IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection methodology
  - Data wrangling
  - Exploratory data analysis (EDA) using visualization and SQL
  - Interactive visual analytics using Folium and Plotly Dash
  - Predictive analysis using classification models
- Summary of all results
  - Exploratory data analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

Introduction

- SpaceX, or Space Exploration Technologies Corp., is an American aerospace manufacturer and space transportation company founded by Elon Musk in 2002.
- One of the best innovation ideas in the world is SpaceX creating a reusable rocket technology.
- The Falcon 9 is a two-stage orbital launch vehicle developed and manufactured by SpaceX.The ability to land and reuse the first stage contributes significantly to reducing the cost of space travel.
- The prediction of Falcon 9 Landing using Data Science.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- 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

- Describe how data sets were collected.

Downloading from spacex

```
[6]: spacex_url="https://api.spacexdata.com/v4/launches/past"

[7]: response = requests.get(spacex_url)
```

Converting into dataframe

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

```
[23]: # Create a data from launch_dict
      data=pd.DataFrame(launch_dict)
```

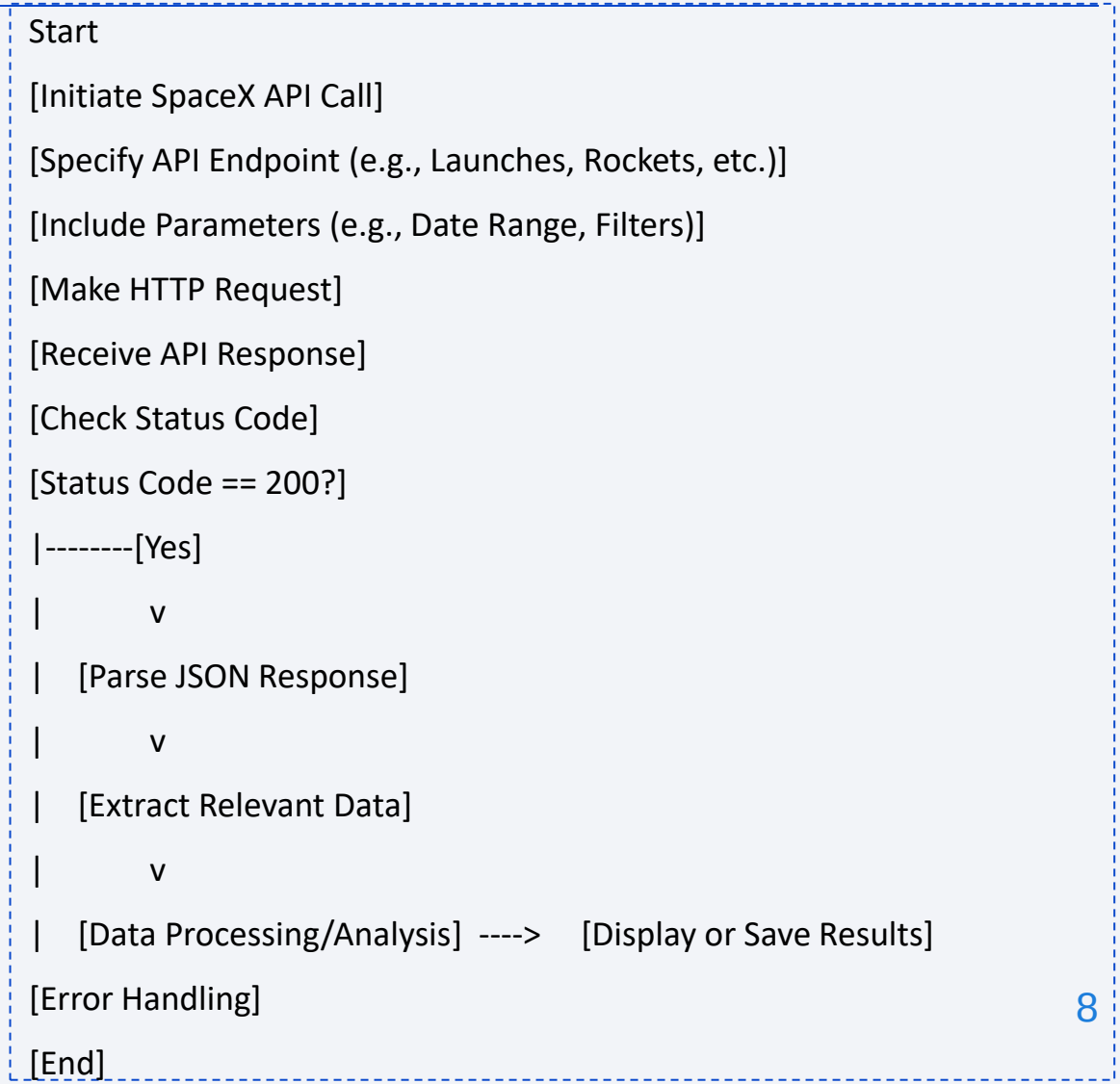
Cleansing data

```
[37]: # Hint data['BoosterVersion']!='Falcon 1'
      data_falcon9=data[data['BoosterVersion']!='Falcon 1']
      data_falcon9.head()
```

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Wrangling

- Describe how data were processed

- Cleansing all the null values and replacing them with the mean value.

```
[32]: data_falcon9.isnull().sum()

[32]: FlightNumber       0
      Date               0
      BoosterVersion     0
      PayloadMass        5
      Orbit              0
      LaunchSite         0
      Outcome            0
      Flights            0
      GridFins           0
      Reused             0
      Legs               0
      LandingPad        26
      Block              0
      ReusedCount        0
      Serial             0
      Longitude          0
      Latitude           0
      dtype: int64
```

```
[34]: # Calculate the mean value of PayloadMass column
      mean=data_falcon9['PayloadMass'].mean()
      # Replace the np.nan values with its mean value
      data_falcon9.replace(np.nan,mean)
```

- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb
- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- There are various types of charts and graphs, each suitable for different data types and purposes.
- Bar chart:Show the relationship between individual data points and a categorical variable
- Line chart: Display trends over a continuous interval or time series.
- Pie chart:Represent parts of a whole; each slice represents a proportion of the entire dataset.
- Scatter Plot:Display the relationship between two continuous variables, showcasing individual data points.
- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- SQL (Structured Query Language) is commonly used in conjunction with machine learning tasks, especially when dealing with databases and data preprocessing.
- Some of the SQL commands used in the project apart from the basic DDL and DML commands are:
- Functions like Rank(),GroupBy(),OrderBY(),Min(),Max(),Avg(),Sum()

- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- In the context of creating and adding elements to a Folium map, various map objects such as markers, circles, and lines are commonly utilized.
  - Markers are used to pinpoint specific locations on the map.
  - Circles are employed to represent areas of influence or coverage around a central point.
  - Lines are used to depict paths, routes, or connections between two or more points on the map.

(download was not supporting so I added a pdf below)

- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/JupyterLite-Folium.pdf

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- I have calculated and predicted the output using four different algorithms and compared their accuracy.

