IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

MHD KHAIR SULTAN
April 1, 2023

Outline

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

Executive Summary

- The data has been collected using Wiki pages and SpaceX REST API. We have transformed the data into a clean data set by filtering the features to include the info about Falcon 9 Launches after that we have dealt with the missing data by replacing them with the mean of the data. Next, we applied Exploratory data analysis (EDA) and data interactive visualizations to explore the data to help us choose the best predictive analysis methodology. Next, we standardized the data and then split it into training and testing sets and applied Regression, KNN, and Decision Tree to predict SpaceX Falcon 9 first stage outcome. Finally, we did Grid search to find the hyperparameter tuning and calculate the accuracy for each model then verified it using the test set and drawing the confusion matrix for our predictive models.

- The best predictive analysis methodology is the decision tree with a score equal to 0.8892857142857142.

Introduction

- ❖This project Aims to predict the Flacon 9 Launches. And if the launch can recover the first stage.
- ❖Problems you want to find answers to:
  - ❑ What is the success rate for each launch site?
  - ❑ What features can help to predict whether the Falcon 9 Launch will successfully recover or not?
  - ❑ What is the total payload mass carried by boosters launched by NASA (CRS)?
  - ❑ what is the date for the first successful landing outcome in ground pad was achieved ?
  - ❑ How does the payload mass, Launch sites, and the flights number affect the success rate?
  - ❑ What is the best predictive model that can help us with to solve the problem?

Section 1

Methodology

Methodology

Executive Summary

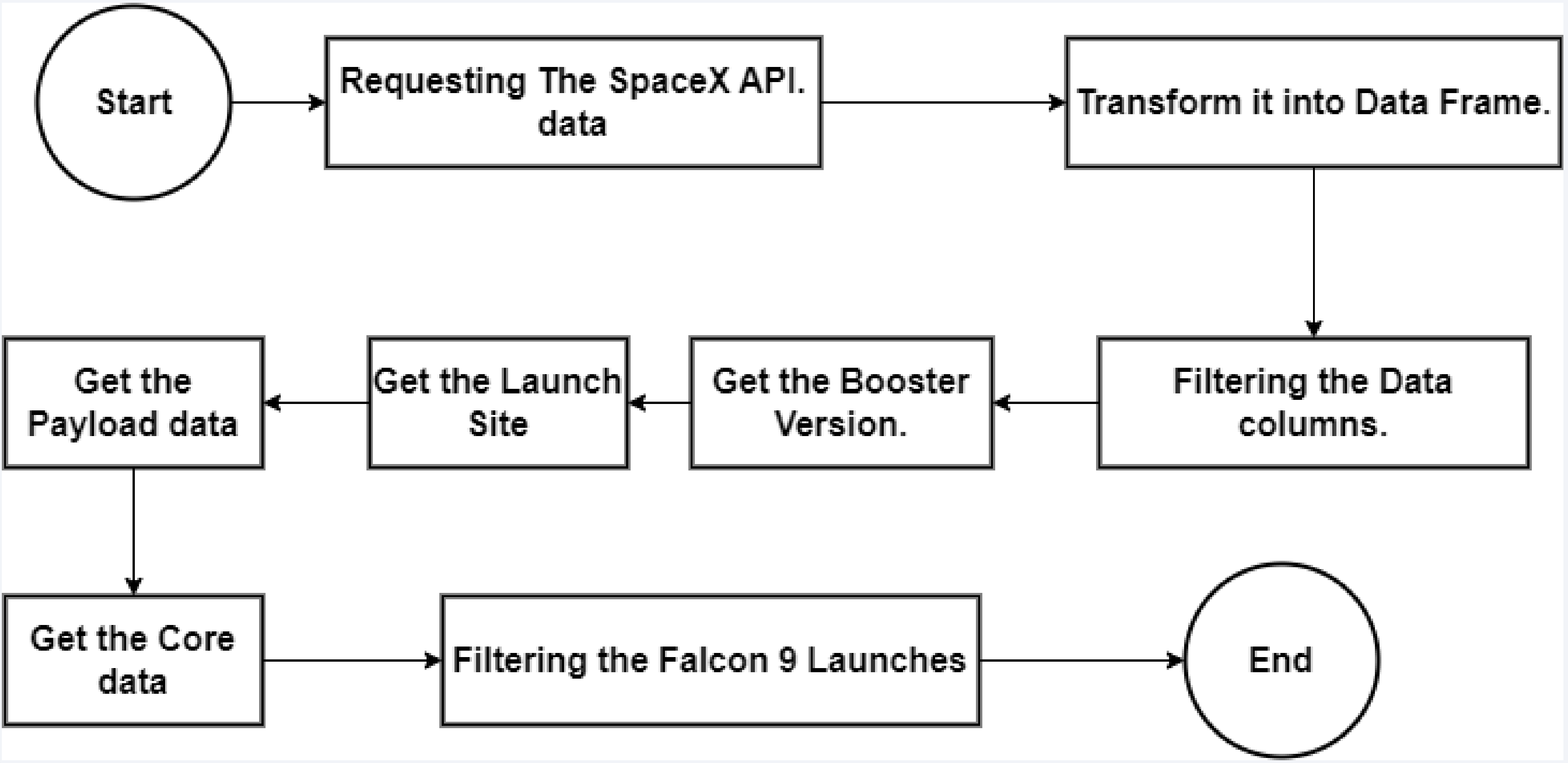
- ❖Data collection methodology:
  - ❑We have collected the data using Wiki pages of Flacon 9 records and SpaceX REST API. We have used BeautifulSoup and Pandas for web scrapping and storing the data in Pandas Data-Frame.
- ❖Perform data wrangling
  - ❑We have replaced the null values with the mean.
- ❖Perform exploratory data analysis (EDA) using visualization and SQL
- ❖Perform interactive visual analytics using Folium and Plotly Dash
- ❖Perform predictive analysis using classification models
  - ❑We standardized the data then split it into training/testing sets then we used Grid search to tune the hyper-parameters then we calculated the accuracy of the models.

Data Collection

- ❖We have collected the data using Wiki pages of Flacon 9 records and SpaceX REST API(https://api.spacexdata.com/v4/). We have used BeautifulSoup and Pandas for web scrapping and storing the data in Pandas Data-Frame then exporting them as CSV files.
- ❖Web Scrapping:
    - ❑Pass the Wiki page`s URL to the requests library then use **BeautifulSoup** to parse the HTML content and extract the targeted tables after that save it to a **CSV** file.
- ❖SpaceX API:
    - ❑Send the Request to the SpaceX API then clean the requested data after that save it to a CSV file.

Data Collection – SpaceX API

- ❖Requesting The SpaceX API.
- ❖Transform it into Data Frame.
- ❖Filtering the Data columns.
- ❖Get the Booster Version.
- ❖Get the Launch Site.
- ❖Get the Payload data.
- ❖Get the Core data.
- ❖Filtering the Falcon 9 Launches

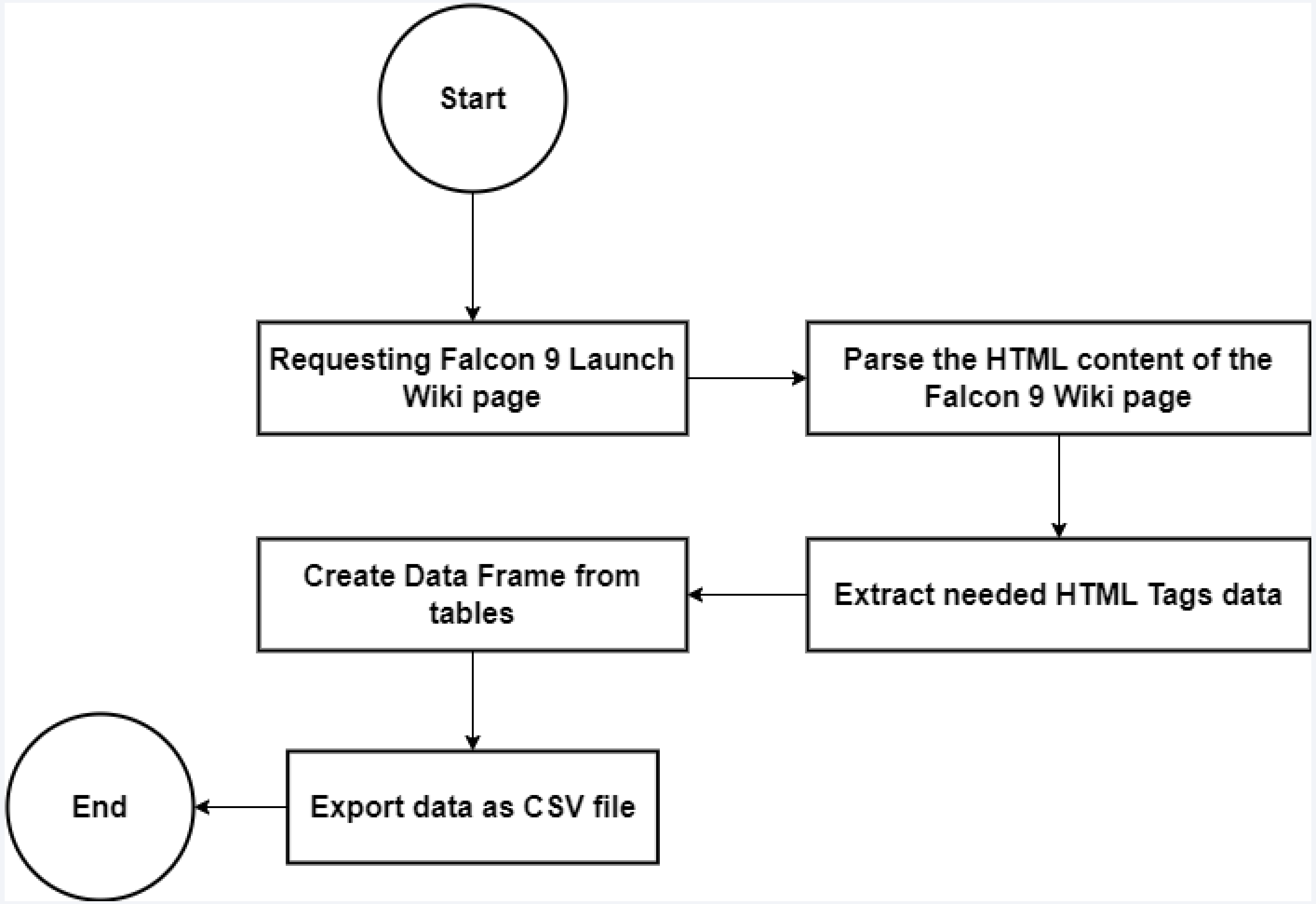

Start
Requesting The SpaceX API. data
Transform it into Data Frame.
Filtering the Data columns.
Get the Booster Version.
Get the Launch Site
Get the Payload data
Get the Core data
Filtering the Falcon 9 Launches
End

- ❖Jupiter Notebook [Link](#) on GitHub.

Data Collection - Scraping

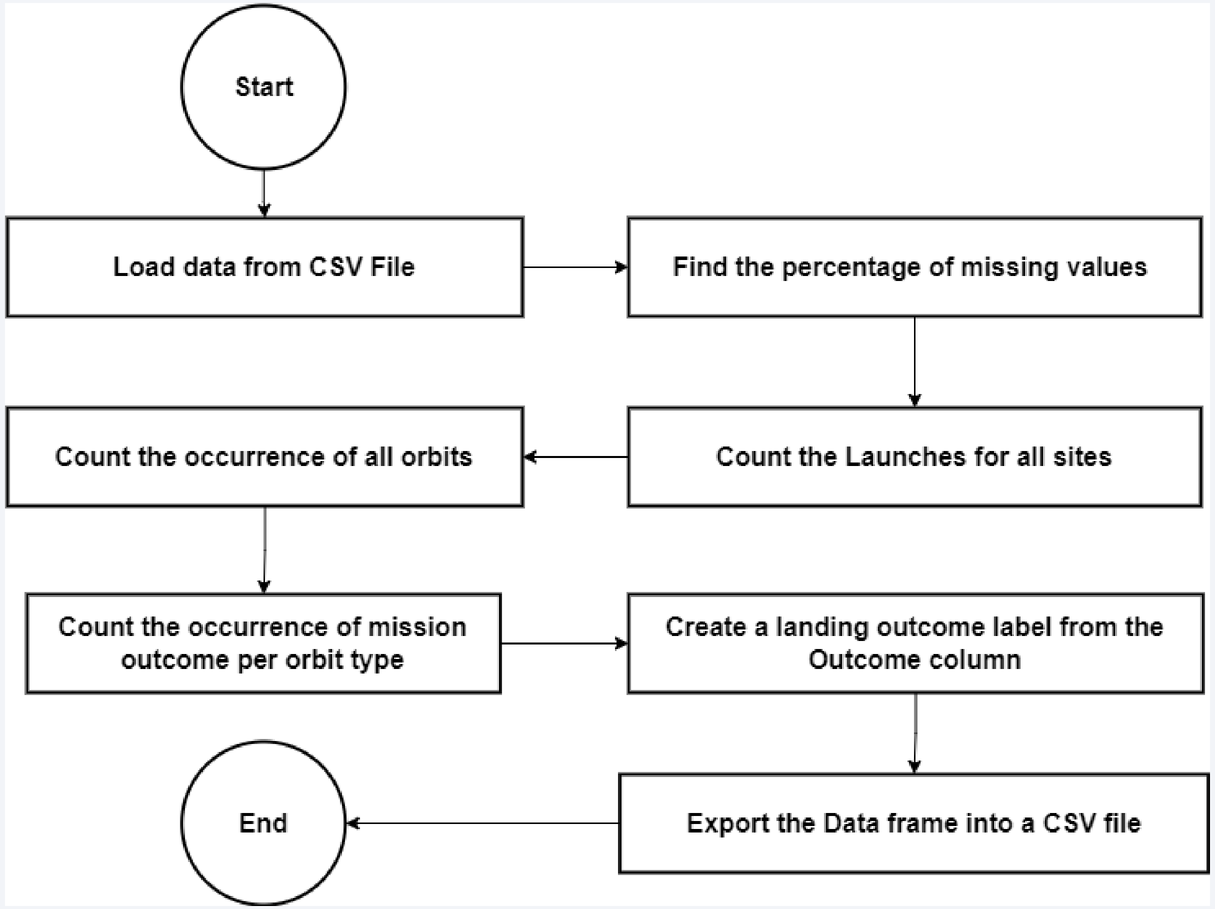
- Requesting Falcon 9 Launch Wiki page.
- Parse the HTML content of the Falcon 9 Wiki page.
- Extract needed HTML Tags data.
- Create a data frame from tables.
- Export data as a CSV file.

