IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

By: Tewebo M. Teshome
On Feb 12, 2024

Presentation Outline

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

Executive Summary

- The SpaceX data is scraped from the web and parsed using Beautifulsoup
- The pandas liberary is utilized for dara wrangling
- Seaborn, matplotlib and SQL query are used for exploratory data analysis
- Folium is utilized for visualizing geographical data
- Plotly is utilzed for interactive dash boards
- Logistic regression, KNN, decision tree classifier and SVM are used to model the data
- The KNN model perform better than the others

Introduction

- Project background and context
- SpaceX Falcon 9 rocket launches with a cost of 62 million while other providers cost upward of 165 million dollars
- Much of the savings is because of reuse the first stage.
- A new SpaceY company to be established wants to use SpaceX data for its upcoming rocket project
- Problems you want to find answers
- The main aim of this data analysis is to predict the success of SpaceX Falcon 9 which will be the baseline for the SpaceY.

Section 1

Methodology

Methodology

Executive Summary

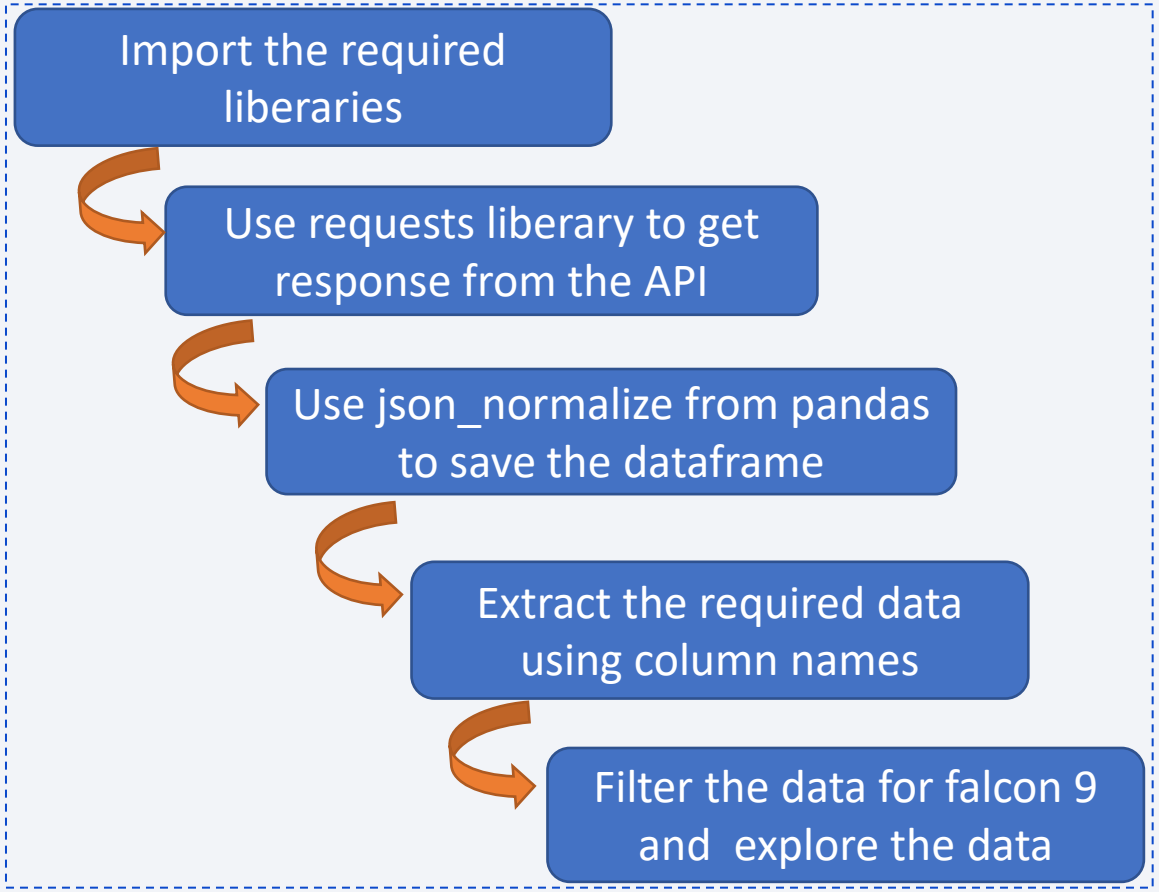
- Data collection methodology:
  - The SpaceX data is collected from the web scraped from the web using requests
- Perform data wrangling
  - The data wrangling is carried out using pandas
- EDA and using visualizations are carried out using SQL and seaborn
- The interactive dashboards are prepared using Plotly Dash
- Perform predictive analysis using classification models
  - The available data is used for training and testing
  - Different models are built and the model accuracy is evaluated

Data Collection

- The historical data of SpaceX falcon 9 are available from SpaceX API
- Alternatively the same data is accessed from wikipedia website
- The html file of the SpaceX data from the web is accessed using requests liberary and parsed using the other python liberary BeautifulSoup
- The pandas liberary, with some other supporting python function, is used to explore the falcon9 dataframe
- the data is filterred by column heads and unneccessary tables are excluded from the data
- Some missing values were noticed on ‘PayloadMass’ and ‘landingPad’
- These missed values were filled with the average value of the available data

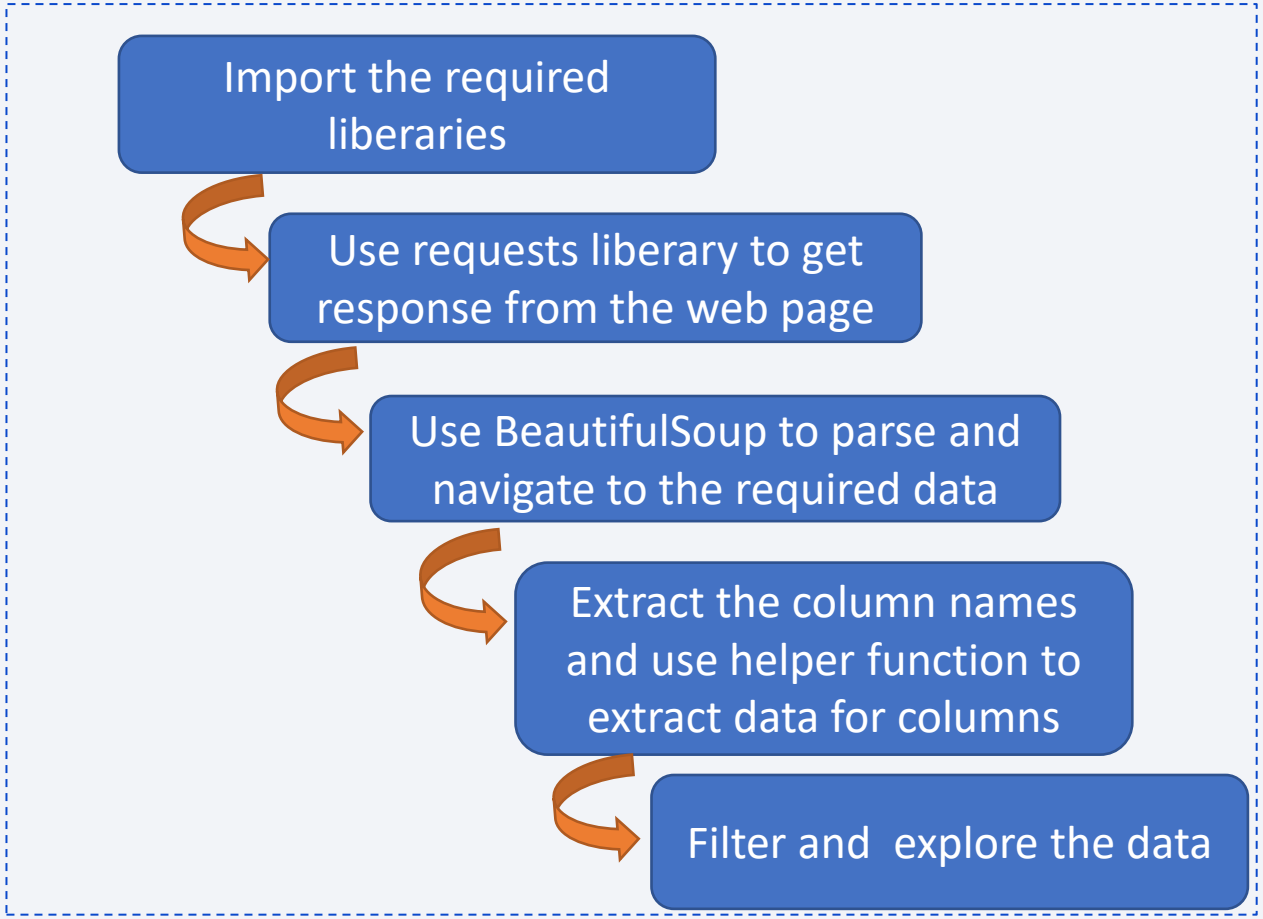
Data Collection – SpaceX API

- Data collection flowchart using SpaceX API is summarized next right here
- The completed note book for accessing data using SpaceX API is available here:
- https://github.com/TeweboM/IBM-capstone-project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- Data collection flowchart from wikitable is summarized next here
- The completed note book for accessing data from wikipedia webpage is available here:
- https://github.com/TeweboM/IBM-capstone-project/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