(download was not supporting so I added a pdf below)

- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone/blob/main/JupyterLite-Machine%20learning%20prediction.pdf

Results

- Exploratory data analysis results

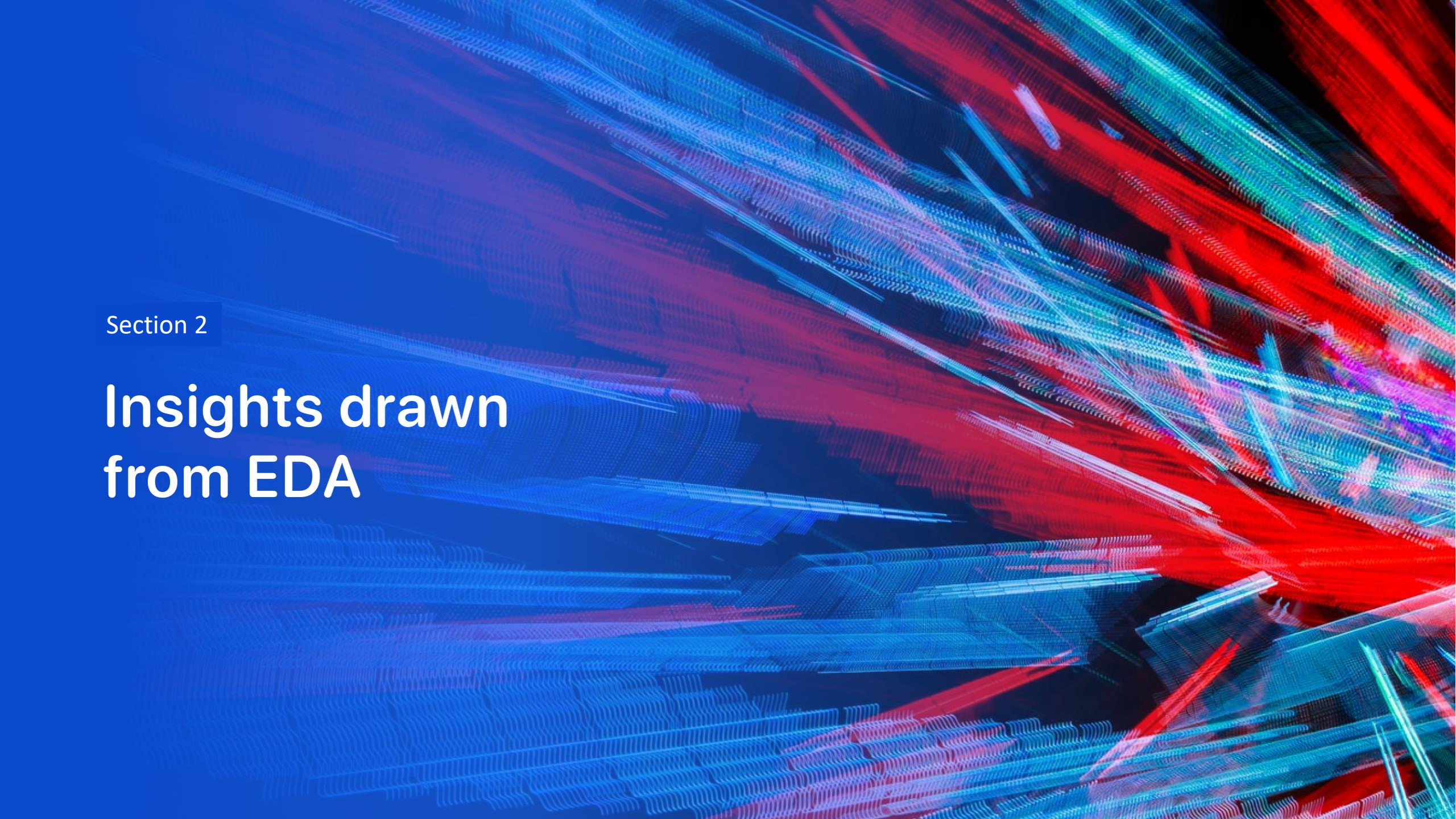
Landing Outcome	RANK() OVER(ORDER BY COUNT(LANDING_OUTCOME) ASC)
Precluded (drone ship)	1
Failure (parachute)	2
Uncontrolled (ocean)	2
Controlled (ocean)	4
Success (ground pad)	4
Failure (drone ship)	6
Success (drone ship)	6
No attempt	8

- Interactive analytics demo in screenshots
- Predictive analysis results

```
Best Algorithm is Tree with a score of 0.875
Best Params is : {'criterion': 'entropy', 'max_depth': 2, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_sam
ples_split': 2, 'splitter': 'best'}
```

