IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Anna Liu
7th June 2022

Outline

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

Executive Summary

- The aim of the project is build a ML model that can predict if the first stage of Falcon 9 will land successfully. Data is first scraped and cleaned from the SpaceX data API and Wikipedia. The data is then processed using SQL and visualized through dashboard and finally a no. of different ML models are tested with the model having the highest accuracy chosen.
- In the end, the best ML model resulted in a accuracy of 83.3% based on test and training data.

Introduction

- SpaceX's Falcon 9 rocket launches cost 62 million dollars compared to the upward cost of 165 million dollars of other providers mainly due to its reusable first stage rocket.
- The aim is to build a ML model that can predict if a rocket will land based on a number of features made available and thus estimate the cost of a launch, allowing alternate companies to bid against SpaceX for a rocket launch.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data is collected from the SpaceX Data API at https://api.spacexdata.com and Wikipedia
- Perform data wrangling
  - A numeric class column is added based on the plethora of possible landing outcome 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
  - Logistic Regression, SVM, Decision Tree, and KNN ML models were built and fitted to the collected data and compared based on accuracy testing on a separate testing data.

Data Collection

- The data is collected by making use of a number of endpoints available at the SpaceX Data API, and working on the data using Pandas to extract relevant column or sub-column
- The general flow used was:
  - Query the API using requests
  - Reshape the data into a Pandas.DataFrame
  - Extracting only the Falcon 9 Launch data
  - Replacing all empty Payload Mass data with the mean
  - Creating a new 'class' numeric column that can be used by models from the various landing outcome values

Data Collection – SpaceX API

- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Flowchart:

- Query the API using requests
- Reshape the data into a Pandas.DataFrame
- Extracting only the Falcon 9 Launch data
- Replacing all empty Payload Mass data with the mean
- Save result dataset to CSV file

Data Collection - Scraping

- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/jupyter-labs-webscraping.ipynb

Flow Chart:

- Retrieve HTML content through requests.get from Wikipedia
- Parse HTML using BeautifulSoup
- Extract the relevant table
- Process the table row by row to get the relevant columns for each row
- Convert the dictionary of lists to a Pandas.DataFrame
- Save output to CSV File

Data Wrangling

- Flowchart
  - Get an idea of the data using various summary methods made available through Pandas
  - Making use of data in the textual Outcome column, create a new numeric column Class that has the two values
    - 0 – Landing Failure
    - 1 – Landing Success
  - Save the processed data to a CSV file
- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- The plots used include:
  - Scatterplot of Payload Mass vs Flight Number
  - Scatterplot of Launch Site vs Flight Number
  - Scatterplot of Launch Site vs Payload Mass
  - Bar chart of Success Rate of each Orbit
  - Scatterplot of Orbit vs Flight Number
  - Scatterplot of Orbit vs Payload Mass
  - Line chart of Success Rate vs Year
- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- The queries performed includes:
  - Get the names of each unique launch sites in the space mission
  - Get 5 records where the launch site begins with 'CCA'
  - Get the total payload mass carried by boosters launched by NASA (CRS)
  - Get the average payload mass carried by booster version F9 v1.1
  - Get the date when the first successful landing outcome in ground pad was achieved
  - Get names of boosters which have success in drone ship and payload mass between 4000 and 6000
  - Get total number of successful and failure mission outcomes
  - Get names of booster_versions which have carried the maximum payload mass
  - Get the booster version and launch site names for failed landing_outcomes in drone ship in 2015
  - Rank the landing outcome counts between 2010/06/04 and 2017/03/20 in descending order
- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/jupyter-labs-eda-sql-coursera.ipynb

Build an Interactive Map with Folium

- Circle, Marker, MarkerCluster, MousePosition, Polyline objects were added to a Folium Map
- These objects were added to the map to represent the launch sites, landing success or failure, measure the distance between landmarks and the launch sites, and represent the distance between landmarks and the launch sites respectively
- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- A piechart and a scatterplot, together with a dropdown list with options for launch sites and a range slider representing the payload range were added to the Dashboard
- The piechart allows visualization of the successful rate at all/ each of the landing sites and the scatterplot allows us to visually the Success/ Failure vs Payload Mass for the selected payload range and launching sites.
- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

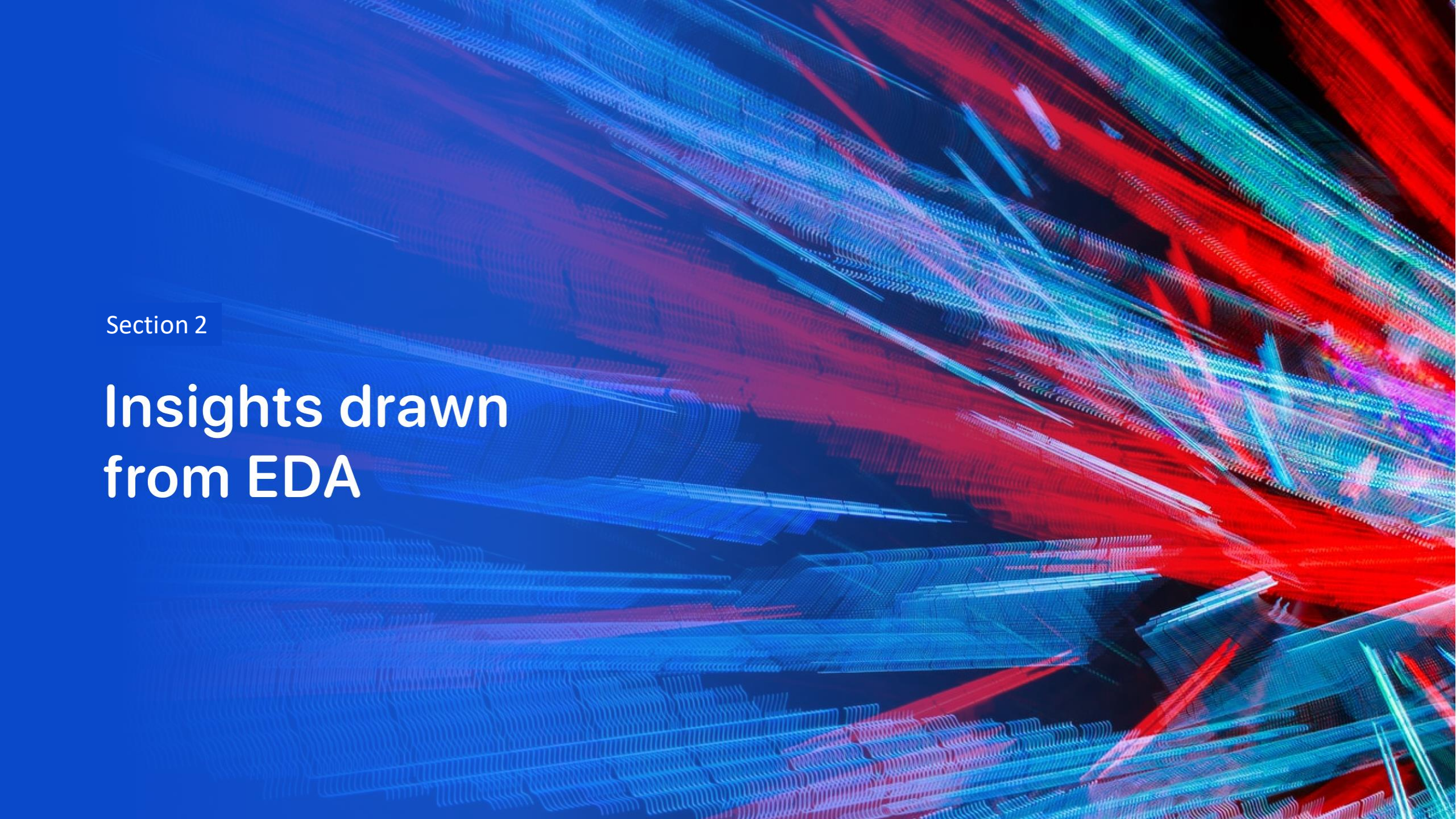
- 4 Different types ML models were build and fitted to the data using train test split and grid search with cross validation on the hyperparameters, with the best model chosen based on the accuracy of correct prediction on the test data.
- Flow chart:
  - Separate the data into X (Features), Y (Label)
  - Split the data into training and testing datasets using train_test_split
  - Find the best hyperparameters for Logistic Regression using GridSearchCV and training data
  - Plot the Confusion Matrix for Logistic Regression using the testing dataset
  - Find the best hyperparameters for SVM using GridSearchCV and training data
  - Plot the Confusion Matrix for SVM using the testing dataset
  - Find the best hyperparameters for Decision Tree Classifier using GridSearchCV and training data
  - Plot the Confusion Matrix for Decsion Tree Classifier using the testing dataset
  - Find the best hyperparameters for KNN using GridSearchCV and training data
  - Plot the Confusion Matrix for KNN using the testing dataset
  - Select the best model based on prediction accuracy result from the test data
- https://github.com/HakureiAnna/IBM-DS-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

