IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dino Dobrinic
11/16/2023

Outline

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

Executive Summary

- Module 1: Data Collection
  - Methodology: RESTful API to gather data on Falcon 9 launches
  - Results: Dataset containing information on Falcon 9 first-stage landings
- Module 2: Data Wrangling and Dashboard
  - Methodology: Built a dashboard using Plotly Dash to interactively analyze launch records
  - Results: Developed an interactive dashboard for exploring launch records
- Module 3: Interactive Map
  - Methodology: Utilized Folium to create an interactive map. Analyzed the launch site proximity and visualized it on the map.
  - Results: Generated an interactive map illustrating the launch site locations
- Module 4: Machine Learning for Landing Prediction
  - Methodology: Applied machine learning algorithms (SVM, Classification Trees, Logistic Regression) for landing prediction
  - Results: Determined the effectiveness of each model in predicting landing success

Introduction

- SpaceX stands as a revolutionary force in the space industry, disrupting traditional norms by offering rocket launches, particularly Falcon 9, at an unprecedented cost of as low as 62 million dollars, a significant reduction compared to other providers charging upwards of 165 million dollars per launch. This substantial cost-saving innovation is primarily attributed to SpaceX's groundbreaking concept of reusing the first stage of the rocket by successfully re-landing it for deployment in subsequent missions. The iterative application of this process promises further reductions in launch costs.
- As a data scientist for a startup aiming to compete with SpaceX, the primary objective of this project is to establish a robust machine learning pipeline capable of predicting the landing outcome of Falcon 9's first stage in future missions. The success of this predictive model is pivotal in determining the optimal bidding price against SpaceX for rocket launches.
- Key challenges addressed in this project include:
  - Identifying all relevant factors influencing the landing outcome.
  - Investigating the relationships between each variable and their impact on the overall outcome.
  - Determining the optimal conditions necessary to enhance the probability of a successful landing.

Section 1

Methodology

Methodology

Executive Summary

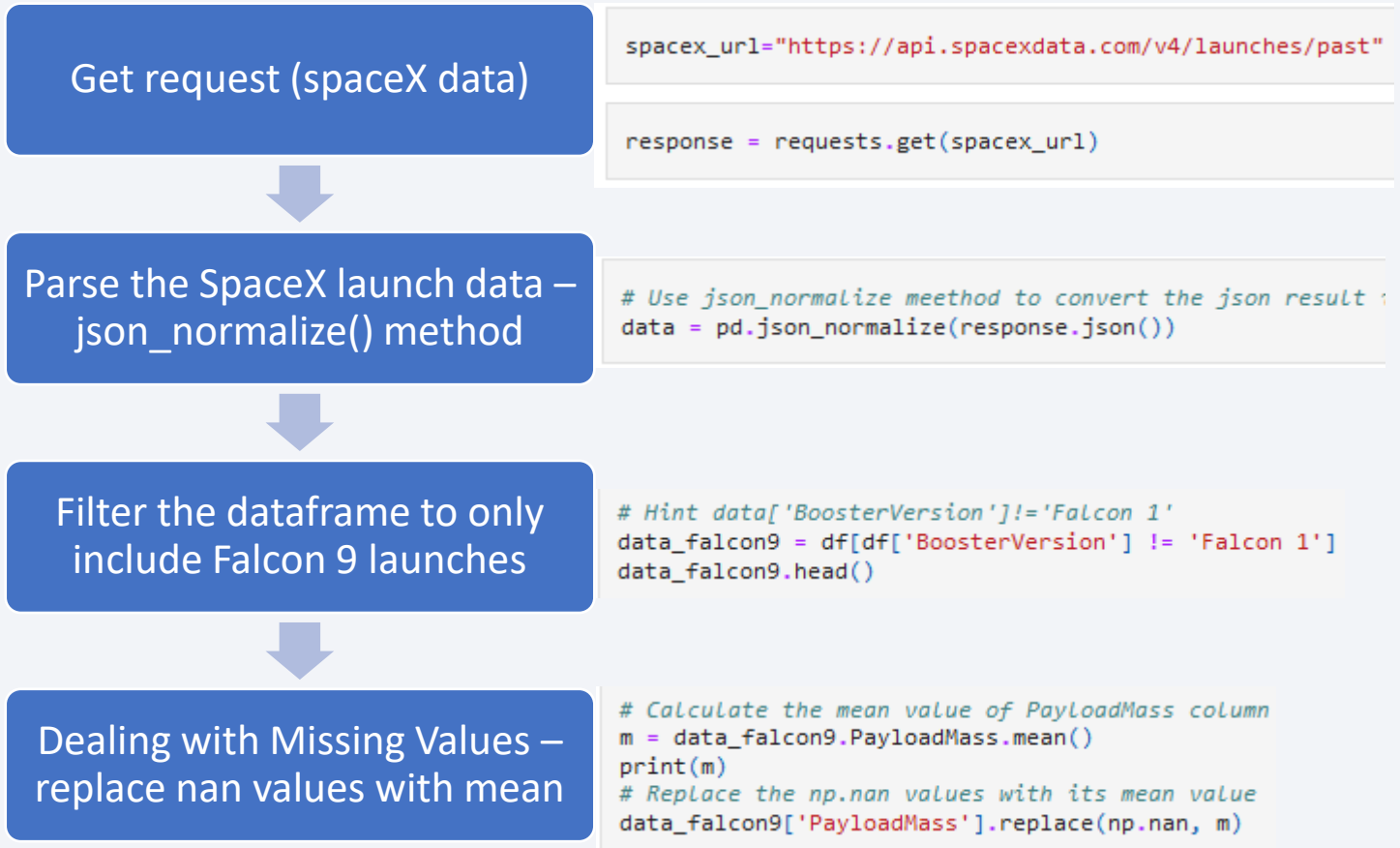
- Data collection methodology:
  - Data was collected using SpaceX REST API and web scrapping from Wikipedia
- Perform data wrangling
  - Data was processed using one-hot encoding for categorical features
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- Data collection involves the systematic gathering and measurement of information pertaining to specific variables within a predefined system. This process allows for the examination of pertinent questions and the evaluation of outcomes. In this case, the dataset was acquired through both REST API and Web Scraping methods from Wikipedia.
- To collect data via REST API, we initiated the process with a get request. Subsequently, we decoded the response content into JSON format and transformed it into a pandas dataframe using the json_normalize() function. Following this, we performed data cleaning procedures, checked for missing values, and filled any gaps with the necessary information.
- In the case of web scraping, we utilized BeautifulSoup to extract launch records presented in HTML tables. The extracted data was then parsed and converted into a pandas dataframe, facilitating further analysis.

