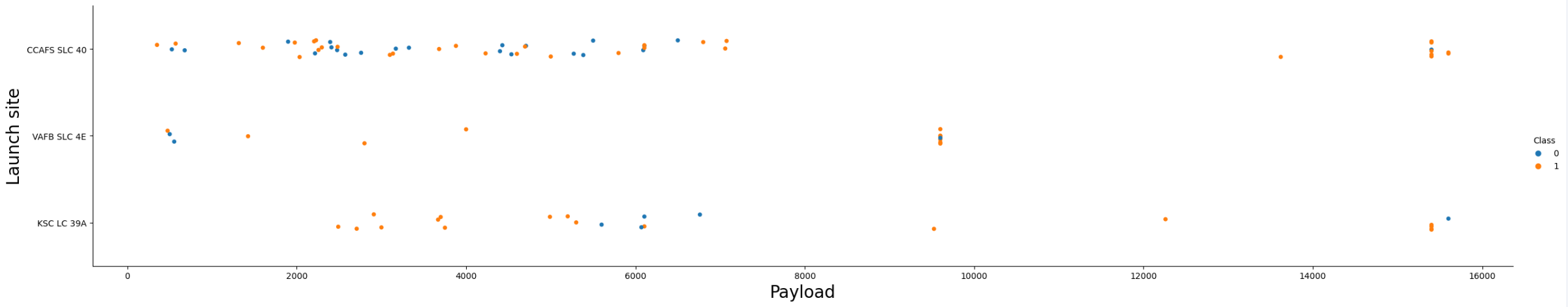
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class va
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch site",fontsize=20)
plt.show()
```



Now try to explain the patterns you found in the Flight Number vs. Launch Site scatter point plots.

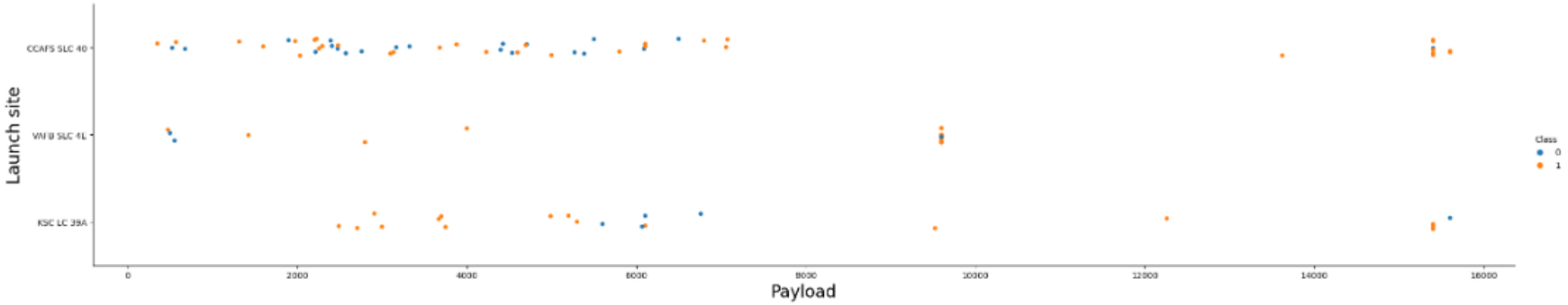
We can see that the success rate increase with the flight number especially in the CCAFS SLC 40

Payload vs. Launch Site



In [12]:

```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the cla
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("Payload",fontsize=20)
plt.ylabel("Launch site",fontsize=20)
plt.show()
```

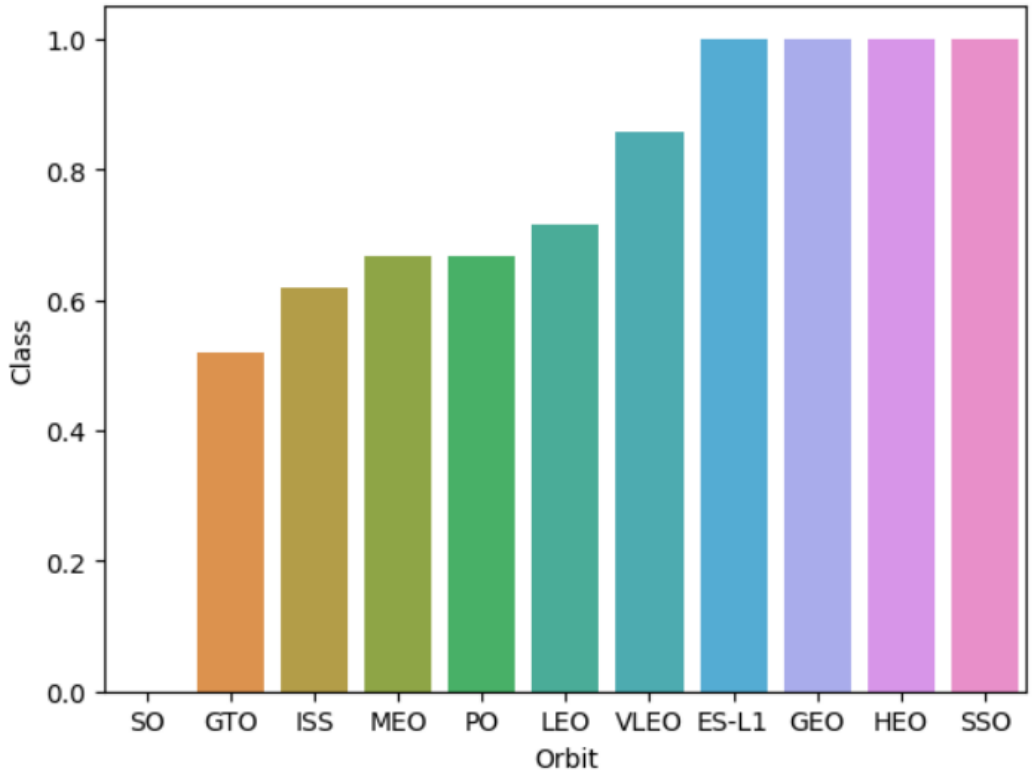


Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type