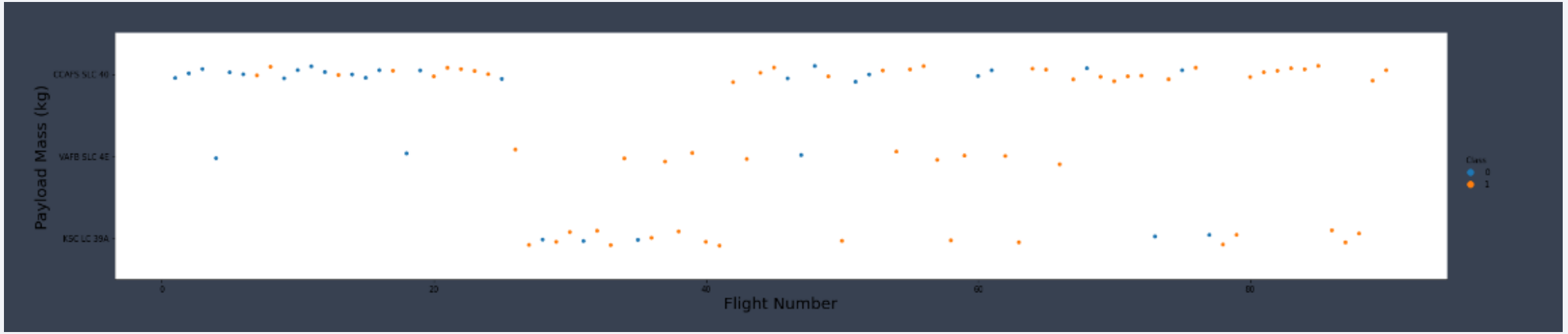
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

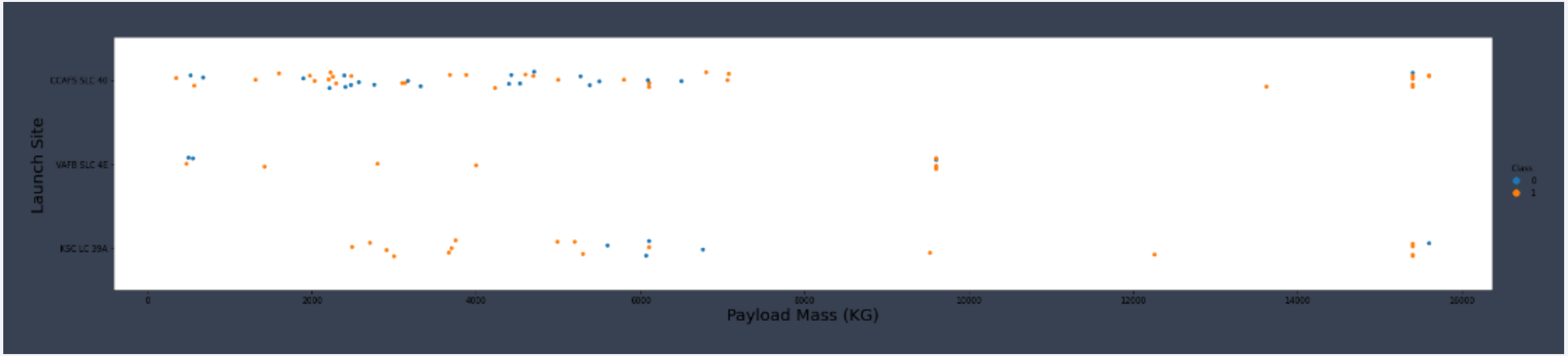


Flight Number vs. Launch Site



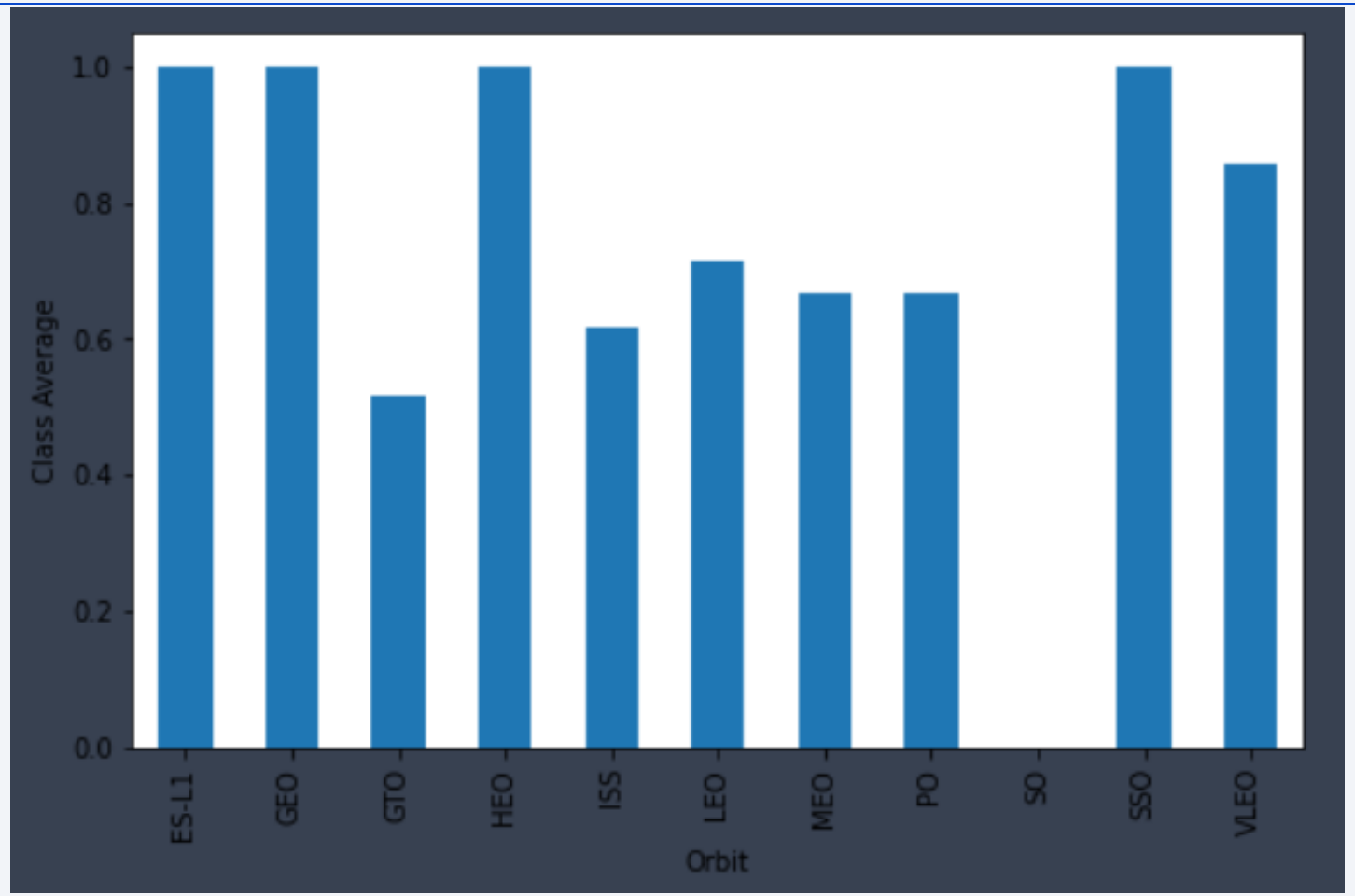
- Since flight number is sequential with respect to the launch date, we can see that majority of the flights are launched from CCAFS SLC 40, while there was a period in the middle where launches are held primary from KSC LC39A. Further, for VAFB SLC 4E, the launches are intermetant throughout the period observed.