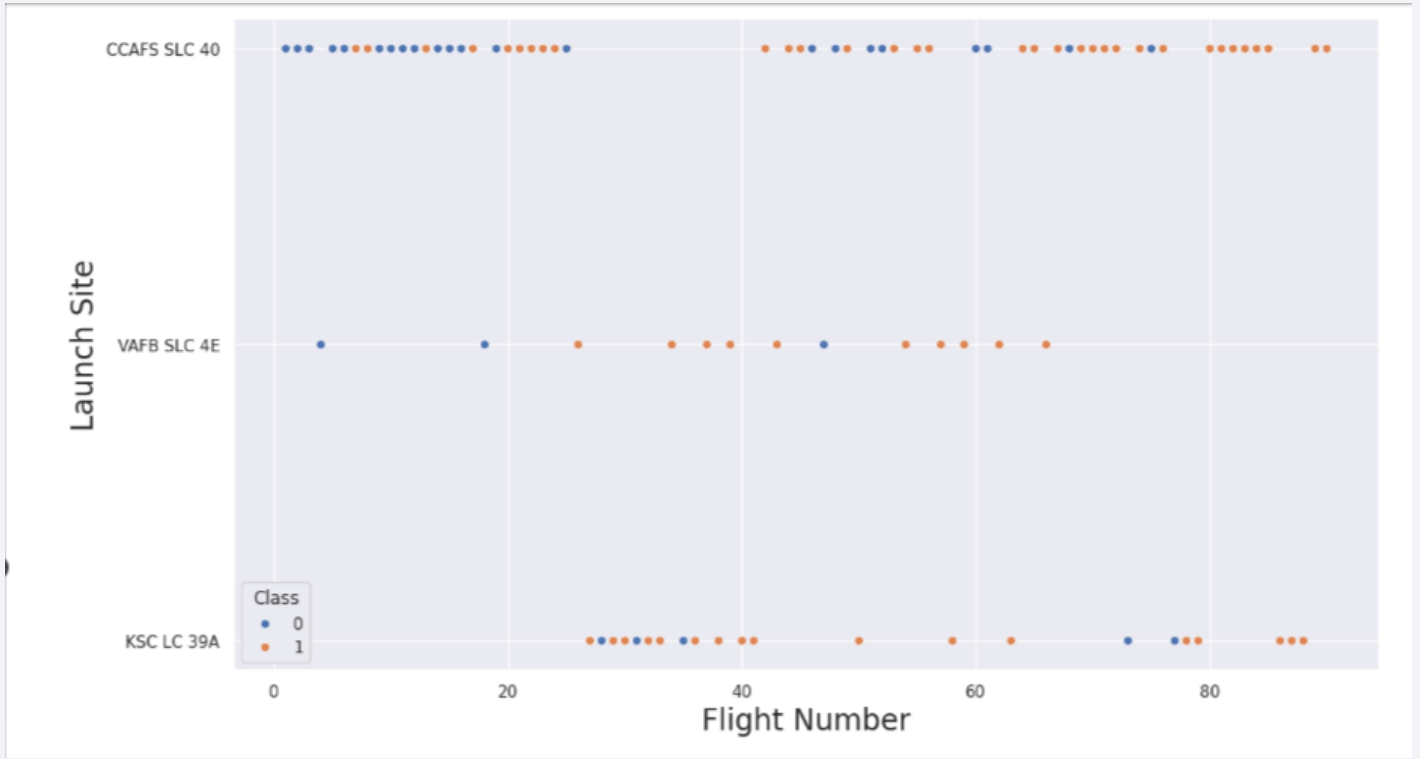
Section 2

Insights drawn from EDA



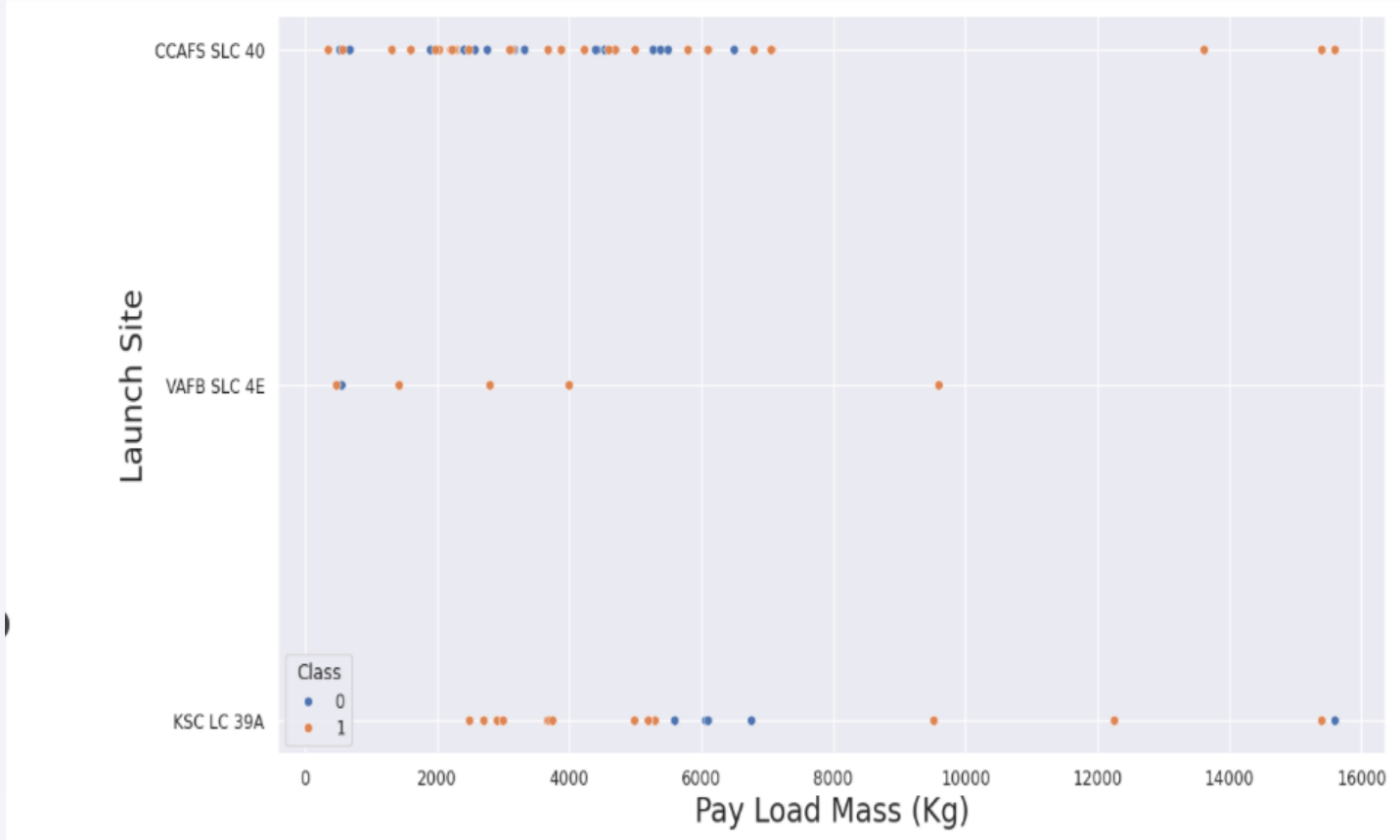
Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site
- Show the screenshot of the scatter plot with explanations



Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site
- Show the screenshot of the scatter plot with explanations

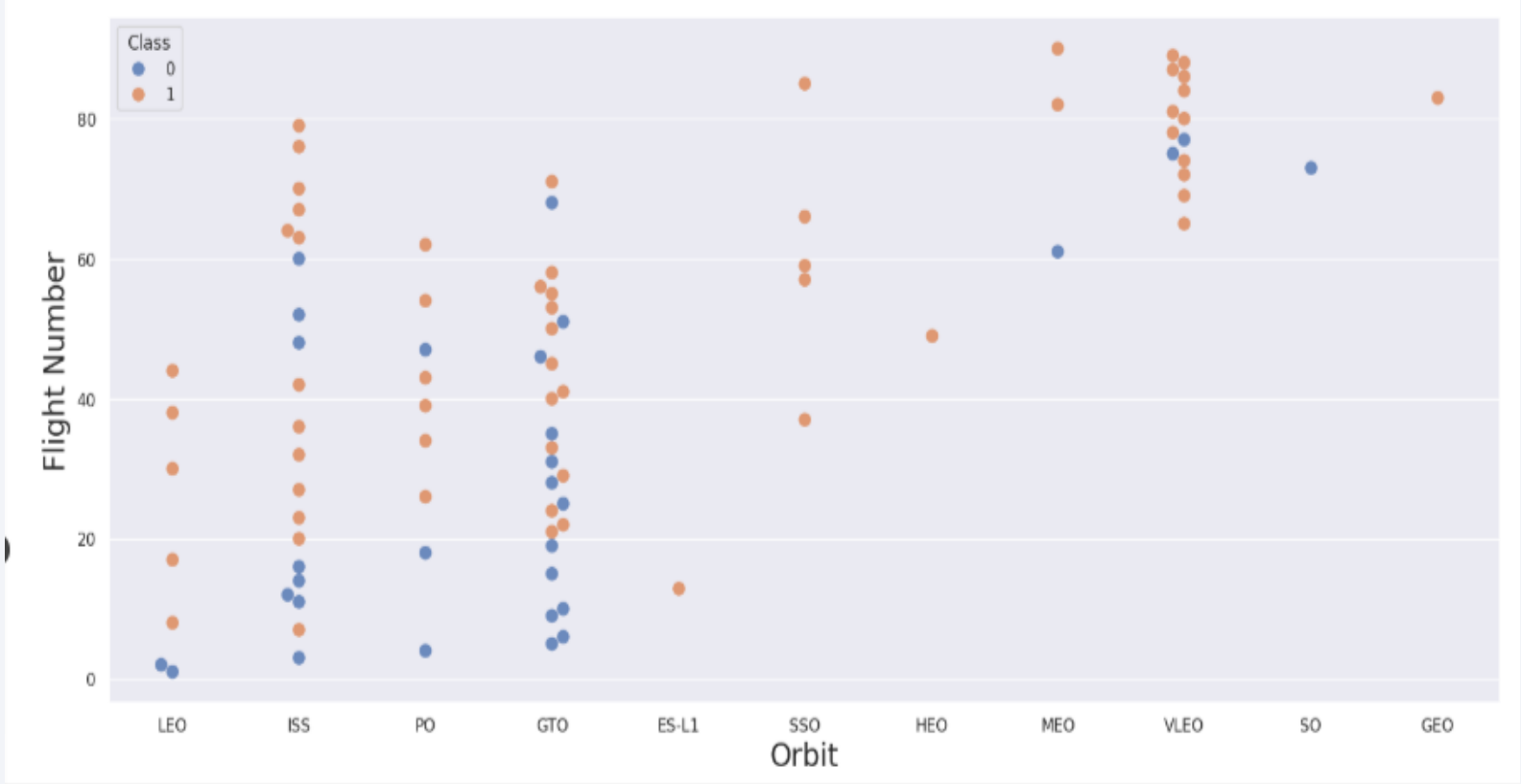


Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type
- Show the screenshot of the scatter plot with explanations