- Webscraping Jupyter Notebook Link on GitHub.


Start
Requesting Falcon 9 Launch Wiki page
Parse the HTML content of the Falcon 9 Wiki page
Extract needed HTML Tags data
Create Data Frame from tables
Export data as CSV file
End

Data Wrangling

- ❖We imported the data from the CSV file after that we found the percentage of missing values for all attributes. Then we counted the Launches for all sites, the occurrence of all orbits, and the occurrence of mission outcome per orbit type, then we created the landing outcome label from the Outcome column. Finally, we exported the data set after data wrangling to a new CSV file.
- ❖The Jupyter Notebook link on GitHub.


Start
Load data from CSV File
Find the percentage of missing values
Count the Launches for all sites
Count the occurrence of all orbits
Count the occurrence of mission outcome per orbit type
Create a landing outcome label from the Outcome column
Export the Data frame into a CSV file
End

EDA with Data Visualization

- ❖We have used Scatter plots, line plots, and Bar charts.
  - ❑Scatter plots used to Visualize the relationship between different attributes i.e. Payload and Orbit type.
  - ❑Bar charts used to Visualize the relationship between different attributes i.e. success rate of each orbit type.
  - ❑Line chart is used to Visualize the launch success yearly trend.
  - ❑The Jupyter Notebook link on GitHub.

EDA with SQL

- ❖ Find the names of the unique launch sites in the space mission.
- ❖ Find 5 records where launch sites begin with the string CCA.
- ❖ Find the total payload mass carried by boosters launched by NASA (CRS)
- ❖ Find the average payload mass carried by booster version F9 v1.1.
- ❖ Find the date when the first successful landing outcome in the ground pad was achieved.
- ❖ Find the names of the boosters who have success in drone ships and payload mass between 4000 and 6000.
- ❖ Find the total number of successful and failed mission outcomes
- ❖ Find the booster versions' names that have carried the maximum payload mass.
- ❖ Find the records which will display the month names, failure landing outcomes in drone ship, booster versions, and launch site for the months in the year 2015.
- ❖ Rank the count of successful landing outcomes between the dates 04-06-2010 and 20-03-2017 in descending order.
- ❖ The Jupyter Notebook link on GitHub.

Build an Interactive Map with Folium

❖Map objects:

- ❑ **Circle** to add a highlighted circle area with a text label on a specific coordinate.
- ❑ **Marker** to show icons/text labels on top of a specific coordinate.
- ❑ **MarkerCluster** to Provides Beautiful Animated Marker Clustering functionality for maps.
- ❑ **Icon** to add icons for a map object.
- ❑ **PolyLine** to show linear elements on the map i.e. coastline coordinates and launch site coordinates or to draw a line between the marker to the launch site, etc.

❖The Jupyter Notebook link on GitHub.

Build a Dashboard with Plotly Dash

❖Interactions Components:

- ❑Drop-down is used to make interactions with the dashboard by changing the plots based on the selected option (Launch site`s value or all Launch sites).
- ❑Range Slider to make interactions with the dashboard by changing the plots based on the selected payload range

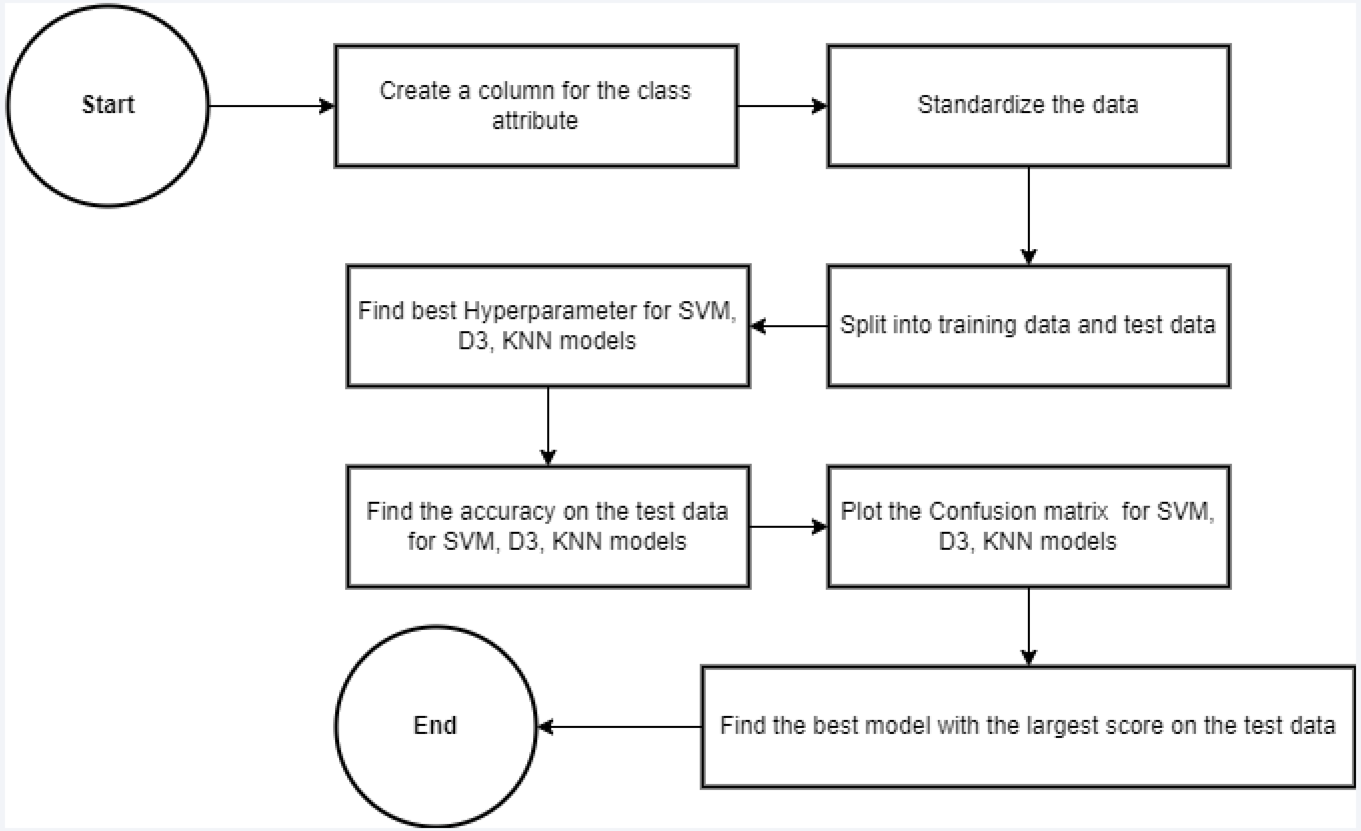
❖Charts and plots:

- ❑Pie chart to visualize the success count for the selected launch site. Or all the total success rate if all value option is selected.
- ❑scatter plot to plot with the x-axis being the payload and the y-axis being the launch outcome.

❖The Jupyter Notebook link on GitHub.

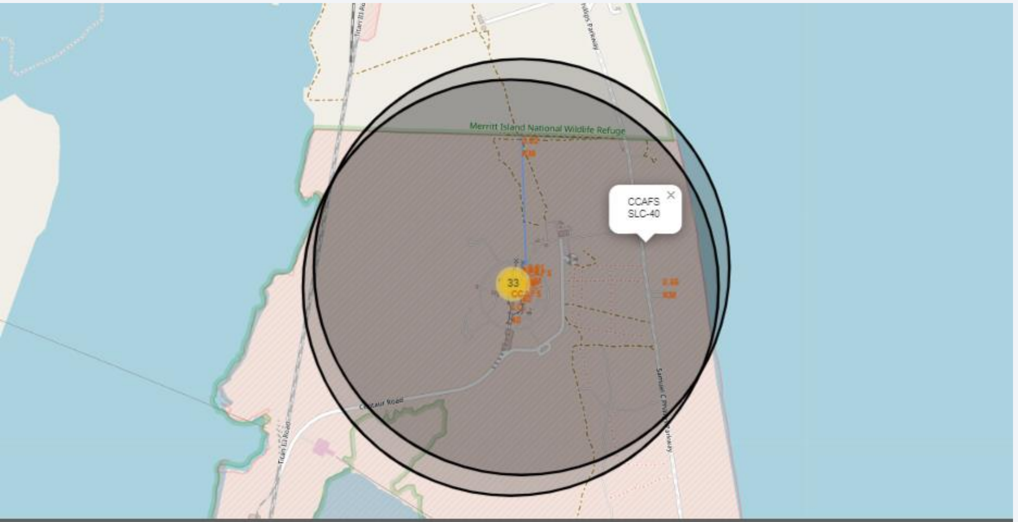
Predictive Analysis (Classification)

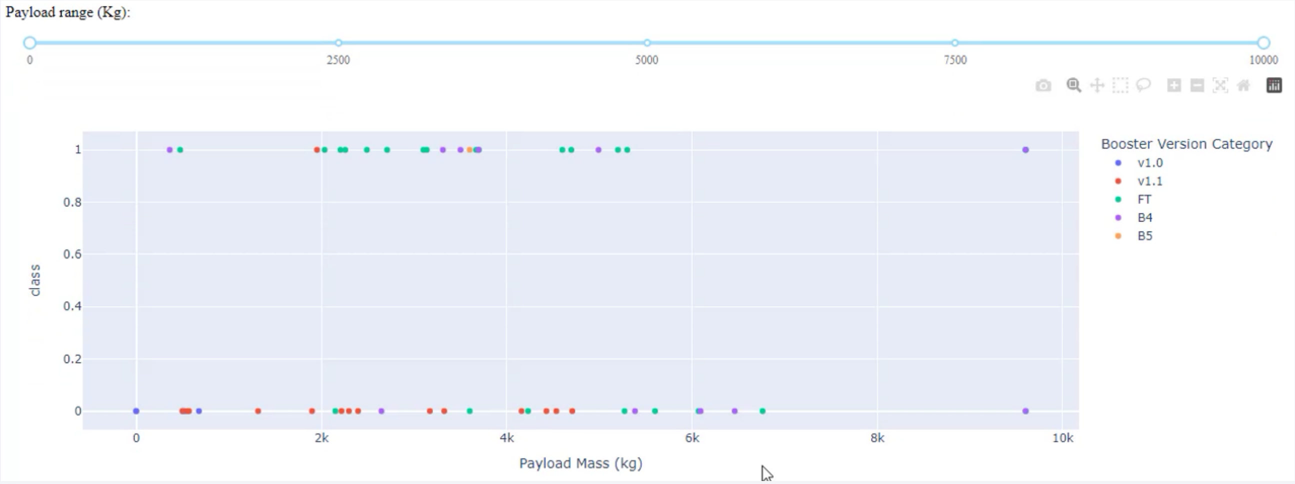
- ❖ Performed EDA and determine Training Labels.
- ❖ Created the class column which represents the target attribute of our dataset.
- ❖ Standardized the data to speed up the training process.
- ❖ Split the dataset into training and testing sets.
- ❖ Used gird search to find the best Hyperparameter for the SVM, D3, and KNN algorithms.
- ❖ Examined all the models and we have chosen the model with the best accuracy and score among all models.
- ❖ The Jupyter Notebook link on GitHub.


Start
Create a column for the class attribute
Standardize the data
Split into training data and test data
Find best Hyperparameter for SVM, D3, KNN models
Find the accuracy on the test data for SVM, D3, KNN models
Plot the Confusion matrix for SVM, D3, KNN models
Find the best model with the largest score on the test data
End

Results

- ❖ **EDA**
  - ❑ Different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E have a success rate of 77%.
  - ❑ The almost 100% success rate orbit types are ES-L1, GEO, HEO, and SSO.
  - ❑ Success rate kept increasing from the year 2013 to 2020.
- ❖ **Predictive analysis:** The decision Tree model is the best model with an accuracy equal to 0.8892857142857142.

