IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Anish Desai
22nd May, 2023

Outline

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

Executive Summary

In this Data Science Capstone Project, I have tried to predict if SpaceX's Falcon9 first stage will land successfully. To accomplish this, I have adopted the following methodologies:

- Summary of methodologies
  - ✓ ***Collection of data*** using SpaceX REST API and Web Scraping
  - ✓ ***Wrangle data*** for appropriate pre-processing
  - ✓ ***Exploratory Data Analysis*** using SQL and Data Visualization
  - ✓ ***Interactive Visual Analytics*** Dashboard using Folium and Plotly
  - ✓ ***Predictive Analysis*** by building ML models to predict landing outcomes

- Summary of all results
  - ✓ Exploratory Data Analysis results – ***Specific orbits*** such as GEO, HEO and SSO have ***100% success rate***.
  - ✓ Interactive Visual Analytics results – Most or nearly all the ***launch sites*** are near to equator and coast, far from populated areas.
  - ✓ Predictive Analysis results – All ***classification models*** have nearly 83.33% accuracy score, while ***Decision Tree model*** fared better when tested on entire dataset.

Introduction

- Project background and context

In this capstone, **I have to predict if the Falcon 9 first stage will land successfully**. 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 we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.**

- Problems you want to find answers
  - What factors largely contribute to the success of first-stage landing?
  - What is the trends in the first-stage landing success over the years?
  - What are the specifications for a successful landing?
  - Which classification models can accurately predict first-stage landing success in terms of binary classification?

Section 1

Methodology

Methodology

- Data collection methodology:

  Request Data from SpaceX API, decode and filter response and convert to Dataframe using .json before exporting it to .csv file.

  Request Falcon9 launch data from Wikipedia, create BeautifulSoup object from HTML response, create Dataframe and export data to .csv file.

- Perform data wrangling

  Filter the data, Handle missing value and apply One-hot encoding.

- 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 classification models to accurately predict first-stage landing outcomes.

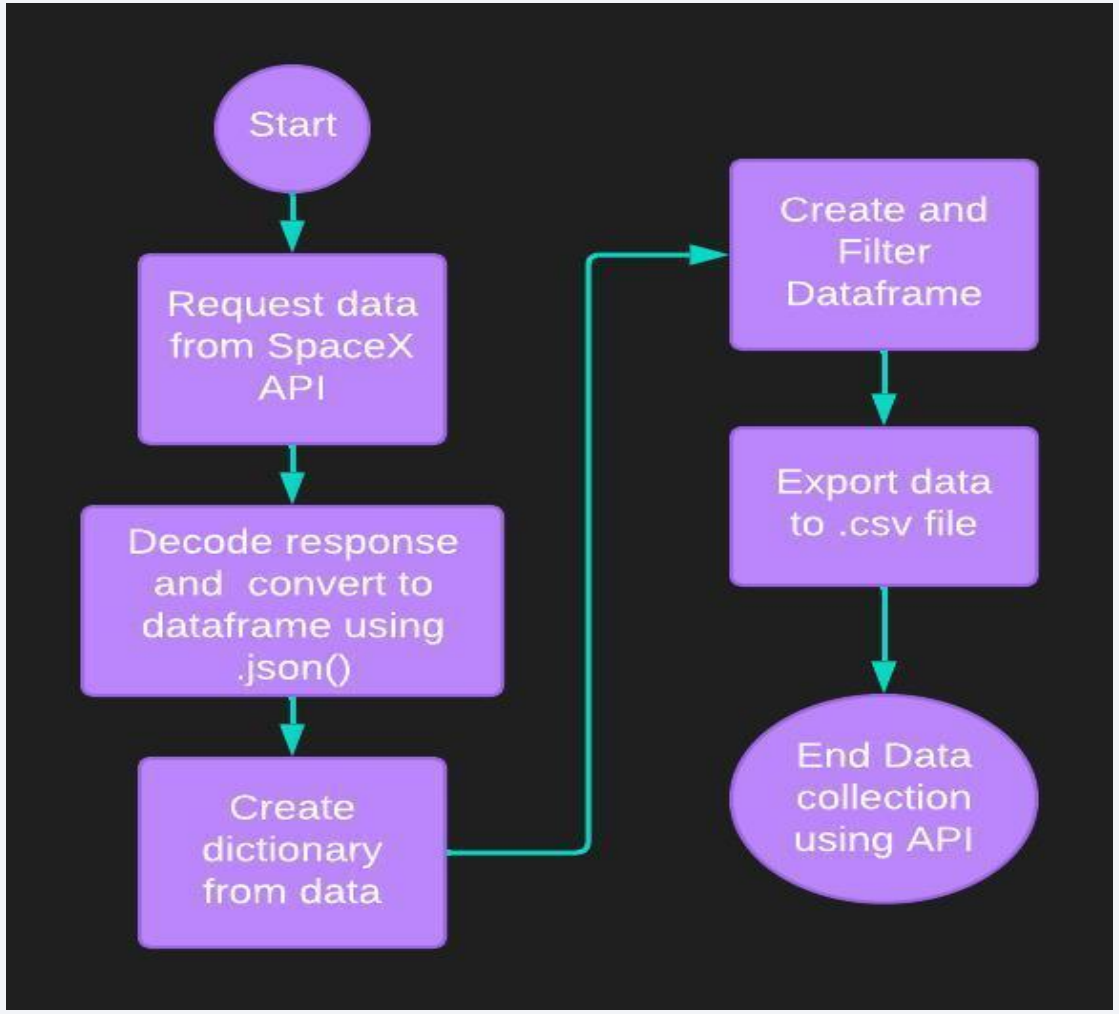
Data Collection

The data collection process involved gathering information from two different sources: the SpaceX REST API and the Wikipedia entry for SpaceX Falcon9 Launch data. I utilized API requests to collect specific data from the SpaceX REST API, while I performed web scraping to extract data from a table within SpaceX's Wikipedia page. It was needed to utilize both of these data collection methods in order to obtain comprehensive information about the launches for a more detailed analysis.

The data columns obtained from the SpaceX REST API include: FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude.

On the other hand, the data columns obtained through Wikipedia web scraping include: Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time.

Data Collection – SpaceX API



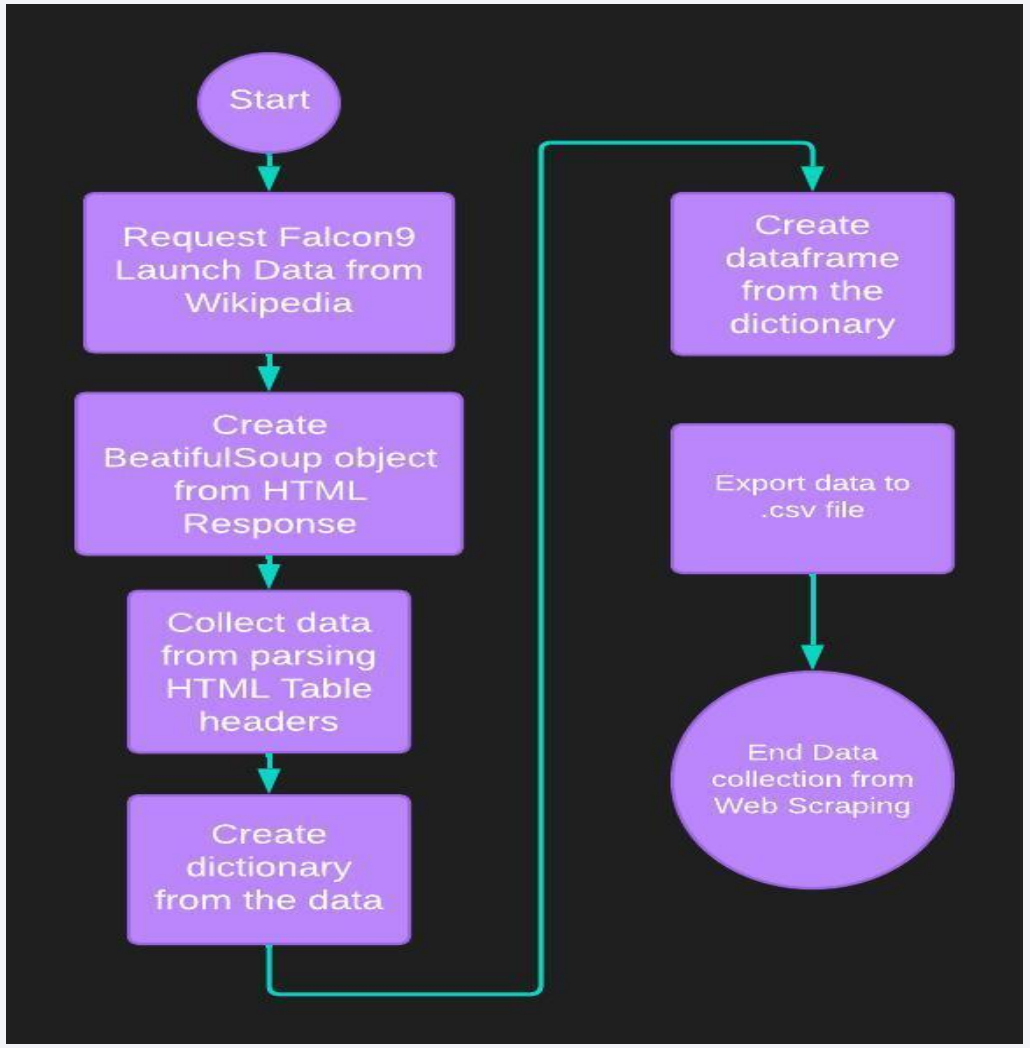
Github URL for the SpaceX API Data collection Methodology:

https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook1%20-%20jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

