Winning Space Race with Data Science

Qing Li
2023-03-13

Outline

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

Executive Summary

- We have collected data from public SpaceX API and by scrapping SpaceX Wikipedia page and used Beautiful soup library for it.
- Created labels column ‘Class’ which denotes all the successful landings.
- We performed EDA (Exploratory Data Analysis) using SQLusing the concepts of magic sql.
- We explored some queries using group by, aggregate functions and so on.
- We created scatter charts as well as bar charts using matplotlib to analyze the relationship among payload mass, flight number etc..
- After than we created an interactive dashboard with Plotty Dash. We also created an interactive map with Plotty Folium
- In the end we build several models using GridSearchCV from sklearn, training, testing and in the end comparing all the models for the best accuracy.
- The main findings are,
    - Payloadmass and flight number plays an important role in increasing the success rate
    - It might be a good idea to launch from Site KSC LC-39A
    - If the model predicts a success landing outcome then chances are it will be successful

Introduction

- The purpose of this project is to correctly predict if the Falcon 9 first stage (from SpaceX) will land successfully. The good prediction would be of good use for an alternative company Space Y.

- The main problems are:
  - What is the price of each launch?
  - Whether SpaceX will reuse the first stage for each launch?

Section 1

Methodology

Methodology

Executive Summary

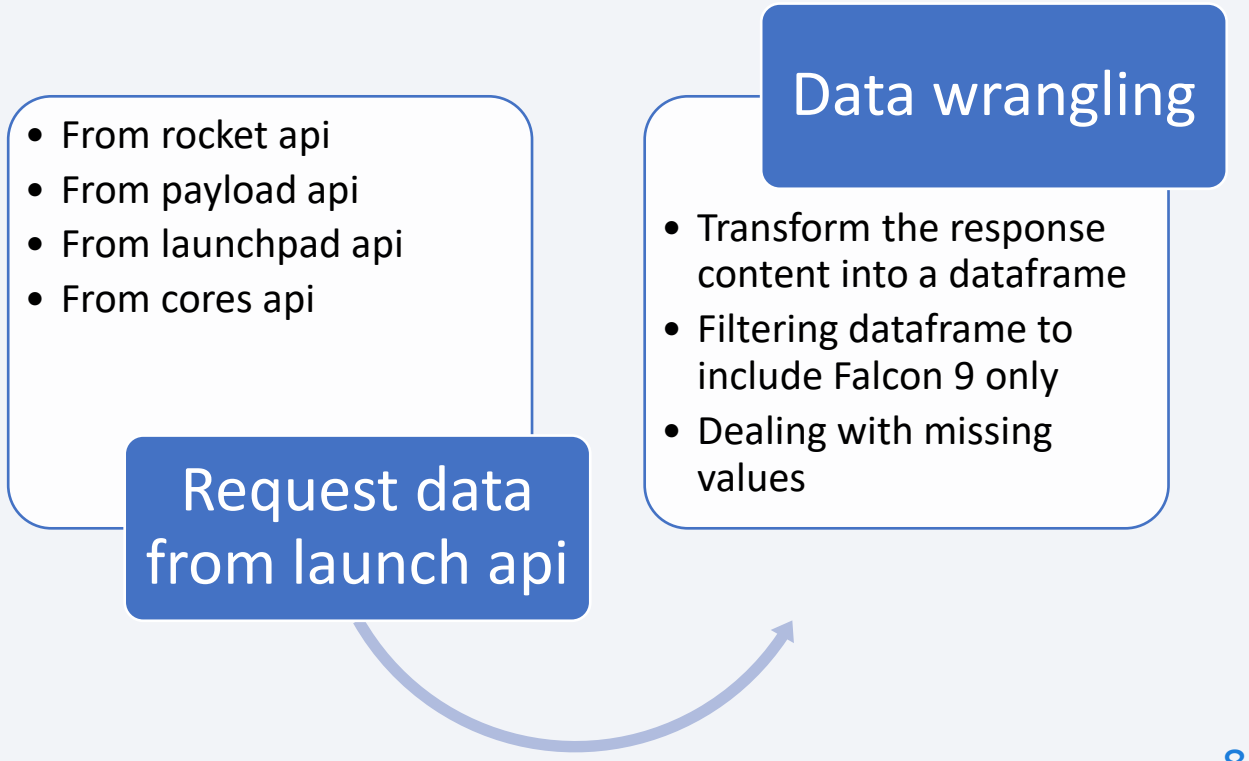
- Data collection methodology:
  - The data is collected using either of below 2 methods,
    - Via the request to the SpaceX API
    - Via web scraping from a Wikipedia page
- Perform data wrangling
  - Process the data using EDA to convert the outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Perform exploratory Data Analysis and determine Training Labels
  - Find the method performs best using test data among SVM, Classification Trees and Logistic Regression

Data Collection

- Data was collected via API and web scraping

Data Collection – SpaceX API

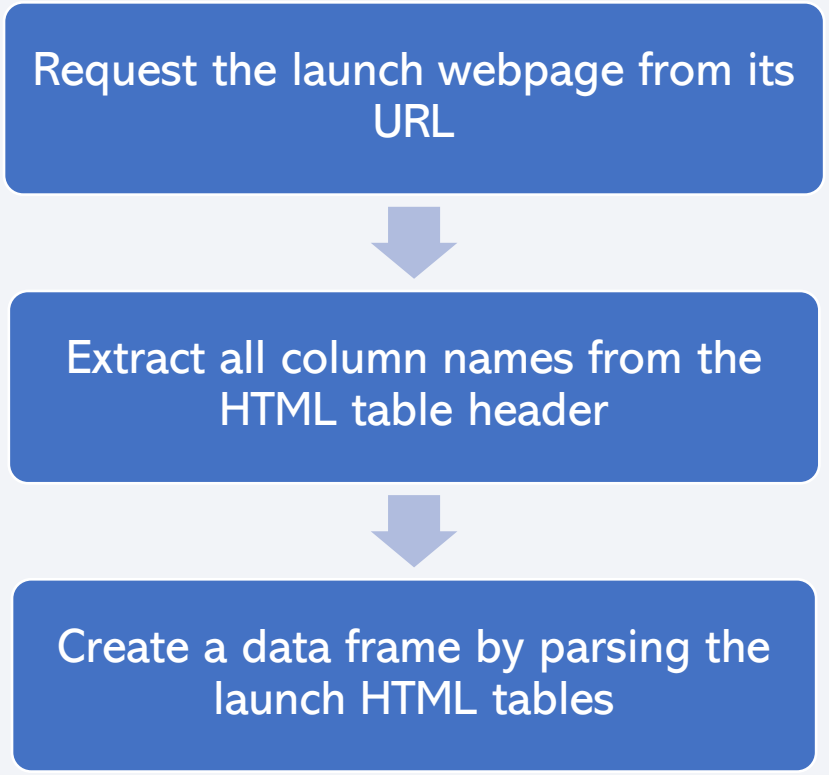
- Request data from launch api
  - From rocket api
  - From payload api
  - From launchpad api
  - From cores api
- Data wrangling
  - Transform the response content into a dataframe
  - Filtering dataframe to include Falcon 9 only
  - Dealing with missing values
- GitHub Link