```
In [14]: sns.barplot(data=df_orbit, y='Class',x='Orbit')
```

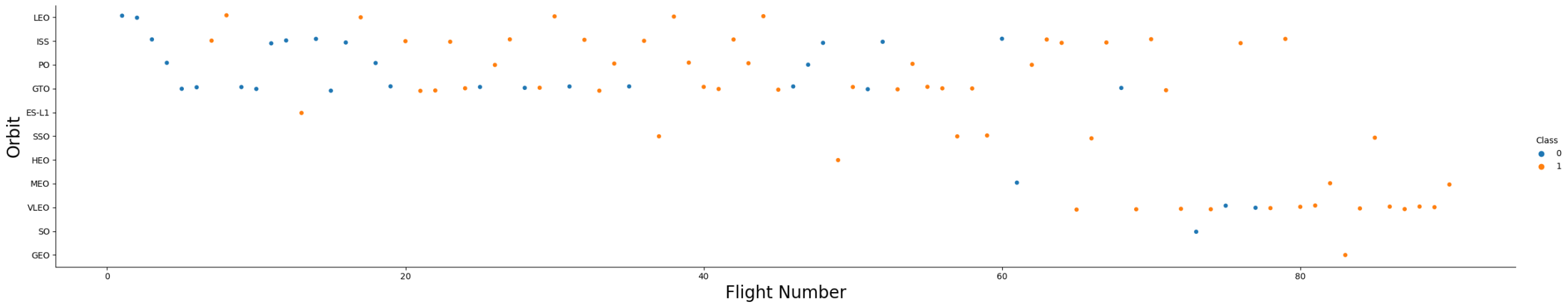
```
Out[14]: <AxesSubplot:xlabel='Orbit', ylabel='Class'>
```



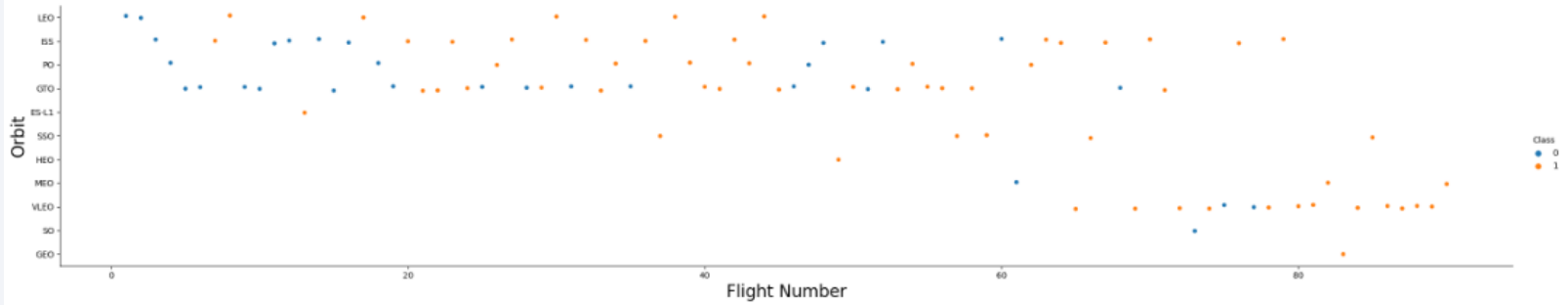
Analyze the ploted bar chart try to find which orbits have high sucess rate.

The orbits with the highest sucess are the ES-L1, GEO, HEO and SSO

Flight Number vs. Orbit Type

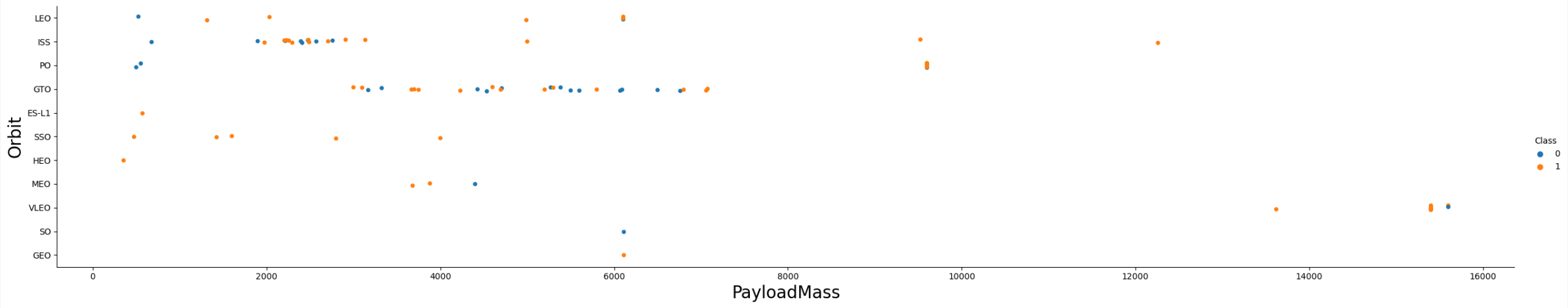