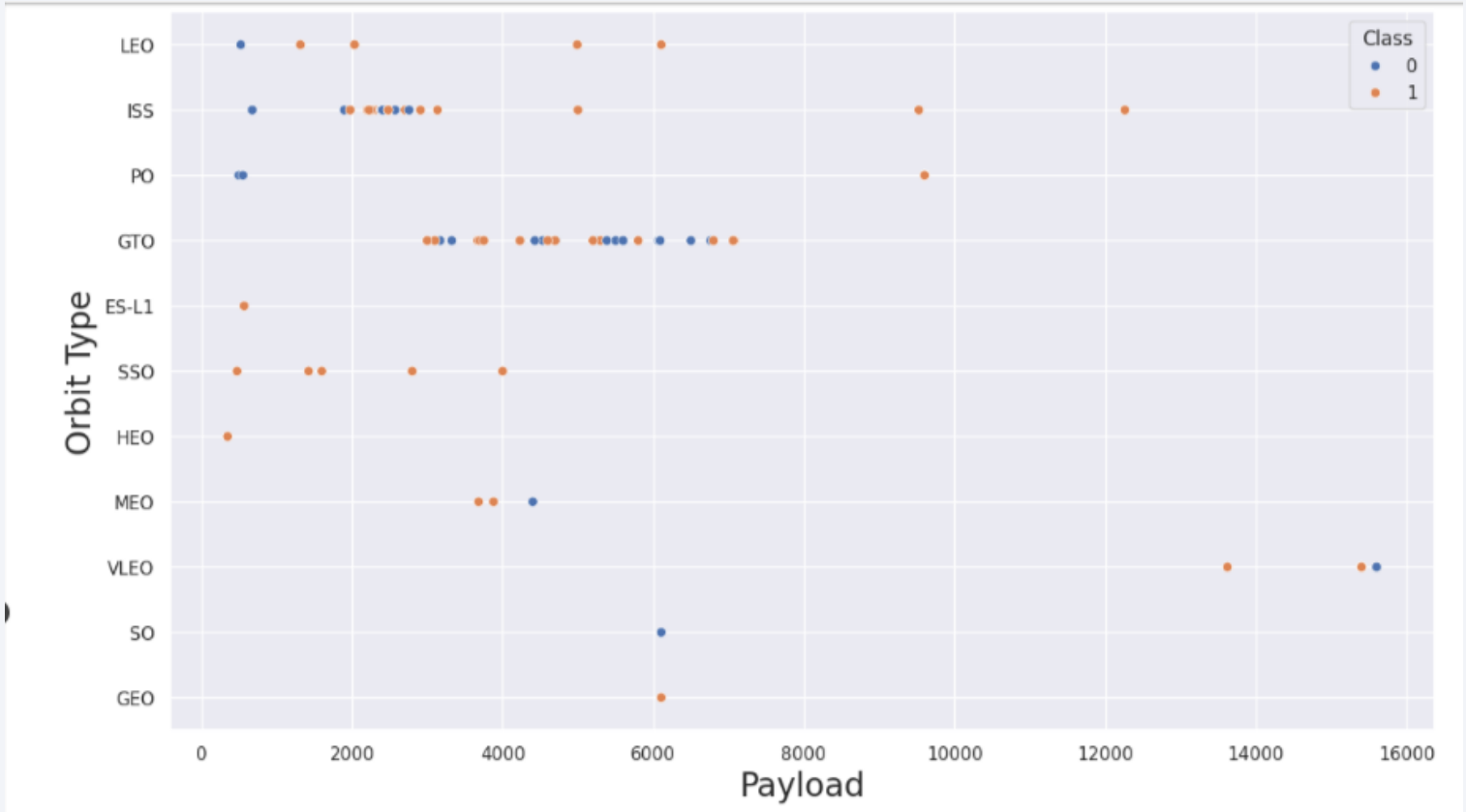
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations



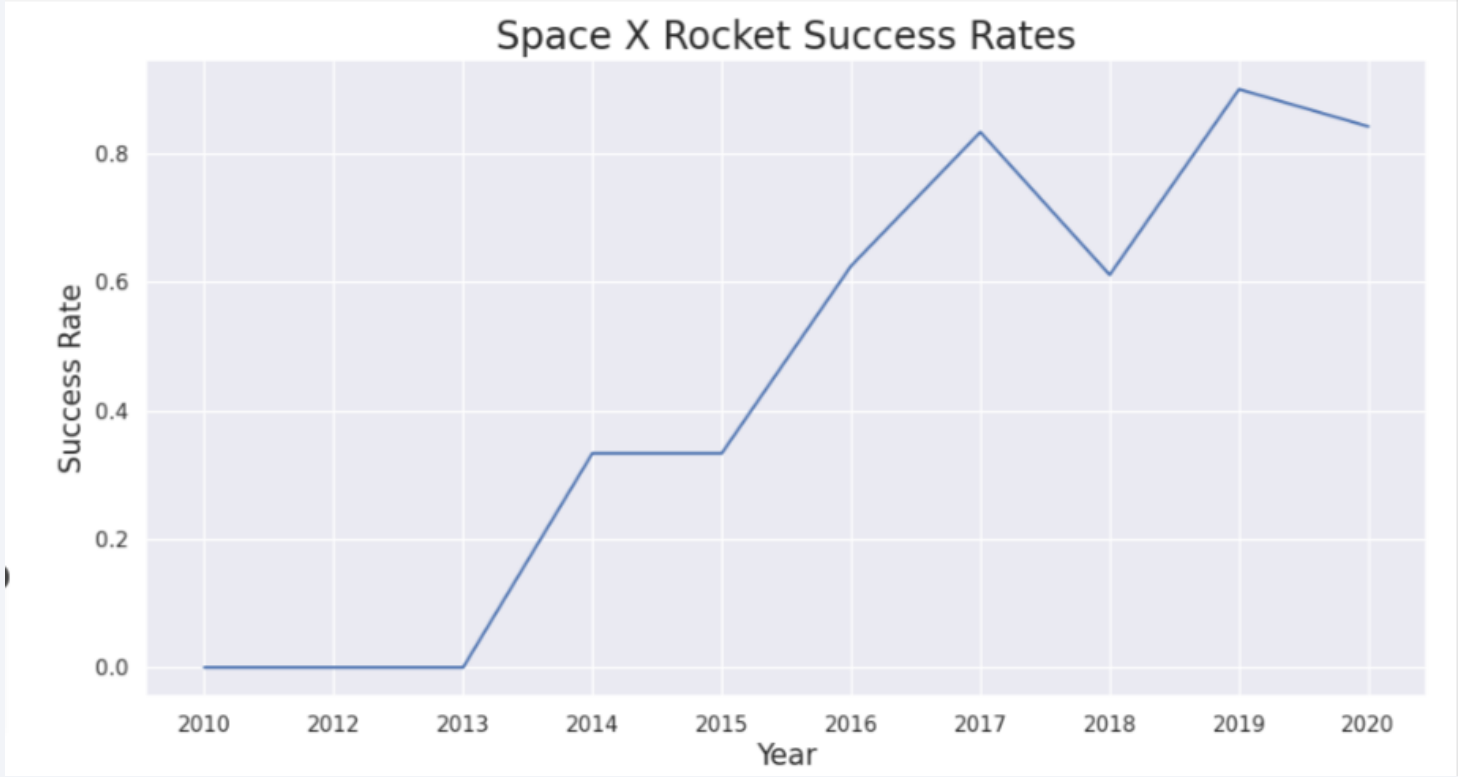
Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type
- Show the screenshot of the scatter plot with explanations



Launch Success Yearly Trend

- Show a line chart of yearly average success rate
- Show the screenshot of the scatter plot with explanations



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here
- DISTINCT keyword is used to return unique values.

```
[8]: %sql SELECT DISTINCT LAUNCH_SITE AS "Launch_Sites" FROM SPACEXTBL;
 * sqlite:///my_data1.db
Done.
```

[8]:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here
- We use LIKE operator to find launch_site starting with CCA and we use LIMIT to reduce the output to 5 records

```
[10]: %sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
 * sqlite:///my_data1.db
Done.
[10]:
```

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Present your query result with a short explanation here
- SUM function is used to calculate the total.

Task 3

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

```
[30]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
 * sqlite:///my_data1.db
Done.
```

[30]:

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here
- AVG function is used to calculate average.

Task 4

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

```
[31]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Booster_Version='F9 v1.1';
 * sqlite:///my_data1.db
Done.
```

[31]:

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here
- MIN function returns the minimum of values.