Data Collection - Scraping

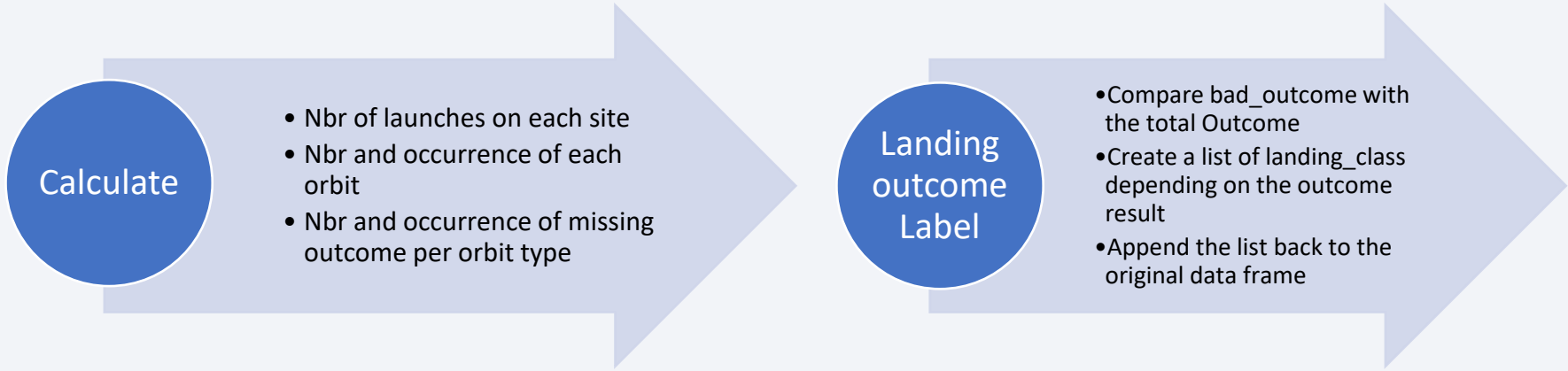
- Request the launch webpage from its URL
- Extract all column names from the HTML table header
- Create a data frame by parsing the launch HTML tables

- GitHub Link



Data Wrangling

- From the dataset we,
  - Calculate the number and occurrence or missing outcome per orbit type;
  - From the result we further filter out a set of “bad outcomes”;
  - Append a column of ‘Class’ by comparing the “bad outcomes” with the total “Outcome”



- GitHub Link

EDA with Data Visualization

- Scatter charts were plotted for
  - Flight Number and Pay load Mass (kg) with an overlay of outcome
  - Flight Number and Launch Site with an overlay of outcome
  - Pay load Mass (kg) and Launch Site with an overlay of outcome
  - Flight Number and Orbit type with an overlay of outcome
  - Pay load Mass (kg) and Orbit type with an overlay of outcome
- The reason for using the scatter charts is because we need to compare the outcome between the elements on Y axis (e.g. Launch site) along the same trend on X axis
- Bar chart was used between success rate and orbit type as we only need to focus on one variable and how it affects the outcome
- Line chart was used between the years and the success rate as we would like to see the yearly trend of the outcome
- GitHub Link

EDA with SQL

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first succesful landing outcome in ground pad was acheived.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- 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.
- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

- GitHub Link

Build an Interactive Map with Folium

- Below map objects were added to the folium map
  - A circle object for each launch site with its name as popup label – This is because we need to locate all the launch site
  - A MarkerCluster object for each launch site as well as several folium.marker objects under it – This is because we need to demonstrate the successful and failed launches under each site
  - A MousePosition object is added – to get coordinates from a mouse over a point on the map
  - A folium.marker object and a PolyLine object – to show the distance between a launch site to the coastline

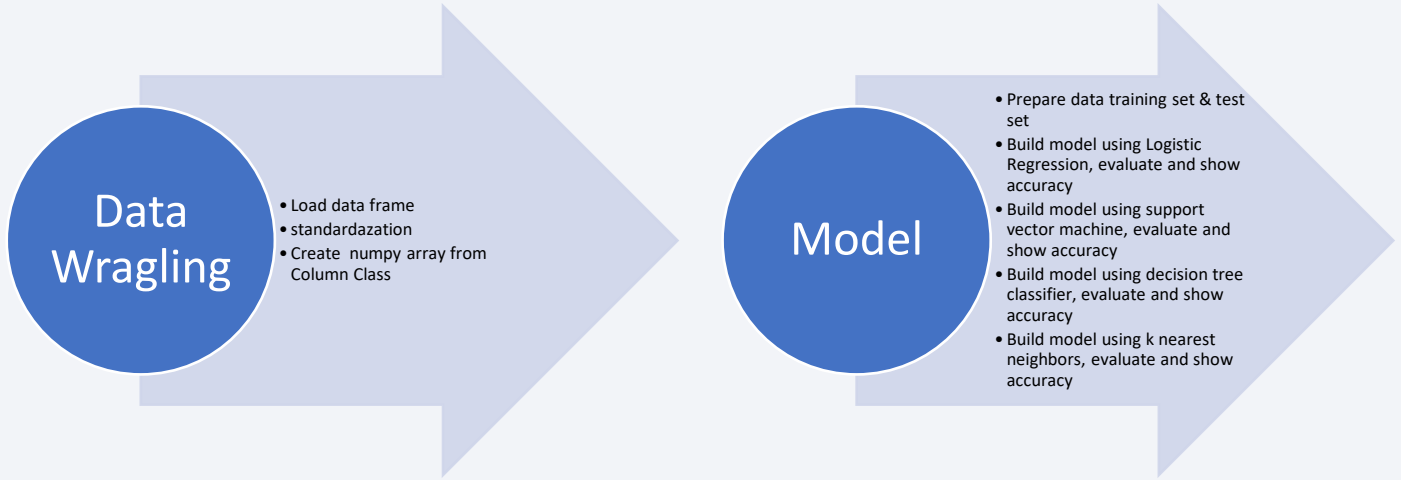
- GitHub Link

Build a Dashboard with Plotly Dash

- A pie chart was added to demonstrate the launch sites and their success counts vs failed counts
  - Reason:This is so we can easily see the launch sites and their success counts overview
- A plot chart was added to demonstrate the correlation between Payload Mass and the outcome, with each plot color-labelled with Booster version
  - Reason: This is so we can visually observe how payload may be correlated with mission outcomes for selected site(s) as well as with different boosters
- On top of the charts
  - A dropdown list with the launch sites was added
  - A range slider for the payload mass was added
  - Reason: we can observe the changes in different combinations (launch sites, payload mass, etc)
- GitHub Link

Predictive Analysis (Classification)