```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```

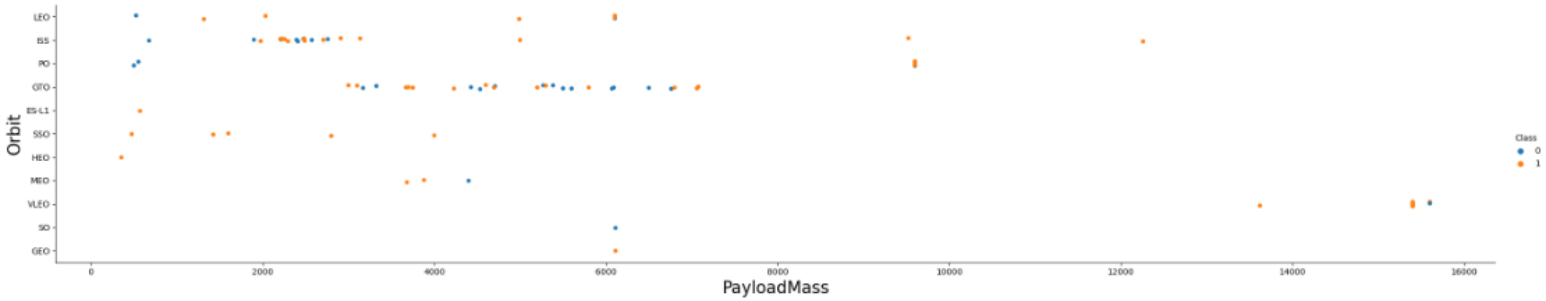


You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type




```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
sns.catplot(y="Orbit", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("PayloadMass",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



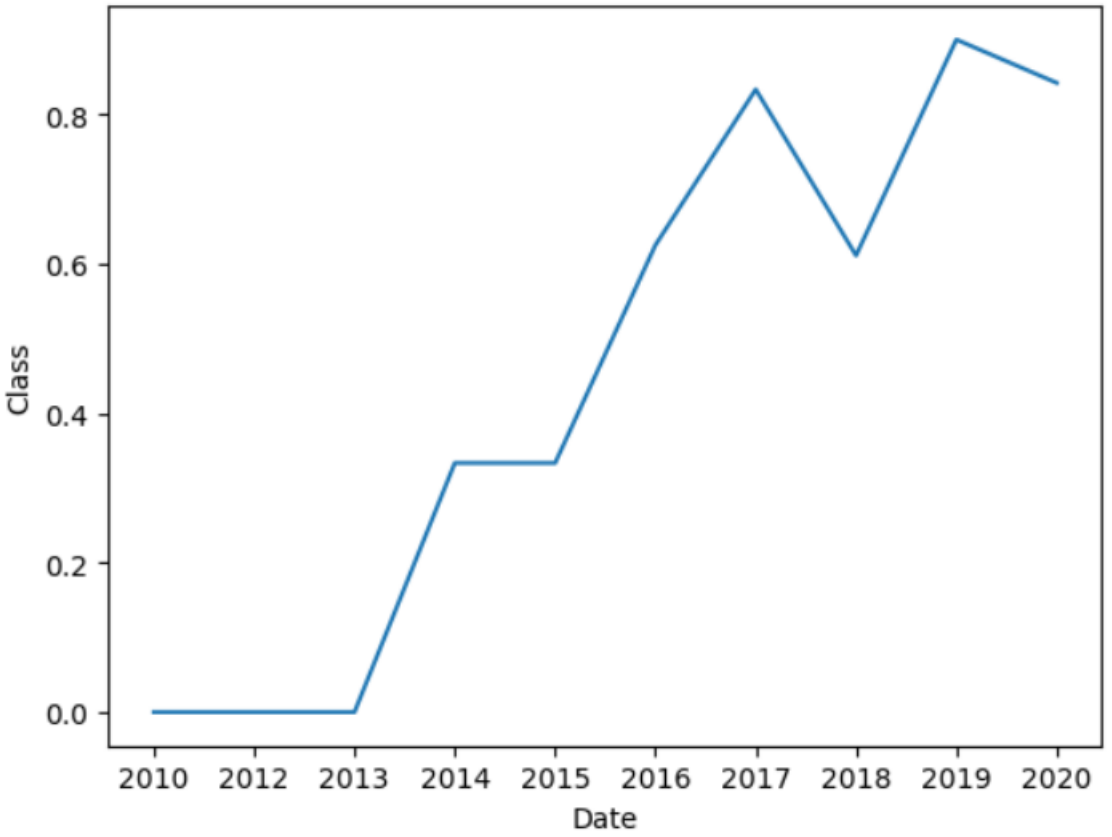
With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

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

```
# Plot a line chart with x axis to be the extracted year and y axis to be the success rate
sns.lineplot(data=df_sucess, x='Date', y='Class')
```

```
<AxesSubplot:xlabel='Date', ylabel='Class'>
```



you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

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

```
%sql select distinct("Launch_Site") from SPACEXTBL
```

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

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

We are using the distinct command from sql to return the unique launch sites.

Launch Site Names Begin with 'CCA'

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

```
%sql select * from SPACEXTBL where "Launch_Site" like "CCA%" limit 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
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

We filter the launch site using the like command and the input “CCA%”

Total Payload Mass

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

```
%sql select sum("PAYLOAD_MASS__KG_") from SPACEXTBL where "Customer" = "NASA (CRS)"
```

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

sum("PAYLOAD_MASS__KG_")
45596

We aggregate using sum the payload mass kg for all the boosters launched by NASA (CRS)

Average Payload Mass by F9 v1.1

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

```
%sql select avg("PAYLOAD_MASS__KG_") from SPACEXTBL where "Booster_Version" = "F9 v1.1"
```

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

avg("PAYLOAD_MASS__KG_")
2928.4

We aggregate using “avg” the payload mass kg for all the booster version = to F9 v1.1

First Successful Ground Landing Date

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

Hint:Use min function

```
%sql select min("Date") from SPACEXTBL where "Landing_Outcome" = "Success (ground pad)"
```

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

min("Date")
2015-12-22

We select the minimum date from the table where the outcome in ground pad was success in order to recover the first successful landing.

Successful Drone Ship Landing with Payload between 4000 and 6000

We combine the distinct command with a specific filter to obtain the result shows below

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

```
%sql select distinct("Booster_Version") from SPACEXTBL where "Landing_Outcome" = "Success (drone ship)" and "PAYLOAD_MASS__K
```

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

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

%sql select distinct("Booster_Version") from SPACEXTBL where "Landing_Outcome" = "Success (drone ship)" and "PAYLOAD_MASS__KG_" between 4000 and 6000

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%sql select  "Mission_Outcome", count(*) as "Total" from SPACEXTBL group by "Mission_Outcome" order by "Mission_Outcome"
```

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

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

We count the total number of successful and failure mission outcomes using a group by function on the “Mission_Outcome” column

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
%sql select "Booster_Version" from SPACEXTBL where "PAYLOAD_MASS__KG_" = (select MAX("PAYLOAD_MASS__KG_") from SPACEXTBL)
```

```
 * 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

We select all the booster version who payload was equal to the maximum payload, for that we use a subquery in order to find the maximum payload.

2015 Launch Records

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, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

```
%sql select substr(Date,6,2) as "Month", "Landing_Outcome", "Booster_Version", "Launch_Site" from SPACEXTBL where "Landing_(
```

```
* 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

%sql select substr(Date,6,2) as "Month", "Landing_Outcome", "Booster_Version", "Launch_Site" from SPACEXTBL where "Landing_Outcome" = "Failure (drone ship)" and substr(Date,1,4)="2015";

The failure landing in drone shipping during the year 2015 happens in April and October.

