



IBM Developer  
SKILLS NETWORK

# Winning Space Race with Data Science

Kerolos George  
30/7/2023



# Outline

---

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

# Executive Summary

---

- Summary of methodologies
- Summary of all results

# Introduction

---

- Project background and context
- Problems you want to find answers



Section 1

# Methodology

# Methodology

---

## Executive Summary

- Data collection methodology:
  - Collected the rocket launch data from the SpaceX API
  - Collect Falcon 9 historical launch records using Web scraping
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

# Data Collection

---

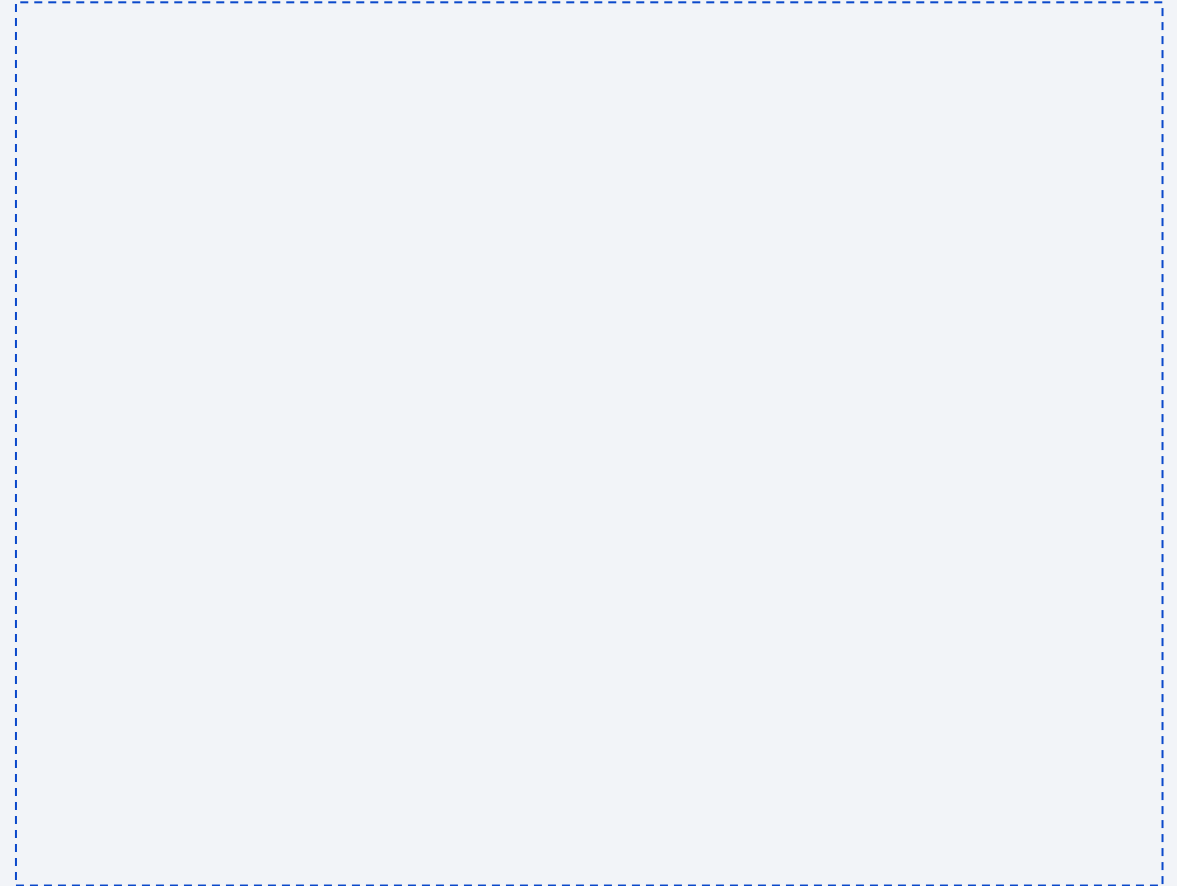
- Collected the rocket launch data from the SpaceX API
- Web-scraped the SpaceX table in the Wikipedia page to collect Falcon 9 historical launch records

# Data Collection – SpaceX API

---

. GitHub URL OF Data Collection:

<https://github.com/Kerolos-George/SpaceX-rocket/blob/main/Collecting%20the%20data.ipynb>

