IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jananjoy Rajkumar
13 December, 2021

Outline on the Presentation

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

Executive Summary

- Summary of methodologies
  - Collect data using REST API and results are viewed by calling json() method and perform data wrangling for Exploratory Data Analysis (EDA) .
  - Perform EDA loading the Spacex dataset into a Db2 database and visualize the attributes relationship using the Matplotlib library.
  - Build machine learning pipeline (Predictive analysis) for Falcon 9 lands consisting of Preprocessing allowing to standardize our data.

- Summary of all results
  - Perform EDA with visualization understand the relationship patterns, launch site information, success rate of the orbit, Launch success yearly trend.
  - EDA with SQL provides information about launch sites, payloads information , successful landing outcome, name of the boosters, success and failure outcome, booster version, failed landing outcome and count of landing outcomes.
  - Interactive map with Folium mark all launch sites, success/failed launches for each site on a map. Besides, calculate the distances between a launch site to its proximities.
  - Predictive analysis find the best Hyperparameters for SVM, Classification Tress and Logistics Regression.

Introduction

- **Project background**
  - Commercial companies are working on many space projects. Such type of projects are expensive because of the complexity and infrastructures. However, Spacex is one of the successful project in this domain.

- **Context**
  - Many commercial companies are interested to execute on space projects. However, efficient optimization is an important aspect because of the infrastructures, complexity, and cost. We need to perform systematic analysis to understand the root cause of the problem

- **Problems you want to find answers**
  - Want to build a machine learning model using the available spacex dataset for optimizing the system. Taking the role of data scientist, I want to gather the data from spacex and create a dashboard for the team members which helps to predict the reusability of the resources.

Section 1

Methodology

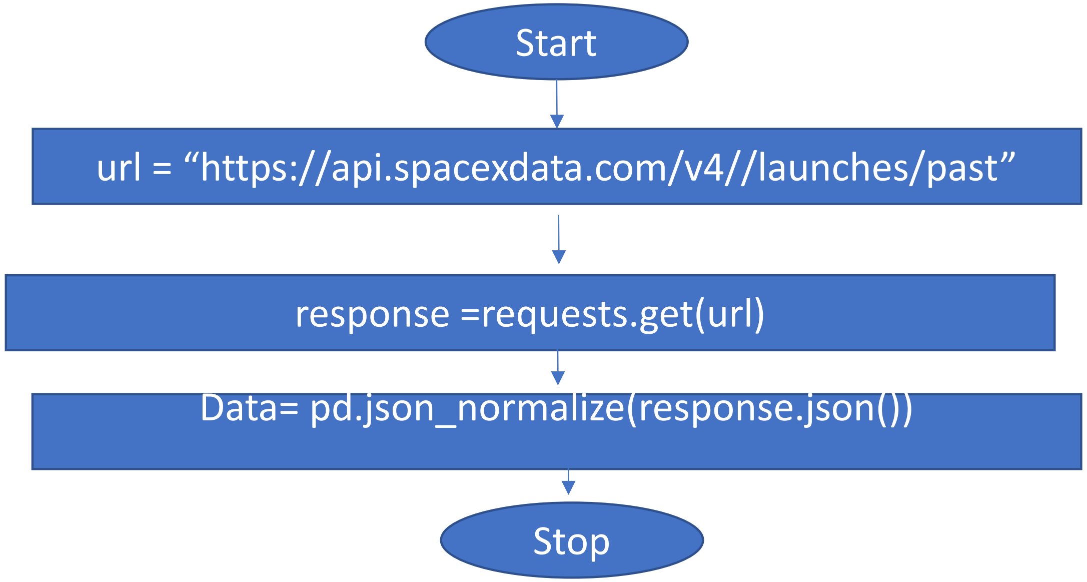
Methodology

Executive Summary

- Data collection methodology:
  - Data are collected using the REST API with exposed endpoints
  - Use the Spacex dataset in .CSV loading to the DB2 databases
- Perform data wrangling
  - Use attributes specific to the exploratory data analysis (EDA) requirements
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Use SQL statement
- Perform interactive visual analytics using Folium and Plotly Dash
  - Use Matplotlib library
- Perform predictive analysis using classification models
  - Build machine learning pipeline (Predictive analysis) model

Data Collection

- Data are collected using the REST API with exposed endpoints.



Start

url = “https://api.spacexdata.com/v4//launches/past”

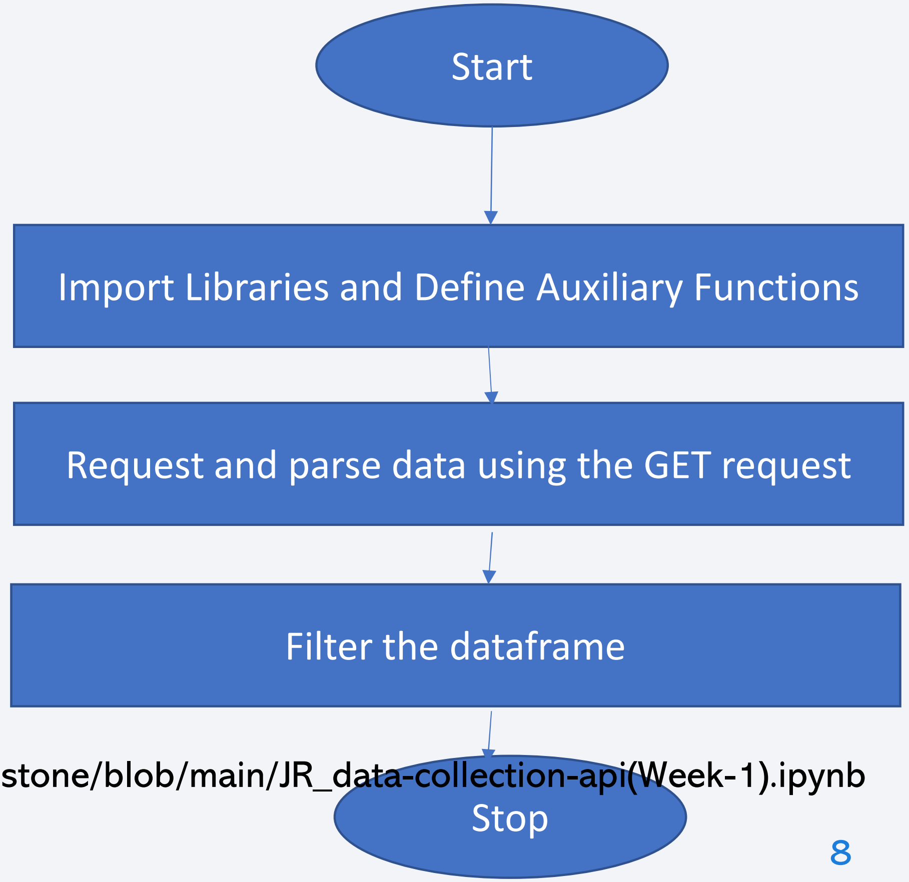
response =requests.get(url)

Data= pd.json_normalize(response.json())

Stop

Data Collection – SpaceX API

- Import the Libraries and define the function
- Request and Parse the data
- Filter the dataframe

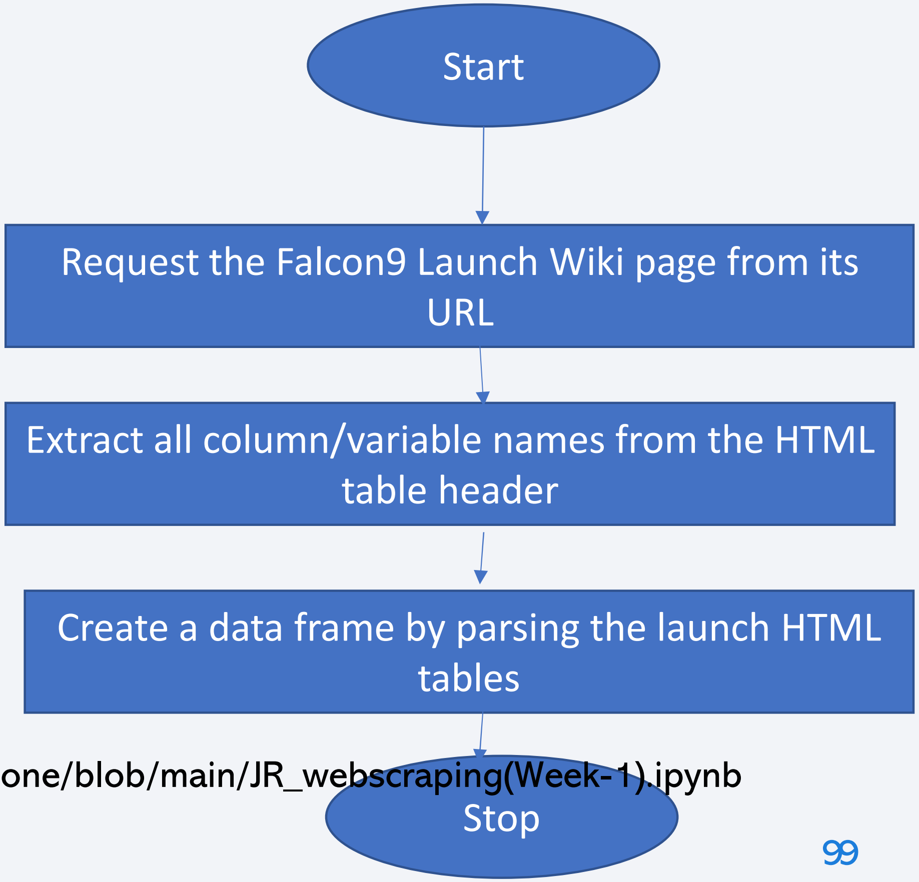


Start
Import Libraries and Define Auxiliary Functions
Request and parse data using the GET request
Filter the dataframe
Stop