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

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 "Landing_Outcome", count(*) as 'Quantity' from SPACEXTBL where "Date" between "2010-06-04" and "2017-03-20" grou
```

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

Landing_Outcome	Quantity
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

%sql select "Landing_Outcome", count(*) as 'Quantity' from SPACEXTBL where "Date" between "2010-06-04" and "2017-03-20" group by "Landing_Outcome" order by "Quantity" desc

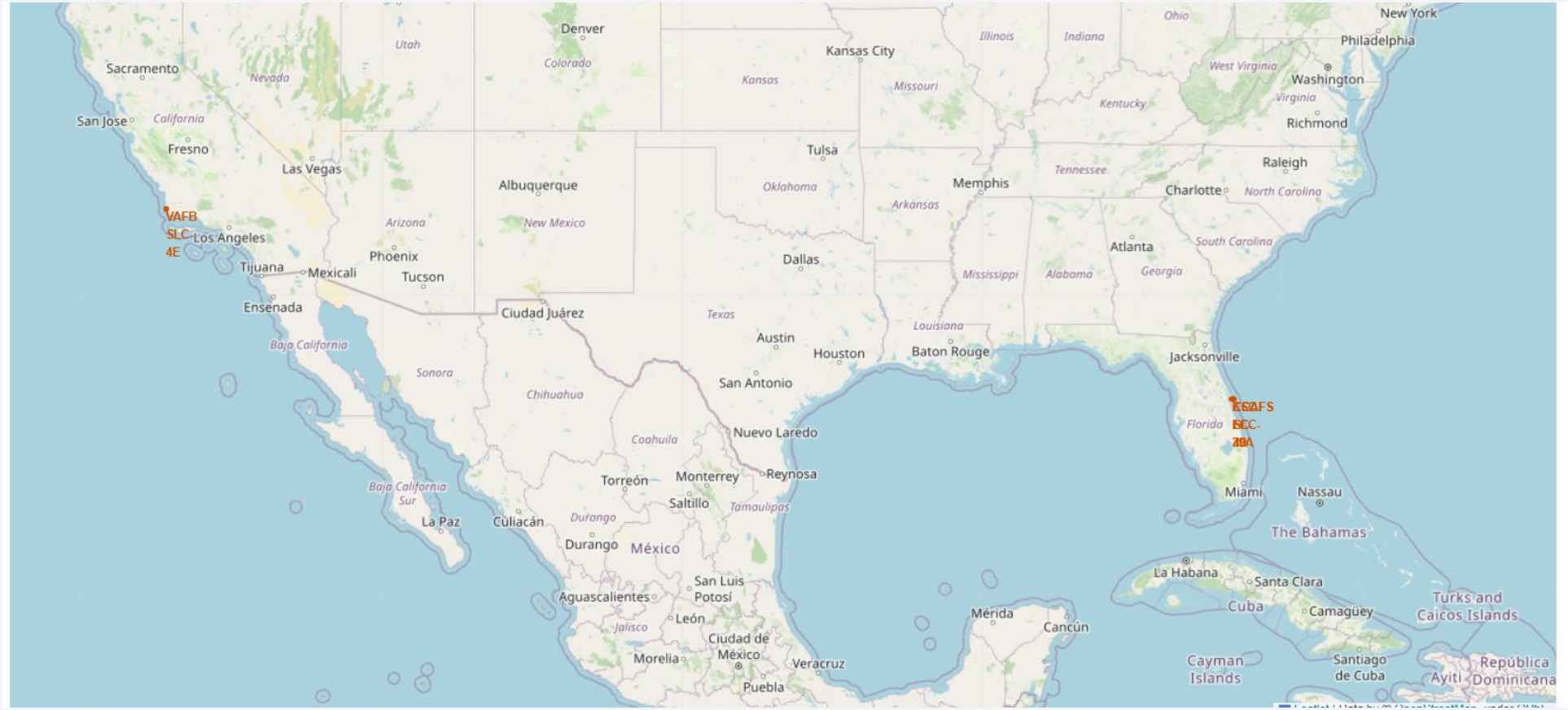
We filter by the date and group by the result using the landing outcome column. We also apply the count method and order the result.