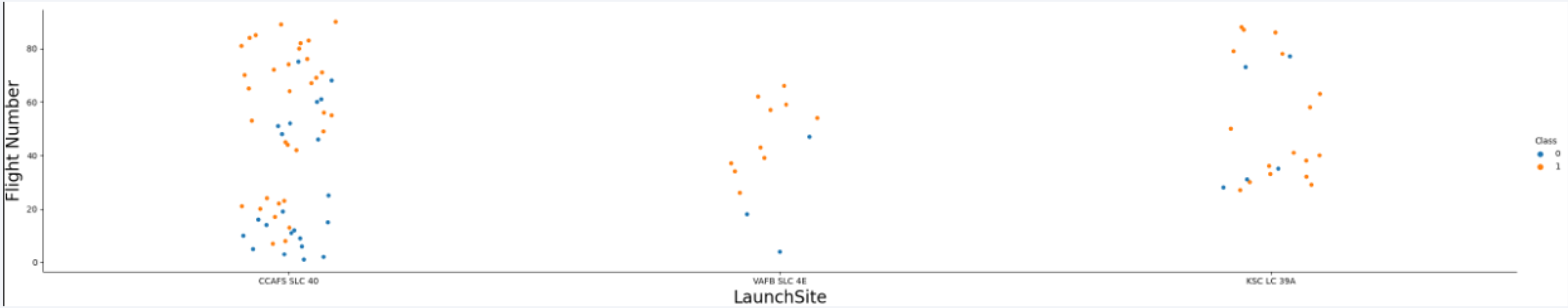




Section 2

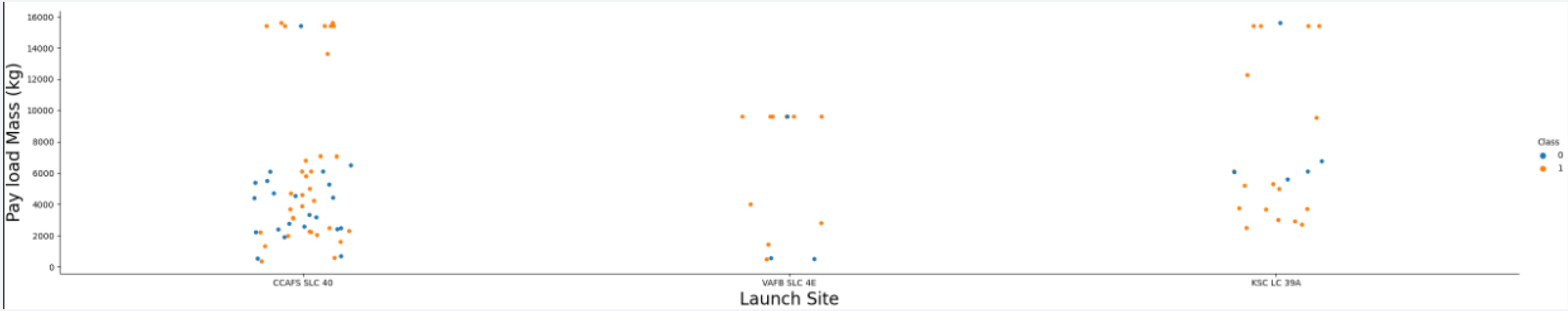
Insights drawn from EDA

Flight Number vs. Launch Site



- ❖ **CCAFS SLC 40** has the maximum flight number overall Launch sites for Falcon 9 Launches.
- ❖ **CCAFS SLC 40** has 55 flight numbers with 22 failures and 33 successes.
- ❖ **VAFB SLC 4E** has the minimum flight number overall Launch sites for Falcon 9 Launches.
- ❖ **VAFB SLC 4E** has 13 flight numbers with 3 failures and 10 successes.

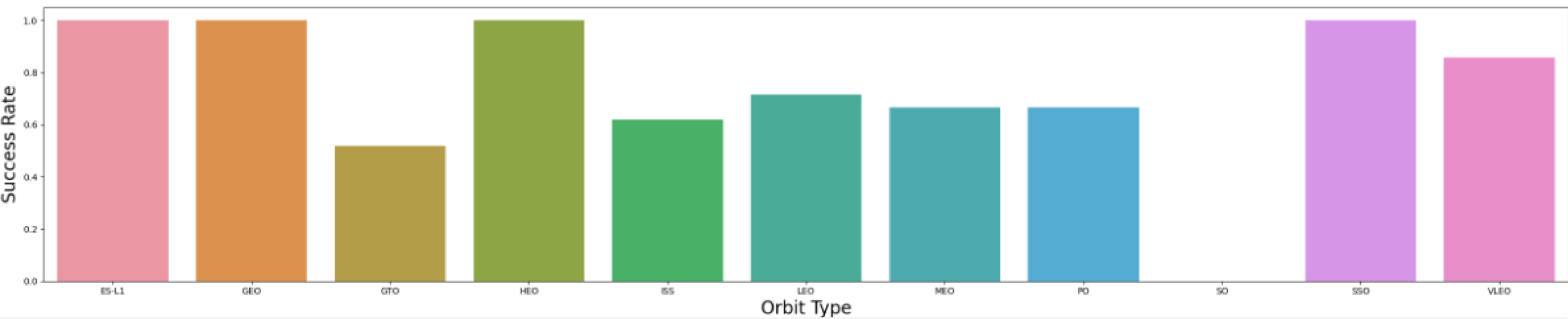
Payload vs. Launch Site



❖The payload mass that is greater or equal to 10k(kg) will have more successful launches.

❖**VAFB SLC 4E** has no launches if the payload mass Is greater or equal to 10k (kg).

Success Rate vs. Orbit Type



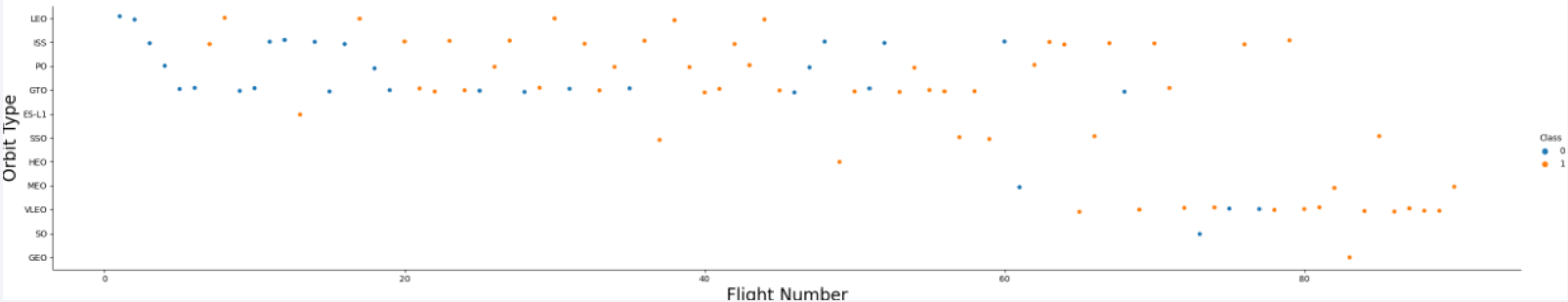
❖ES-L1, GEO, and SSD Orbit types are equal in success rates.

❖ES-L1, GEO, and SSD Orbit types have the maximum values in success rates.

❖GTO and SO Orbit types have the lowest success rate value among all Orbit types.

❖SO Orbit type has a 0% success rate.

Flight Number vs. Orbit Type



❖GEO, ES-L1, and HEO Orbit types have only one launch and it was a successful launch.

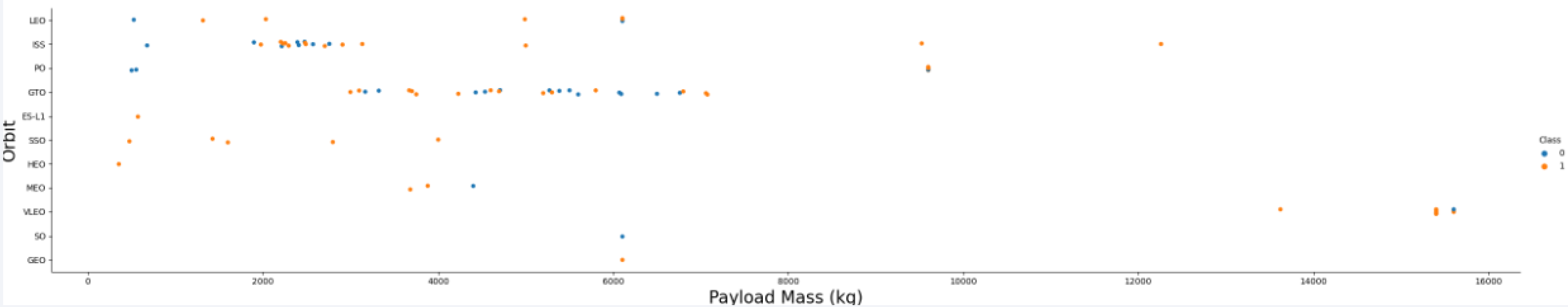
❖LEO Orbit type success more when we got more flights.

❖SSO Orbit type has never failed and its flight numbers are greater than 20.

❖SO Orbit type has only one launch and it was a failed launch.

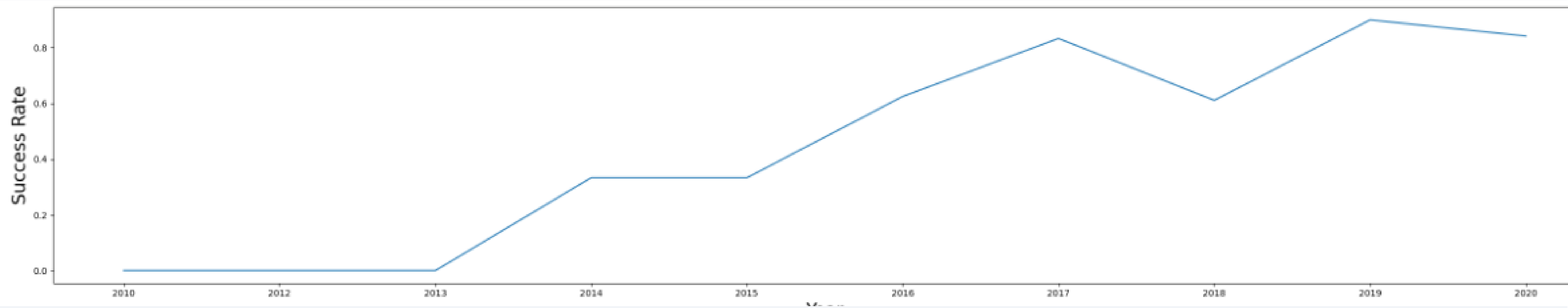
❖VLEO Orbit type has success in all the launches when the flight number is >= 80.

Payload vs. Orbit Type



- ❖GEO, ES-L1, and HEO Orbit types have only one launch and it was a successful launch.
- ❖SO Orbit type has only one launch and it was a failed launch.
- ❖The lower the payload mass goes the higher the failure rate will be for Orbit types.
- ❖SSO, ES-L1, GEO, and HEO Orbit types always have successful launches regardless of the payload mass.
- ❖Only VELO, PO, and ISS Orbit types have launches when the payload mass is greater than 8k kg.

Launch Success Yearly Trend



❖The success rate keeps increasing from the year 2013 to 2020.

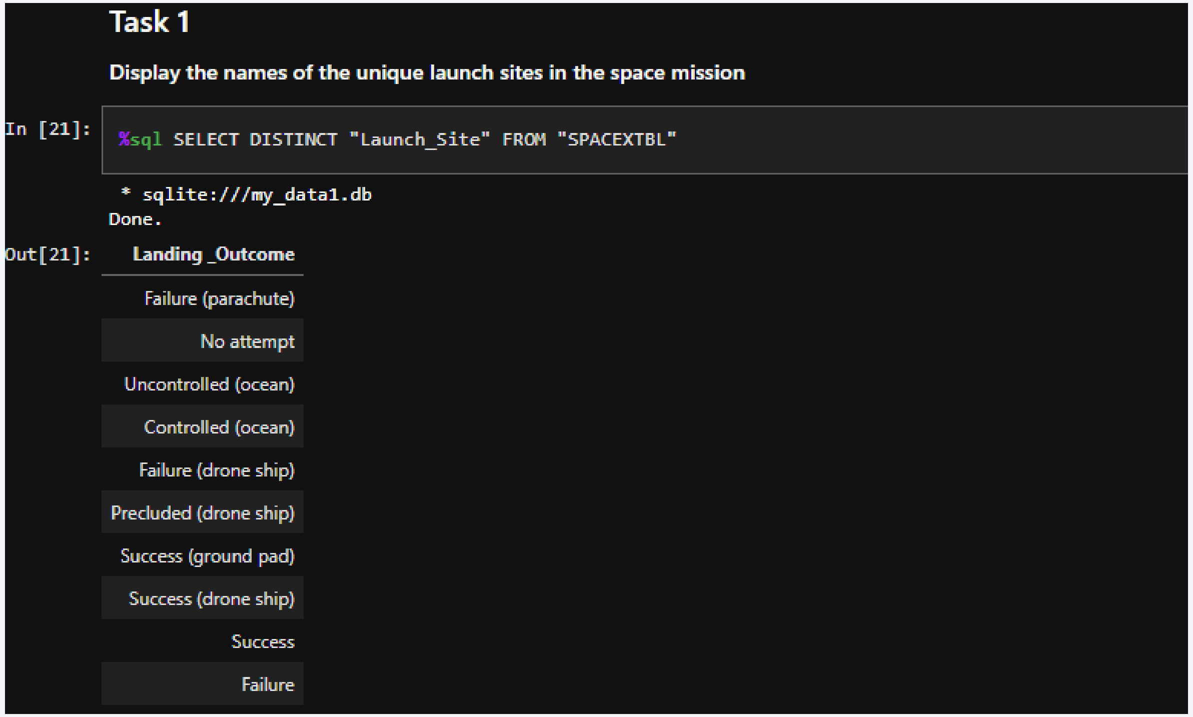
❖The success rate reaches its maximum value in the year 2019.