Github URL for the SpaceX API Data collection Methodology:

https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook2%20-%20jupyter-labs-spacex-data-collection-webscraping.ipynb



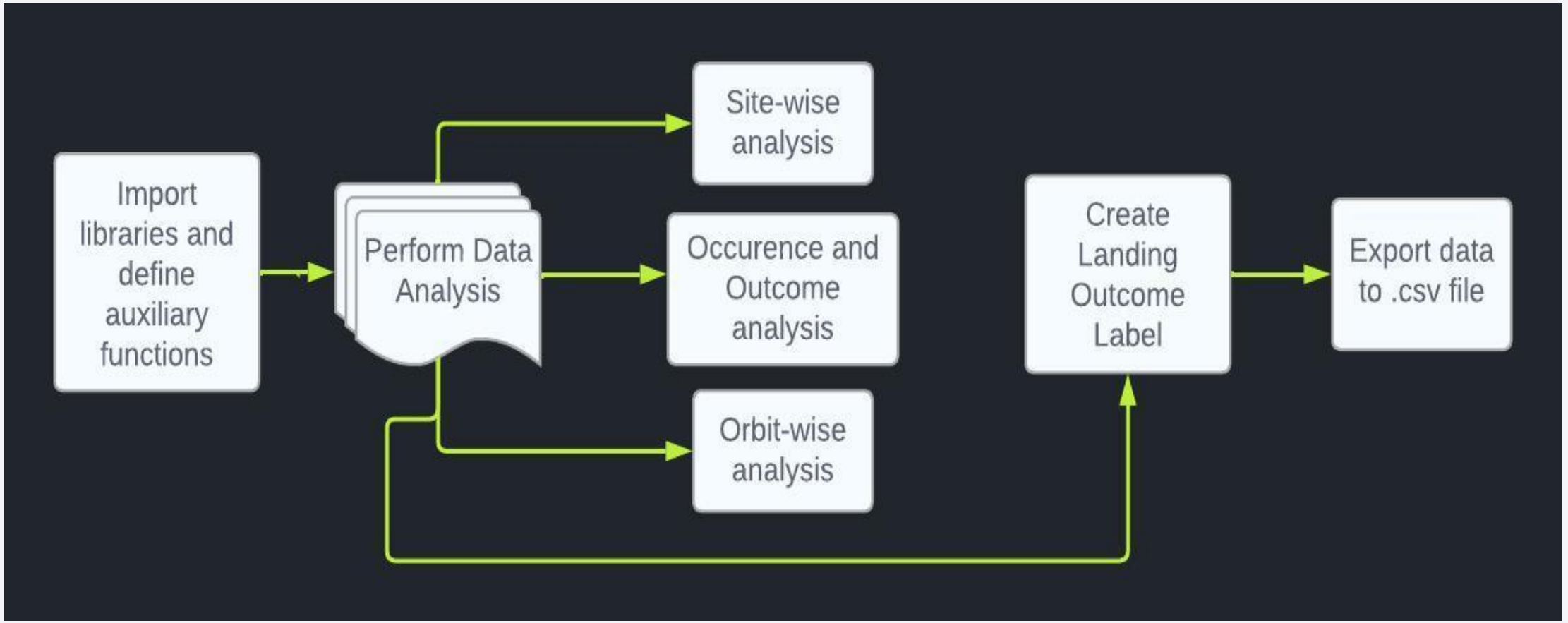
Data Wrangling

Data wrangling involved:

- Performing EDA and determine the training labels
- Calculate the number of launches on each site
- Calculate the number and occurrence of each orbit
- Calculate the number and occurrence of mission outcome per orbit type
- Create a landing outcome label from Outcome column

Github URL for Data Wrangling Implementation:
https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook3%20-%20labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

Data Wrangling



EDA with Data Visualization

Charts plotted include:

- Flight Number vs PayLoad Mass (Kg) Scatterplot
- Flight Number vs Launch Site Scatterplot
- PayLoad Mass (Kg) vs Launch Site Scatterplot
- Orbit Type vs Success Rate Barplot
- Flight Number vs Orbit Scatterplot
- PayLoad Mass (Kg) vs Orbit Scatterplot
- Success Rate Yearly Trends Lineplot

Scatterplots are helpful in visualizing relationships that exists between the features and offer an easy way for analysis using correlation.

Barplots are useful for analysis because they visually display categorical data and their corresponding values, allowing for easy comparison and identification of patterns or trends. They provide a concise summary of the data distribution and facilitate quick insights into the relative magnitudes or frequencies of different categories.

Lineplots help identify trends, seasonal variations, or cyclical patterns in the data, making them valuable for time-dependent analysis and identifying correlations or changes in variables over time.

Github URL for Data Visualization Implementation: https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook5%20-%20jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

The SQL query analysis performed include:

- Displaying names of unique launch sites
- Displaying launch sites starting with CCA
- Display 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 successful landing outcome in ground pad was achieved
- 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
- 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 URL for the SQL Query Analysis Implementation: https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook4%20-%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Using Folium, I have marked **all the launch sites on the map**. This includes creating a blue circle at NASA Johnson Space Center's coordinate with a popup label showing its name. For each launch site, I **added a Circle object** based on its coordinate (Lat, Long) values in addition to launch site name. This **helps in visualizing the launch sites' proximity** to the equator as well as the coasts.

Further, I have enhanced the map by adding the **launch outcomes for each site**, and see which sites have high success rates. For this, I added **colored markers of green and red** for successful and unsuccessful launch outcomes respectively at each launch site.

The **distances** between a launch site to its proximities such as nearest city, railways, highways and coats have also been calculated and visualized by drawing a colored line marker between them.

Github URL for Folium Implementation : https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook6%20-%20lab_jupyter_launch_site_location_folium.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

Plotly Dash was effectively used to build an interactive dashboard that essentially consists of a **dropdown list** to enable launch-site selection.

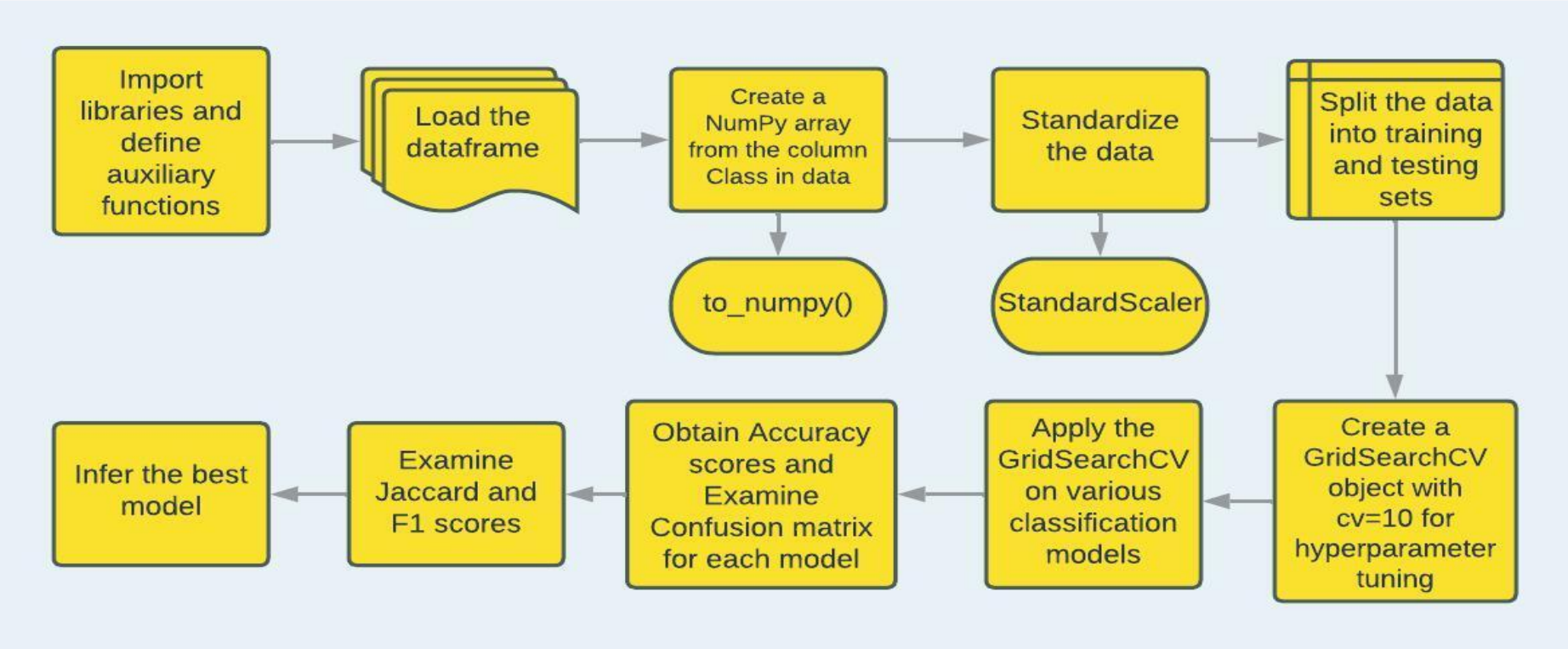
I have also added a **pie chart** to show the total successful launches count for all sites. If a specific launch site was selected, show the Success vs. Failed counts for the site.