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

[34]:
```
%sql SELECT MIN(Date) FROM SPACEXTBL WHERE  Landing_Outcome='Success (ground pad)';
```

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

[34]:

MIN(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

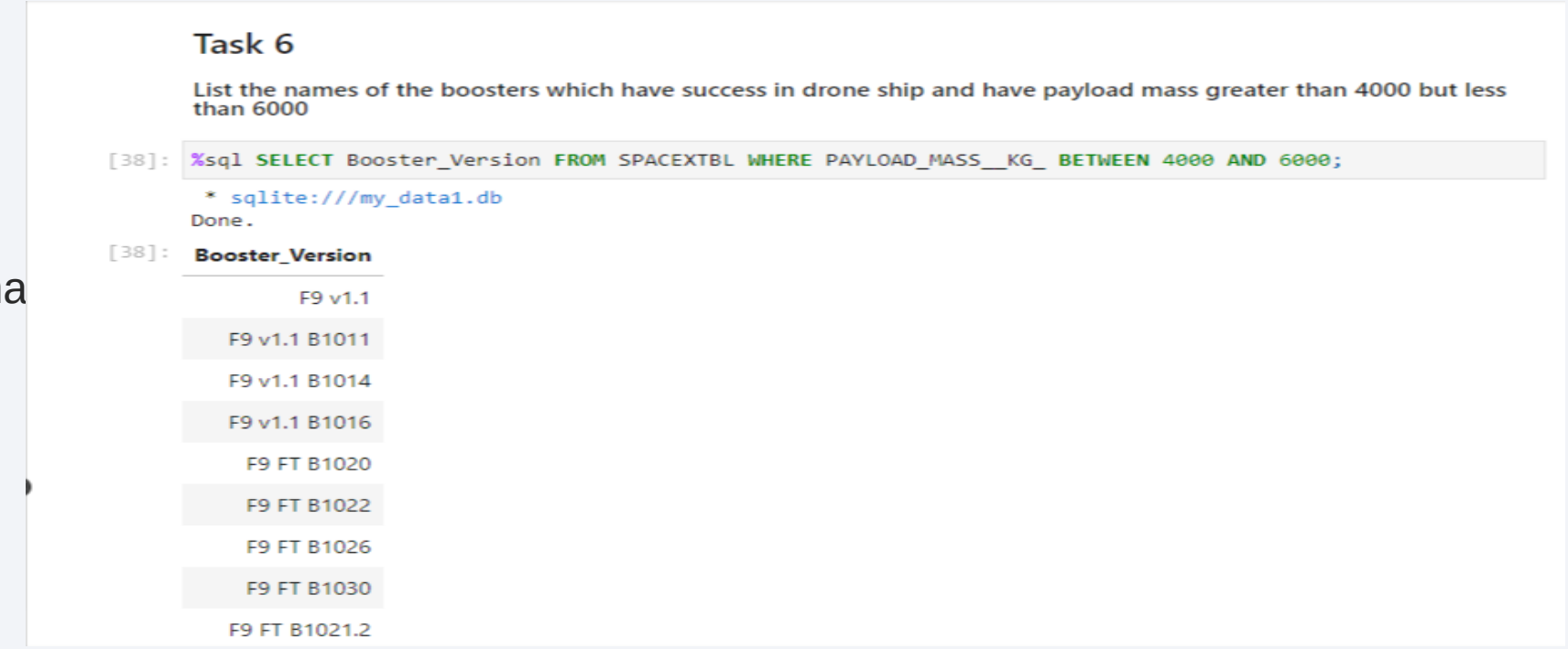
- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- Present your query result with a short explanation here
- BETWEEN funtions

gives values in between

a range.

There are some additiona

-l rows in the output.



Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
[38]: %sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

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

[38]:

Booster_Version
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1014
F9 v1.1 B1016
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1030
F9 FT B1021.2

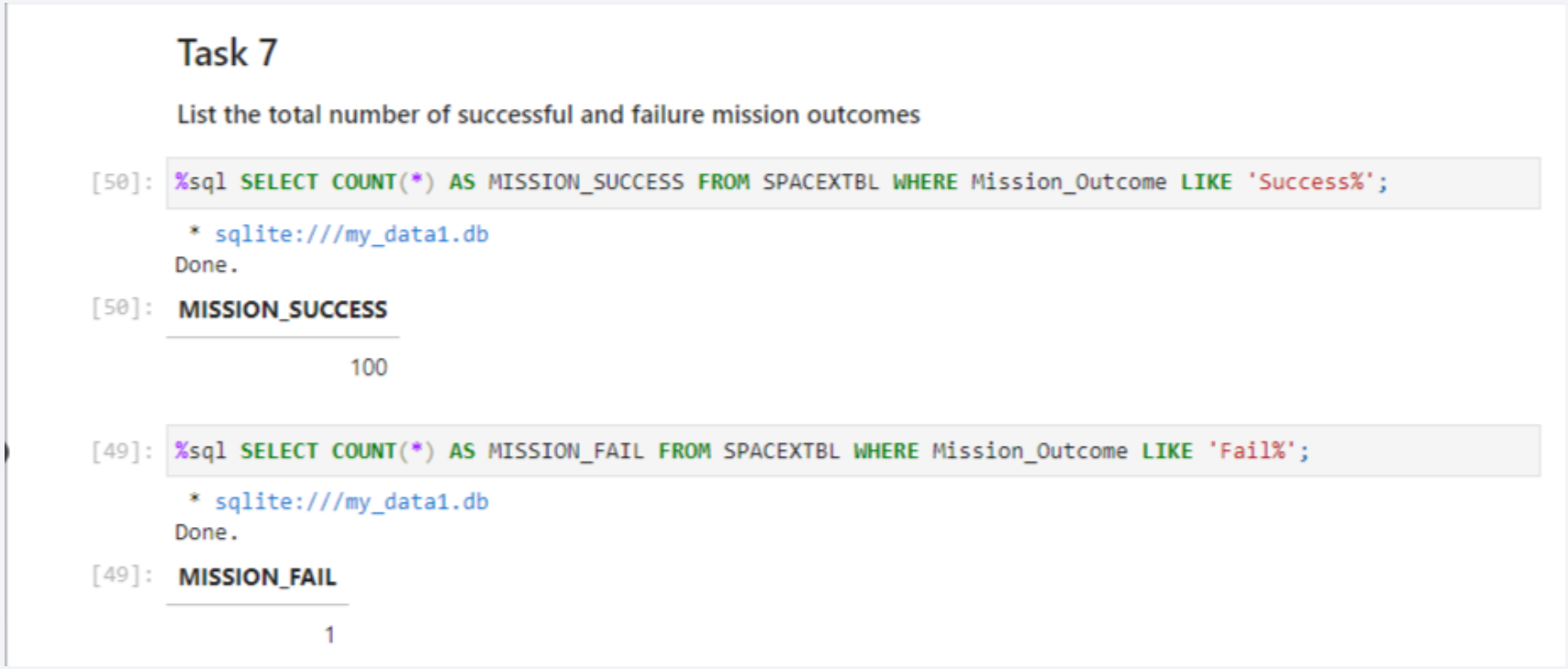
Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here
- COUNT function

returns the number of

Items and LIKE functiom

Compares the similarity.