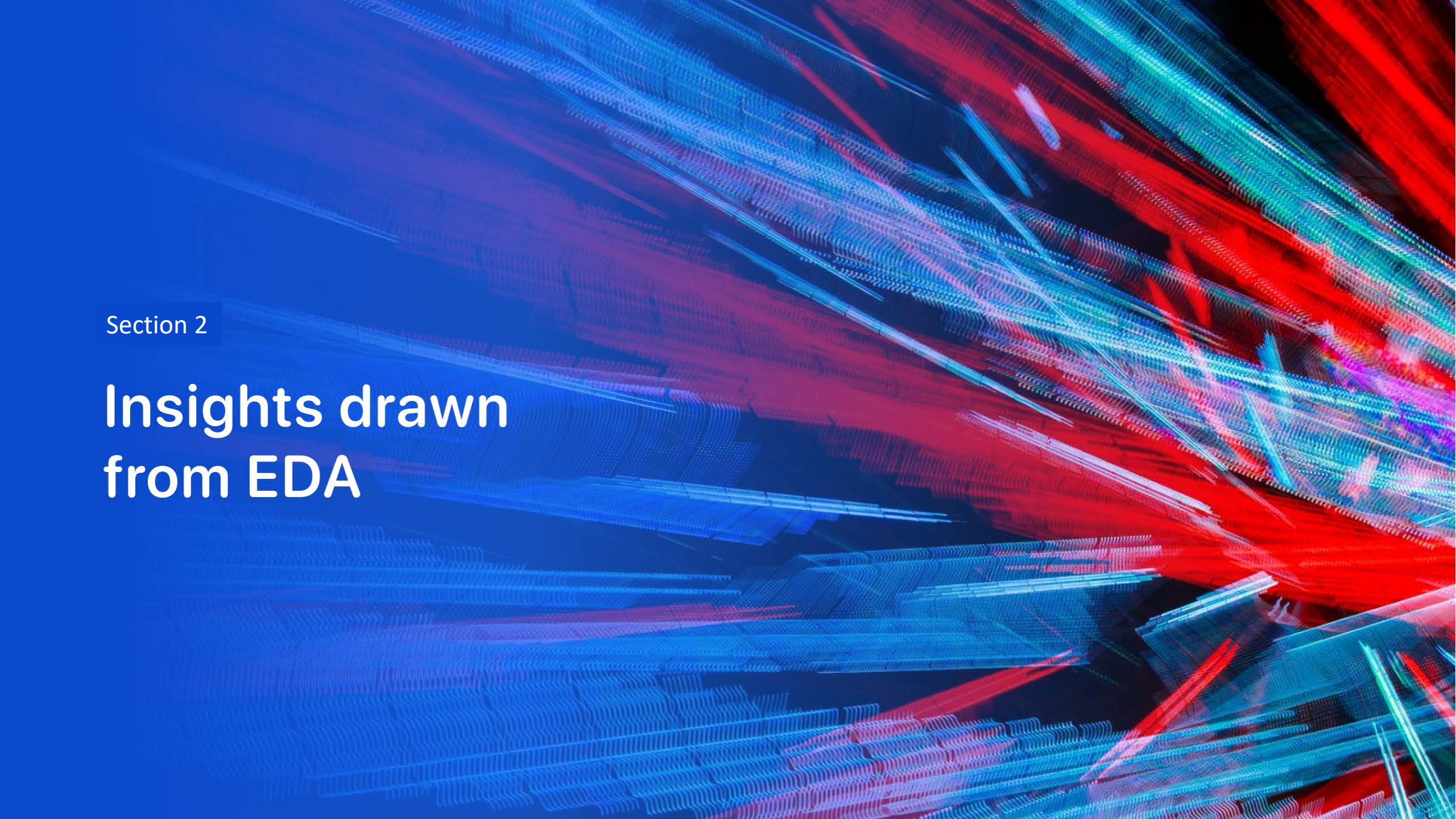
- The model was built based on the column of the class, via standarzation; a training split of 80% vs 20%; then trained using logistic regression, svm, decision tree classifier and knn; the best method is chosen by comparing the accuracy among all those models



- GitHub Link

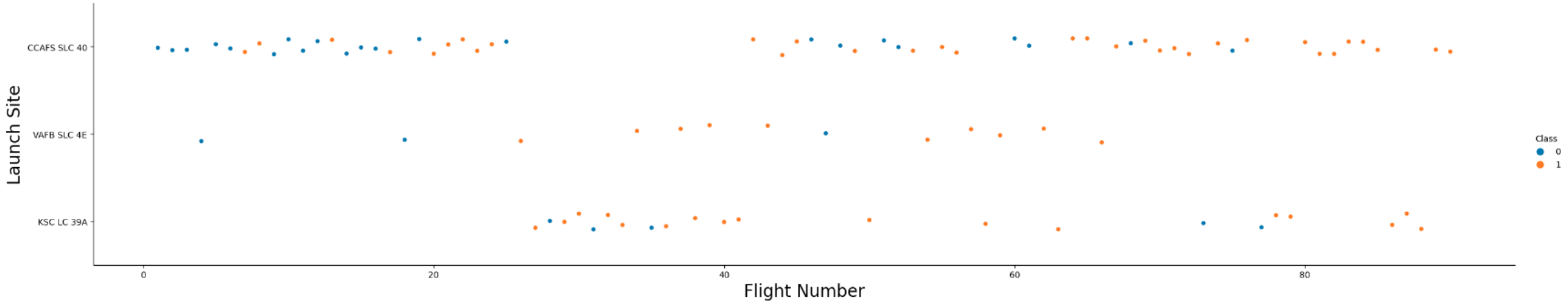
Results

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


Section 2
Insights drawn from EDA

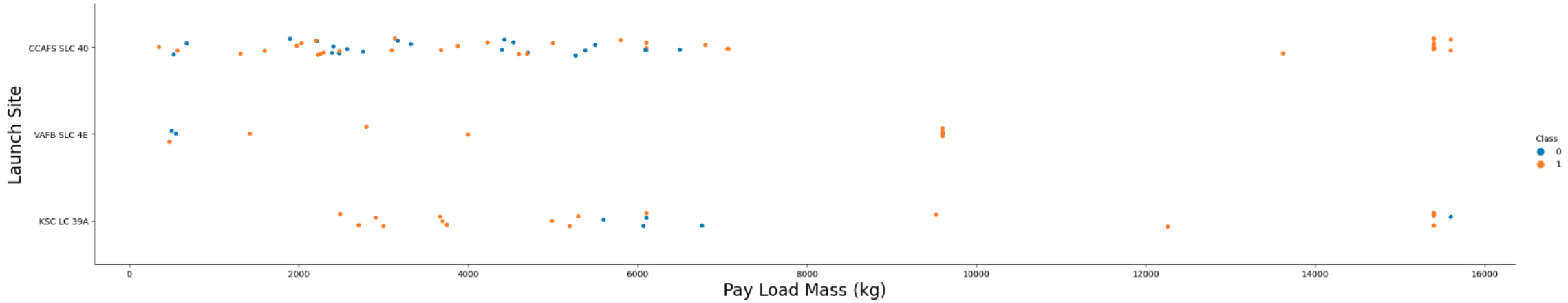
Flight Number vs. Launch Site

- From the observation we find that
  - At Site CCAFS SC 40, the success rate increase when the flight number is above 80
  - At Site VAFB SLC 4E, the success rate increase when the flight number is above around 50 but there is no lunch after 80
  - At Site KSC LC 39A, the success rate is at its highest between flight number 40 and 60