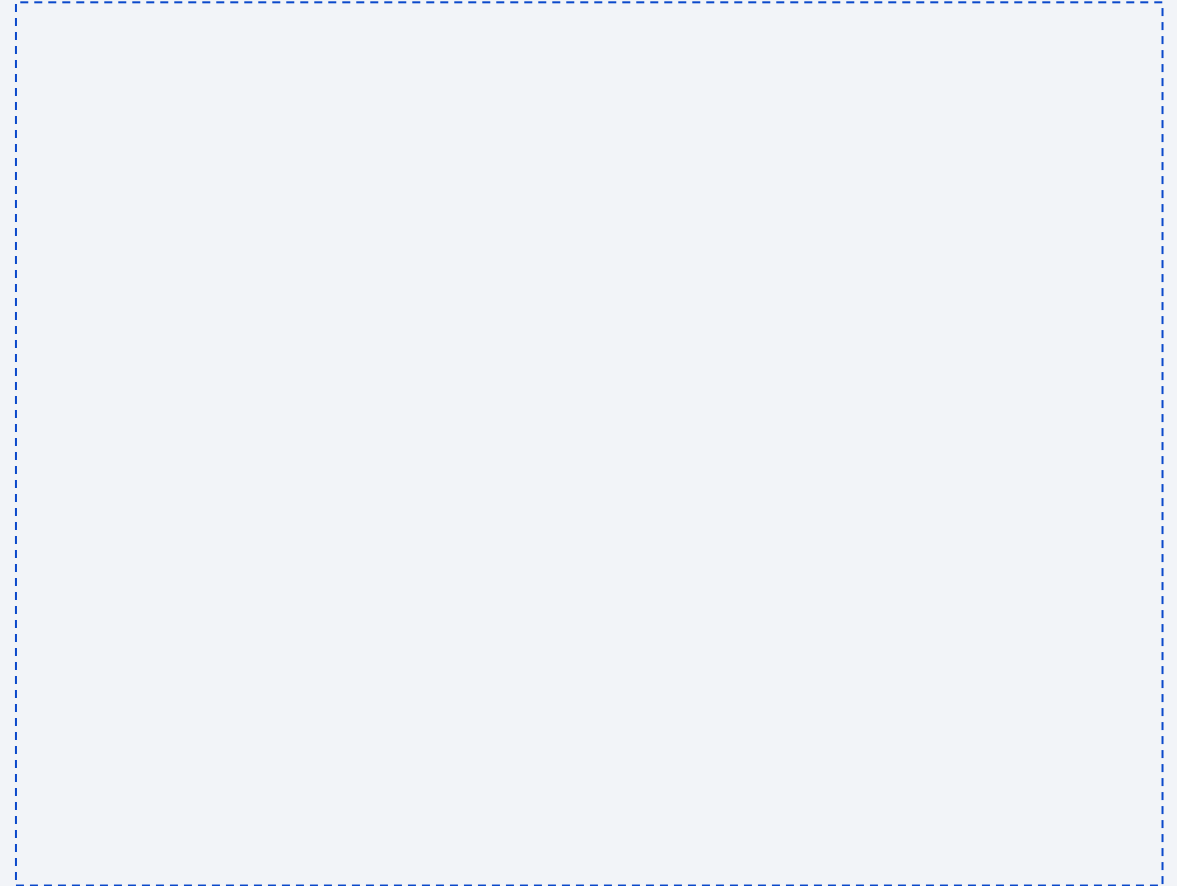




# Data Collection - Scraping

---

- The GitHub URL of the completed web scraping :
- <https://github.com/Kerolos-George/SpaceX-rocket/blob/main/webscraping.ipynb>



# Data Wrangling

---

- Calculated the number of launches on each site
  - Calculated the number and occurrence of each orbit
  - Calculated the number and occurrence of mission outcome per orbit type
  - Create a landing outcome label from Outcome column
- 
- The GitHub URL: <https://github.com/Kerolos-George/SpaceX-rocket/blob/main/Data%20wrangling.ipynb>

# EDA with Data Visualization

---

- Add the GitHub URL of your completed EDA with data visualization notebook:

<https://github.com/Kerolos-George/SpaceX-rocket/blob/main/eda-dataviz.ipynb>

# EDA with SQL

---

- Loaded the SpaceX dataset into a sqllite database.
- We applied EDA with SQL to get insight from the data. We wrote queries to find out for instance:
  - The names of unique launch sites in the space mission.
  - The total payload mass carried by boosters launched by NASA (CRS)
  - The average payload mass carried by booster version F9 v1.1
  - The total number of successful and failure mission outcomes
  - The failed landing outcomes in drone ship, their booster version and launch site names.
- Add the GitHub URL of your completed EDA with SQL notebook:

[https://github.com/Kerolos-George/SpaceX-rocket/blob/main/eda-sql\\_sqllite.ipynb](https://github.com/Kerolos-George/SpaceX-rocket/blob/main/eda-sql_sqllite.ipynb)

# Build an Interactive Map with Folium

---

- Finding an optimal location for building a launch site certainly involves many factors and hopefully we could discover some of the factors by analyzing the existing launch site locations.
- 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.
- 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 question for instance:
  - Are launch sites near railways, highways and coastlines.
  - Do launch sites keep certain distance away from cities.
- Add the GitHub URL of your completed interactive map with Folium map:

[https://github.com/Kerolos-George/SpaceX-rocket/blob/main/launch\\_site\\_location.ipynb](https://github.com/Kerolos-George/SpaceX-rocket/blob/main/launch_site_location.ipynb)

# Build a Dashboard with Plotly Dash

---

- Built an interactive dashboard with Plotly dash
- Plotted pie charts showing the total launches by a certain sites
- Plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version
- [https://github.com/Kerolos-George/SpaceX-rocket/blob/main/plotly\\_dash.py](https://github.com/Kerolos-George/SpaceX-rocket/blob/main/plotly_dash.py)



# Predictive Analysis (Classification)

---

- Loaded the data using numpy and pandas, transformed the data, split our data into training and testing.
- Built different machine learning models and tune different hyperparameters using GridSearchCV.
- Used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.
- Found the best performing classification model
- [https://github.com/Kerolos-George/SpaceX-rocket/blob/main/SpaceX Machine Learning Prediction.ipynb](https://github.com/Kerolos-George/SpaceX-rocket/blob/main/SpaceX%20Machine%20Learning%20Prediction.ipynb)

