IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Diya Venugopal
30th July, 2023

Outline

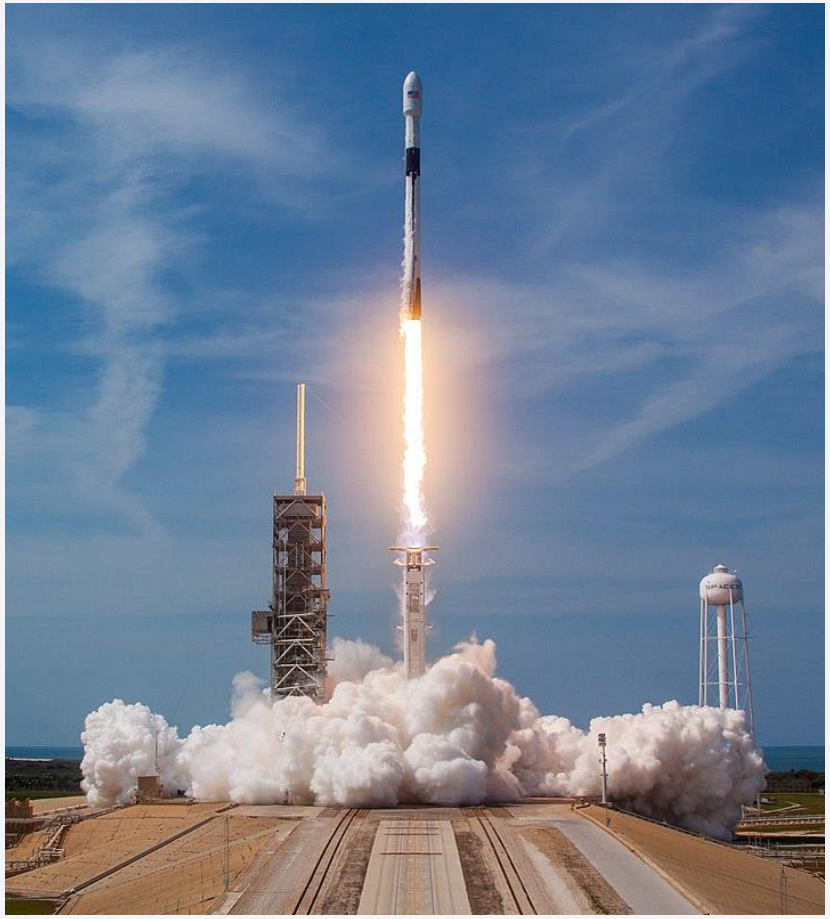
- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Summary of methodologies:
  - The required data was collected on SpaceX Falcon9 launches and landings using SpaceX API.
  - Web scraping was done from Wikipedia using BeautifulSoup.
  - Data wrangling was performed to obtain training labels.
  - Exploratory Data Analysis using SQL helped query the data to answer key questions.
  - Data Visualization using Pandas and Matplotlib helped understand the relationship between different variables.
  - Launch sites were visualized with outcomes using Folium.
  - Machine Learning models were built and optimized to pick the best prediction model.
- Summary of all results
  - Requested data was obtained from the API.
  - Data was cleaned and explored to understand the relationship between the variables and prepare the data for modeling.
  - Features were selected and accuracy from different machine learning models were compared to pick the best method.

Introduction

- Project background and context:
  - SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; whereas other providers cost upward of 165 million dollars each. Much of the savings is because SpaceX can reuse the first stage. If we can determine if the first stage will land, we can determine the cost of a launch. This information can be used by our client's company that wants to bid against SpaceX for a rocket launch.
- Problems you want to find answers:
  - In this project we will explore the variables behind a successful first stage landing and we will predict if the Falcon9 first stage will land successfully.



Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Rocket data was requested from the SpaceX API, cleaned and merged. Historical Falcon9 launch records were web scraped from a Wikipedia page using BeautifulSoup.
- Perform data wrangling
  - Data was explored using Pandas attributes to identify patterns in the data and determine training labels.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Different classification models were built and tuned using GridSearchCV and the accuracy scores and confusion matrices were compared.

Data Collection

1. Using an API:
   - Request rocket launch data from SpaceX API with the URL: https://api.spacexdata.com/v4/launches/past
     - The response contains massive information about SpaceX rocket launches.
   - Define following helper functions that will help us extract information from the response:
     - getBoosterVersion gets the booster name.
     - getPayloadData gets the mass of the payload and orbit that it is going to.
     - getLaunchSite gets the launch site, longitude, and latitude.
     - getCoreData gets the outcome & type of the landing, no. of flights with that core, etc.
2. Web scraping:
   - Use BeautifulSoup to web scrape Falcon 9 launch records from Wikipedia URL: https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
   - Define some helper functions to parse the web scraped HTML table and extract:
     - Data and time
     - Booster version
     - Landing status

Data Collection – SpaceX API

Task 1: Request and parse the SpaceX launch data using the GET request

To make the requested JSON results more consistent, we will use the following static response object for this project:

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_
```

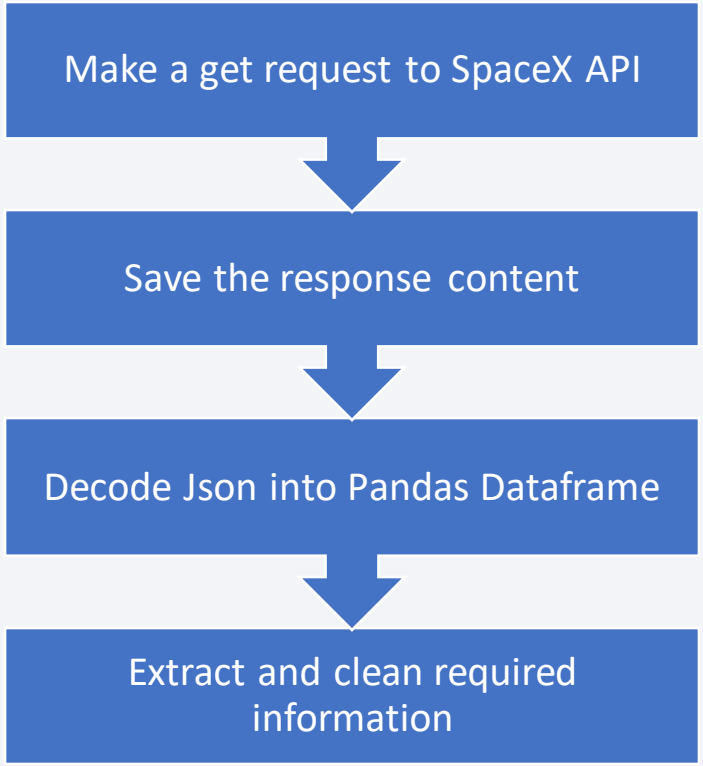
We should see that the request was successfull with the 200 status response code

```
response.status_code
```

```
200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

```
# Use json_normalize meethod to convert the json result into a dataframe
data=response.json()
data=pd.json_normalize(data)
```



- Make a get request to the API and save the response content.
- Convert the Json content into a data frame and extract the required data.
- GitHub URL of completed Data Collection with API notebook: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/1_Spacex_DataCollection_API.ipynb

Data Collection - Scraping

TASK 1: Request the Falcon9 Launch Wiki page from its URL

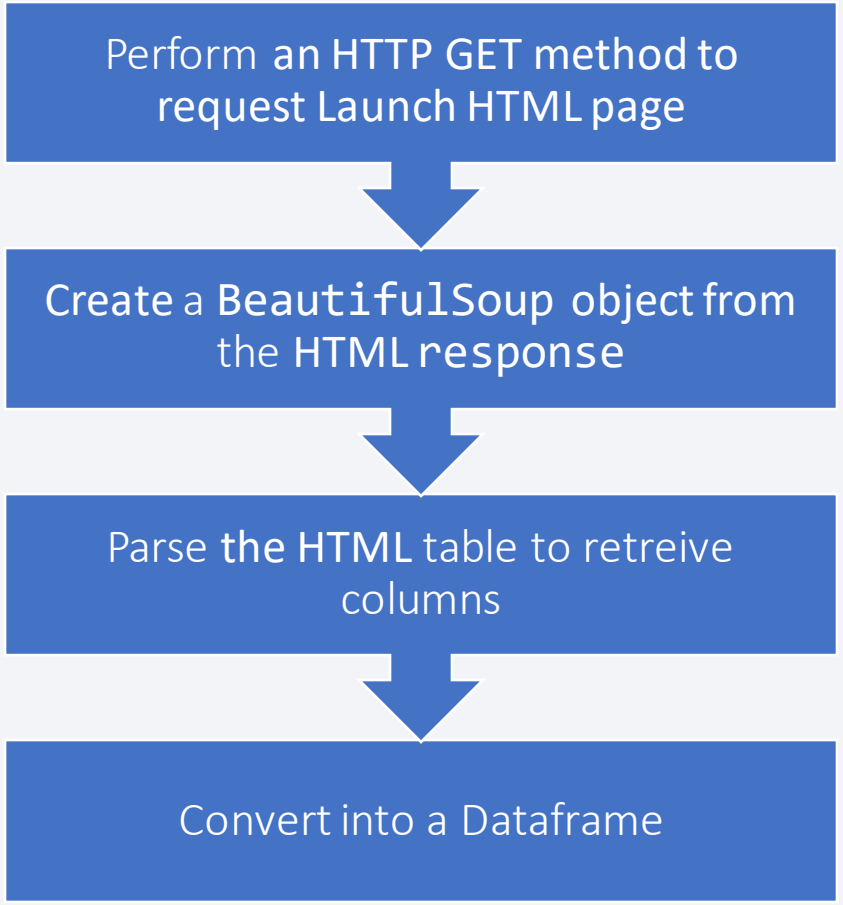
First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url)
```

Create a `BeautifulSoup` object from the HTML `response`

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
BeautifulSoup = BeautifulSoup(response.content, 'html.parser')
```

- GitHub URL of completed Web Scraping notebook: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/2_Spacex_Webscraping.ipynb



Data Wrangling

1. Initial Exploratory Data Analysis (EDA)
   - Identify and calculate proportion of Missing Values in each column
   - Review Data Types of attributes
   - Look at distribution of data using .value_counts()
2. Determine training labels as column Class where:
   - 1 means the booster successfully landed
   - 0 means the booster did not successfully land

GitHub URL of completed Data Wrangling notebook: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbff f2d12107cef58facc9e9e608e9a6/3_Spacex-Data%20wrangling.ipynb