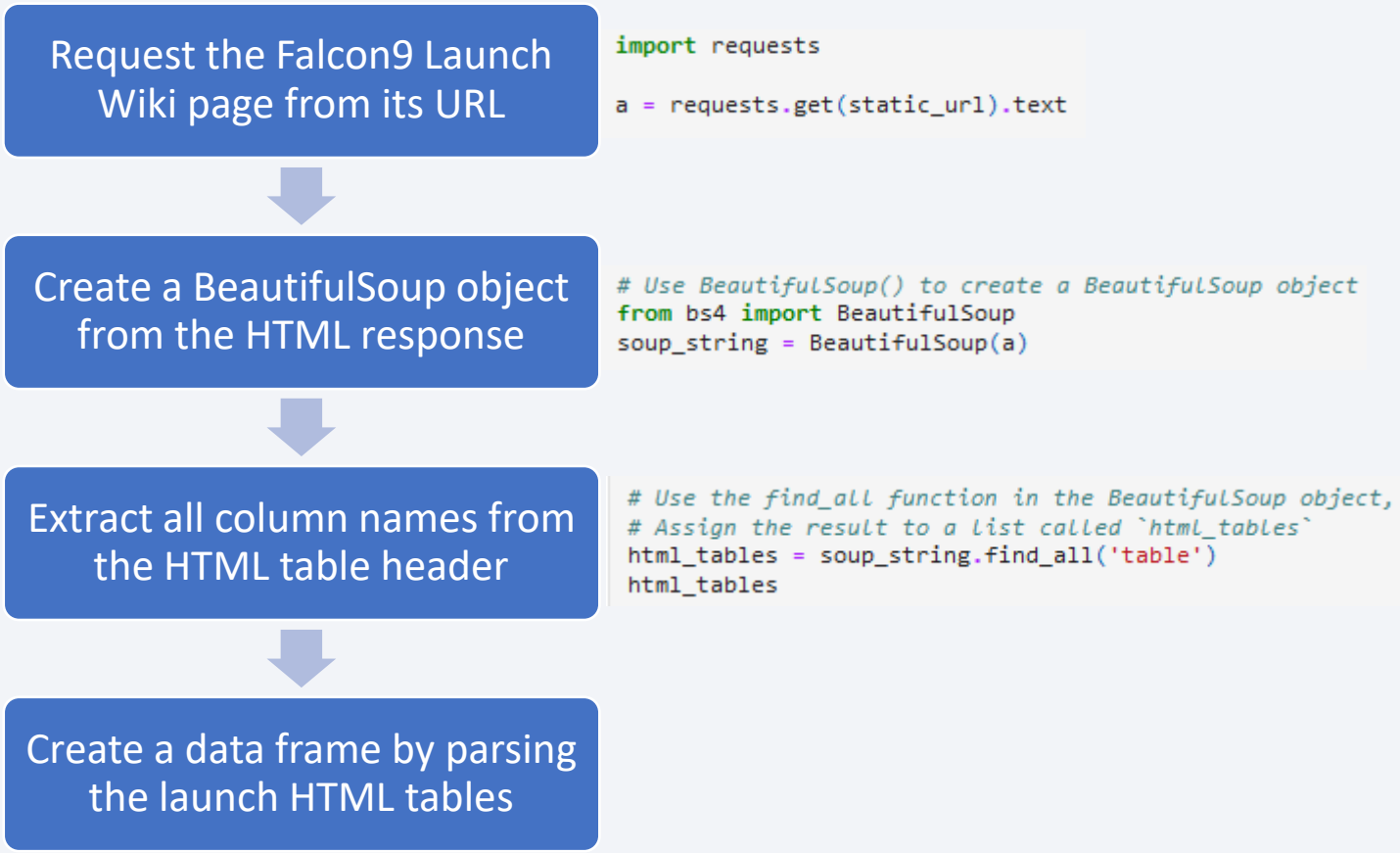
Data Collection – SpaceX API

- Request and parse the SpaceX launch data using the GET request
- Use json_normalize() method to convert json result to dataframe
- Dealing with Missing Values (i.e., performed data cleaning and filling the missing value)
- From:
  - https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



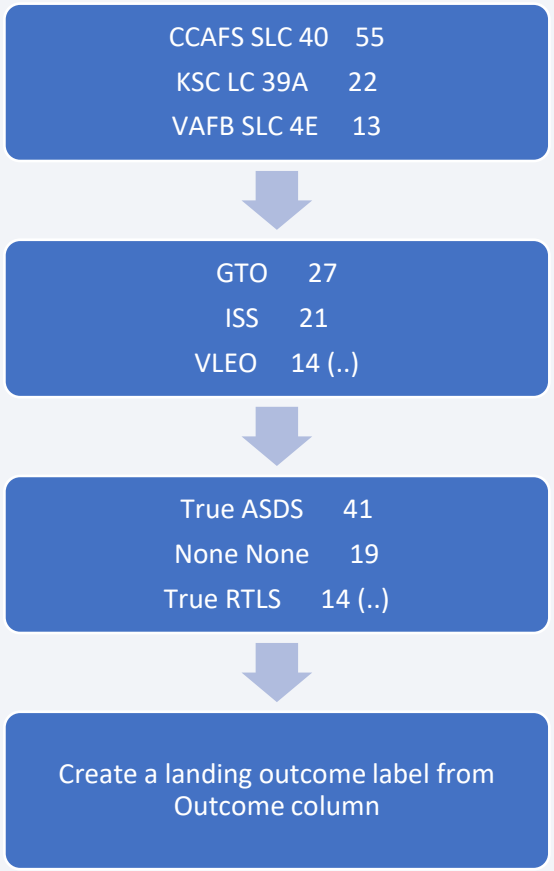
Data Collection - Scraping

- Web scraping of the Falcon 9 launch records with Beautiful Soup
- Extract a Falcon 9 launch records HTML table from Wikipedia
- Parse the table and convert it into a Pandas data frame
- From:
  - https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Calculate the number of launches on each site
- Calculate the number and occurrence of each orbit
- Calculate the number and occurence of mission outcome of the orbits
- Create a landing outcome label from Outcome column
- From:
  - https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb




```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
# landing_outcomes = values on Outcome column
landing_outcomes = df['Outcome'].value_counts()
landing_outcomes
```

EDA with Data Visualization

- Scatter Plot
  - Flight Number vs Payload Mass
  - Flight Number vs Launch Site
  - Payload vs Launch Site
  - Orbit vs Flight Number
  - Payload vs Orbit Type
  - Orbit vs Payload Mass

  Scatter plots are an essential type of data visualization that shows relationships between variables.

- Bar Graph
  - Success rate vs Orbit

  Bar graphs show the relationship between numeric and categoric variables.

- Line Graph
  - Sucess Rate vs Year

  Line graphs is commonly drawn to show information that changes over time.

From: https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/jupyter_labs_eda_dataviz.ipynb

EDA with SQL

- SpaceX DataSet was firstly loaded into the corresponding table in Db2 database (Link: https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb)
- Then, a SQL queries were executed to answer the following questions:
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'CCA'
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first succesful landing outcome in ground pad was acheived.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass.
  - 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 landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

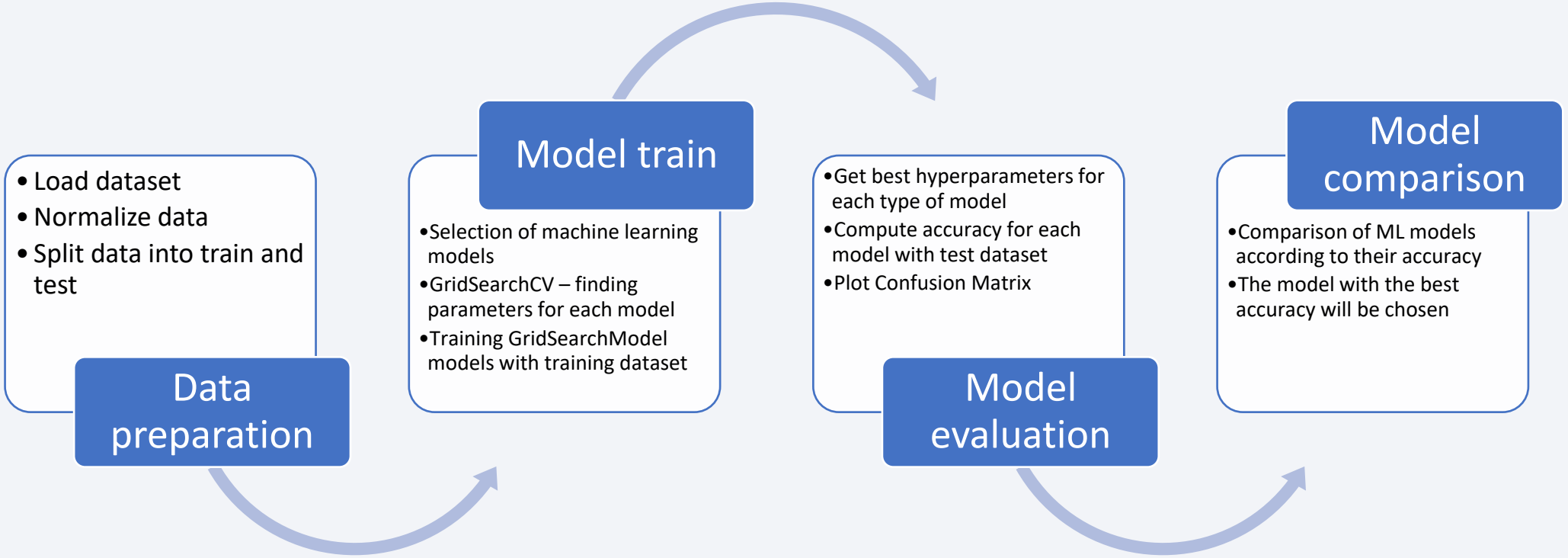
Build an Interactive Map with Folium

- Folium makes it easy to visualize data that's been manipulated in Python on an interactive leaflet map.
  - Folium.map.Marker and folium.Circle were used to show NASA Johnson Space Center and each launch site
  - MarkerCluster() method was used to display multiple objects located on the same coordinate
  - Then dataframe launch_outcomes(failure,success) with classes 0 and 1 was mapped Red and Green markers on the map using MarkerCluster() method
- These objects are created in order to understand better the problem and the data. We can show easily all launch sites, their surroundings and the number of successful and unsuccessful landings.
- Link to the code: https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Combined with Python, Plotly Dash delivers interactive, customizable data apps.
- For this project, a dashboard was created in order to analyze launch records interactively
- Some of the methods that were added to the dashboard are:
  - dcc.Dropdown (choose launch site)
  - dcc.Graph(pie chart and scatter plot)
  - dcc.RangeSlider(select payload range)
- Link to the code: https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/spacex_dash_app_2.py