# Results

---

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





Section 2

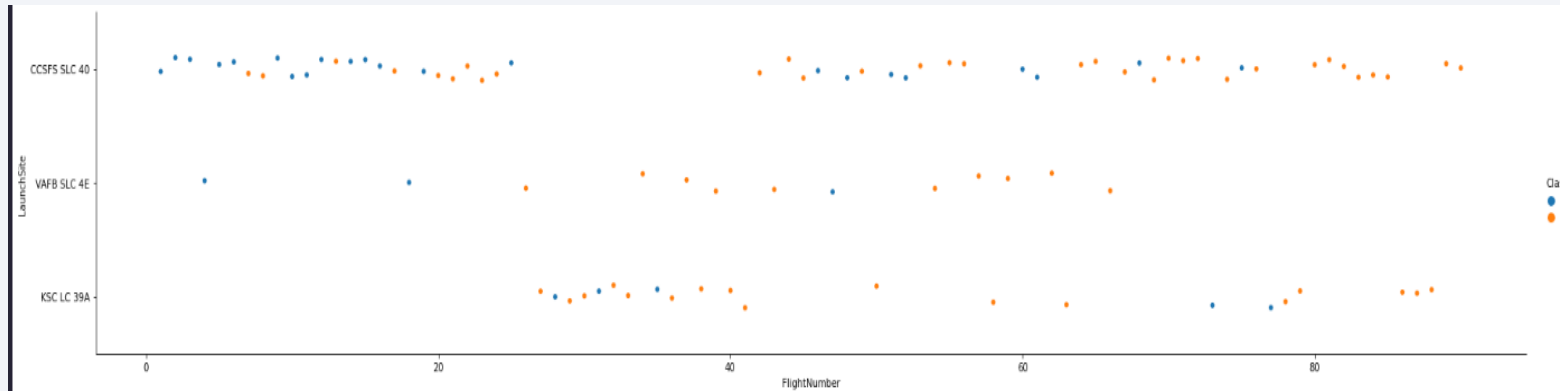
# Insights drawn from EDA



# Flight Number vs. Launch Site

---

- Show a scatter plot of Flight Number vs. Launch Site

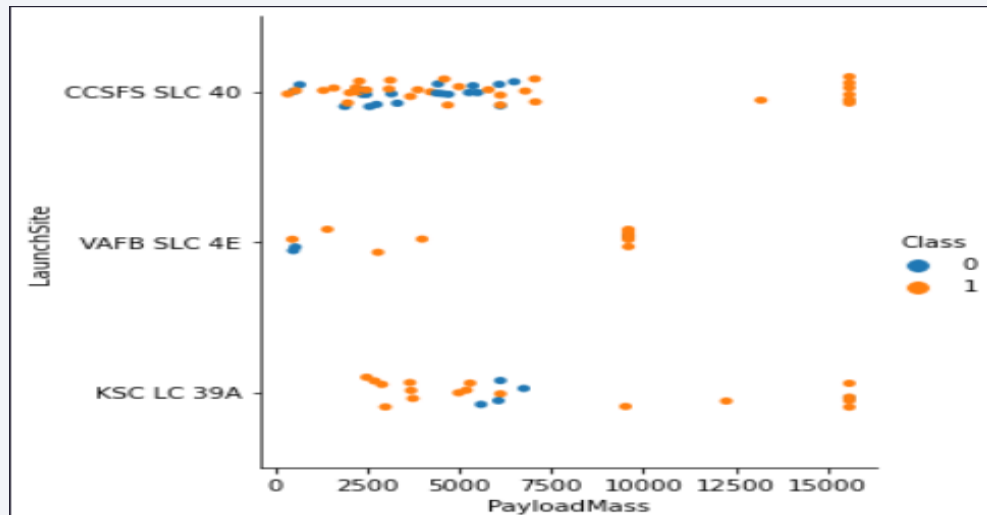


- **Explanation:** when the flight number is  $> 80$  the success rate is 100%

# Payload vs. Launch Site

---

- Show a scatter plot of Payload vs. Launch Site

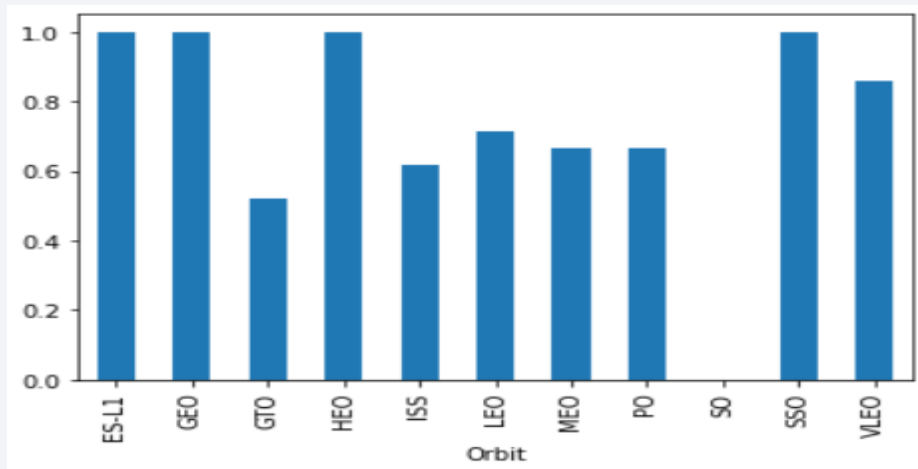


no rockets launched for heavypayload  
mass(greater than 10000)

# Success Rate vs. Orbit Type

---

- Show a bar chart for the success rate of each orbit type

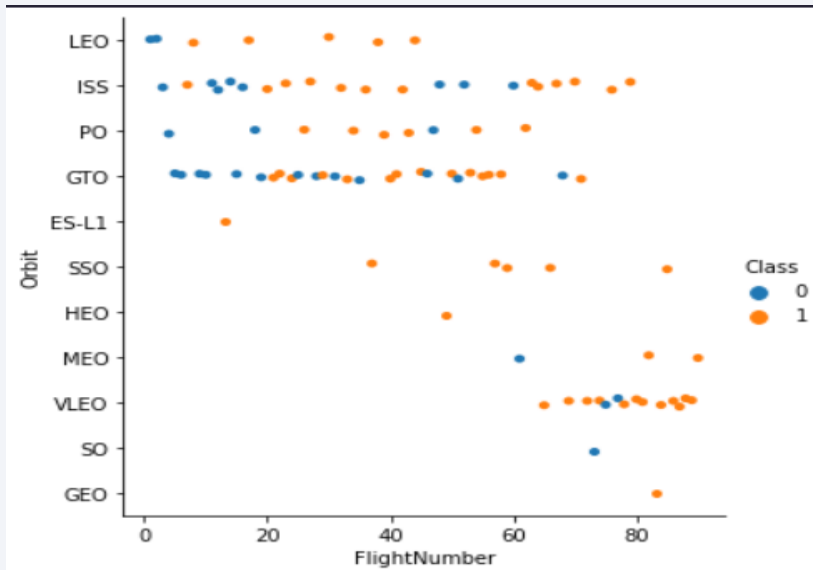


- **Explanation:** The ES-L1, GEO, HEO and SSO orbit have the maximum success rate



