IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Gerardo Mora Cuevas
12/12/2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection using APIs
  - Data Collection using Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis and Data Visualization
  - Data Visualization using Geospatial Data
  - Interactive Visual Analytics using Folium and Plotly Dash
  - Applied Machine Learning Models
- Summary of all results
  - Exploratory Data Analysis Results
  - Interactive Analytics in Jupyter Notebooks
  - Predictive Analytics results from ML

Introduction

- Project background and context

The commercial space age is here, companies are making space travel affordable for everyone. One of the most successful companies is SpaceX. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards of 165 million dollars each. Much of the savings is because SpaceX can reuse the first stage. This stage does most of the work and is quite large and expensive. Sometimes the first stage does not land. Sometimes it will crash. Other times, Space X will sacrifice the first stage due to the mission parameters. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch.

- Problems you want to find answers
  - Is it possible to predict whether a Falcon 9 rocket launch test will be successful or fail based on historical data and relevant variables?
  - What variables or factors have the most significant influence on determining the success or failure of a launch test?
  - Are there any discernible trends in the success rate of Falcon 9 launches over time?
  - Can specific areas for improvement or adjustment in future launch tests be identified based on data analysis results and machine learning models?
  - How accurate is the predictive model based on historical data? Can it be trusted?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - The data for this project was collected using APIs and web scraping from Wikipedia
- Perform data wrangling
  - The data for this project was processed handling missing and null values, correcting errors, standardizing date formats and applying data transformation
- 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.

The data was collected using a combination of APIs and web scrapping techniques. APIs were utilized to directly access structured data from specific sources, while web scraping involved extracting information from various web pages, particularly from Wikipedia, by parsing the HTML content to gather the required data.

Data Collection

API Access

- Retrieve data using SpaceX API
- Fetch specific details
- Save data into structured format

Web Scraping

- Identify Wikipedia pages related to Falcon 9 launches
- Extract relevant data using web scraping tools
- Convert scraped data into usable format

Data Integration

- Merge data obtained from APIs and web scraping
- Handle data compatibility issues
- Create a unified dataset

Data Cleaning

- Address missing values and inconsistencies
- Remove duplicates and handle null entries
- Standardize formats for better analysis

Data Storage

- Store the cleaned and integrated dataset securely
- Ensure accessibility for analysis

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts
- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose
- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/ee309ba57d40a86dc4821b72caafcb10c7b2cfc4/1.%20jupyter-labs-spacex-data-collection-api.ipynb

1
- Initiating SpaceX REST Calls

2
- Endpoint Calls for Launch Data

3
- Rocket Specifications via API Calls

4
- Mission Details Retrieval

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts
- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose
- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/d442194955f2078b4e82efed52fafb1d8a3c128a/2.%20jupyter-labs-webscraping.ipynb

1
- Initiating Web Scraping

2
- Wikipedia Scraping for Launch Details

3
- Data Extraction Process

Data Wrangling

- Describe how data were processed
  - Data Cleaning

    Handled missing values, removed duplicates, standardized formats, and addressed outliers.
  - Data Integration

    Combined data from various sources into a unified dataset
  - Feature Engineering

    Create new variables for better analysis and modeling
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose
- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/1d99b8eb73a045cb3402e807b5d5110932770944/3.%20labs-jupyter-spacex-Data%20wrangling.ipynb

1. Data Collection
2. Data Cleaning
3. Data Integration
4. Feature Engineering

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts

We explored the data by visualizing the relationship between flight number and launch site, payload and launch site, the success rate of each orbit type, flight number, and orbit type, and the launch success yearly trend.

- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/5593663661644ece4059d563e7c1a7010d94a55e/5.%20jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the strin 'CCA'
  - Display the total payload mass carried by boosters launched by NASA
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first successful landing outcome in ground pad was achieved
  - 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 droneship. 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.

EDA with SQL

- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose
- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/02732eaa135cb4692d8a723972712e9458bfc57f/4.%20jupyter-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

We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.

We assigned the feature launch outcomes (failure or success) to class 0 and 1

Using the color-labeled marker clusters, we identified which launch sites have relatively high success rate

We calculated the distances between a launch site to its proximities. We answered some questions about the distance between the launch site and railways, highways, cities and coastlines.

- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/99e8a06b7707ecaff98e440a42312476dcdfd78f/6.%20lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard

We built an interactive dashboard with Plotly dash

We plotted pie charts showing the total launches by a certain sites

We plotted scatter graph showing the relationship with Outcome and Payload Mass (kg) for the different booster version

- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/4917f5ff0839305267dd34a8bc025f728e1f9a1a/spacex_dash_app.py

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model

We loaded the data using NumPy and pandas, transformed the data and split our data into training and testing

We built different machine learning models and tuned different hyperparameters using GridSearchCV

We used accuracy as the metric for our model and improved the model using feature engineering and algorithm tuning

We found the best performing classification model

- https://github.com/JerryMora/IBM-Data-Science-Capstone/blob/e9eb7fe319a94eb94ca4605fd84687ce9d26a1a9/7.%20SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