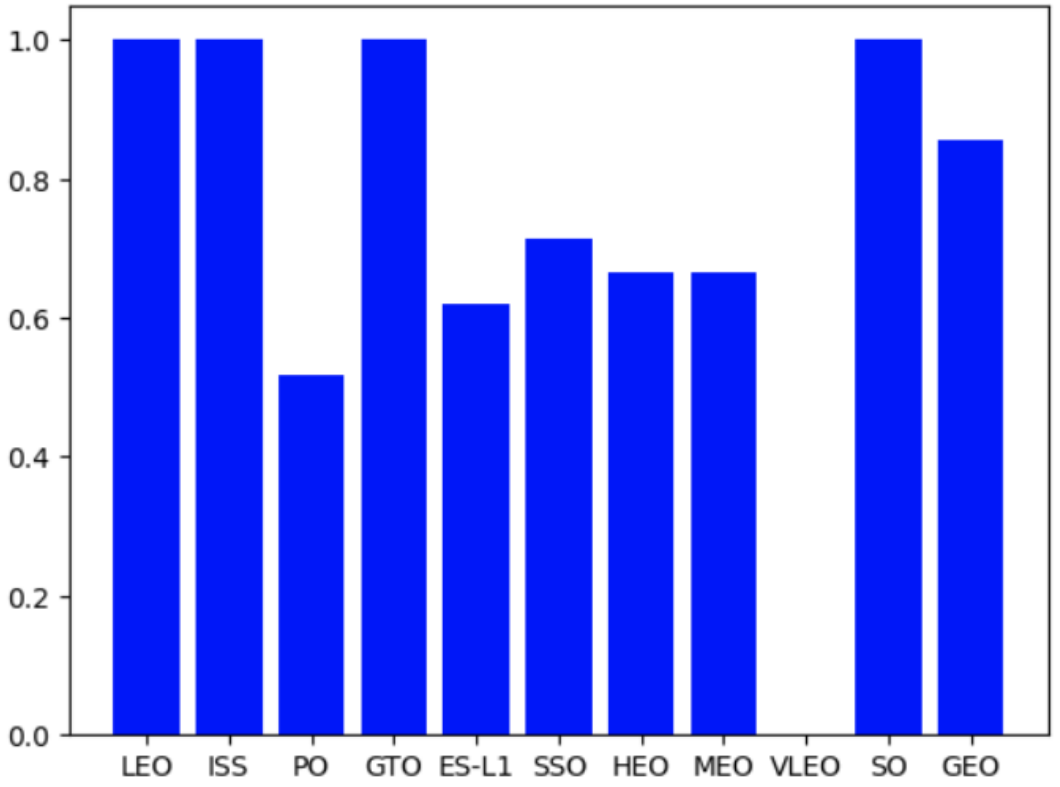
Payload vs. Launch Site

- From the observation we find that
  - At site CCAFS SC 40, the success rate increase when the flight number is above 8000 kg
  - At site VAFB-SLC there are no rockets launched for heavypayload mass(greater than 10000).



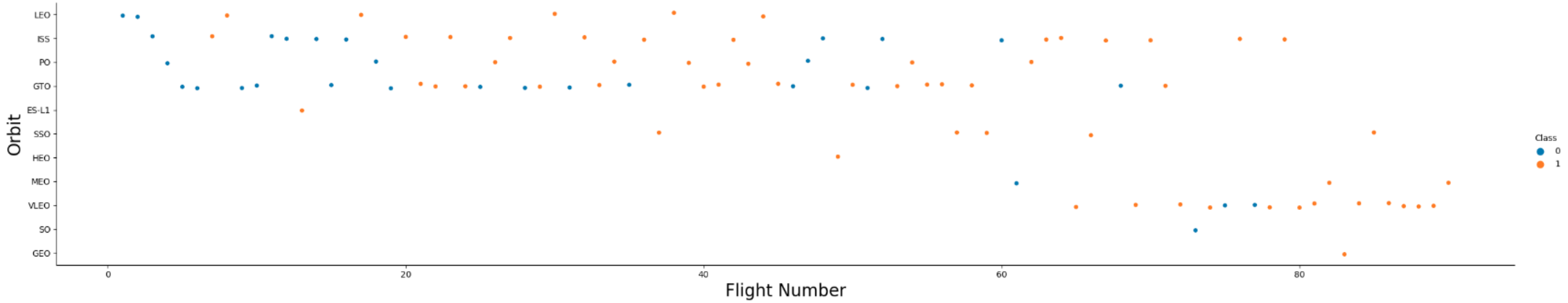
Success Rate vs. Orbit Type

- From the observation we find that
  - Orbit LEO, ISS and SO have the highest success rate
  - Orbit VLEO has the lowest success rate



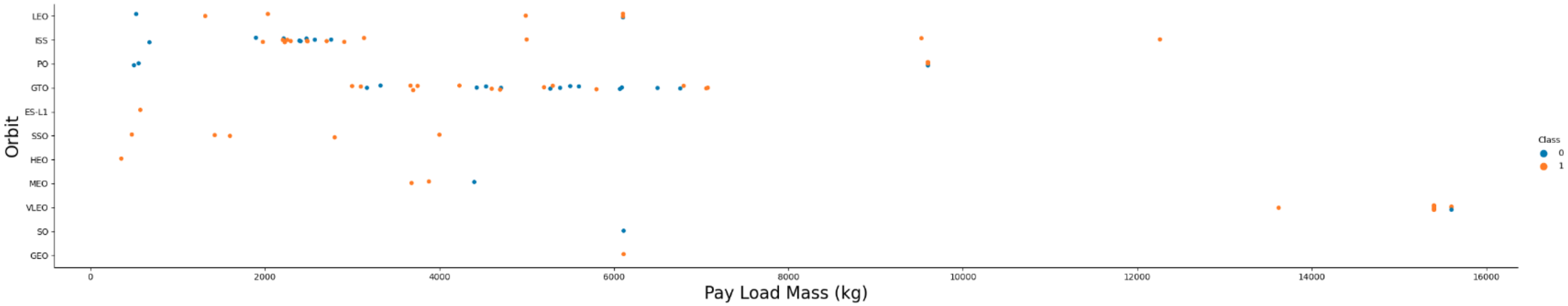
Flight Number vs. Orbit Type

- From the observation we find that
  - In the LEO orbit the success rate appears related to the number of flights
  - In GTO orbit the success rate appears NOT related to the number of flights