❖The success rate progress decreased in the years 2015, 2018, and 2020.

❖The success rate was 0 in the years 2010 and 2012.

❖All the years that are greater than 2015 have a success rate greater than 40%.

All Launch Site Names



Task 1

Display the names of the unique launch sites in the space mission

```
In [21]: %sql SELECT DISTINCT "Launch_Site" FROM "SPACEXTBL"
```

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

Out[21]:

Landing _Outcome
Failure (parachute)
No attempt
Uncontrolled (ocean)
Controlled (ocean)
Failure (drone ship)
Precluded (drone ship)
Success (ground pad)
Success (drone ship)
Success
Failure

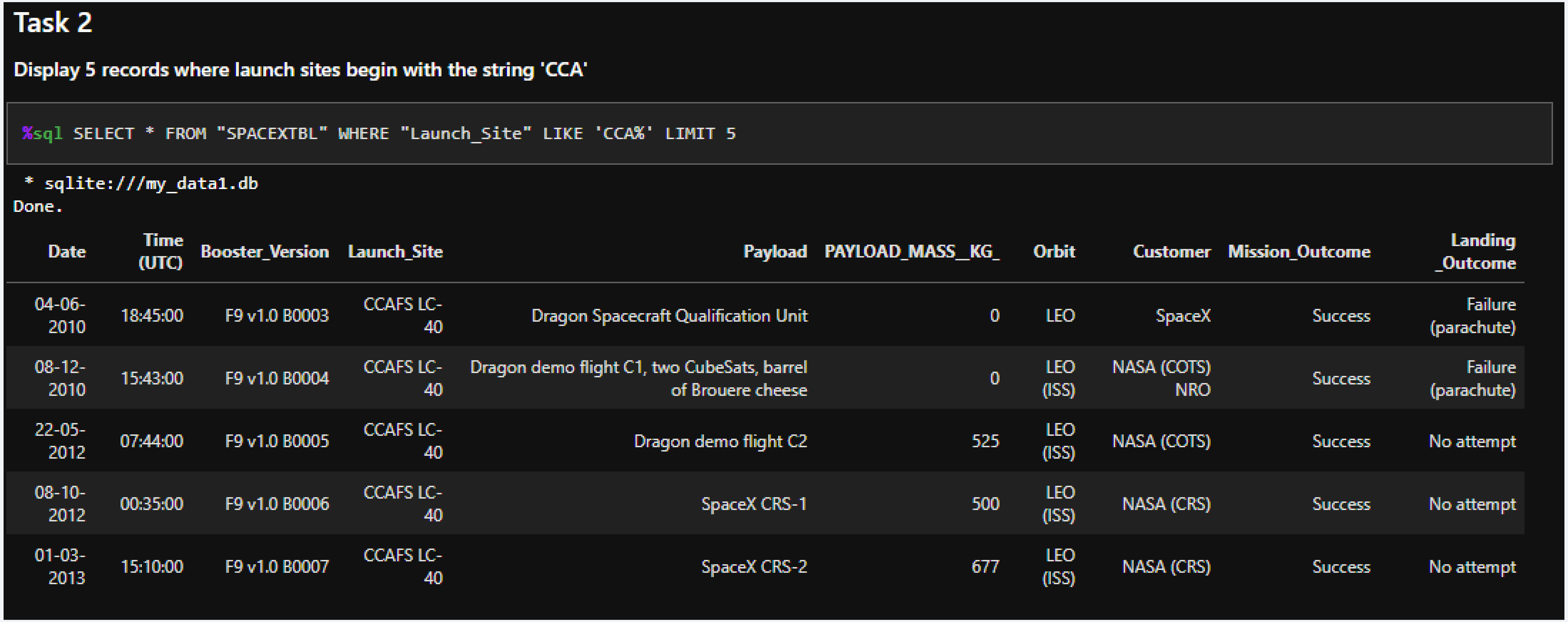
❖%sql SELECT DISTINCT "Launch_Site" FROM "SPACEXTBL“

❖We will select all the unique values from the SPACXTBL using the Distinct keyword and specify the Launch site as the output of the SQL select command.

❖Unique Launch sites names:

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

Launch Site Names Begin with 'CCA'



Task 2

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

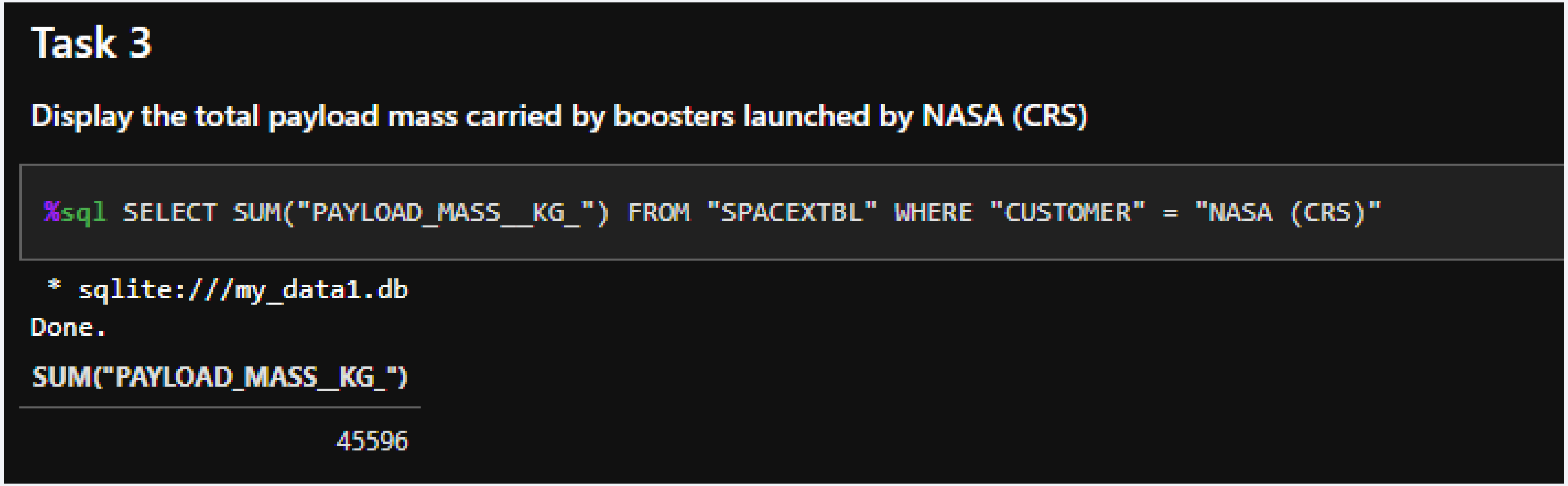
```
%sql SELECT * FROM "SPACEXTBL" WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	15:43:00	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- ❖ %sql SELECT * FROM "SPACEXTBL" WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5
- ❖ We are selecting all columns from the SPACEXTBL database with the condition that the launch site name starts with CCA then we will only limit the output of the command using the limit SQL function.

Total Payload Mass



Task 3

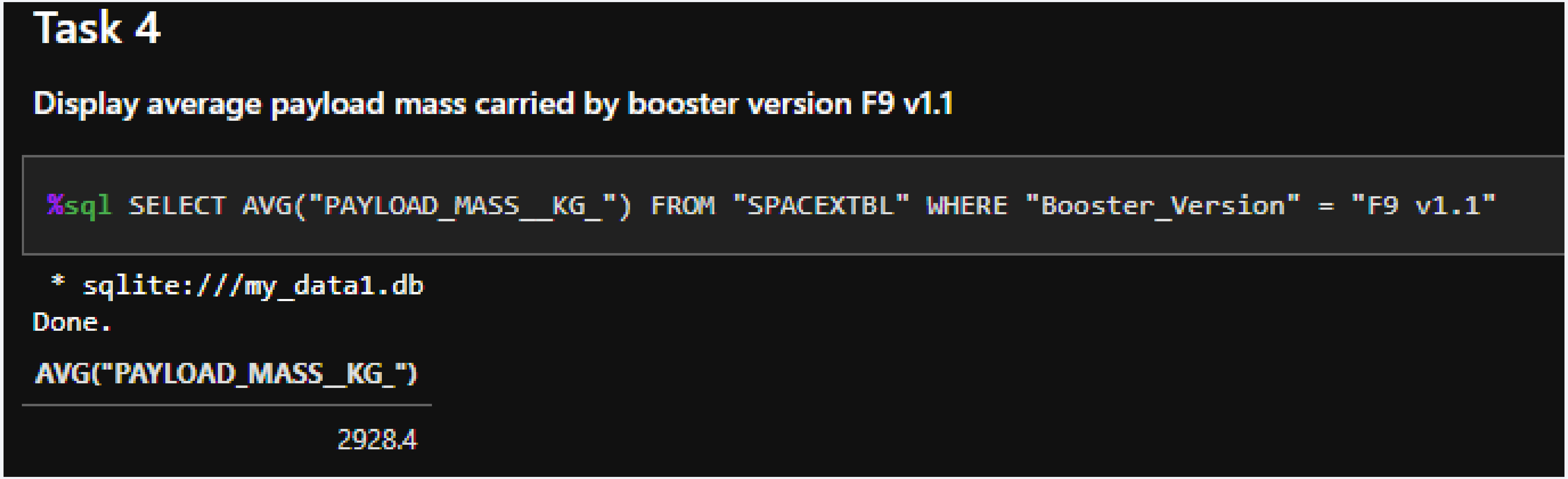
Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM "SPACEXTBL" WHERE "CUSTOMER" = "NASA (CRS)"
```

```
 * sqlite:///my_data1.db
Done.
SUM("PAYLOAD_MASS__KG_")
                   45596
```

- ❖%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM "SPACEXTBL" WHERE "CUSTOMER" = "NASA (CRS)“
- ❖We sum all the values of the Payload mass column from the SPACEXTBL table then we will look for the customer value that equals NASA(CRS).
- ❖The answer is 45596 KG.

Average Payload Mass by F9 v1.1



Task 4

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"
```

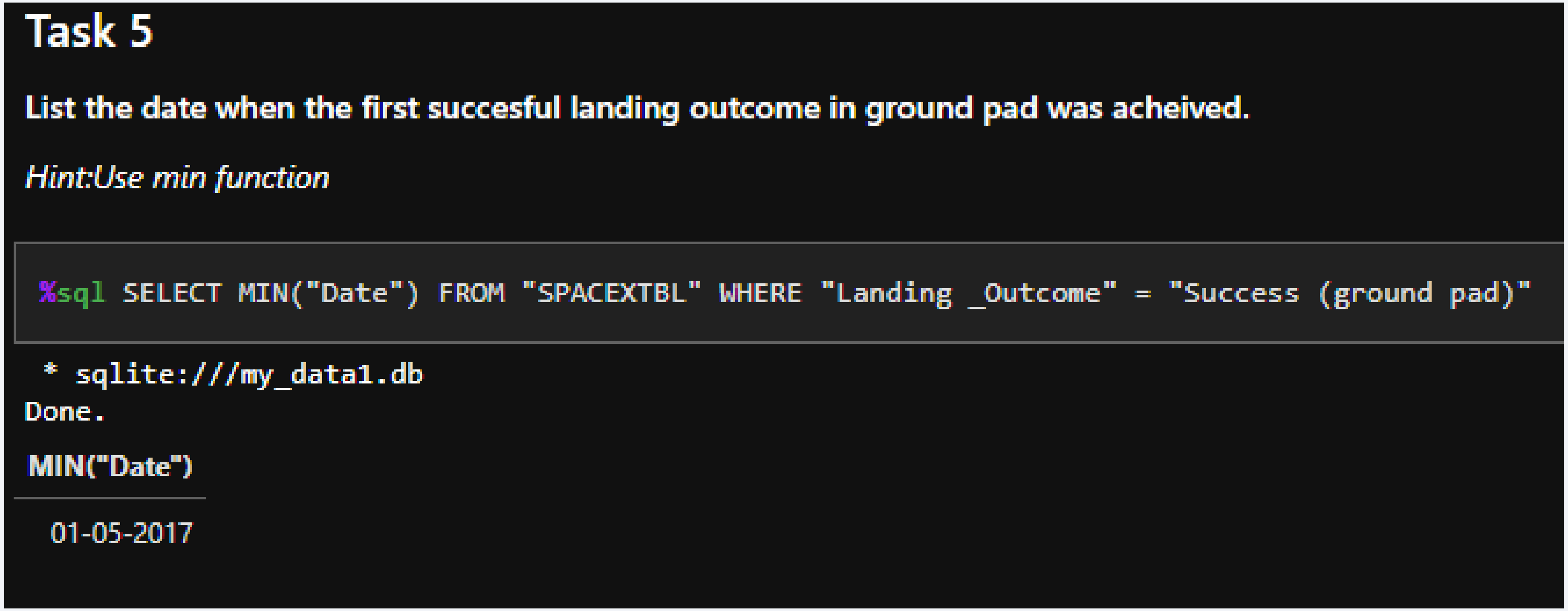
```
 * sqlite:///my_data1.db
Done.
AVG("PAYLOAD_MASS__KG_")
------------------------
                  2928.4
```

- %sql SELECT AVG("PAYLOAD_MASS__KG_") FROM "SPACEXTBL" WHERE "Booster_Version" = "F9 v1.1“

- We will calculate the Average of the payload mass using the AVG SQL function for all the Booster_Version that are equal to F9v1.1.

- The answer is 2928.4.

First Successful Ground Landing Date



Task 5

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

Hint:Use min function