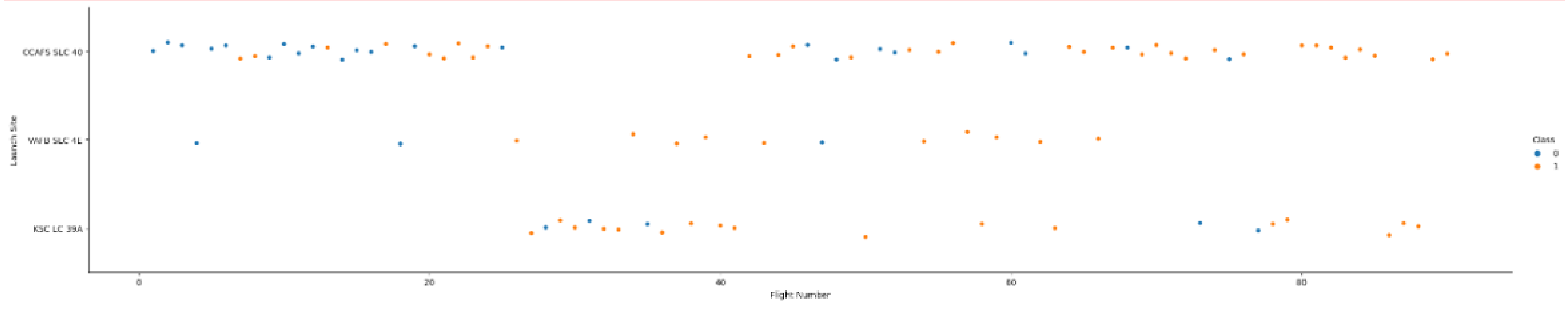
Results

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

Section 2

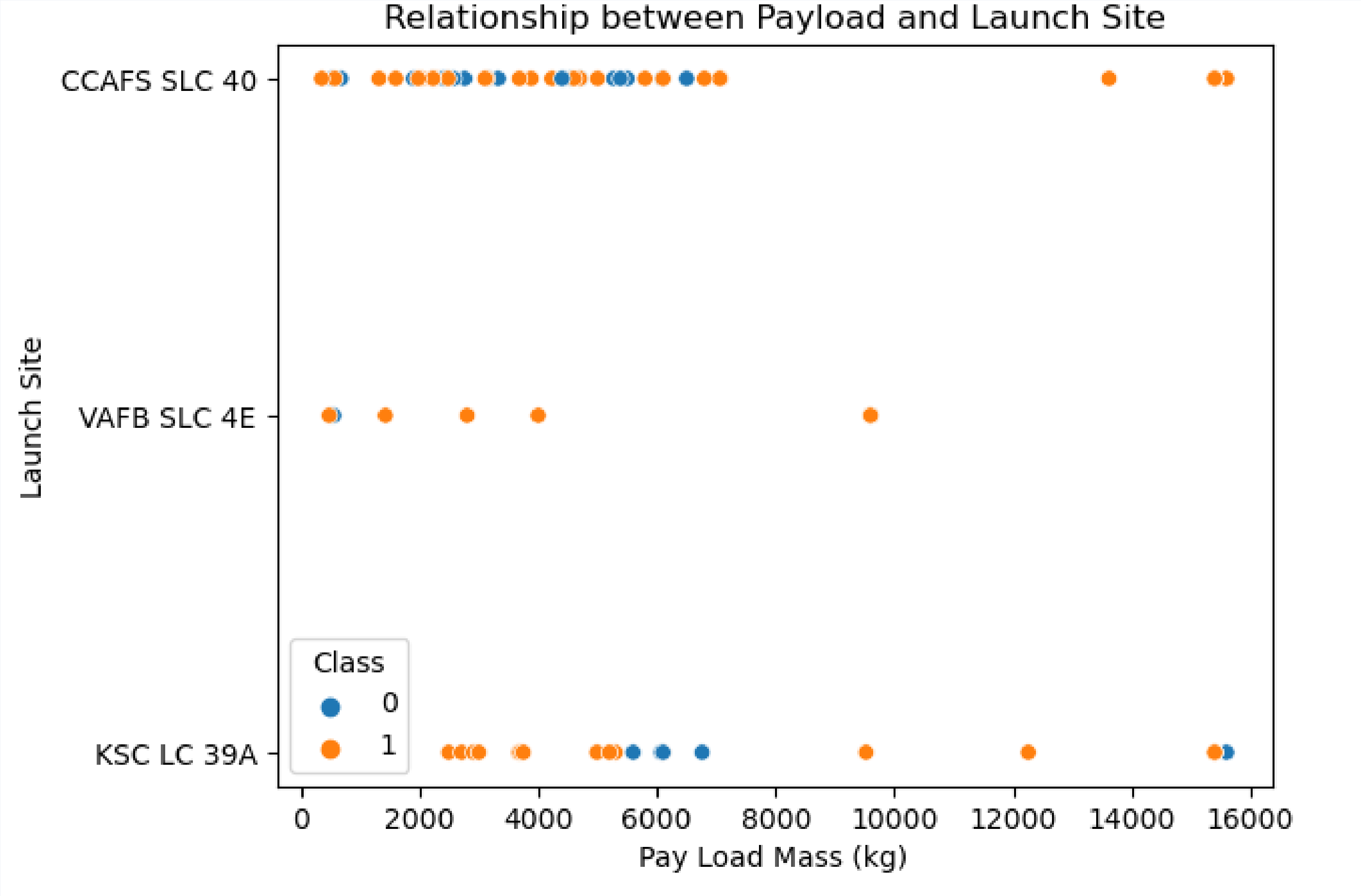
Insights drawn from EDA

Flight Number vs. Launch Site



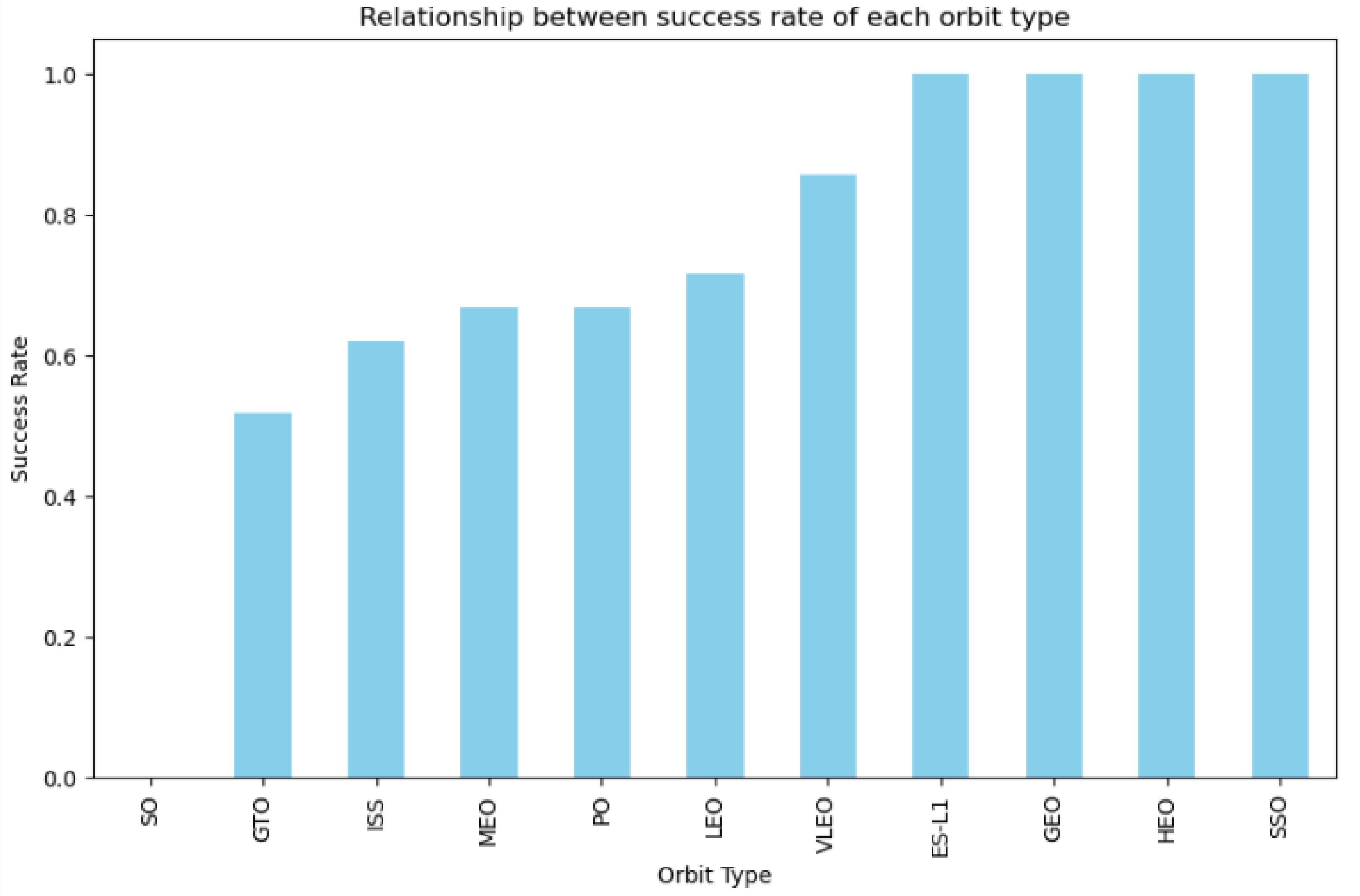
The first launching attempts where performed in the CCAFS SLC 40 launch site, that's why it has a 60% of success.