- GitHub URL: https://github.com/jananjoy/Applied-data-science-capstone/blob/main/JR_data-collection-api(Week-1).ipynb

Data Collection - Scraping

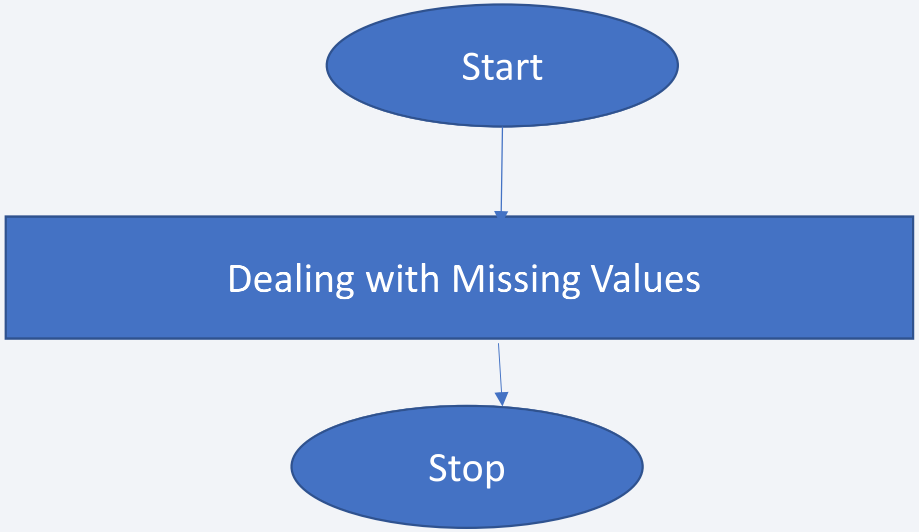
- Extract a Falcon 9 launch records HTML table from Wikipedia*
- Parse the table and convert it into a Pandas data frame


Start
Request the Falcon9 Launch Wiki page from its URL
Extract all column/variable names from the HTML table header
Create a data frame by parsing the launch HTML tables
Stop

- GitHub url: https://github.com/jananjoy/Applied-data-science-capstone/blob/main/JR_webscraping(Week-1).ipynb

Data Wrangling

- Wrangling data with API
- Sampling Data
- Dealing with Nulls



Start

Dealing with Missing Values

Stop

- GitHub URL:-https://github.com/jananjoy/Applied-data-science-capstone/blob/main/JR_data_wrangling(Week-1).ipynb

EDA with Data Visualization

- Perform exploratory Data Analysis and Feature Engineering using `Pandas` and `Matplotlib`
  - Exploratory Data Analysis
  - Preparing Data Feature Engineering
- Lists of the tasks perform with data visualization are as follows:
  - Visualize the relationship between Flight Number and Launch Site`
  - Visualize the relationship between Flight Number and Launch Site
  - Visualize the relationship between success rate of each orbit type
  - Visualize the relationship between Flight Number and Orbit type
  - Visualize the relationship between Payload and Orbit type
  - Visualize the launch success yearly trend
  - Create dummy variables to categorical columns
  - Cast all numeric columns to `float64`

- GitHub URL:- https://hub.gke2.mybinder.org/user/ipython-ipython-in-depth-r5j2z0la/notebooks/binder/Index.ipynb#

EDA with SQL

- 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 successful landing outcome in ground pad was acheived. Using bullet point format, summarize the SQL queries you performed
- 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. Use a subquery
- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for 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

- GitHub URL:-https://github.com/jananjoy/Applied-data-science-capstone/blob/main/My_EDA_with_SQL(Week-2).ipynb

Build an Interactive Map with Folium

- The generated map with marked launch sites which include Cceating `folium.Circle` and `folium.Marker` for each launch site on the site map.
- Mark the success/failed launches for each site on the map
- Calculate the distances between a launch site to its proximities
- Use `folium.Circle` to add a highlighted circle area with a text label on a specific coordinate and folium.Marker for plotting markers on a map

- GitHub URL :- https://github.com/jananjoy/Applied-data-science-capstone/blob/main/JR_Interactive%20Visual%20Analytics%20with%20Folium%20lab(Week-3).ipynb

Build a Dashboard with Plotly Dash

- Dashboard application contains input components such as a dropdown list and a range slider to interact with a pie chart and a scatter point chart. Lists of the tasks involved are :
  - Add a Launch Site Drop-down Input Component
  - Add a callback function to render success-pie-chart based on selected site dropdown
  - Add a Range Slider to Select Payload
  - Add a callback function to render the success-payload-scatter-chart scatter plot

Note:- Since my lab environment is not working. I kept the screeshot from the lab instruction.

At the same time, I refer my answer with reference to the mentioned screenshot. During the course I already executed the code.

- GitHub URL:

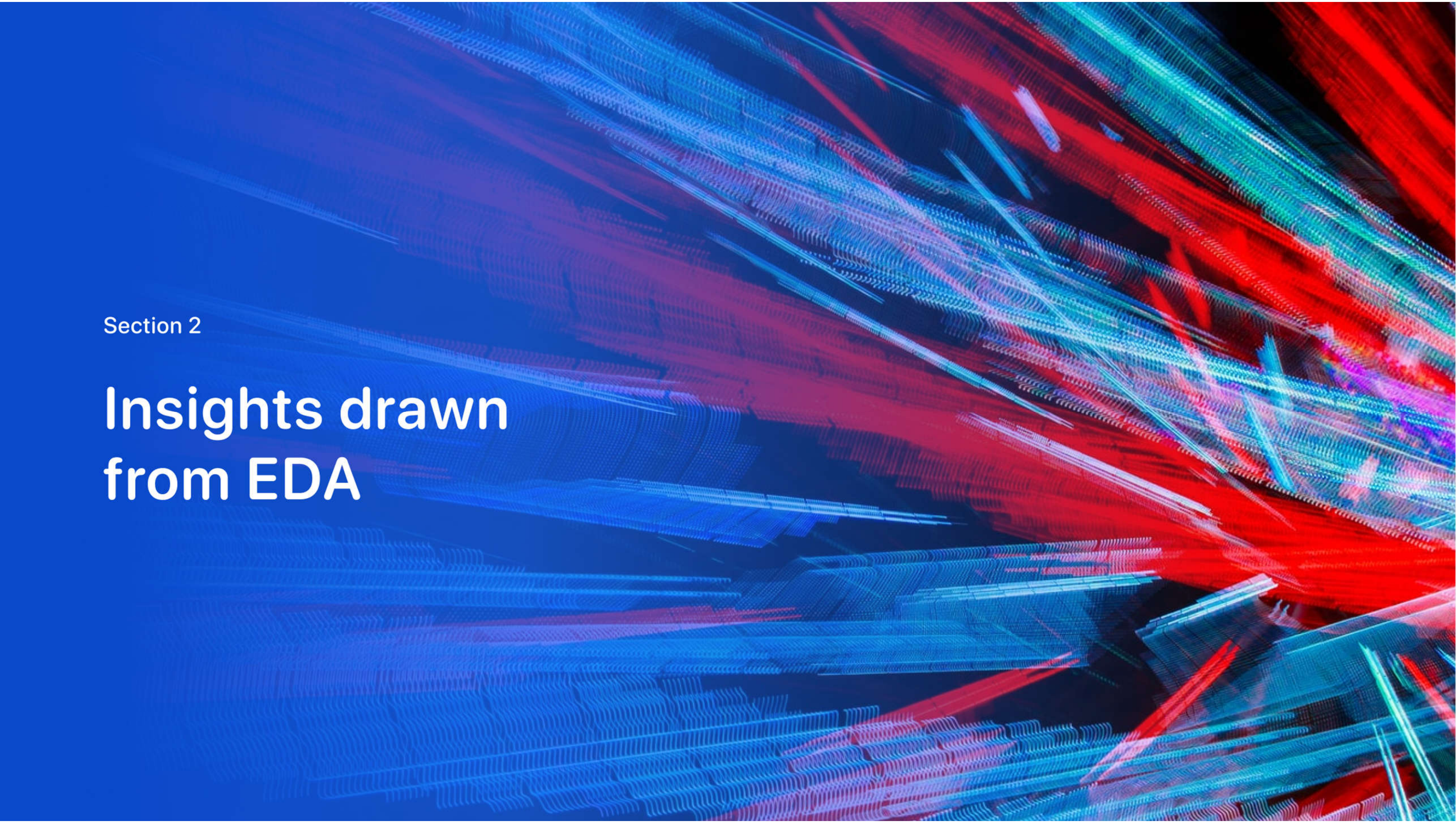
Predictive Analysis (Classification)

- Perform exploratory Data Analysis and determine Training Labels
  - Create a column for the class
  - Standardize the data
  - Split into training data and test data
- Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
  - Find the method performs best using test data