Predictive Analysis (Classification)

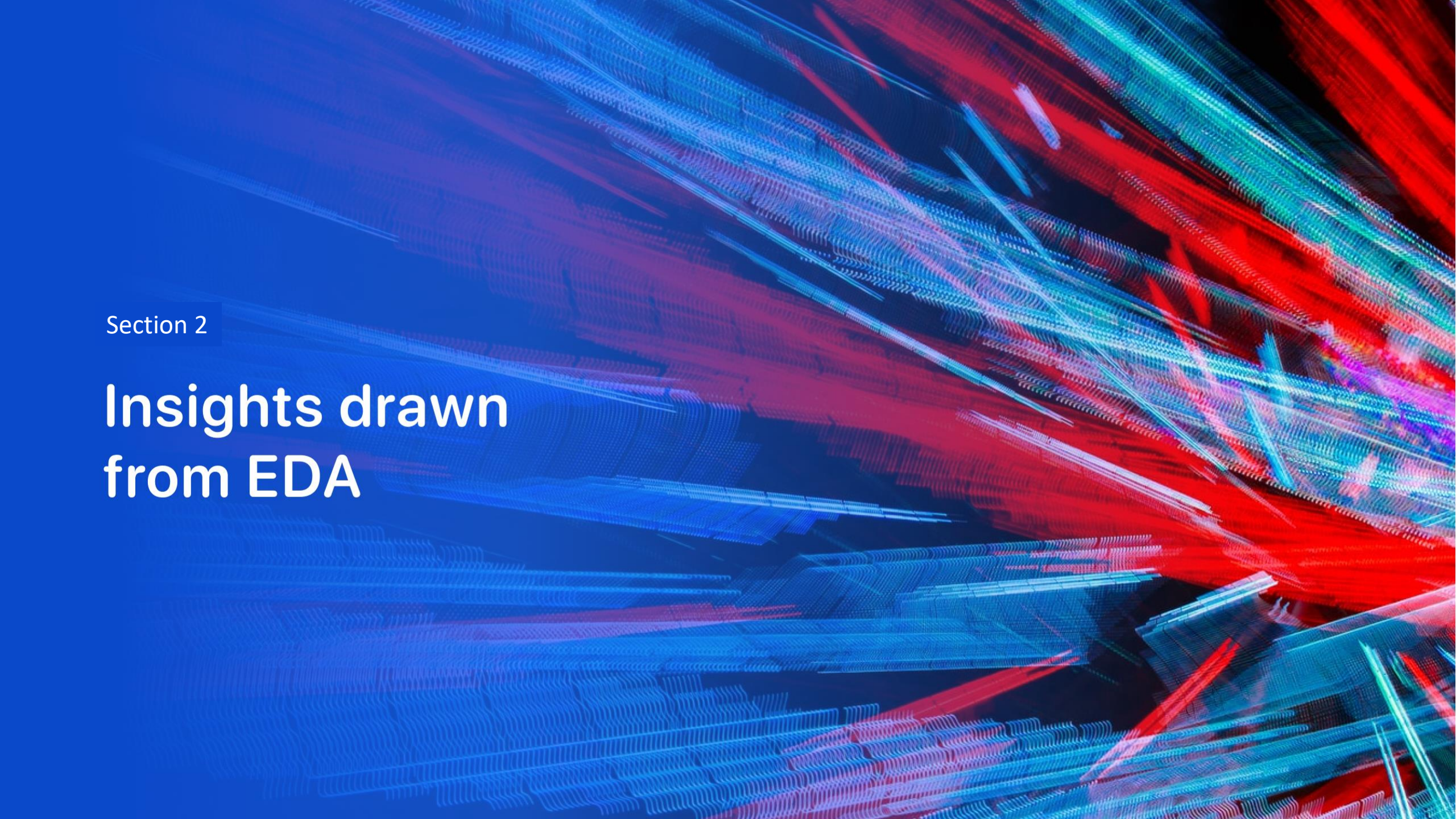


Link to the code:

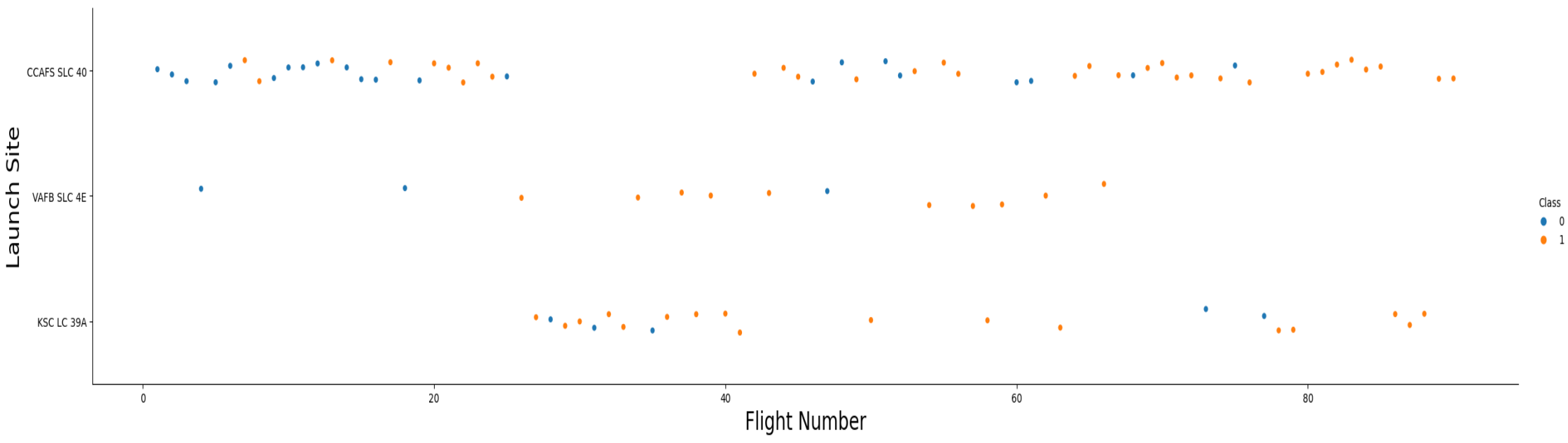
https://github.com/ddobrinic/IBM-Applied-Data-Science-Capstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5_jupyterlite.ipynb