Payload vs. Launch Site



The grater the payload mass the higher the success rate for the rocket

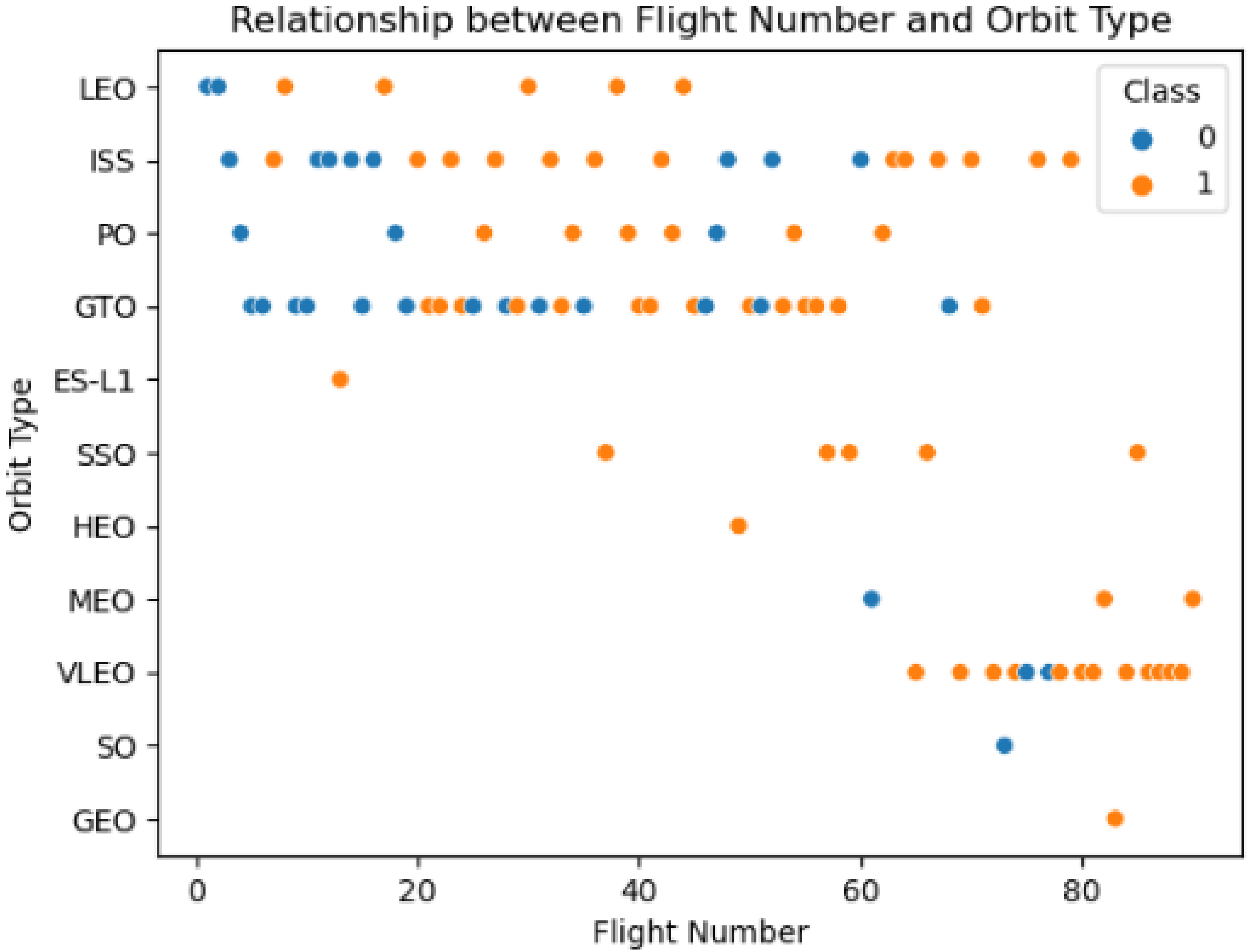
Success Rate vs. Orbit Type



Most successful orbits are:

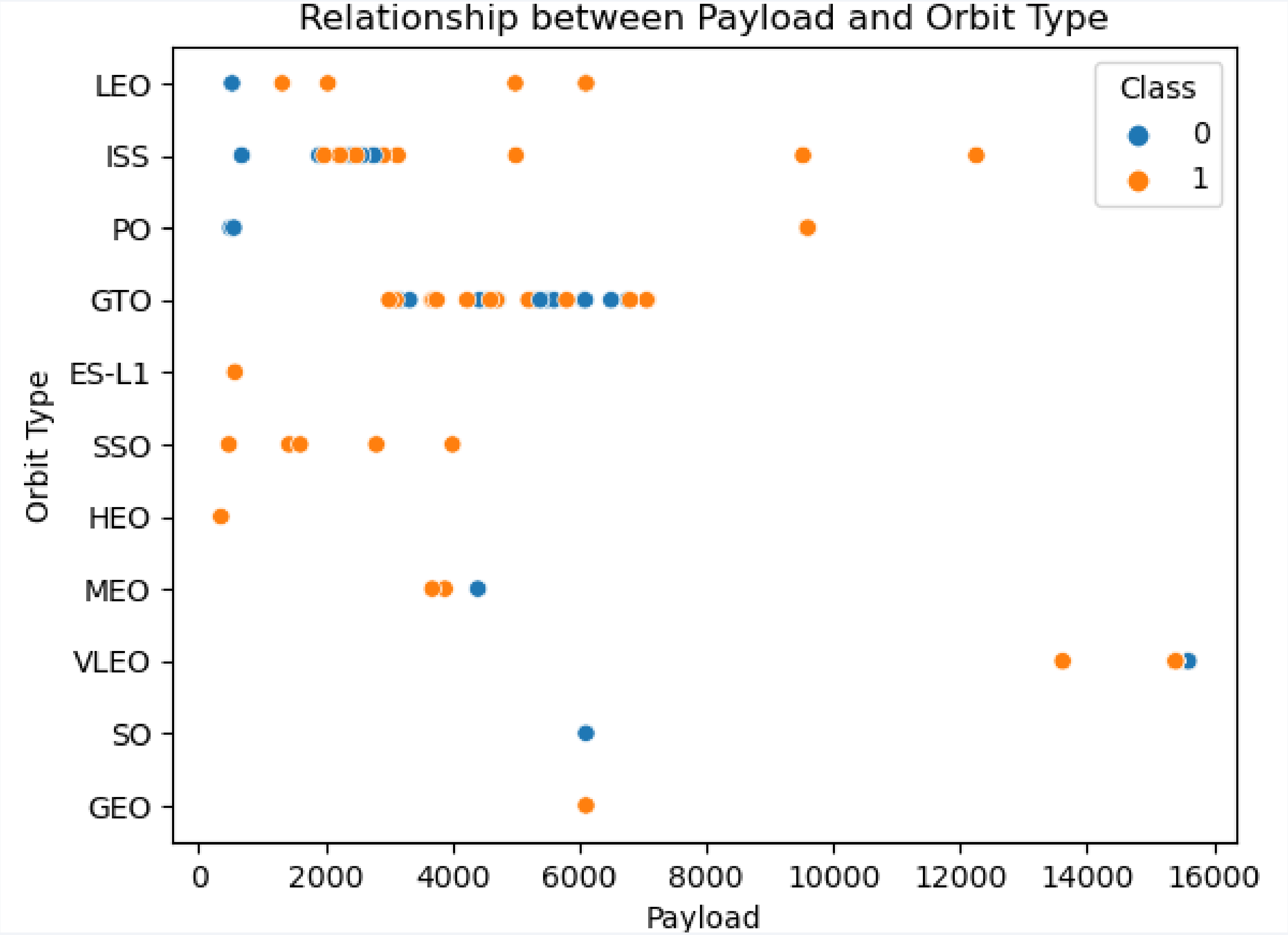
- SSO
- HEO
- GEO
- ES L1
- VLEO

Flight Number vs. Orbit Type



Success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.

