IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Thomas John
27th July 2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection is done from sources such as SpaceX API and Webscraping from Wikipedia Webpage
  - Data Wrangling is done to transform the data, so that we can summarize the outputs for easy classification.
  - EDA is done using Visualization and SQL queries to gain first insights.
  - Using interactive map explore launch sites with regards to their outcomes, proximities to landmarks
  - Classification is done using Logistic Regression, SVM, Decision Tree, KNN and analyzed.
- Summary of all results
  - Though the success rate was very low start of experiment, towards the end it is very high
  - All launch sites are in good proximity to coastlines and equator and have good connectivity to roads, highway and railways
  - For the orbits, ES-L1, SSO and HEO, there is low payload and 100% success rate
  - KSC LC-39A is the launch site with highest success rate
  - Decision Tree Model showed the highest classification accuracy

Introduction

- Problem Statement - Predict if the Falcon 9 first stage will land successfully
  - SpaceX advertises their Falcon 9 rocket launches with a cost of 62 million dollars. Much of the savings is because SpaceX can reuse the first stage
  - Other providers cost upward of 165 million dollars each
  - If we can predict, that the first stage will land or not, we can determine the cost of a launch. And an alternate company can bid against accordingly

- Goals
  - Collect Falcon 9 launch data from SpaceX API and using Webscraping
  - Prepare data, from various outputs using Data Wrangling
  - Explore relation between variables like Flight Number, Payload, Orbit, Success etc.
  - Explore various Launch Sites and their Success Rates and Location Significance
  - Predict the launch success by various machine learning algorithms and compare

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Falcon 9 data is obtained from SpaceX API and using Webscraping.
- Perform data wrangling
  - Transform different data outcomes to binary outcomes, ‘Success’ and ‘Failure’
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Build a Machine Learning Pipeline for the prediction.
  - Its initial stages include preprocessing and train test dataset split.
  - Use different algorithms and determined their best hyperparameters using GridSearch
  - Determine model with best accuracy for training model and use confusion matrix to assess output

Data Collection

- Data Collection is done from two sources
  1. SpaceX REST API
  2. Webscraping data from Wikipedia website

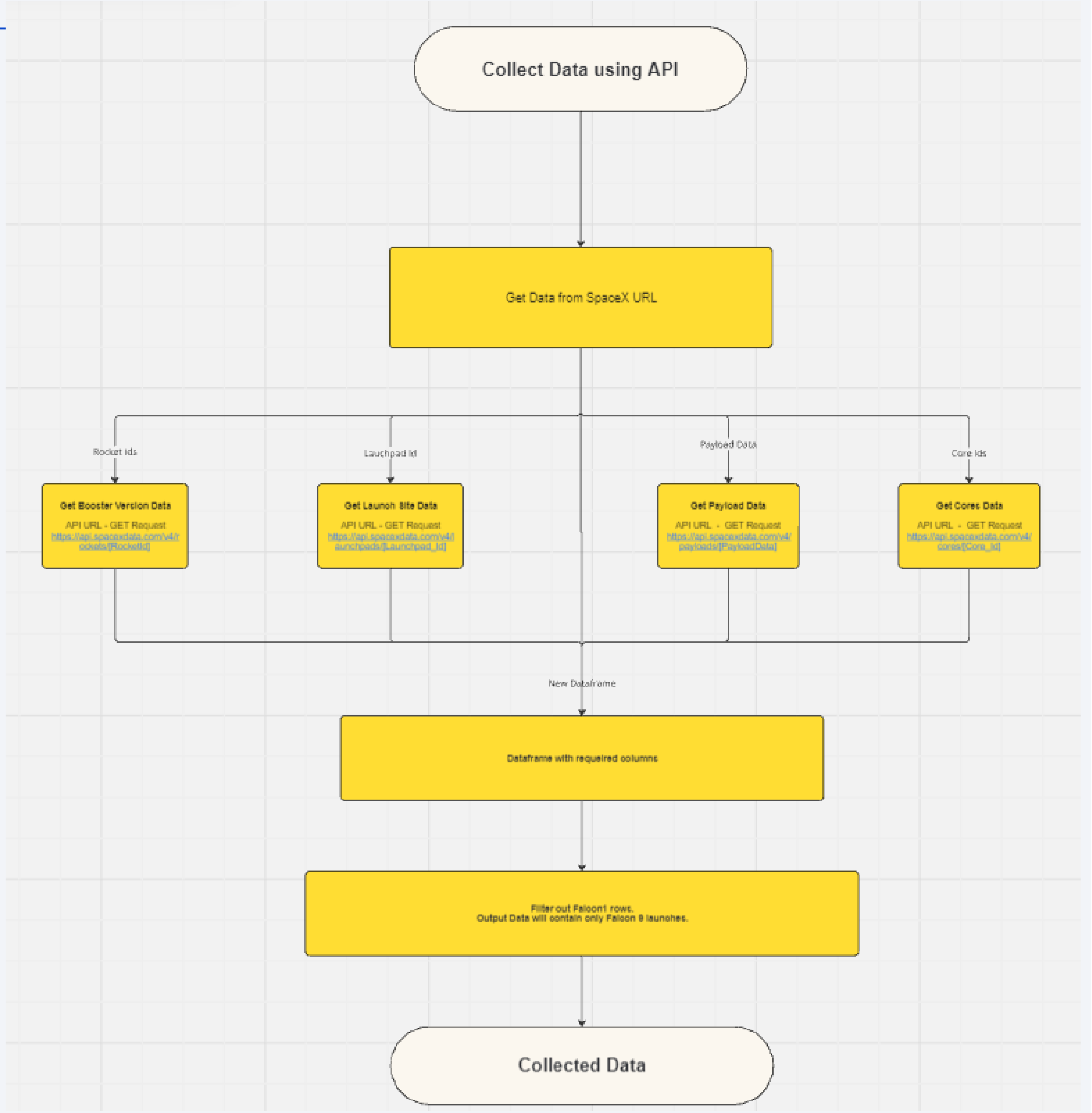
- Data collected will have data of all launches and we use filtering to get Falcon 9 data

- Data Wrangling is done to convert data outcomes to either Success or Failure

Data Collection – SpaceX API

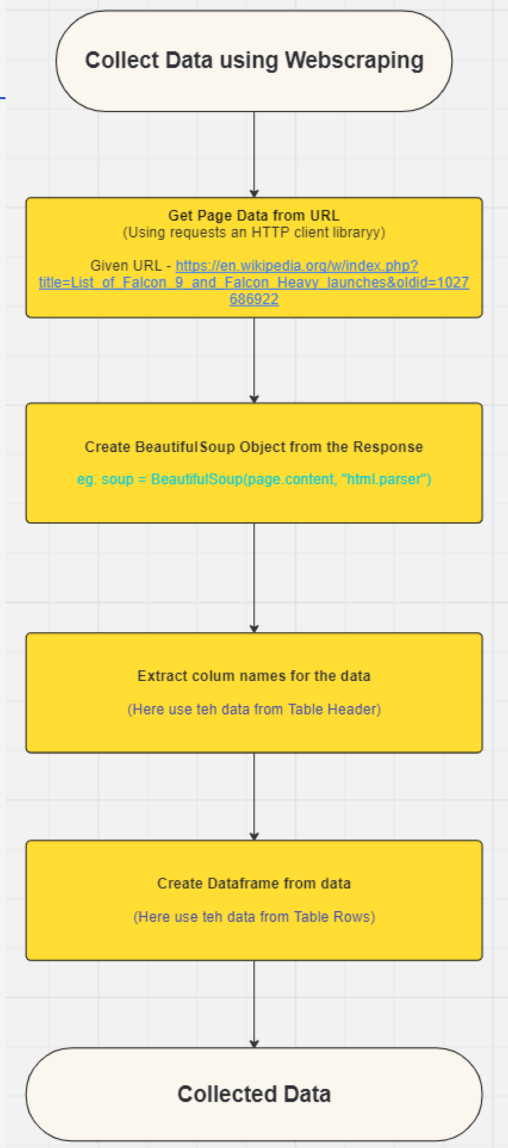
- Basic data of past launches is collected first by calling the main API url which has past launch site data.
- Detailed information on BoosterVersion, Cores, PayloadMass, LauchSite is collected later by calling the required APIs using ids of them received in first API call and it is stored in the dataframe.
- GitHub URL –

https://github.com/devmanac/MOOC_Assignment/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