- GitHub URL:-https://github.com/jananjoy/Applied-data-science-capstone/blob/main/JR_ML_prediction(Week-4).ipynb

Results

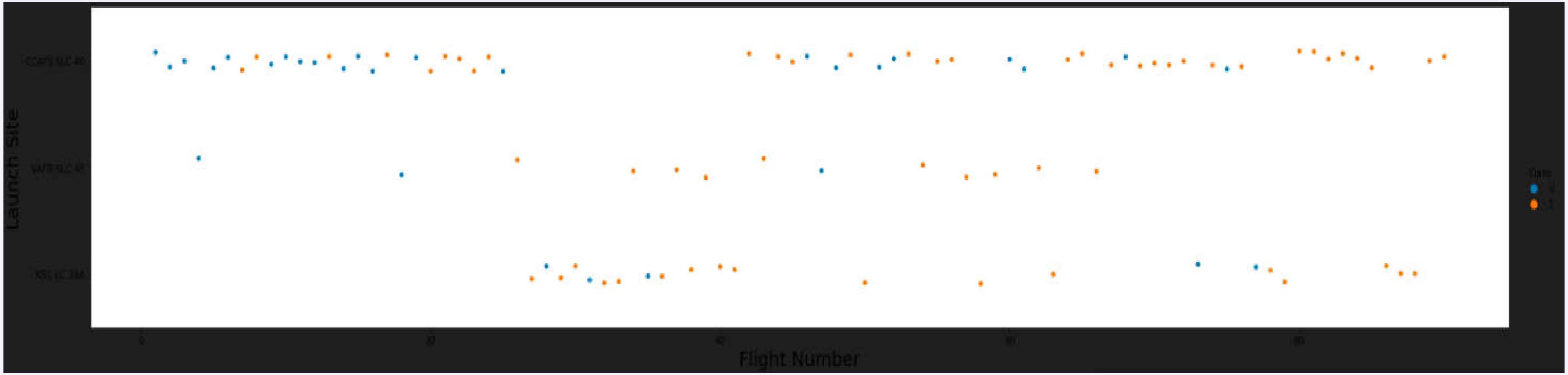
- Perform Exploratory Data Analysis and Preparing Data Feature Engineering. We find the co-relation between the attributes. For example, We see that different launch sites have different success rates from the results. Again, from observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launch site there are no rockets launched for heavypayload mass(greater than 10000).
- Find best Hyperparameter for Support Vector Machine (SVM), Classification Trees and Logistic Regression. We found different accuracy values (share in the previous slide). Classification Trees provides better accuracy.
- In Interactive analytics demo, first, Mark all launch sites on a map. Then, Mark the success/failed launches for each site on the map. Finally, Mark the success/failed launches for each site on the map. (Screenshot is presented in the slide also).



Section 2

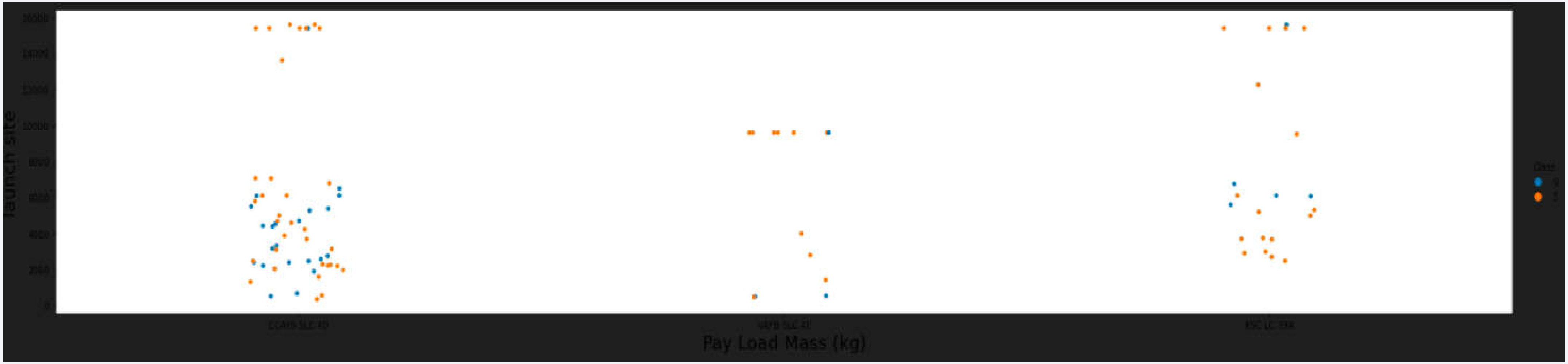
Insights drawn from EDA

Flight Number vs. Launch Site



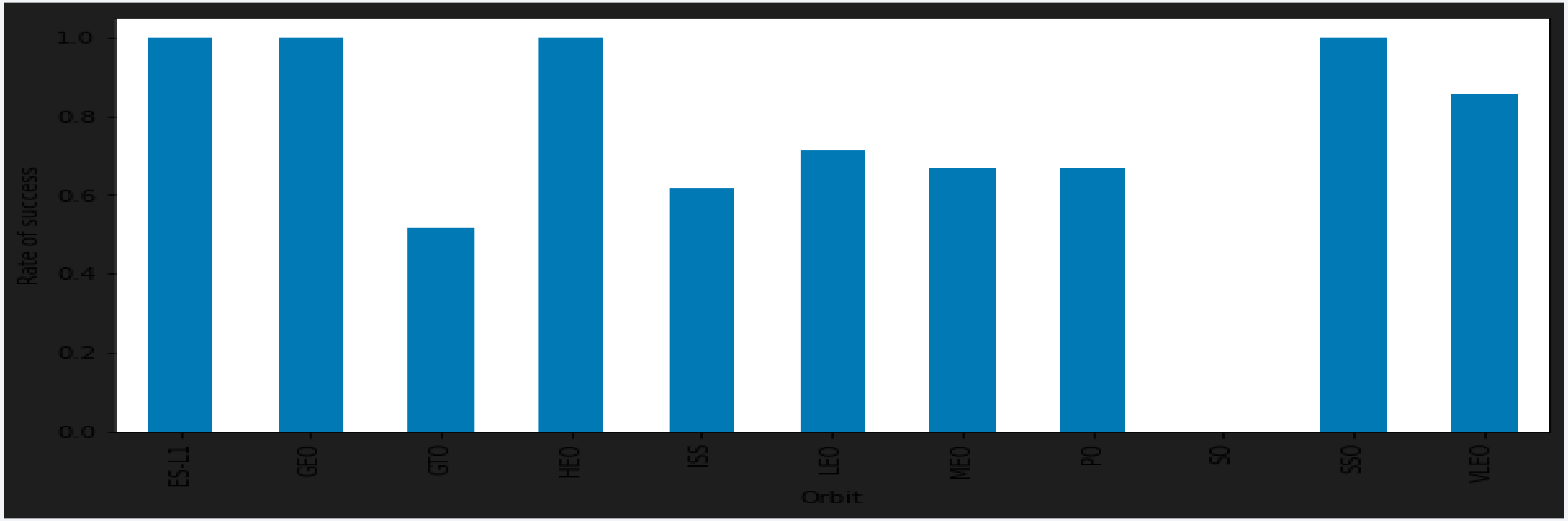
- Explain the relationship between the attributes Flight Number and Launch site

Payload vs. Launch Site



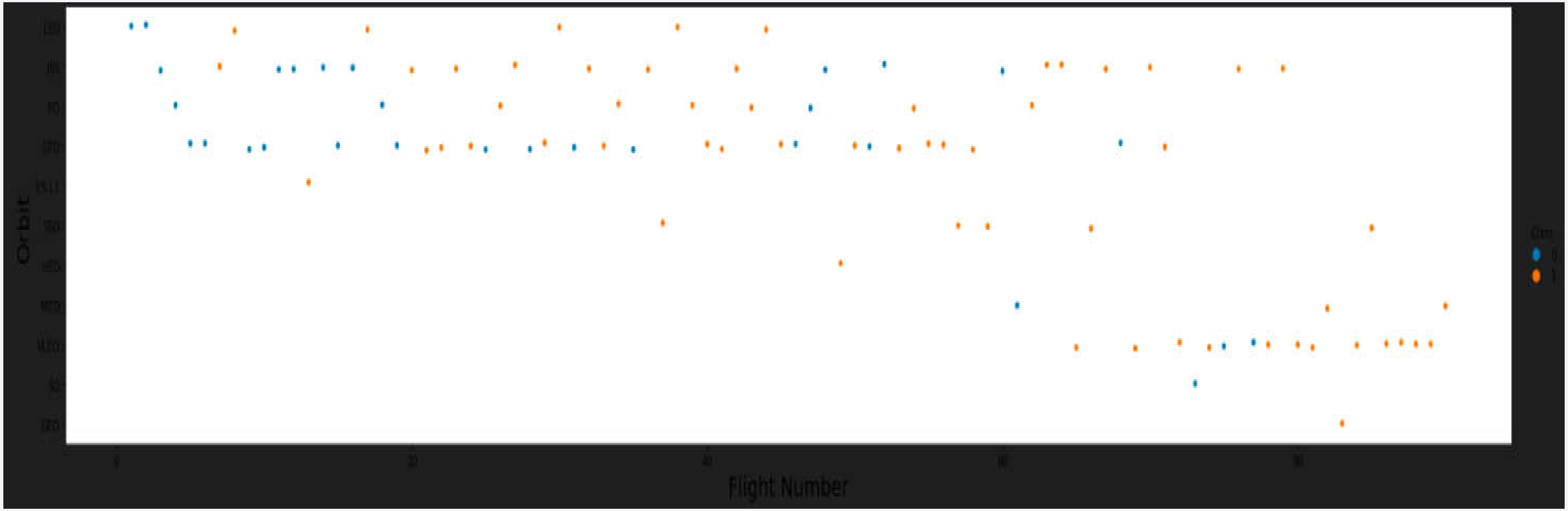
- Explain the relationship between the attributes payload and Launch site