```
%sql SELECT MIN("Date") FROM "SPACEXTBL" WHERE "Landing _Outcome" = "Success (ground pad)"
```

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

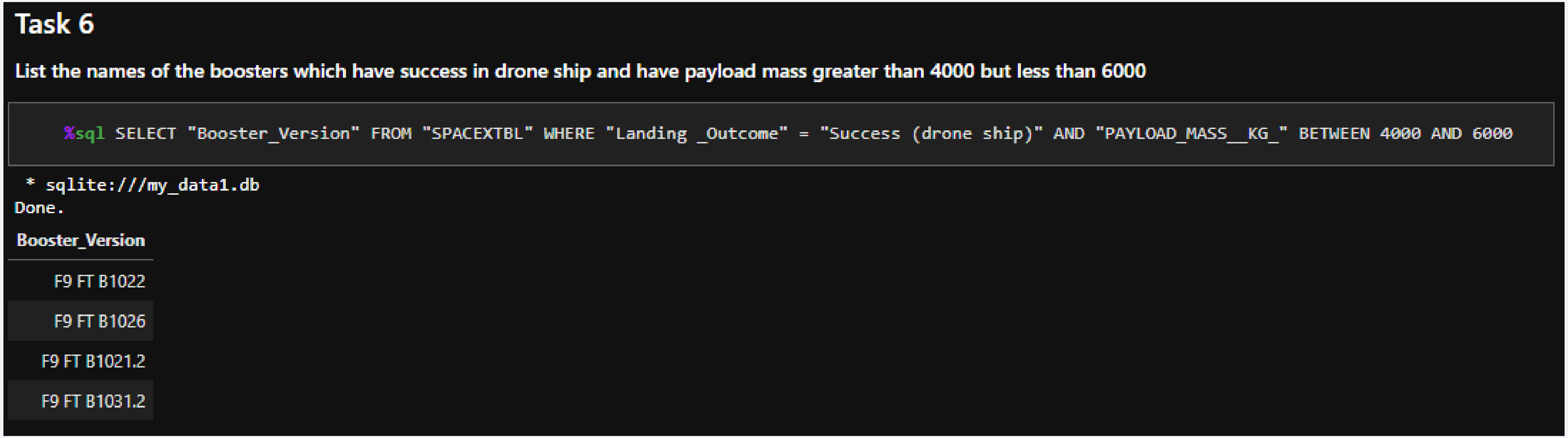
MIN("Date")
01-05-2017

❖%sql SELECT MIN("Date") FROM "SPACEXTBL" WHERE "Landing _Outcome" = "Success (ground pad)“

❖We used the Min function on the Date column and checked the landing outcome equal Success (ground pad).

❖The answer is 01-05-2017.

Successful Drone Ship Landing with Payload between 4000 and 6000



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

```
%sql SELECT "Booster_Version" FROM "SPACEXTBL" WHERE "Landing _Outcome" = "Success (drone ship)" AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000
```

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

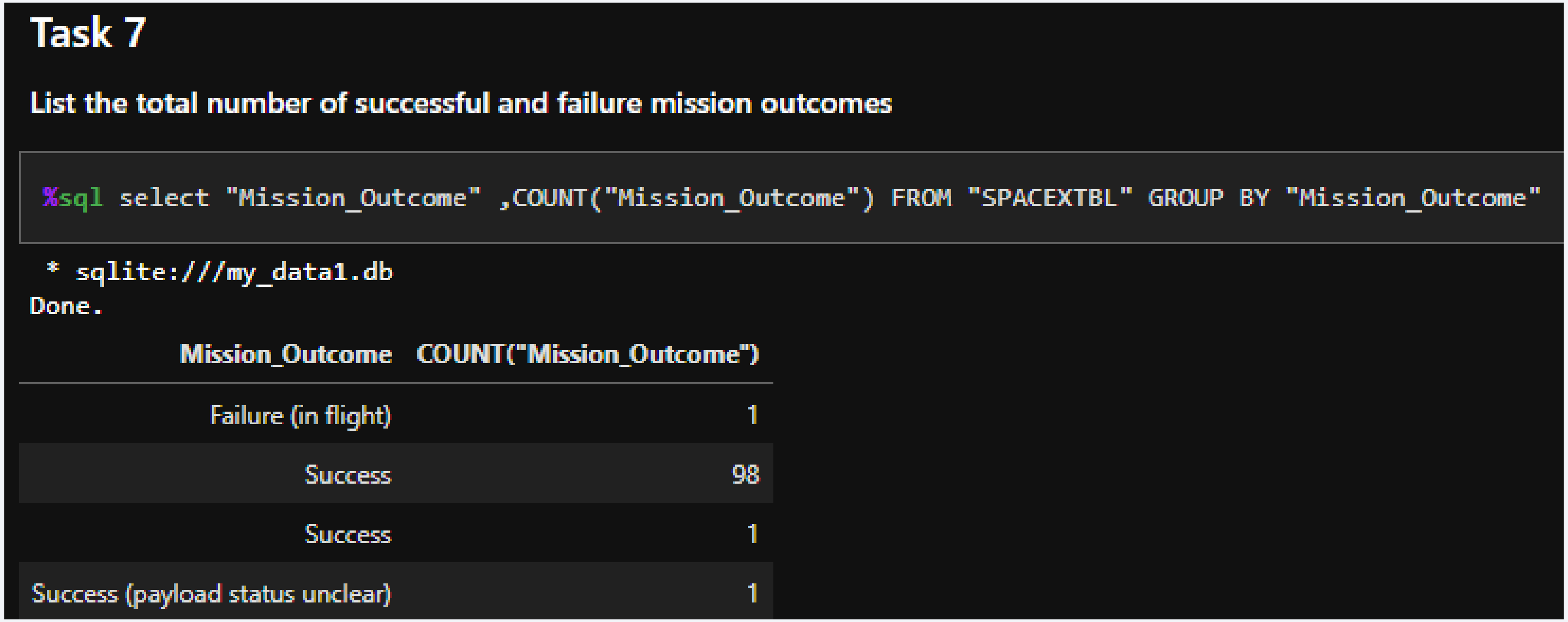
Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

- ❖ %sql SELECT "Booster_Version" FROM "SPACEXTBL" WHERE "Landing _Outcome" = "Success (drone ship)" AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000

- ❖We have looked at the Booster version column and checked the landing outcome value to match the success (drone ship) value and the payload mass value that belongs to the range from 4k to 6k.

- ❖The answer is: F9 FT B1022, F9 FT B1026, F9 FT B1021.2, AND F9 FT B1031.2.

Total Number of Successful and Failure Mission Outcomes



Task 7

List the total number of successful and failure mission outcomes

```
%sql select "Mission_Outcome" ,COUNT("Mission_Outcome") FROM "SPACEXTBL" GROUP BY "Mission_Outcome"
```

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

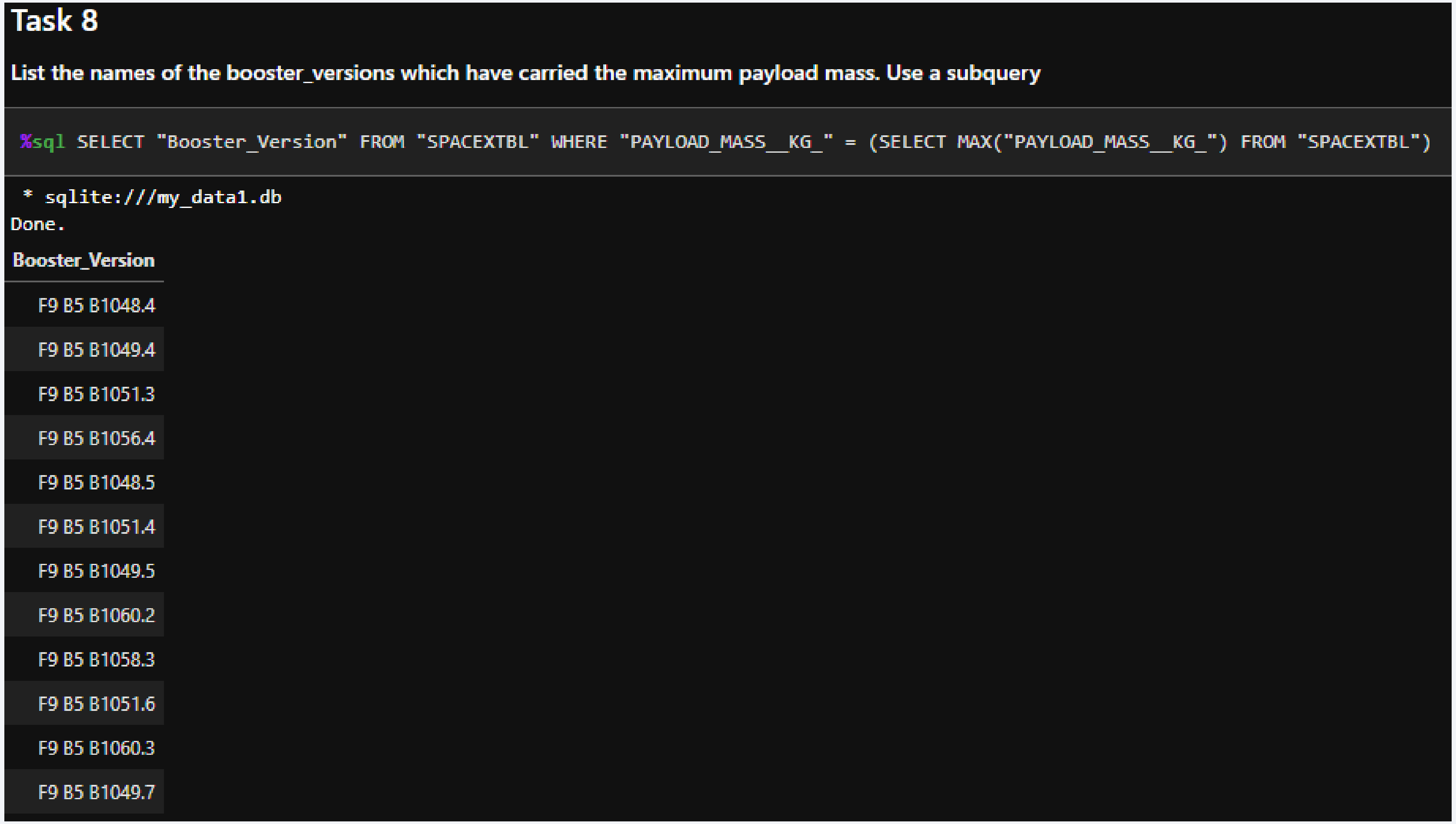
Mission_Outcome	COUNT("Mission_Outcome")
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

❖%sql select "Mission_Outcome" ,COUNT("Mission_Outcome") FROM "SPACEXTBL" GROUP BY "Mission_Outcome“

❖We selected the mission outcome column and then found its count of it after that we grouped the results by the mission outcome.

❖The answer is described in the picture above.

Boosters Carried Maximum Payload



Task 8

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 "SPACEXTBL")
```

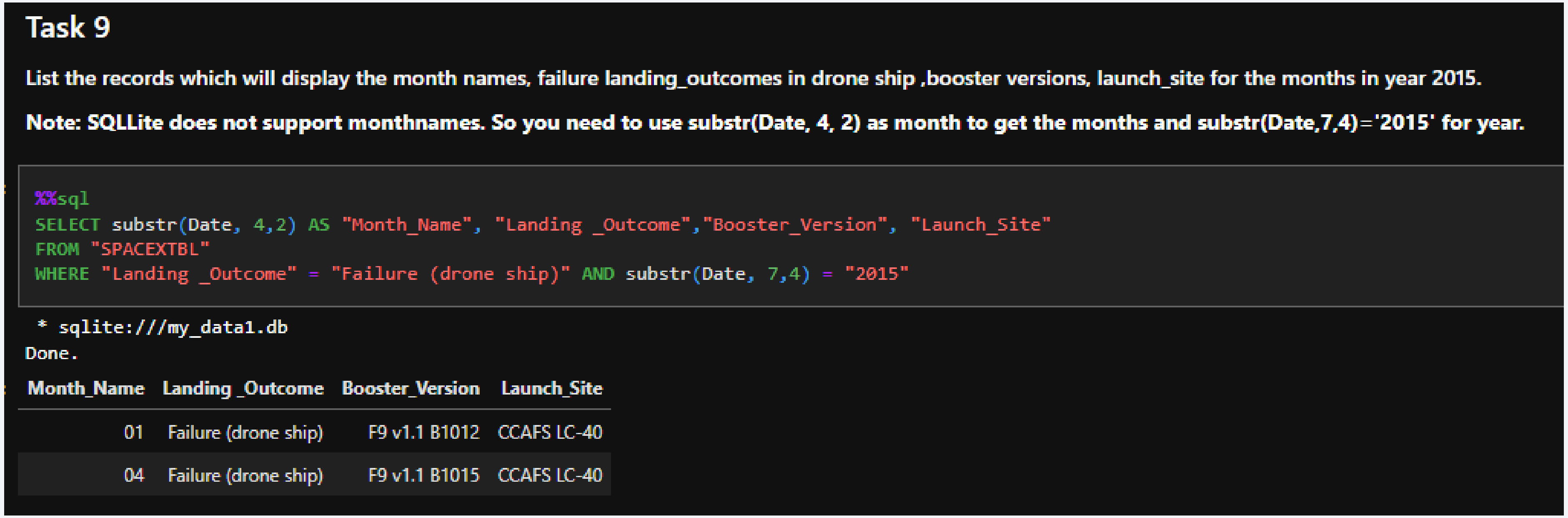
```
 * sqlite:///my_data1.db
Done.
```

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

- We have used the subquery that selects the maximum payload mass value and then keeps all the records of the Booster version column that matches the maximum payload mass value.