The data wrangling is carried out using pandas liberary

The flow chart includes:

1. import the required liberaries (pandas and numpy)
2. read the .csv file from SkillsNetwork at coursera using pandas
3. check for missed values using isnull() method
4. check the data types of columns using dtypes attribute

The completed note book is uploaded at https://github.com/TeweboM/IBM-capstone-project/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- As part of the exploratory data analysis and to visualize the relationship between features data visualization is used
- Using seaborn liberary different visualizations are made
- The number of launchs from each launch site is visualized using barcharts
- The relation between launch site, payload mass and class; orbit and class; orbit and success rate are visualized using scatter plots
- The relation between flight number, payloads and class; launch site, payloads and class were ploted using catplots
- The complete note book is available here at github https://github.com/TeweboM/IBM-capstone-project/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

Some of the lists of sql quieries performed are:

- %sql create table SPACEXTABLE as select * from SPACEXTBL where Date is not null
- %sql SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE
- %sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
- %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer LIKE '%NASA (CRS)%‘
- %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%‘
- %sql SELECT * FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%Success%' ORDER BY Date LIMIT 1

The complete noted book is available at github https://github.com/TeweboM/IBM-capstone-project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Using folium liberary location visualizations are explored
- All launching sites are explored on map
- Markers and circles are used to specifically indicate the location of the launch sites
- Nearby cities and facilities are investigated on map and the distance from launch sites explored
- Distances from nearby cities and facilities are calculated in km
- The completed note book is uploaded at github https://github.com/TeweboM/IBM-capstone-project/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- The dash board was prepared to include pie charts which will be updated based on the launch site selection
- If the user selects all sites the plot will show proportions of each launch site
- If the user selects specific launch site the pie chart will display the success rate
- Dropdowns and Range Slider are availed for user interactive input
- Scatter graph for launch site and payload mass based on user selection
- The complete python code for running the dashboard is at github https://github.com/TeweboM/IBM-capstone-project/blob/main/interactive_dashboard.py

Predictive Analysis (Classification)

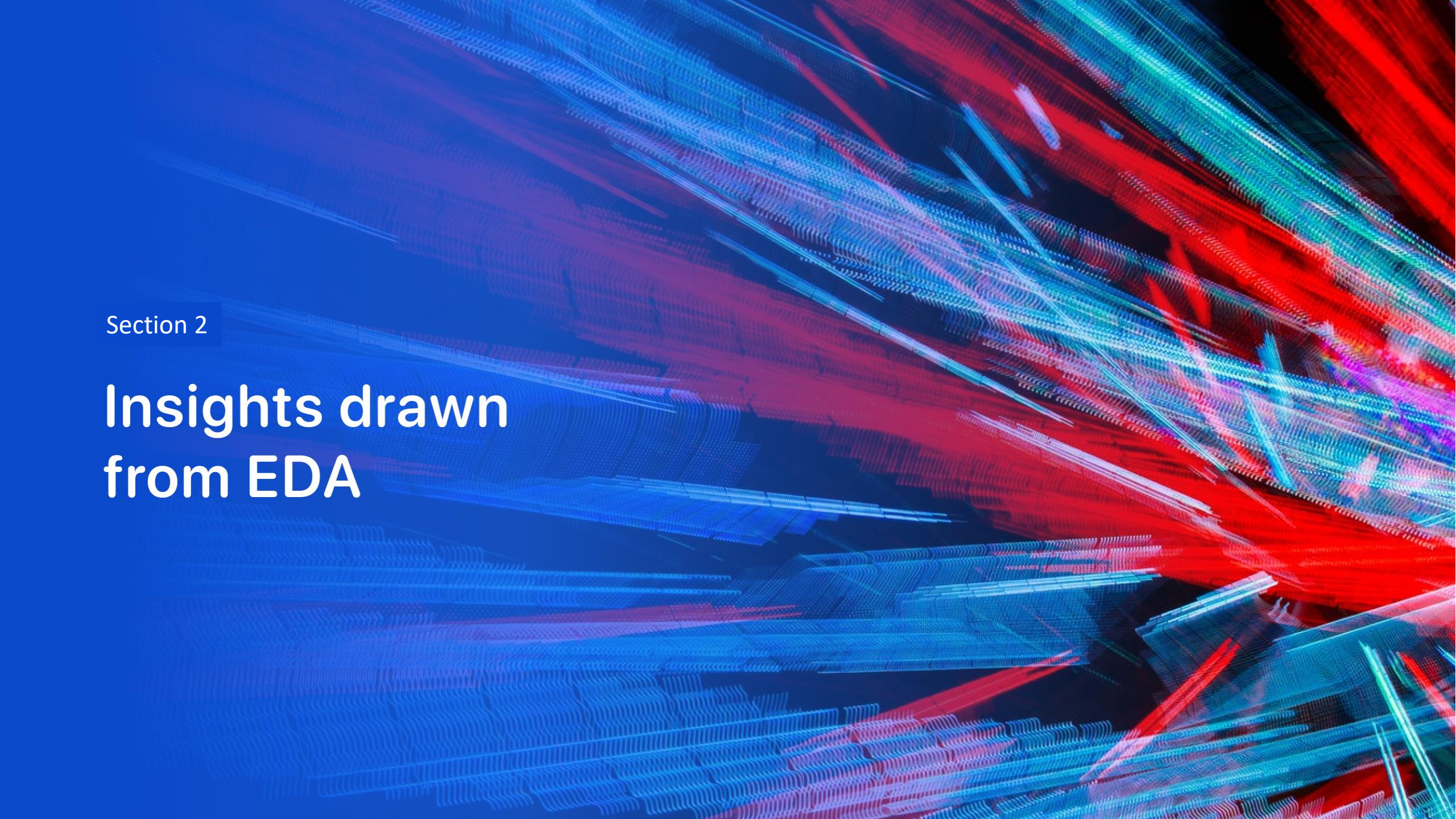
- The success of Falcon9 is predicted utilizing the available data and different models
- The available data is standardized, transformed and split into two for training and testing the model
- The different models developed are logistic regression, support vector machine, decision tree cassifier and KNearest neighbors
- The accuracy score for each model is evaluated and confusion matrices drawn
- The completed notebook uploaded at https://github.com/TeweboM/IBM-capstone-project/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

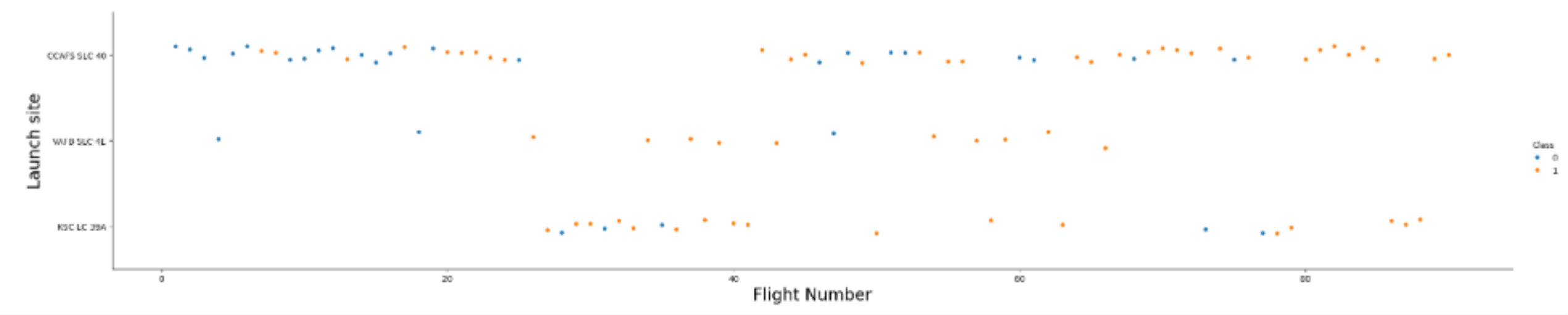
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