Task 7

List the total number of successful and failure mission outcomes

```
[50]: %sql SELECT COUNT(*) AS MISSION_SUCCESS FROM SPACEXTBL WHERE Mission_Outcome LIKE 'Success%';
```

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

[50]:

MISSION_SUCCESS
100

```
[49]: %sql SELECT COUNT(*) AS MISSION_FAIL FROM SPACEXTBL WHERE Mission_Outcome LIKE 'Fail%';
```

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

[49]:

MISSION_FAIL
1

Boosters Carried Maximum Payload

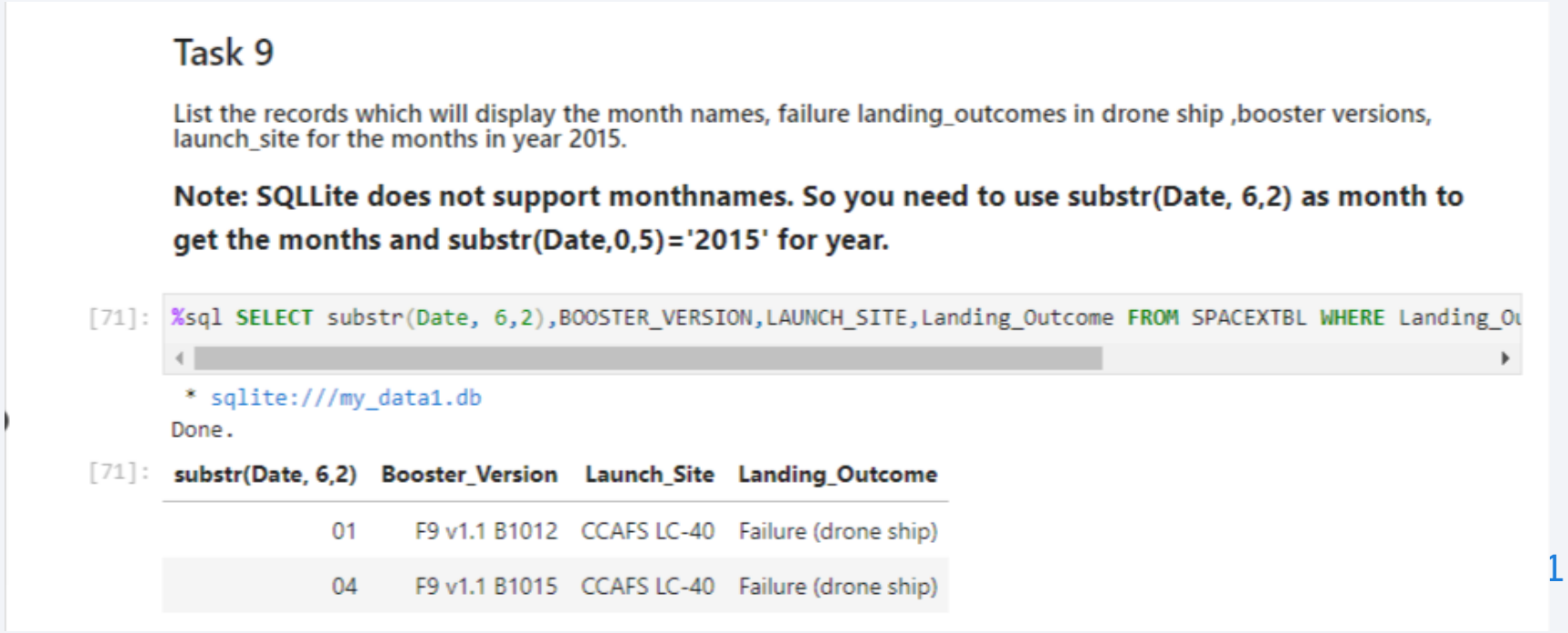
- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here
- SubQuery is used

in this case.

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
- Sunbstr gives the

substring .



Task 9

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, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

[71]:
```
%sql SELECT substr(Date, 6,2),BOOSTER_VERSION,LAUNCH_SITE,Landing_Outcome FROM SPACEXTBL WHERE Landing_Ou
```

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

[71]:

substr(Date, 6,2)	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	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 landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order
- Present your query result with a short explanation here
- The **ORDER BY** keyword sorts

the records in ascending **order by** default.

Task 10

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.

```
[83]: %sql SELECT LANDING_OUTCOME as "Landing Outcome" , RANK() OVER(ORDER BY COUNT(LANDING_OUTCOME) ASC) FROM
```

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

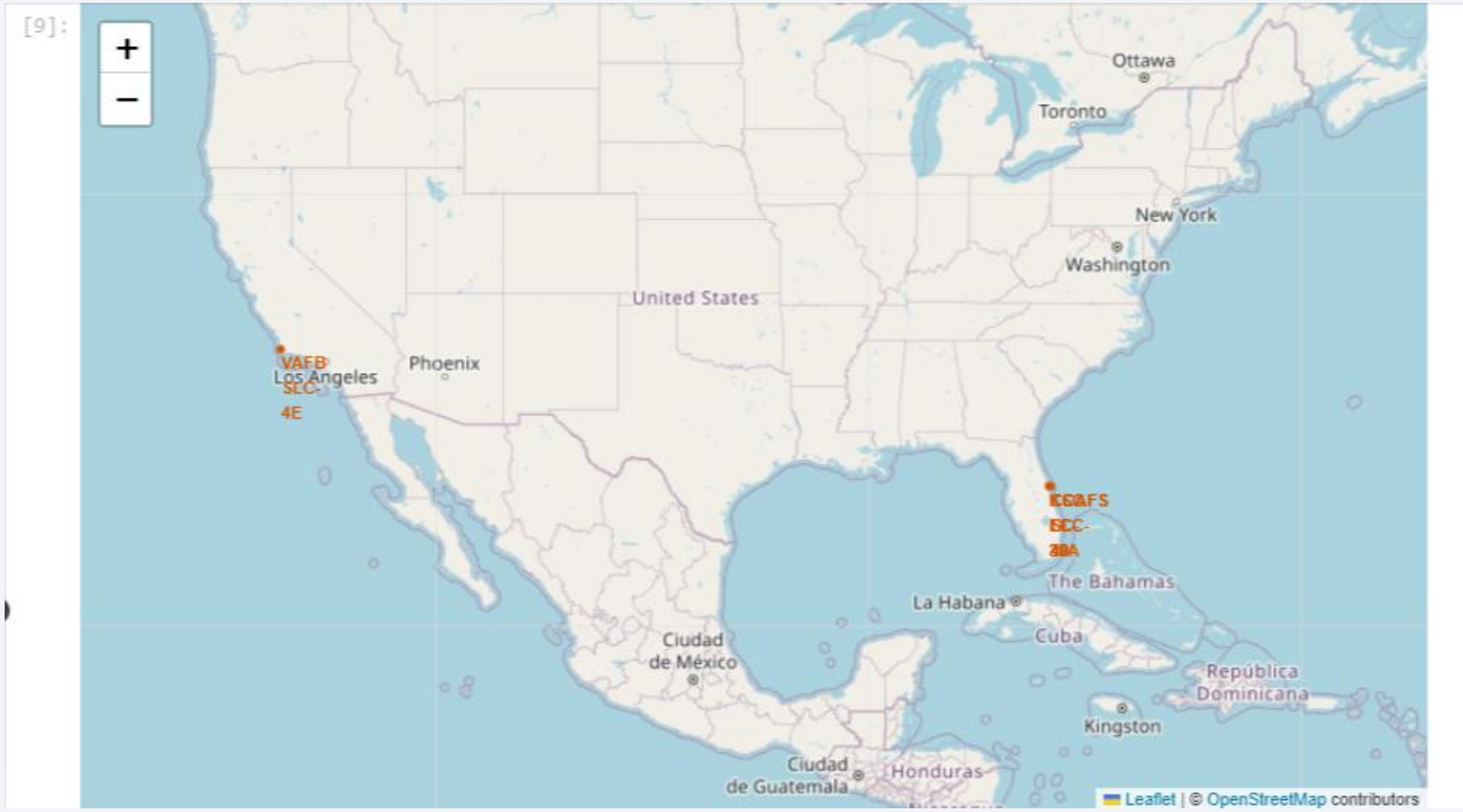
[83]:

Landing Outcome	RANK() OVER(ORDER BY COUNT(LANDING_OUTCOME) ASC)
Precluded (drone ship)	1
Failure (parachute)	2
Uncontrolled (ocean)	2
Controlled (ocean)	4
Success (ground pad)	4
Failure (drone ship)	6
Success (drone ship)	6
No attempt	8