# Flight Number vs. Orbit Type

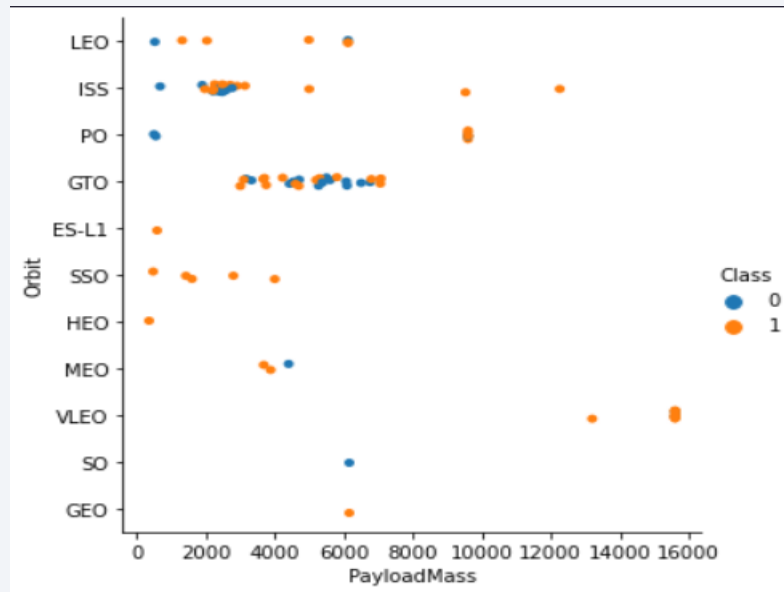
- Show a scatter point of Flight number vs. Orbit type



# Payload vs. Orbit Type

---

- Show a scatter point of payload vs. orbit type



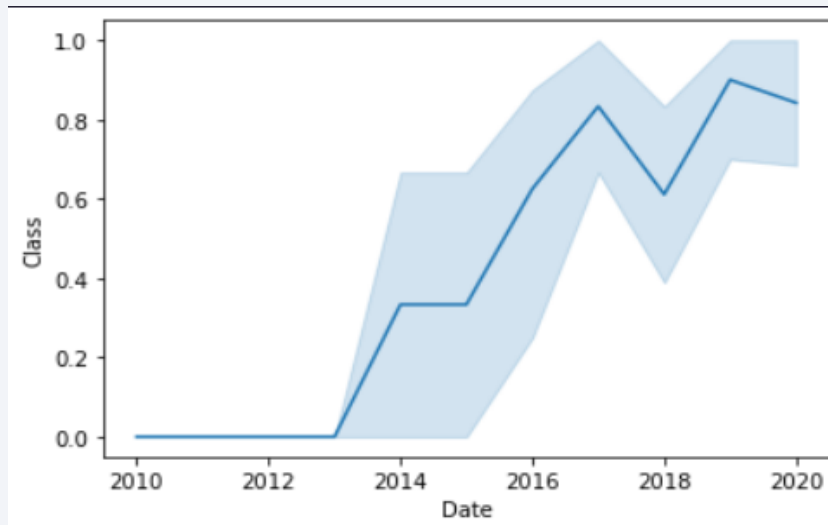
- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccesful mission) are both there here.

# Launch Success Yearly Trend

---

- Show a line chart of yearly average success rate



- **Explanation:** the success rate since 2013 kept increasing till 2020

# All Launch Site Names

---

Present your query result with a short explanation here

```
%sql SELECT DISTINCT Launch_Site FROM SPACEX_DATA
```

```
* sqlite:///SpaceX.db
```

```
Done.
```

```
Launch_Site
```

```
CCAFS LC-40
```

```
VAFB SLC-4E
```

```
KSC LC-39A
```

```
CCAFS SLC-40
```

# Launch Site Names Begin with 'CCA'

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

```
%sql SELECT * FROM SPACEX_DATA WHERE Launch_Site LIKE 'CCA%' LIMIT(5)
```

Python

```
* sqlite:///SpaceX.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

# Total Payload Mass

---

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

```
%sql SELECT SUM(PAYLOAD_MASS_KG_) as Total_Payload_Mass FROM SPACEX_DATA WHERE Customer == 'NASA (CRS)'
```

```
* sqlite:///SpaceX.db
```

```
Done.
```

```
Total_Payload_Mass
```

```
45596
```



# Average Payload Mass by F9 v1.1

---

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

```
%sql SELECT AVG(PAYLOAD_MASS_KG_) as Avg_Payload_Mass FROM SPACEX_DATA WHERE Booster_Version == 'F9 v1.1'
```

[21]

```
... * sqlite:///SpaceX.db
```

Done.

</>

Avg\_Payload\_Mass

2928.4

# First Successful Ground Landing Date

---

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

```
%sql SELECT MIN(Date) AS FirstSuccessfull_landing_date FROM SPACEX_DATA WHERE [Landing_Outcome] == 'Success (ground pad)'
```

```
* sqlite:///SpaceX.db
```

```
Done.
```

```
FirstSuccessfull_landing_date
```

```
01-05-2017
```

## Successful Drone Ship Landing with Payload between 4000 and 6000

---

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 SPACEX_DATA WHERE [Landing_Outcome] = 'Success (drone ship)' AND PAYLOAD_MASS_KG_ > 4000 AND PAYLOAD_MASS_KG_ < 6000
```

Python

```
* sqlite:///SpaceX.db
```

Done.

Booster\_Version

F9 FT B1022

F9 FT B1026

F9 FT B1021.2

F9 FT B1031.2

# Total Number of Successful and Failure Mission Outcomes

---

List the total number of successful and failure mission outcomes

```
%sql SELECT (SELECT COUNT(*) FROM SPACEX_DATA WHERE [Landing _Outcome] LIKE 'Success%') as Success_outcomes, (SELECT COUNT(*) FROM SPACEX_DATA WHERE [Landing _Outcome] LIKE 'Failure%') as Failure_outcomes
```

Python

```
* sqlite:///SpaceX.db
```

Done.

Success_outcomes	Failure_outcomes
------------------	------------------

61	10
----	----

# Boosters Carried Maximum Payload

List the names of the booster\_versions which have carried the maximum payload mass. Use a subquery

```
%sql SELECT Booster_Version, PAYLOAD_MASS_KG_ FROM SPACEX_DATA WHERE PAYLOAD_MASS_KG_ == (SELECT MAX(PAYLOAD_MASS_KG_) FROM SPACEX_I
```

Python

```
* sqlite:///SpaceX.db
```

Done.