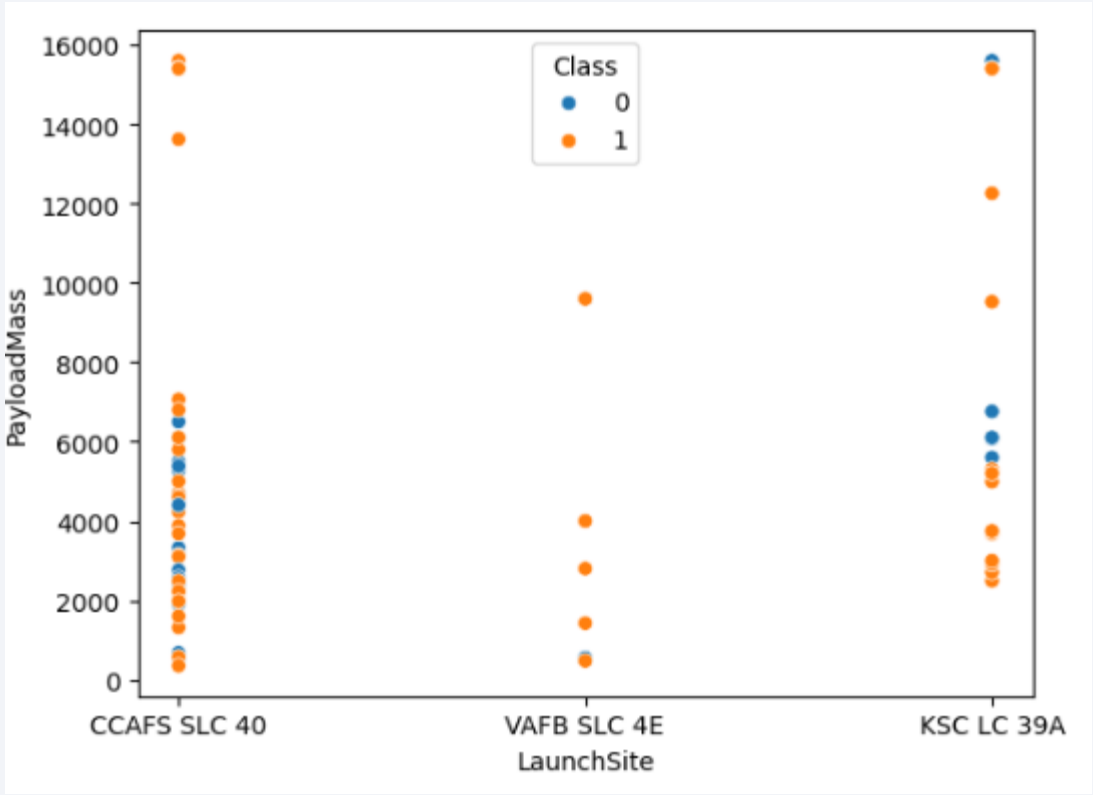


Flight Number vs. Launch Site



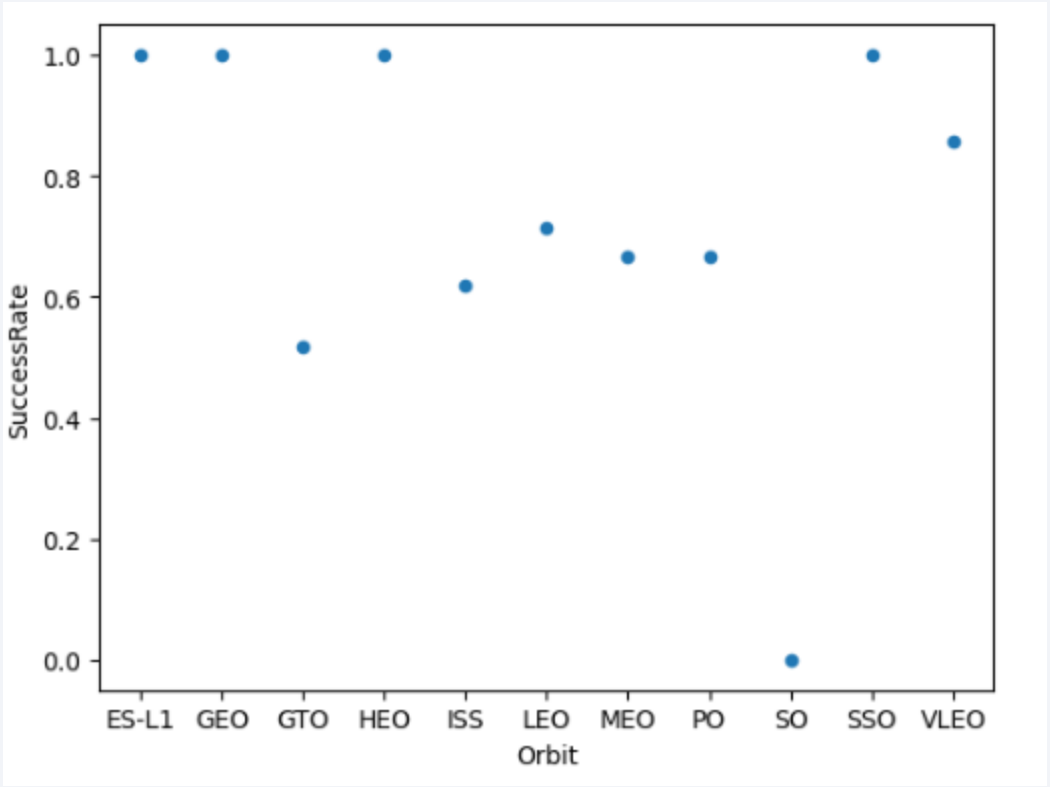
- It is noted that the earliest flights were from launch site CCAFS SLC 40
- The flight numbers has been shifted to VAFB SLC 4E which has been re-shifted later

Payload vs. Launch Site



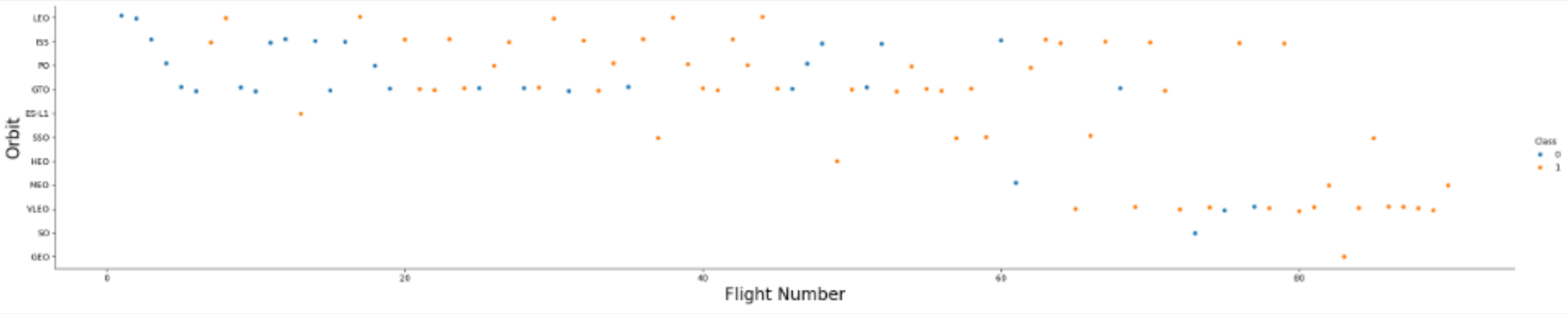
- Launch site CCAFS SLC 40 is used for light and heavy payloads
- Whereas launch site KSC LC 39A is used for medium payloads