Further, another important aspect effecting the landing outcome i.e., Pay Load Mass (Kg) has been visualized. I have added a **slider** to select payload range and subsequently a **scatter chart** to show the correlation between payload and launch success.

Github URL for Plotly Implementation : https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook7%20-%20spacex_dash_app.py

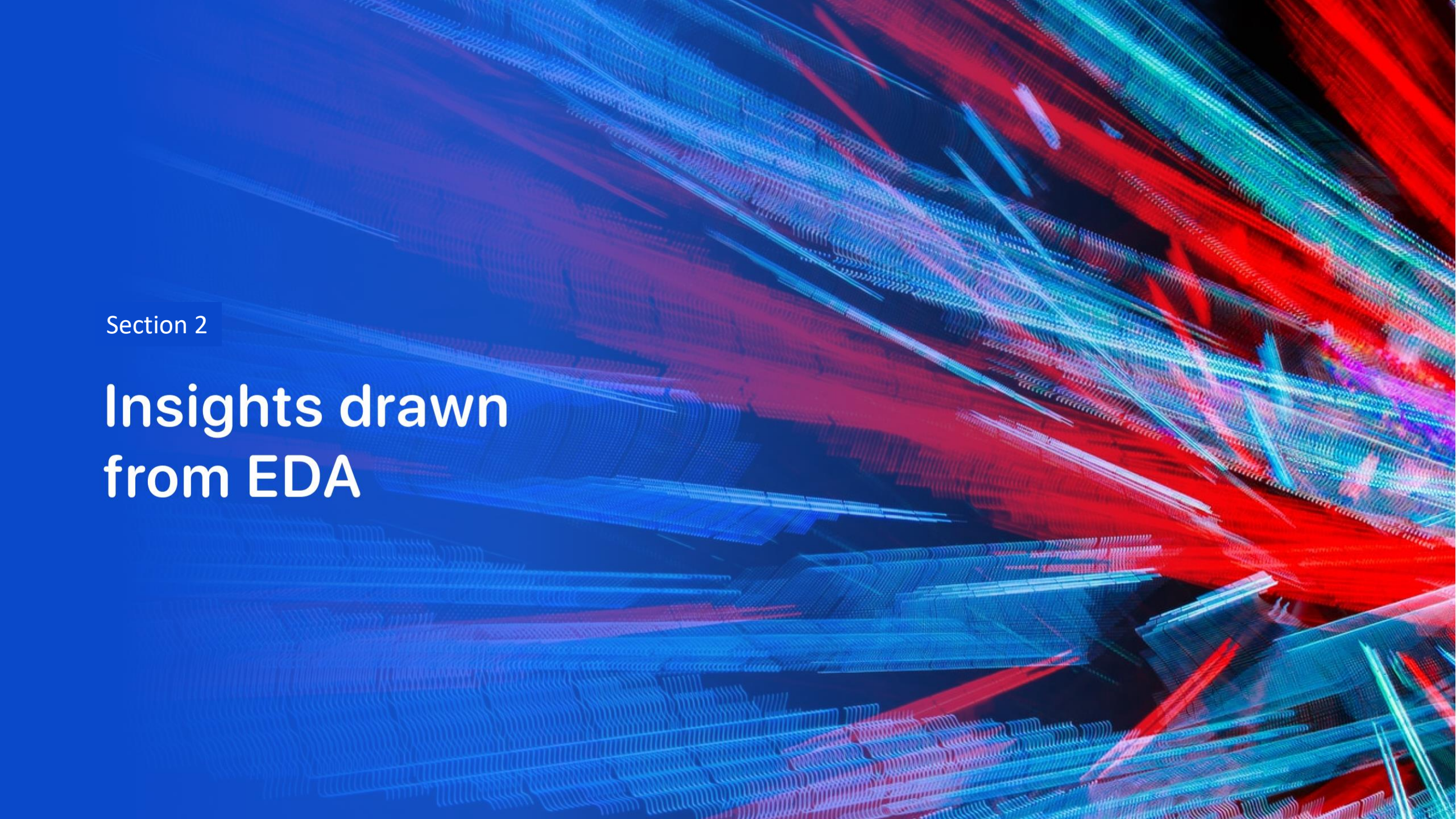
Predictive Analysis (Classification)

Github URL for Predictive Analysis Implementation : https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Notebook8%20-%20SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb



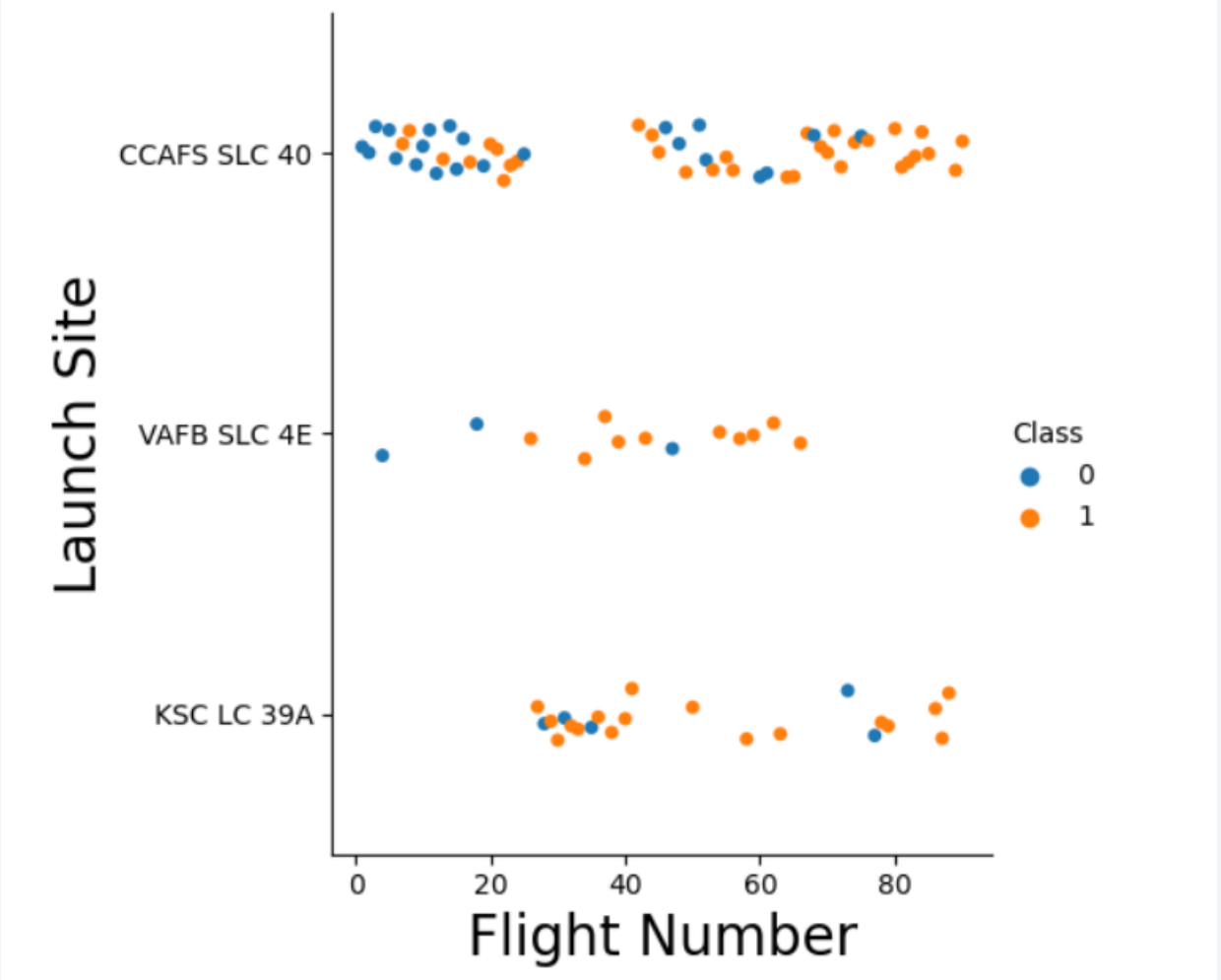
Results

- Exploratory data analysis results
  - ✓ Launch success has improved over time.
  - ✓ KSC LC-39A has the highest success rate among landing sites.
  - ✓ Orbits ES-L1, GEO, HEO and SSO have a 100% success rate.
- Interactive analytics results
  - ✓ Most launch sites are near the equator, and all are close to the coast.
  - ✓ Launch sites are far enough away from anything a failed launch can damage (city, highway, railway), while still close enough to bring people and material to support launch activities.
- Predictive analysis results
  - ✓ Decision Tree model is the best predictive model for the dataset.