Section 3
Launch Sites
Proximities Analysis

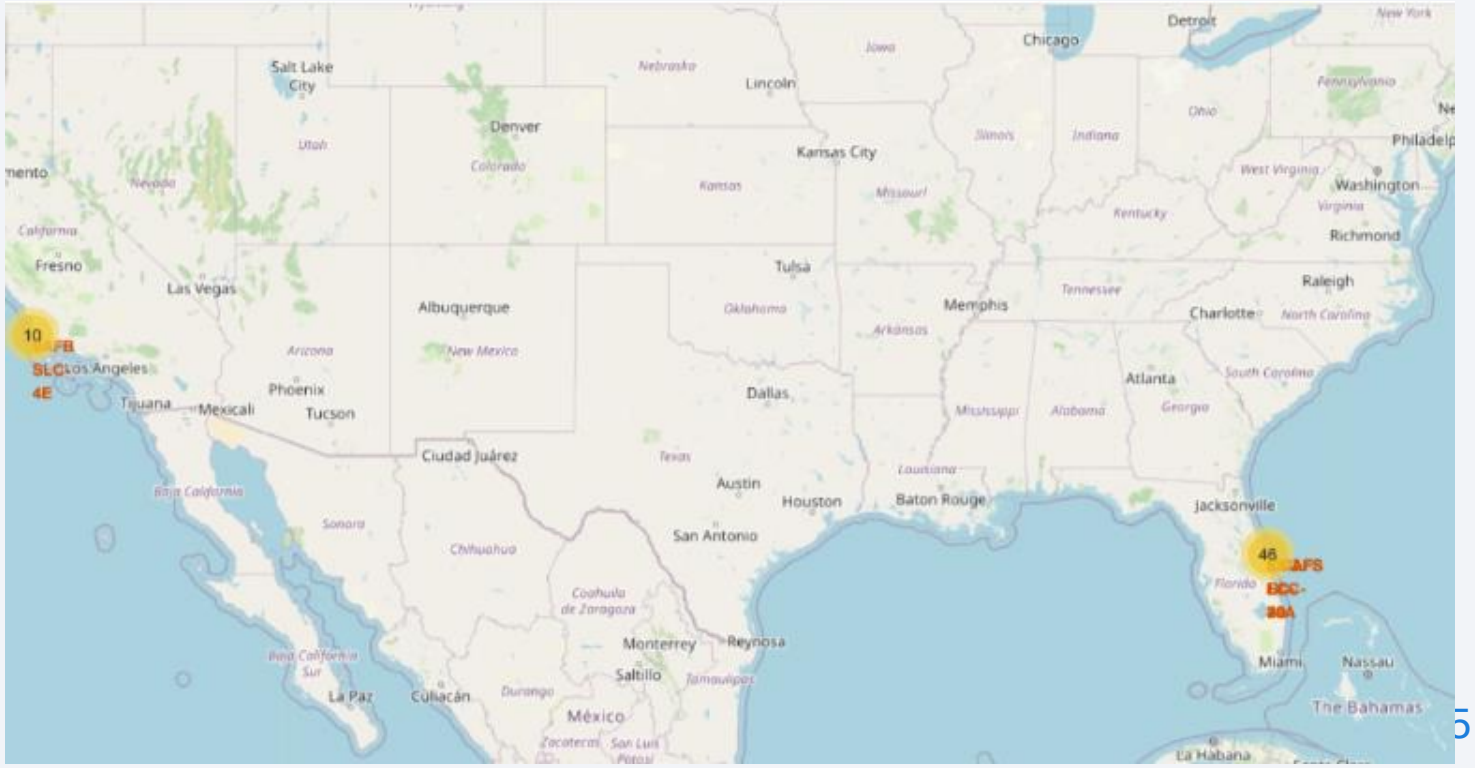
Folium Map Of Launch Sites Locations

The generated folium map that includes all launch sites' location markers on a global map



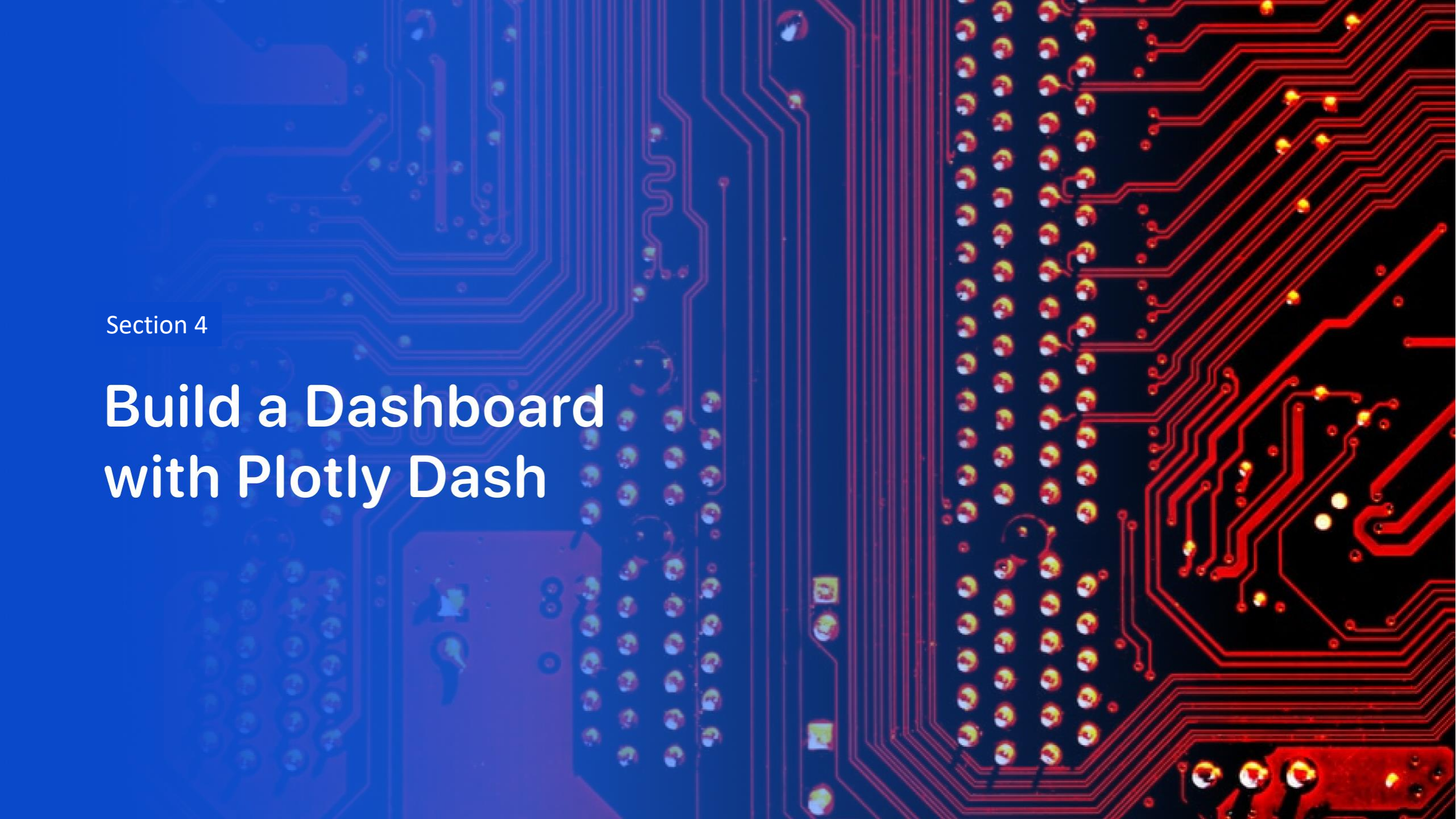
Folium Map - Color Labeled Launch Outcomes

Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map



Folium Map – Launch Site Proximities

The generated folium map of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed


Section 4
Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>

- Replace <Dashboard screenshot 1> title with an appropriate title
- Show the screenshot of launch success count for all sites, in a piechart
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 2>

- Replace <Dashboard screenshot 2> title with an appropriate title
- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 3>

- Replace <Dashboard screenshot 3> title with an appropriate title
- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.