Success Rate vs. Orbit Type



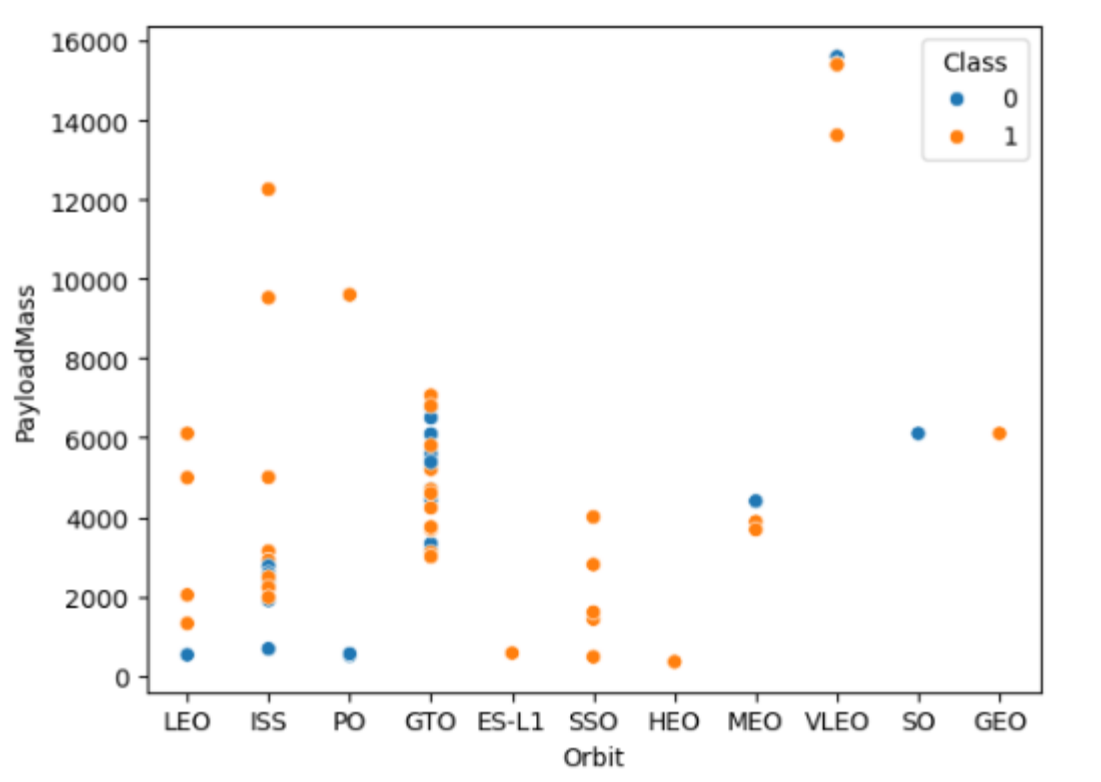
- Some of the orbits, ES-L1, GEO, HEO and SSO were successful at 100%.
- One of the orbits, SO, is the most unsuccessful at 0% success rate.

Flight Number vs. Orbit Type



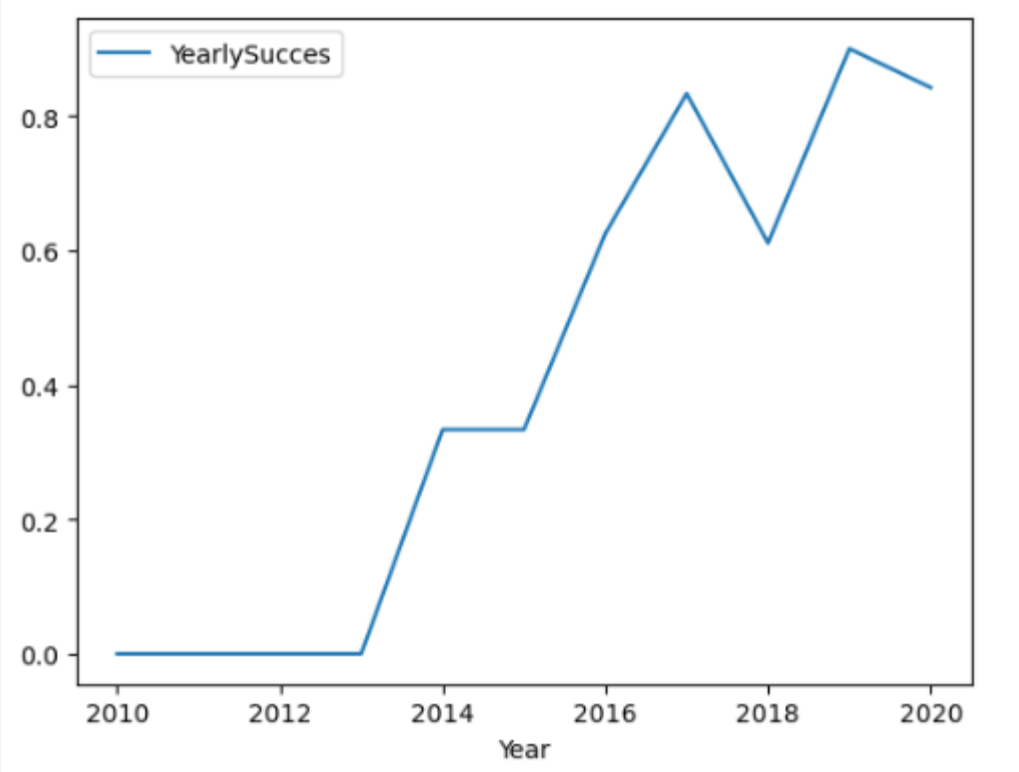
- The oldest orbits are LEO, ISS, PO and GFO
- The recent orbits are GEO, SO, VLEO etc.

Payload vs. Orbit Type



- The orbit VLEO is used for heavy pay load mass
- The orbit HEO, SSO, ES-L1 are used for less heavy pay load mass

Launch Success Yearly Trend



- The success rate starting from 2013 has been increasing
- The success rates beyond 2016 are beyond 60%

All Launch Site Names

The names of unique launch sites are extracted using sql query

“%sql SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE”

The results are:

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

Launch Site Names Begin with 'CCA'

The query is made on sql

“%sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5”

- The wild card ‘%’ and the key word ‘LIKE’ are jointly utilized
- Which extracts the first five row of the launch sites that start with ‘CCA’

Total Payload Mass

- The aggregate function sum() and the key word ‘Like’ with the wild card ‘%’ are used to query the values
- The sql query used is
- %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer LIKE '%NASA (CRS)%‘
- The query result is 48,213Kg

Average Payload Mass by F9 v1.1

Sql Query

- %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%‘

Query result

2534.6666666666665

First Successful Ground Landing Date

SQL query

- %sql SELECT Date FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%Success%' ORDER BY Date LIMIT 1

Query result

- 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL query

- %sql SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
- Query result next here to the right

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

Total Number of Successful and Failure Mission Outcomes

SQL query

- %sql SELECT COUNT(*) FROM SPACEXTABLE WHERE Mission_Outcome LIKE '%Success%'
- %sql SELECT COUNT(*) FROM SPACEXTABLE WHERE Mission_Outcome LIKE '%Failure%'

Results

100

1

Boosters Carried Maximum Payload

Booster_Versionz
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

2015 Launch Records

SQL query

- %sql SELECT Date, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Fail%' and Date Like '2015%'
- Result

Date	Landing_Outcome	Booster_Version	Launch_Site
2015-01-10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

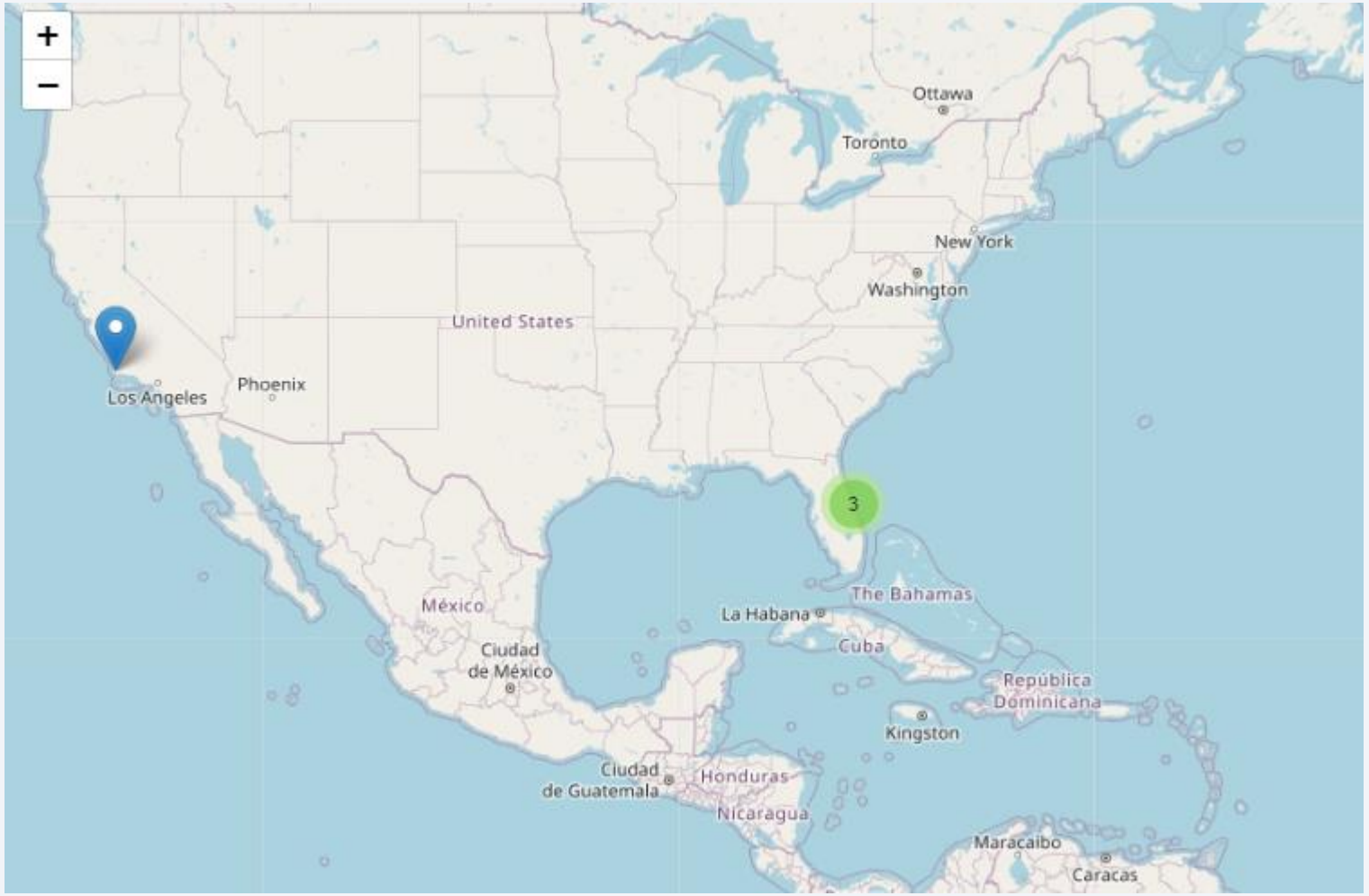
- SQL query
- %sql SELECT COUNT(Landing_Outcome) as Total_Outcomes FROM SPACEXTABLE WHERE substr(Date,0,5) > '2010-06-04' and substr(Date,0,5) < '2017-03-20‘
- Result