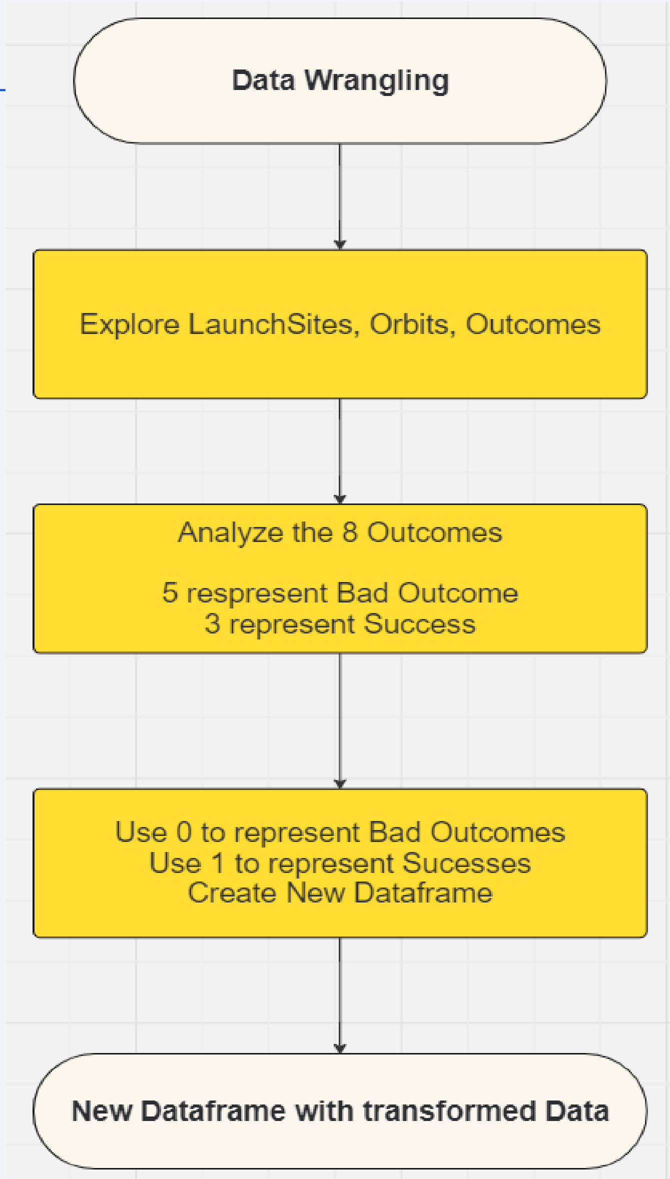
Data Collection - Scraping

- For Webscraping first the HTML page is accessed and stored into an object
- From this objects the headers and rows to be filled in dataframe is extracted.
- GitHub URL - https://github.com/devmanac/MOOC_Assignment/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Explore the data to find the details and use Data Wrangling to create the Dataframe we need
- In the analysis we will see the types of Launchsites, Orbits and Outcomes.
- There are 8 outcomes from which we need to find the outcome is 'Success' or 'Failure'. We reduce the outcomes to these binary outcomes for further analysis we need.
- GitHub URL - https://github.com/devmanac/MOOC_Assignment/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

Github URL -
https://github.com/devmanac/MOOC_Assignment/blob/main/edadataviz.ipynb

- List of charts

1. Seaborn Categorical Plot Flight Number vs Payloadmass. Outcome is presented as an overlay. We could see the progress in Payloadmass in time and also the Success Rate with time.

2. Seaborn Categorical Plot LaunchSite vs. Flight Number. Outcome is presented as an overlay. Analyse the outcome in various LaunchSites in relation to FlightNumber.

3. Seaborn Scatter Plot LaunchSite vs. Flight Number. Outcome is presented as an overlay

4. Using scatter plot analyze the same relation, LaunchSites in relation to FlightNumber

EDA with Data Visualization

5. Seaborn Barplot to visualize relation between Success Rate and Orbit Type

6. Visualize the relationship between FlightNumber and Orbit with Outcome as an overlay in a Scatter Plot.

7. Visualize the relationship between Payloadmass and Orbit with Outcome as an overlay in a Scatter Plot.

8. Use a line plot to visualize Success Rate over the Years

EDA with SQL

- We analyze the Launch Sites in the first couple of SQL queries
- Payload mass tendencies, Booster Version, Launches related to particular customer etc. are analyzed in the next couple of SQL queries
- In the following few queries we analyze the Landing Outcome with regards to launch pad, booster version, payload mass, number, max value etc.
- In the last queries we check the Success and Failure in missions in certain years, time frames, their counts, booster version

GitHub URL - https://github.com/devmanac/MOOC_Assignment/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Initially the number of launches are shown in a circle over launch sites
- Clicking further we could see the Success and Failures denoted by green and red markers
- Popups and labels are provided to see the name of each launch site
- Lines showing distance is drawn to important geographic locations like highway, roads, coastline etc. with distance shown
- Launch sites are all in close proximity to equator, coastline, railway and roads
- Github URL - https://github.com/devmanac/MOOC_Assignment/blob/main/lab_jupyter_launch_site_location.ipynb

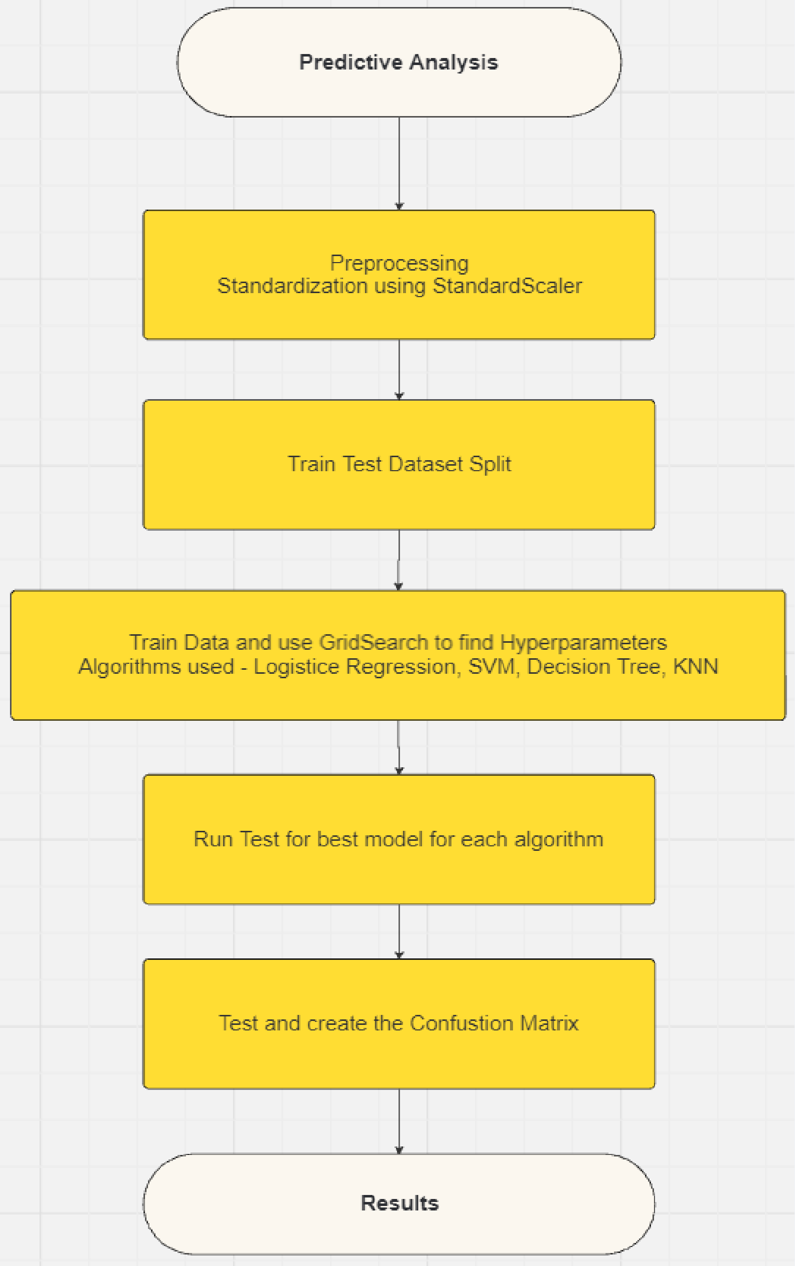
Build a Dashboard with Plotly Dash

- Dynamic loading of data is done in chart outputs with inputs from Dropdown and Slider values
- Dropdown contains the launch sites where all or any launch site could be selected
- Slider corresponds to Payload Mass
- Pie chart shows the success rate of the launch site(s) selected in Dropdown
- Scatter plot takes the launch site input from dropdown and payload range from slider as inputs and shows the Success or Failure with Booster Version information
- Github URL – https://github.com/devmanac/MOOC_Assignment/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- We create a machine learning pipeline for prediction
  - Standardize Data
  - Train Test Split
  - Find best Hyperparameters during Training for different algorithms
  - Test the models and create Confusion Matrix to see the best model