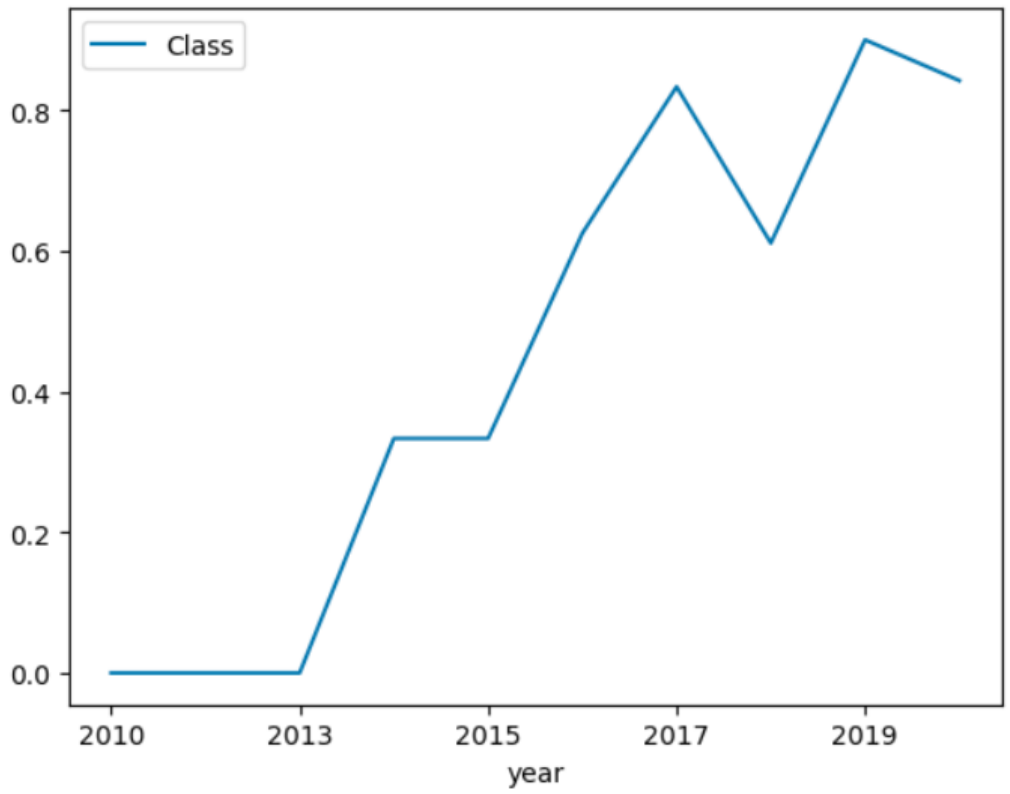


Payload vs. Orbit Type

- From the observation we find that
  - In Polar, LEO and ISS orbit, success rate is positively related to the payloads
  - In GTO orbit the success rate appears NOT related to the payloads



Launch Success Yearly Trend



- From the observation we find that
  - Success rate kept increasing since 2013 till 2020

All Launch Site Names

```
%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

- Use an SQL select query with DISTINCT function of “Launch_Site” from the SPACEXTBL table to get all the unique site names

Launch Site Names Begin with 'CCA'

- Use an SQL select query with Like function of “Launch_Site” in the where clause to extract the names which begins with “CCA”
- Only one “%” is used at the end of the wildcard string since we need to get the names that BEGIN with the keyword

```
%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
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Use an SQL select query with SUM function of “PAYLOAD_MASS_KG” to extract the total amount
- In the where clause we specify that Customer is equal to “NASA (CRS)”

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL where Customer = "NASA (CRS)"
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Use an SQL select query with AVG function of “PAYLOAD_MASS_KG” to extract the total amount
- In the where clause we specify that Booster_Version is equal to “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

First Successful Ground Landing Date

- Use an SQL select query with MIN function of “Date” to extract the smallest amount, i.e., the earliest date of the first successful landing outcome

```
%sql SELECT MIN(Date) FROM SPACEXTBL
```

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

MIN(Date)
01-03-2013

Successful Drone Ship Landing with Payload between 4000 and 6000

- Use an SQL select query with DISTINCT function of “Booster_Version” to extract the unique drone ship versions
- In the where clause we specify that,
  - Mission_Outcome is equal to “Success”
  - PAYLOAD_MASS__KG is between 4000 and 6000

```
%sql SELECT distinct[Booster_Version] FROM SPACEXTBL where Mission_Outcome = "Success" and (PAYLOAD_MASS__KG_ between 4000 and 6000)
 * sqlite:///my_data1.db
Done.
```

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 FT B1032.2
F9 B4 B1040.2
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5 B1046.3
F9 B5 B1048.3
F9 B5 B1051.2
F9 B5B1060.1
F9 B5 B1058.2
F9 B5B1062.1

Total Number of Successful and Failure Mission Outcomes

- Use an SQL select query for,
  - the Mission_Outcome
  - With COUNT function of “Mission_Outcome” to extract the total number of each outcome
- The result is grouped by different types of outcomes

```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) FROM SPACEXTBL Group by Mission_Outcome
```

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

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

Boosters Carried Maximum Payload

- Use an SQL select query to select “Booster_Version”
- In the where clause we use a subquery to select the booster version that carries the maximum payload mass

```
%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

2015 Launch Records

```
%sql SELECT substr("Date", 4, 2) as "Month Name", "Landing _Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTBL where substr(Date, 7, 4) = "2015" and "Landing _Outcome" = "Failure (drone ship)"
 * sqlite:///my_data1.db
Done.
```

Month Name	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

- Use an SQL select query,
  - with a substr function to select get the month of the data as “Month Name”
  - "Landing _Outcome", "Booster_Version“ and "Launch_Site"
- In the where clause we specify that
  - The year of the date is equal to 2015
  - The landing outcome is equal to “Failure (drone ship)”

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

- Use an SQL select query for,
  - the Landing_Outcome
  - With COUNT function of “Landing_Outcome” to extract the total number of each outcome
- In the where clause we specify that
  - the “Landing Outcome” should include the keyword “Success”
  - The date should be between 2010-06-04 and 2017-03-20
- The result should be grouped by Landing_Outcome