Total_Outcomes
44

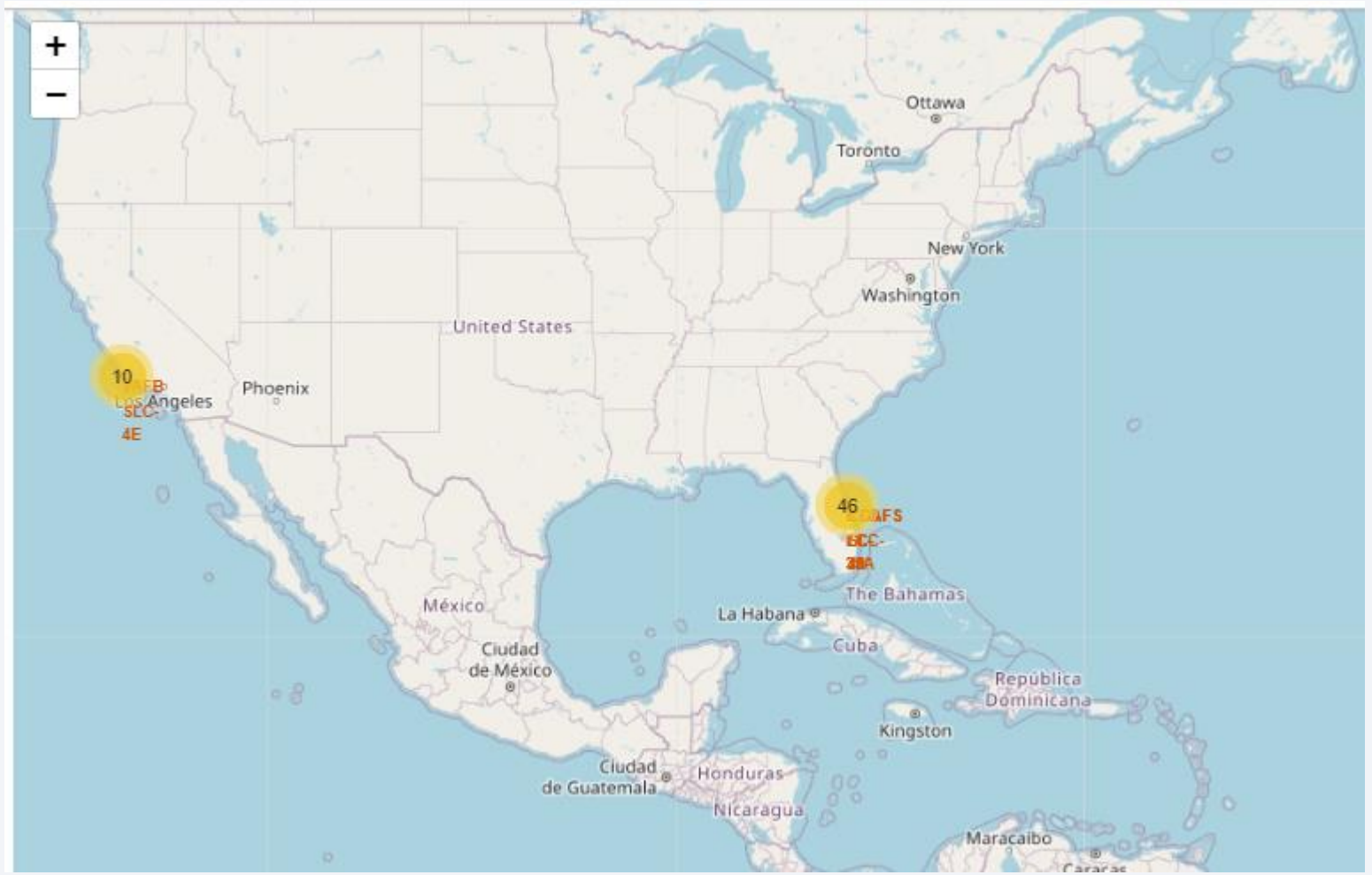

Section 3
Launch Sites
Proximities Analysis

Launch sites on shown on map

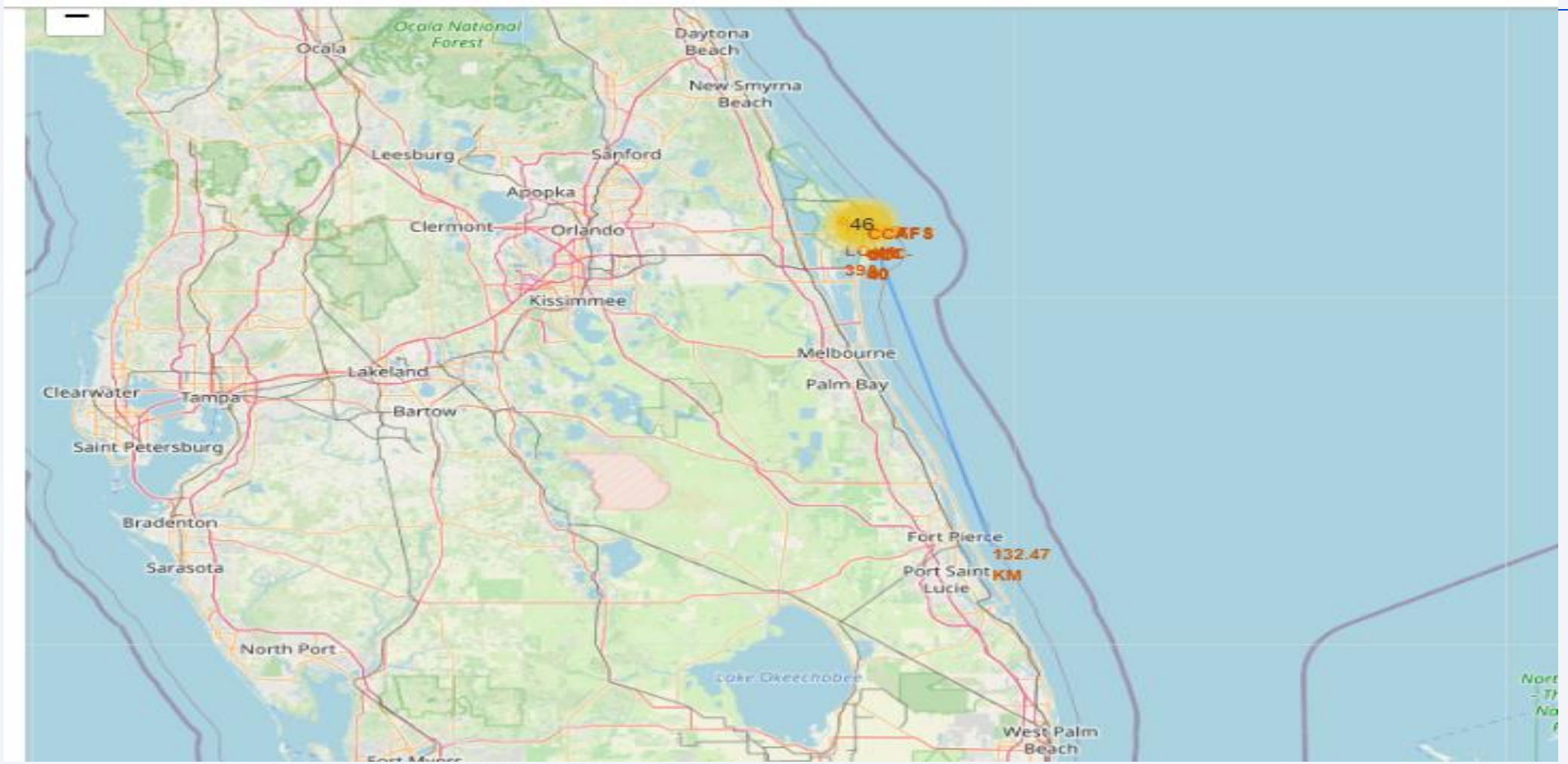
- One of the sites is locates at west coast and three of the costs are adjacent to each other