Payload vs. Launch Site



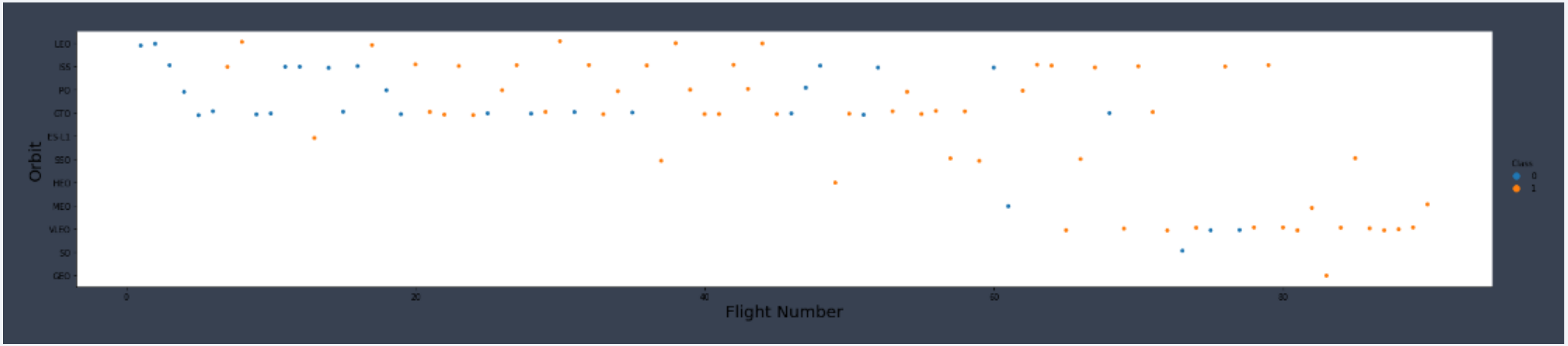
- Launchsite VAFB-SLC has no rockets launched for heavy payload mass (> 10000)

Success Rate vs. Orbit Type



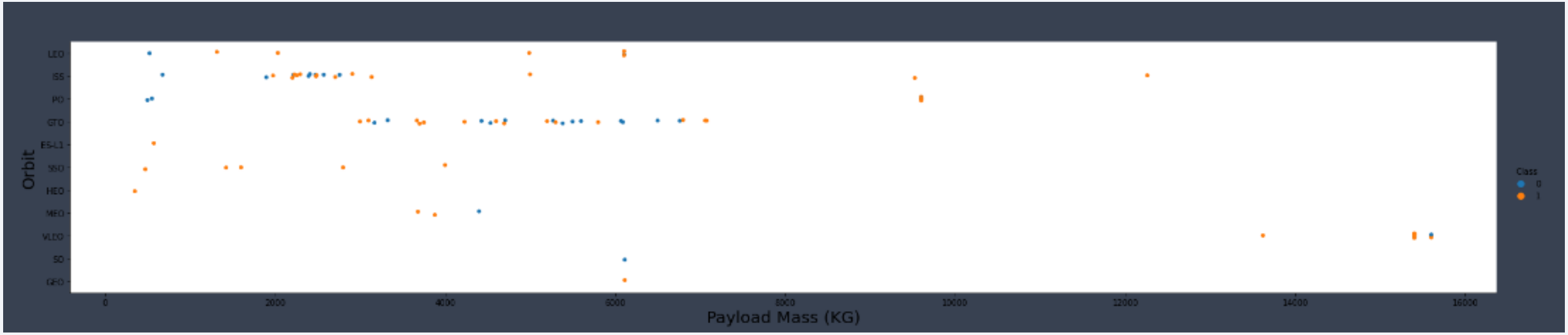
- Defining high success rate as > 80%, we can see that there's high sucess for orbits of ES-L1, GEO, HEO, SSO and VLEO.

Flight Number vs. Orbit Type



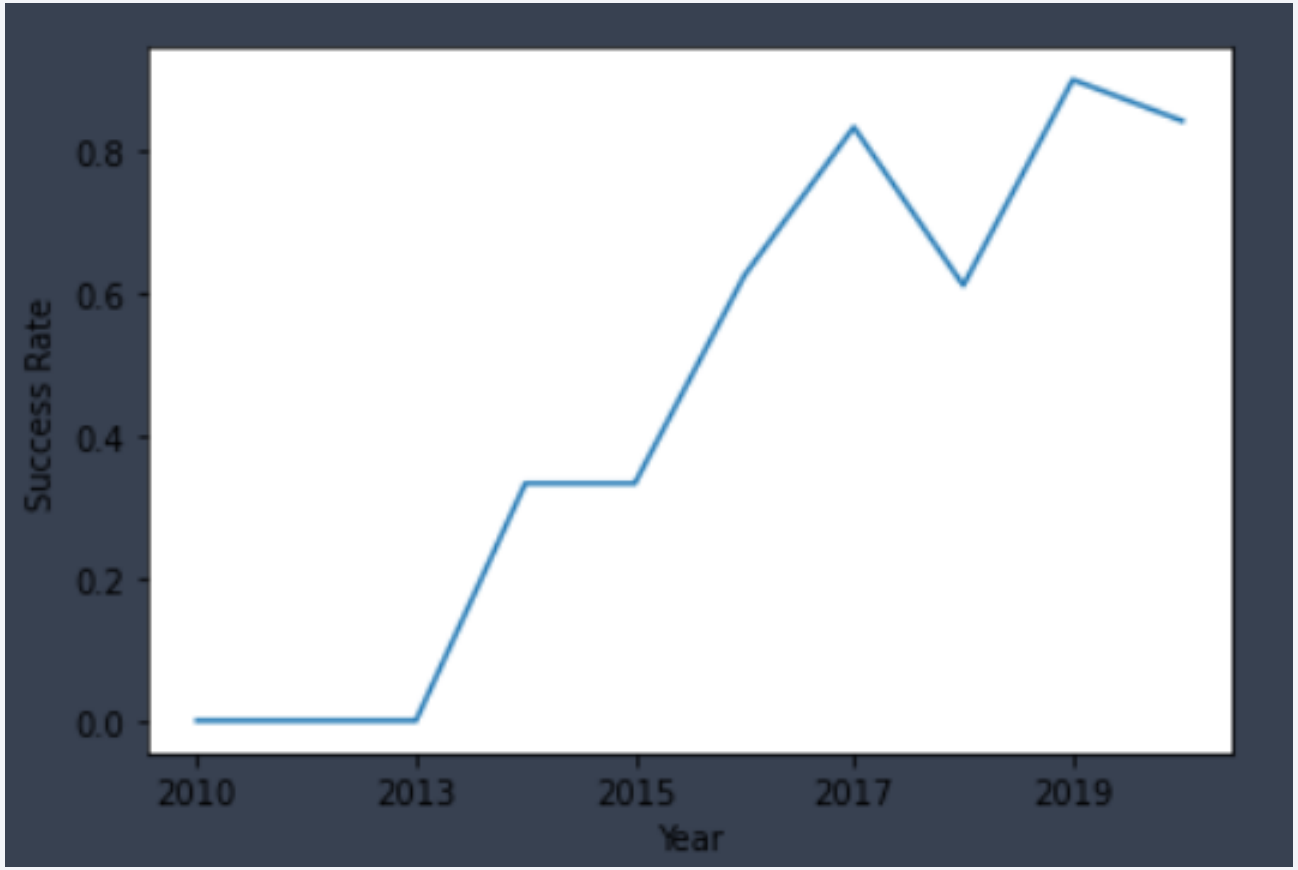
- In the LEO Oribit, the success seems to relate to the flight number. While for the GTO orbit, there seems to be no relationship between success and flight number.