```
%sql select "Landing _Outcome", count("Landing _Outcome") as "Count of Landing Outcomes" from SPACEXTBL  \
   where "Landing _Outcome" like "Success%" and (Date between '04-06-2010' and '20-03-2017') group by "Landing _Outcome"
```

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

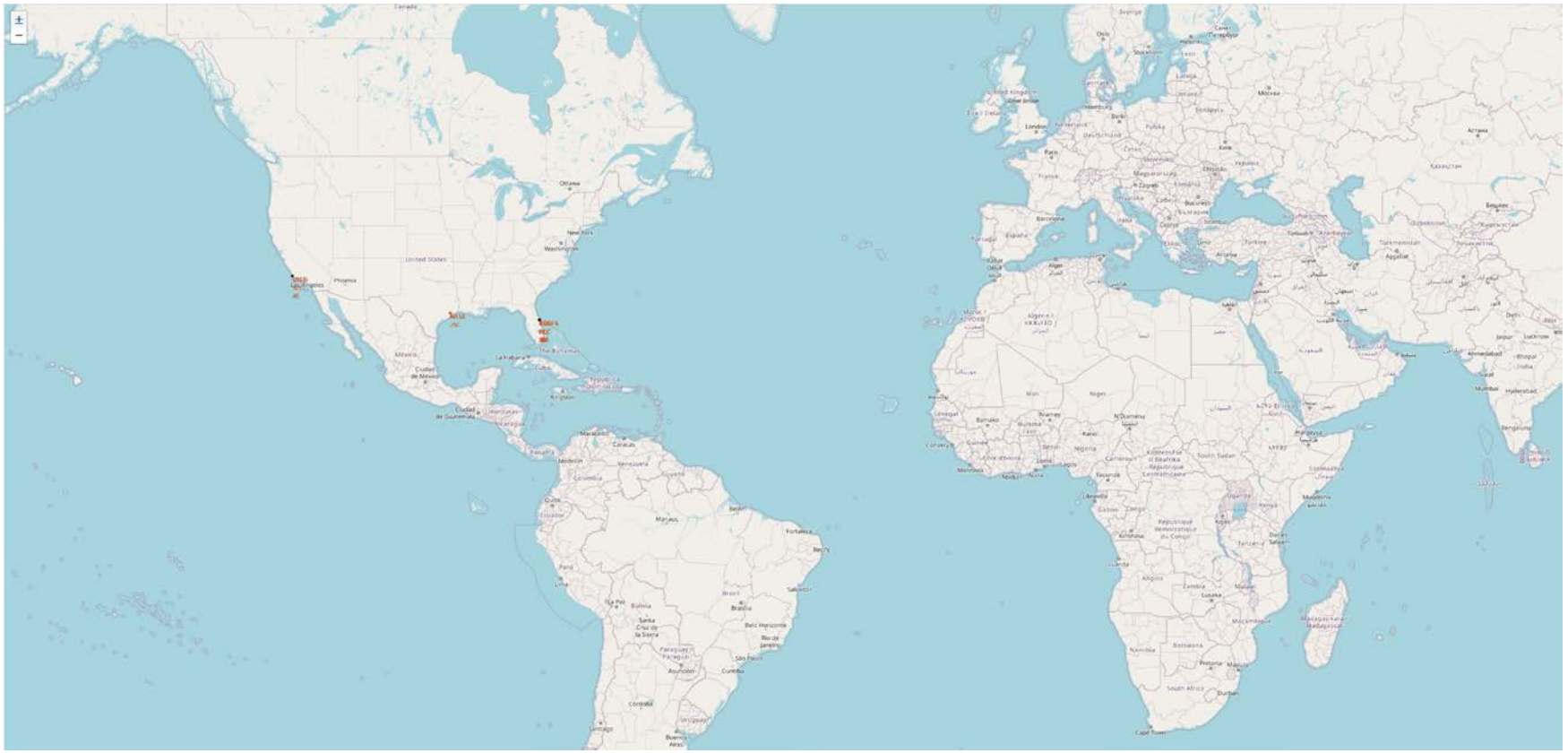
Landing _Outcome	Count of Landing Outcomes
Success	20
Success (drone ship)	8
Success (ground pad)	6

Section 3

Launch Sites Proximities Analysis

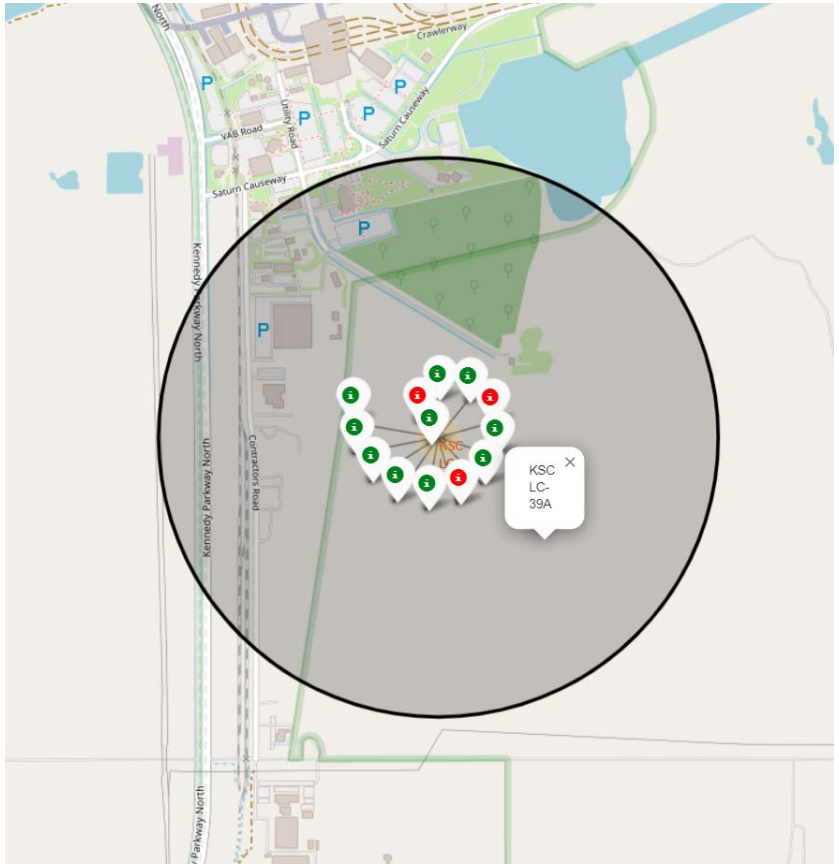
All Launch Site Locations

- All launch sites are in proximity to the Equator line
- All launch sites are in very close proximity to the coast



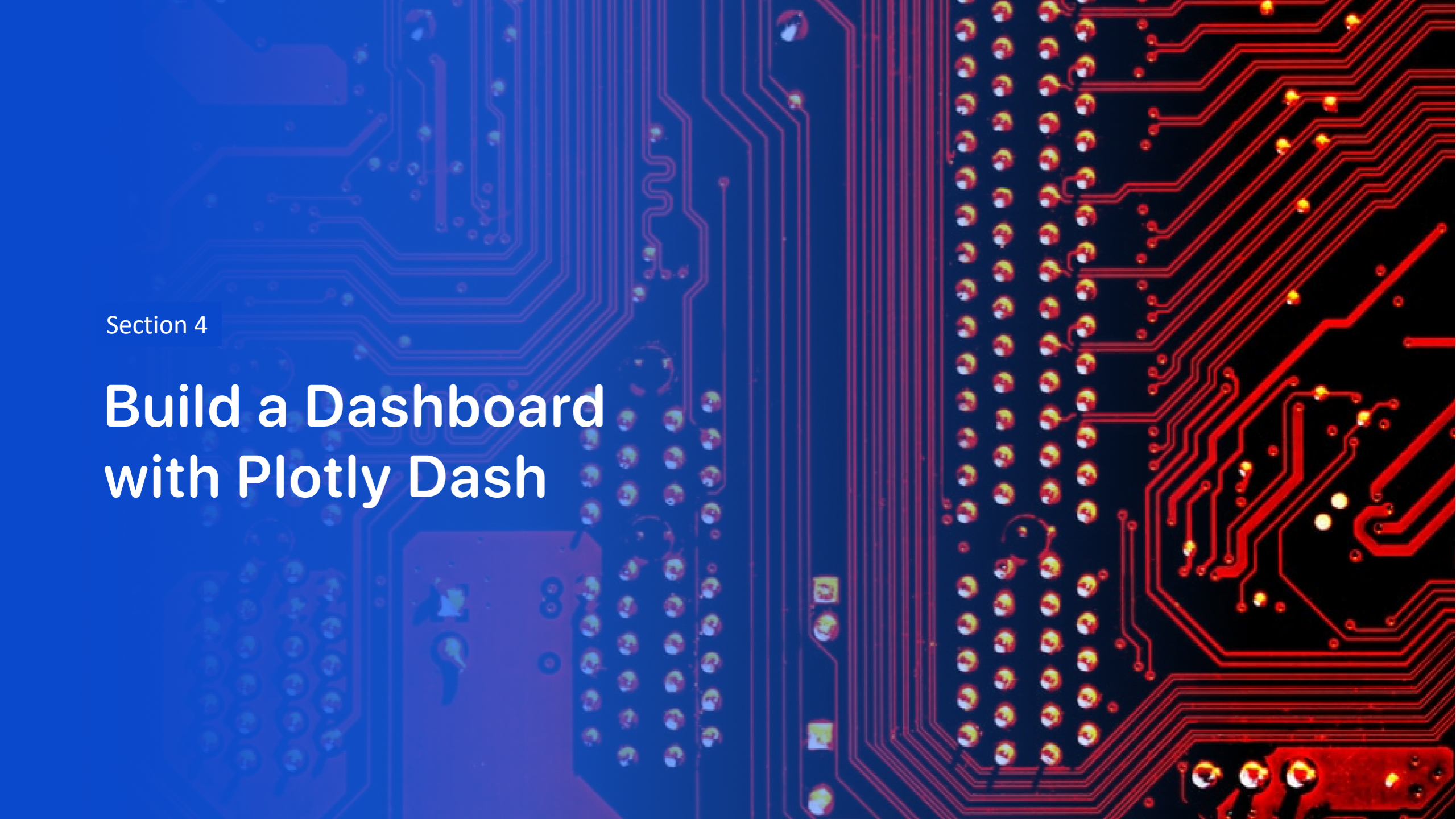
Success/failed Launches for Each Site

- The success launches are labelled with green
- The failed launches are labelled with red
- After visually comparing each site we find that KSC LC-39A has the highest success rate