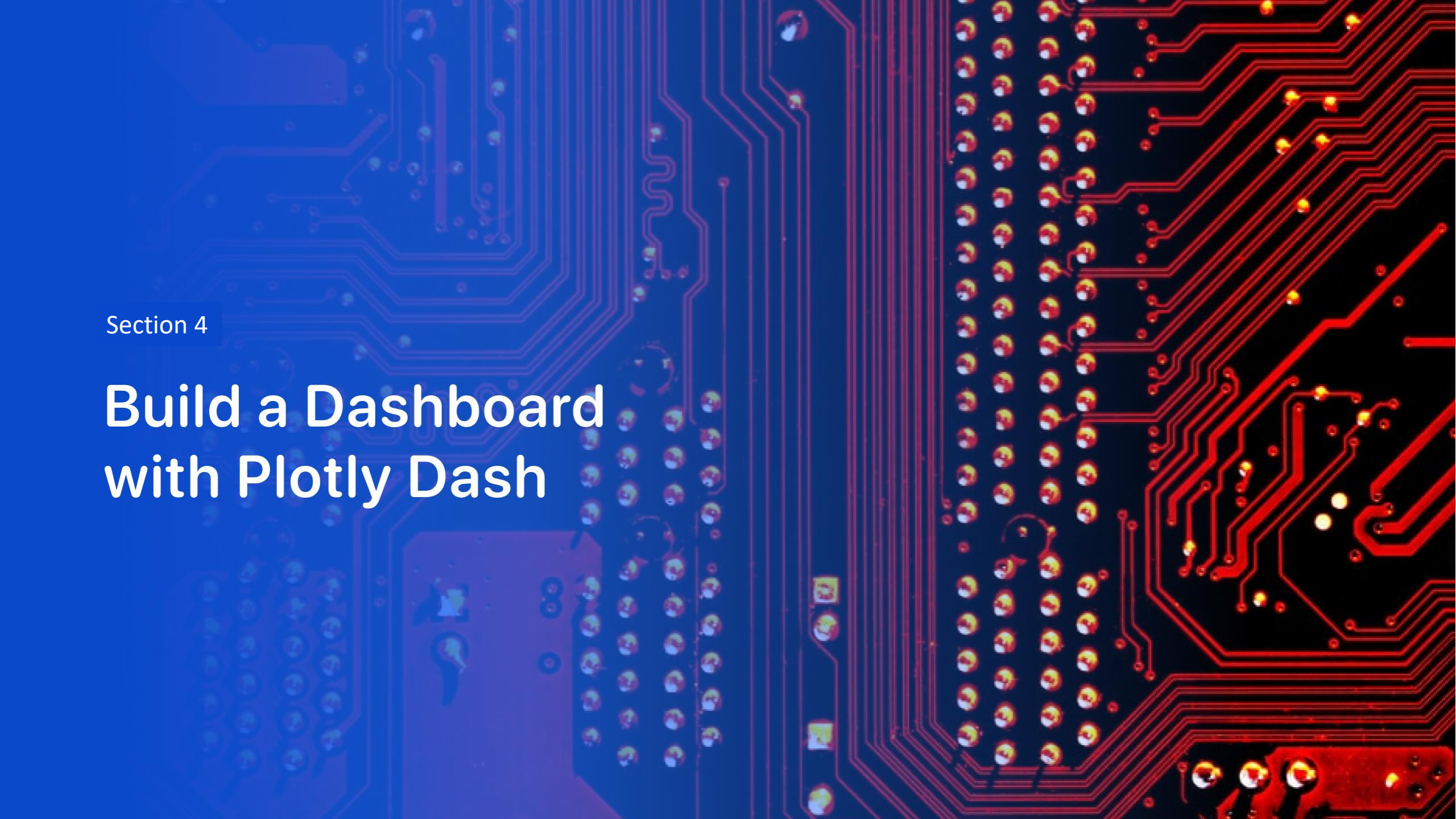


Launches from each site

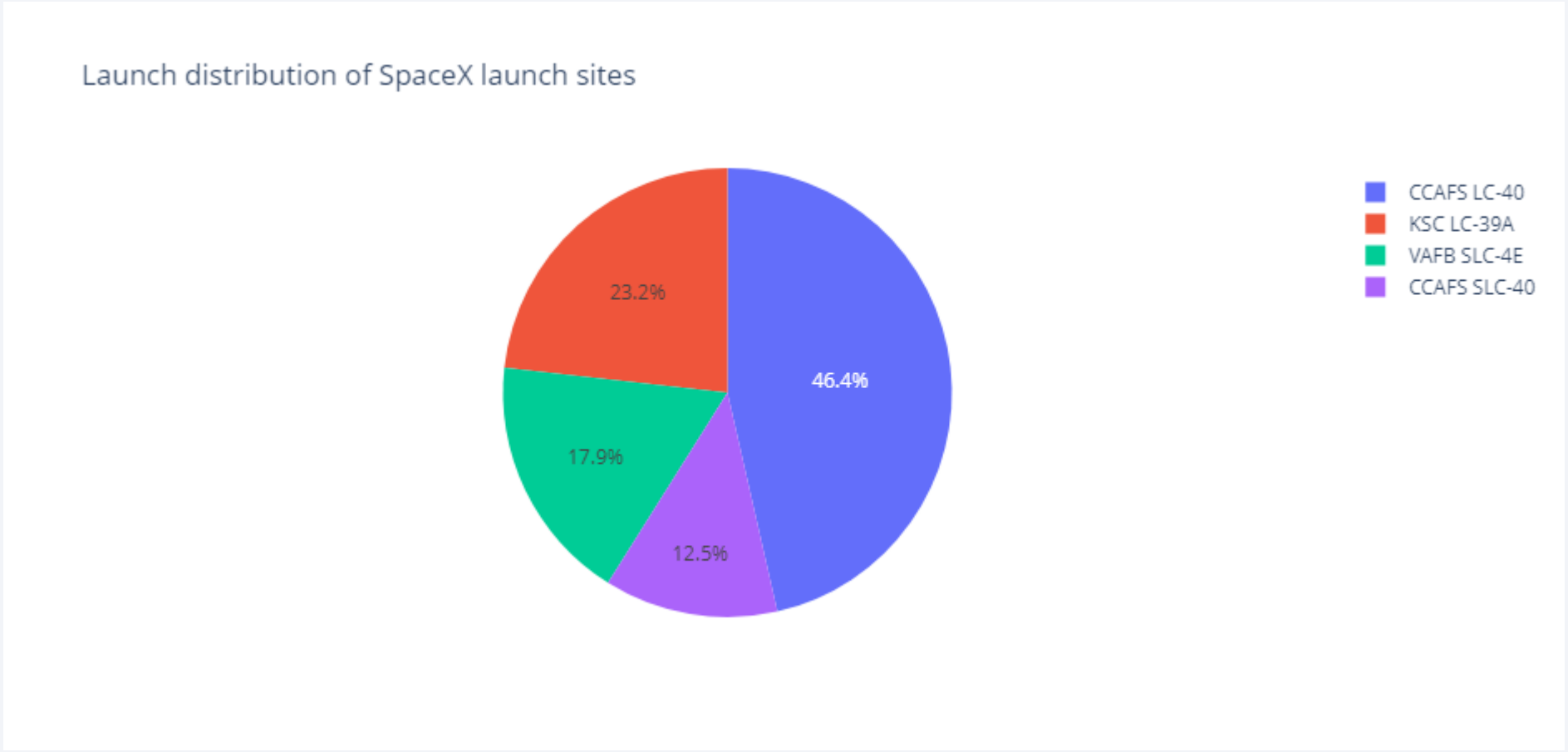


Launch site 1 with proximities

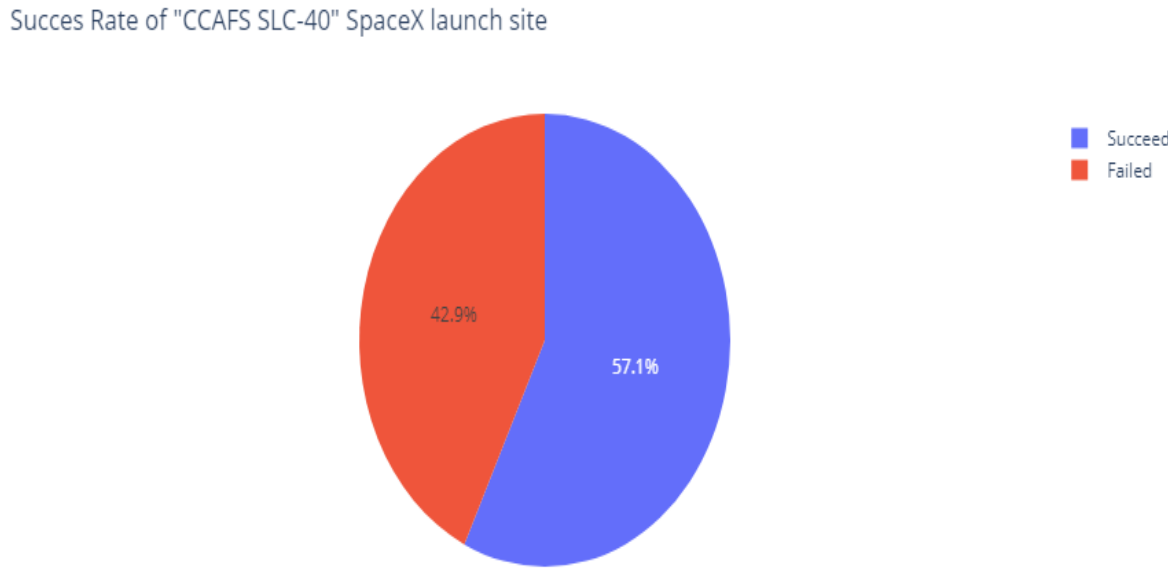


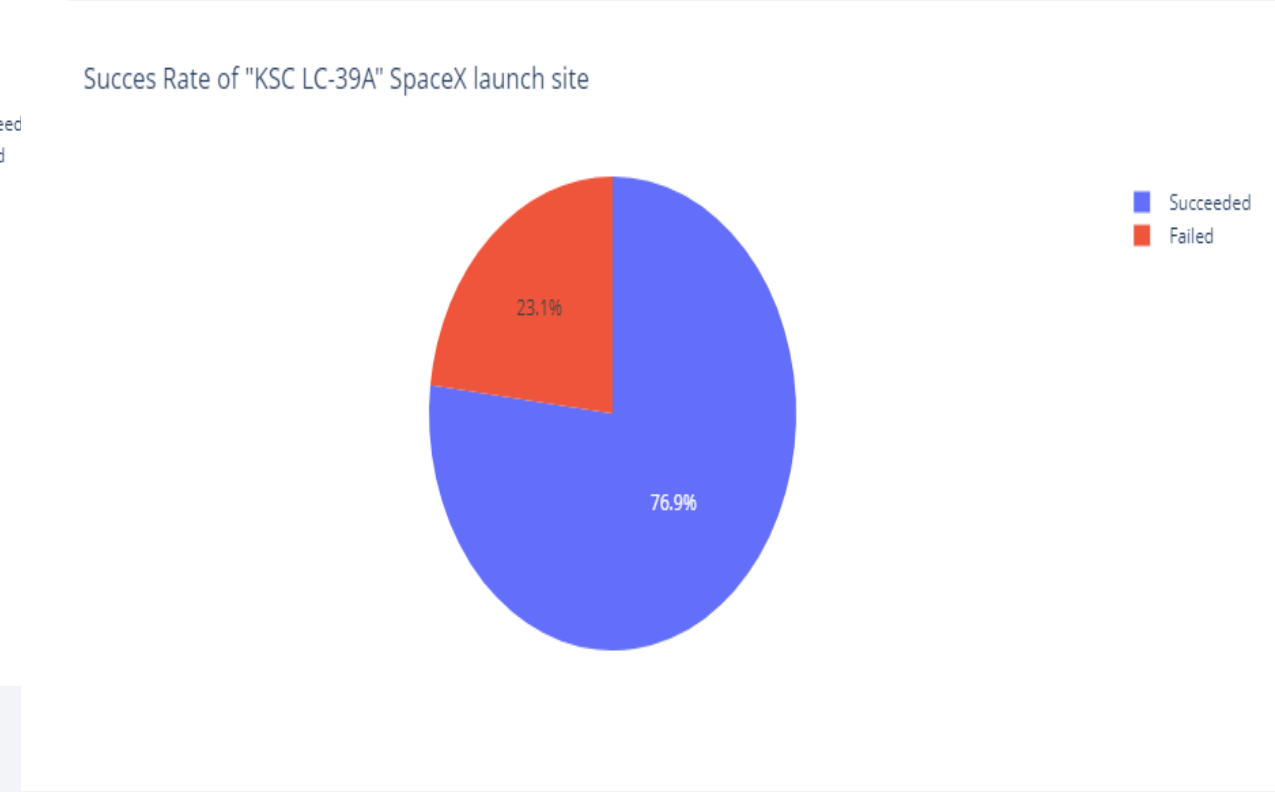

Section 4
Build a Dashboard with Plotly Dash

Launch frequencies from different launch sites

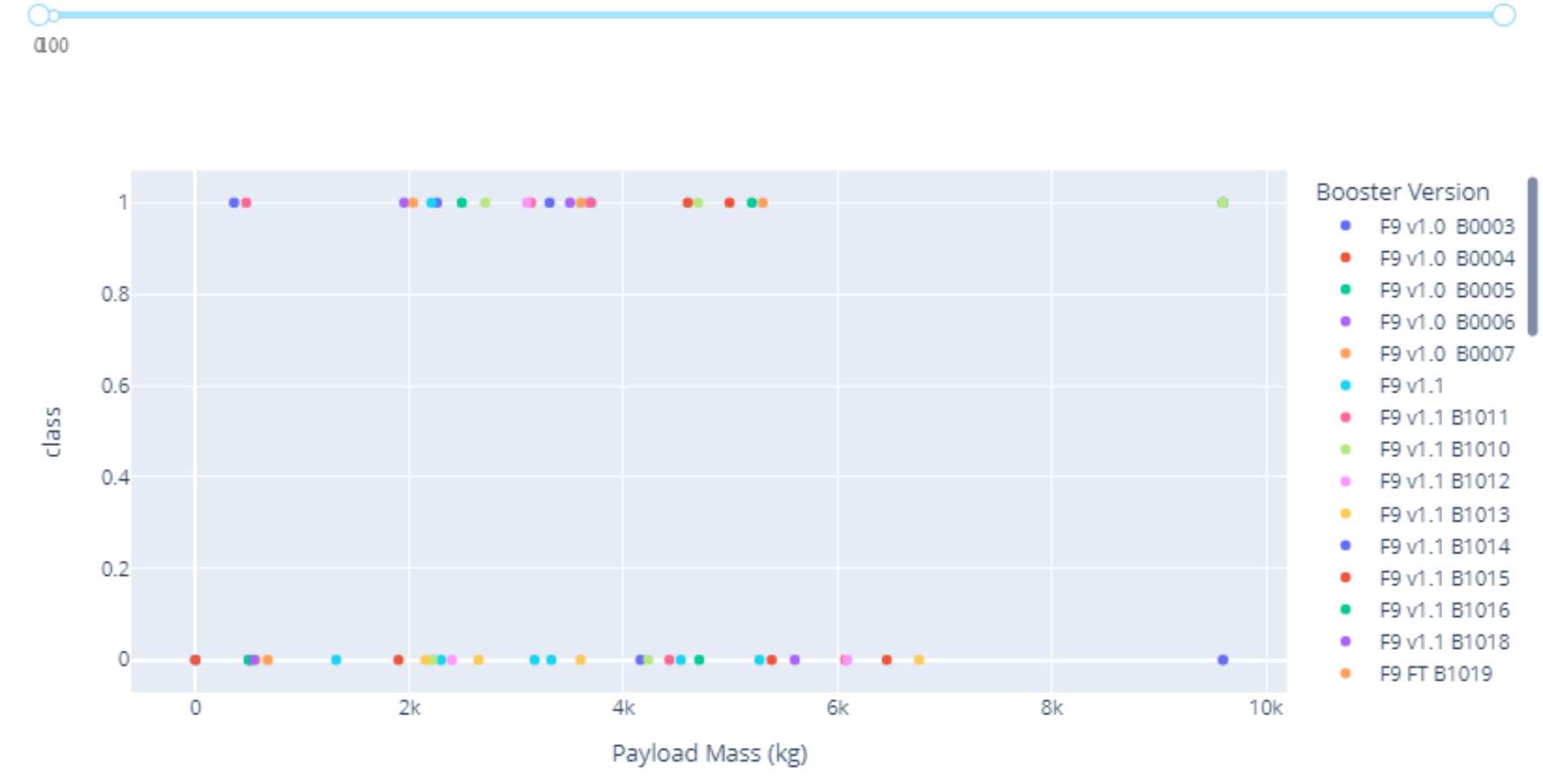


Success ratio from launch sites





Payload vs. Launch Outcome scatter plot