Results

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

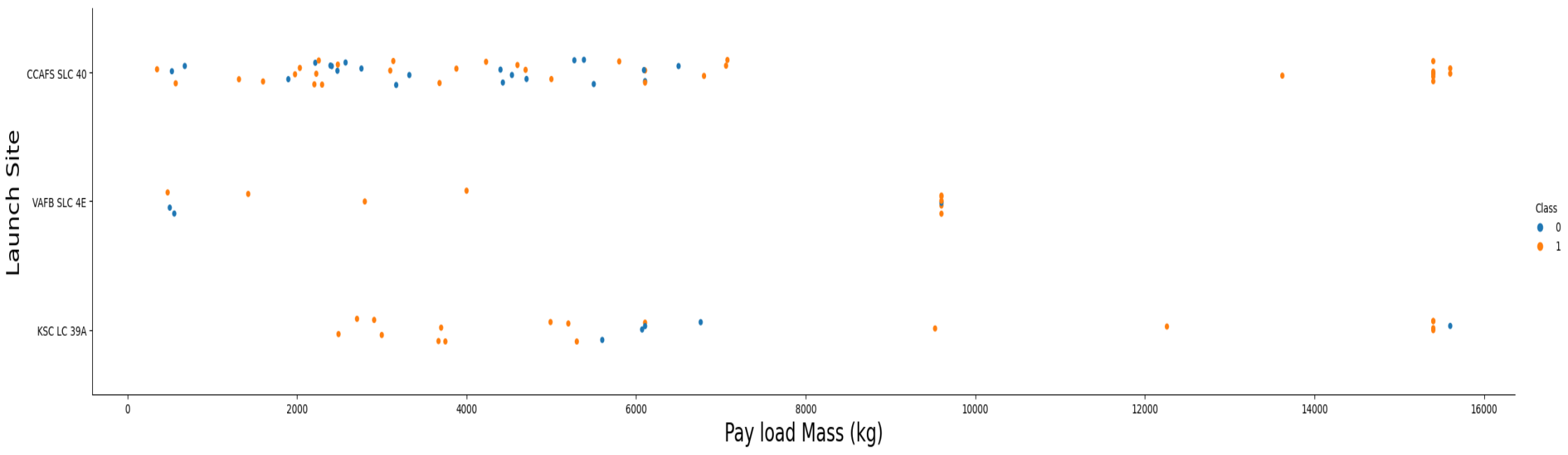

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



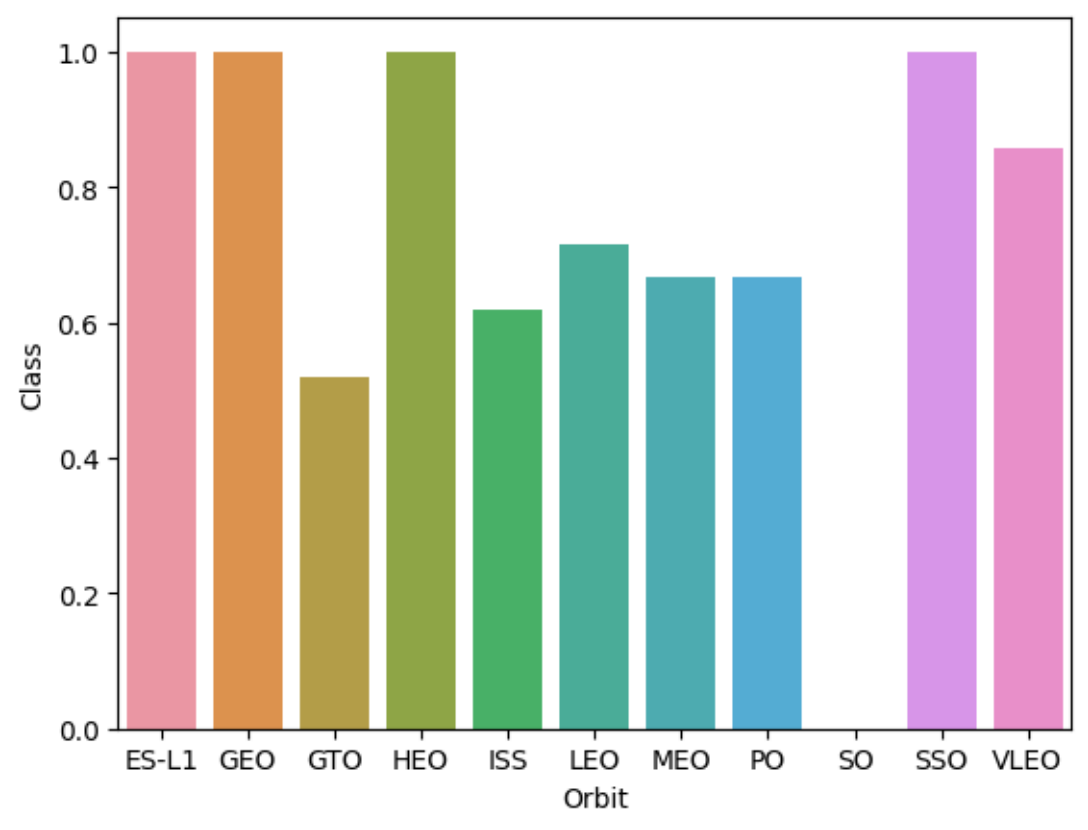
- This scatter plot shows the relationship between the number of flights and launch sites. From this plot, we can see that, for each site, the success rate is increasing with the number of flights.

Payload vs. Launch Site



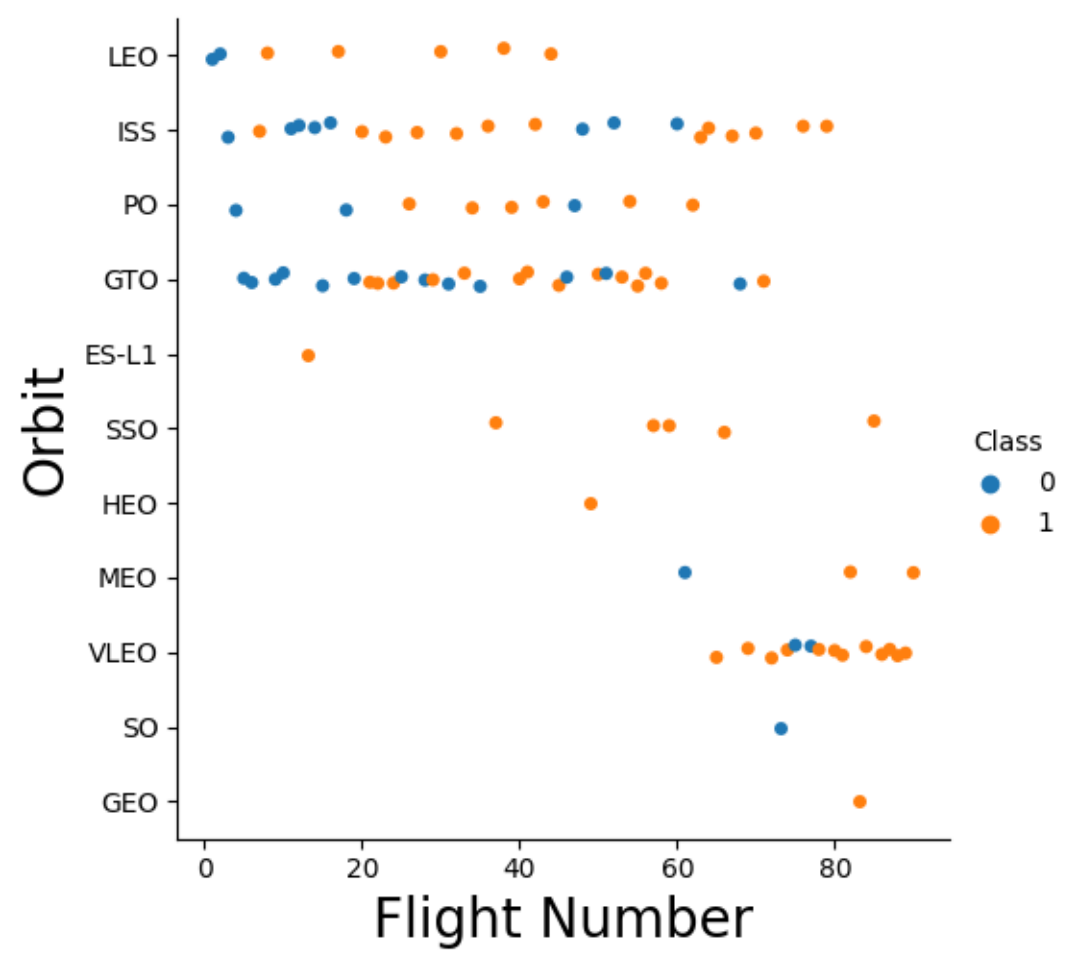
- This scatter plot shows the relationship between the Payload mass and launch sites. From this plot, we can see that, there is no clear pattern to say the launch site is dependent to the pay load mass for the positive success rate.
- CCAFS SLC 40 points out as the launch site with most sucessful landings with higher payload mass