Section 3

Launch Sites Proximities Analysis

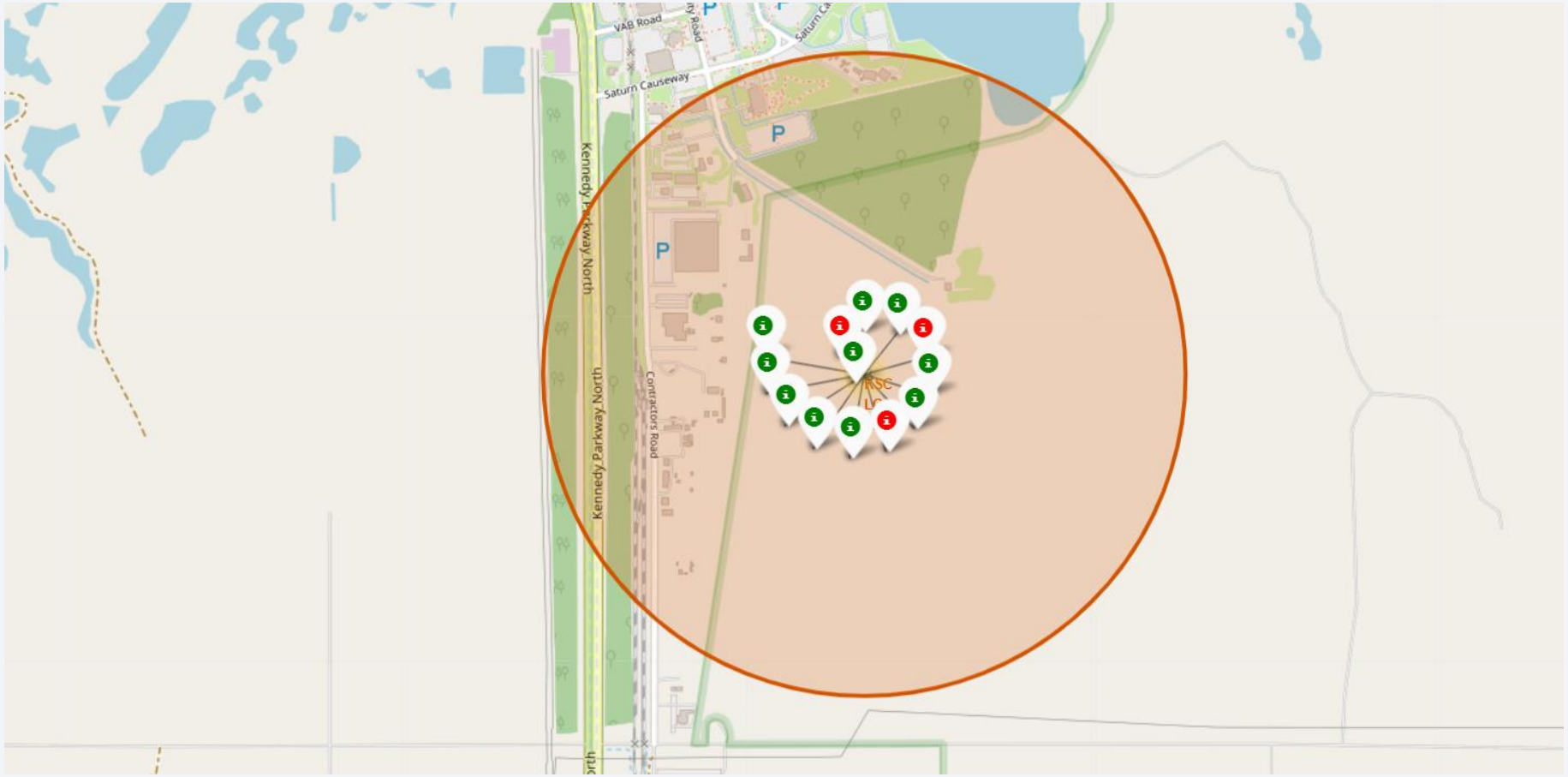
Launch site location on global map

It can be clearly seen that the launch site location are near a coastline and south of the united state (near the equator line).



Success/failed launches for each site

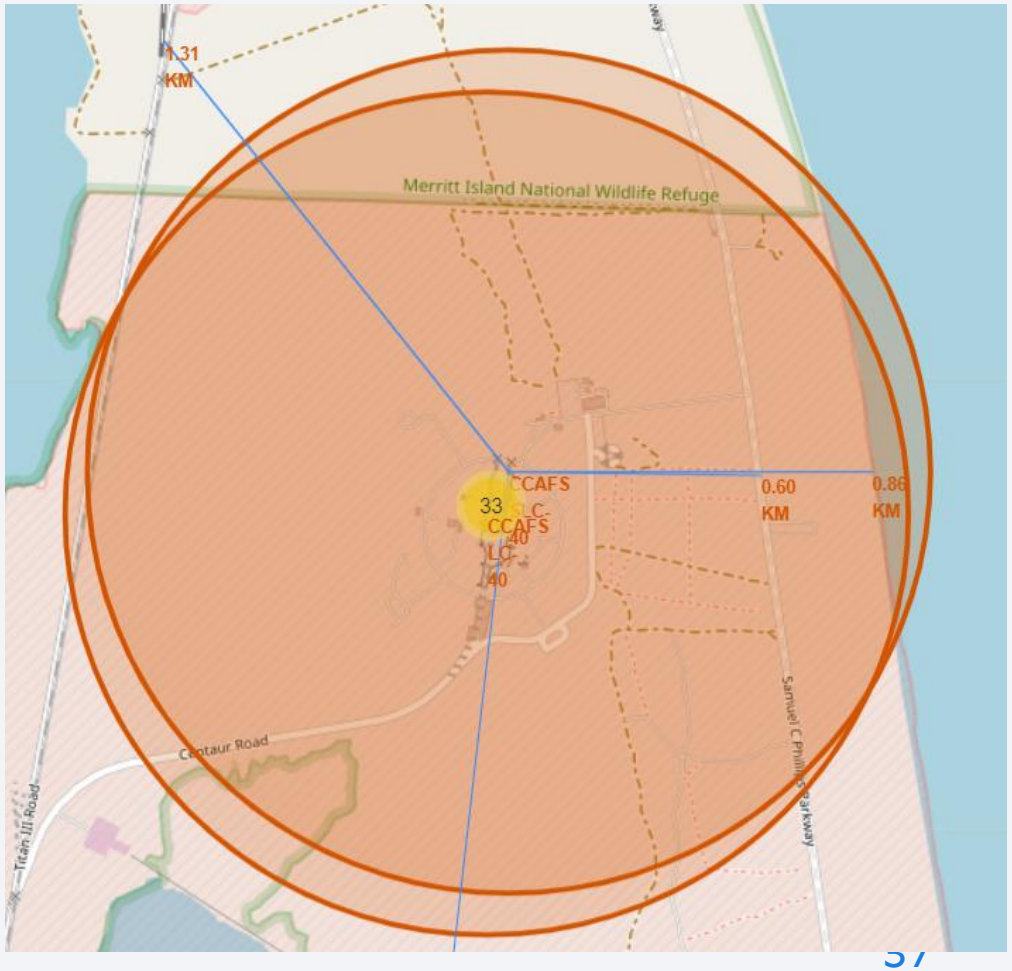
The launch site showed below has more success launches that failed.

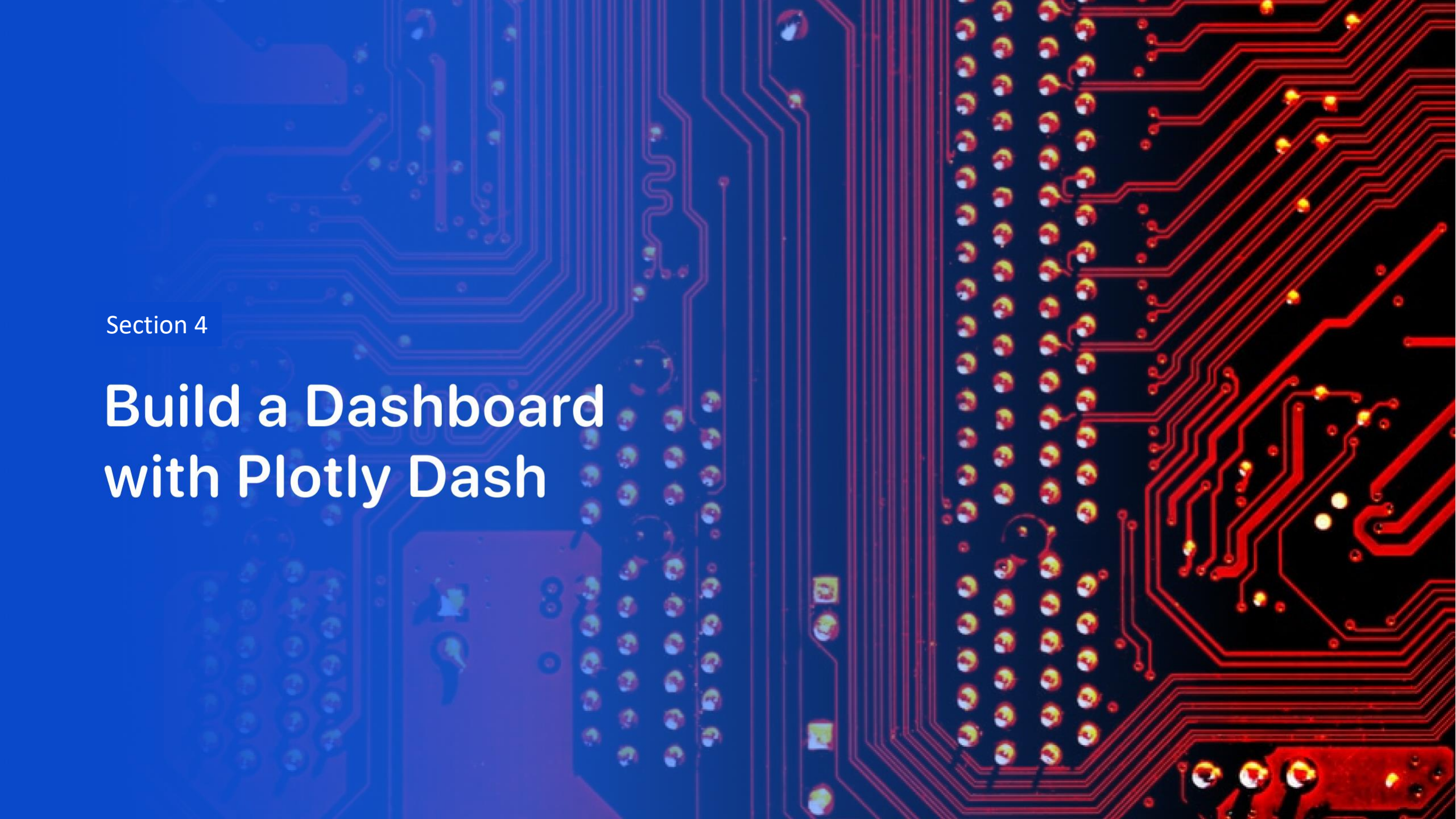


Launch site distance from coastline, city, highway…etc