Success Rate vs. Orbit Type



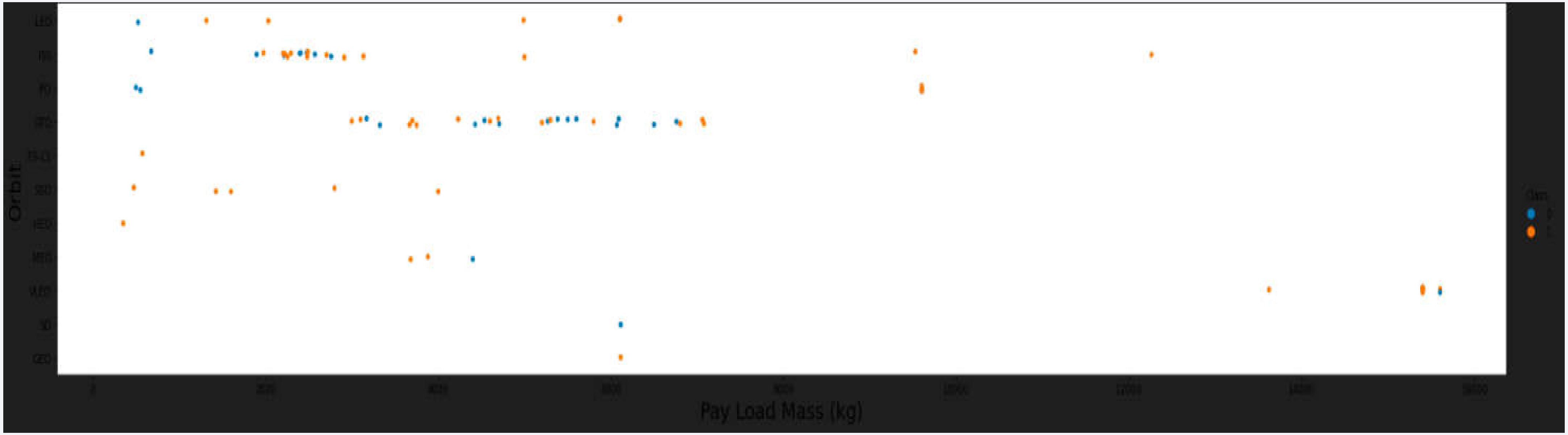
- Show the screenshot of the scatter plot with explanations

Flight Number vs. Orbit Type



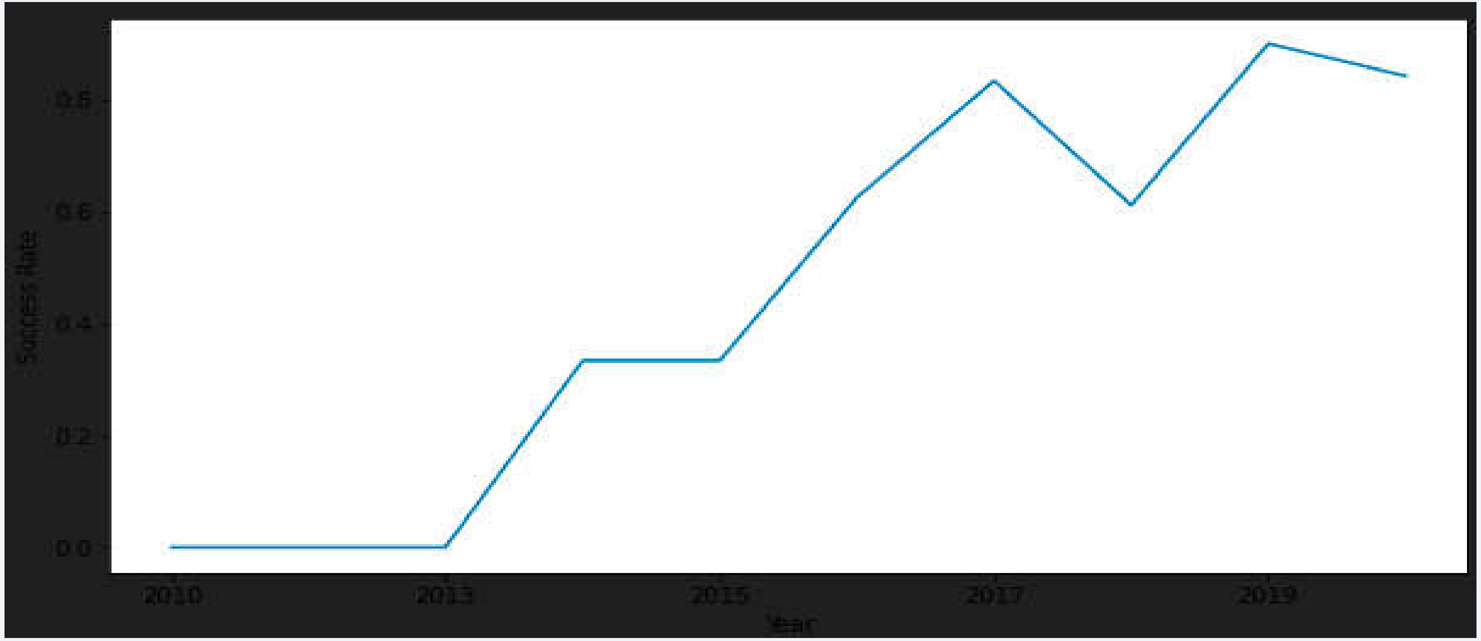
- Explain the relationship between the attributes Flight Number and Orbit Type

Payload vs. Orbit Type



- Explain the relationship between the attributes Payload and Orbit Type

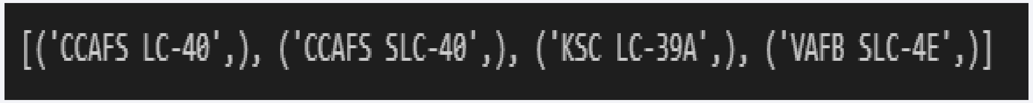
Launch Success Yearly Trend



- Explain the launch success yearly Trend based on the historical data in the data source

All Launch Site Names

- Find the names of the unique launch sites

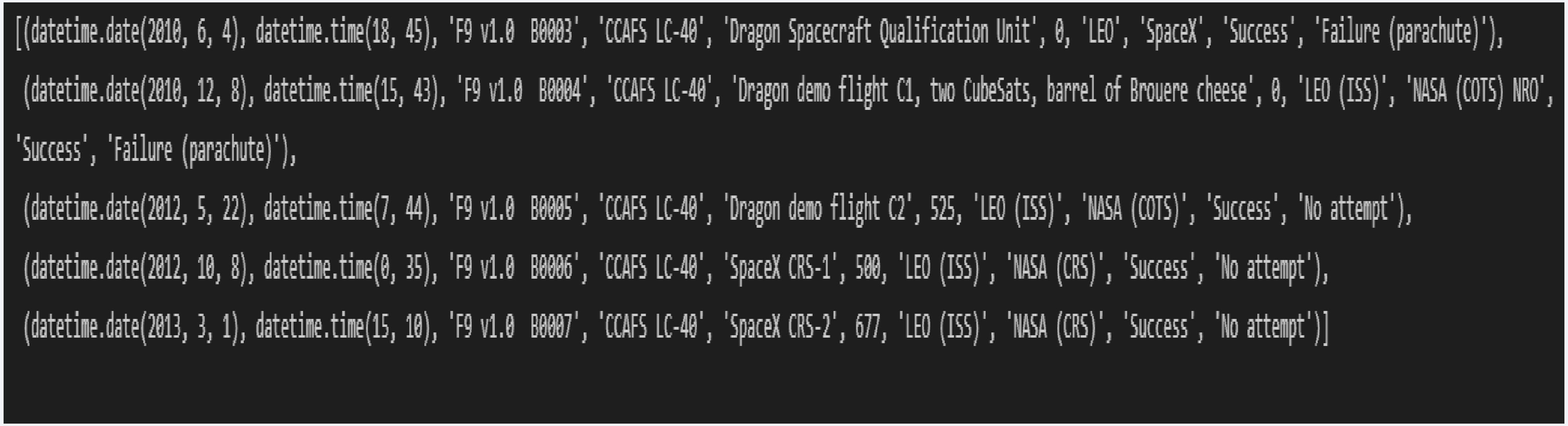


```
[('CCAFS LC-40',), ('CCAFS SLC-40',), ('KSC LC-39A',), ('VAFB SLC-4E',)]
```

- Present your query result with a short explanation here

Launch Site Names Begin with 'CCA'