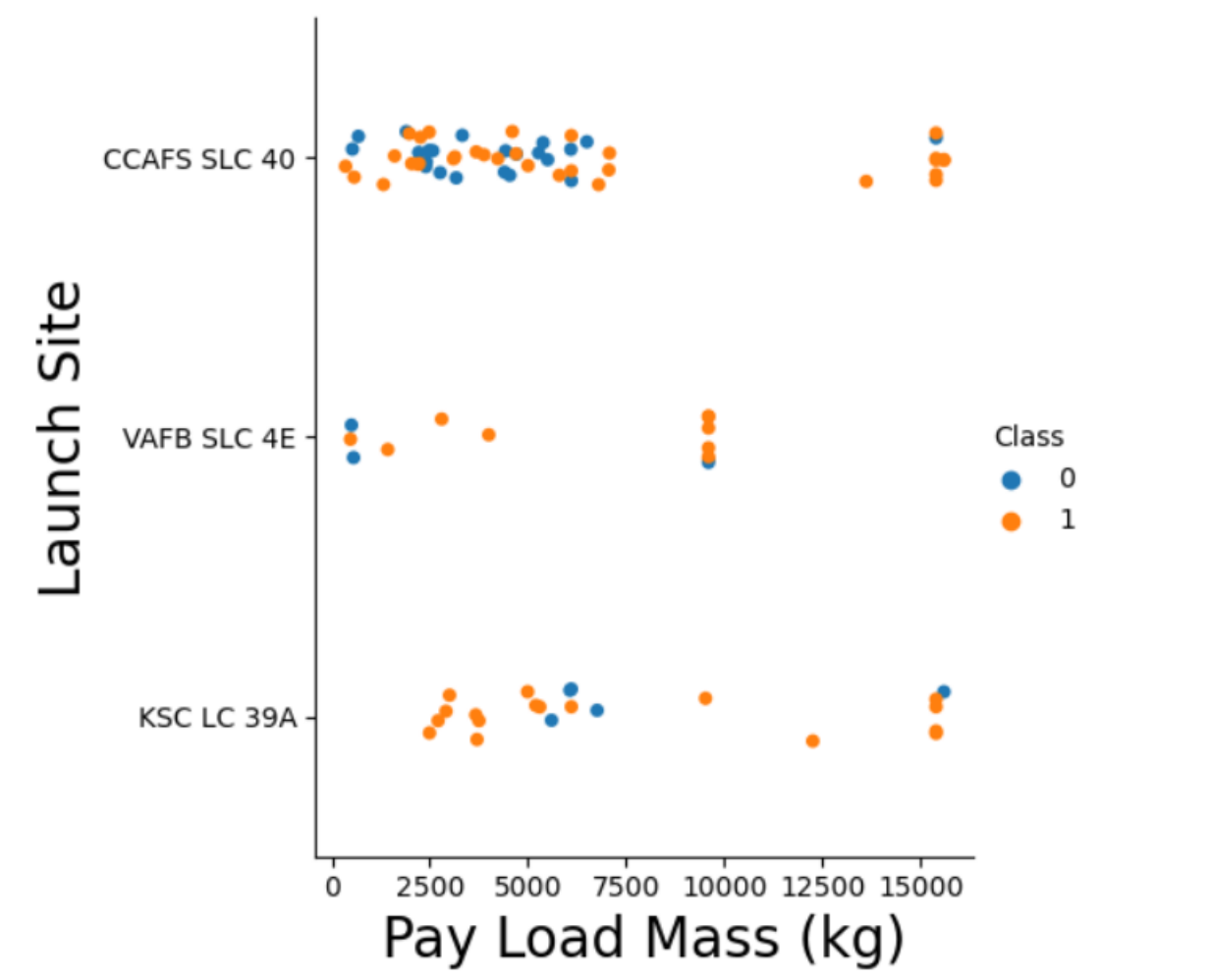

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

- As the flight number is increasing, the success rate (class) is also increasing.
- The launch site KSC LC 39A has higher success rate.



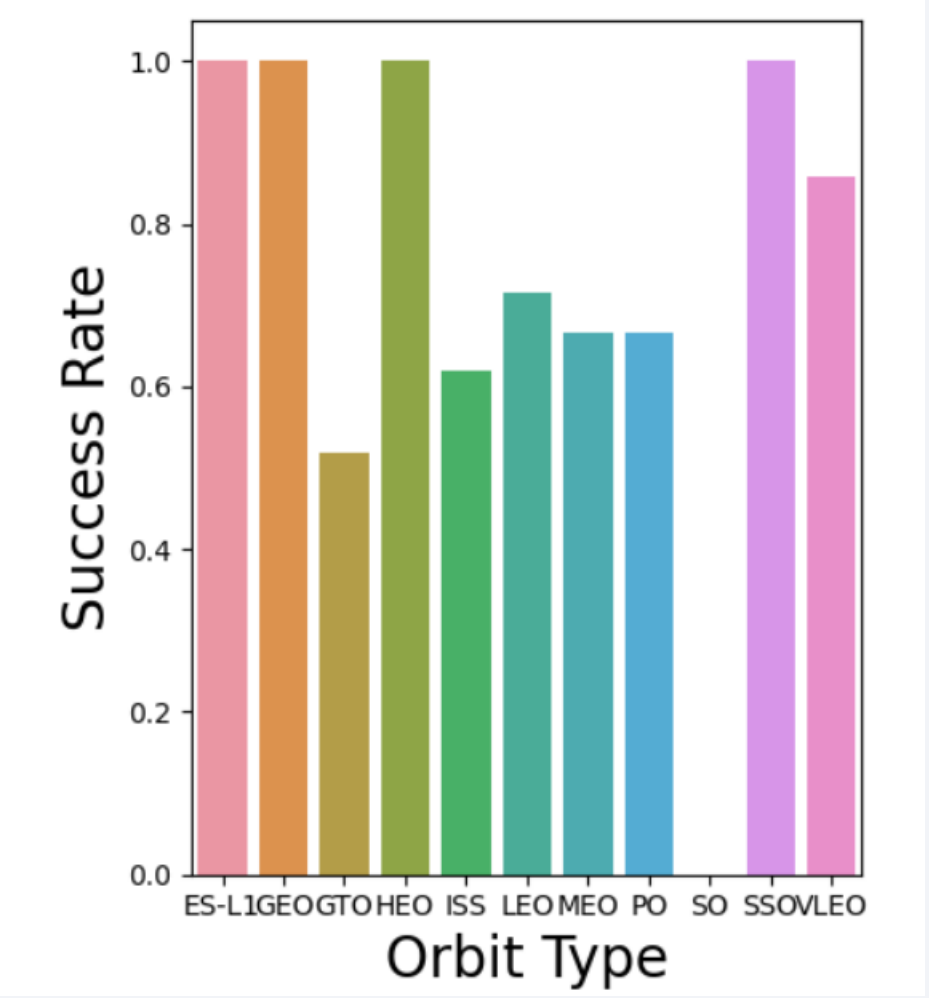
Payload vs. Launch Site



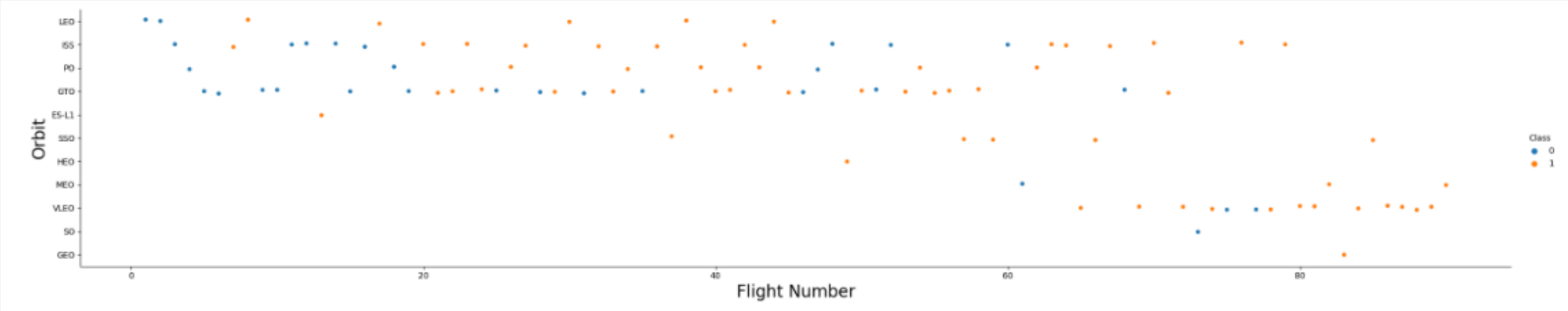
- For every launch site the higher the payload mass, the higher the success rate.
- The VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).
- KSC LC 39A has a 100% success rate for launches less than 5,500 kg.

Success Rate vs. Orbit Type

- The orbits ES-L1, GEO, HEO and SSO have nearly 1.0 (100%) success rate.
- The orbit SO has 0.0 (0%) success rate.