EDA with Data Visualization

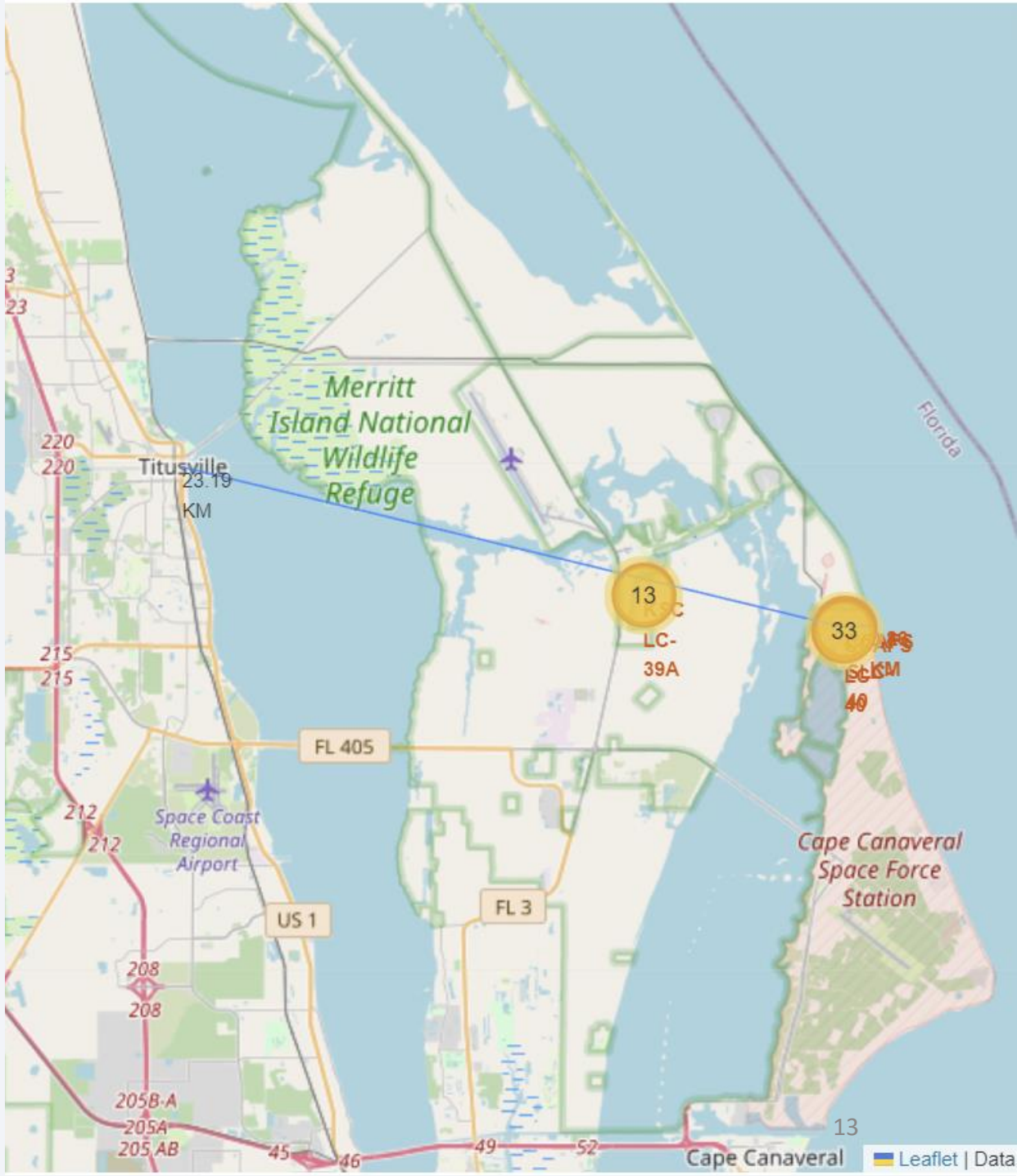
- Charts were plotted using pandas and matplotlib to understand how the outcome varies over different values of the variables.
  - Scatterplots between FlightNumber, PayloadMass and LaunchSite with Class as color helps us identify patterns and trends.
  - Bar chart helps us visualize the average success rate for each orbit type.
  - Line chart with x axis to be Year and y axis to be average success rate gets the average launch success trend.
- Select features to be used and create dummy variables for categorical variables and save the data as csv file.
- GitHub URL of completed EDA with Data Visualization notebook: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/5_Spacex-EDA-DataViz.ipynb

EDA with SQL

- Execute SQL Queries to answer the following questions:
  - Unique Launch sites
  - Some records where launch sites begin with the string 'CCA'
  - Total payload mass carried by boosters launched by NASA (CRS)
  - Average payload mass carried by booster version F9 v1.1
  - Date when the first successful landing outcome in ground pad was achieved
  - Boosters which have success in drone ship and have payload mass between 4000 to 6000
  - Total number of successful and failure mission outcomes
  - Booster versions which have carried the maximum payload mass
  - Failure landing outcomes in drone ship ,booster versions, launch site in 2015 months
  - Ordered count of landing outcomes between the date 2010-06-04 and 2017-03-20
- GitHub URL of completed EDA with SQL notebook: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/4_Spacex-EDA-SQL.ipynb

Build an Interactive Map with Folium

- Performed interactive visual analytics using Folium
- Created and added following map objects to a Folium Map with NASA as it's center:
  - A Circle and Marker for each launch site
  - A MarkerCluster with outcome based color markers for each launch record
  - A MousePosition to get coordinate for a mouse over a point on the map
  - A Marker to explore and show the distance of a launch site to any railway, highway, coastline, etc.
- GitHub URL of completed Folium Maps notebook: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/6_Spacex_LaunchSite_Location.ipynb

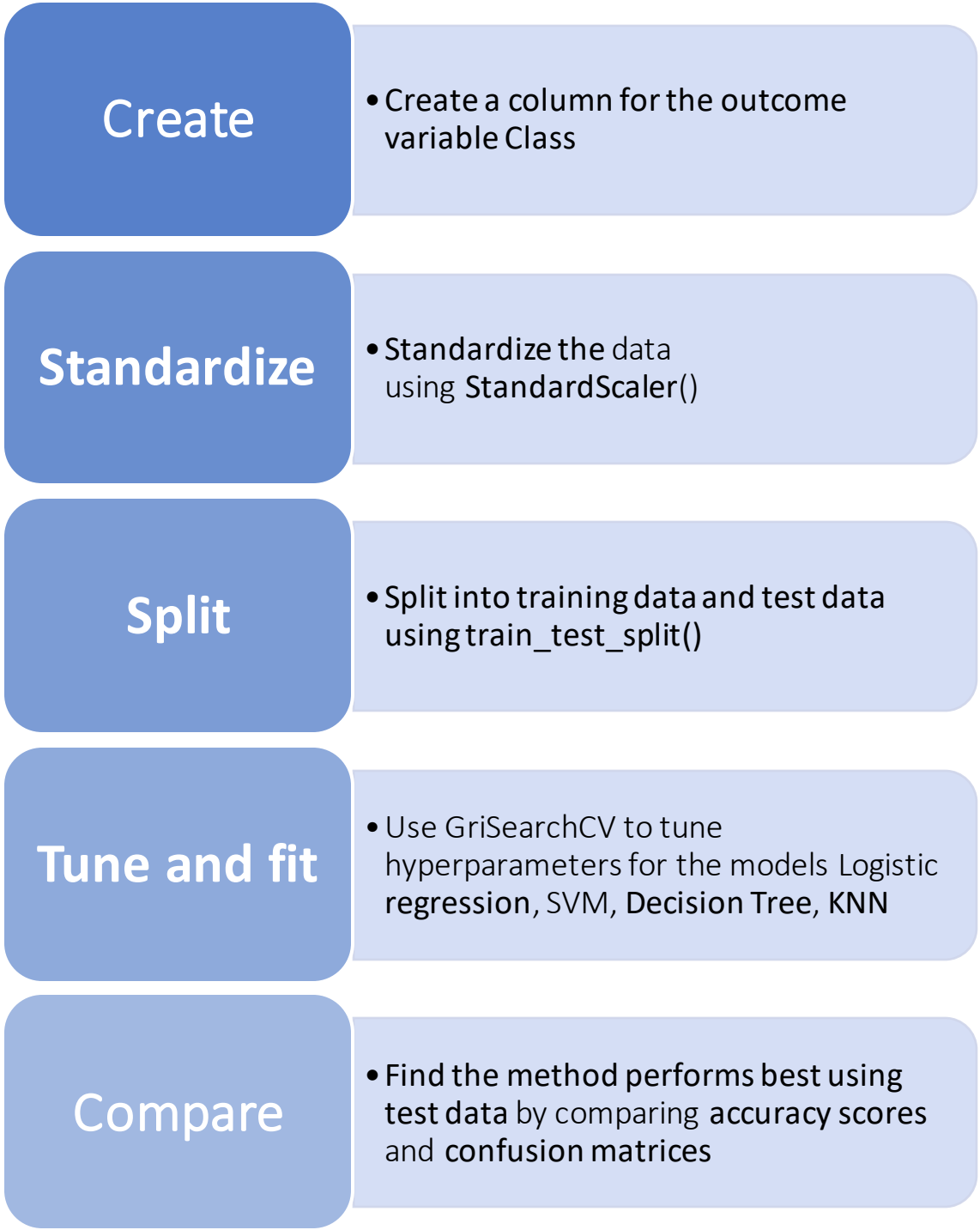


Build a Dashboard with Plotly Dash

- Build a Plotly Dash application for users to perform interactive visual analytics on SpaceX launch data in real-time.
- Add the following elements to the dashboard:
  - Launch Site Drop-down Input Component allows user to filter the charts by launch sites
  - Callback function renders success-pie-chart based on selected site dropdown
  - Range Slider to allow user to select a range of Payload values
  - Callback function to render the success-payload-scatter-chart scatter plot for the selected payload range
- GitHub URL of completed Plotly Dash lab: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/7_Spacex_PlotlyDash_App.py

Predictive Analysis (Classification)

- Create a machine learning pipeline to predict if the first stage will land given the data
- After exploratory Data Analysis and determining Training Labels, follow the steps as shown.
- GitHub URL of completed predictive analysis lab: https://github.com/Diyav/SpaceX_Falcon9_Launches/blob/98fc4d8cbbfff2d12107cef58facc9e9e608e9a6/8_SpaceX_ML_Prediction.ipynb