We can appreciate that the launch sites are near coastline to reduce the impact of any catastrophic launches. They are also near highway and railway in order to facilitate the construction and transportation of the necessary elements for the mission.

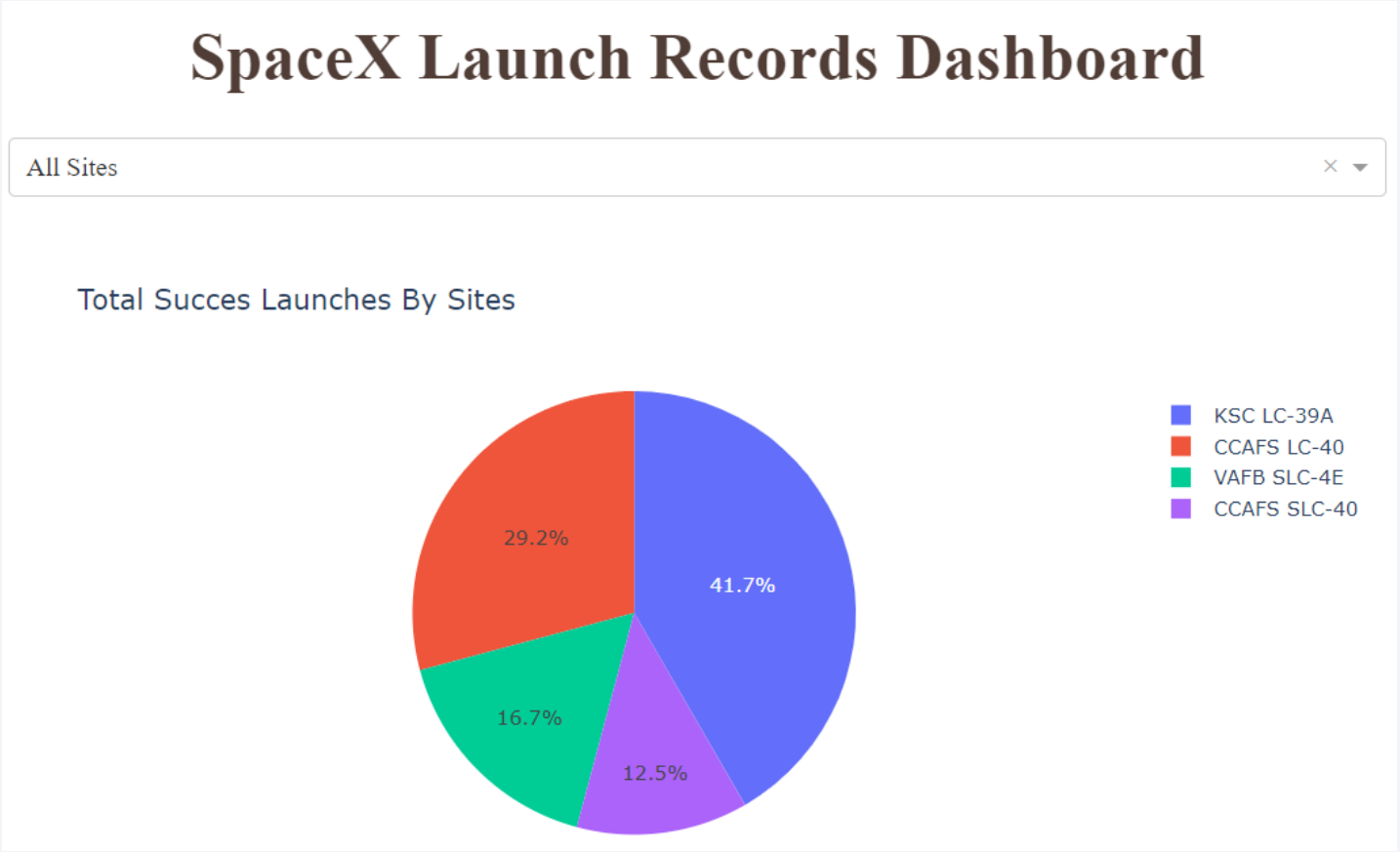
They are also far away from cities to avoid any impact on them.




Section 4
Build a Dashboard with Plotly Dash

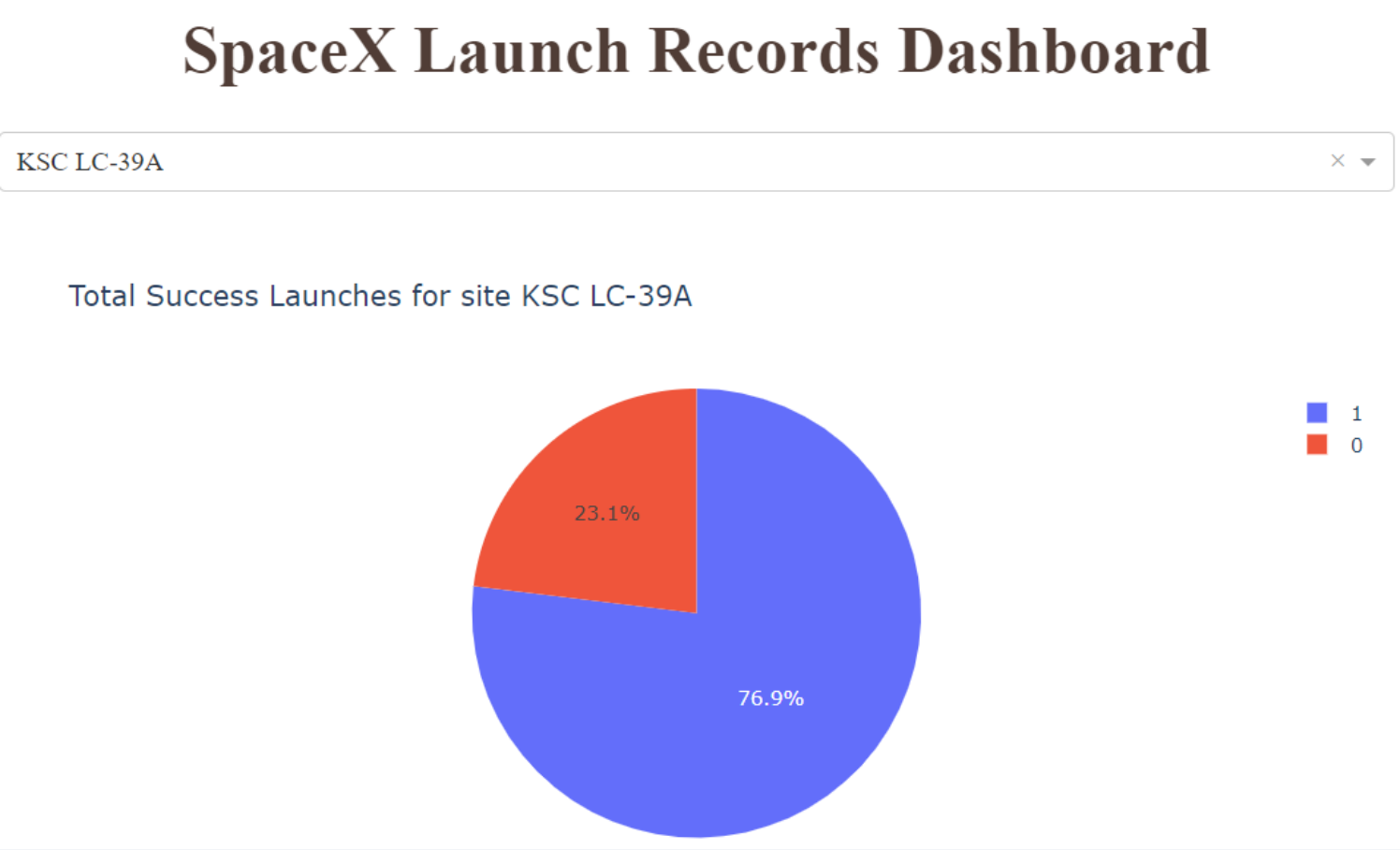
Total success launches by sites

We can see that the biggest success ratio of launch is 41,7% and the site is the KSC LC-39A.



Launch success ratio for KSC LC-39A

The site (KSC LC-39A) with the biggest launch success ratio of all the sites as a launch success ratio of 76,9% which seem to be very good.