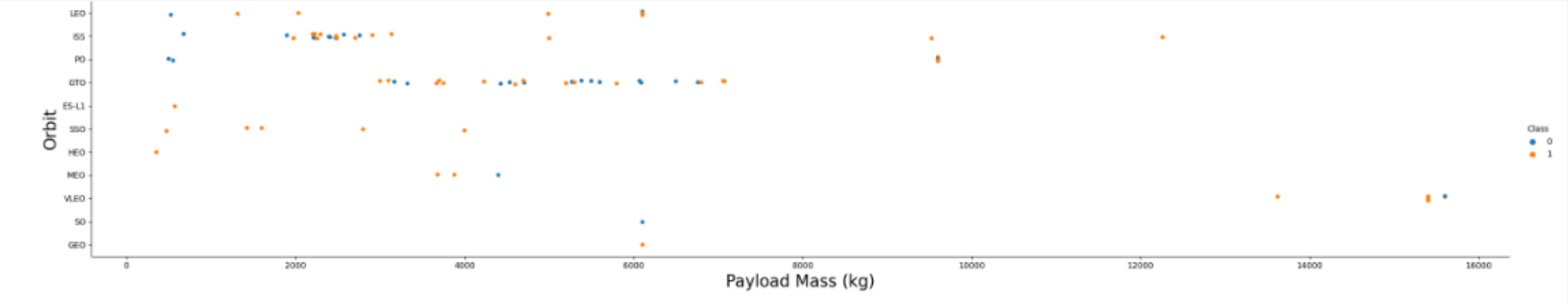


Flight Number vs. Orbit Type



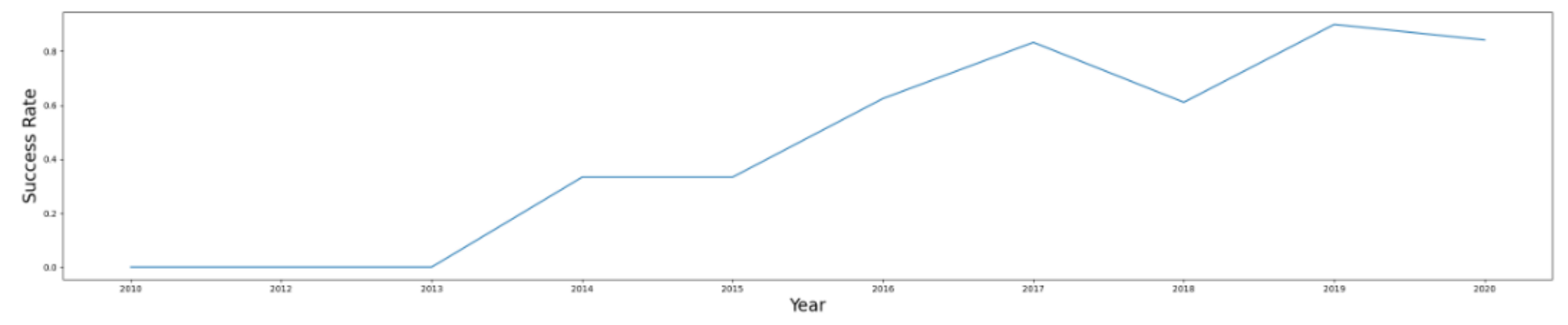
- We can see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



- 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



- The success rate has seen an increase between 2013-2017 and between 2018-2019.
- There was a slight dip between 2017-2018 and from 2019-2020.
- Overall, the success rate has improved since 2013.

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

There are 4 launch sites in the space mission :

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

Launch Site Names Begin with 'CCA'

Displays the names of all launch sites which starts with ‘CCA’

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

```
%sql select Launch_Site from SPACEXTBL where Launch_Site like 'CCA%' limit 5;
```

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

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

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 like 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596.0

The total payload mass carried by boosters launched by NASA (CRS) is 45596.0 Kgs.

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

The average payload mass carried by booster version F9 v1 is 2928.4 Kgs.

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) as First_Successful_Landing from SPACEXTBL where Landing_Outcome like 'Success (ground pad)'
```

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

First_Successful_Landing
01/08/2018

Successful Drone Ship Landing with Payload between 4000 and 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 Booster_Version from SPACEXTBL where Landing_Outcome = 'Success (drone ship)'
            and PAYLOAD_MASS__KG_ > 4000
            and PAYLOAD_MASS__KG_ < 6000
```

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

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

The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 are listed above using SQL EDA.

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

In [13]:
```
%%sql select count(Mission_Outcome) AS SuccessOutcome
        FROM SPACEXTBL
        WHERE Mission_Outcome LIKE 'Success%'
```

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

Out[13]:

SuccessOutcome
100

In [14]:
```
%%sql select count(Mission_Outcome) AS FailureOutcome
        FROM SPACEXTBL
        WHERE Mission_Outcome LIKE 'Failure%'
```

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

Out[14]:

FailureOutcome
1

The number of successful mission outcomes are 100 whereas there is only 1 mission which failed.

Boosters Carried Maximum Payload

The names of the booster which have carried the maximum payload mass have been listed using SQL EDA.

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

```
%%sql select Booster_Version,PAYLOAD_MASS__KG_ from SPACEXTBL
        where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
        order by Booster_Version
```

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

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600.0
F9 B5 B1048.5	15600.0
F9 B5 B1049.4	15600.0
F9 B5 B1049.5	15600.0
F9 B5 B1049.7	15600.0
F9 B5 B1051.3	15600.0
F9 B5 B1051.4	15600.0
F9 B5 B1051.6	15600.0
F9 B5 B1056.4	15600.0
F9 B5 B1058.3	15600.0
F9 B5 B1060.2	15600.0
F9 B5 B1060.3	15600.0

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 Booster_Version, Launch_Site, Landing_Outcome from SPACEXTBL
        where Landing_Outcome = 'Failure (drone ship)'
        and Date>='01-01-2015' and Date<='31-12-2015'
```

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

Booster_Version	Launch_Site	Landing_Outcome
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