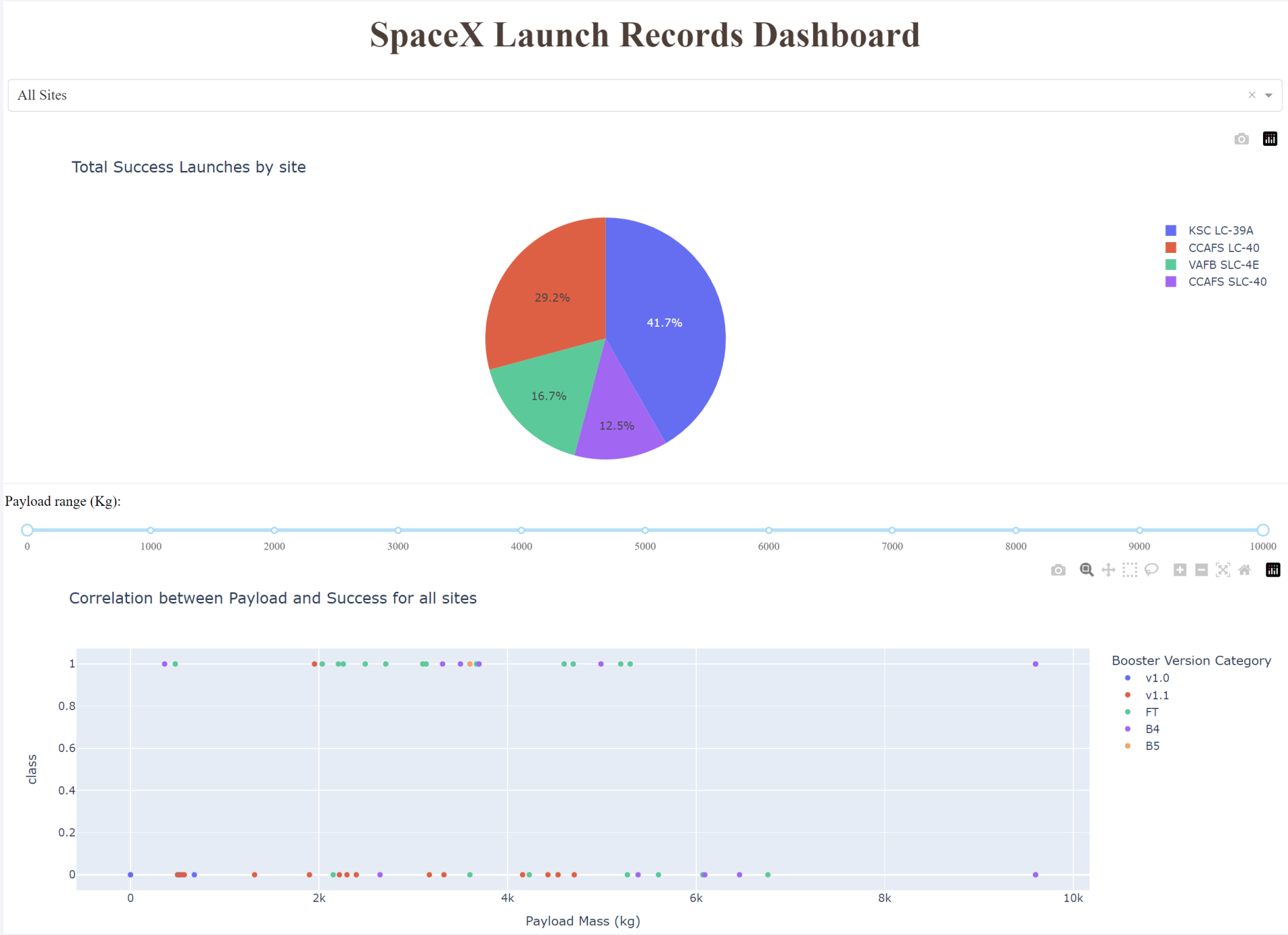


Results

- Exploratory data analysis results:
  - Different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%
  - ES-L1, GEO, HEO and SSO orbits have higher success rate.
  - 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.
  - With heavy payloads the successful landing rate is more for Polar, LEO and ISS.
  - Success rate since 2013 kept increasing till 2020.

Results (contd.)

- Interactive analytics result:
  - Plotly dashboard showing success rates by site and the correlation between payload and success rate.



Results (contd.)

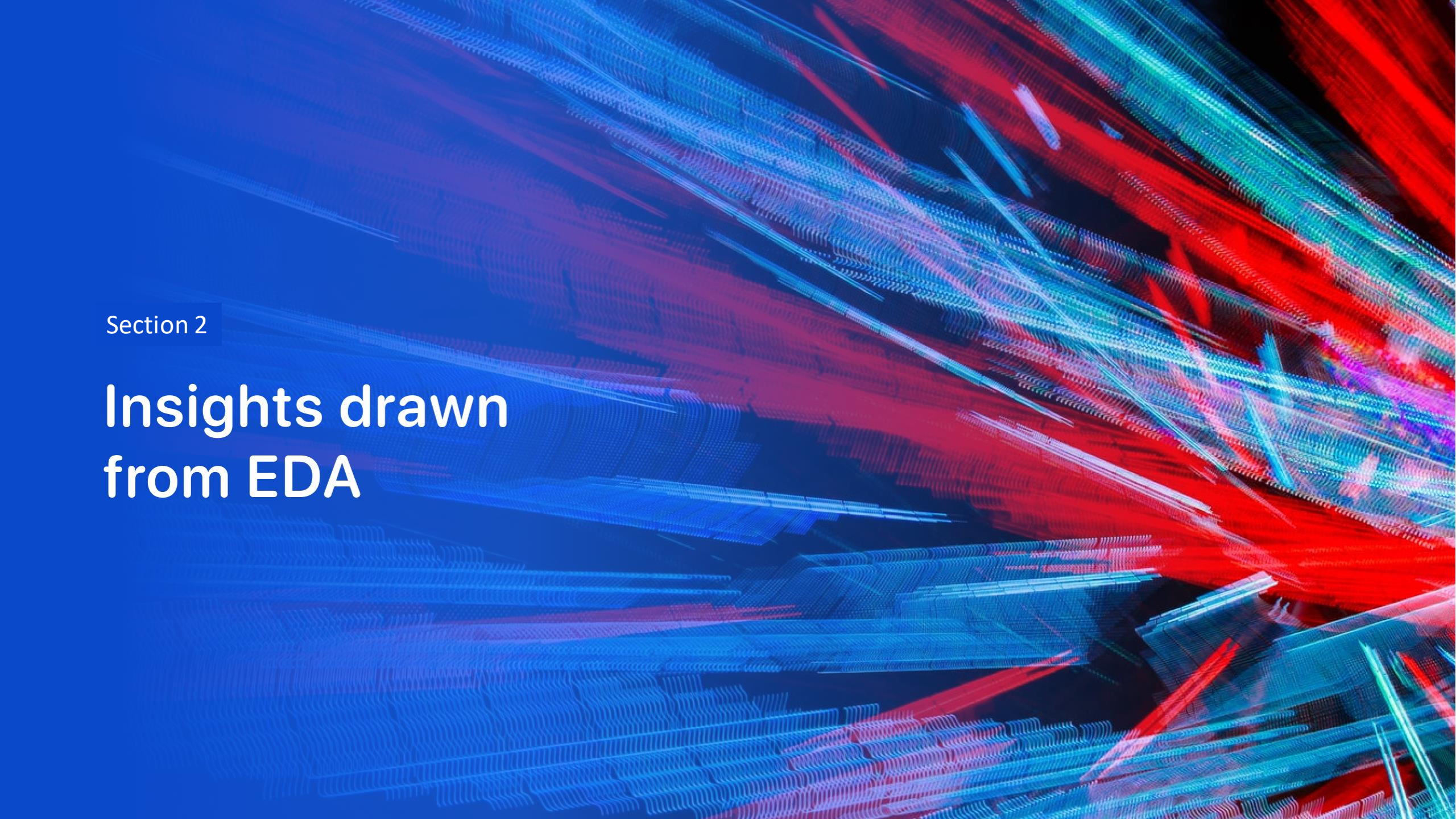
- Logistic regression, Support Vector Machine, Decision Tree and k Nearest Neighbors algorithms were used to build classification models and their accuracy scores were compared:

Method	Test Data Accuracy
Logistic_Reg	0.833333
SVM	0.833333
Decision Tree	0.833333
KNN	0.833333

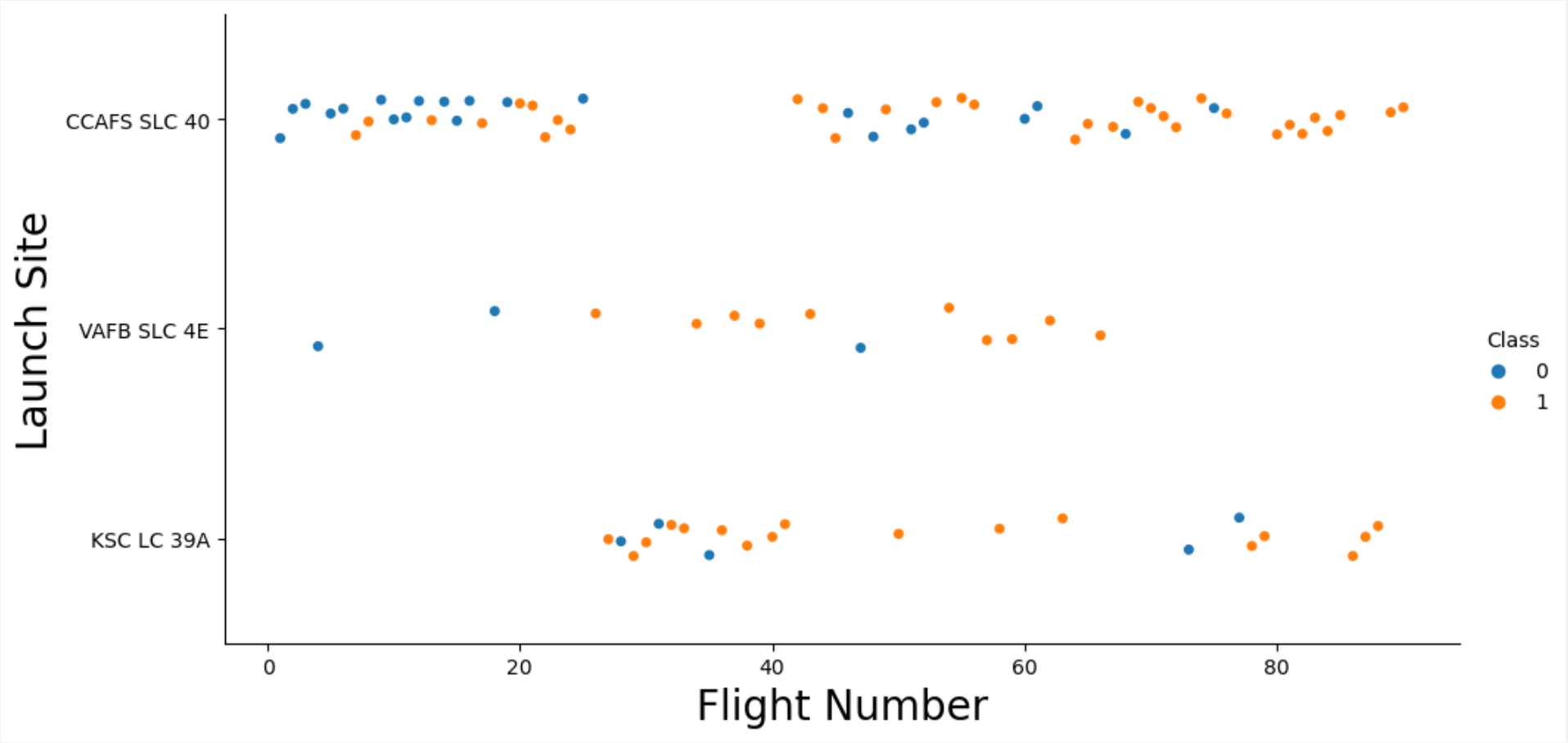
All the models have the same accuracy after tuning, hence they perform equally well.