Booster_Version	PAYLOAD_MASS_KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

# 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



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

---

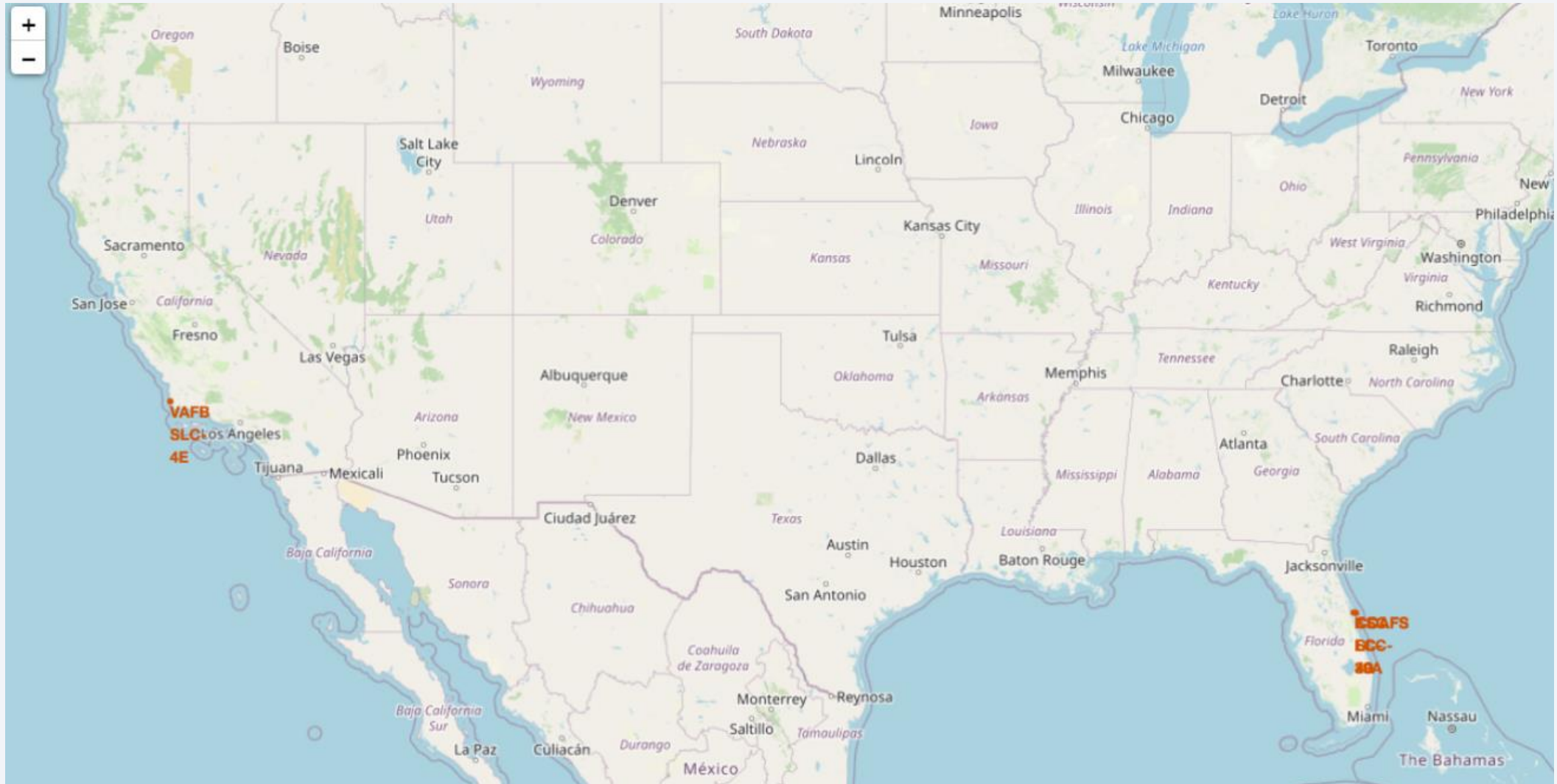
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The image is a composite of a dark blue sky and a view of the Earth's surface, which is covered in a dense network of city lights and clouds. The lights are concentrated in the lower right portion of the image, while the upper left shows a clear blue sky.