```
%%sql select Landing_Outcome,count(Landing_Outcome) from SPACEXTBL
        where Date between '04-06-2010' and '20-03-2017' and Landing_Outcome like 'Success%'
        group by Landing_Outcome
        order by count(Landing_Outcome) desc
```

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

Landing_Outcome	count(Landing_Outcome)
Success	20
Success (drone ship)	8
Success (ground pad)	7

Section 3

Launch Sites Proximities Analysis

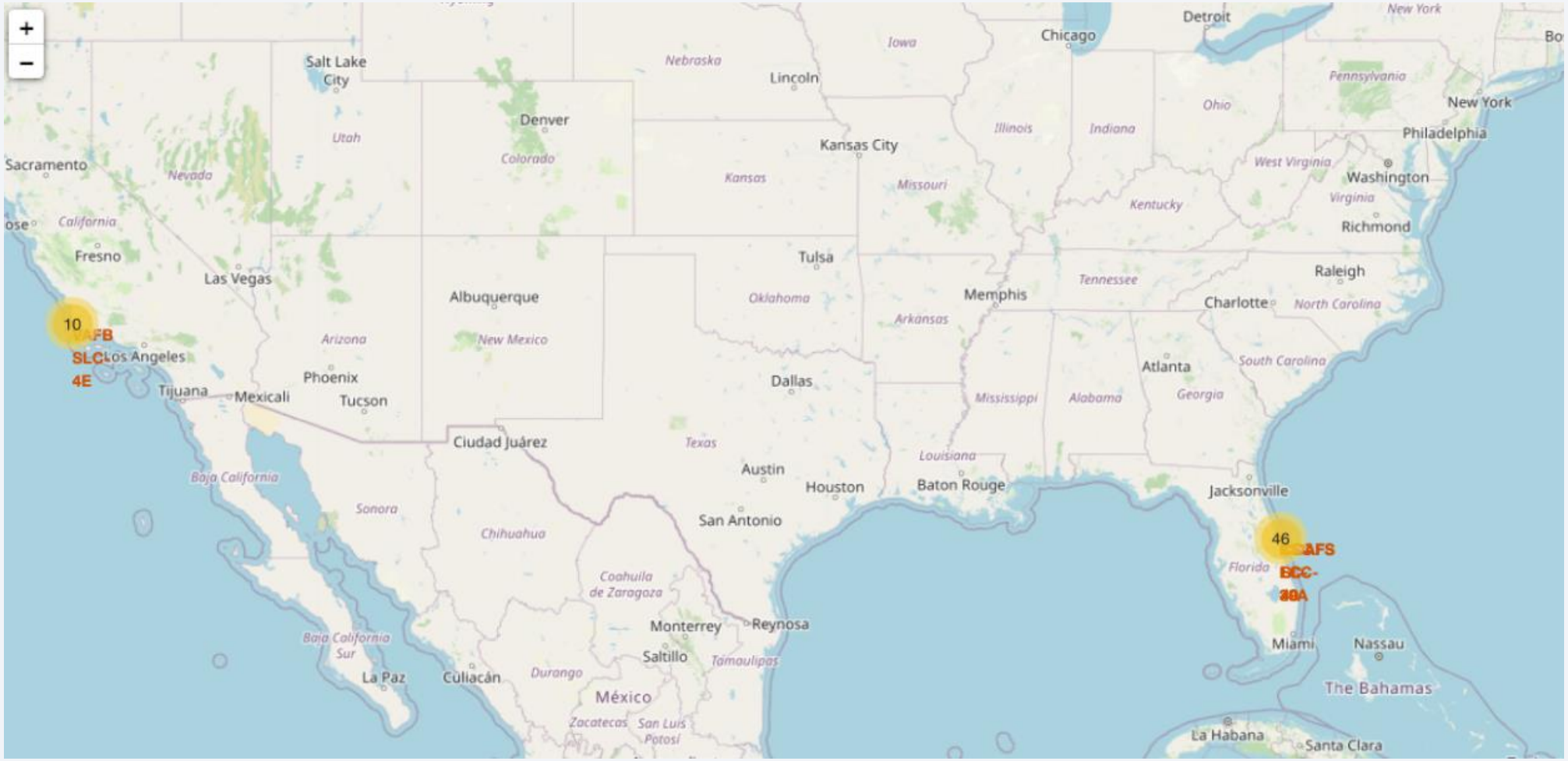
Mark all launch sites on a map

Mark all launch sites on a map

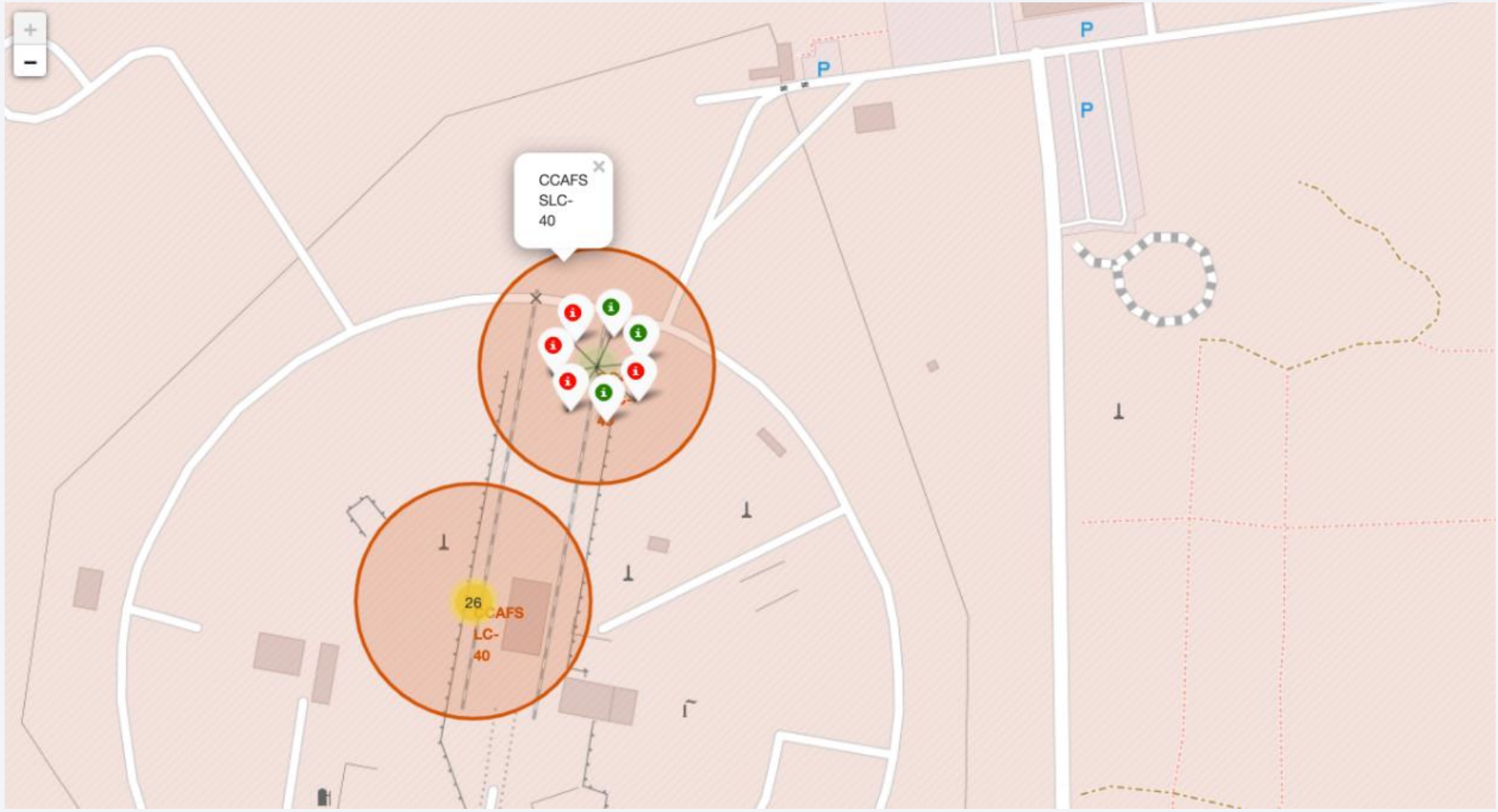
All of the launch sites are close to the equator. This is because rockets launched from sites near the Equator get an additional natural boost that helps save the cost of putting in extra fuel and boosters.

Also, yes, All the launch sites are close to the coasts, so that, just in case of failure of the launch, the satellite does not fall on built-up hinterland.

Mark the success/failed launches for each site on the map



Mark the success/failed launches for each site on the map



Mark the success/failed launches for each site on the map

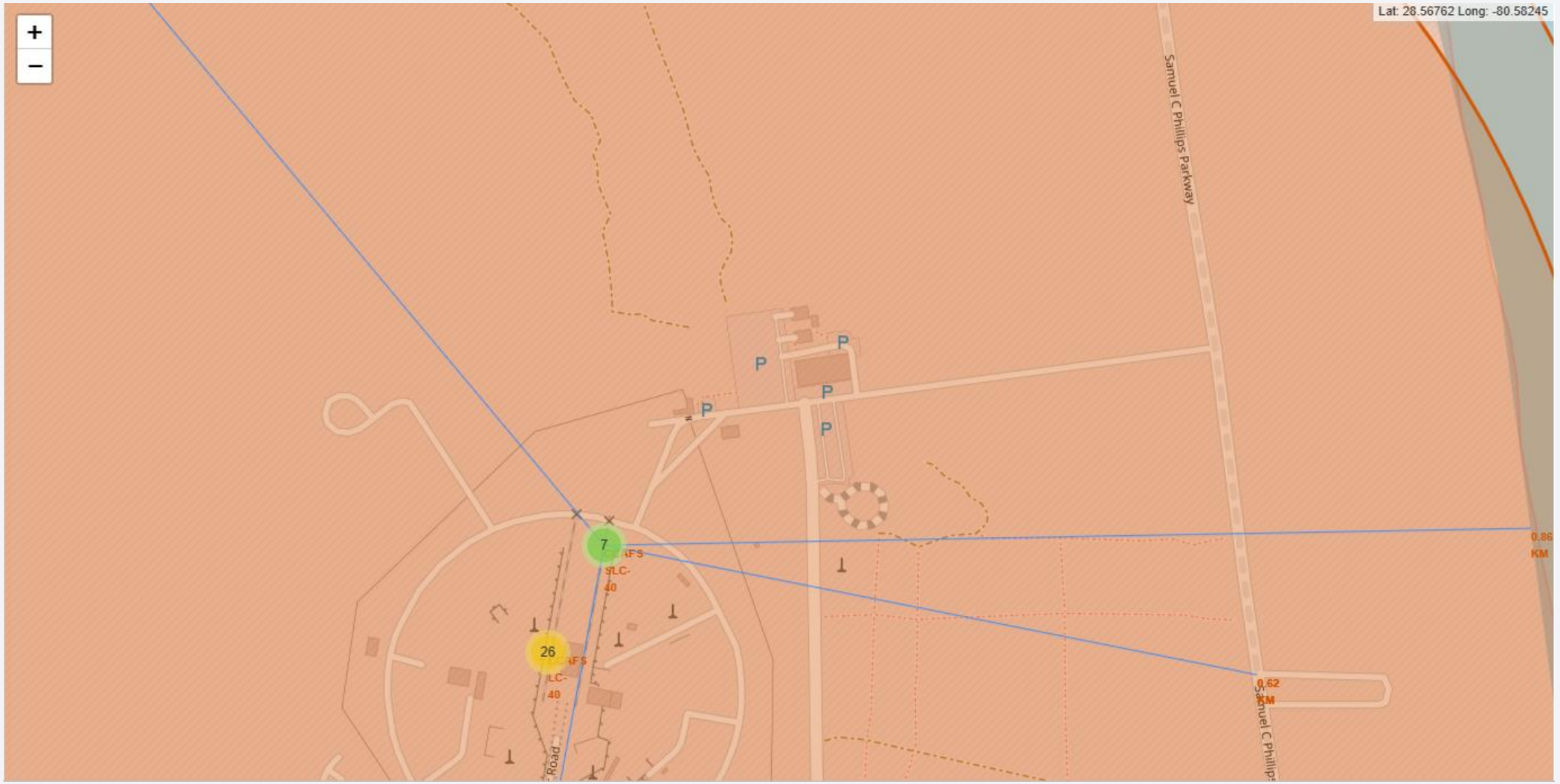
From the color-labeled markers, we should be able to easily identify which launch sites have relatively high success rates.

The ‘green’ label marker implies successful launch from that launch site.

The ‘red’ label marker implies unsuccessful launch from that launch site.

The launch Site KSC LC-39A has a very high success rate, whereas launch site CCAFS SLC-40 has a 3/7 success rate (42.9%).

Calculate the distances between a launch site to its proximities



Calculate the distances between a launch site to its proximities

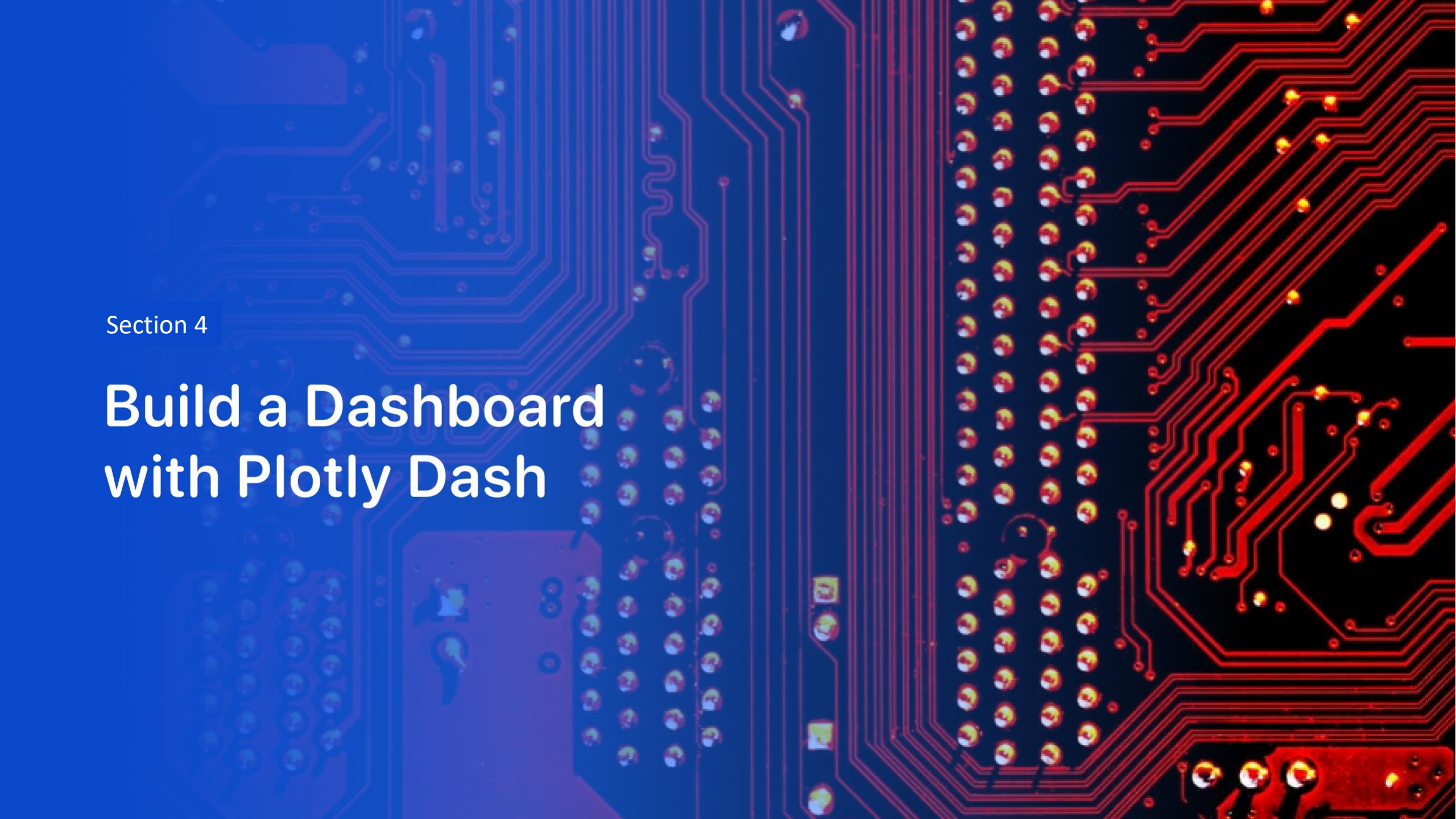