Payload vs. Orbit Type



- 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



- Success rate since 2013 kept increasing until 2020

All Launch Site Names

- This can be obtained by a SELECT DISTINCT query on Launch_Site

Row	Launch_Site
1	KSC LC-39A
2	CCAFS LC-40
3	VAFB SLC-4E
4	CCAFS SLC-40

Launch Site Names Begin with 'CCA'

- This can be obtained using LIKE in WHERE clause and 5 in the LIMIT clause

Row	Date	Time__UTC_	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer
1	2013-12-03	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES
2	2014-01-06	22:06:00	F9 v1.1	CCAFS LC-40	Thaicom 6	3325	GTO	Thaicom
3	2014-08-05	08:00:00	F9 v1.1	CCAFS LC-40	AsiaSat 8	4535	GTO	AsiaSat
4	2014-09-07	05:00:00	F9 v1.1 B1011	CCAFS LC-40	AsiaSat 6	4428	GTO	AsiaSat
5	2015-03-02	03:50:00	F9 v1.1 B1014	CCAFS LC-40	ABS-3A Eutelsat 115 West B	4159	GTO	ABS Eutelsat

Total Payload Mass

- This can be obtained with SUM on PAYLOAD_MASS__KG and setting the WHERE clause on Customer

Row	TOTAL_PAYLOAD_MASS_KG
1	45596

Average Payload Mass by F9 v1.1

- This can be obtained using AVG function on PAYLOAD_MASS__KG and setting a WHERE clause on Booster_Version

Row	AVERAGE_PAYLOAD_MASS_KG
1	2928.4

First Successful Ground Landing Date

- This can be obtained using MIN on Date and setting the WHERE clause on Landing__Outcome

Row	Earliest_Ground_Landing_Success_Date
1	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- This can be obtained by using DISTINCT on Booster_Version and setting the WHERE clause on Payload_Mass__KG

Row	Booster_Version
1	F9 FT B1030
2	F9 FT B1021.2
3	F9 FT B1031.2
4	F9 B5 B1047.2
5	F9 FT B1032.1
6	F9 B4 B1040.1
7	F9 v1.1
8	F9 v1.1 B1011
9	F9 v1.1 B1014
10	F9 v1.1 B1016
11	F9 FT B1020
12	F9 FT B1022
13	F9 FT B1026
14	F9 B5 B1051.2
15	F9 FT B1032.2
16	F9 B4 B1040.2

Total Number of Successful and Failure Mission Outcomes

- This can be obtained by using COUNT on CASE selecting between Mission_Outcome of success or failure

Row	Failures_Count	Success_Counts
1	1	100

Boosters Carried Maximum Payload

- This can be obtained using a subquery to obtain the maximum Payload_Mass__KG and selecting the DISTINCT Booster_Version in the main query

Row	Booster_Version
1	F9 B5 B1048.5
2	F9 B5 B1051.4
3	F9 B5 B1060.2
4	F9 B5 B1058.3
5	F9 B5 B1051.6
6	F9 B5 B1048.4
7	F9 B5 B1049.4
8	F9 B5 B1051.3
9	F9 B5 B1056.4
10	F9 B5 B1049.5
11	F9 B5 B1060.3
12	F9 B5 B1049.7

2015 Launch Records

- This can be obtained by setting the WHERE clause on Landing__Outcome and getting the YEAR from the Date

Row	Booster_Version	Launch_Site
1	F9 v1.1 B1012	CCAFS LC-40
2	F9 v1.1 B1015	CCAFS LC-40

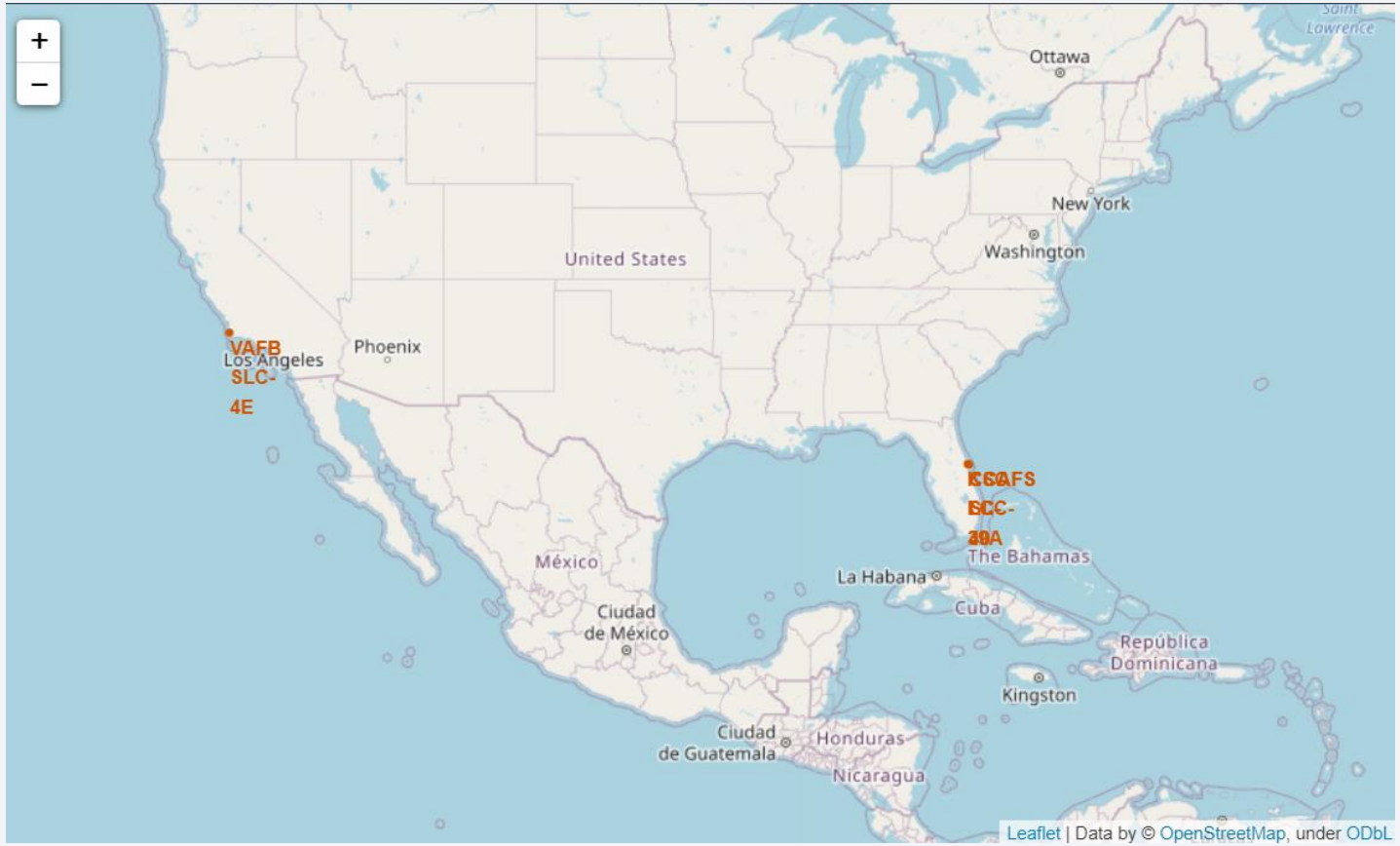
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- This can be obtained using the COUNT rows, setting the WHERE clause on Date and using GROUP BY on Landing__Outcome and finally ORDER BY DESC on COUNT row