- GitHub URL - https://github.com/devmanac/MOOC_Assignment/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



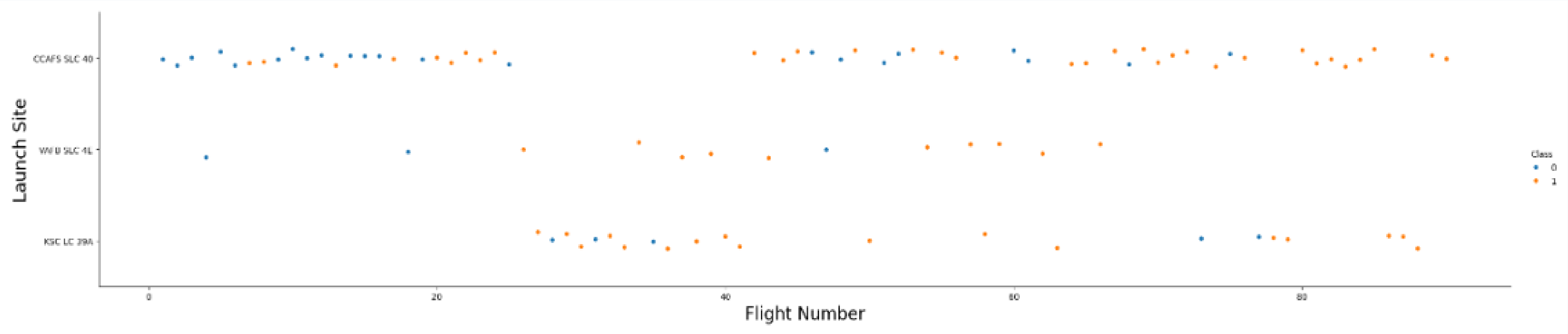
Results

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

Section 2

Insights drawn from EDA

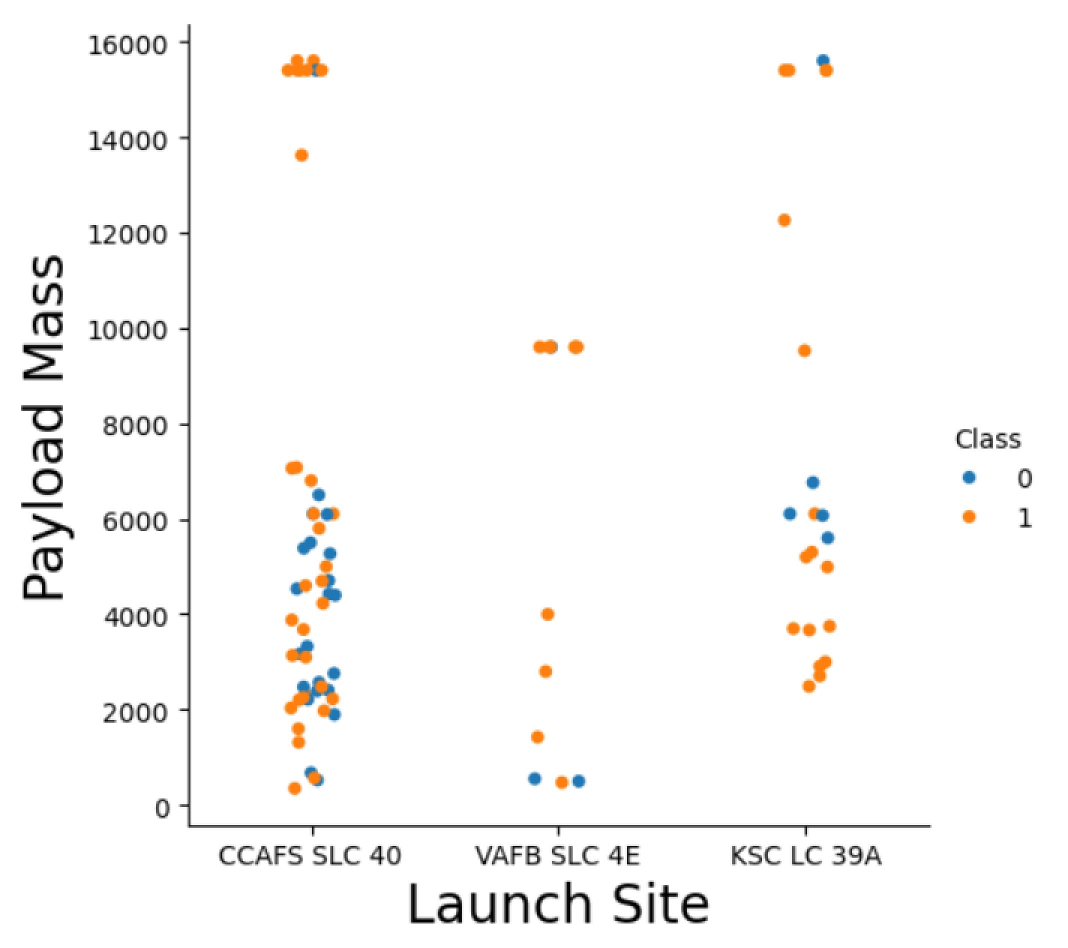
Flight Number vs. Launch Site



- As the flight number increases, we could see the Success launches denoted by orange dots increase and towards the end we could see it increases steeply towards success for all the launch sites.
  - NB - The flight number increases with each launch. So, we could measure progress in time using the flight number.