The colored blue lines from the launch site numbered ‘7’ shows the distance between various proximities such as railways, highways, city, coastlines and itself.

Closest City is Melbourne with a distance of 50.30645509775321 kms.

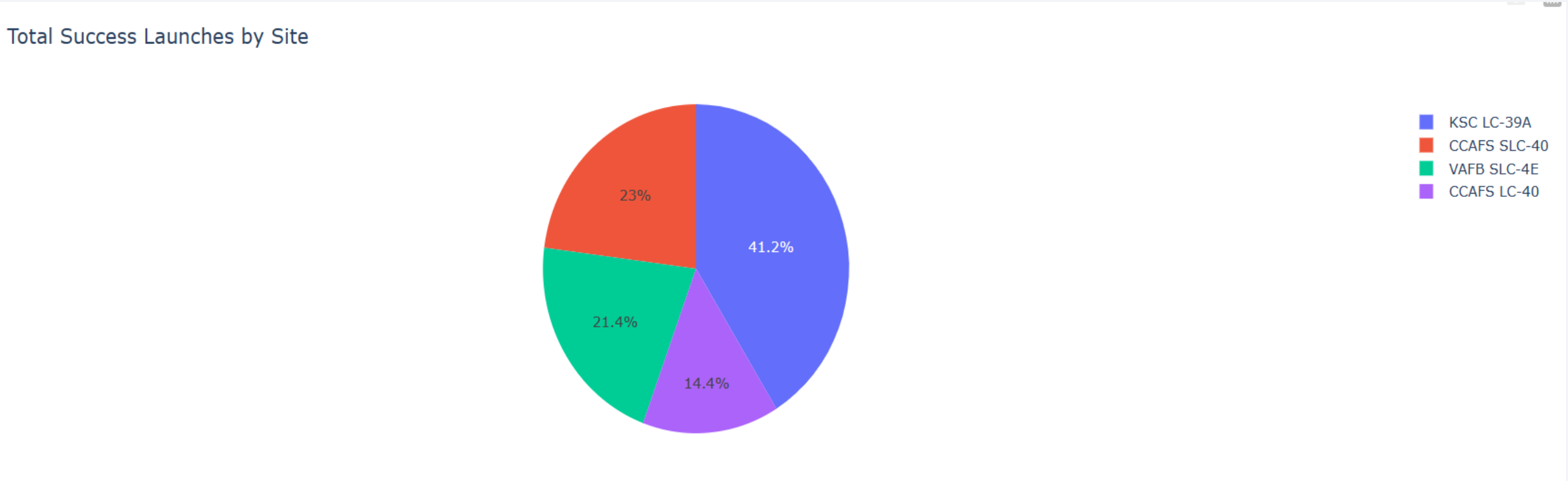
Closest Railway starts after Titan III Road with a distance of 1.285787892865894 3 kms.

Closest Highway is Samuel C Philips Parkway with a distance of 0.6193437173748175 kms.

Coastline Distance 0.8627671182499878 kms.

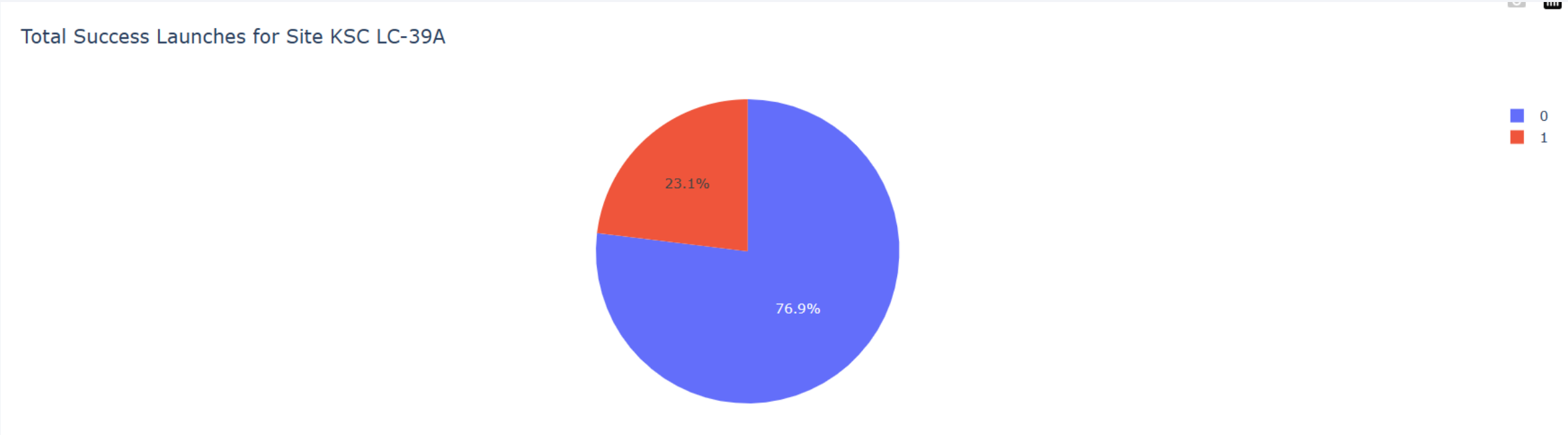

Section 4
Build a Dashboard with Plotly Dash

Total Success Launches by site



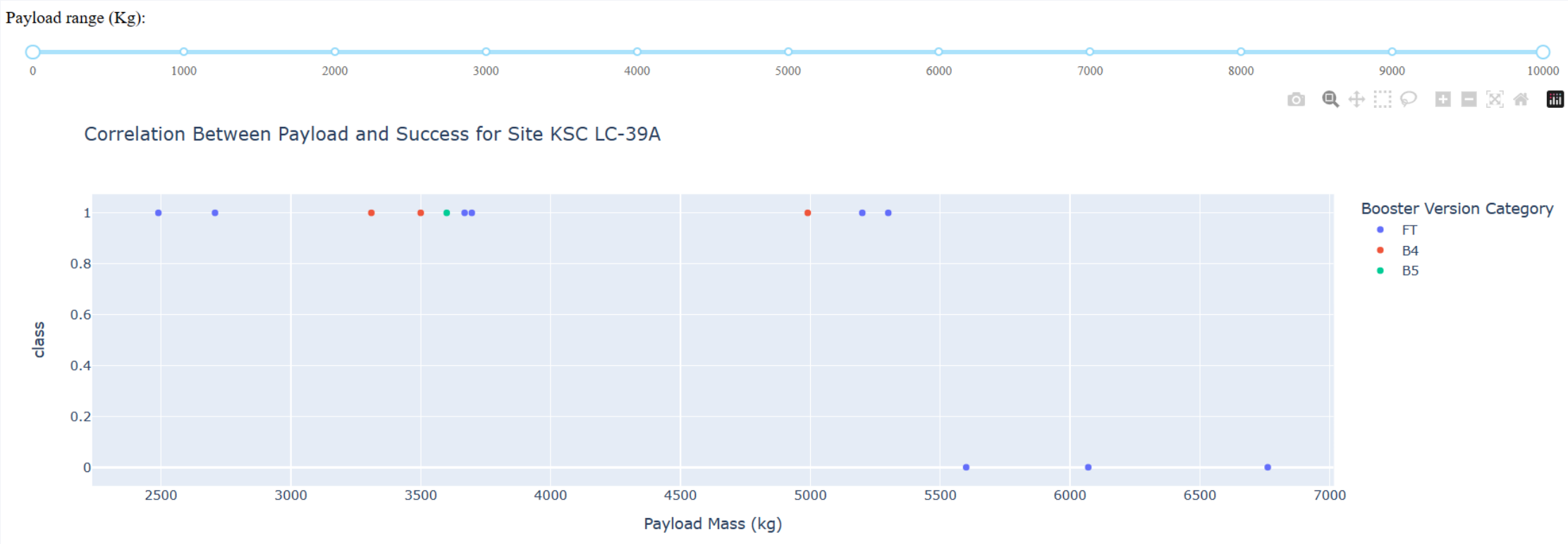
The launch site KSC LC-39A has the highest number of successful launches (41.2%) with the least successful launches being CCAFS LC-40 with just 14.4% of the share in pie chart.

Most Successful Launch Site



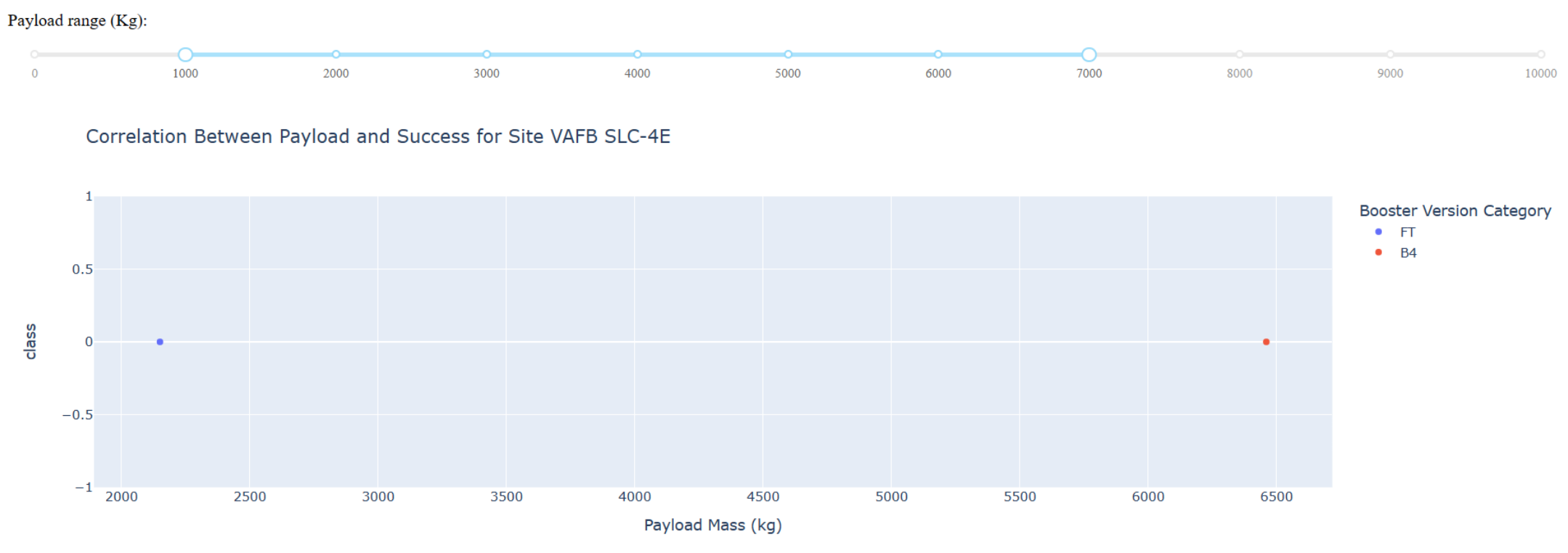
The launch site KSC LC-39A is the most successful launch site with 76.9% success rate and 23.1% failure rate.

Payload vs Success for launch site KSC LC-39A



The payloads below 5500 Kgs have 100% success rate while those above 5500 Kgs have 0% success rate.

Payload vs Success for launch site VAFB SLC-4E



The payloads between 1000 and 7000 Kgs have 100% failure rate or 0% success rate.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

All the models performed at about the same level and had the same scores and accuracy (83.33%).

The Decision Tree model slightly outperformed the rest when looking at .best_score_.

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.800000	0.800000	0.733333	0.800000
F1_Score	0.888889	0.888889	0.846154	0.888889
Accuracy	0.833333	0.833333	0.833333	0.833333