Row	Landing__Outcome	Count
1	No attempt	10
2	Failure (drone ship)	5
3	Success (drone ship)	5
4	Success (ground pad)	3
5	Controlled (ocean)	3
6	Failure (parachute)	2
7	Uncontrolled (ocean)	2
8	Precluded (drone ship)	1

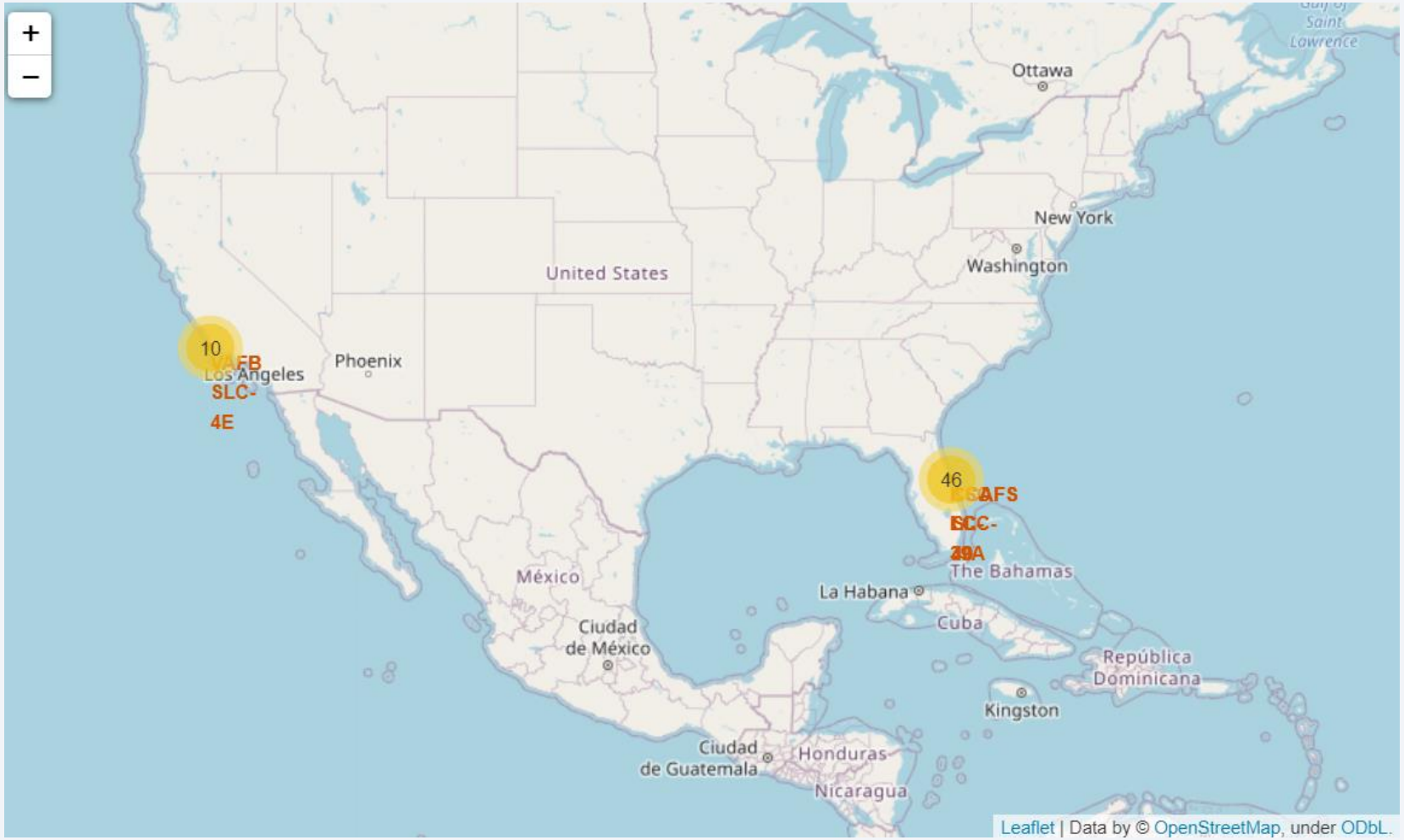

Section 3
Launch Sites
Proximities Analysis

Map of All Launch Sites



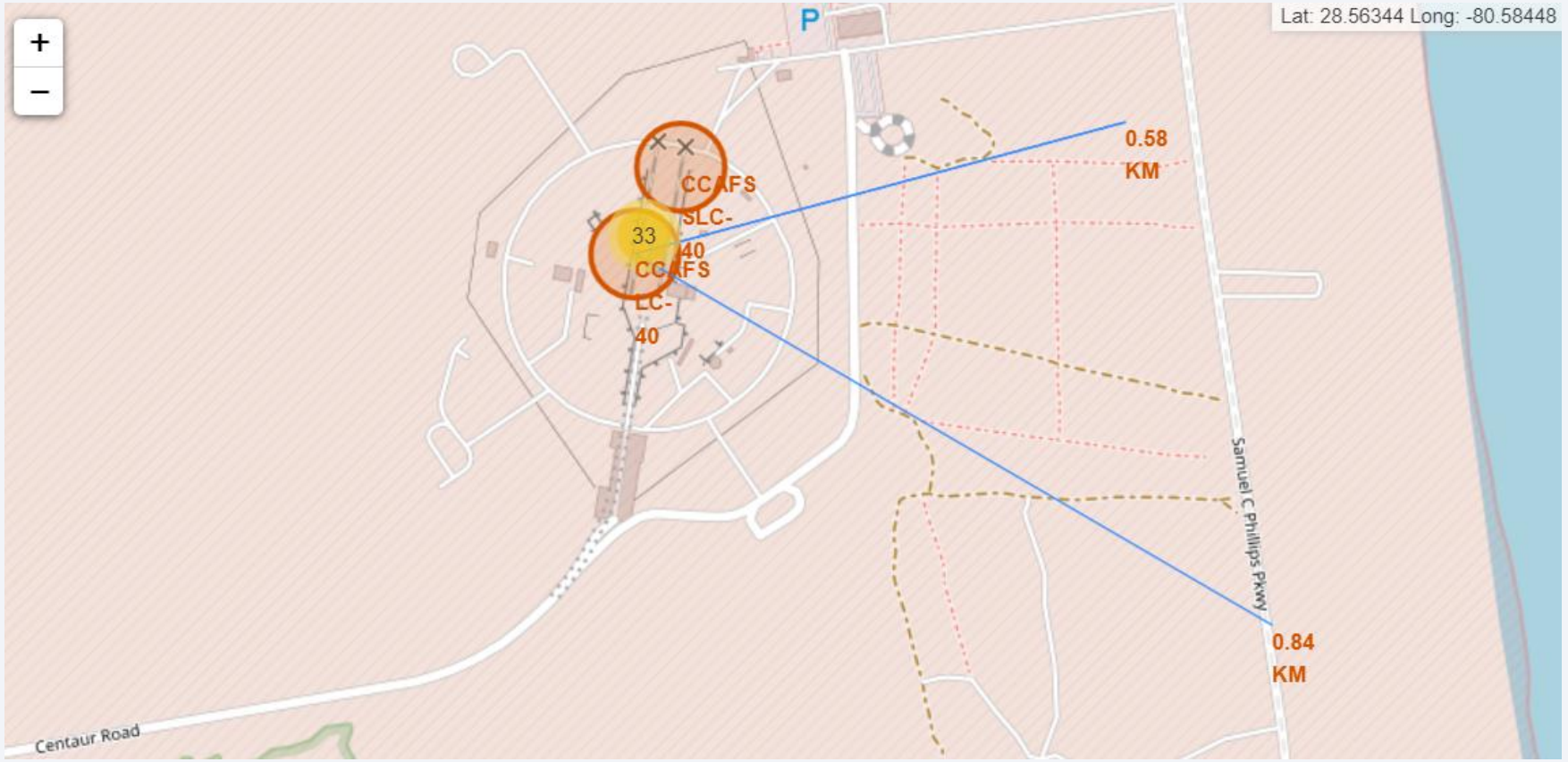
- This map shows all the launch sites we are analyzing on, with 1 on the top left and two (overlapped) in the bottom right portion on the map