Success Rate vs. Orbit Type



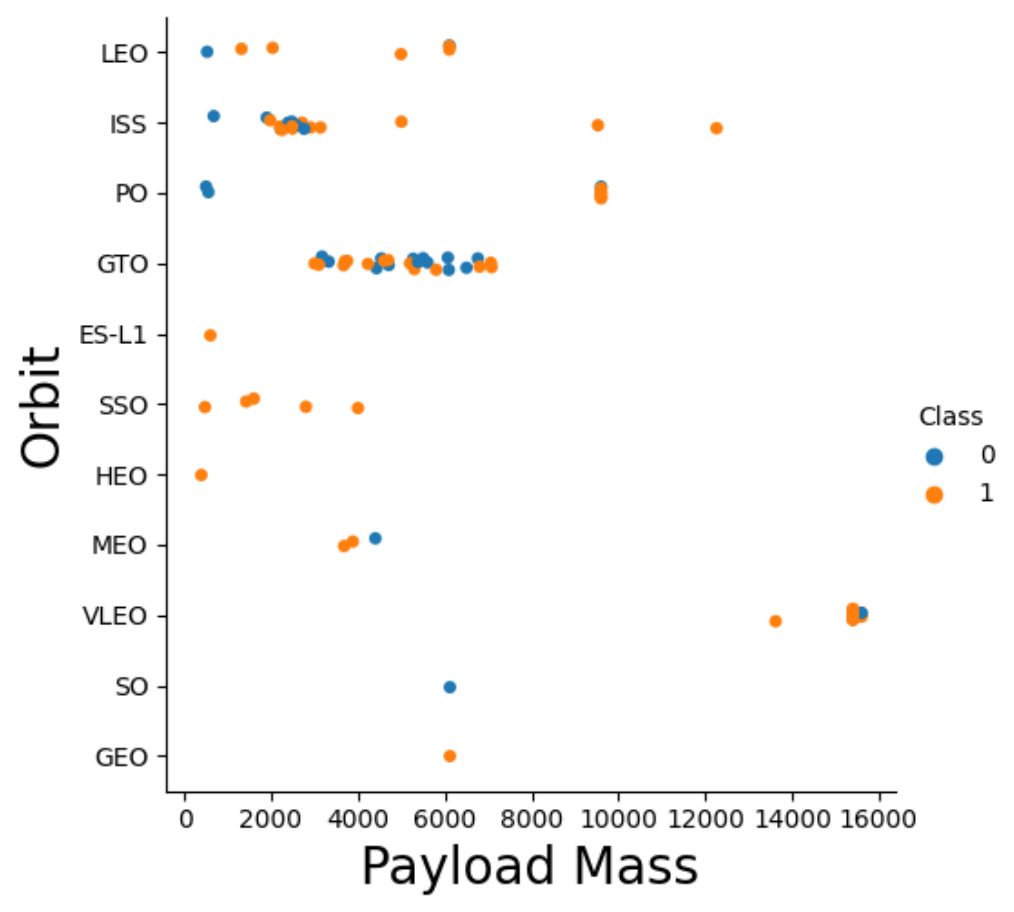
- This bar chart shows the success rate for different orbit types. We can see that ES-L1, GEO, HEO, and SSO have 100% success rate
- However, some orbits have only 1 occurence and therefore, a deeper analysis is needed to conclude which orbit has the best sucessful rate

Flight Number vs. Orbit Type



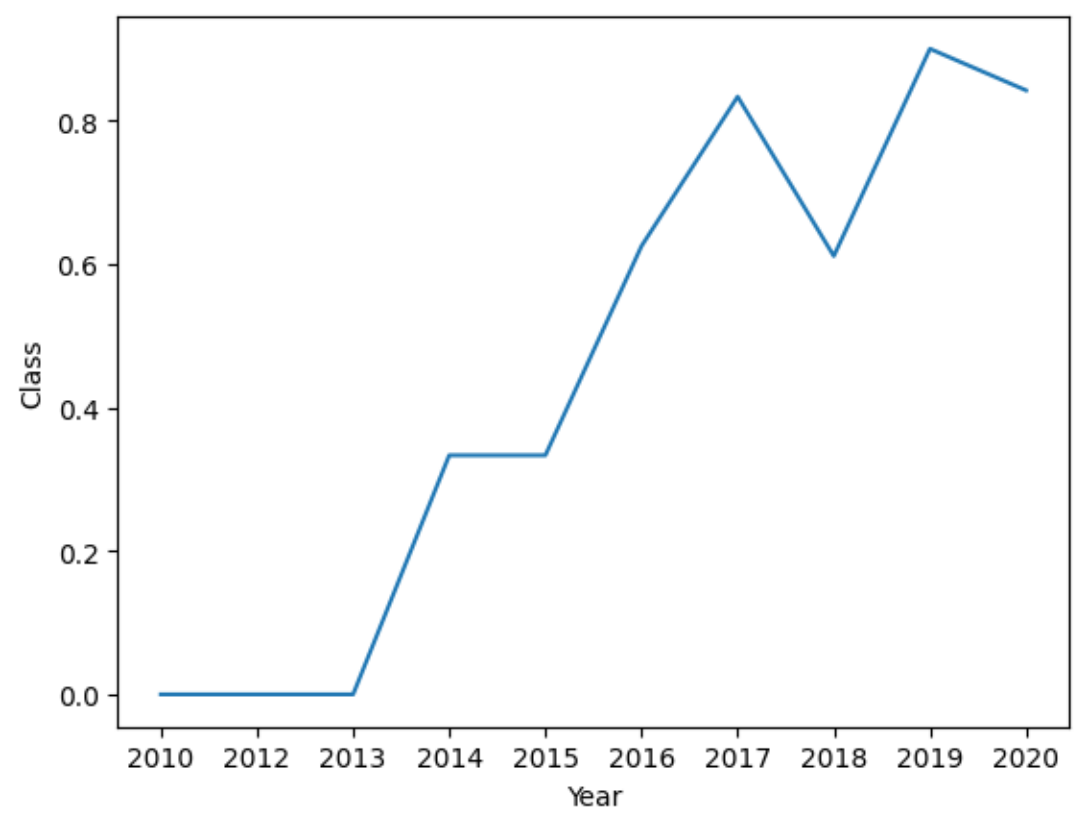
- LEO orbit the Success appears related to the number of flights (it increases); on the other hand, there seems to be no relationship between flight number when in GTO orbit.
- Also, orbits with only 1 landings should be excluded from the further analysis

Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar, VLEO 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



- You can observe that the success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

All Launch Site Names

```
In [9]: %sql SELECT DISTINCT("Launch_Site") FROM SPACEXTBL
 * sqlite:///my_data1.db
Done.
```

Out[9]:

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

- DISTINCT keyword in SQL eliminates all duplicate records from the result returned by the SQL query.
- Only unique records are returned when the DISTINCT keyword is used while fetching records from a table having multiple duplicate records.
- Unique launch sites are:
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40

Launch Site Names Begin with 'CCA'

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

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

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_
6/4/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0
12/8/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525
10/8/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500
3/1/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677

- The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.
- The percent sign % represents zero, one, or multiple characters
- This query displayed 5 records where launch sites begin with `CCA`

Total Payload Mass

Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS "total payload mass" FROM SPACEXTBL WHERE "Customer" = "NASA (CRS)"
```

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

total payload mass
45596

- This query returns the sum of all payload masses where the customer is NASA (CRS).

Average Payload Mass by F9 v1.1

Display average payload mass carried by booster version F9 v1.1

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") AS "average payload mass" FROM SPACEXTBL WHERE "Booster_Version" = "F9 v1.1"
```

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

average payload mass
2928.4

- This query returned the average payload mass carried by booster version F9 v1.1
- The result is 2928.4 kg

First Successful Ground Landing Date

```
In [23]: %sql SELECT MIN("Date") FROM SPACEXTBL WHERE "Landing_Outcome" = "Success (ground pad)"
 * sqlite:///my_data1.db
Done.
Out[23]: MIN("Date")
         1/8/2018
```