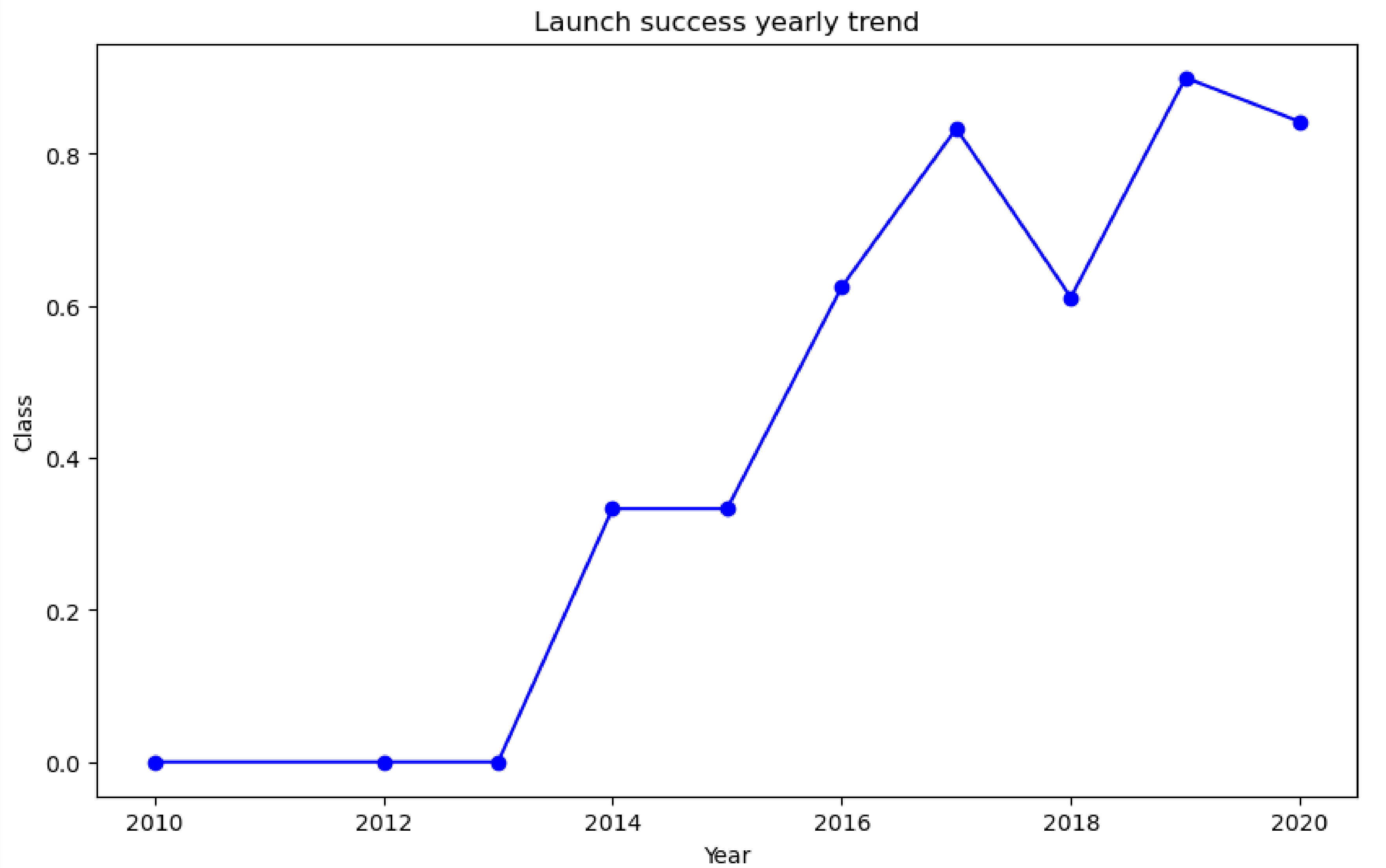
Payload vs. Orbit Type



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



The success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

All Launch Site Names

Task 1

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

[9]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

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

[9]:

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

We used the keyword DISTINCT to show only unique launch sites from the SpaceX data.

Launch Site Names Begin with 'CCA'

Task 2

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

```
[10]: %sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

[10]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	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)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

We used the query with a wildcard character to find 5 records where launch sites begin with 'CCA'

Total Payload Mass

Task 3

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

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS 'Total Playload mass carried by boosters launched by NASA (CRS)' FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)';
```

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

Total Playload mass carried by boosters launched by NASA (CRS)
45596

We used the function SUM to calculate the summatory of the total mass carried by boosters launched by NASA

Average Payload Mass by F9 v1.1

We used the function AVG to calculate the mean payload mass carried specifically by booster version F9 v1.1

Task 5

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

Hint:Use min function

```
[20]: %sql SELECT AVG("PAYLOAD_MASS__KG_") AS 'Average Payload Mass Carried by Booste
```

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

[20]:

Average Payload Mass Carried by Booster Versión F9 v1.1
2928.4

First Successful Ground Landing Date

We used the MIN function to calculate the date when the first successful landing outcome in ground pad was acheived

Task 5

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

Hint:Use min function

[14]:
```
%sql SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';
```

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

[14]:

MIN(Date)
2015-12-22

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

[15]:
```
%sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (dr
```

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

[15]:

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

We used the WHERE clause to filter for boosters that have successfully landed on the drone ship and applied the condition to determine successful landing with payload mass greater than 4000 but less than 6000.

Total Number of Successful and Failure Mission Outcomes

We used the COUNT function to list the total number of successful and failure mission outcomes grouped by mission outcomes.

Task 7

List the total number of successful and failure mission outcomes

[16]:
```
%sql SELECT Mission_Outcome, COUNT (*) AS 'Number of Successful and failure mission outcom
```

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

[16]:

Mission_Outcome	Number of Successful and failure mission outcomes
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

Task 8

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

[17]:
```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) F
```

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

[17]:

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 used a subquery with the MAX function to determine the max payload mass kg so we can query the booster versions with the max payload mass kg.