Launch Site Map with Landing Success/ Failure

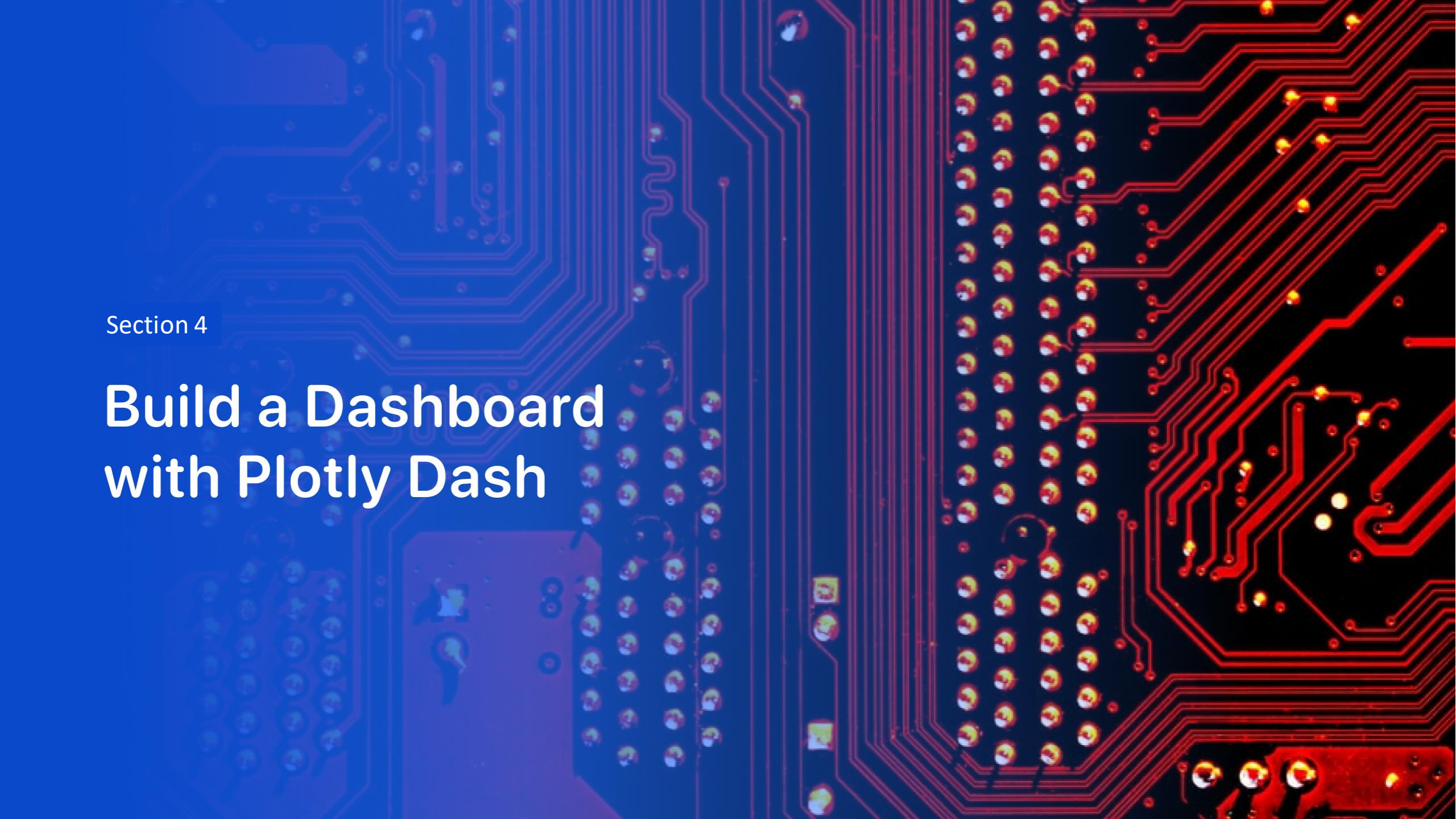


- This maps shows the number of landing success and failures and can be further distinguished by zooming in

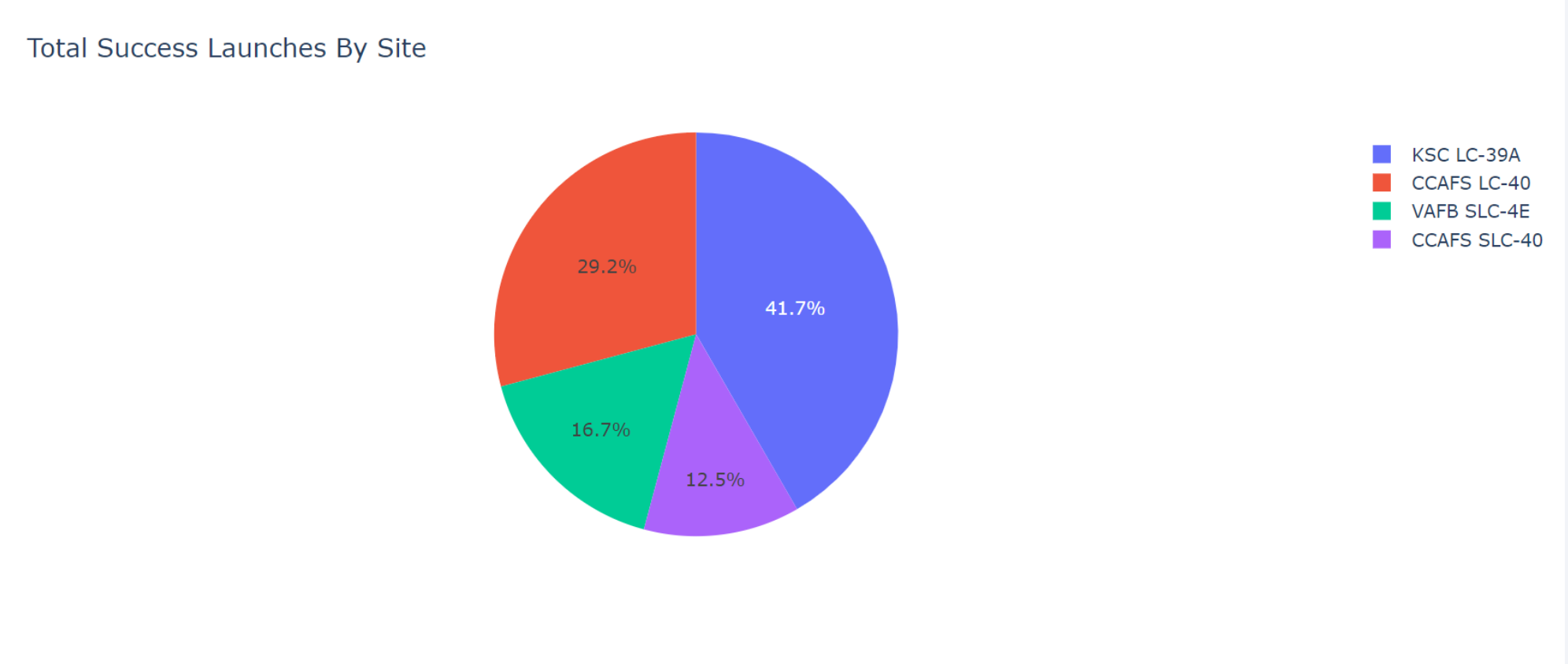
Map showing distance between launch site and landmarks