- %sql SELECT "Booster_Version" FROM "SPACEXTBL" WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM "SPACEXTBL")

2015 Launch Records



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, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

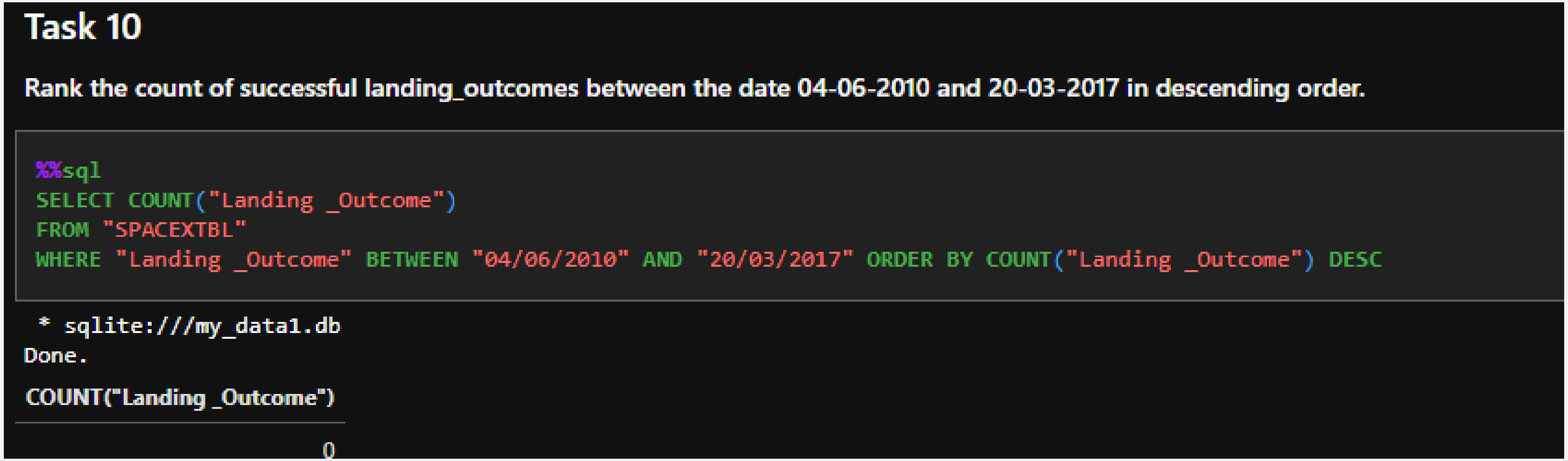
```
%%sql
SELECT substr(Date, 4,2) AS "Month_Name", "Landing _Outcome","Booster_Version", "Launch_Site"
FROM "SPACEXTBL"
WHERE "Landing _Outcome" = "Failure (drone ship)" AND substr(Date, 7,4) = "2015"
```

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

Month_Name	Landing _Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- %%sql
- SELECT substr(Date, 4,2) AS "Month_Name", "Landing _Outcome","Booster_Version", "Launch_Site“
- FROM "SPACEXTBL“
- WHERE "Landing _Outcome" = "Failure (drone ship)" AND substr(Date, 7,4) = "2015“
- We have used the substr(Date, 4,2) function to find the month name and substr(Date, 7, 4) for the years.

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



Task 10

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

```
%%sql
SELECT COUNT("Landing _Outcome")
FROM "SPACEXTBL"
WHERE "Landing _Outcome" BETWEEN "04/06/2010" AND "20/03/2017" ORDER BY COUNT("Landing _Outcome") DESC
```

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

COUNT("Landing _Outcome")
0

❖%%sql

❖SELECT COUNT("Landing _Outcome")

❖FROM "SPACEXTBL"

❖WHERE "Landing _Outcome" BETWEEN "04/06/2010" AND "20/03/2017" ORDER BY COUNT("Landing _Outcome") DESC

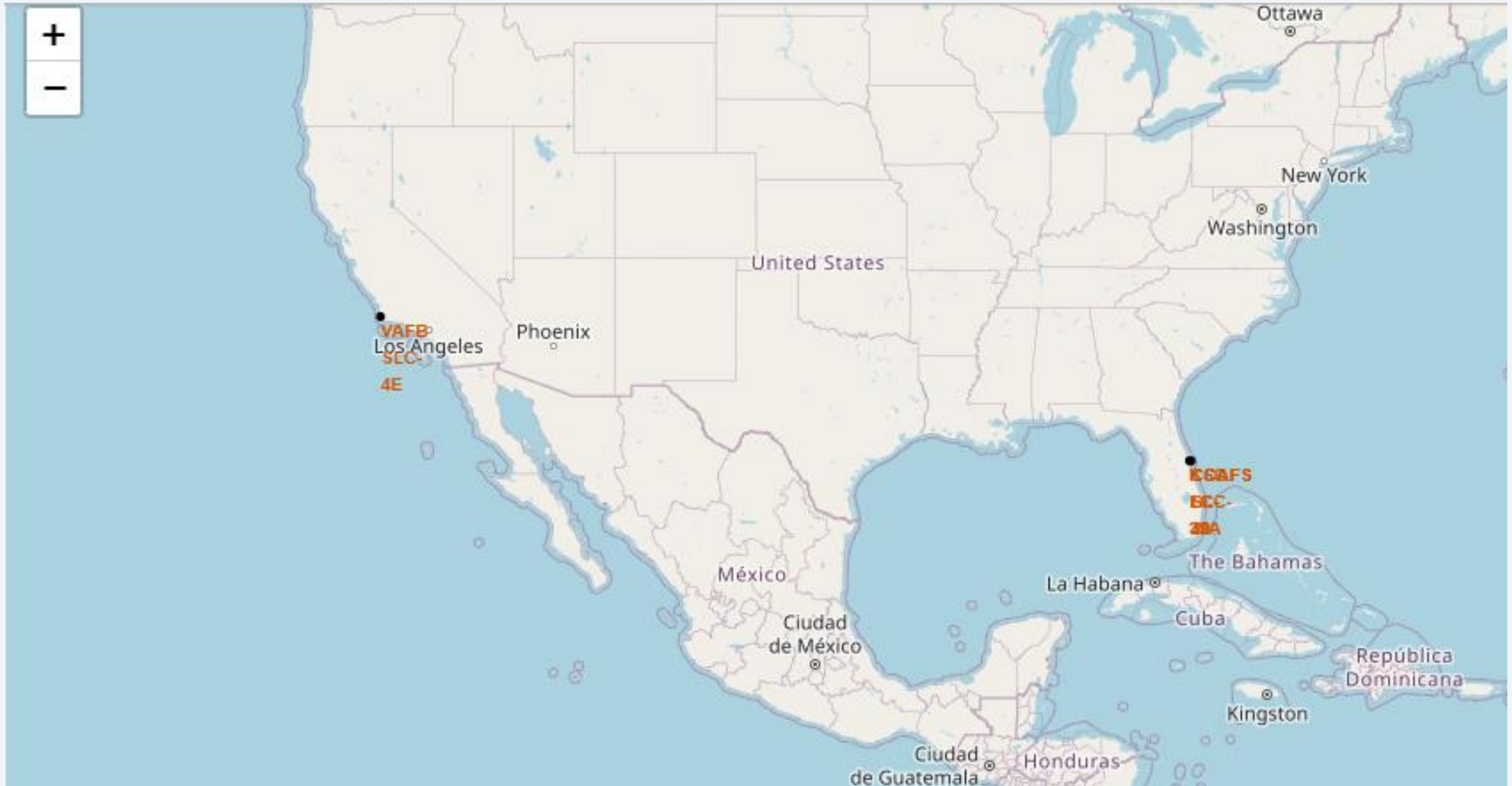
❖Find the count of the landing outcome if the condition is true then sort them from biggest to lowest.

Section 3

Launch Sites Proximities Analysis

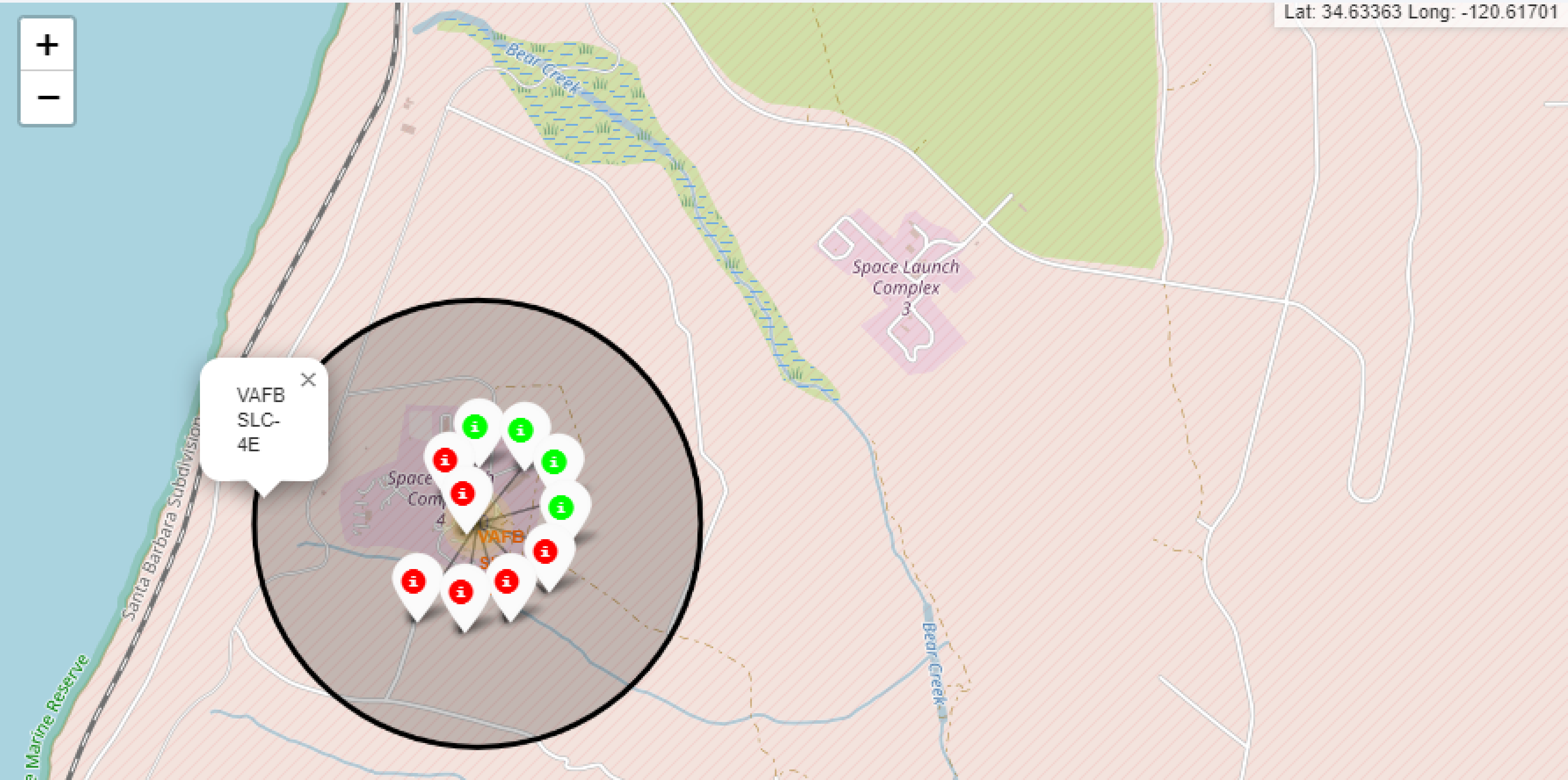
Launch sites locations

- **VAFB SLC-4E** is located in Los Angeles and it is so far away from other launch sites.
- **K SLC-39A, CCAFS SLC-40, and CCAFS LC-40** are located in Florida next to the Merritt Island