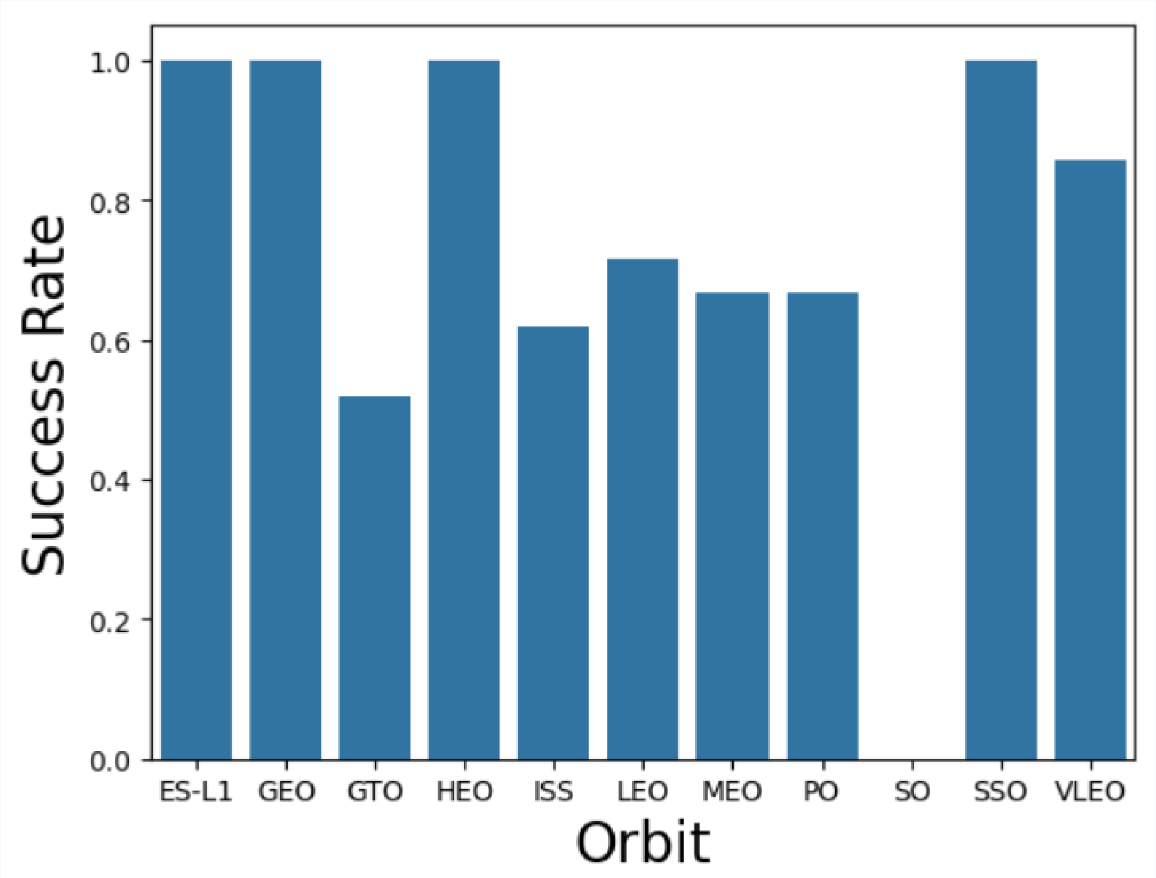
Payload vs. Launch Site

- For high payloads the success rate is quite high
- For the launch site, KSC LC-39A, even for very low payload success rate is very high



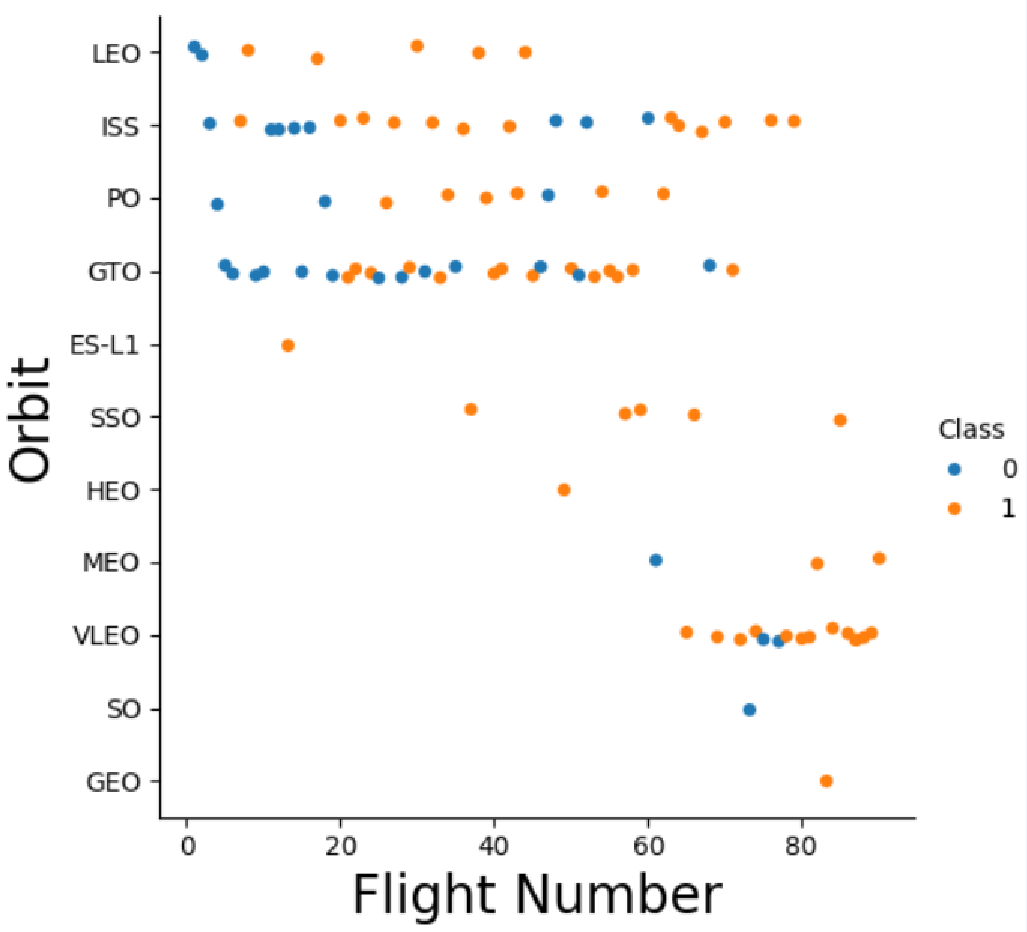
Success Rate vs. Orbit Type

- For the orbits ES-L1, GEO, HEO and SSO the success rate is 100%
- VLEO and LEO follows with approximately 80% and 70%
- For GTO orbit the success rate is below 50%



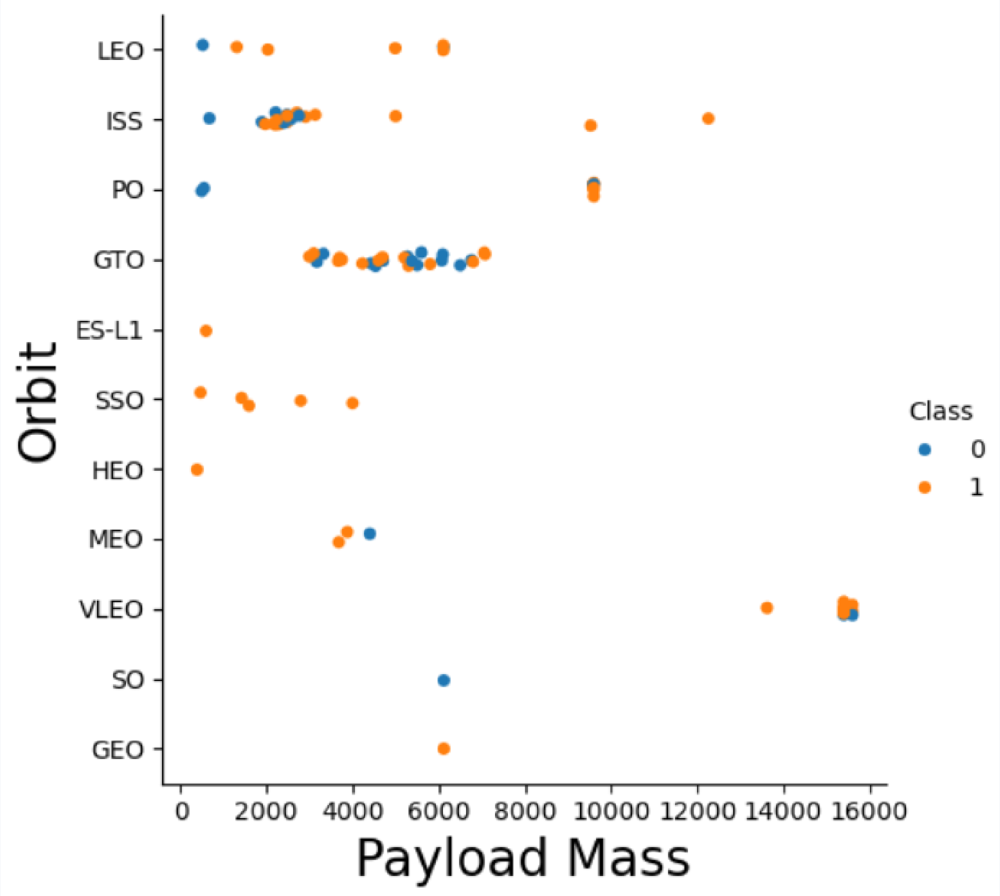
Flight Number vs. Orbit Type

- Towards the end of flight number range, the success rate is 100% for the all Orbits
- In the starting of launches, generally many failures represented by blue dots can be seen
- But steadily the orange dot density increases for all orbits.