- This maps shows the distance between the launch site and nearby landmarks.

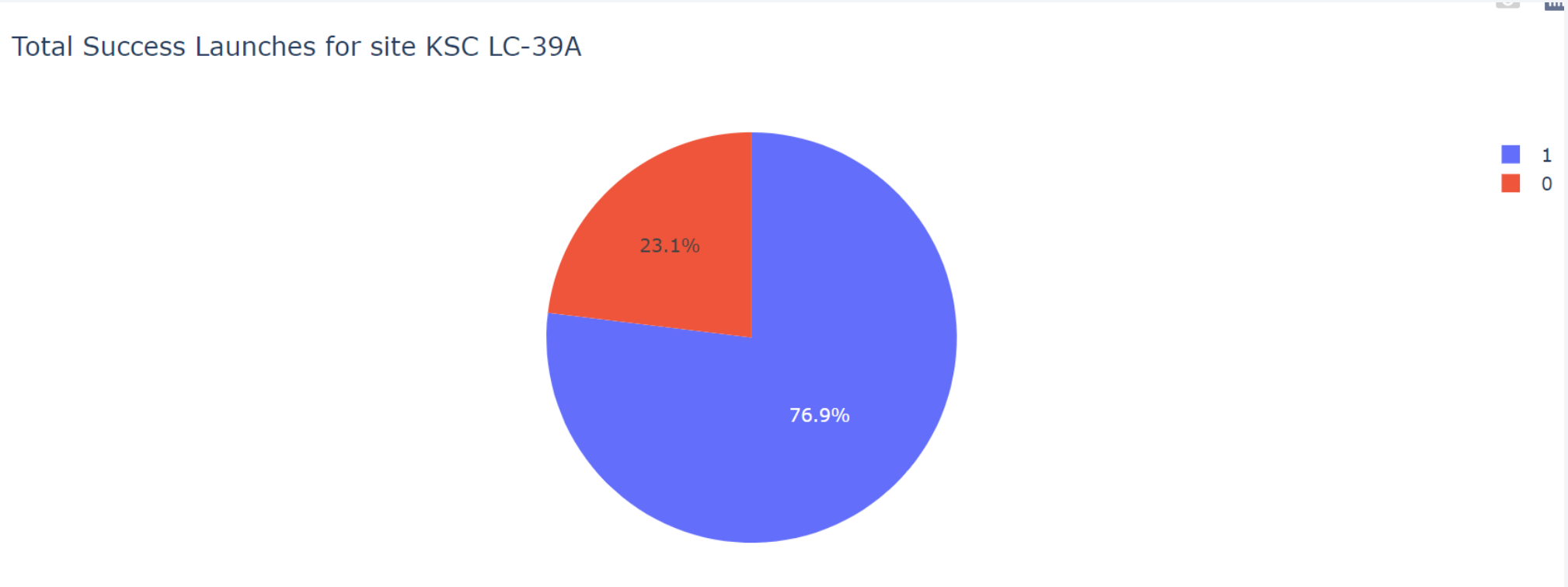

Section 4
Build a Dashboard with Plotly Dash

Piechart of All Launch Site in Dashboard



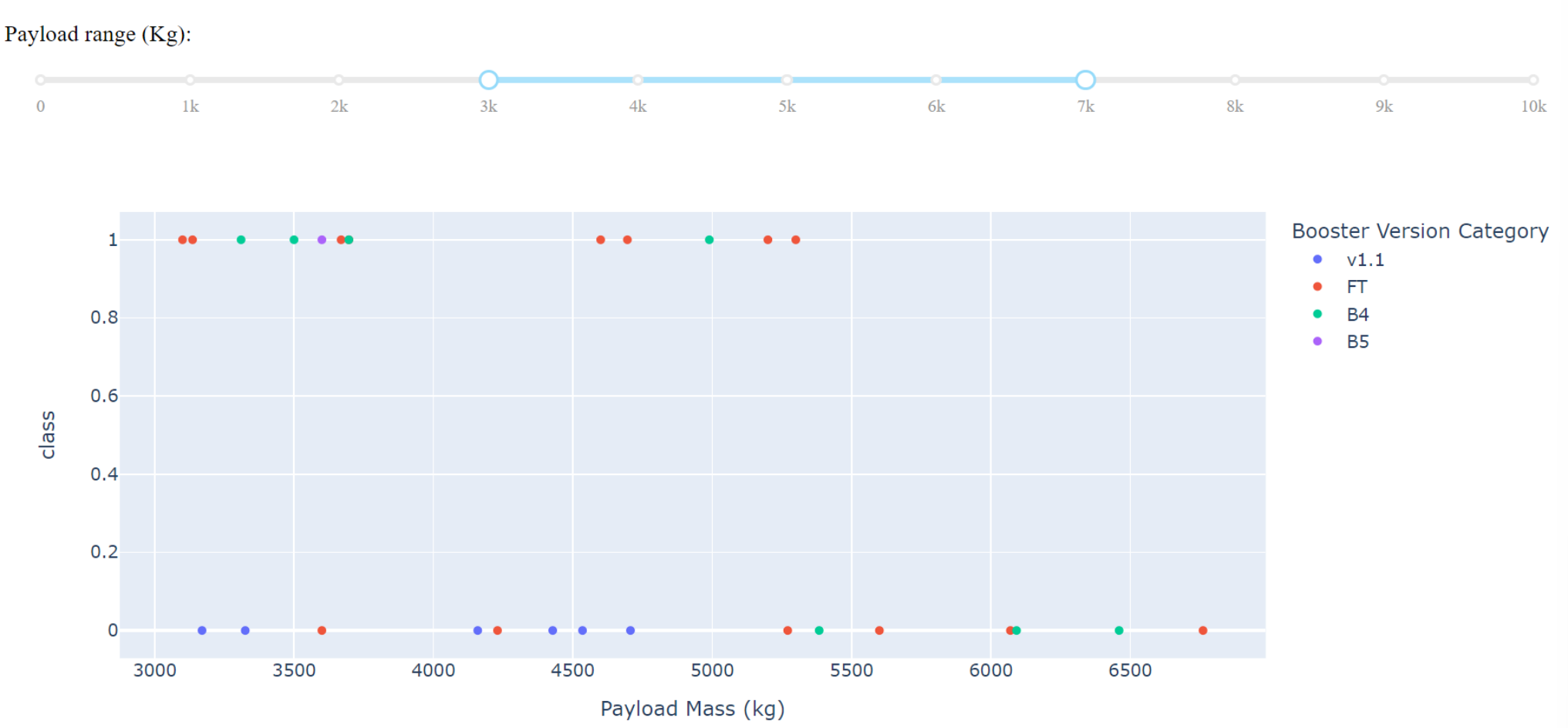
- Each partition on the pie chart shows the launch success rate (as percentage of all launches) at the individual launch site

Piechart for Site with Highest Launch Site on the Dashboard



- The blue portion represents successful launches while the red portion represents the failed launches.