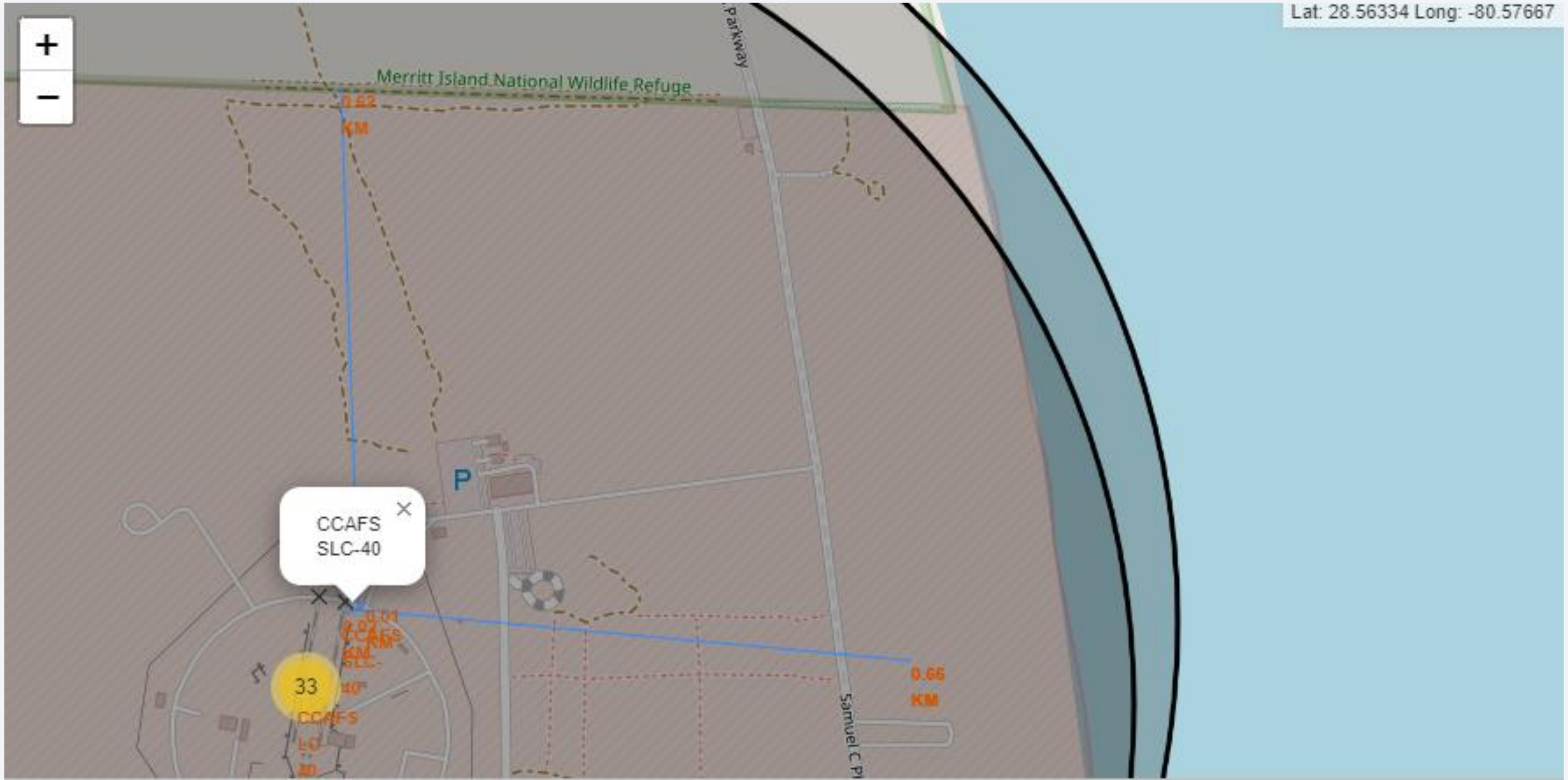
VAFB SLC-4E Success VS Failure Launches

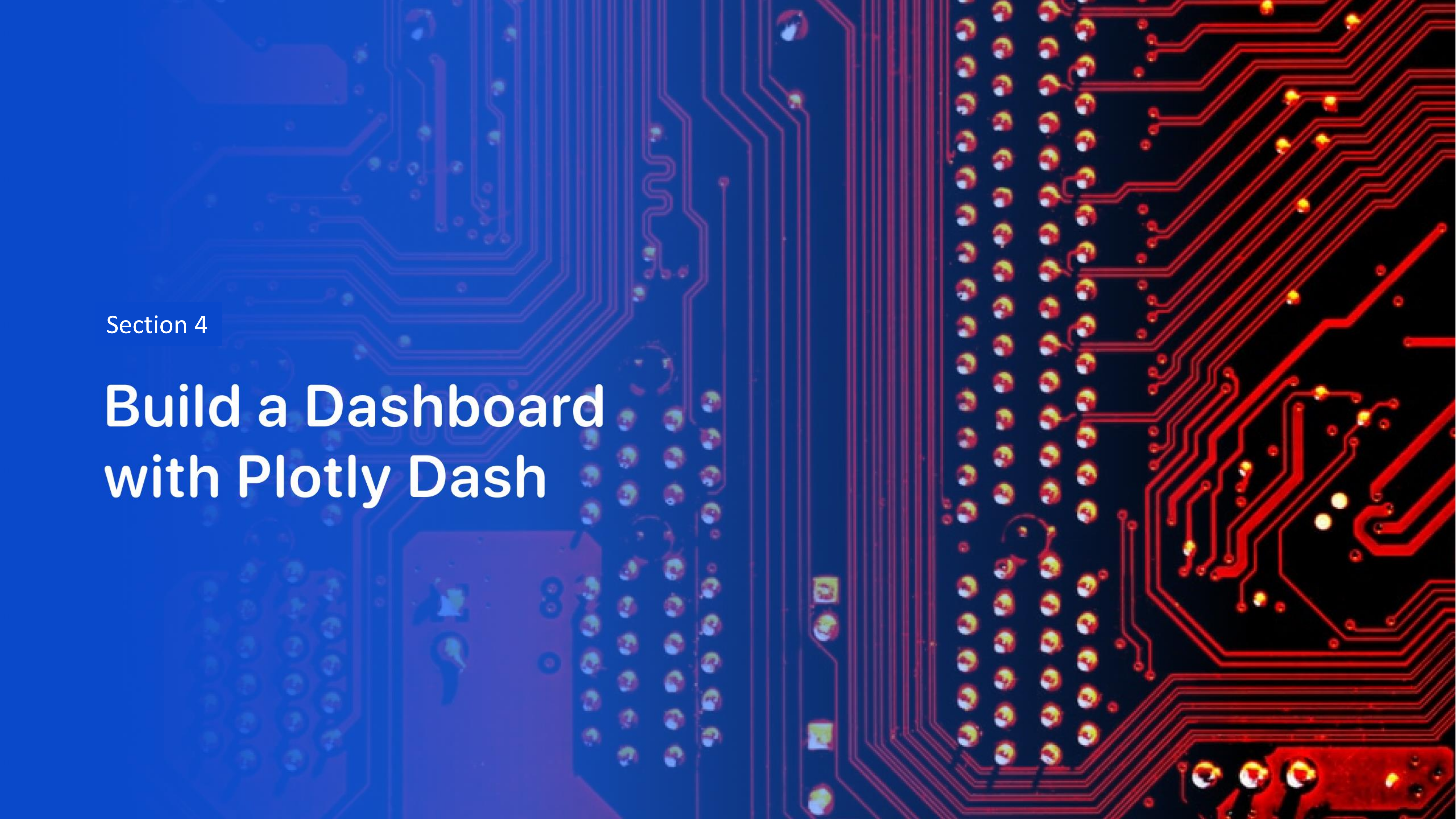
- The green color indicates a successful launch.
- The red color indicates a failed launch.



CCAFS SLC-40 Approximate coastline and highway

❖The blue line indicates the distance i.e. CCAFS SLC-40 is far 0.66km from the highway and 0.62km from the coastline.


Lat: 28.56334 Long: -80.57667
+
−
Parkway
Merritt Island National Wildlife Refuge
0.62 KM
P
CCAFS SLC-40
×
33
0.66 KM
Samuel C Pl

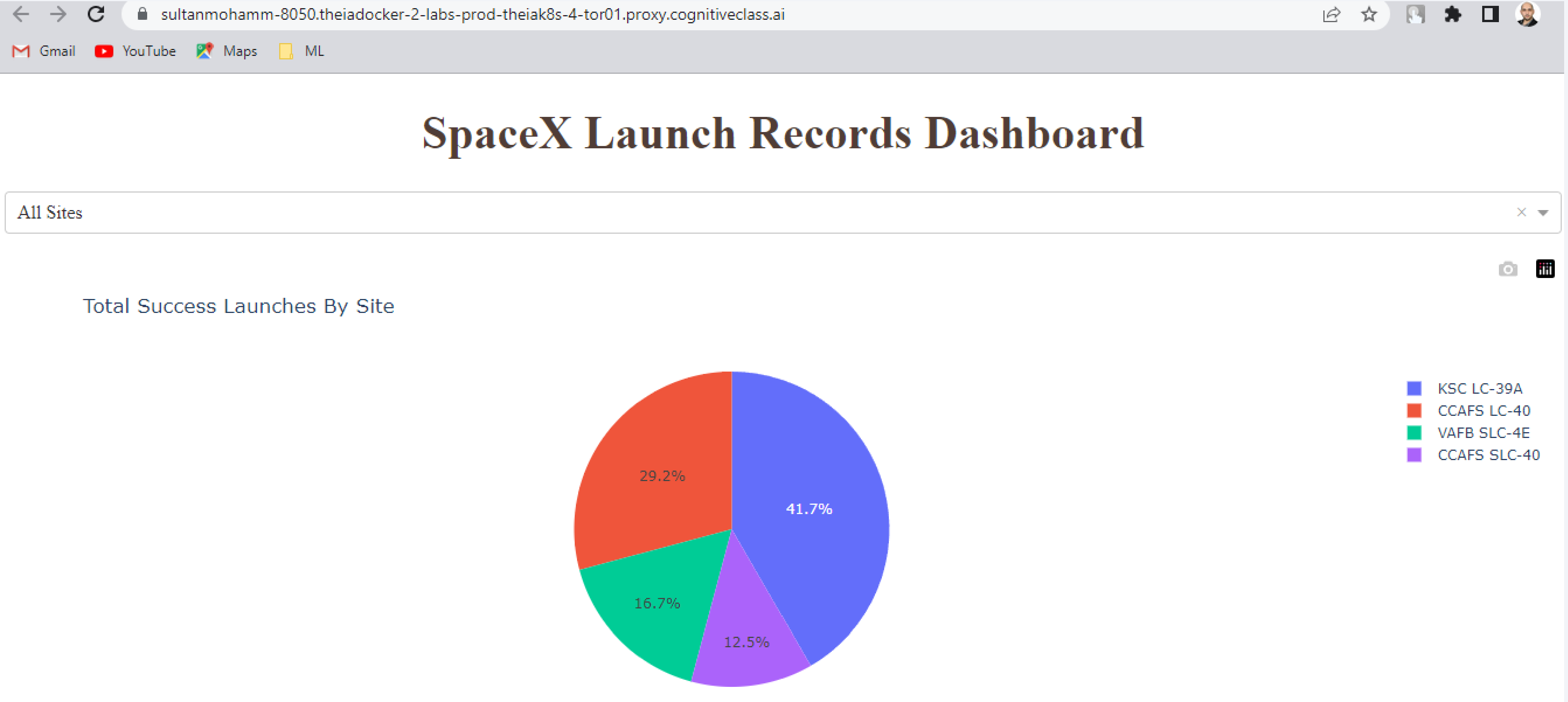


Section 4

Build a Dashboard with Plotly Dash

SpaceX Launch success count for all Sites

- ❖KSC LC-39A has the highest success launch rate overall sites.
- ❖CCAFS SLC-40 has the lowest success launch rate overall sites.


sultanmohamm-8050.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai
Gmail
YouTube
Maps
ML

SpaceX Launch Records Dashboard

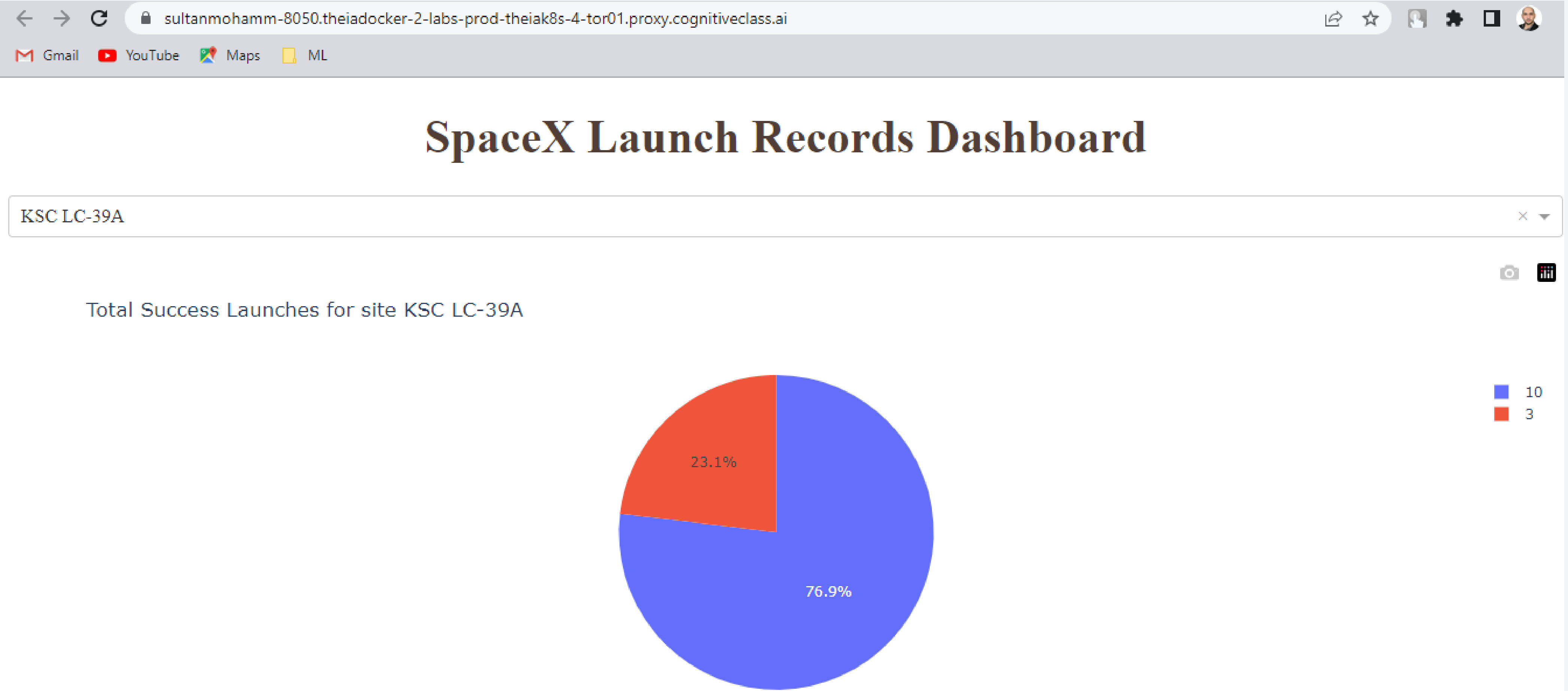
All Sites

Total Success Launches By Site

29.2%
41.7%
16.7%
12.5%
KSC LC-39A
CCAFS LC-40
VAFB SLC-4E
CCAFS SLC-40

KSC LC-39A Success launches

- ❖**KSC LC-39A** has a 23.1% failure launch rate and a 76.9% success launch rate.