Section 2

Insights drawn from EDA

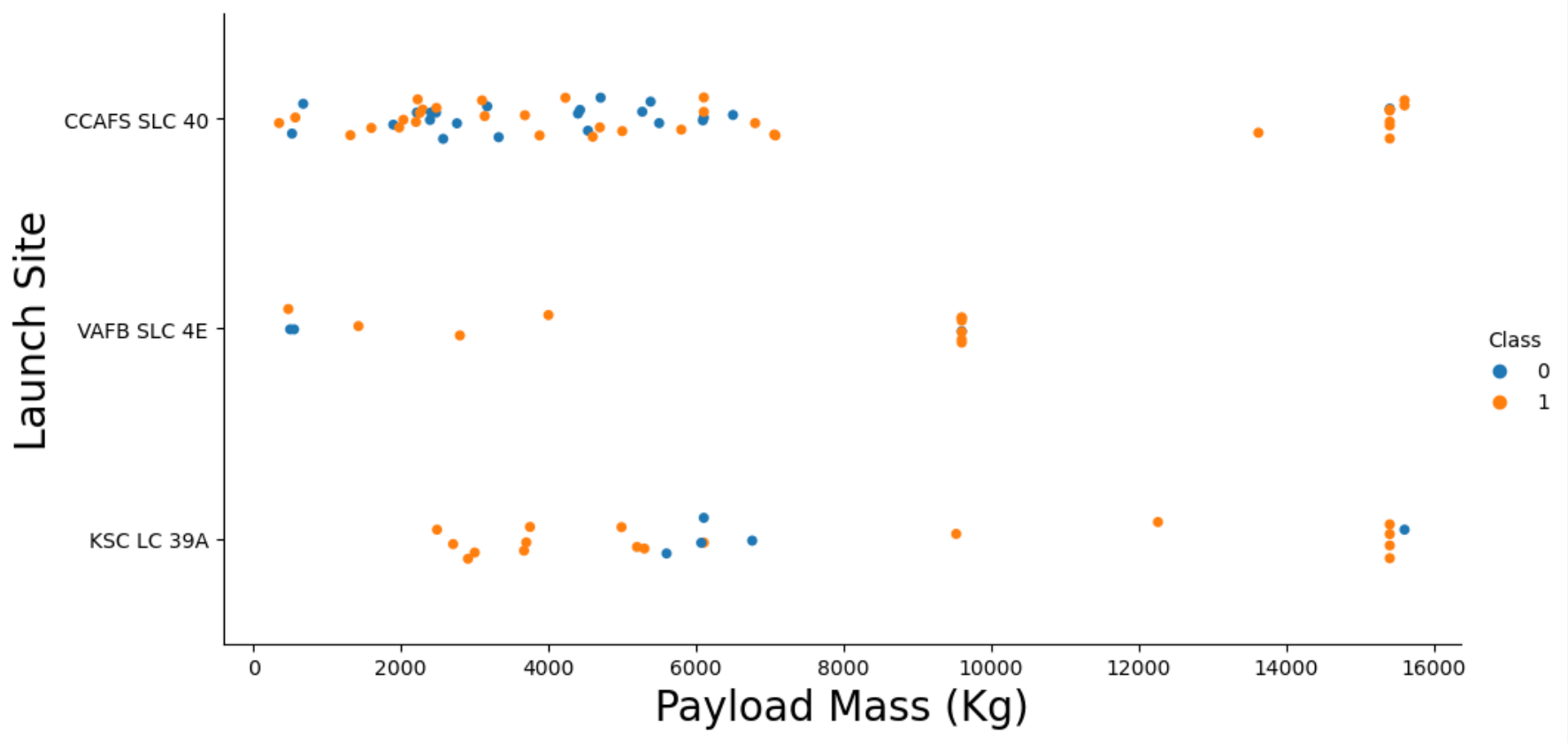


Flight Number vs. Launch Site

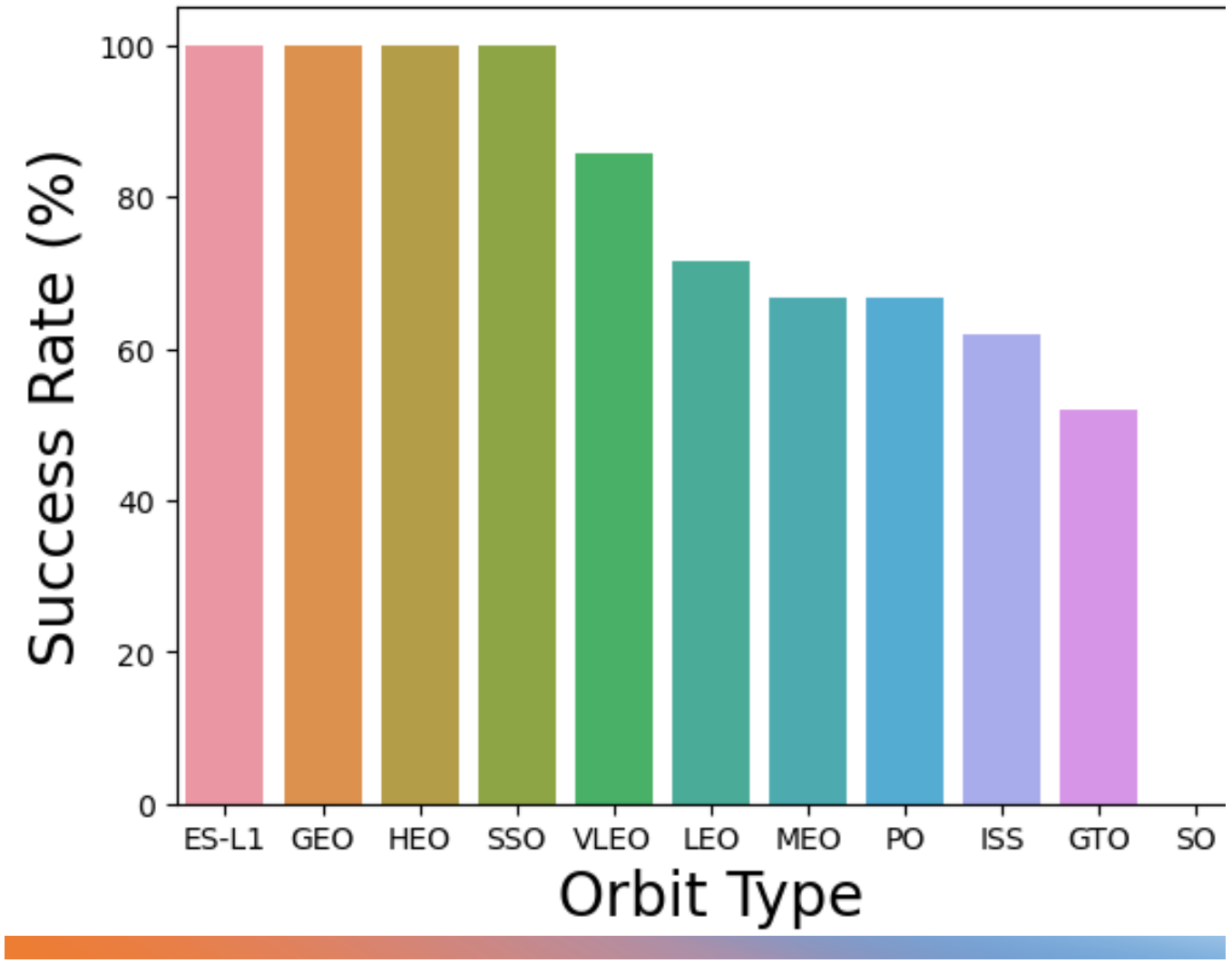


- *Site CCAFS SLC 40 has the lowest success rate.*
- *Success Rate for the launch sites does not seem to be correlated with flight number*

Payload vs. Launch Site



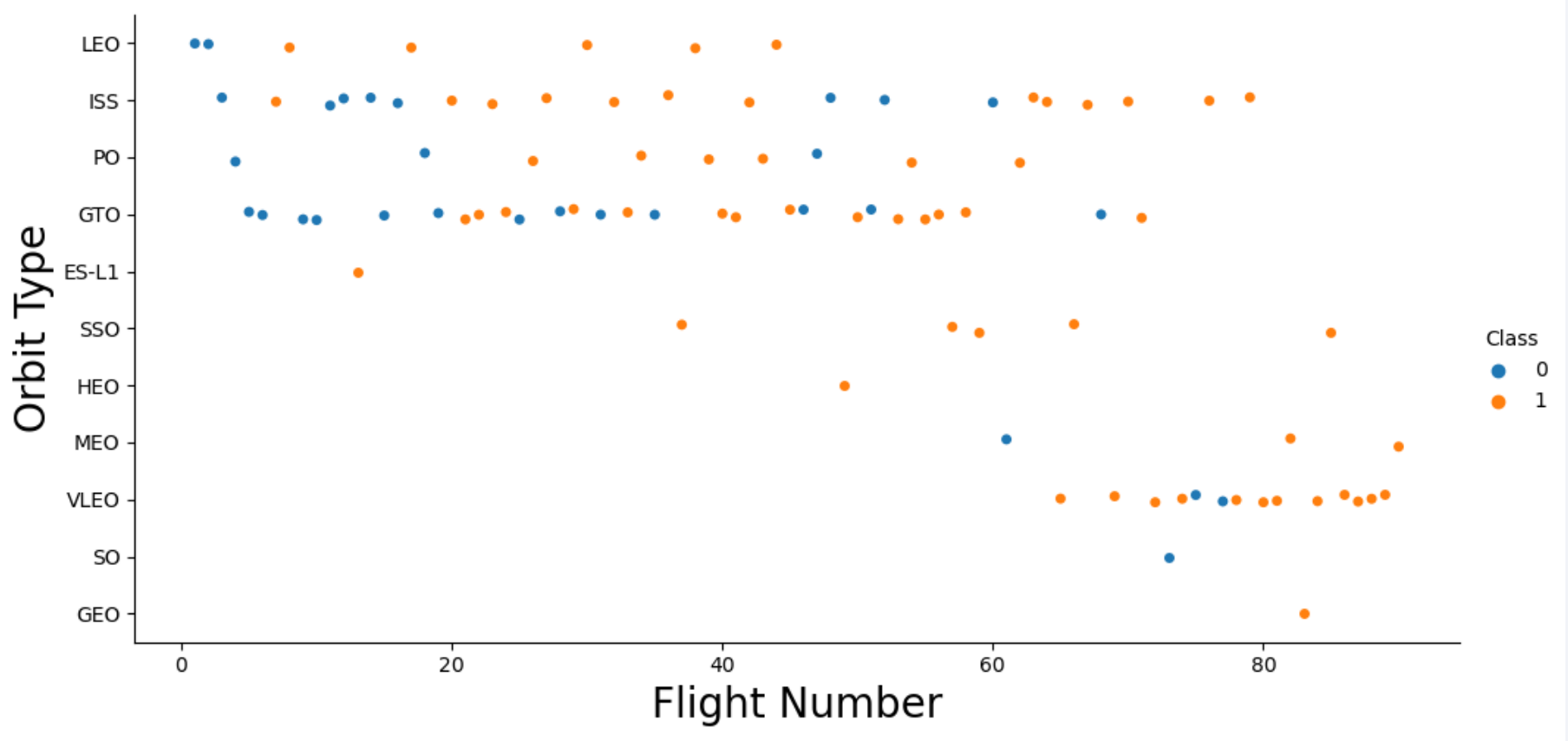
- *Site VAFB SLC 4E has no payload value over 10,000 kgs.*
- *Successful landing outcome for all launch sites seems to improve with higher payloads.*