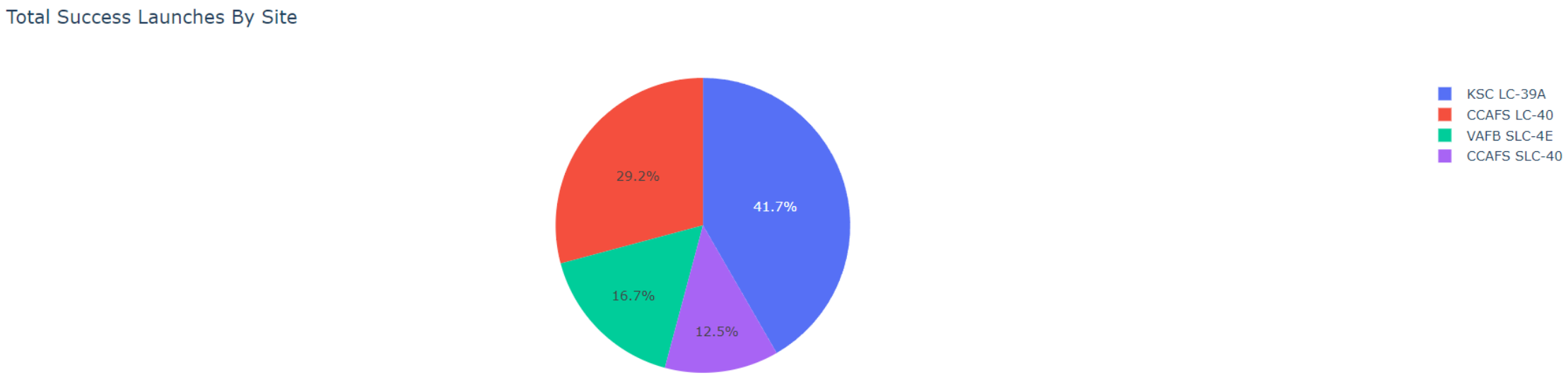
Distance to Coastline (CCAFS SLC-40)

- In the map we added a distance marker to display the distance from Site CCAFS SLC-40 to the one of the closest coastline point
- The distance was calculated via a custom function called “calculate_distance”
- The coordinates was taken from the previously added MousePosition object


Section 4
Build a Dashboard with Plotly Dash

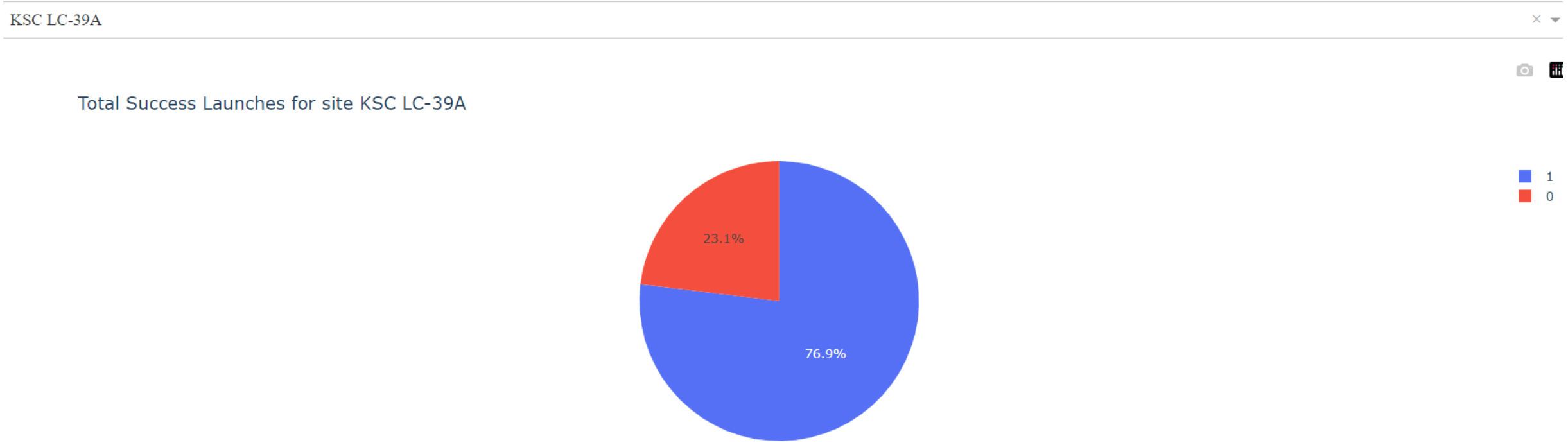
Total Success Launches By Site

- All launch sites are in proximity to the Equator line
- All launch sites are in very close proximity to the coast



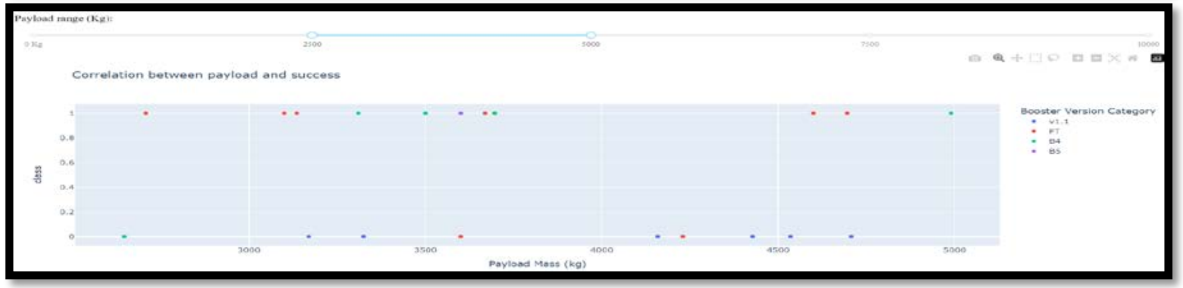
Total Success Launches for Site KSC LC-39A

- All launch sites are in proximity to the Equator line
- All launch sites are in very close proximity to the coast



Payload Vs Launch Outcome

- Payload range 2500-5000 kg has the largest success rate
- FT booster version has the largest success rate in above range