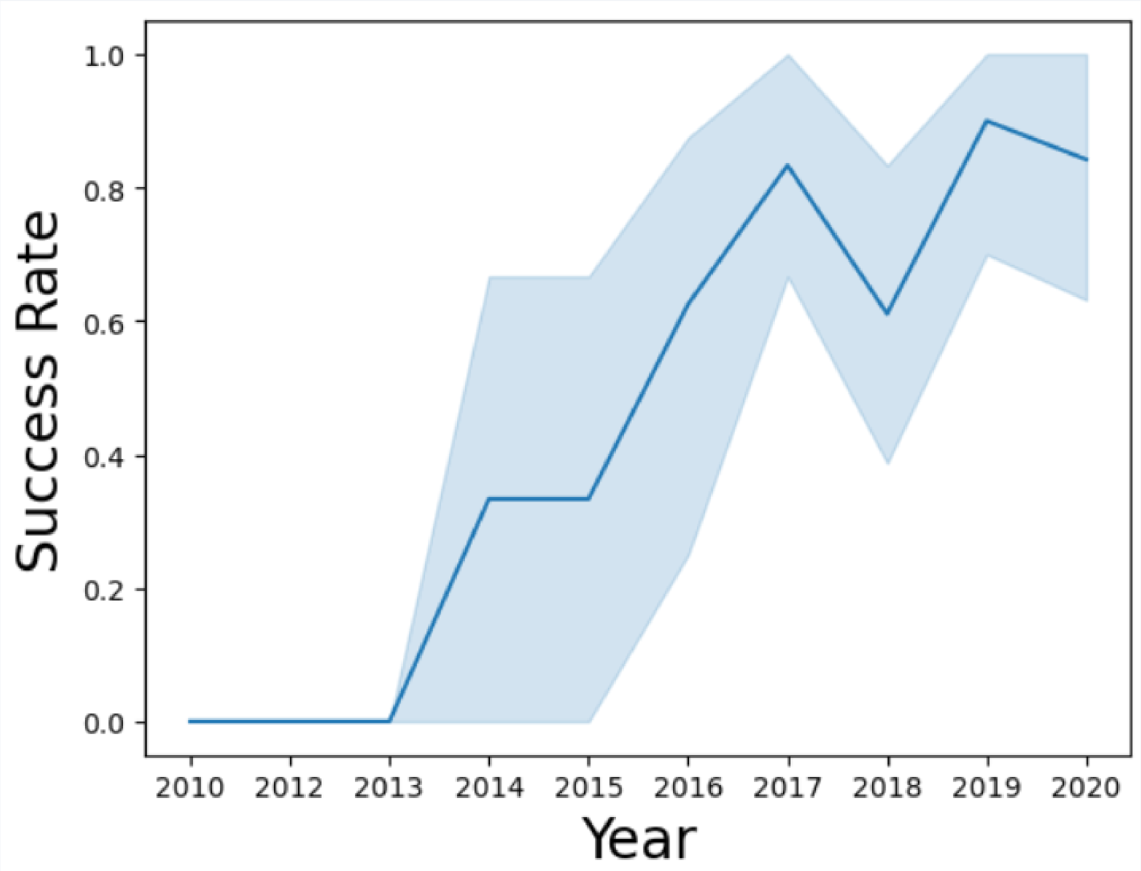
Payload vs. Orbit Type

- For the orbits, ES-L1, SSO and HEO, there is low payload and 100% success rate
- For orbits, LEO, ISS and PO, lower the payload, higher the failures
- For very high payload success rate for orbits, LEO, ISS and PO success rate is 100%



Launch Success Yearly Trend

- We could see there are no successes till the year 2013
- After 2013 till 2017 there is very steep increase in success rate from 0 to 85%
- In year 2019 the highest overall success rate of around 90% can be seen



Launch Sites

```
%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

- DISTINCT keyword is used to get the Launch Sites without duplicates from all the Launch Sites selected

Launch Site Names starting with 'CCA’

```
%sql SELECT * FROM SPACEXTABLE 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
2010-06-04	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

- LIKE keyword is used with the query string including % symbol to select all Launch Sites which start with CCA
- LIMIT keyword is used to limit the row number to 5

Total Payload Mass

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) TOTAL_PAYLOAD_MASS FROM SPACEXTABLE where Customer='NASA (CRS)'
```

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

TOTAL_PAYLOAD_MASS
45596

- To select rows corresponding to customer NASA we use the string for NASA in the WHERE clause
- SUM() keyword is used to get the total of the Payload Mass launched for the customer

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AVG_PAYLOAD_MASS FROM SPACEXTABLE where Booster_Version='F9 v1.1'
```

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

AVG_PAYLOAD_MASS
2928.4

- To select rows for Booster Version F9 v1.1 the comparison string is used in the WHERE clause
- AVG() keyword is used to calculate the average of the Payload Mass

First Successful Ground Landing Date

```
%sql SELECT MIN(Date) EARLIEST_GROUNDPAD_LANDING FROM SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'
```

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

EARLIEST_GROUNDPAD_LANDING
2015-12-22

- We use the MIN() keyword for the date column to obtain the earliest ground pad landing date.
- We filter the Landing Outcomes to Success (ground pad) in WHERE clause
- Result was 22nd Dec 2022 as earliest successful landing on ground pad

Successful Drone Ship Landing Payload 4000 - 6000

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) BOOSTERS FROM SPACEXTABLE where Landing_Outcome = 'Success (drone ship)'
AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

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

- In the WHERE clause, we use BETWEEN keyword to obtain rows with the specified payload mass and Success (drone ship) as Landing Outcome
- We use DISTINCT keyword to avoid duplicates

Total Number of Successful and Failure Mission Outcomes

```
%%sql SELECT 'SUCCESSES' Mission_Outcome, COUNT(Mission_Outcome) AS COUNT FROM SPACEXTABLE
    WHERE Mission_Outcome LIKE 'Success%'
    UNION ALL
    SELECT 'FAILURES' Mission_Outcome, COUNT(Mission_Outcome) FROM SPACEXTABLE
    WHERE Mission_Outcome LIKE 'Failure%'
```

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

Mission_Outcome	COUNT
SUCCESSES	100
FAILURES	1

- We analyze the Mission Outcomes using LIKE keyword and comparing the respective keywords
- We find the count using the COUNT() keyword
- Finally, the whole result is obtained using UNION ALL which shows only 1 failure and 100 successes

Boosters Carried Maximum Payload

```
%%sql SELECT DISTINCT(Booster_Version), PAYLOAD_MASS__KG_ FROM SPACEXTABLE
    WHERE PAYLOAD_MASS__KG_ =
    (SELECT MAX(PAYLOAD_MASS__KG_) as HIGHEST_MASS FROM SPACEXTABLE)
```

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

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

- Using the MAX() function in subquery we determine the highest payload which came out to be 15600 kg
- We will find the Booster Versions which carry this load without duplicates with DISTINCT keyword

2015 Launch Records

```
%%sql SELECT substr(Date, 6,2) as month, Landing_Outcome, Booster_Version, Launch_Site
     FROM SPACEXTABLE
    WHERE Landing_Outcome='Failure (drone ship)' and substr(Date,0,5)='2015'
```

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

month	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

- To select rows for Failure using drone ship we use query string comparison with the ‘Landing_Outcome’ column
- substr() function is used to extract ‘year’ from date and we compare it with string ‘2015’ to get corresponding rows
- We could see there were 2 failures in year 2015, one each in months of January and April

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

```
%%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS Count FROM SPACEXTABLE
    WHERE Date BETWEEN '2010-06-04' and '2017-03-20'
    GROUP BY Landing_Outcome
    ORDER BY Count DESC
```

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

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

- BETWEEN keyword is used in WHERE clause to get rows in the date range
- GROUP BY is used to group the rows on Landing Outcome and we calculate the corresponding count using COUNT() keyword
- To sort in descending order, we use DESC keyword with ORDER BY