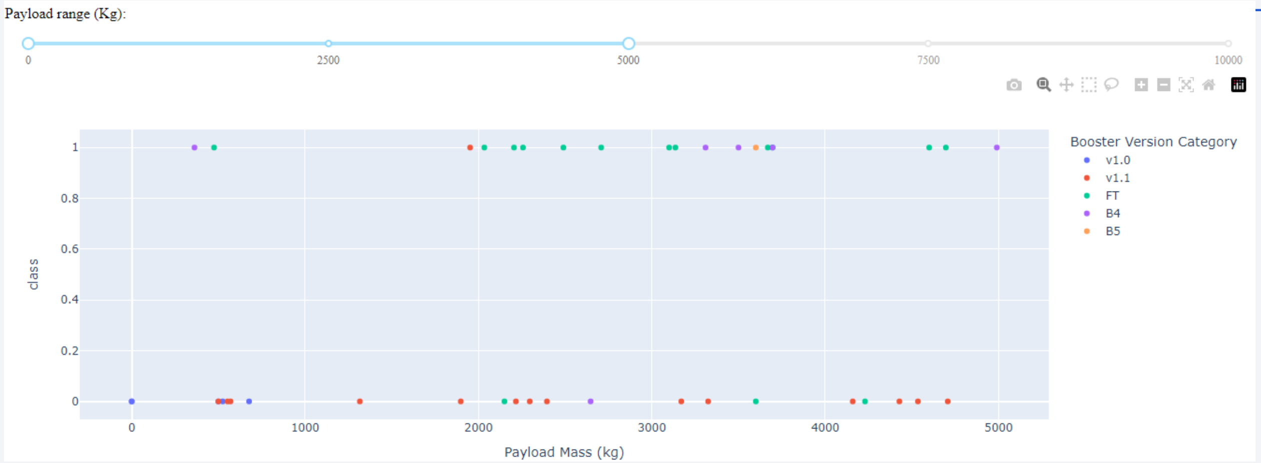
sultanmohamm-8050.theiadocker-2-labs-prod-theiak8s-4-tor01.proxy.cognitiveclass.ai
Gmail
YouTube
Maps
ML

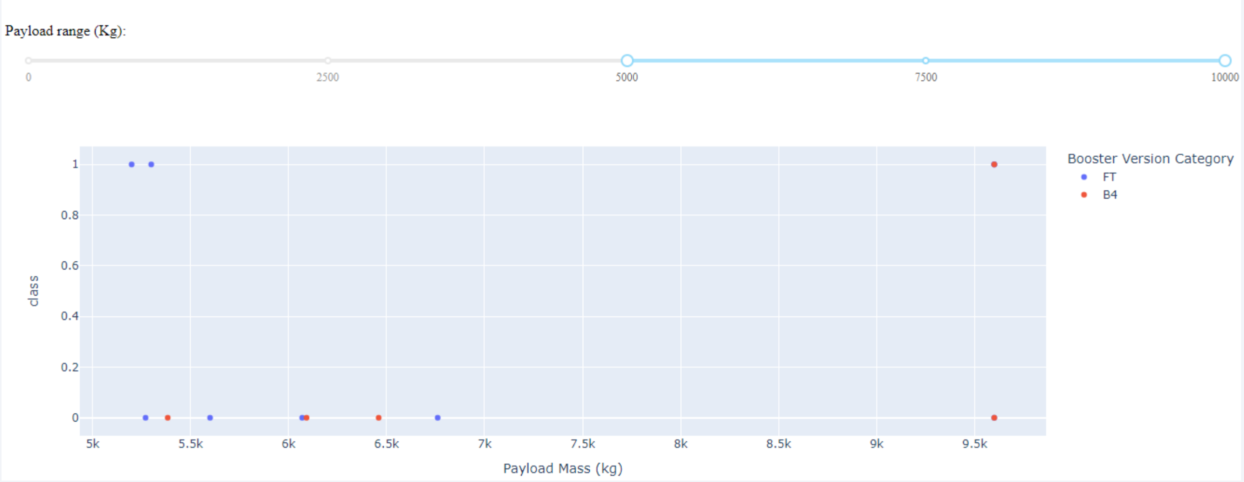
SpaceX Launch Records Dashboard

KSC LC-39A

Total Success Launches for site KSC LC-39A
23.1%
76.9%
🟦 10
🟥 3

Payload vs. Launch Outcome scatter plot for all sites





❖ This Figure shows that in the payload mass range between 0 and 5k, all Booster Versions are exists.

- ❑ FT Booster version has 12 of 15 successful launches in the above range.
- ❑ B4 Booster version has 5 of 6 successful launches in the above range.

❖ This Figure shows that in the payload mass range between 5k and 10k only FT and B4 Booster Versions are exists.

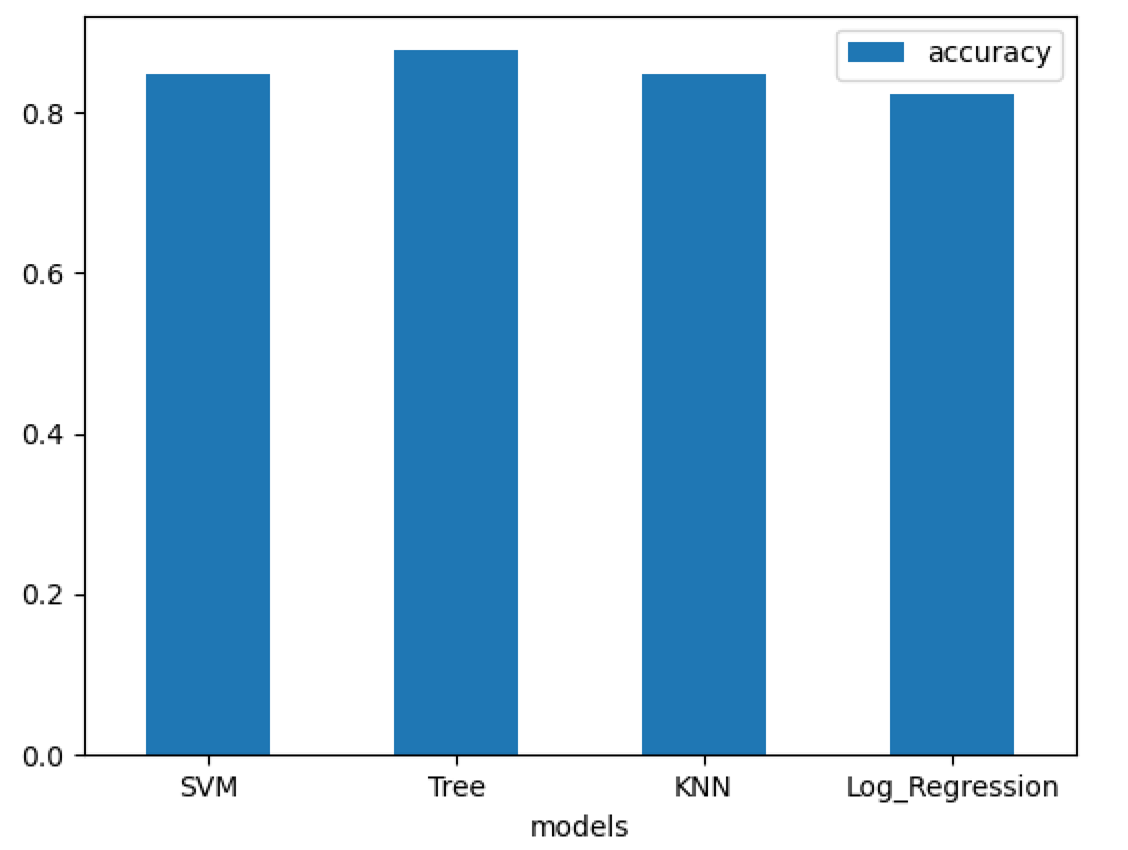
- ❑ FT Booster version has 3 of 8 successful launches in the above range.
- ❑ B4 Booster version has 1 of 5 successful launches in the above range.
- ❑ V1.0, V1.1, and B5 Booster versions are disappeared.

❖ We can say that FT is the highest successful launch rate overall Booster Versions.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

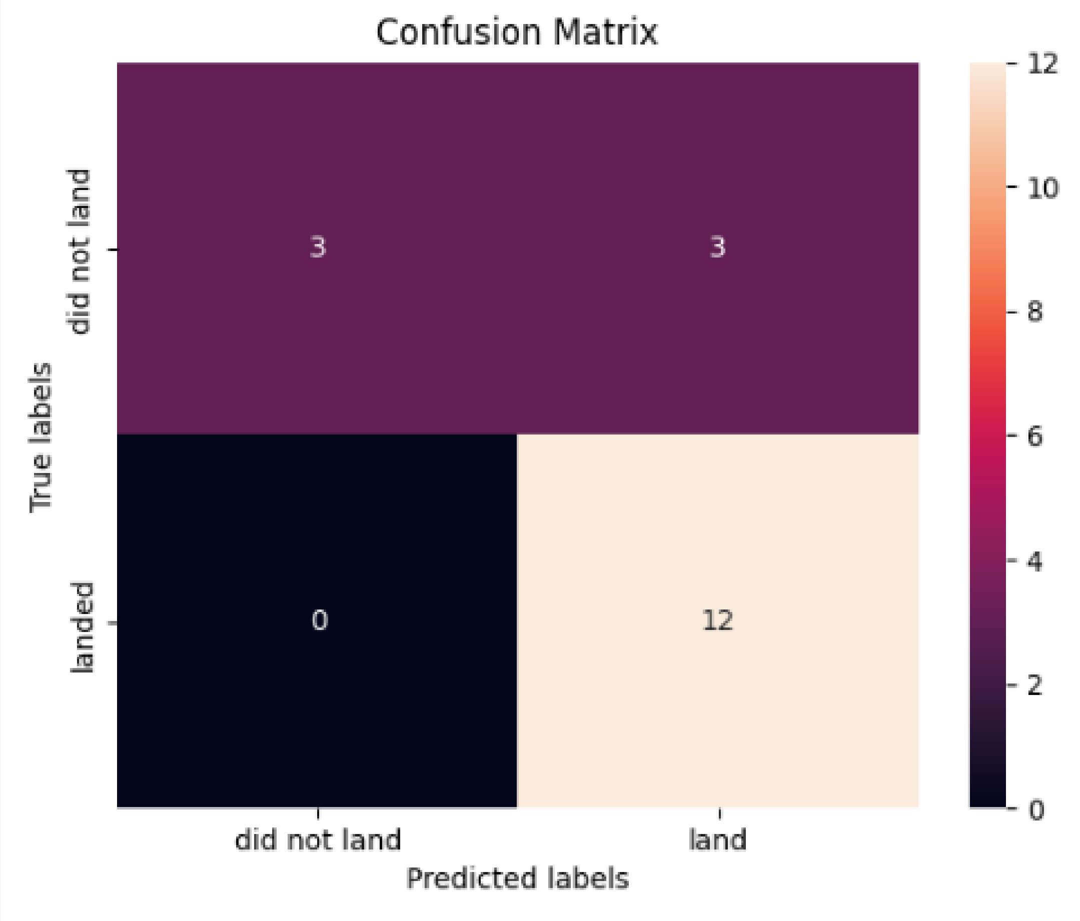


- ❖Models Accuracy:
  - ❑SVM: 0.8482142857142856
  - ❑Tree: 0.8767857142857143
  - ❑KNN: 0.8482142857142858
  - ❑Logistic Regression: 0.822
- ❖Best Method is the Decision Tree with a score of 0.88

Confusion Matrix

❖The Figure represents the Confusion Matrix of the D3 model.

❖We have 3 False positives out of 18 which means we have a 16.66% error in predicting whether the first stage of Falcon 9 will land successfully or not.



Conclusions

❖Payload mass has the greatest impact among all other features.

❖the first successful landing outcome in ground pad was achieved in 01-05-2017

❖The KSC LC-39A launch site has the highest success launch rate 76.9%.

❖The FT Booster Versions has the highest successful launch rate overall Booster versions.

❖Best Method is the Decision Tree with a score of 0.88.

Appendix

- ❖ My Project Notebook on GitHub:
  - ❖ Data Collection link.
  - ❖ Web scraping link.
  - ❖ EDA data visualization link.
  - ❖ EDA with SQL link.
  - ❖ Interactive Visualization with Folium link.
  - ❖ Interactive Visualization with Plotly link.
  - ❖ Predictive analytics link.
  - ❖ Data Wrangling link.
- ❖ Useful code snippets:
  - ❖ How to plot a bar chart using Pandas https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.plot.bar.html

Thank you!