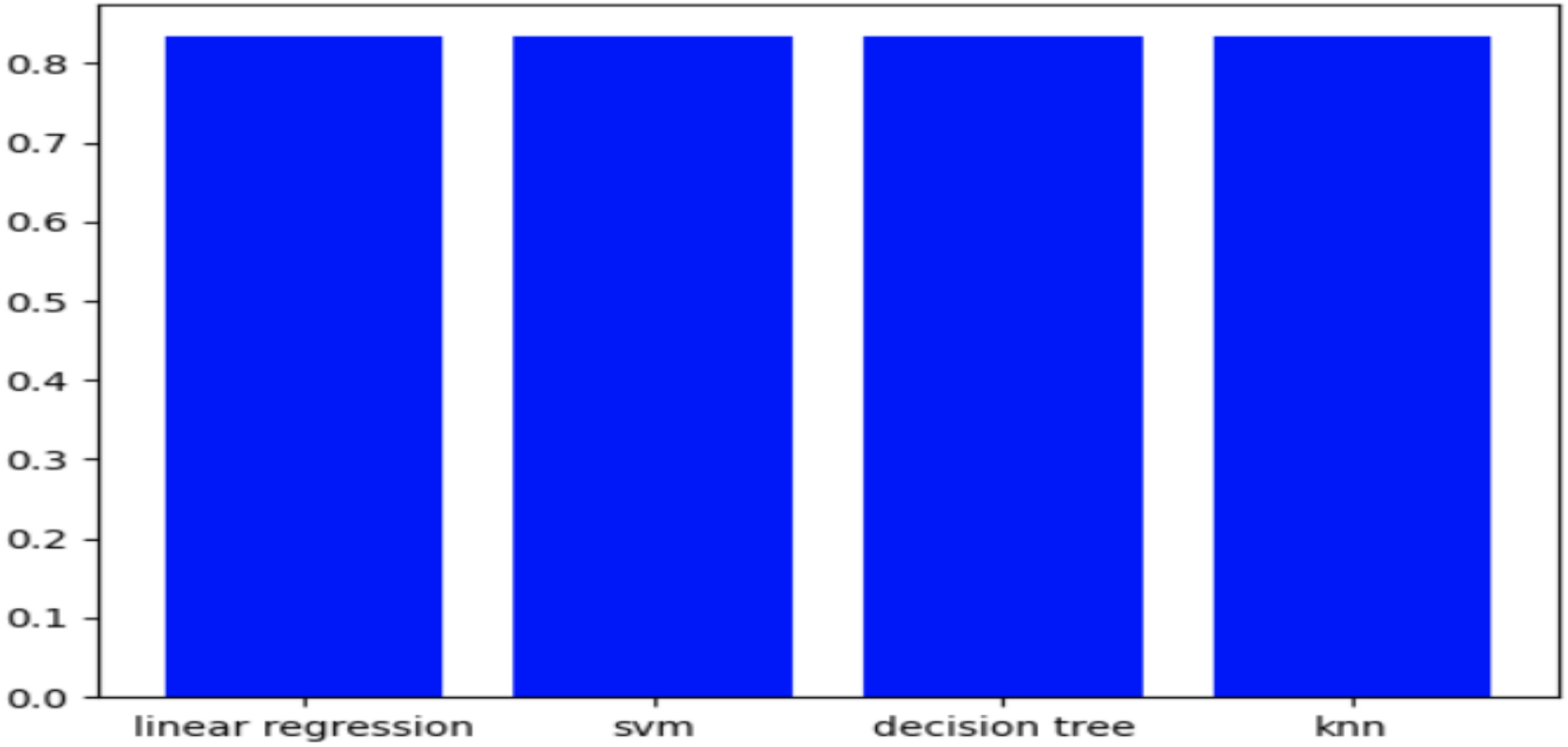


Section 5

Predictive Analysis (Classification)

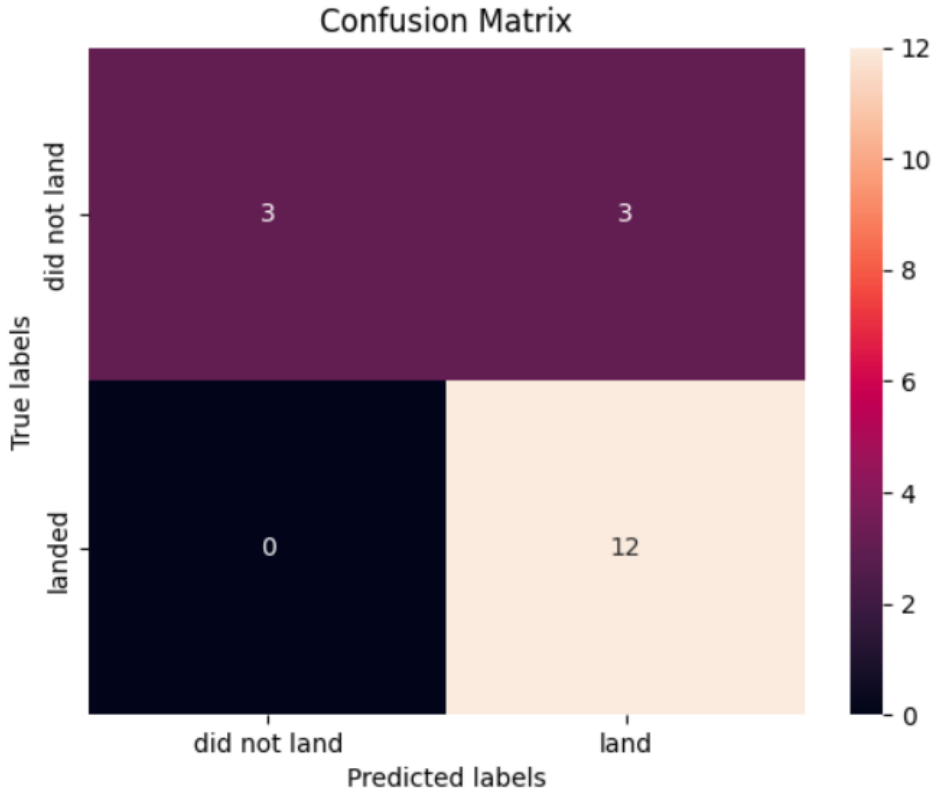
Classification Accuracy

- All four models produce the same accuracy, therefore all of them are equally good in predicting the results



Confusion Matrix

- All four models successfully predict the success landing outcomes (True Positives)
- All has a major problem in false positives



Conclusions

- Launch Site KSC LC-39A has the highest success launch rate
- Payload mass has a positive relationship with success launch rate
- Flight number has a positive relationship with success launch rate
- All four models successfully predict the success landing outcomes (True Positives)
- If the landing outcome is predicted to be success, then there is a very big chance that it will be successful

Appendix

- Datasets
  - Part 1
  - Part 2
  - Part 3
- SpaceX launch history data
- SpaceX launch geo data

Thank you!