Success Rate vs. Orbit Type

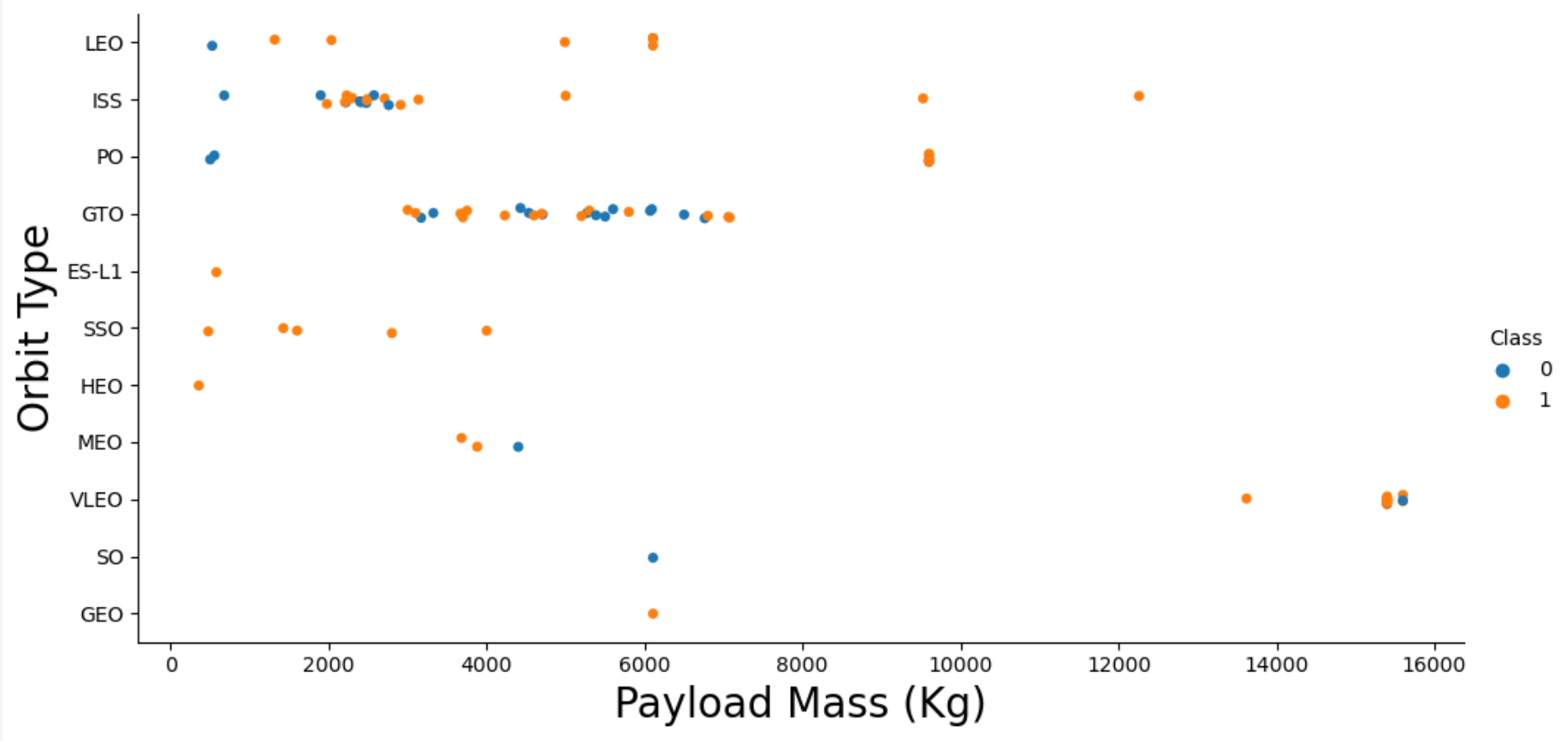
- *Orbits ES-L1, GEO, HEO and SSO have highest success rates.*
- *GTO has the lowest success rate less than 60%.*

Flight Number vs. Orbit Type



- *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.*

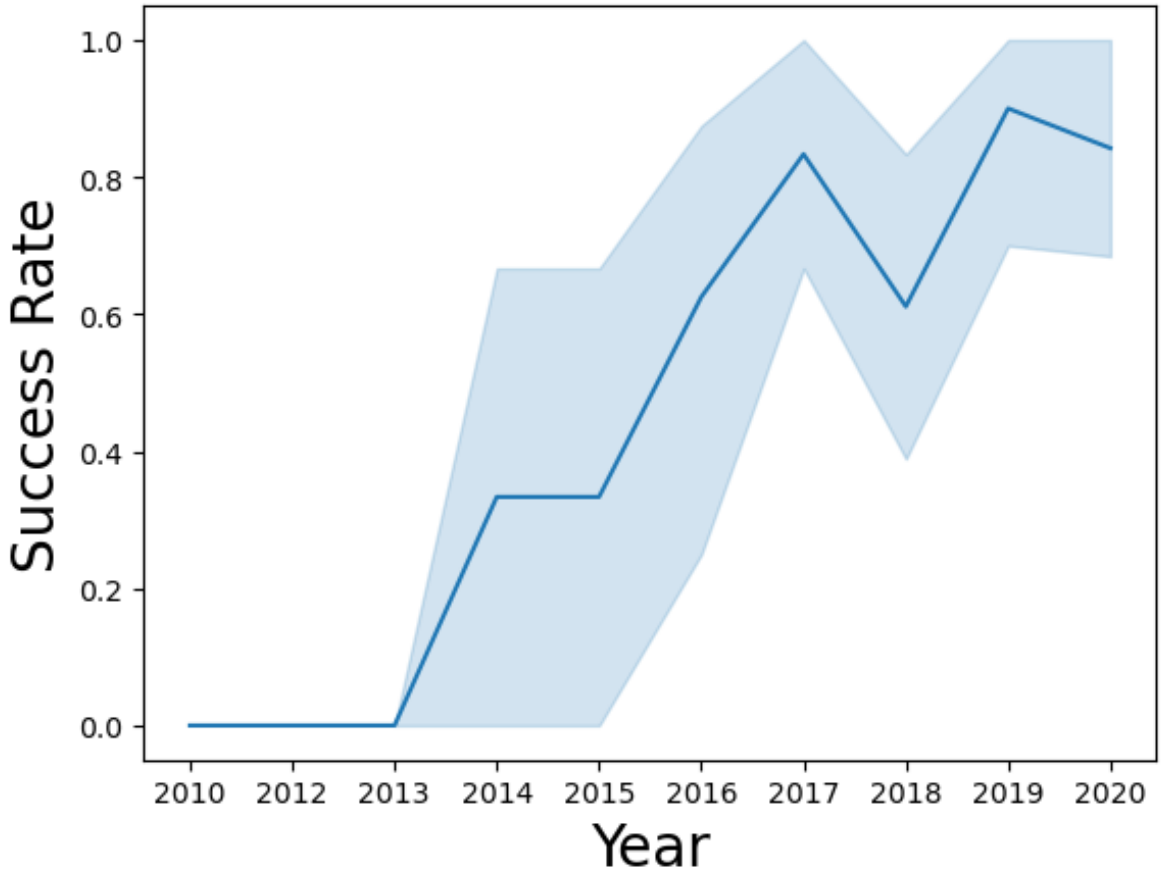
Payload vs. Orbit Type



- *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 kept increasing from 2013 to 2017.*



All Launch Site Names

```
%sql SELECT DISTINCT Launch_Site AS "Unique_Launch_Sites" FROM SPACEXTBL
```

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

- A total of 4 launch sites were found.

Launch Site Names Begin with 'CCA'

%sql SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%' LIMIT 5

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

- Using the wildcard operator with 'LIKE', we get the records where launch site starts with "CCA"

Total Payload Mass

```
%sql SELECT SUM(PAYLOAD_MASS__KG_)  AS "total_payload_mass",
  Customer FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
```

total_payload_mass	Customer
45596	NASA (CRS)

- Total payload mass used by NASA (CRS) is 45,596 kgs.

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_)  AS "average_payload_mass", Booster_Version
   FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.1%';
```

average_payload_mass	Booster_Version
2534.6666666666665	F9 v1.1 B1003

- Average payload mass carried by F9 v1.1 is 2,534.7 kgs.

First Successful Ground Landing Date

```
%sql SELECT MIN(Date)  AS "First_successful_ground_pad_landing_date"
  FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)';
```

First_successful_ground_pad_landing_date
2015-12-22

- The first successful ground pad landing was in December 2015.

Successful Drone Ship Landing with Payload between 4000 and 6000

%sql SELECT DISTINCT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;

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

- There are 4 booster versions with successful drone ship landing carrying payload between 4000 to 6000 kgs.

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) AS "Number_of_Missions"
FROM SPACEXTBL GROUP BY Mission_Outcome;
```

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

- There is only 1 failure in the mission outcomes.

Boosters Carried Maximum Payload

```
%sql SELECT Booster_Version AS
"Boosters_with_max_payload",
PAYLOAD_MASS__KG_ FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT
MAX(PAYLOAD_MASS__KG_) FROM
SPACEXTBL);
```

- The maximum payload is at 15,600 kgs.

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

2015 Launch Records

%sql SELECT substr(Date,4,1) AS "MONTH", substr(Date,9,2), Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL WHERE Landing_Outcome = 'Failure (drone ship)' AND substr(Date,9,2)='15';

MONTH	substr(Date,9,2)	Landing_Outcome	Booster_Version	Launch_Site
6	15	Failure (drone ship)	F9 FT B1024	CCAFS LC-40

- June is the month in 2015 when there was a drone ship failed landing.