Section 3

Launch Sites Proximities Analysis

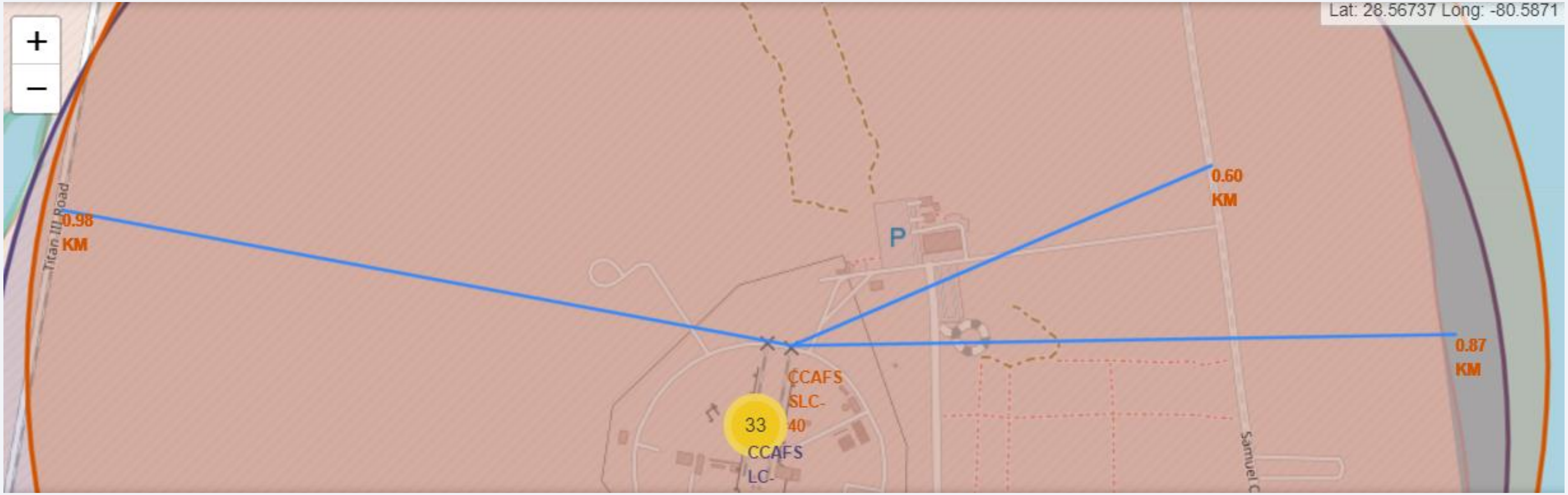
Launch Sites

- There are 4 launch sites, one in the west coast and the other three in the east coast
- Location of launch sites is near to the equator and coastlines

Launch Site Outcomes

- Launch site success is marked with green and failure with red.
- In a popup we could see the name of the launch site
- Clicking on the launch sites we could see the markers in green and red
- We could see, success Rate is relatively high for launch sites like KSC LC-39A

Proximity to Geographic Locations

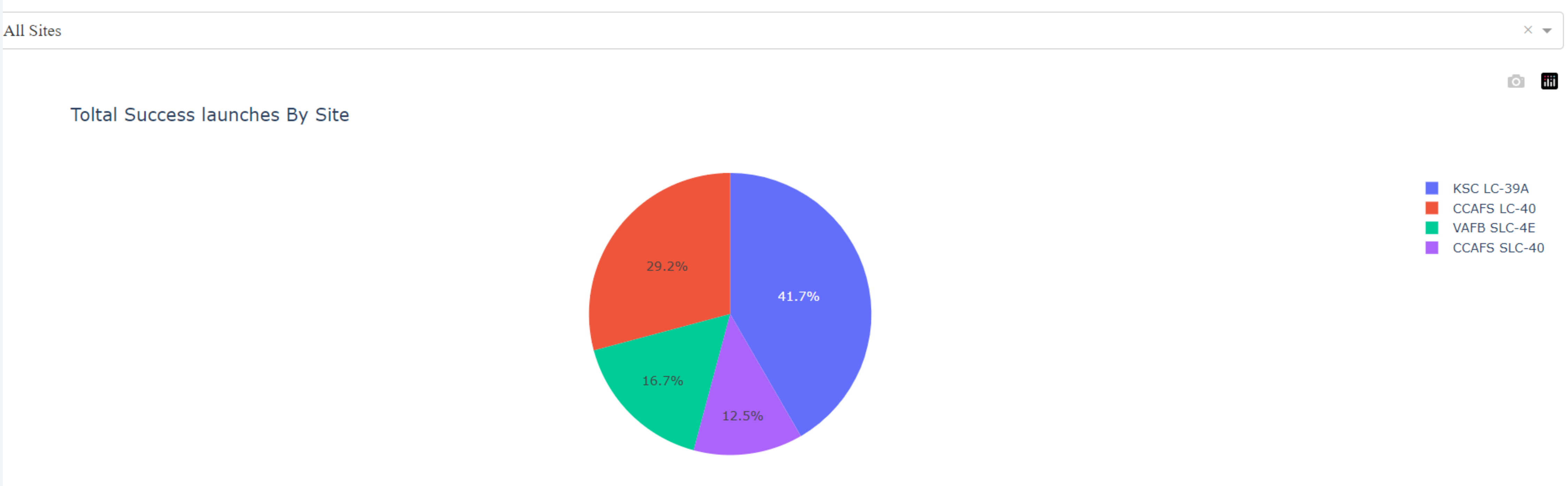


- From the launch site we have drawn lines with distance marked to nearest coastline, to highway and to road from the launch site, CCAFS SLC-40
- Distance to nearest coastline is 0.87 km
- Distance to highway is approximately 0.60 km
- Distance to main road is approximately 0.98km

Section 4

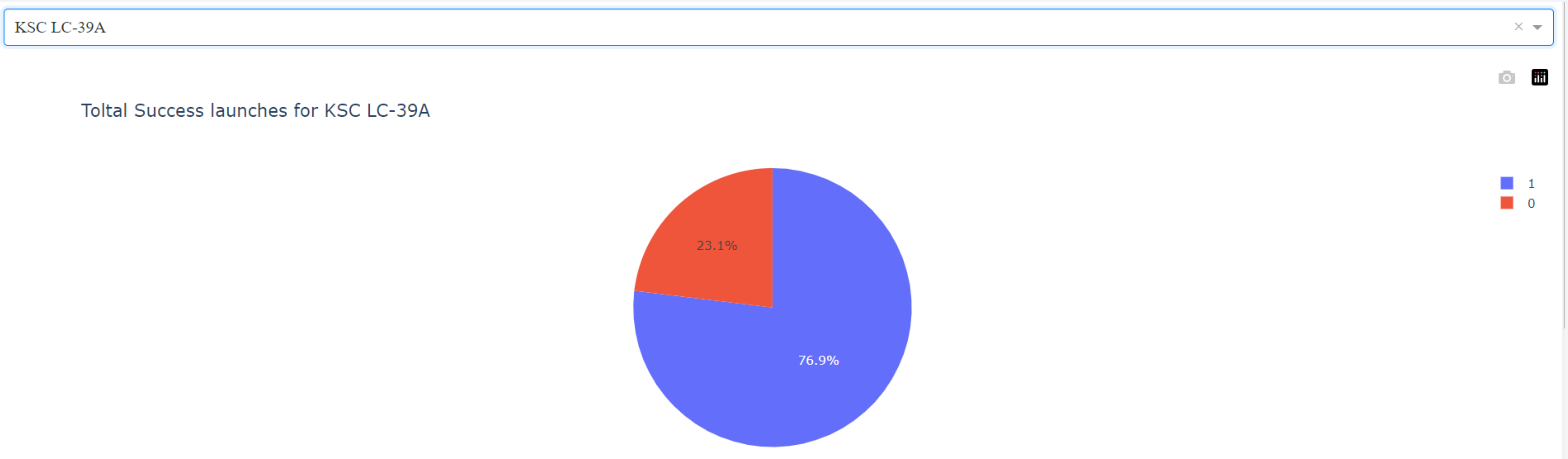
Build a Dashboard with Plotly Dash

Launch Success by Site



- There are 4 launch sites KSC LC-39A, CCAFS LC-40, VAFB SLC-4E and CCAFS SLC-40
- KSC LC-39A has the Highest Number of Successes with 41.7%