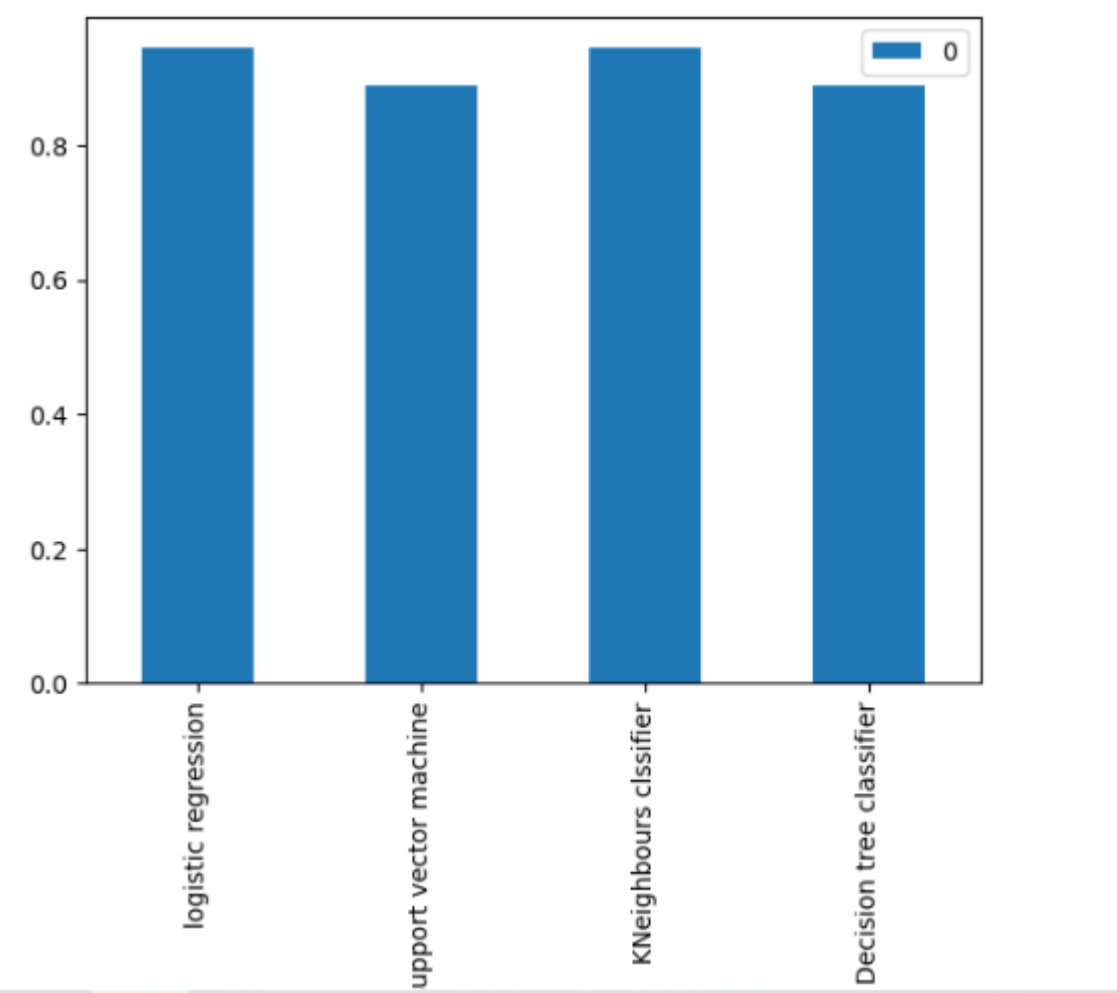


Section 5

Predictive Analysis (Classification)

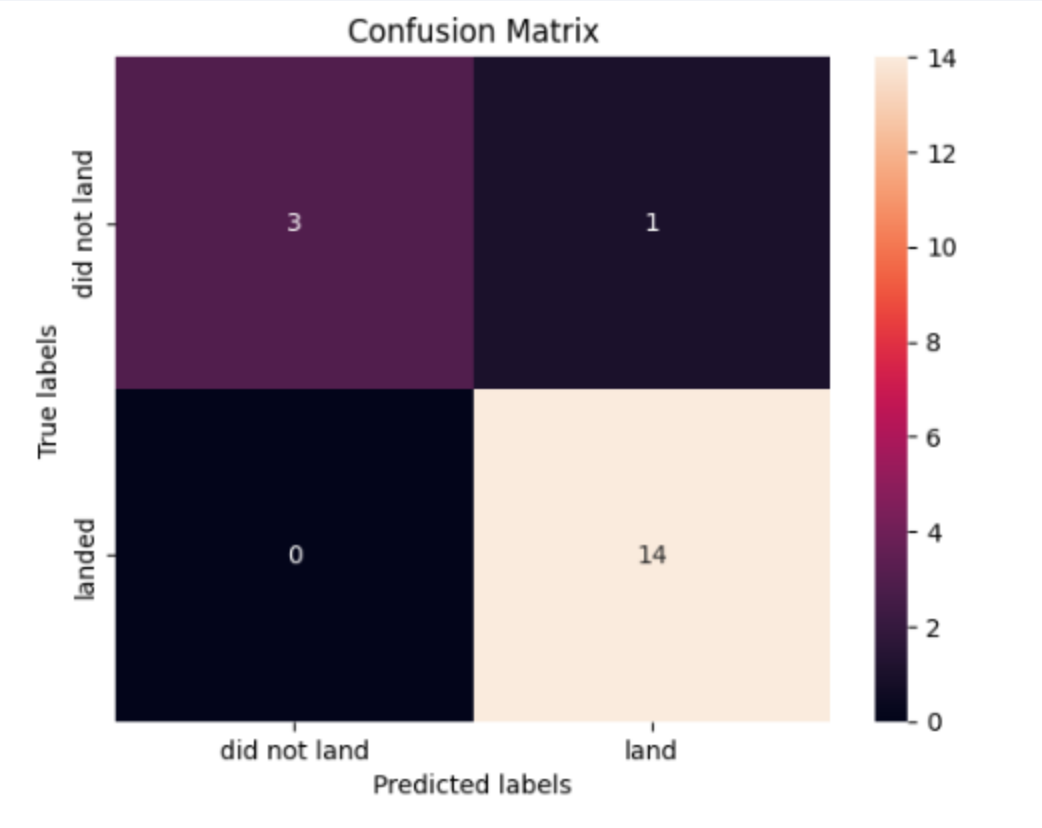
Classification Accuracy

- The model accuracy of the four models is shown here on the right
- Logistic regression and Knearest neighbour classifier performs better



Confusion Matrix

- The confusion matrices predicted by logistic regression and KNN are similar and shown to the right here

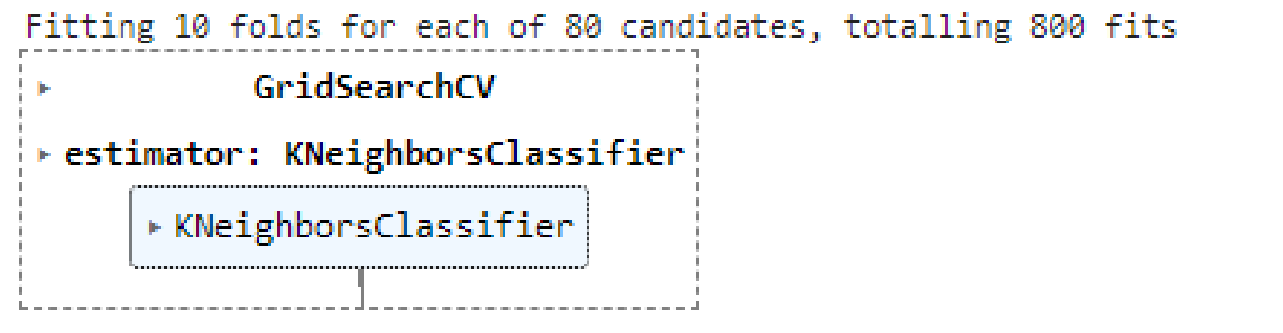


Conclusions

- The SpaceX historical data collected from SpaceX API and/or wikipedia table is well organized data and could be used for data analysis
- The success rate of SpaceX launches has gone increasing from year to year
- Logistic regression and KNN can be used to better predict the success of a falcon9 launch
- As an outcome of analysis of the SpaceX data the success rate of a falcon9 rocket is 94%

Appendix

- KNN model



Thank you!