2015 Launch Records

```
[18]: %%sql

SELECT
    CASE substr(Date,6,2)
        WHEN '01' THEN 'January'
        WHEN '02' THEN 'February'
        WHEN '03' THEN 'March'
        WHEN '04' THEN 'April'
        WHEN '05' THEN 'May'
        WHEN '06' THEN 'June'
        WHEN '07' THEN 'July'
        WHEN '08' THEN 'August'
        WHEN '09' THEN 'September'
        WHEN '10' THEN 'October'
        WHEN '11' THEN 'November'
        WHEN '12' THEN 'December'
    END AS Month_Name,
    Landing_Outcome,
    Booster_Version,
    Launch_Site
FROM
    SPACEXTABLE
WHERE
    substr(Date,0,5)='2015'
    AND Landing_Outcome='Failure (drone ship)';

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

[18]:

Month_Name	Landing_Outcome	Booster_Version	Launch_Site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

We had to implement a Switch Case expression in order to extract the month of the column Date. Then, we specified which columns we wanted to retrieve and the specific conditions in WHERE

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

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.

[21]:
```
%%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY COUNT(Landing_Outcome) DESC;
```

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

[21]:

Landing_Outcome	COUNT(Landing_Outcome)
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

We selected landing outcomes and the COUNT function of landing outcomes from the data and used the WHERE clause to filter for landing outcomes between 2010-06-04 and 2017-03-20, then we grouped the landing outcomes an ordered in descending order.

Section 3

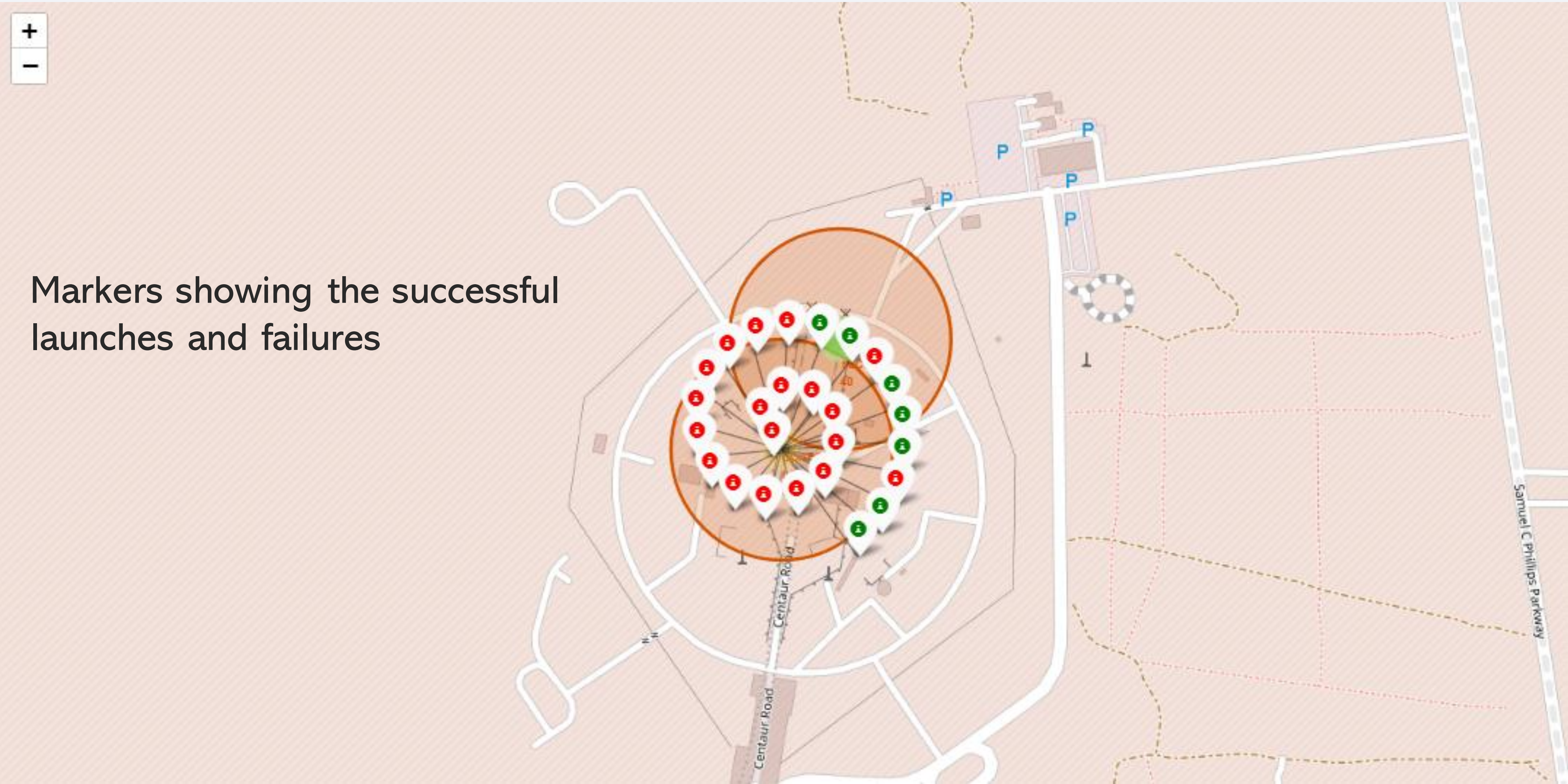
Launch Sites Proximities Analysis

Map with all launch sites

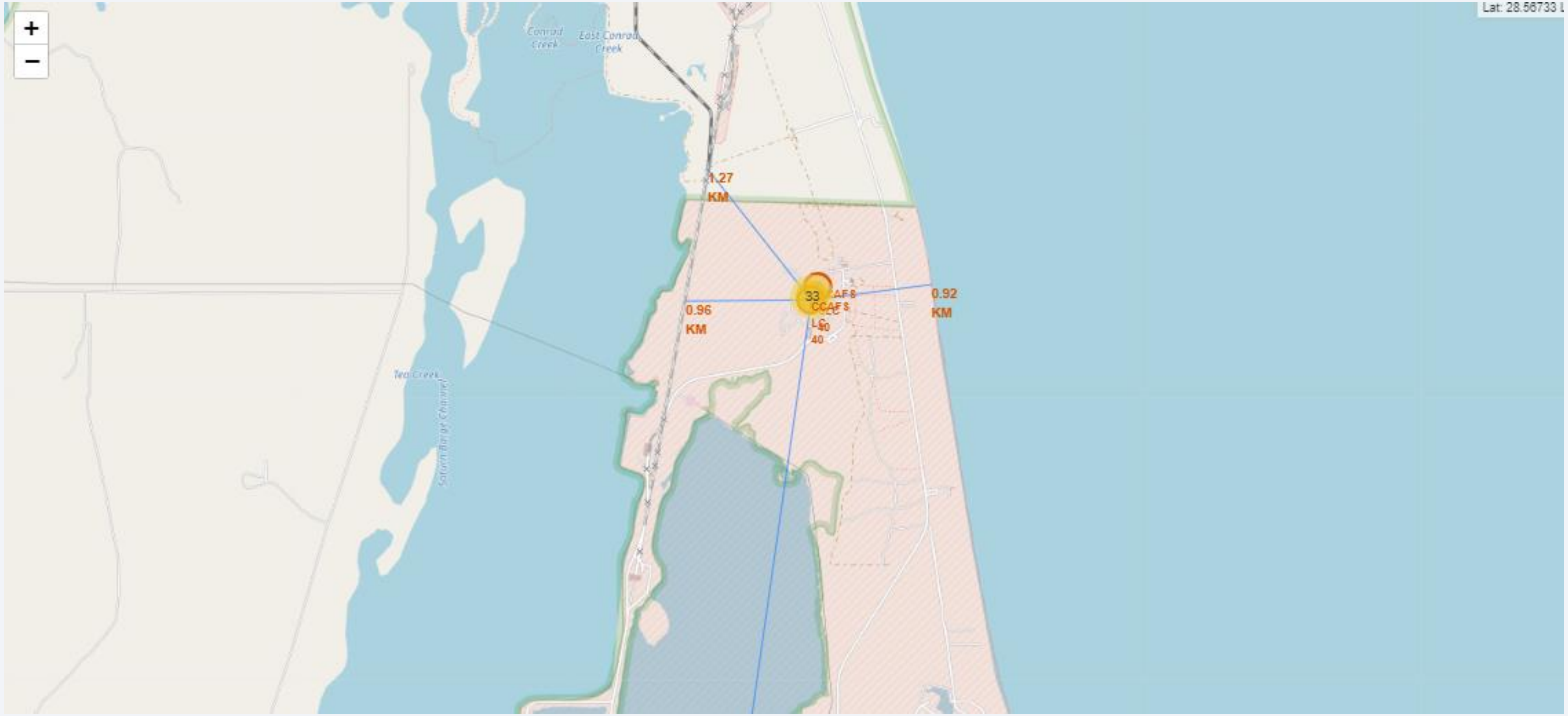
Map with markers of where the space stations are (California and Florida)