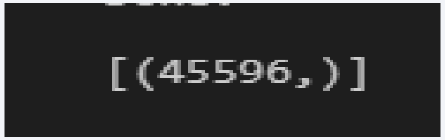
- Find 5 records where launch sites begin with `CCA`
- The query displays the launch sites started with the string CCA



```
[(datetime.date(2010, 6, 4), datetime.time(18, 45), 'F9 v1.0  B0003', 'CCAFS LC-40', 'Dragon Spacecraft Qualification Unit', 0, 'LEO', 'SpaceX', 'Success', 'Failure (parachute)'),
 (datetime.date(2010, 12, 8), datetime.time(15, 43), '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)'),
 (datetime.date(2012, 5, 22), datetime.time(7, 44), 'F9 v1.0  B0005', 'CCAFS LC-40', 'Dragon demo flight C2', 525, 'LEO (ISS)', 'NASA (COTS)', 'Success', 'No attempt'),
 (datetime.date(2012, 10, 8), datetime.time(0, 35), 'F9 v1.0  B0006', 'CCAFS LC-40', 'SpaceX CRS-1', 500, 'LEO (ISS)', 'NASA (CRS)', 'Success', 'No attempt'),
 (datetime.date(2013, 3, 1), datetime.time(15, 10), 'F9 v1.0  B0007', 'CCAFS LC-40', 'SpaceX CRS-2', 677, 'LEO (ISS)', 'NASA (CRS)', 'Success', 'No attempt')]
```

Total Payload Mass

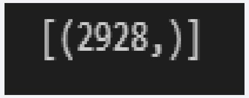
- Calculate the total payload carried by boosters from NASA
- The above query display the total payload carried by boosters from NASA



```
[(45596,)]
```

Average Payload Mass by F9 v1.1

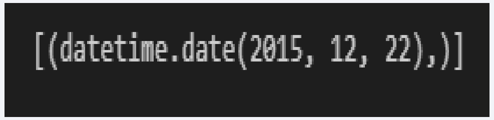
- Calculate the average payload mass carried by booster version F9 v1.1
- The above query calculate the average payload mass carried by booster version F9 v1.1



```
[(2928,)]
```

First Successful Ground Landing Date

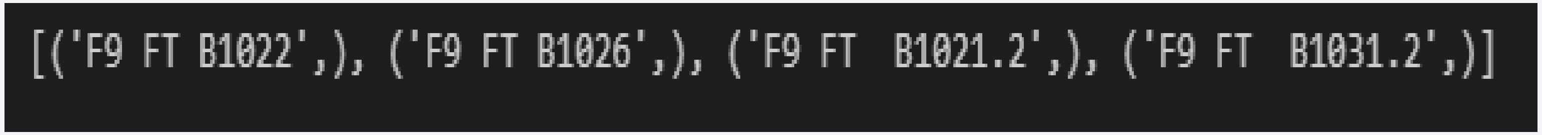
- Find the dates of the first successful landing outcome on ground pad
- The above query find and return the date of the first successful landing outcome on ground pad



```
[(datetime.date(2015, 12, 22),)]
```

Successful Drone Ship Landing with Payload between 4000 and 6000

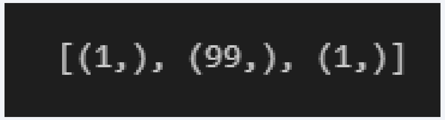
- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- The above query returns the names of the boosters meeting the condition



```
[('F9 FT B1022',), ('F9 FT B1026',), ('F9 FT  B1021.2',), ('F9 FT  B1031.2',)]
```

Total Number of Successful and Failure Mission Outcomes

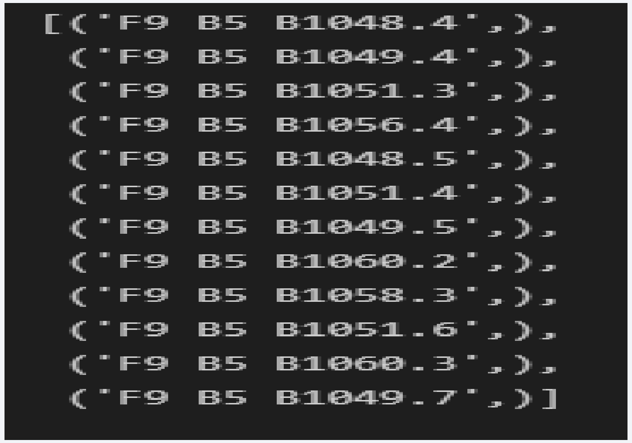
- Calculate the total number of successful and failure mission outcomes
- The above query calculate the total number of successful and failure mission outcomes



```
[(1,), (99,), (1,)]
```

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- The above query lists the of the booster which have carried the maximum payload mass

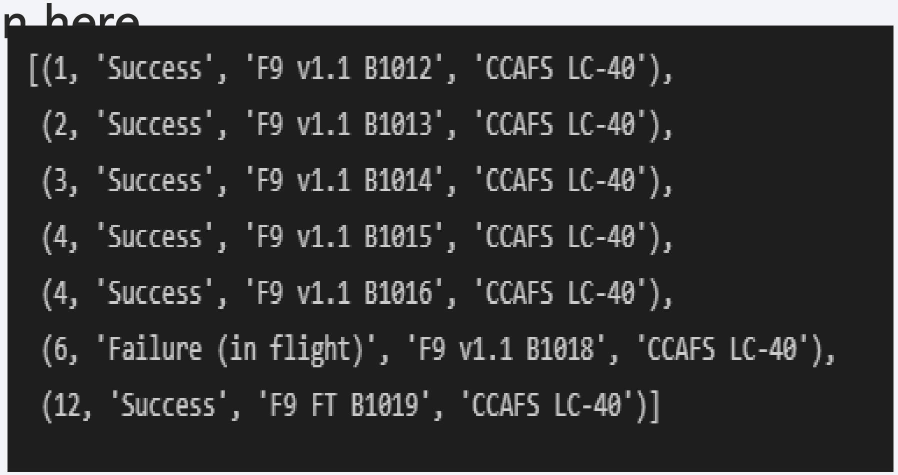


```
[('F9 B5 B1048.4',),
 ('F9 B5 B1049.4',),
 ('F9 B5 B1051.3',),
 ('F9 B5 B1056.4',),
 ('F9 B5 B1048.5',),
 ('F9 B5 B1051.4',),
 ('F9 B5 B1049.5',),
 ('F9 B5 B1060.2',),
 ('F9 B5 B1058.3',),
 ('F9 B5 B1051.6',),
 ('F9 B5 B1060.3',),
 ('F9 B5 B1049.7',)]
```

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

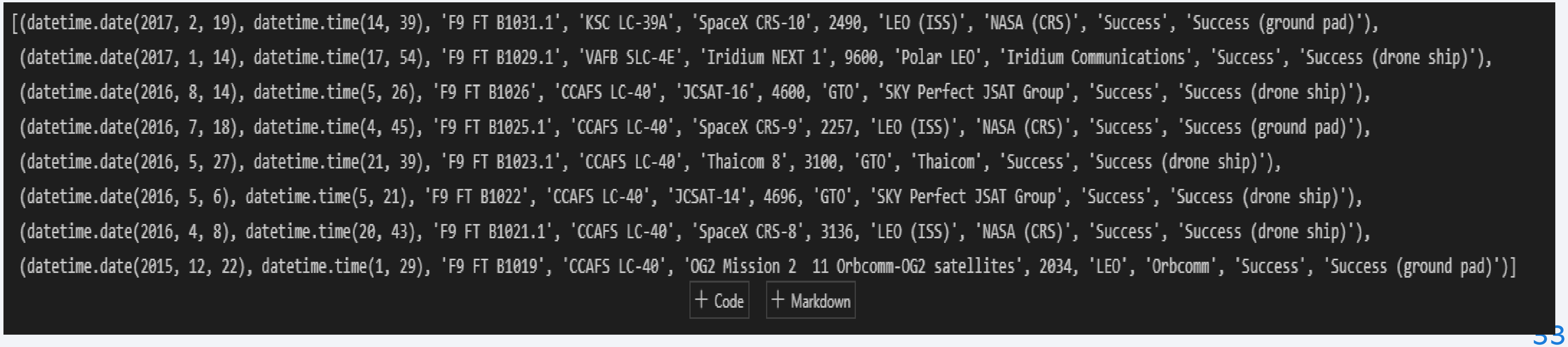
- Present your query result with a short explanation here



```
[(1, 'Success', 'F9 v1.1 B1012', 'CCAFS LC-40'),
 (2, 'Success', 'F9 v1.1 B1013', 'CCAFS LC-40'),
 (3, 'Success', 'F9 v1.1 B1014', 'CCAFS LC-40'),
 (4, 'Success', 'F9 v1.1 B1015', 'CCAFS LC-40'),
 (4, 'Success', 'F9 v1.1 B1016', 'CCAFS LC-40'),
 (6, 'Failure (in flight)', 'F9 v1.1 B1018', 'CCAFS LC-40'),
 (12, 'Success', 'F9 FT B1019', 'CCAFS LC-40')]
```

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

- 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
- The above query rank the count of the landing outcomes meeting the specify condition