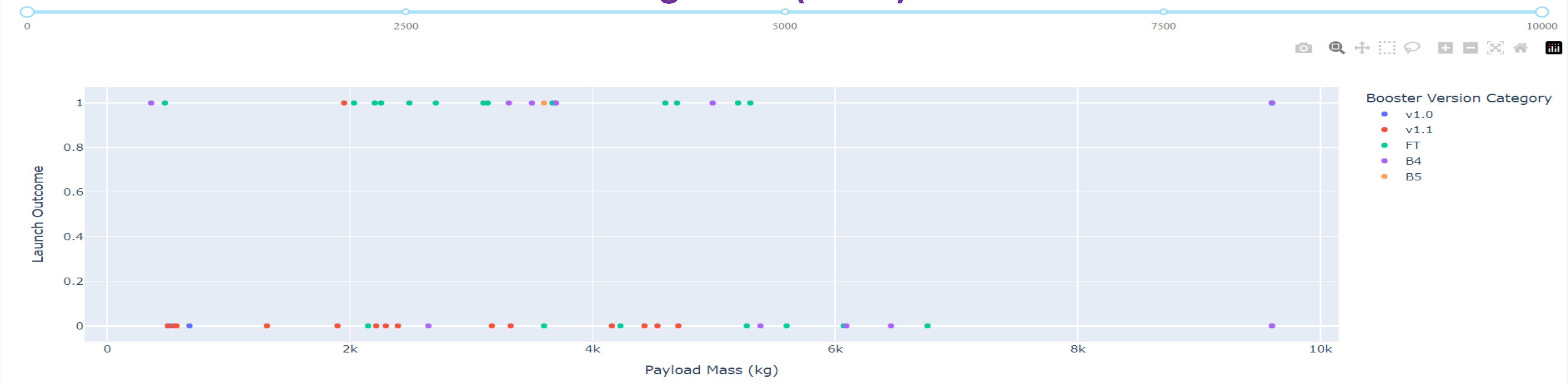
Launch Site with Highest Success Rate



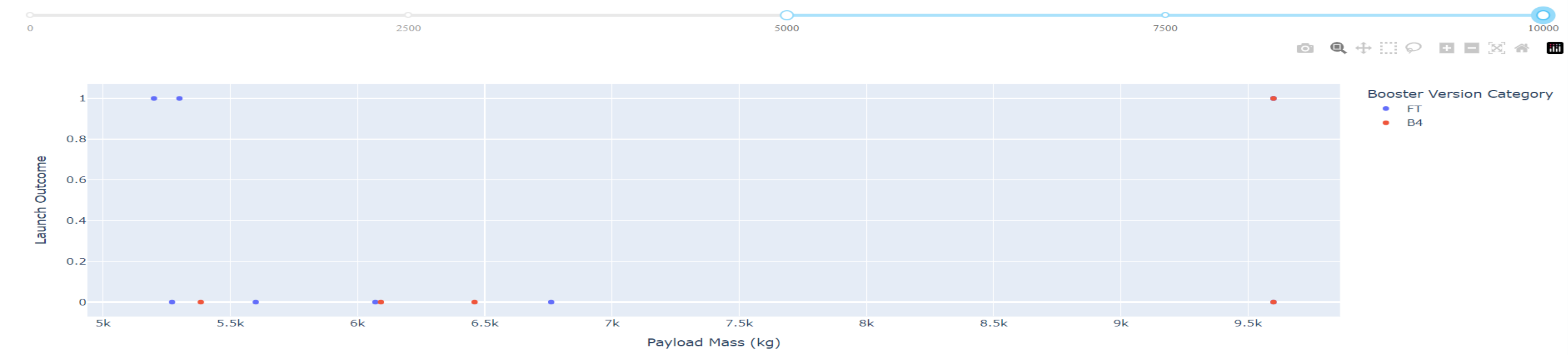
- KSC LC-39A is the Launch Site with Highest Success Rate which is 76.9%

Payload Vs. Launch Outcome

Slider Range – Full(0-10k)