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

```
%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) FROM SPACEXTBL WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY COUNT(Landing_Outcome) DESC;
```

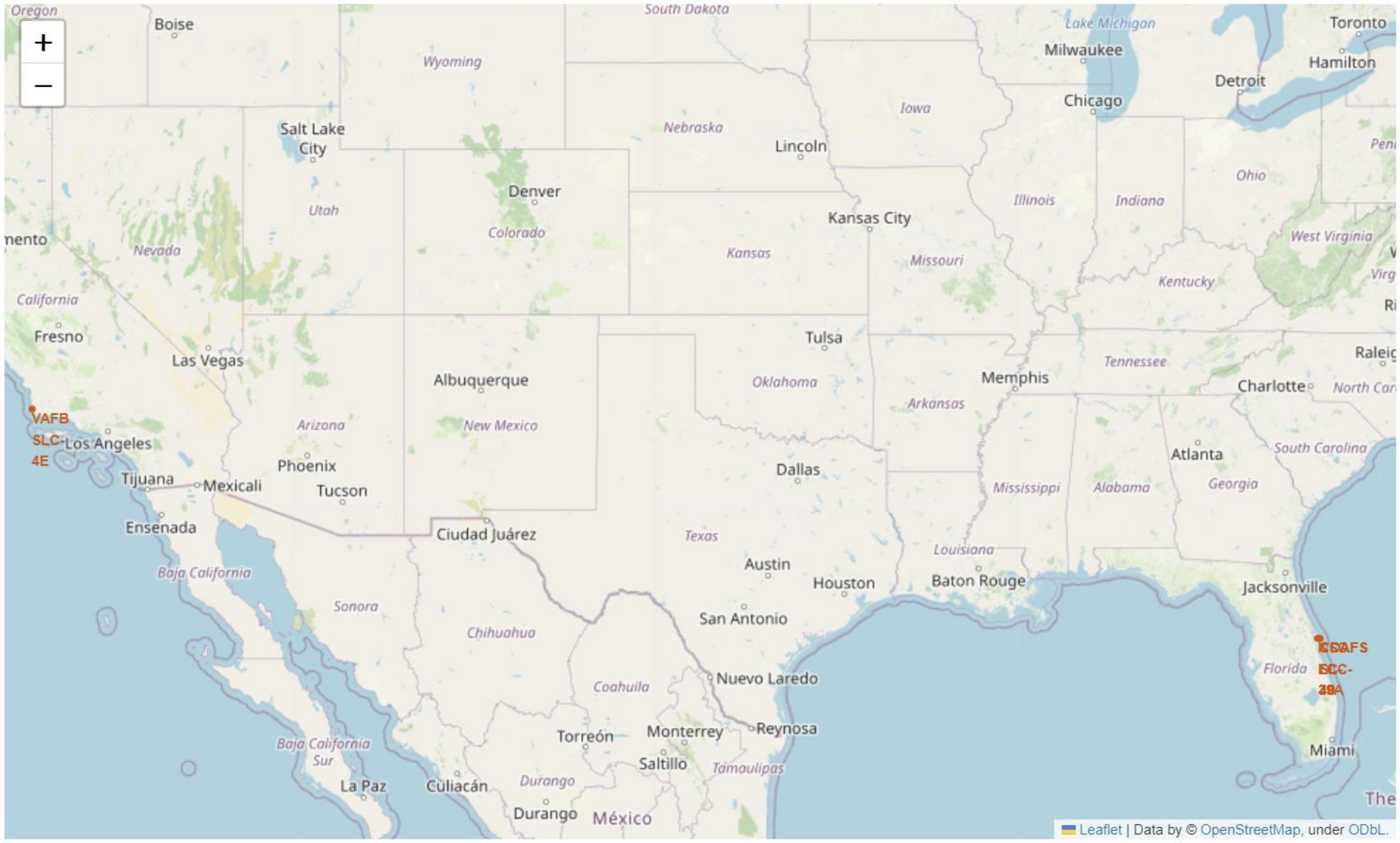
- Landing outcomes during the given date range is ranked from highest to lowest.

Landing_Outcome	COUNT(Landing_Outcome)
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

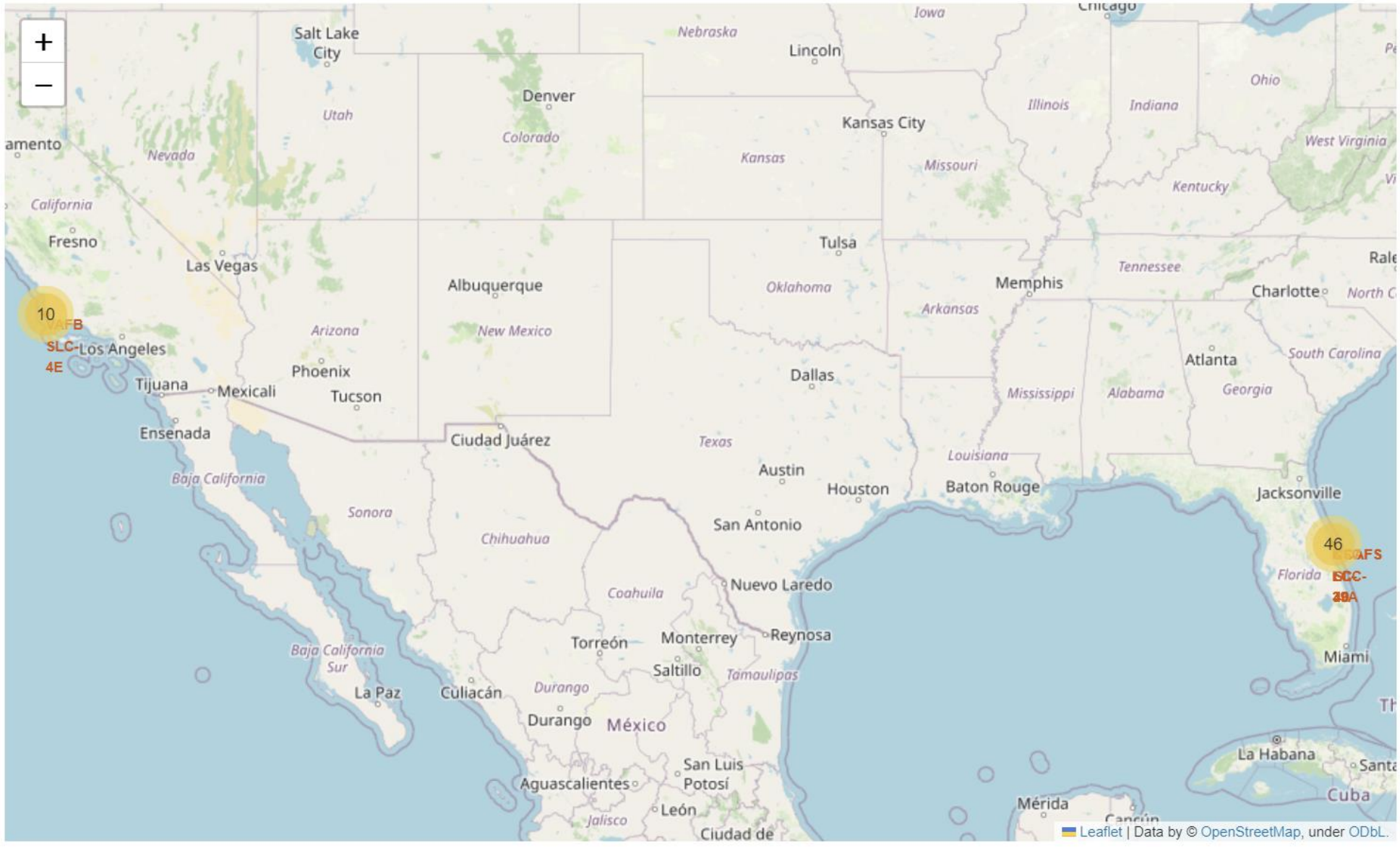

Section 3
Launch Sites
Proximities Analysis

SpaceX launch sites

- Created a folium Map object, with an initial center location to be NASA Johnson Space Center at Houston, Texas.
- Added a circle and marker for each launch site in the data.