Markers with the success/failed launches for each site on the map

Markers showing the successful launches and failures



Launch Site Distance to Landmarks

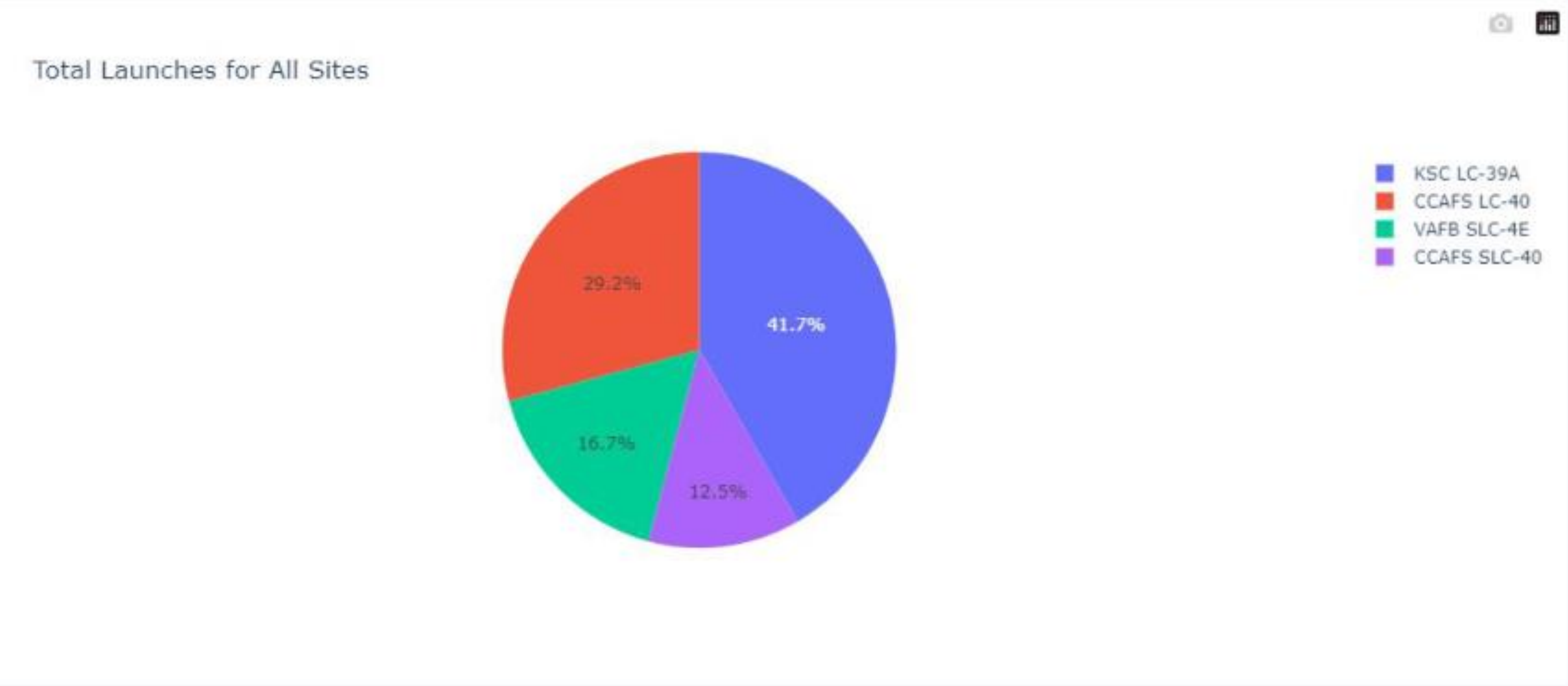


In this map we can see the distance between a few reference points in which can explain the elements near the launch station

Section 4

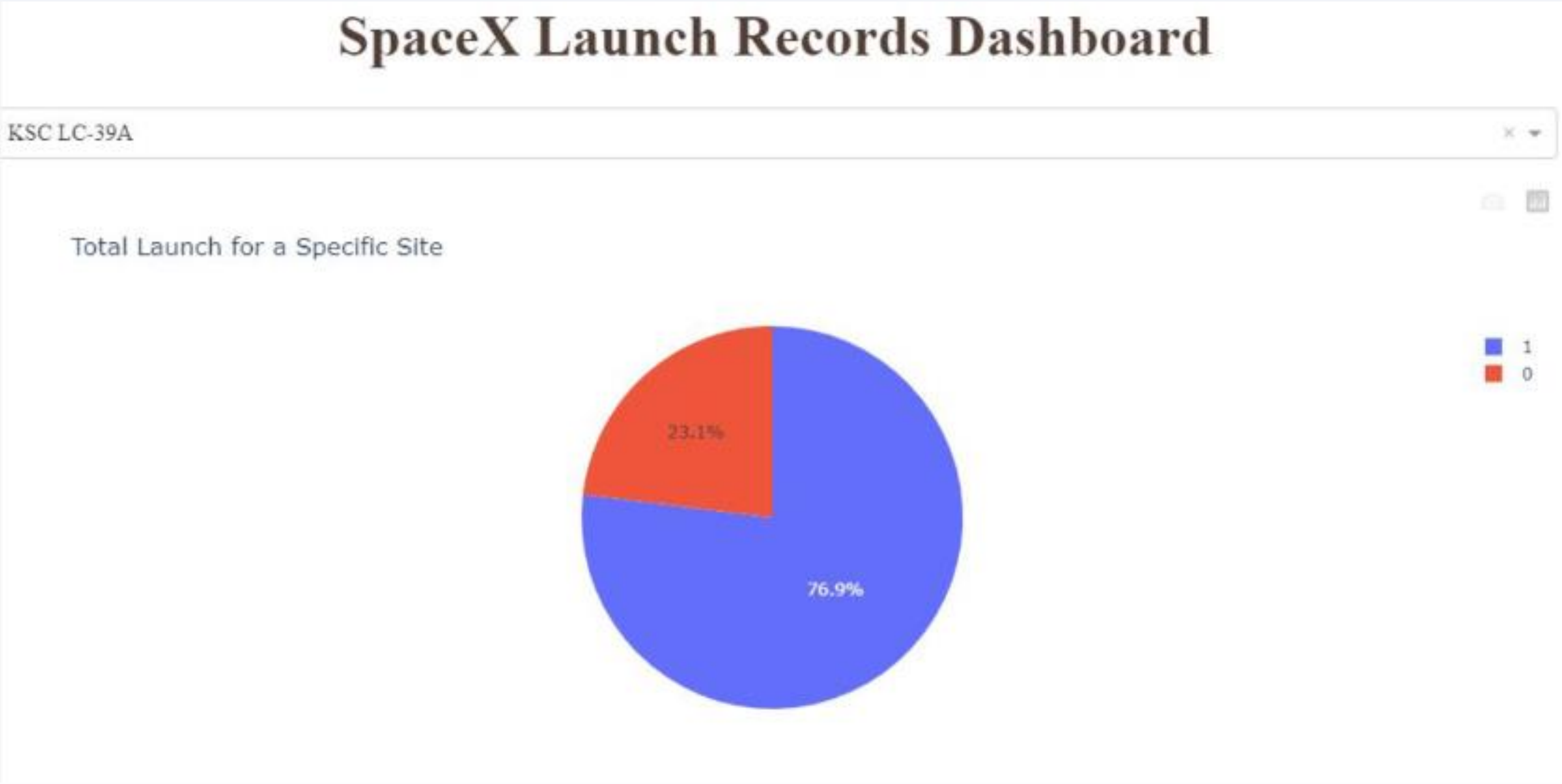
Build a Dashboard with Plotly Dash

Pie chart showing the success percentage achieved by each launch site



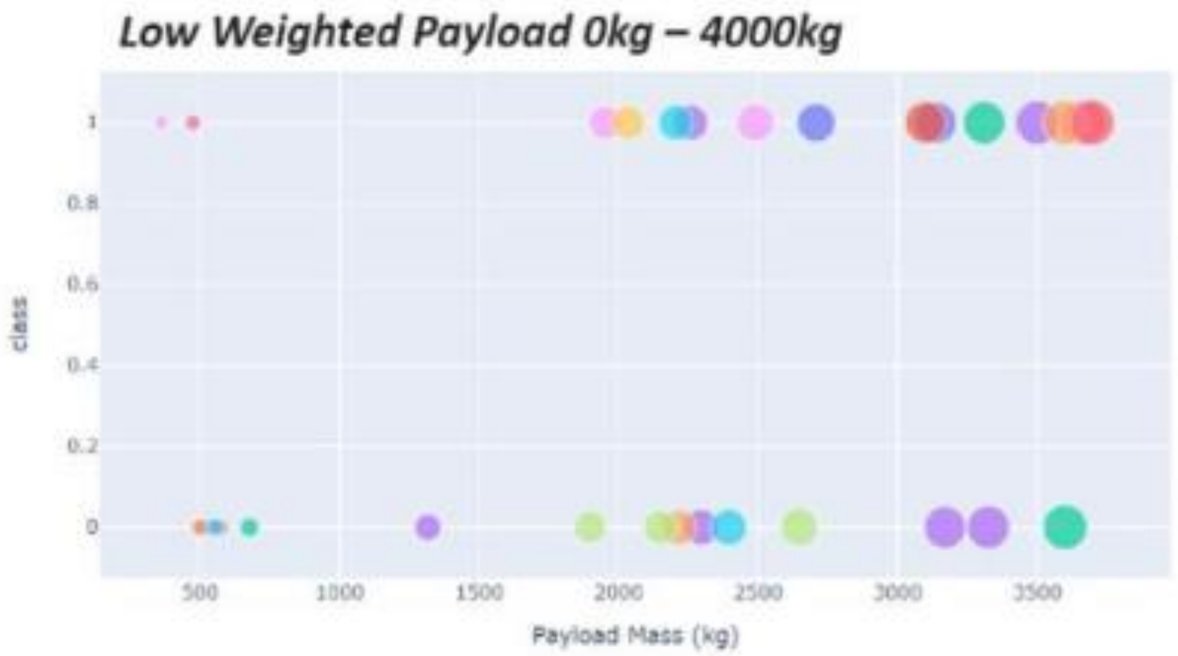
We can see that KSC LC-39A had the most successful launches from all the sites

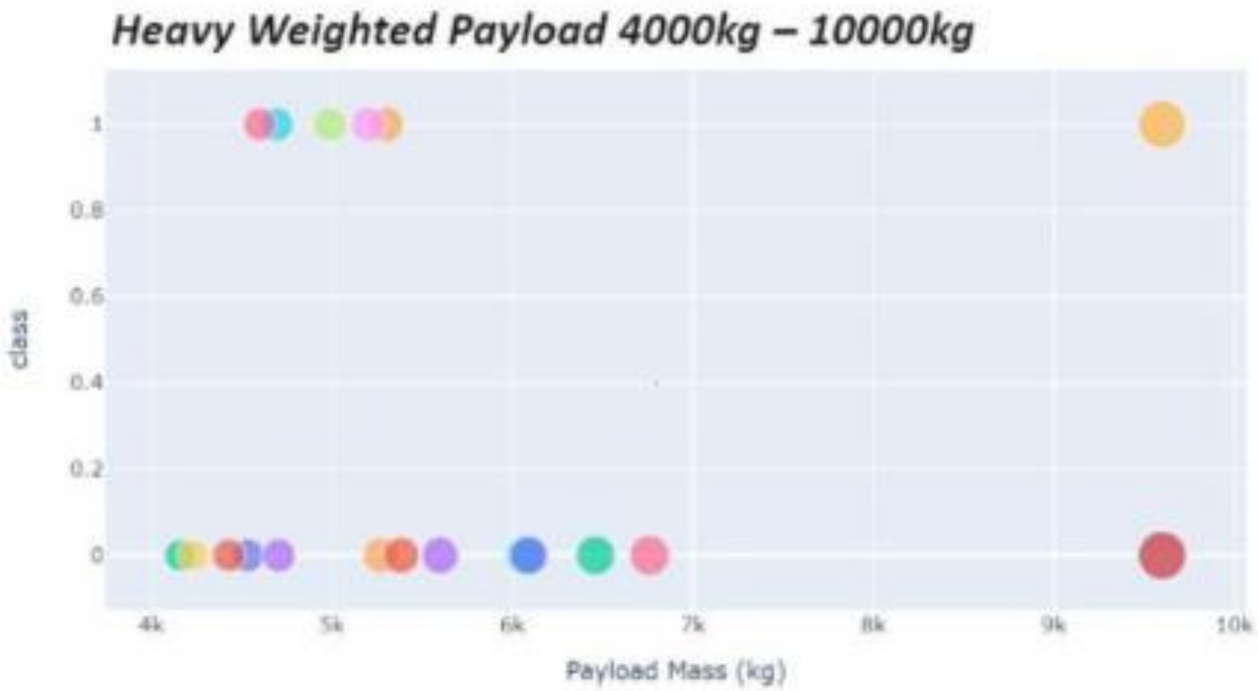
Pie chart showing the launch site with the highest launch success ratio



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

<Dashboard Screenshot 3>



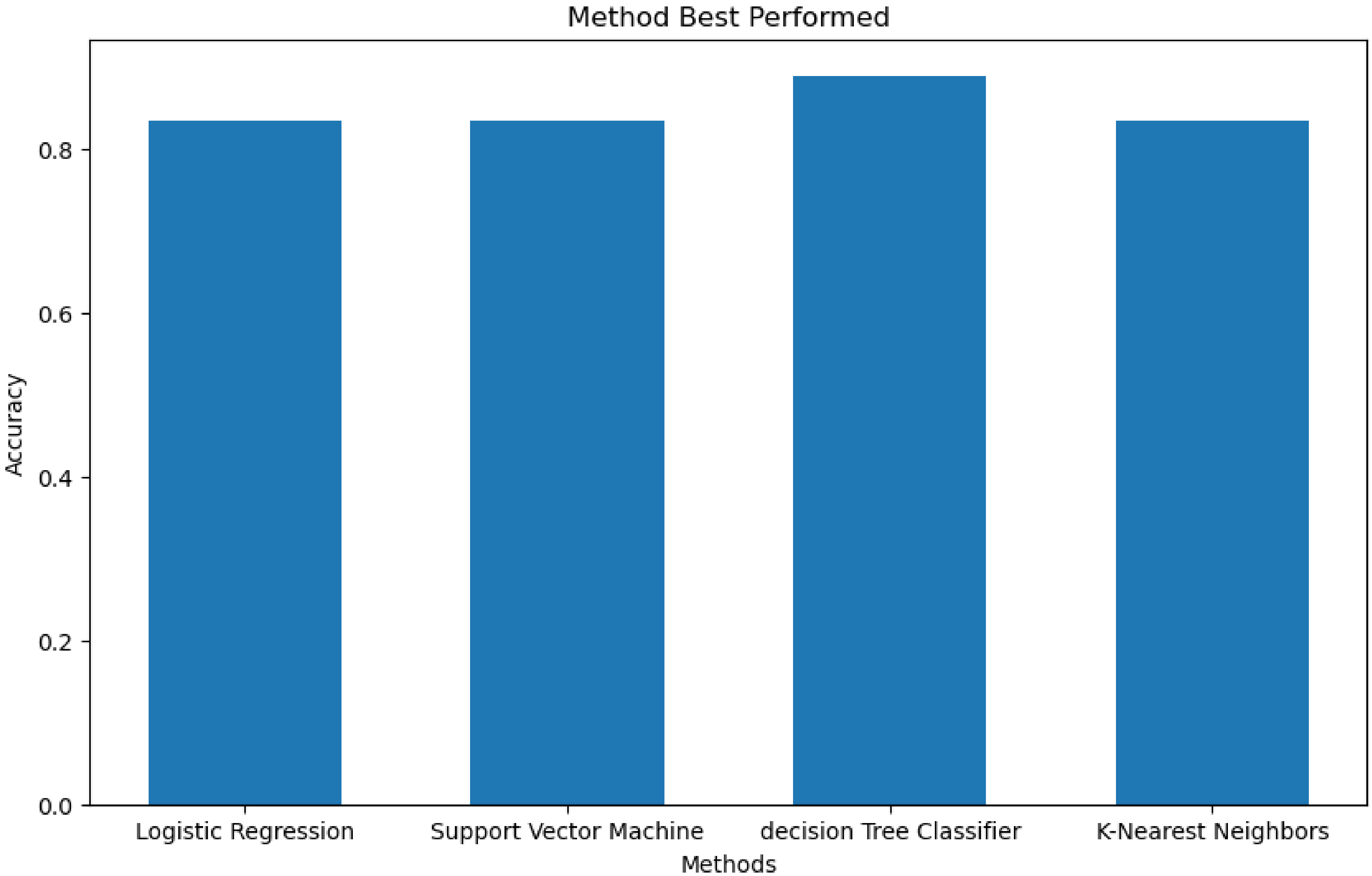


We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