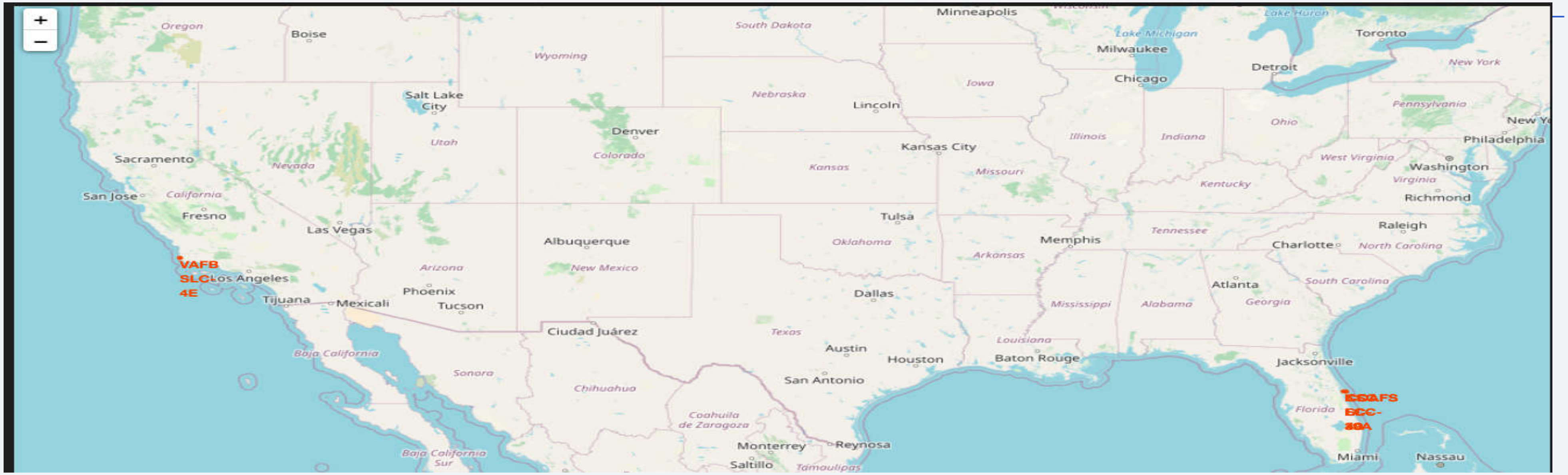


```
[(datetime.date(2017, 2, 19), datetime.time(14, 39), 'F9 FT B1031.1', 'KSC LC-39A', 'SpaceX CRS-10', 2490, 'LEO (ISS)', 'NASA (CRS)', 'Success', 'Success (ground pad)'),
 (datetime.date(2017, 1, 14), datetime.time(17, 54), 'F9 FT B1029.1', 'VAFB SLC-4E', 'Iridium NEXT 1', 9600, 'Polar LEO', 'Iridium Communications', 'Success', 'Success (drone ship)'),
 (datetime.date(2016, 8, 14), datetime.time(5, 26), 'F9 FT B1026', 'CCAFS LC-40', 'JCSAT-16', 4600, 'GTO', 'SKY Perfect JSAT Group', 'Success', 'Success (drone ship)'),
 (datetime.date(2016, 7, 18), datetime.time(4, 45), 'F9 FT B1025.1', 'CCAFS LC-40', 'SpaceX CRS-9', 2257, 'LEO (ISS)', 'NASA (CRS)', 'Success', 'Success (ground pad)'),
 (datetime.date(2016, 5, 27), datetime.time(21, 39), 'F9 FT B1023.1', 'CCAFS LC-40', 'Thaicom 8', 3100, 'GTO', 'Thaicom', 'Success', 'Success (drone ship)'),
 (datetime.date(2016, 5, 6), datetime.time(5, 21), 'F9 FT B1022', 'CCAFS LC-40', 'JCSAT-14', 4696, 'GTO', 'SKY Perfect JSAT Group', 'Success', 'Success (drone ship)'),
 (datetime.date(2016, 4, 8), datetime.time(20, 43), 'F9 FT B1021.1', 'CCAFS LC-40', 'SpaceX CRS-8', 3136, 'LEO (ISS)', 'NASA (CRS)', 'Success', 'Success (drone ship)'),
 (datetime.date(2015, 12, 22), datetime.time(1, 29), 'F9 FT B1019', 'CCAFS LC-40', 'OG2 Mission 2  11 Orbcomm-OG2 satellites', 2034, 'LEO', 'Orbcomm', 'Success', 'Success (ground pad)')]
```

Section 4

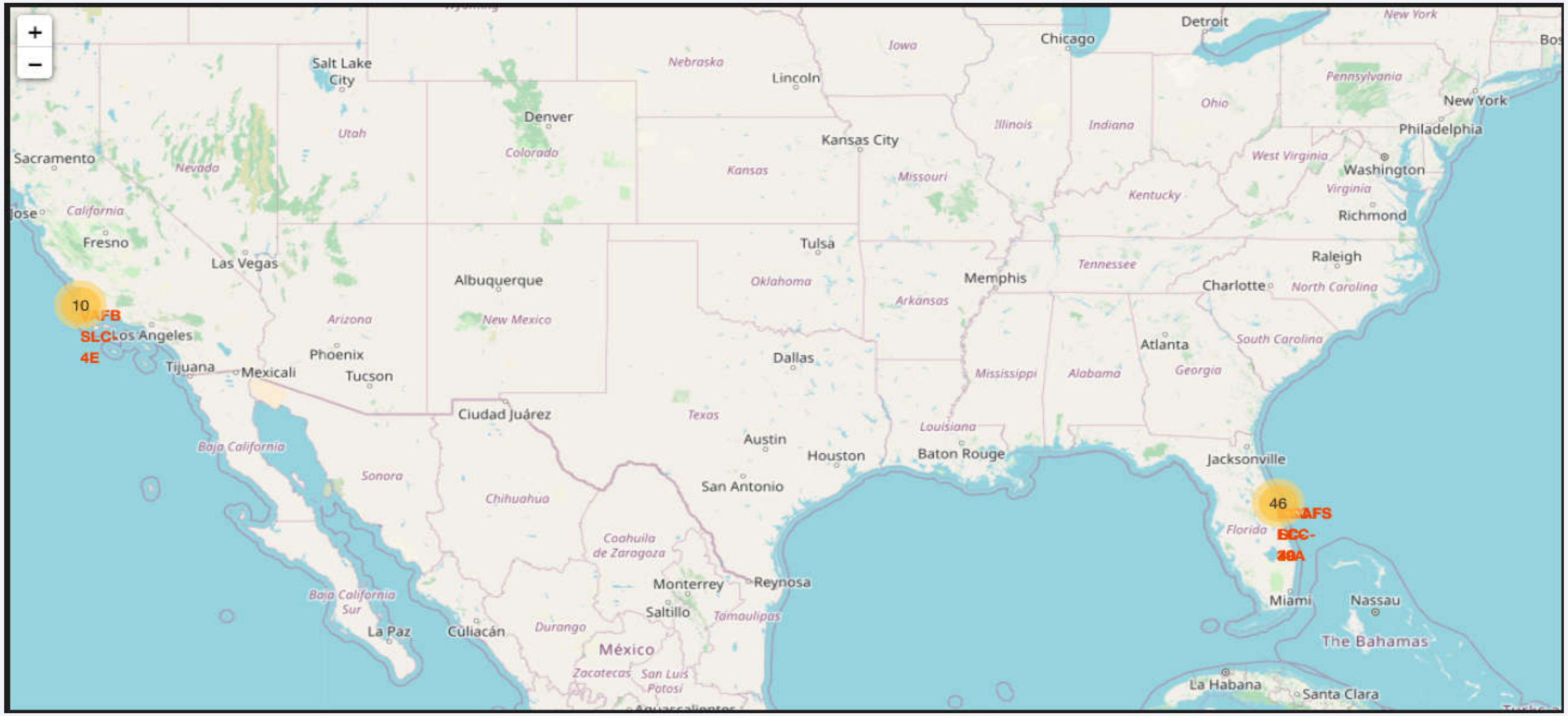
Launch Sites Proximities Analysis

Mark all launch site on the map



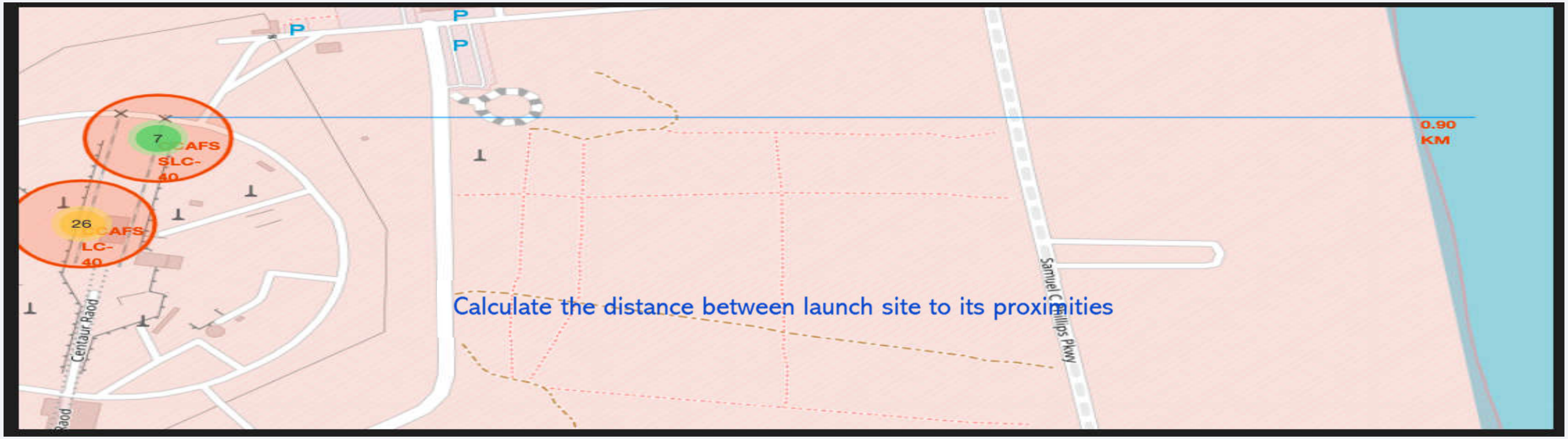
- The above picture displays all the launch site on the map