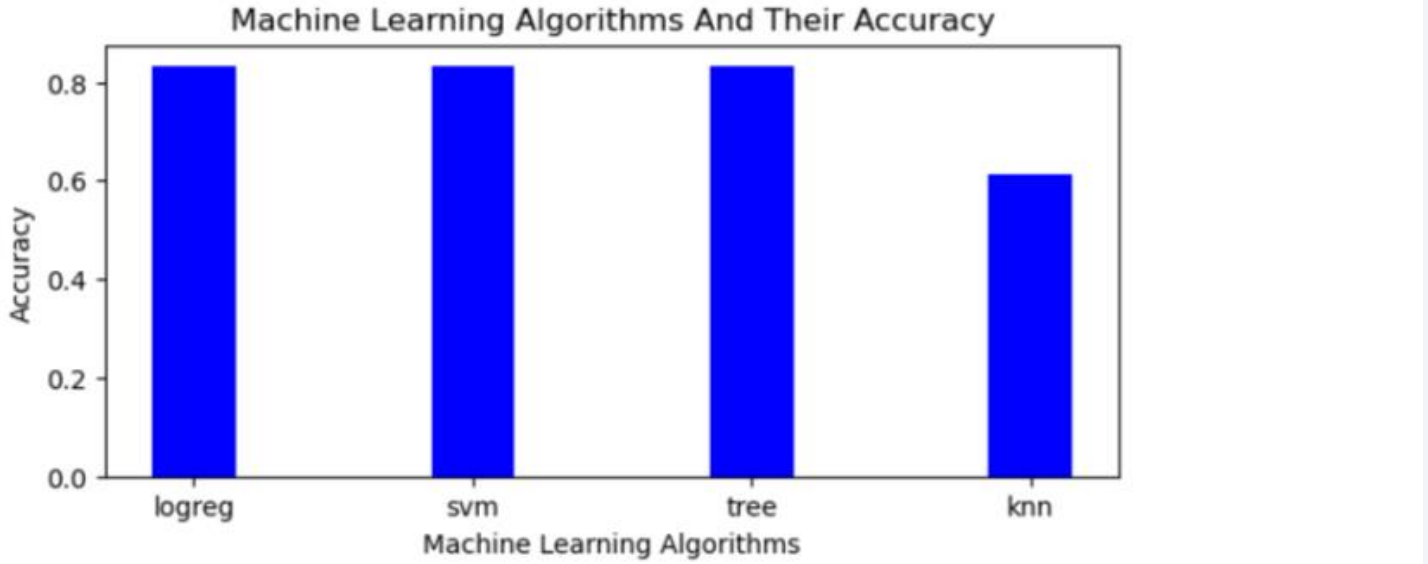
Section 5

Predictive Analysis (Classification)

Classification Accuracy

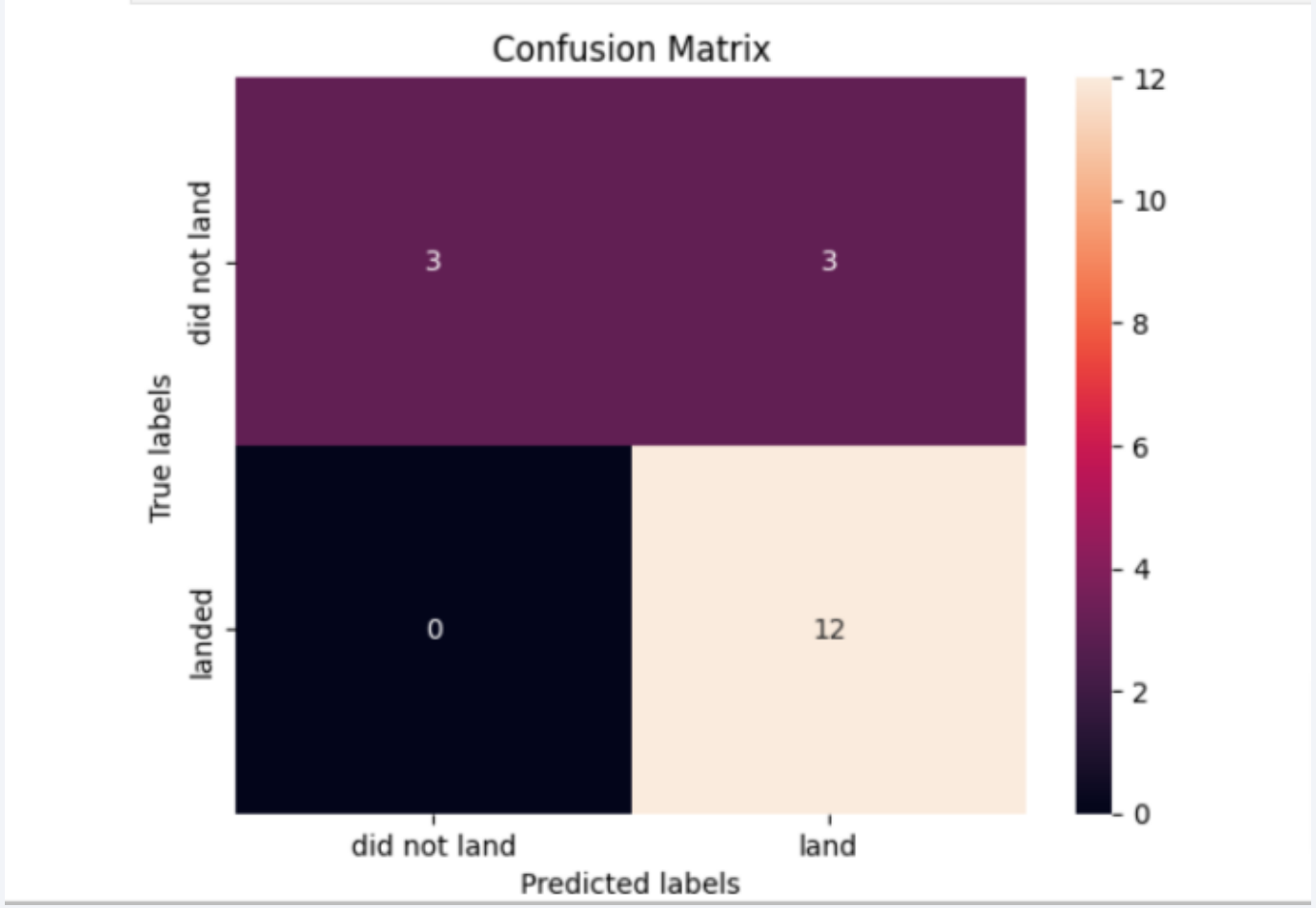
- Visualize the built model accuracy for all built classification models, in a bar chart
- Best Algorithm is Tree with a 0.875.

```
fig = plt.figure(figsize = (7, 3))
plt.bar(['logreg', 'svm', 'tree','knn'],[logreg_cv,svm_cv,tree_cv,knn_cv], color ='blue',
        width = 0.3)
plt.xlabel("Machine Learning Algorithms")
plt.ylabel("Accuracy")
plt.title("Machine Learning Algorithms And Their Accuracy")
plt.show()
```



Confusion Matrix

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



Conclusions

- In conclusion, our machine learning project successfully developed a predictive model for Falcon 9 landings, contributing valuable insights into the factors influencing mission success. The model's predictions and accuracy make it a valuable tool for enhancing the efficiency in prediction of Falcon 9 Landing.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project
- https://github.com/gayathridevi1256/Applied-Data-Science-Capstone

Thank you!