```
Best model is DecisionTree with a score of 0.8732142857142857
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2,
'splitter': 'best'}
```

.best_score_ is the average of all cv folds for a single combination of the parameters.

Confusion Matrix

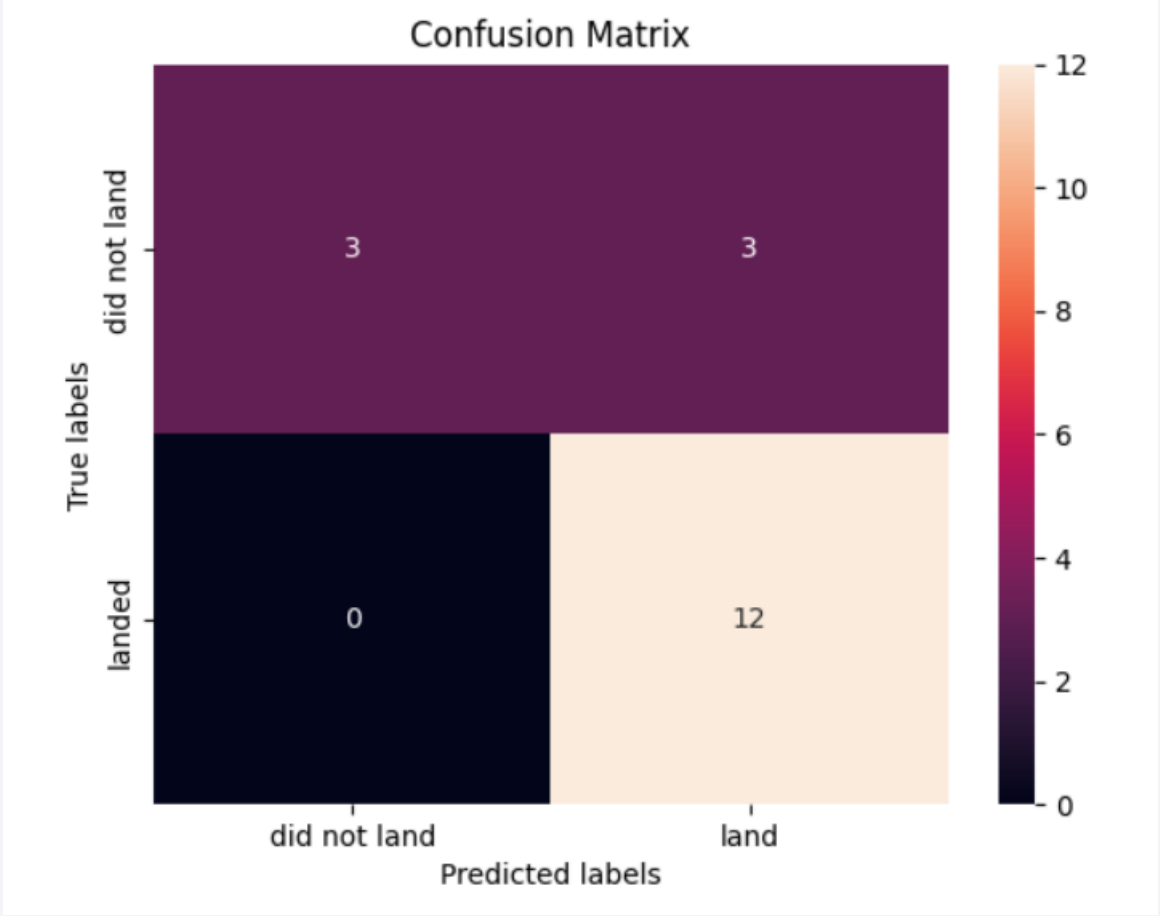
Confusion matrix is an indication of a model's performance.

It consists of : False Positives(3), False Negatives(0), True Negatives(3) and True Positives(12).

Precision = TP/(TP+FP) = 12/15 = 0.80

Recall = TP/(TP+FN) = 12/12 = 1

F1 score = Harmonic mean of Precision and Recall = 0.89



Conclusions

- Exploratory data analysis results
  - ✓ Launch success has improved over time.
  - ✓ KSC LC-39A has the highest success rate among landing sites.
  - ✓ Orbits ES-L1, GEO, HEO and SSO have a 100% success rate.
- Interactive analytics results
  - ✓ Most launch sites are near the equator, and all are close to the coast.
  - ✓ Launch sites are far enough away from anything a failed launch can damage (city, highway, railway), while still close enough to bring people and material to support launch activities.
- Predictive analysis results
  - ✓ Decision Tree model is the best predictive model for the dataset.

Appendix

Github URL to SpaceX dataset in CSV format : https://github.com/AnishD642/IBM-Data-Science-Capstone/blob/d9d5c06ac44716cd0ab27e219753f83d6f3cedd8/Spacex.csv

Thank you!