Mark success/failure launches for each site on the map

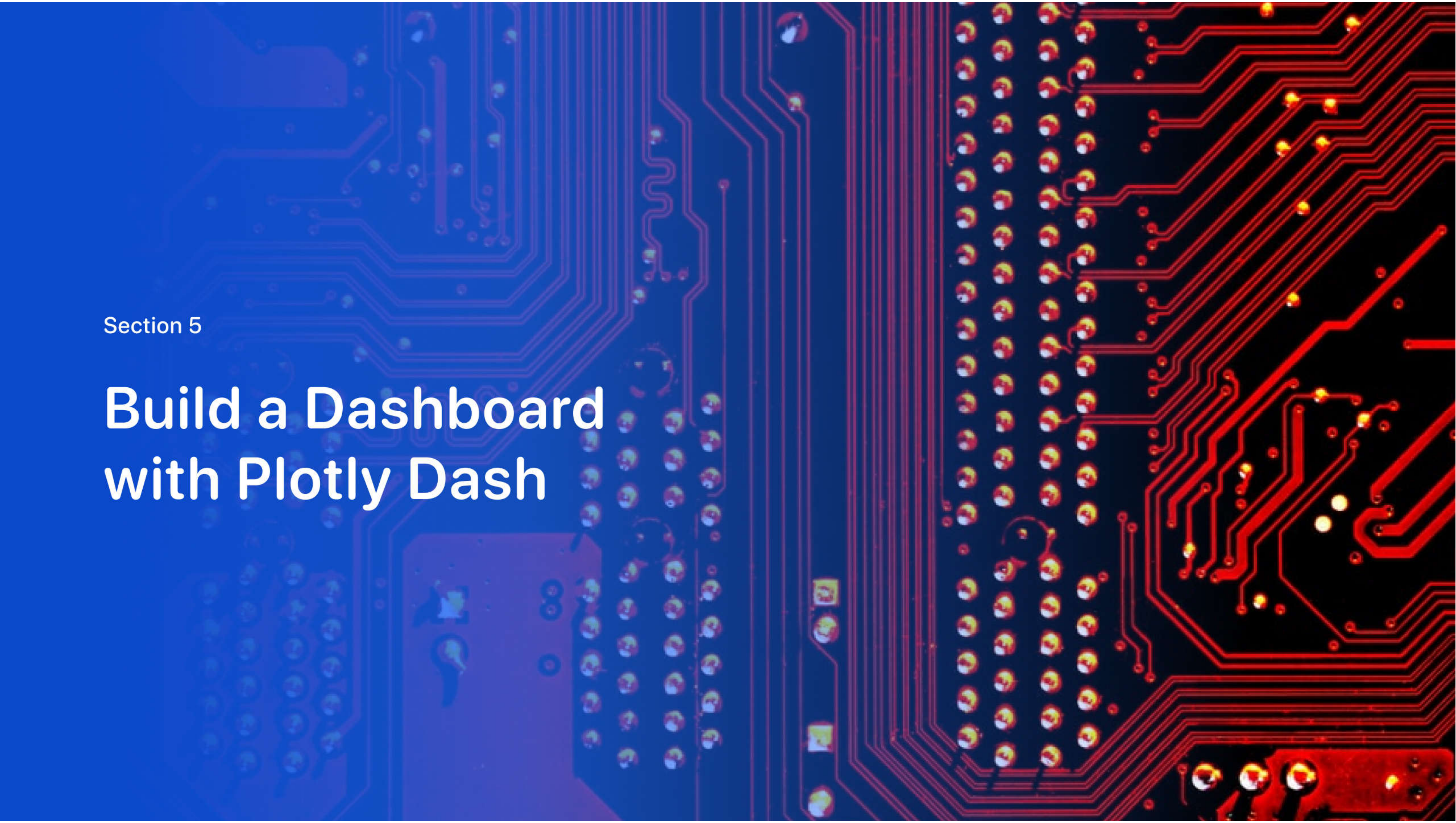


- The above picture mark success/failure launches for each site on the map

Calculate the distance between launch site to its proximities



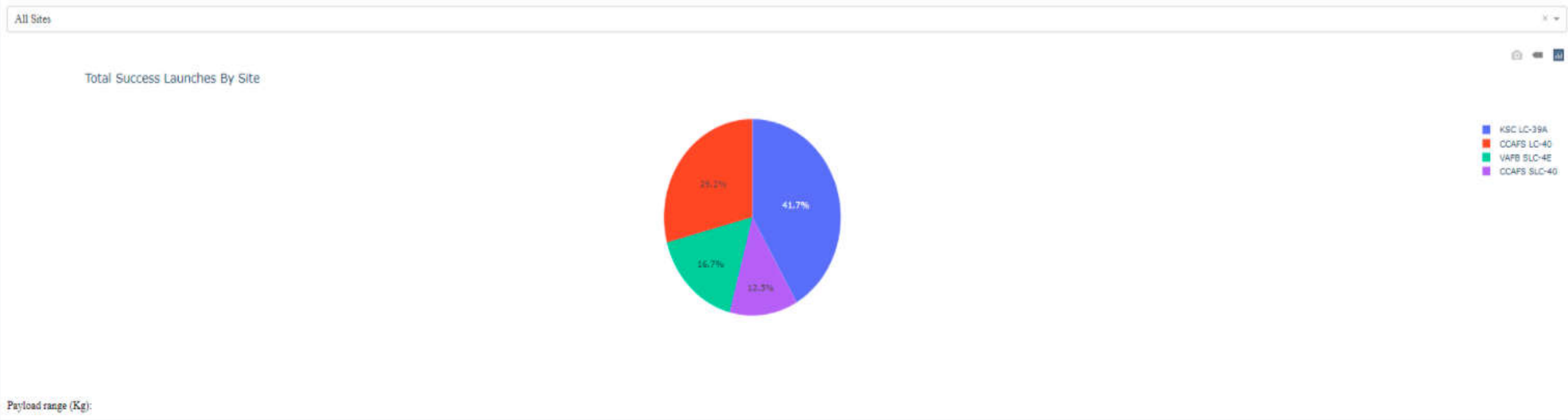
- The above picture calculate the distance between launch site to its proximities



Section 5

Build a Dashboard with Plotly Dash

Launch success Dashboard



All Sites

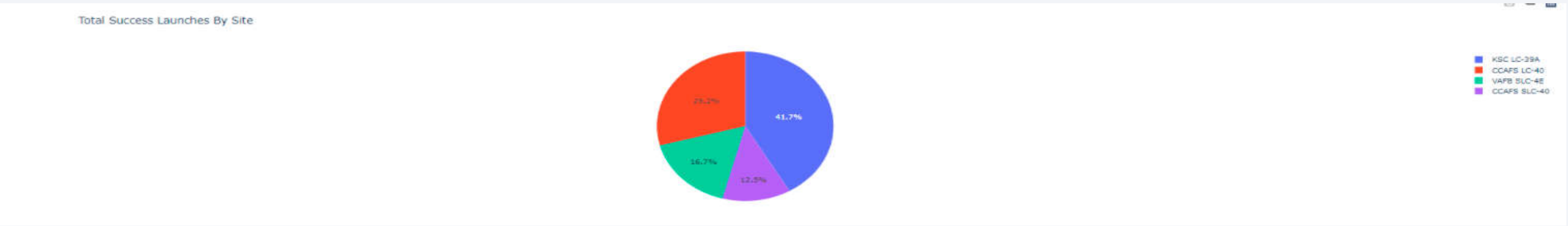
Payload range (Kg):

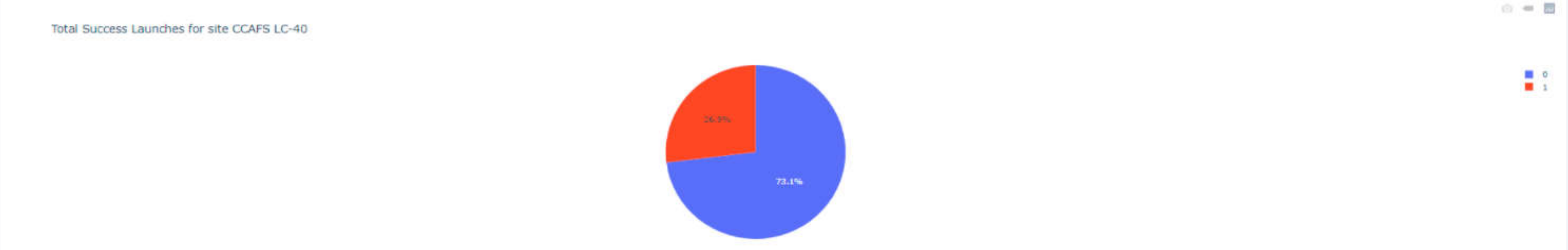
Note:- Since my lab environment is not working. I kept the screeshot from the lab instruction.