Section 3

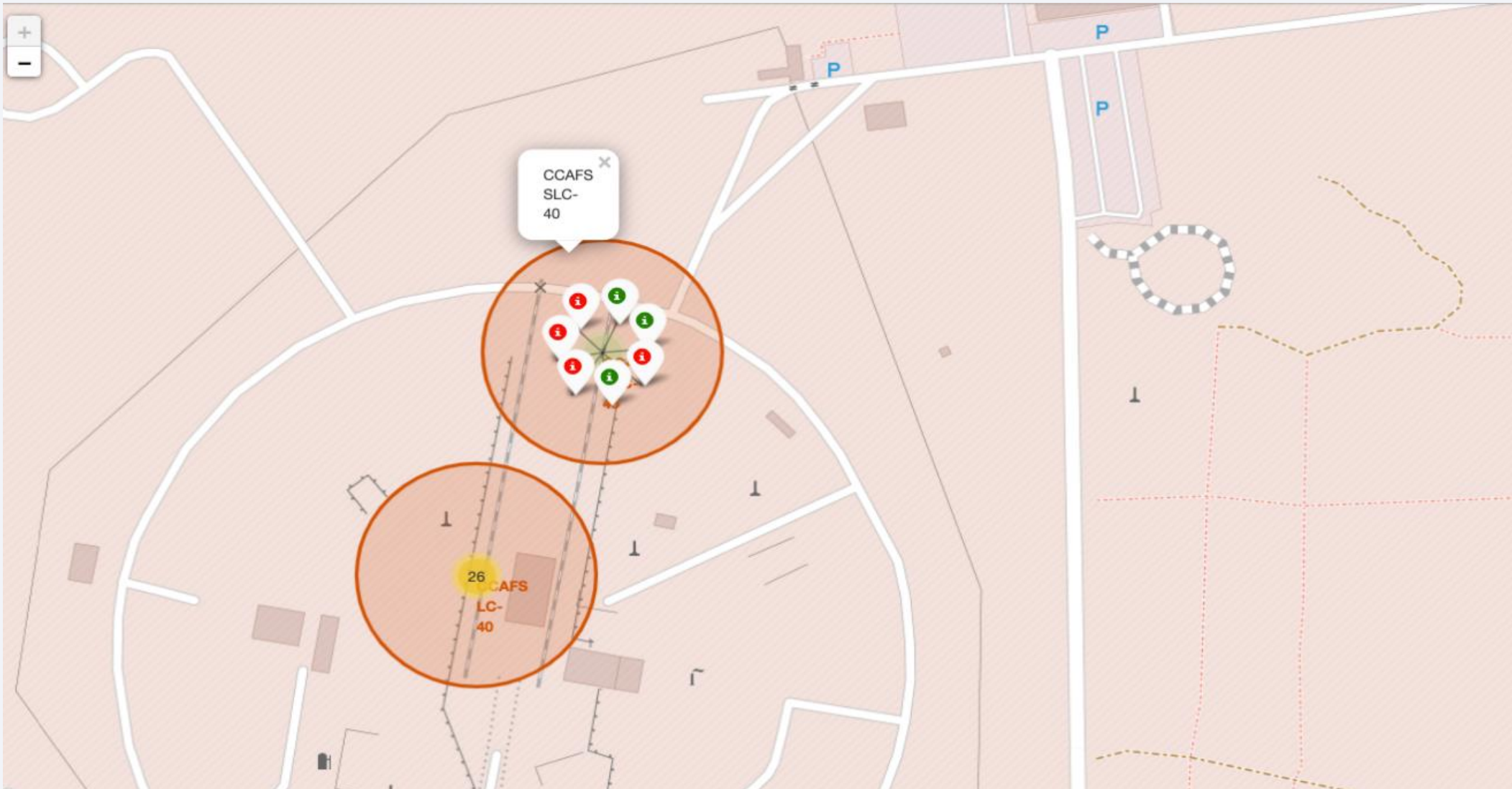
# Launch Sites Proximities Analysis

# <Folium Map Screenshot 1>

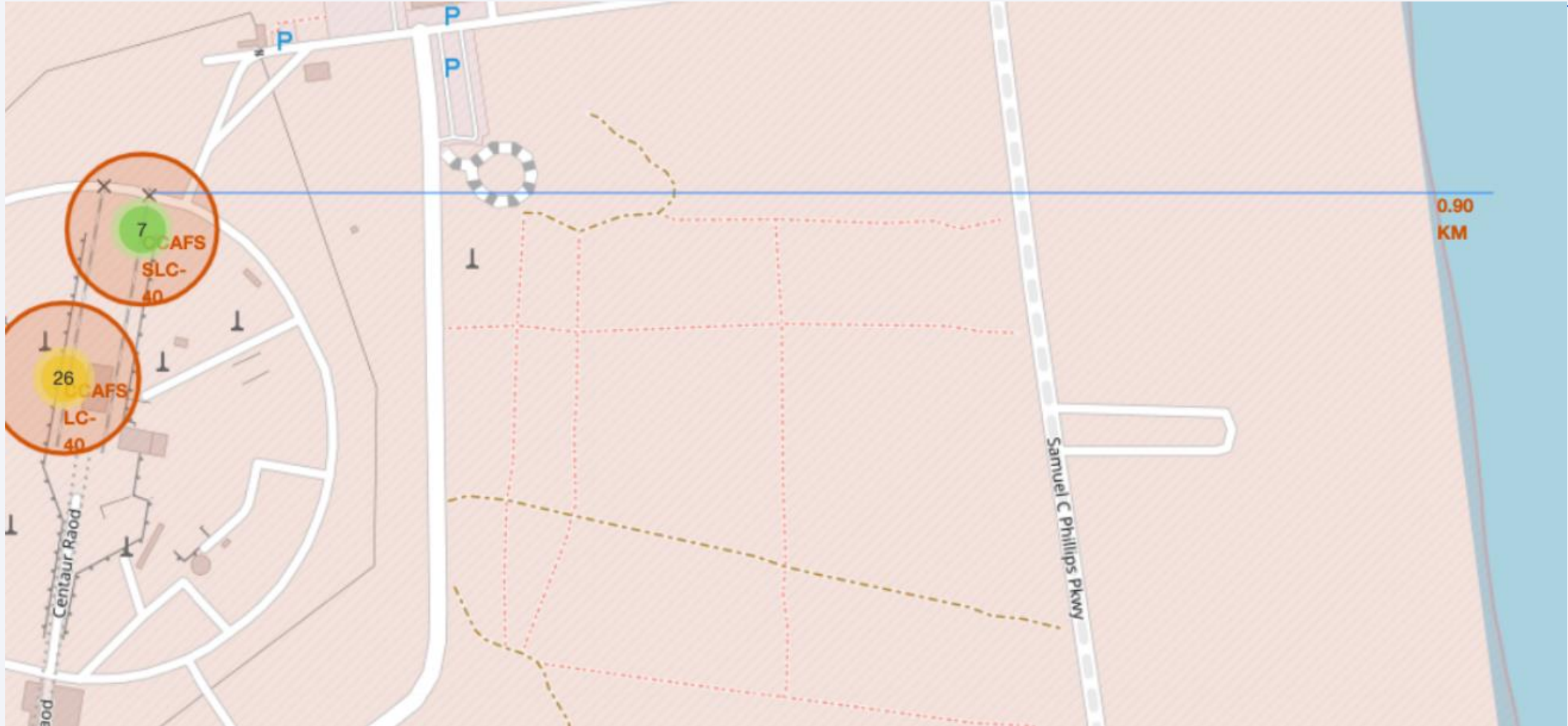




## <Folium Map Screenshot 2>



## <Folium Map Screenshot 3>





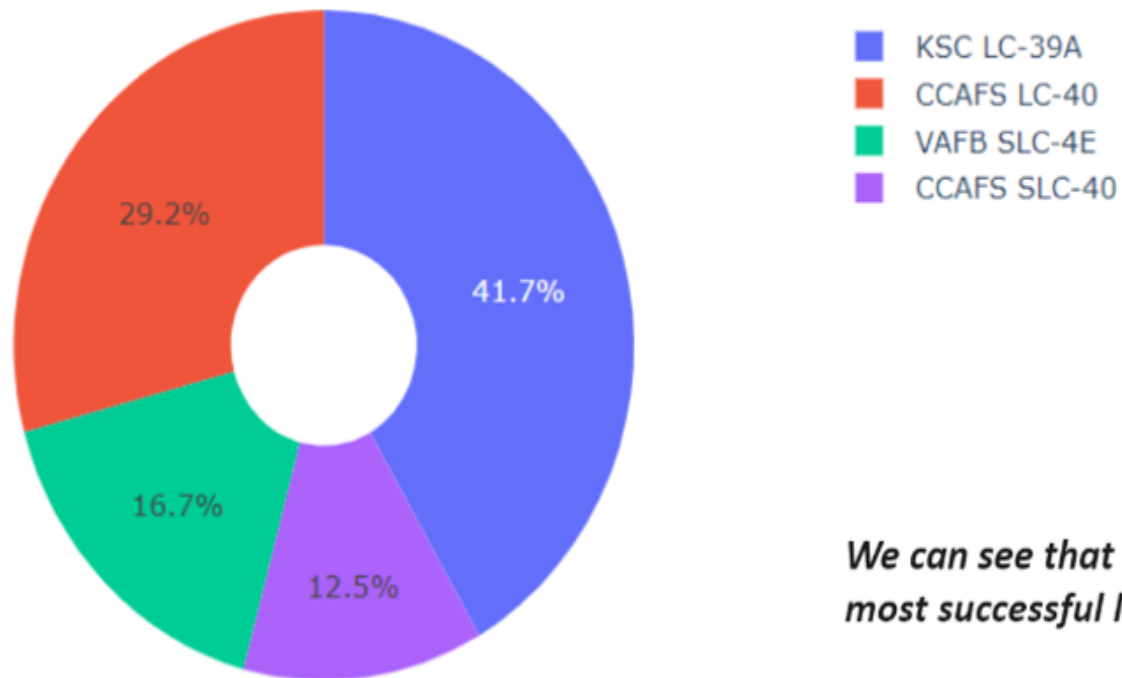


Section 4

# Build a Dashboard with Plotly Dash

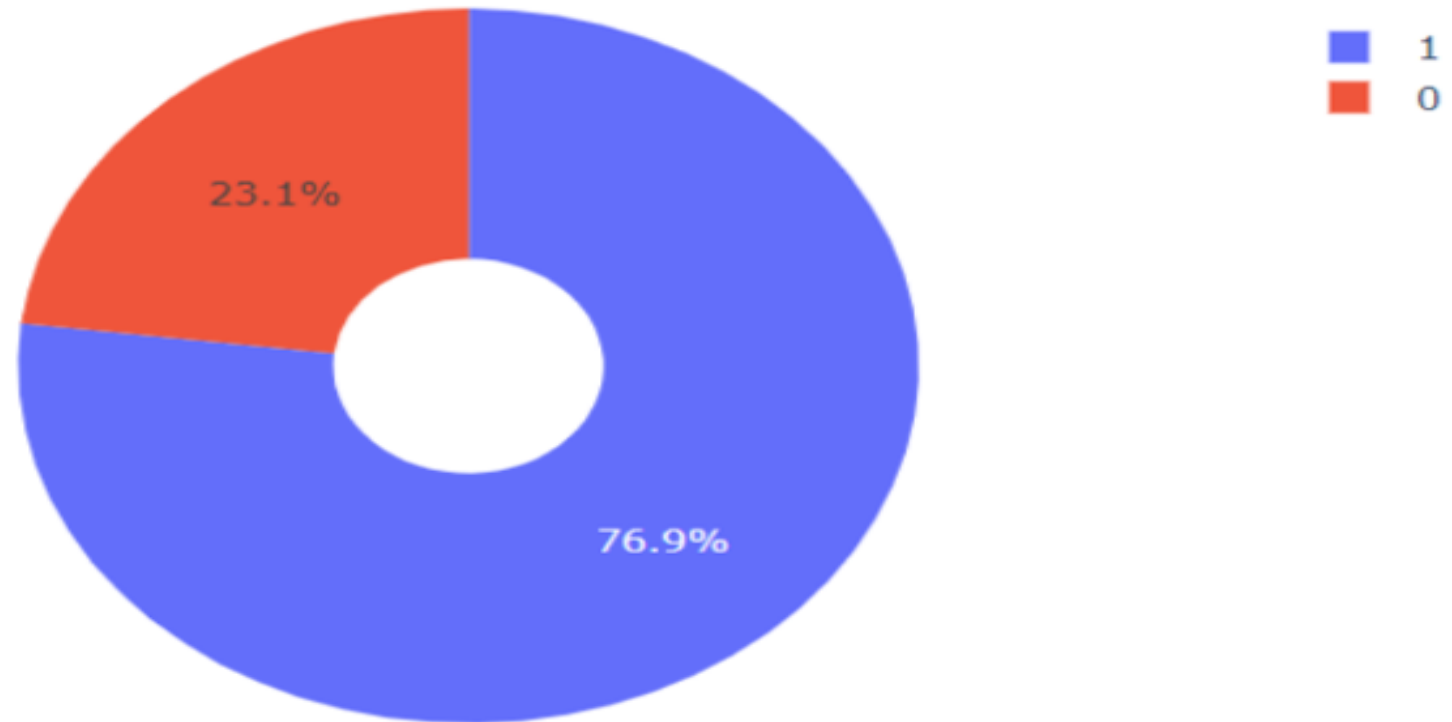
# <Dashboard Screenshot 1>

Total Success Launches By all sites