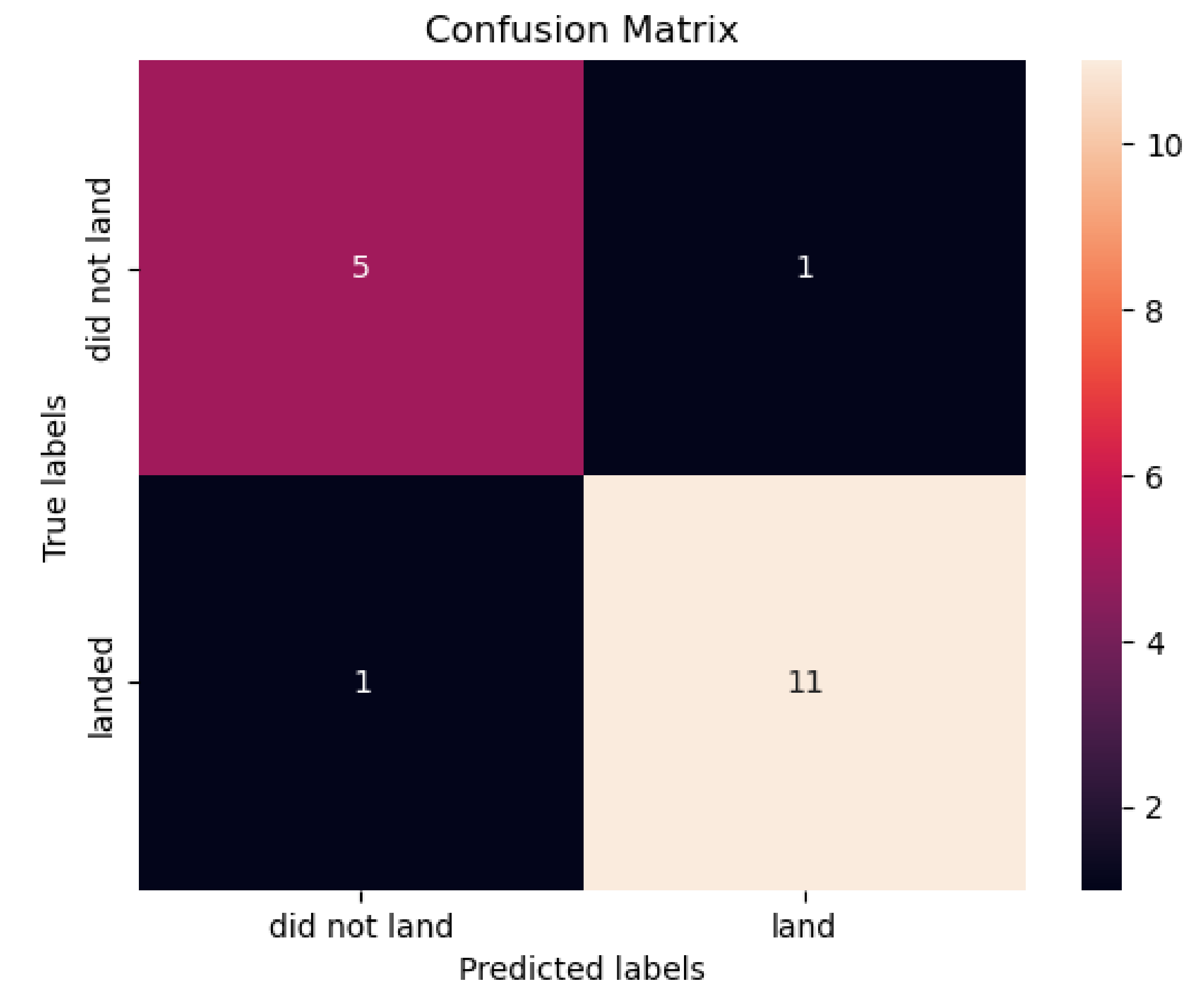
Predictive Analysis (Classification)

Classification Accuracy



The best model is Decision Tree Classifier with a score of 0.88888

Confusion Matrix



- The confusion matrix for the decision tree classifier shows that there is a very good prediction rate, only with one prediction incorrect in false negative and one false positive

Conclusions

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate has overall been successful and has only increased from 2013-2020
- Orbits ES-L1, GEO, HEO, SSO VLEO had the most success rate.
- KSC LC-39A had the most successful launches.
- Decision Tree Classifier was the best machine learning algorithm for predictive behavior due to most of our observations and predictions being related to binary values (whether a launch was successful or not).

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

Thank you!