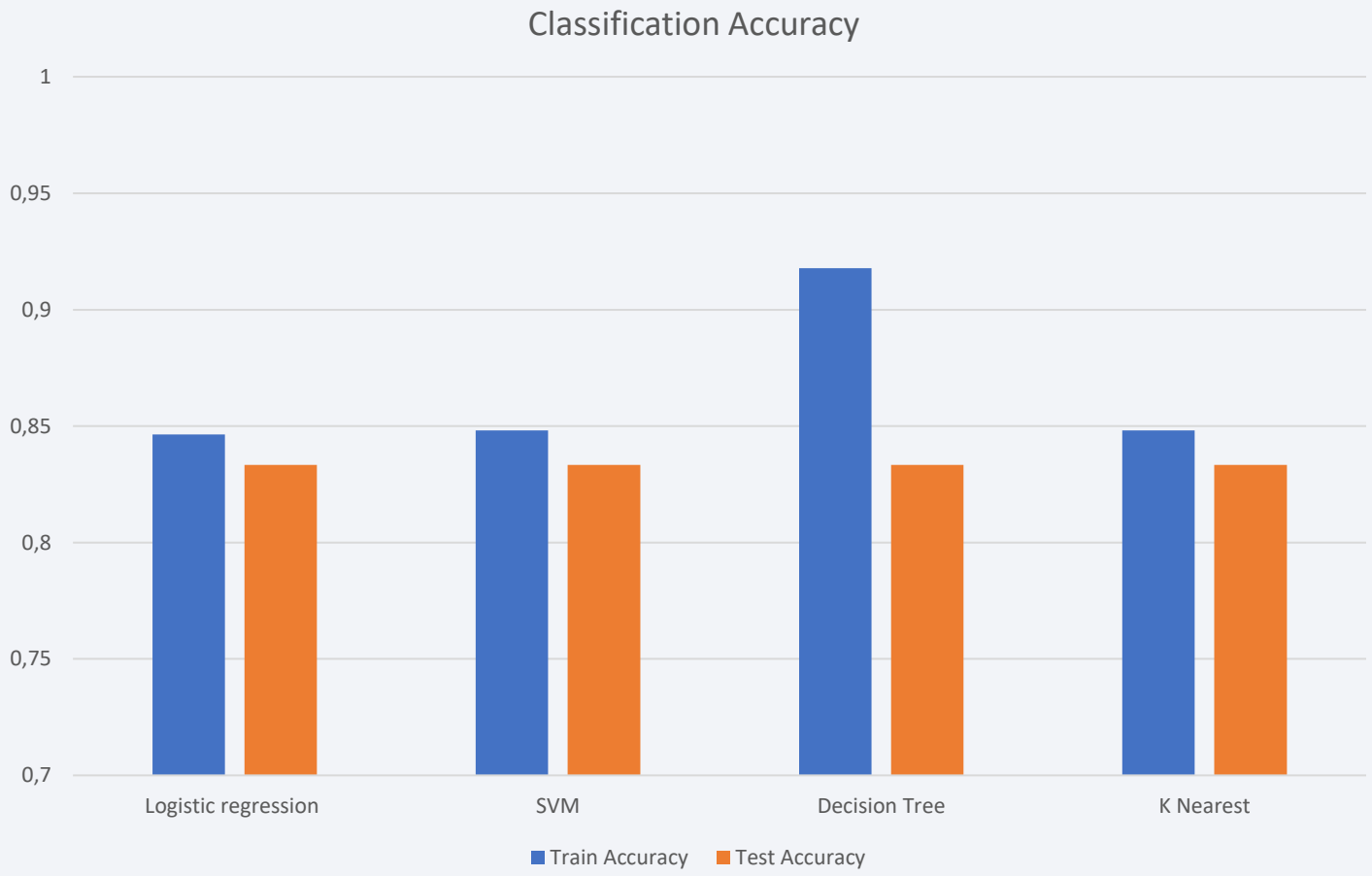
Payload vs Launch Outcomes all sites

- The launches with the biggest payload (>6000 kg) tend to fail.
- The booster version FT has a really good rate of success.
- Booster version V1.1 has a very high failure rate.

Section 5

Predictive Analysis (Classification)

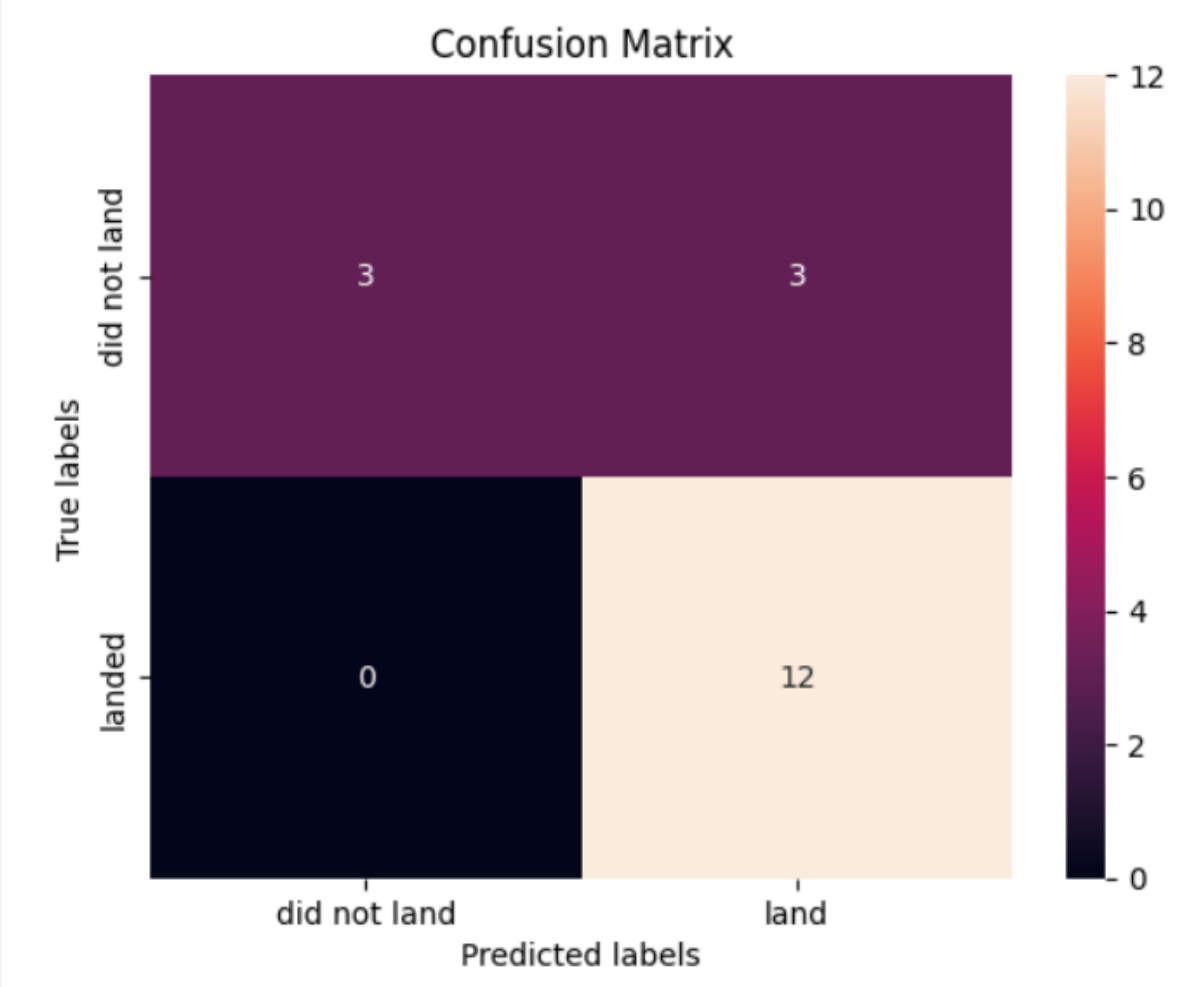
Classification Accuracy



- The model with the highest train accuracy is the Decision Tree.
- All the models has the same test accuracy therefore any model is suitable to be used.

Confusion Matrix

- Any model is the best performing model since all the models has the same test accuracy.
- The model can distinguish between the different classes, the major problem is false positives.



Conclusions

- We can conclude that the success ratio of the launch has been improving over time with the launch of newest booster.
- With the predictive models created we can assert with at 83,3% accuracy the result of the mission previous the launch.
- We can see that certain variable has a greater impact on the outcome of the mission as the orbit type, booster version and payload mass

Appendix

- This is the github repository with all the information and notebook https://github.com/jmonsa13/coursera_ibm/tree/main

Thank you!