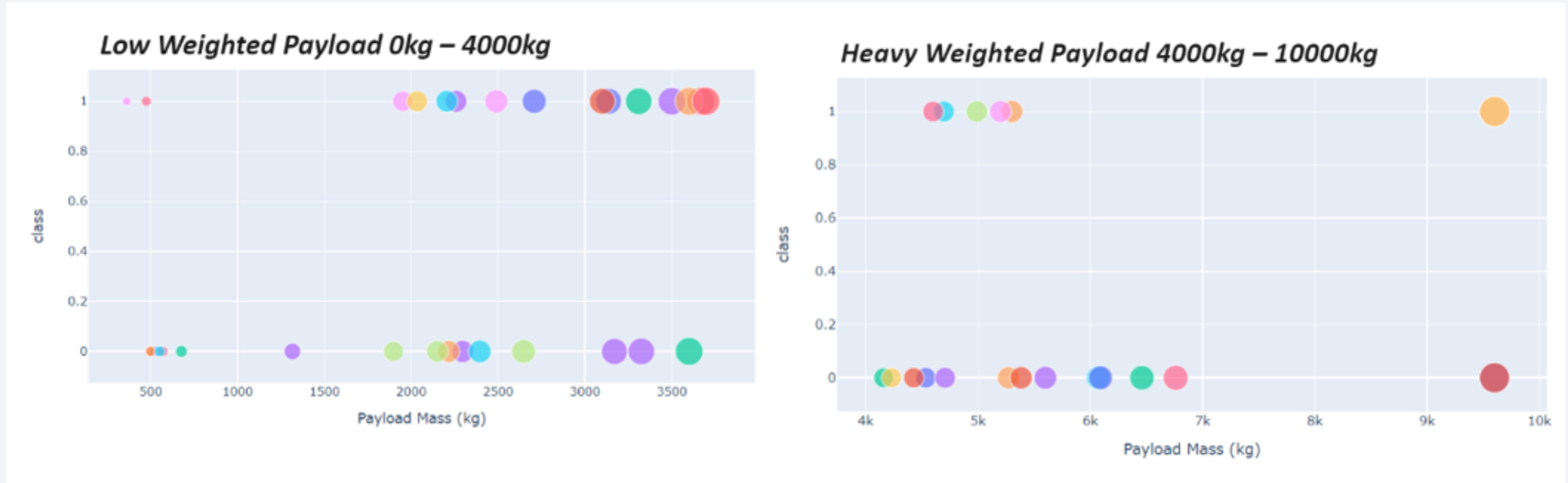
*We can see that KSC LC-39A had the most successful launches from all the sites*

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





# <Dashboard Screenshot 3>



Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

---

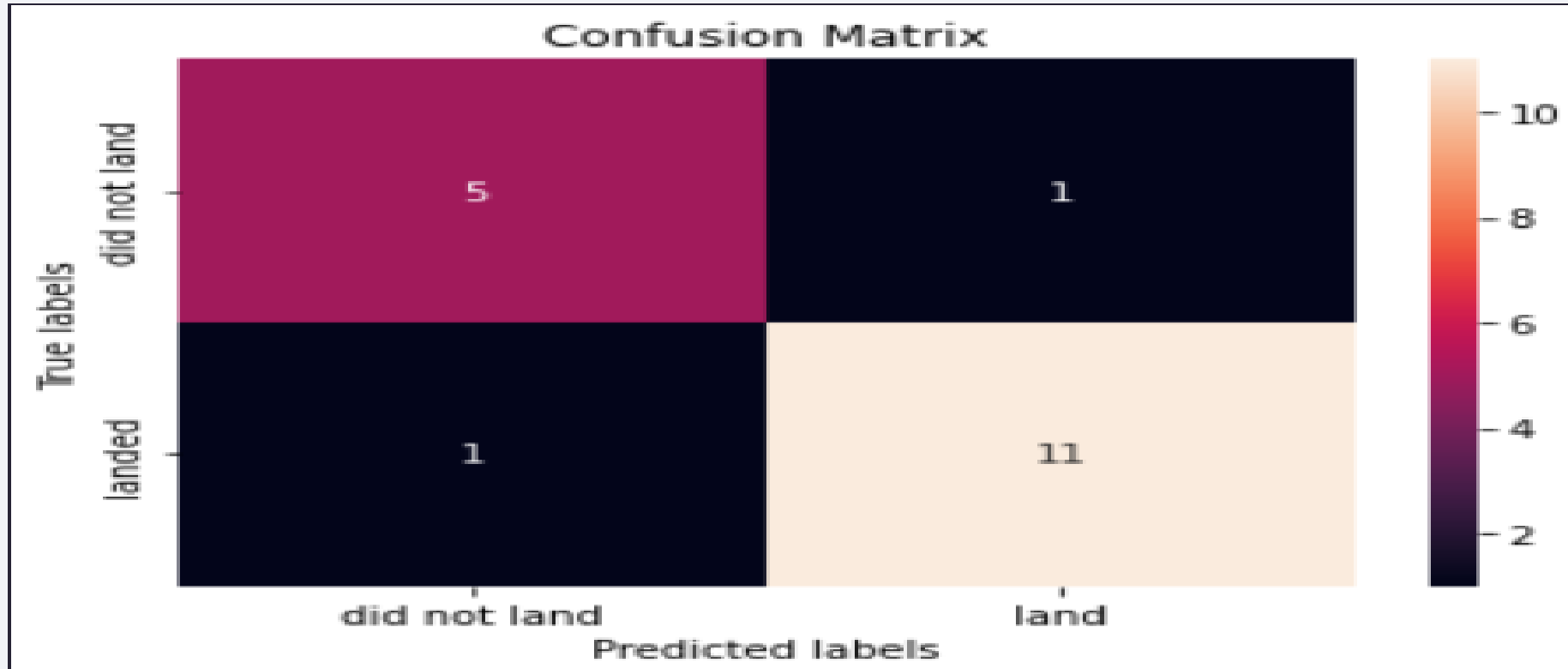
- Visualize the built model accuracy for all built classification models, in a bar chart



- Find which model has the highest classification accuracy ;  
KNN

# Confusion Matrix

---



# Conclusions

---

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate started to increase in 2013 till 2020.
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- KSC LC-39A had the most successful launches of any sites.
- The KNNclassifier is the best machine learning algorithm for this task

# Appendix

---

- <https://github.com/Kerolos-George/SpaceX-rocket>

Thank you!