Scatterplot of Launch Success/ Failure vs Payload Mass (kg)



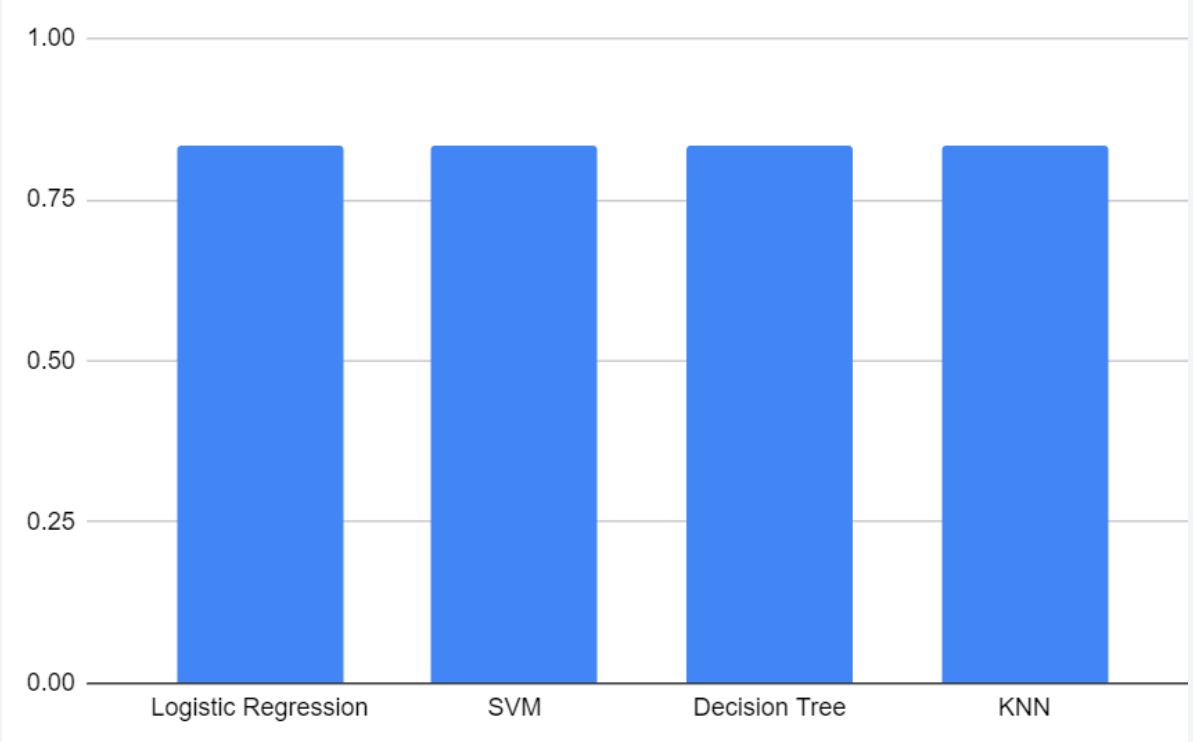
- The range slider allows dynamic configuration of the range of payload mass (kg) to visualize

Section 5

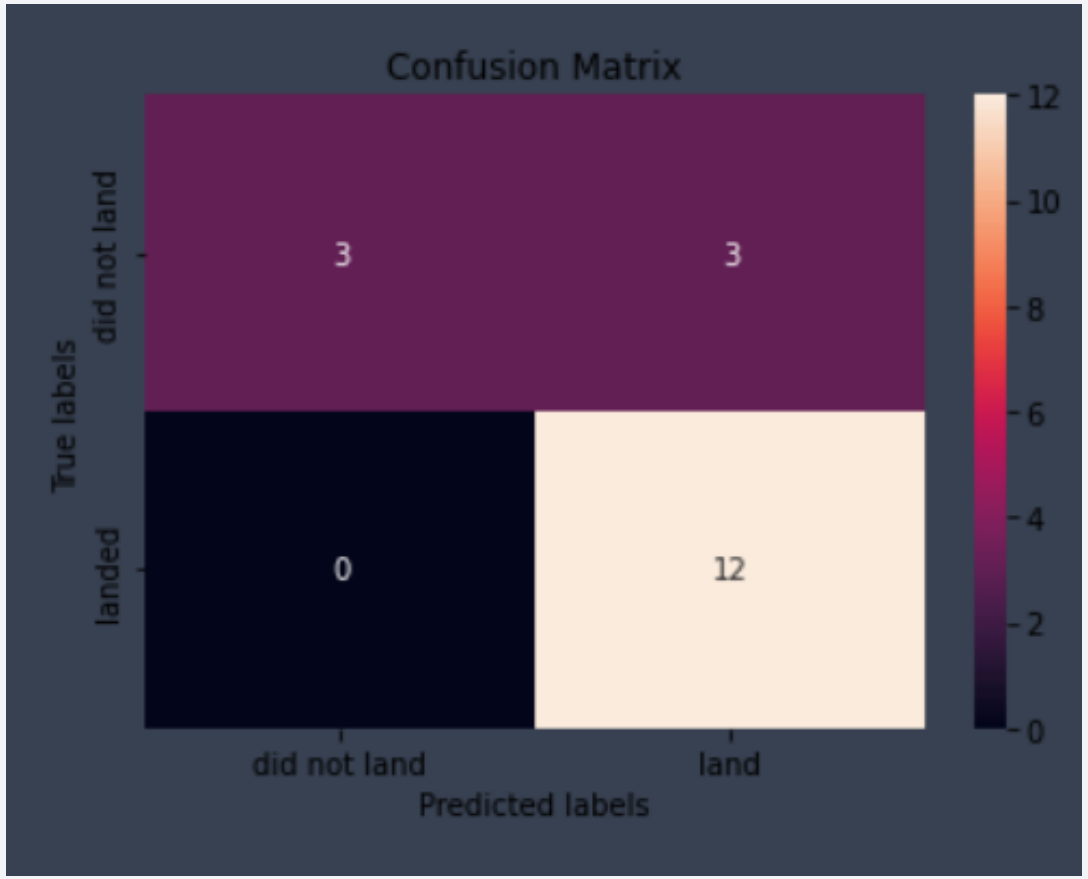
Predictive Analysis (Classification)

Classification Accuracy

- Due to the small dataset used for training and testing, all 4 models with best hyperparameters selected using GridSearchCV has the same accuracy



Confusion Matrix



- The confusion matrix shows that the best model predicts landed class correctly for all 12 samples while it is only accurate for did not land class 50% of the time

Conclusions

- Data obtained through APIs and webscraping need to be cleaned and wrangled in order for the ML models to be able to process them properly
- We gained a number of interesting insights into the data using SQL and Python for data processing
- The dashboard allows us to gain insights into the data by allowing us set dynamic criteria on the partition of the data we are actually visualizing
- We are able to obtain an accuracy of 83.3% with all models tested on a small dataset consisting of only 100 samples.
- We need more data to have a better gauge of the accuracy of the models and allow us to select the best model based on their different performances

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!