At the same time, I refer my answer with reference to the mentioned screenshot. During the course I already executed the code.

- The Screenshot provide the information about the largest successful launches and the highest launch success rate .

Launch site with highest launch success ratio



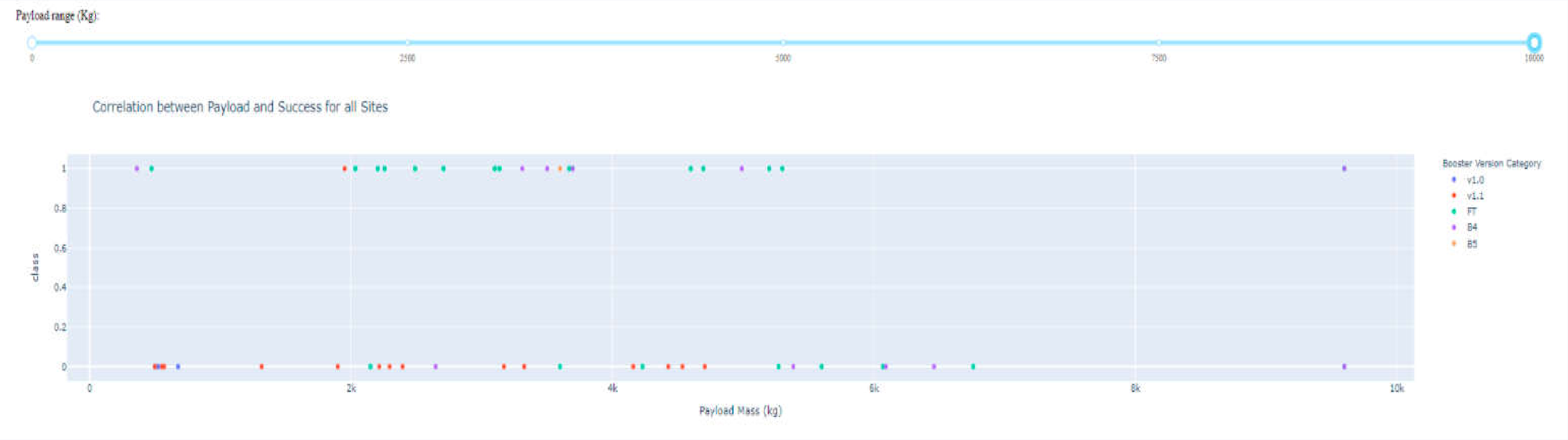


Note:- Since my lab environment is not working. I kept the screeshot from the lab instruction.

At the same time, I refer my answer with reference to the mentioned screenshot. During the course I already executed the code.

- These screenshot provide information of the success ratio for all the site and for specific launch site.

Payload vs. Launch Outcome scatter plot for all sites



Note:- Since my lab environment is not working. I kept the screeshot from the lab instruction.

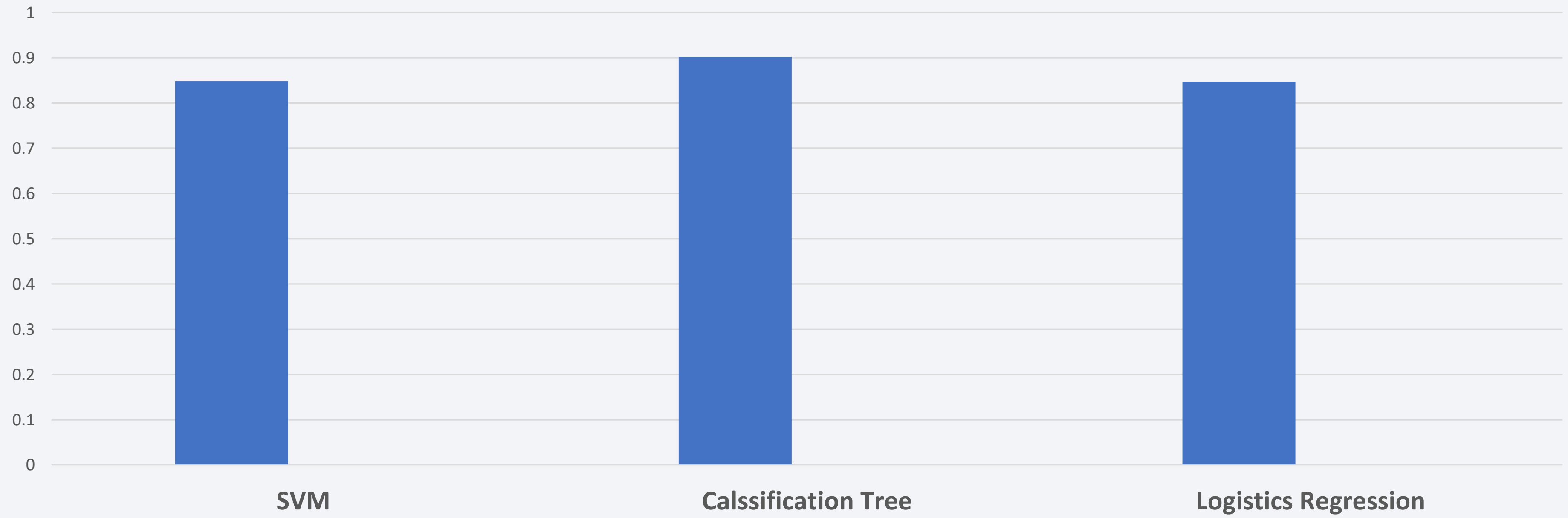
At the same time, I refer my answer with reference to the mentioned screenshot. During the course I already executed the code.

- From the screenshot, the following information can be observed such as largest successful launches and highest launch success rate of a site, payload range(s) having highest/lowest launch success rate. Besides, v1.0 F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest launch success rate.

Section 6

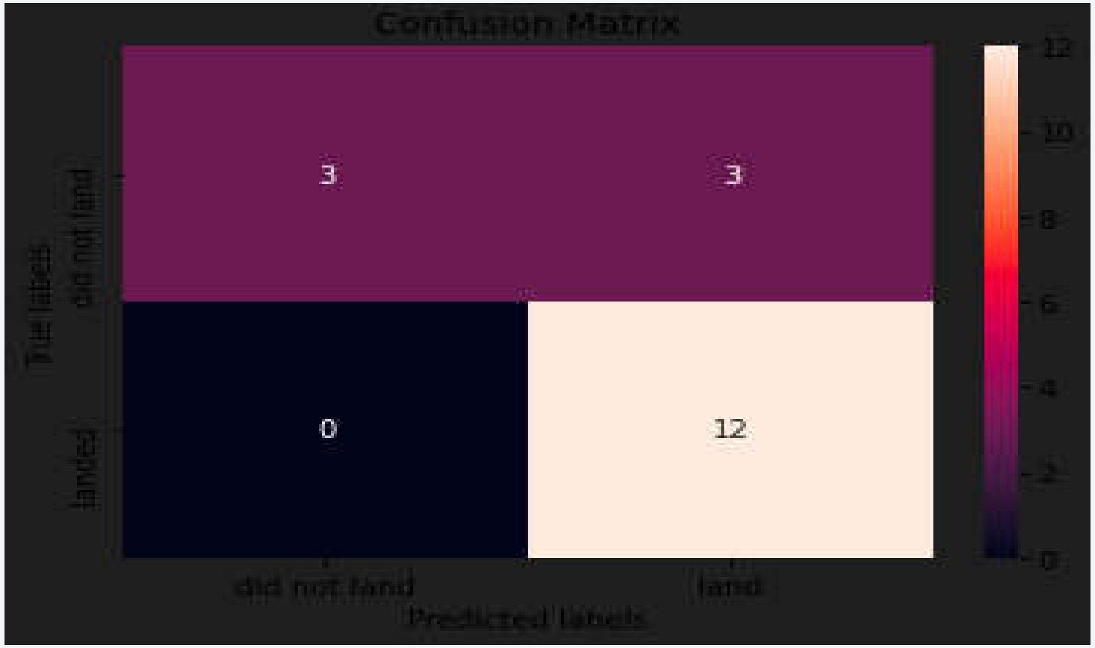
Predictive Analysis (Classification)

Classification Accuracy



- Classification tree model has the highest accuracy [accuracy : 0.9017857142857144]

Confusion Matrix



- Show the confusion matrix of the best performing model with an explanation

Conclusions

- Perform data collection using REST API and data wrangling operation which includes sampling, dealing with nulls (missing data points)
- Perform exploratory data analysis and execute SQL query. Besides we plot the data to visualize the result.
- Build an interactive dashboard with ploty dash for visual data analytics.
- Build a predictive machine learning model and analyze the results with accuracy.

Appendix

- Spacex dataset in the .csv format
- Virtual Lab environment for running the code
- IBM cloud platform for executing, storing and managing the code and documents.
- Use the GitHub as repository.
- All codes, documents, presentation are available in the repository.

Thank you!