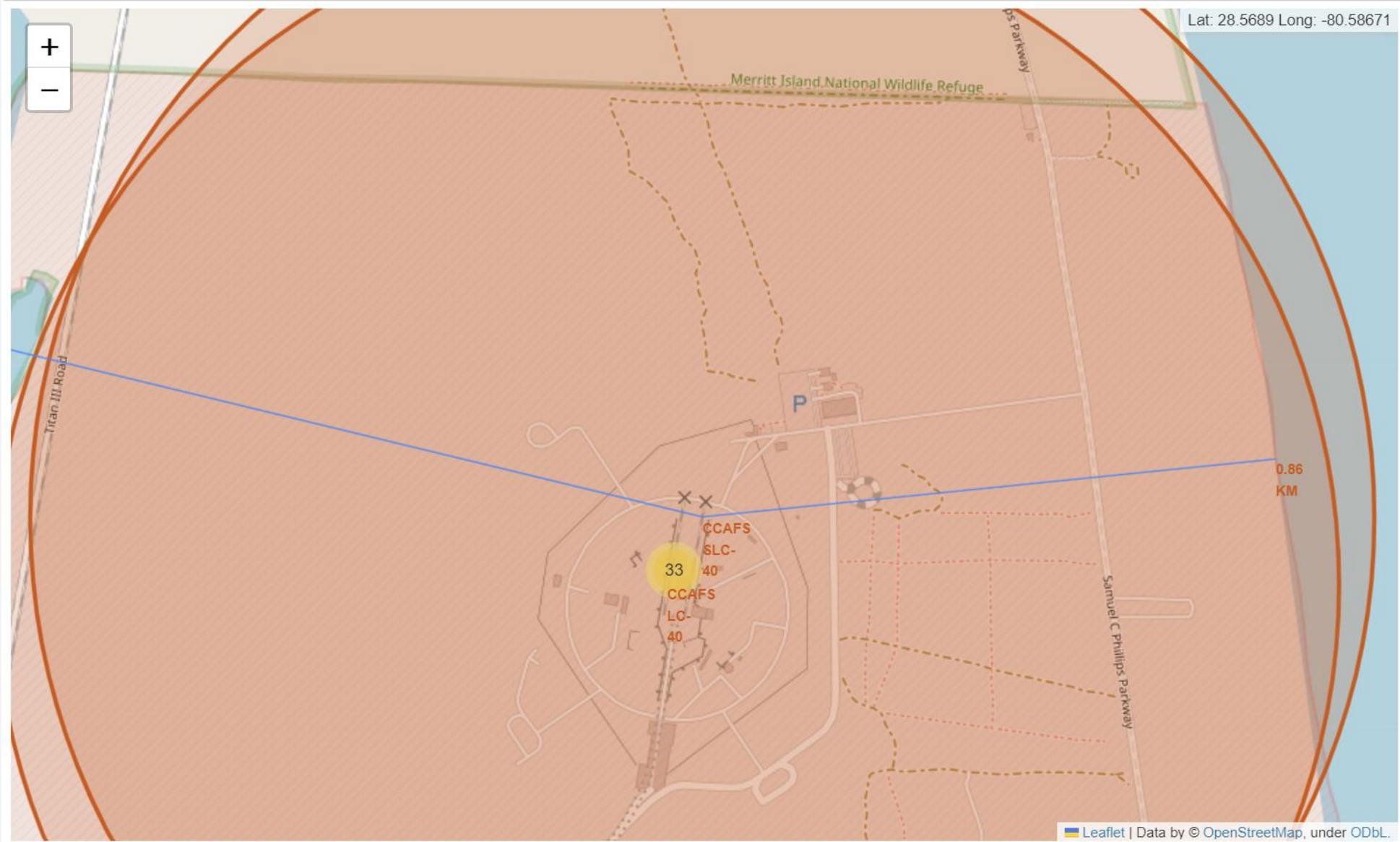
Success/failed launches for each site

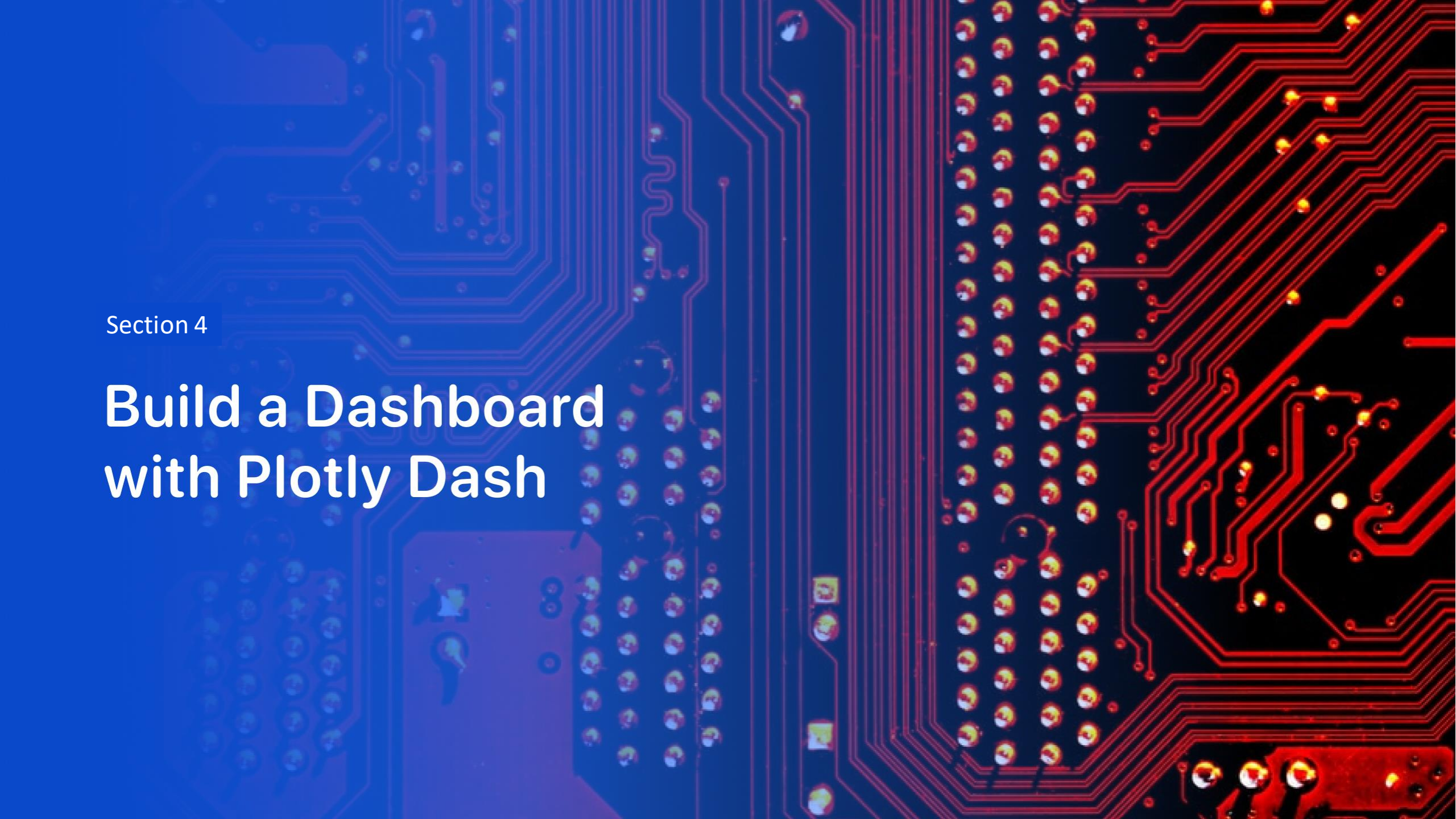


- Created color coded markers for all launch records. Green for success, red for not.
- Since many launch records will have the exact same coordinate, marker clusters simplify the map.

Distances between a launch site to its proximities

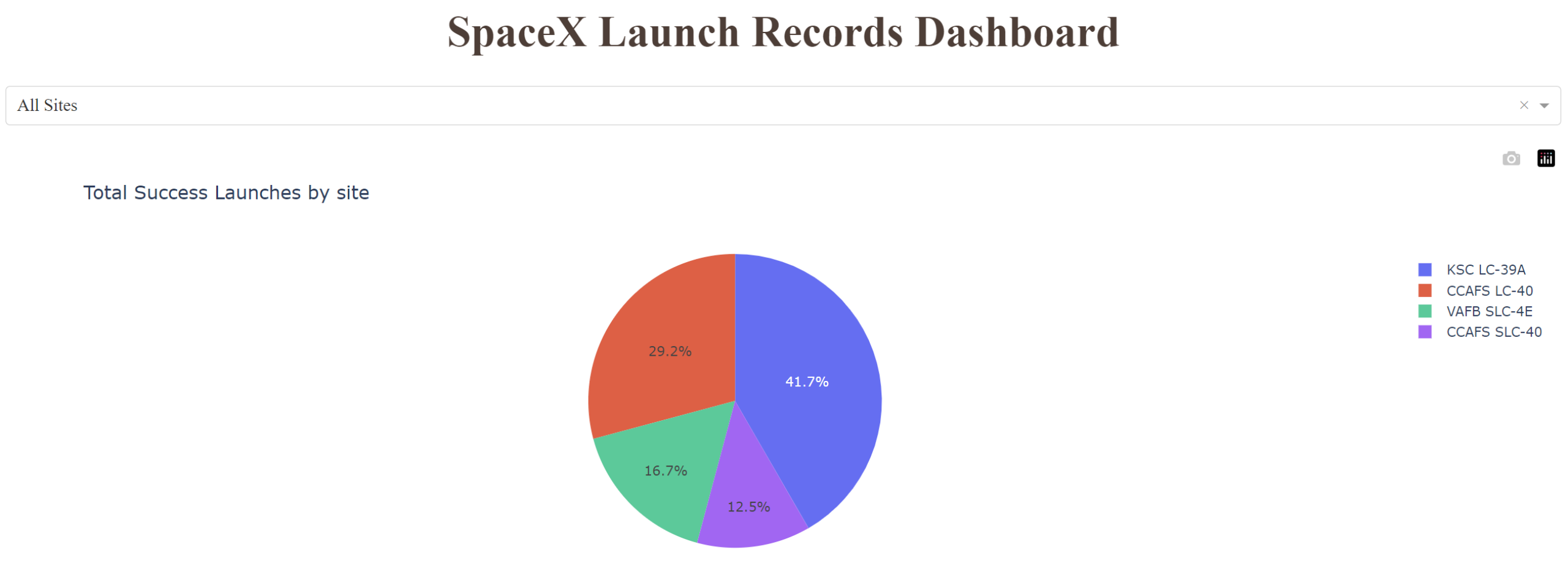
- MousePosition gets coordinate for as the user's mouse moves over the map.
- Add a marker to show the distance between the nearest coastline point and the launch site.
- Add similar markers to other places of interest close to the launch site.




Section 4
Build a Dashboard with Plotly Dash

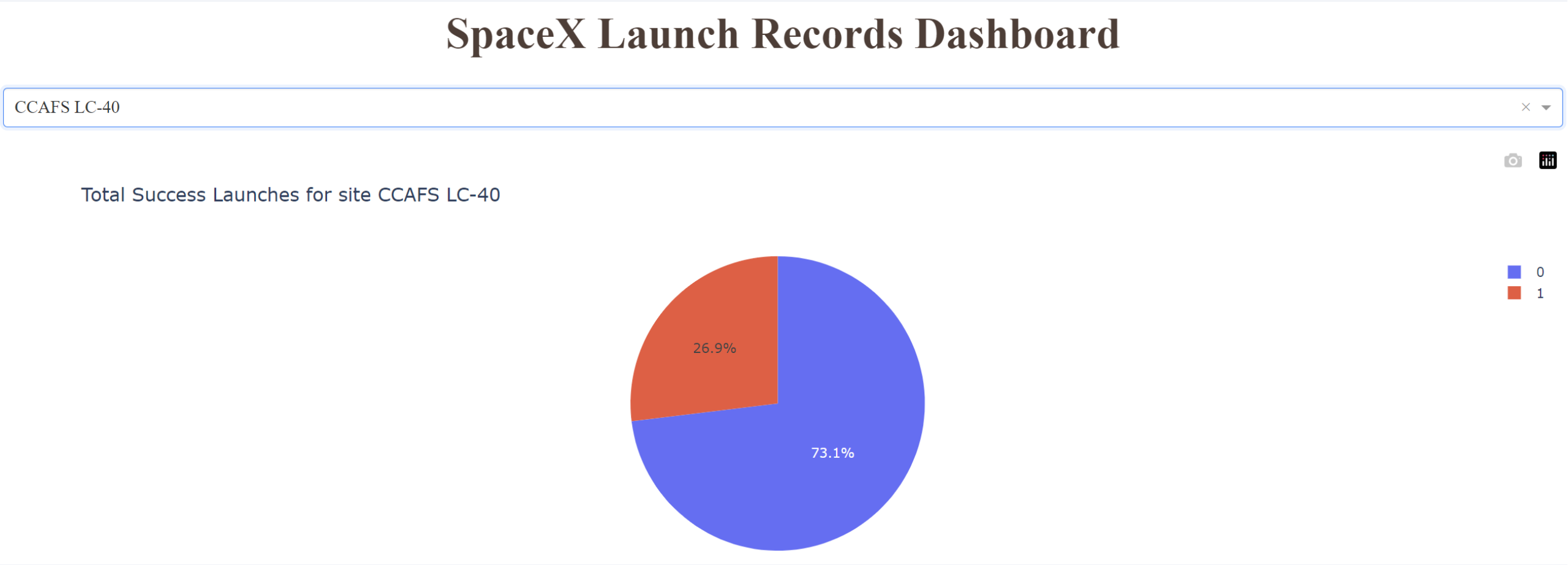
Launch successes for all sites

- Following piechart shows launch success count for all sites.
- KSC LC-39A has the most number of successful launches.