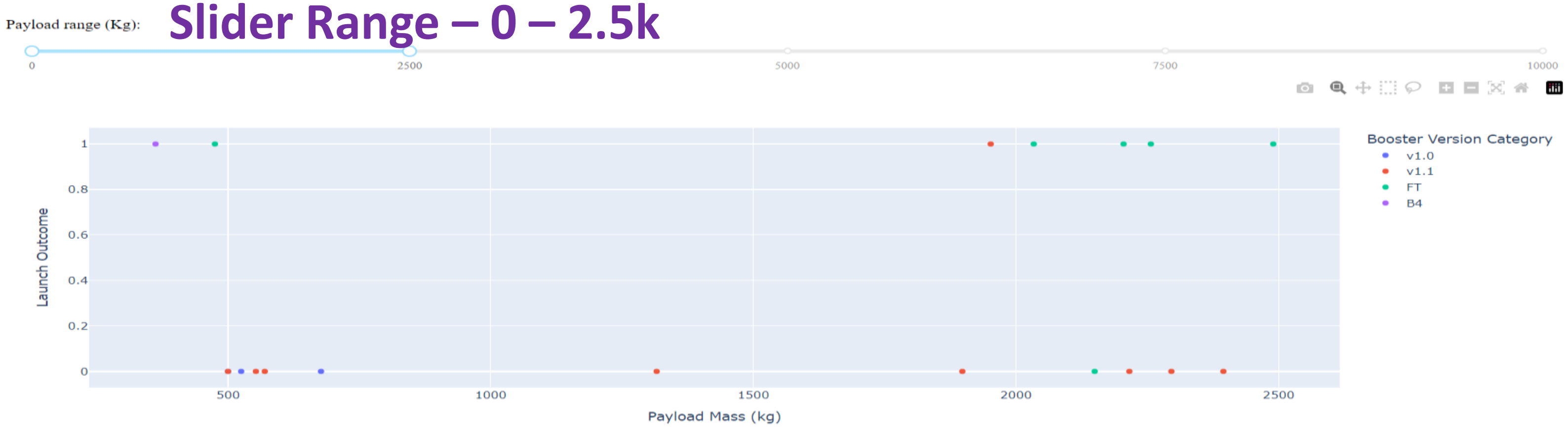


Slider Range – 5k -10k

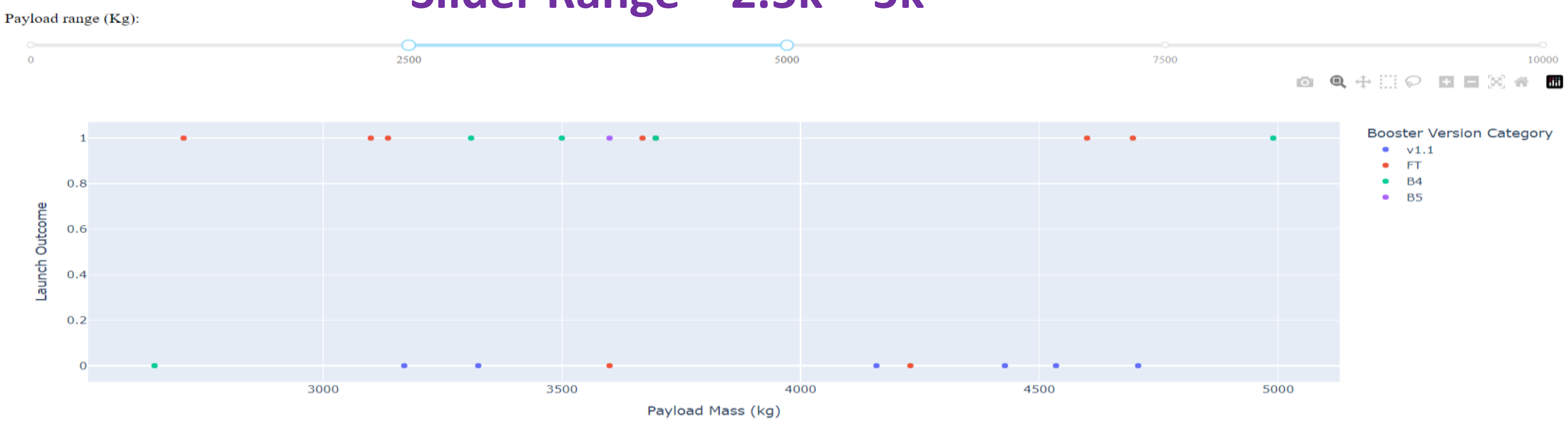


Payload Vs. Launch Outcome

Slider Range – 0 – 2.5k



Slider Range – 2.5k – 5k

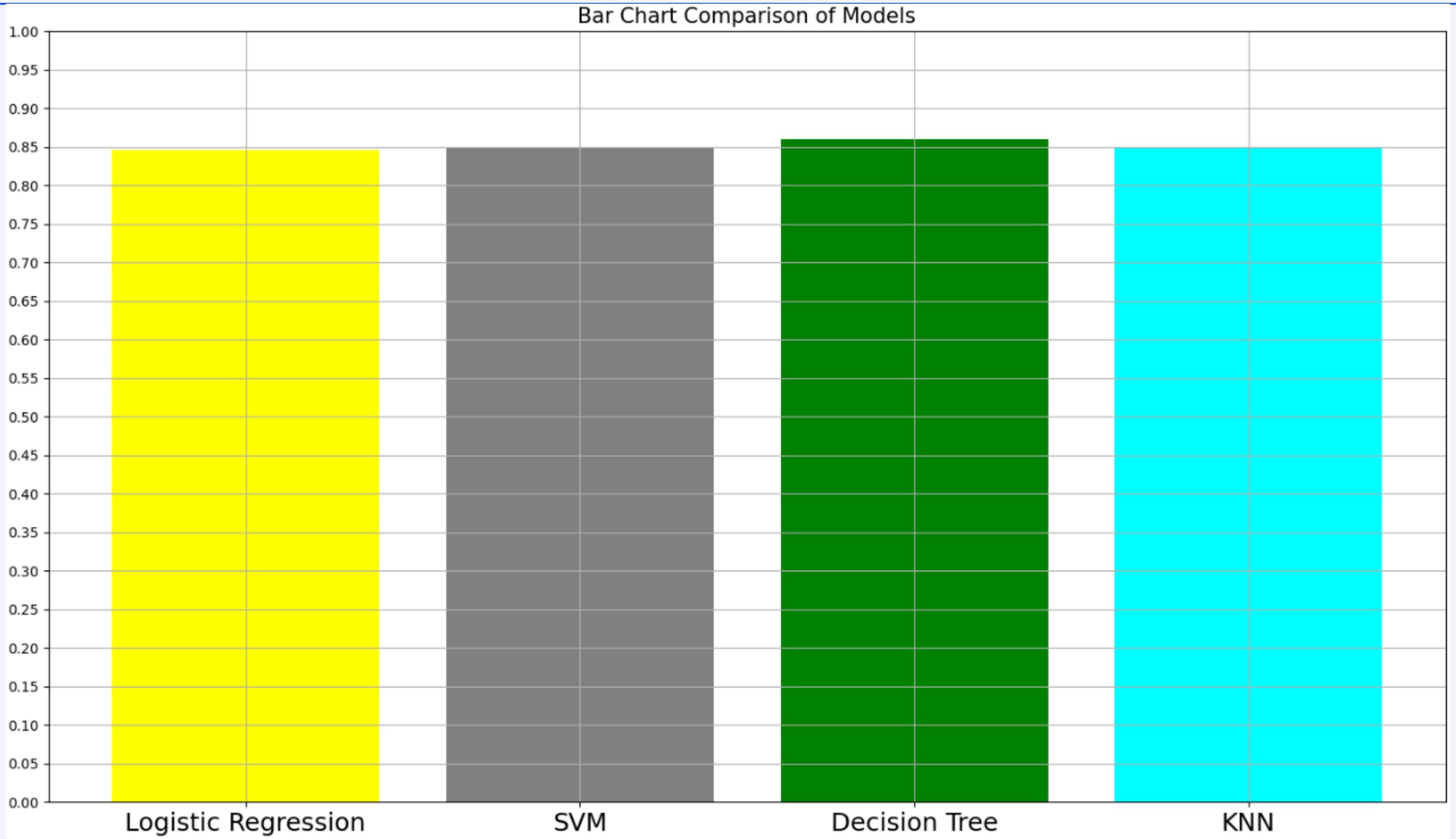


- Success rate is very low in the payload range 5k – 10k
- Booster versions BT and F4 are used in this payload range.
- In the range 0 – 2.5k we could see many success for FT Booster Version

Section 5

Predictive Analysis (Classification)

Classification Accuracy



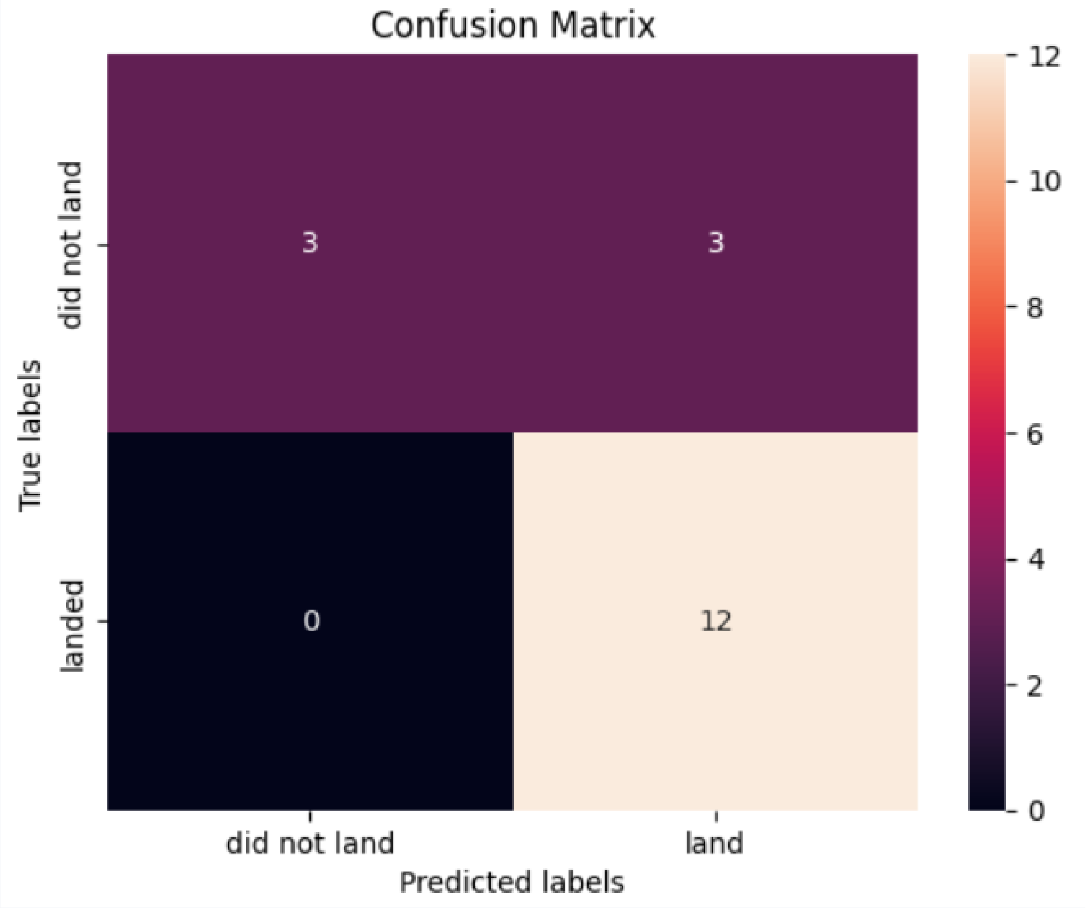
- Decision Tree Model has the highest classification accuracy

Confusion Matrix

- Out of 18, 3 predictions were wrong
- These 3 were false positives which predicted a successful landing
- Precision (P) – TP / (TP + FP) = 12/12+3 = 0.80
- Recall (R) – TP / (TP + FN) = 12 / (12+0) = 1
- F1 score – 2PR / (P+R)

  \- 2 x (0.8 x 1) / (0.8 + 1)

  = 0.888



Conclusions

- When the experiment started in with first stage return, we could see many failures. But after the few years the success rate in increased steeply to almost no failure.
- For the orbits ES-L1, GEO, HEO and SSO the success rate is 100%
- For GTO orbit the success rate is below 50%
- For orbits, LEO, ISS and PO, lower the payload, higher the failures
- KSC LC-39A is the Launch Site with Highest Success Rate
- All launch sites are in good proximity to coastlines and equator. There is good connectivity to roads, highway and railways
- Decision Tree Model had the highest classification accuracy
- As a concern, there are some false positives in prediction

Improvements

- Concerns
  - All the wrong predictions are Type I errors
- Improvements
  - We need to improve the model to avoid Type I errors
- Suggestions
  - Do more analysis with more data, different machine algorithms and techniques

Appendix

- TP – True Positive
- FP – False Positive
- FN – False Negative
- P – Precision
- R – Recall

Thank you!