- This query returned the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here

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
```

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

Booster_Version
F9 v1.0 B0003
F9 v1.0 B0004
F9 v1.0 B0005
F9 v1.0 B0006
F9 v1.0 B0007
F9 v1.1 B1003

- This query returned the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%sql SELECT COUNT(MISSION_OUTCOME) AS "Successful Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Success%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32731/bludb
Done.
```

Successful Mission
100

```
%sql SELECT COUNT(MISSION_OUTCOME) AS "Failure Mission" FROM SPACEX WHERE MISSION_OUTCOME LIKE 'Failure%';
```

```
 * ibm_db_sa://zpw86771:***@fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.clou
d:32731/bludb
Done.
```

Failure Mission
1

- This query listed the total number of successful and failure mission outcomes

Boosters Carried Maximum Payload

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

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTE
```

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

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

- This query showed the booster that have carried the maximum payload using a subquery in the WHERE clause and the MAX() function.

2015 Launch Records

```
%sql SELECT substr("DATE", 4, 2) AS MONTH, "BOOSTER_VERSION", "LAUNCH_SITE" FROM SPACEXTBL\
WHERE "LANDING _OUTCOME" = 'Failure (drone ship)' and substr("DATE",7,4) = '2015'
```

MONTH	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

- This query returns month, booster version, launch site where landing was unsuccessful and landing date took place in 2015. Substr function process date in order to take month or year. Substr(DATE, 4, 2) shows month. Substr(DATE,7, 4) shows year.

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

SQL Query