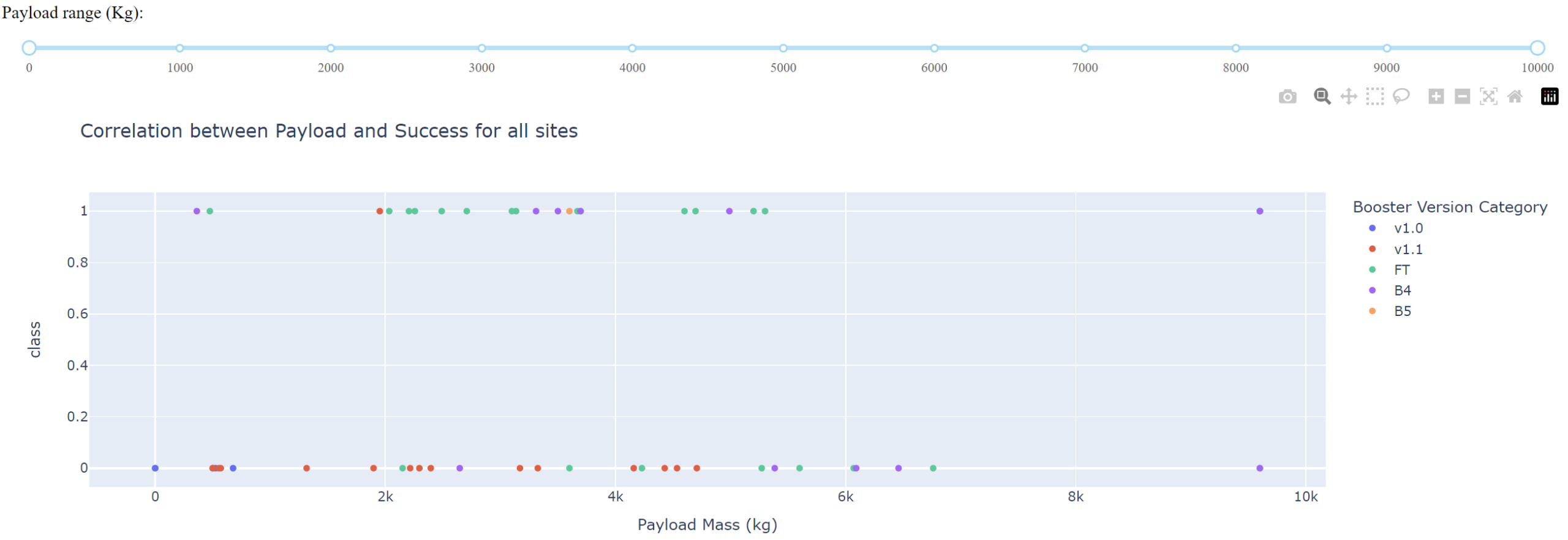
Launch site with highest launch success ratio

- The pie chart is filtered using the dropdown to show the success ratio for one site.



Payload vs. Launch Outcome for all sites

- The following scatter plot shows relationship between payload and launch outcome for all sites, with an interactive payload range slider.

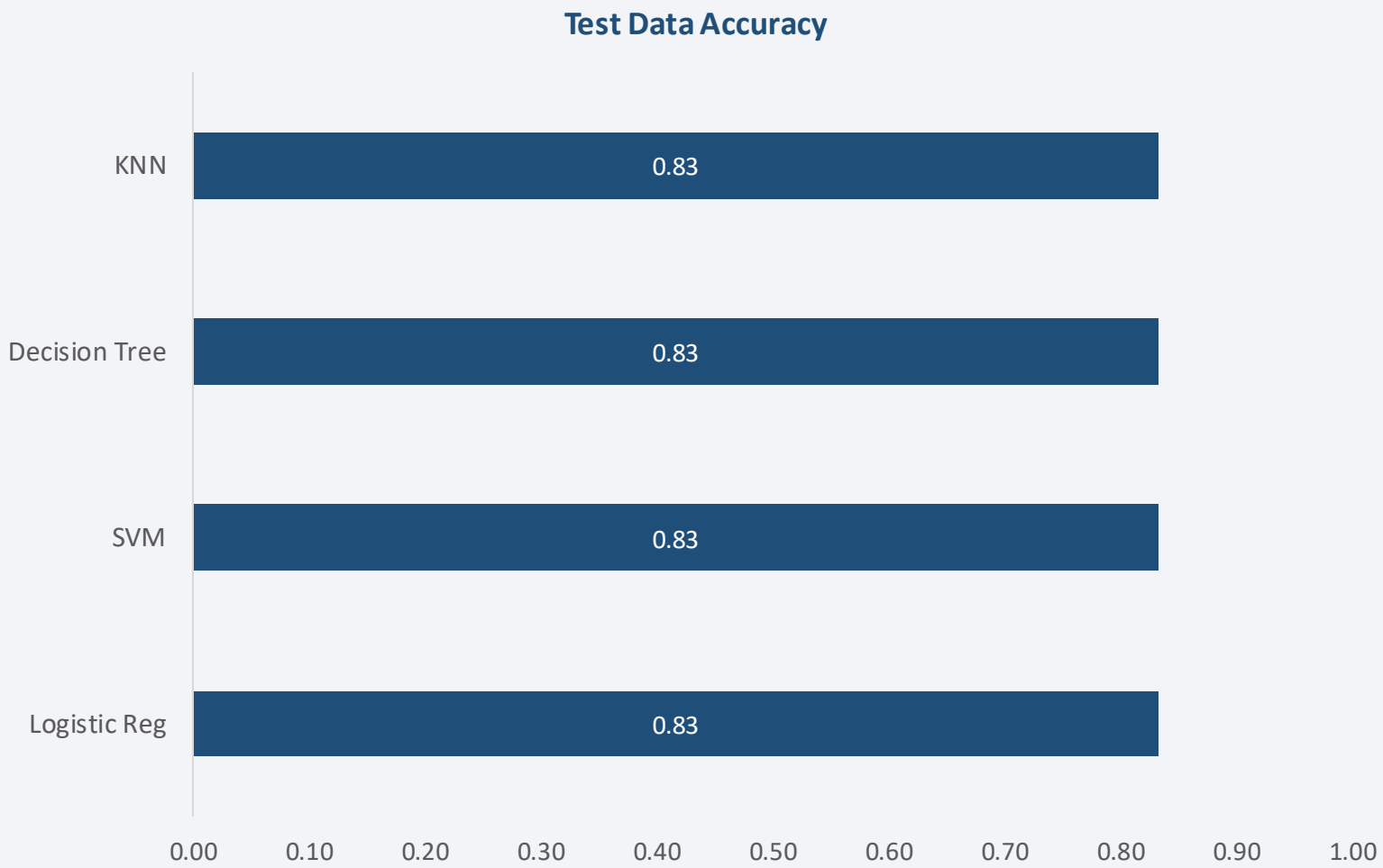


Section 5

Predictive Analysis (Classification)

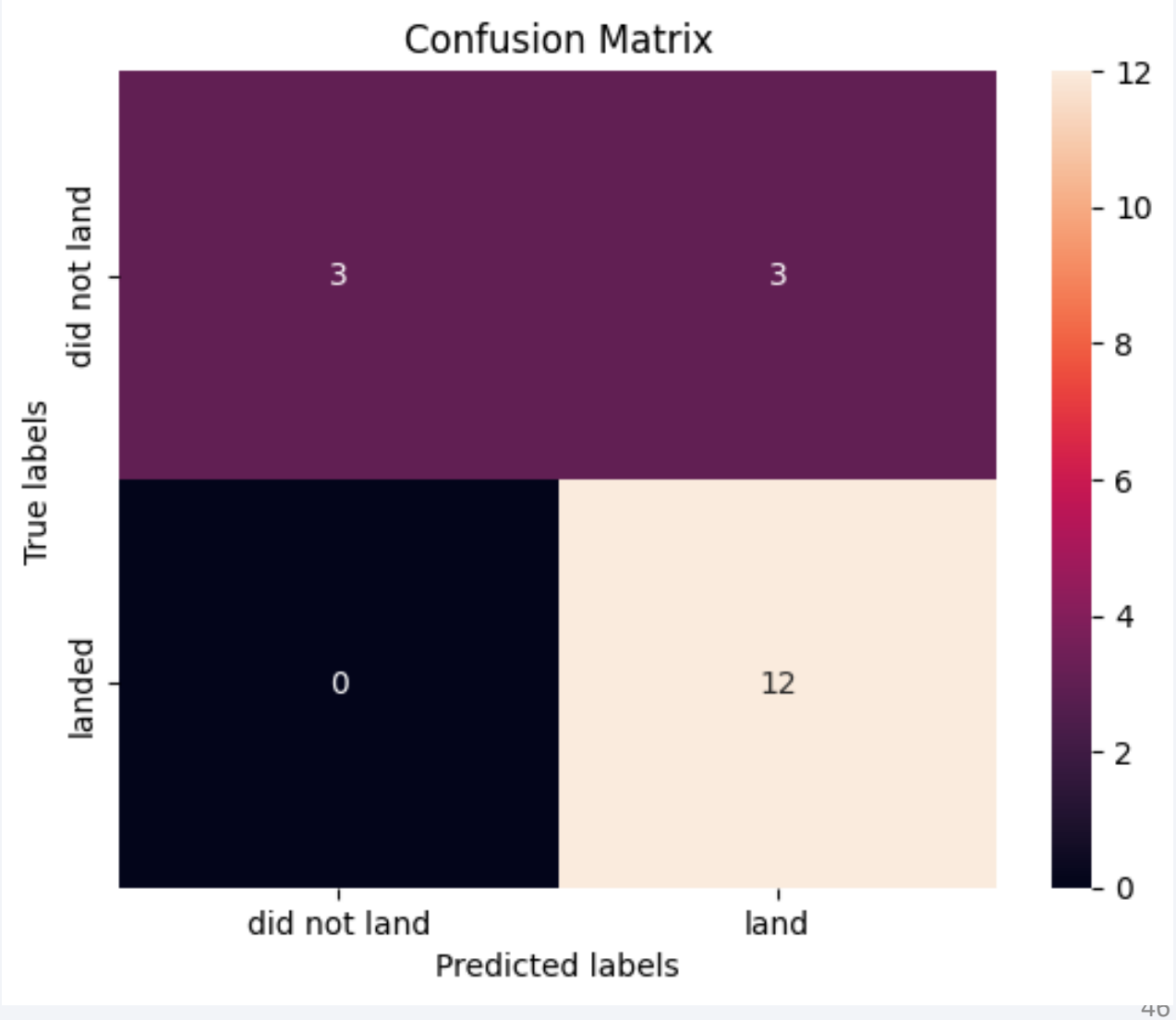
Classification Accuracy

- As seen in this chart, the test accuracies are the same at 0.83 for all models. Hence they perform equally well.



Confusion Matrix

- The confusion matrix is as shown here:
- All the successful landings were predicted correctly, while only half of the unsuccessful landings were given the right label.



Conclusions

- Different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%
- Orbits ES-L1, GEO, HEO and SSO have higher success rate.
- With heavy payloads the successful landing rate is more for Polar, LEO and ISS.
- Success rate since 2013 kept increasing till 2020.
- Logistic regression, Support Vector Machine, Decision Tree and k Nearest Neighbors algorithms were used to build classification models and all the models have the same accuracy after tuning.

Appendix

- Project Files Repository: https://github.com/Diyav/SpaceX_Falcon9_Launches

Thank you!