```
%sql SELECT "LANDING _OUTCOME", COUNT("LANDING _OUTCOME") FROM SPACEXTBL\
WHERE "DATE" >= '04-06-2010' and "DATE" <= '20-03-2017' and "LANDING _OUTCOME" LIKE '%Success%'\
GROUP BY "LANDING _OUTCOME" \
ORDER BY COUNT("LANDING _OUTCOME") DESC ;
```

Results

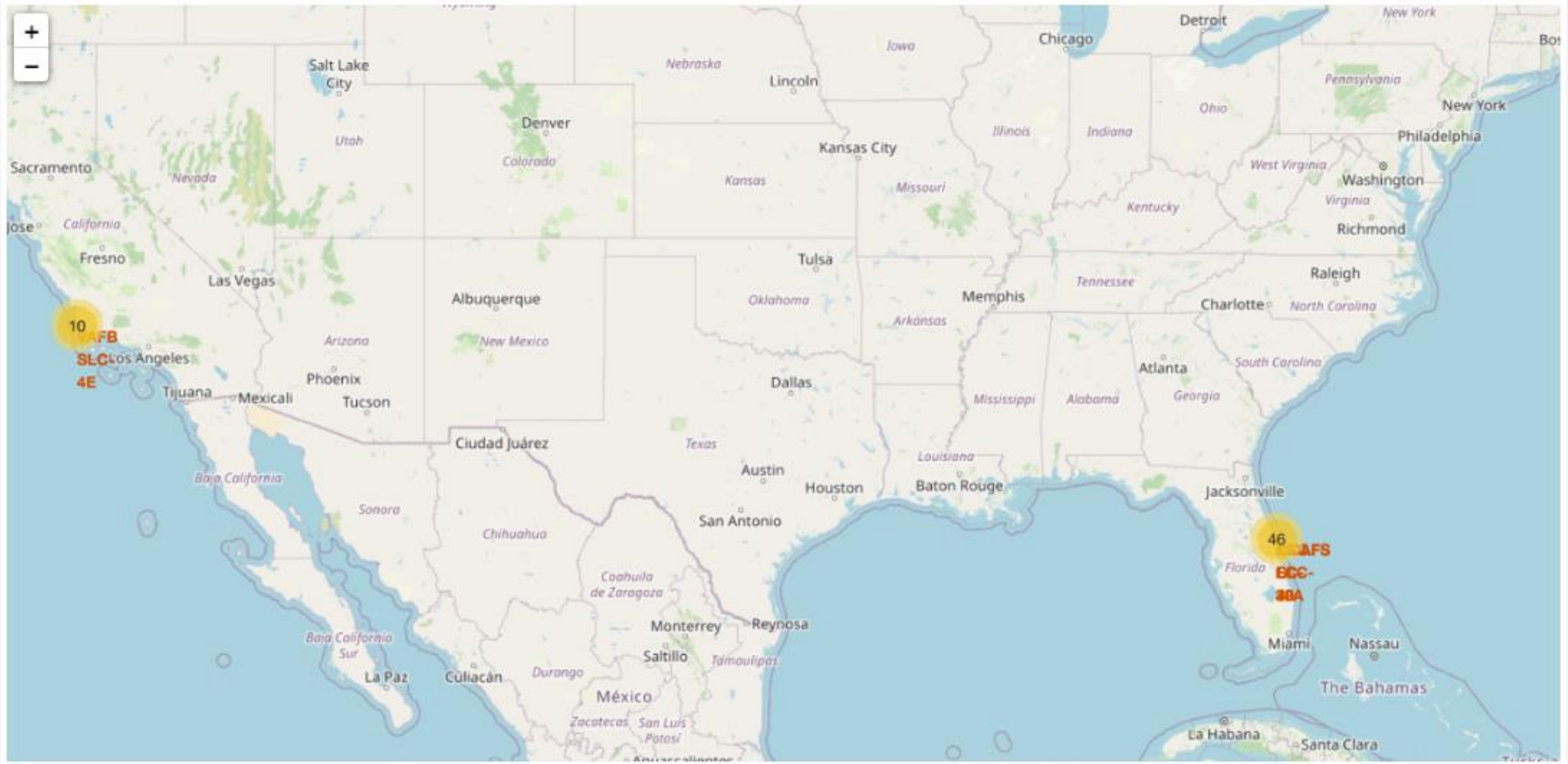
Landing _Outcome	COUNT("LANDING _OUTCOME")
Success	20
Success (drone ship)	8
Success (ground pad)	6

- This query returns landing outcomes and their count where mission was successful and date is between 04/06/2010 and 20/03/2017. The GROUP BY clause groups results by landing outcome and ORDER BY COUNT DESC shows results in decreasing order.

Section 3

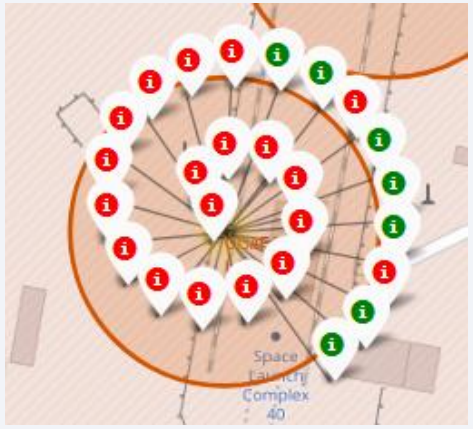
Launch Sites Proximities Analysis

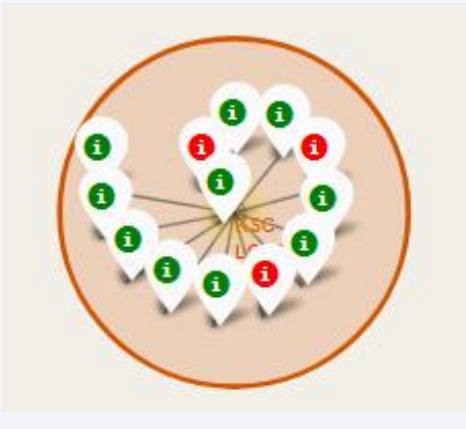
Folium Map - Space X Launch Sites



- Launch sites for Space X are located on the coast of the United States

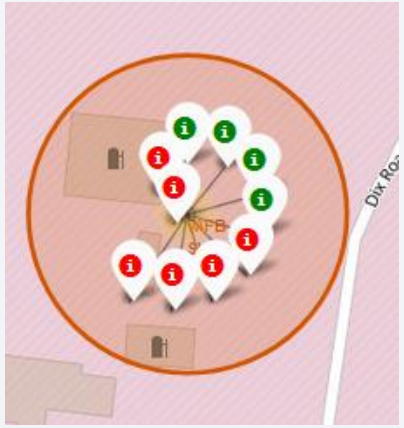
Folium Map - Markers





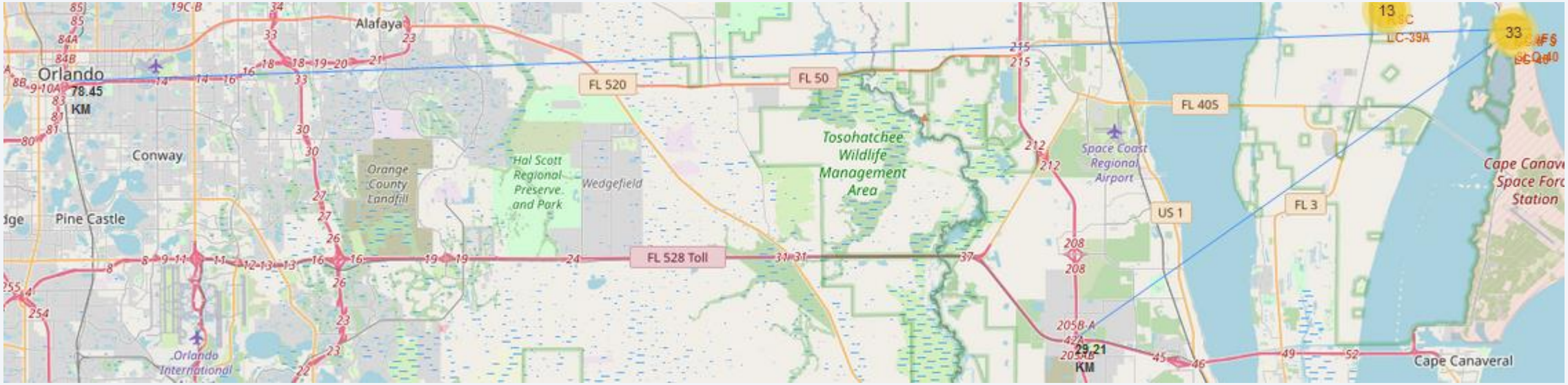
Florida Launch Sites

Green marker indicates successful launches, whereas
Red marker indicated unsuccessful launches.

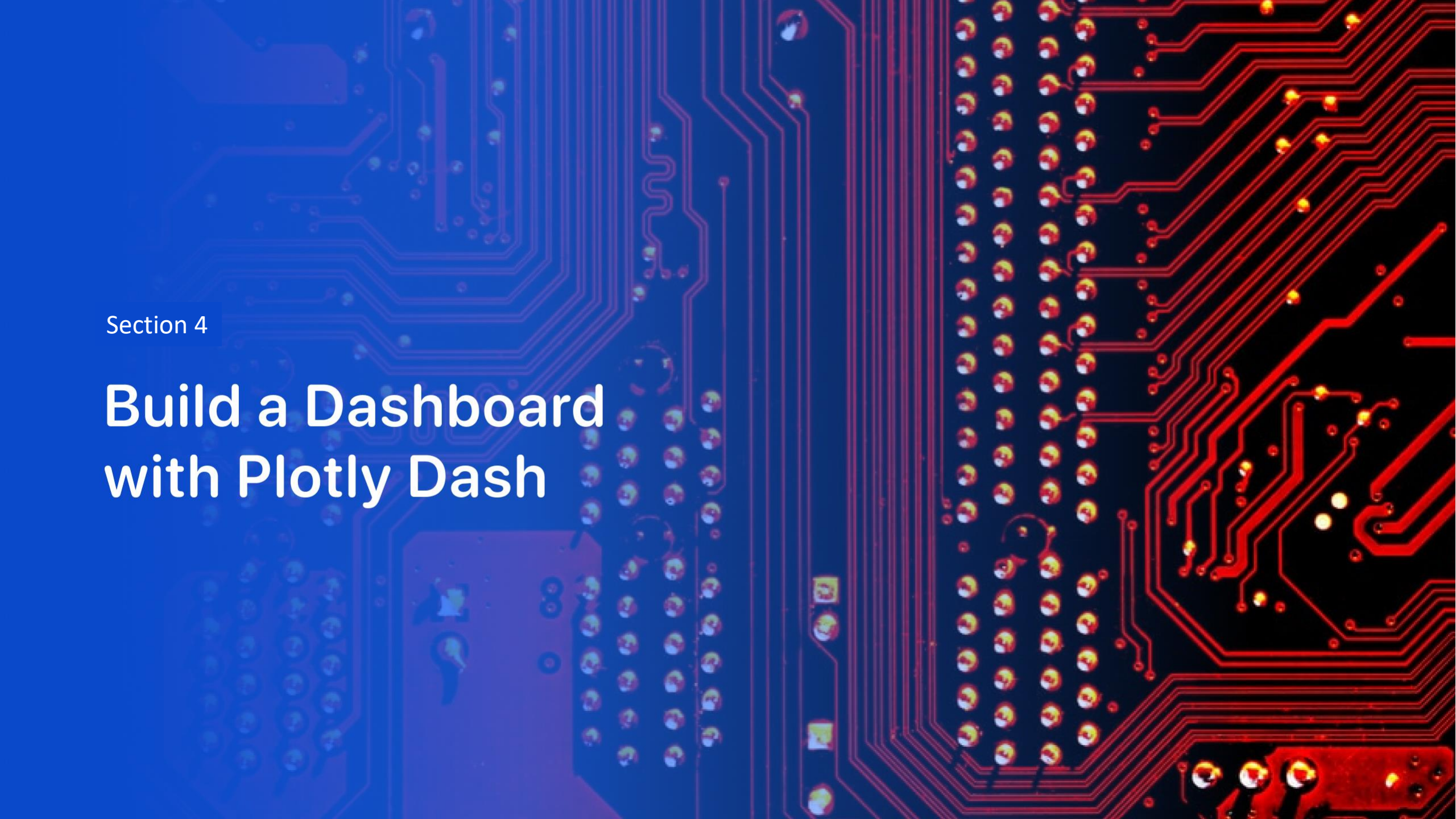


California Launch Site

Folium Map - Distances between Launch Sites and landmarks



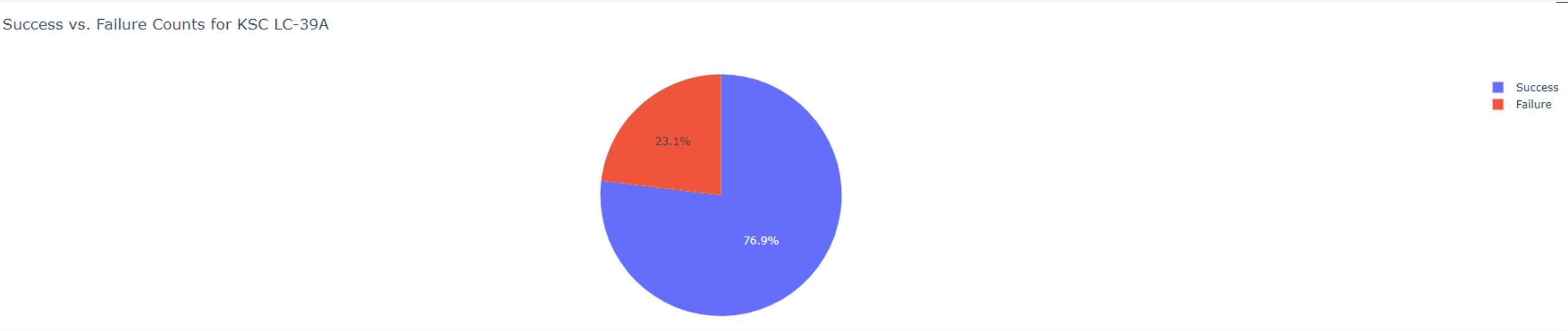
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Are launch sites in close proximity to the cities? No


Section 4
Build a Dashboard with Plotly Dash

Dashboard – Success percentage for each launch site

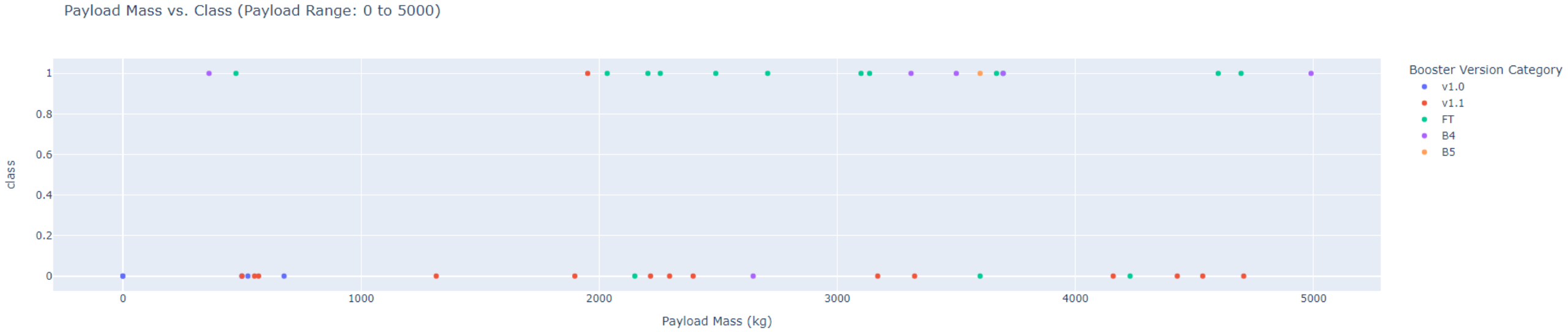
- From this chart, we can see that KSC LC-39A has the best success rate of launches, as compared to other launch sites

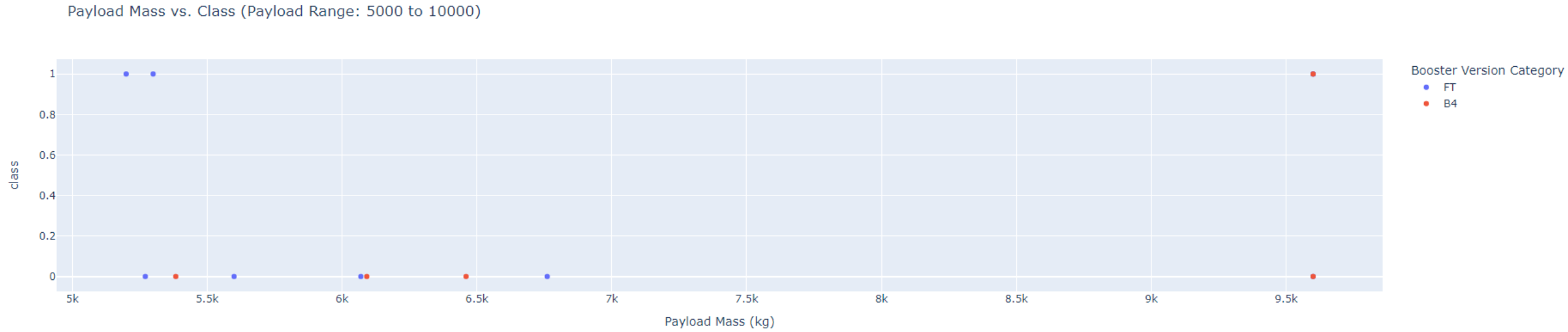
Dashboard – Total success launches for KSC LC-39A



- KSC LC-39A Launch Site achieved 76.9% success launces, and 23.1% failed launches.

Dashboard - Payload vs Launch Outcome





- Low weighted payloads (upper Figure) have a better success rate than the heavy weighted payloads (lower Figure).

Section 5

Predictive Analysis (Classification)

Classification Accuracy

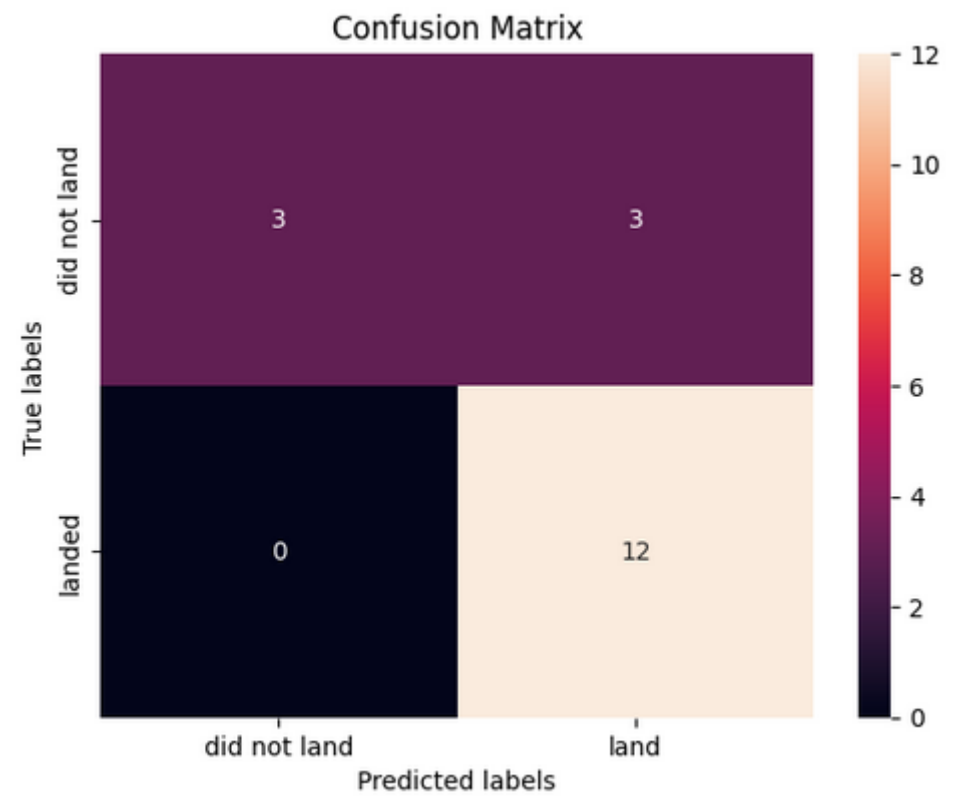
Find the method performs best:

```
print("Accuracy of the logistic regression is: " + str(l) + " %.")
print("Accuracy of the SVM is: " + str(s) + " %.")
print("Accuracy of the Decision Tree is: " + str(t) + " %.")
print("Accuracy of the K-Nearest Neighbor is: " + str(k) + " %.")
```

```
Accuracy of the logistic regression is: 0.8333333333333334 %.
Accuracy of the SVM is: 0.8333333333333334 %.
Accuracy of the Decision Tree is: 0.7222222222222222 %.
Accuracy of the K-Nearest Neighbor is: 0.8333333333333334 %.
```

- In my research, all of the used machine learning models have achieved the same test accuracy of 83.33%, except the Decision Tree algorithm, which achieved 72.22%

Confusion Matrix



- Since the test accuracy for three ML methods are identical, one confusion matrix is presented
- We can see that our ML model has:
  - 12 True Positive instances
  - 3 True Negative instances
  - 3 False Positive instances
  - 0 False Negative instances
- Further goal should be reducing of the false positive cases

Conclusions

- Payload mass is a crucial factor influencing mission success, with low-weighted payloads (defined as 4000kg and below) consistently performing better than heavier payloads.
- Notably, from 2013 onward, the success rate of SpaceX launches has demonstrated a consistent increase, indicating a positive trend that is expected to further enhance mission success in the future.
- Among launch sites, **KSC LC-39A** stands out with the highest success rate of 76.9%.
- The success rates for specific orbits vary, with **GEO, HEO, SSO, and ES L1** exhibiting the best performance.
- The dataset suggests a correlation between launch success and factors such as the launch site, orbit, and the number of previous launches. This correlation may be indicative of an evolving knowledge base contributing to mission success.
- Although the data doesn't currently explain the disparities between launch sites, further exploration, possibly including atmospheric or other relevant data, could provide more insights.
- Additional observations include the geographical advantage of launch sites near the equator, the proximity of all launch sites to the coast, and the positive correlation between launch success and increasing payload mass across all launch sites.

Thank you!