IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

LIMOUNI Elias
09/15/2022

# Outline

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

# Executive Summary

- Summary of methodologies
  - Data collection and wrangling
  - EDA with SQL and visualization
  - Creation of an interactive map with Folium and a dashboard with Plotly Dash
  - Predictive analysis using several classifiers and parameters optimization with GridSearchCV
- Summary of all results
  - EDA results
  - Interactive analysis
  - Predictive analysis

# Introduction

- Project background and context
  - SpaceX intends to reduce the costs of spatial flights by reusing the first stage of their rockets.
- Problems you want to find answers
  - This project goal is to predict if the first stage of a Falcon 9 rocket will lang back to earth successfully.

Section 1

# Methodology

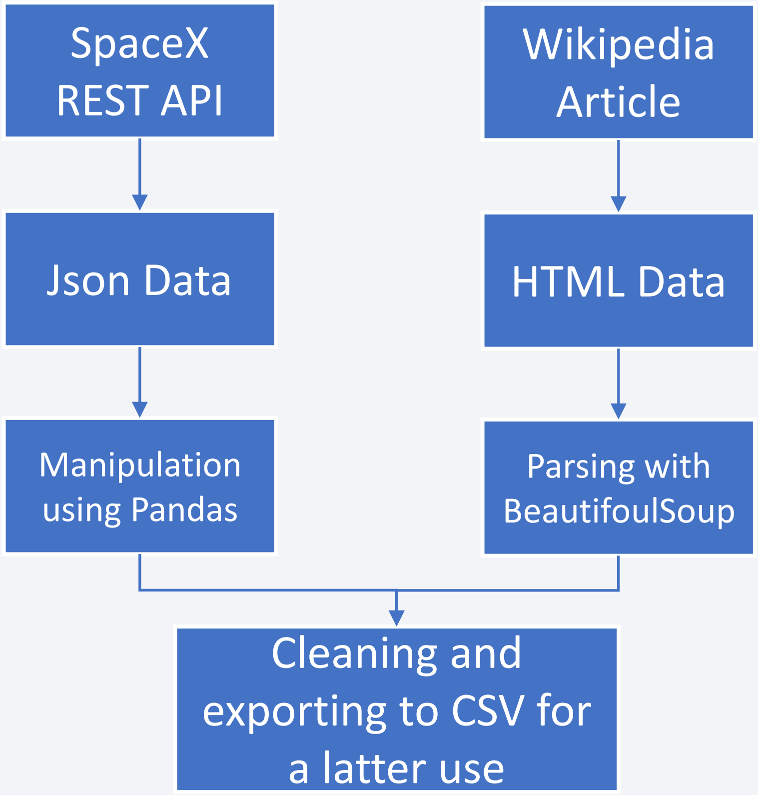
# Methodology

## Executive Summary

- Data collection methodology
  - SpaceX Rest API
  - Wikipedia website using web scrapping
- Perform data wrangling
  - OneHotEncoding categorical fields, replacing missing values by the mean value, scaling the numerical values using a StandardScaler
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Comparing 4 models Linear Regression, K-Nearest Neighbours, SVM, and Decision Tree with optimized parameters using GridSearchCV, and plotting plotting the confusion matrix on the training set and the accuracy on the testing set.

# Data Collection

- Two datasets were collected:
  - One from SpaceX REST API, containing the launch data :
    - Booster version
    - Payload mass
    - Launching site
    - Landing specifications
    - Landing outcome
  - The second by performing web scrapping on Wikipedia.



# Data Collection – SpaceX API

- Data collection with SpaceX REST API

- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%201%20-%20Data%20Collection%20API%20Lab.ipynb)

https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%201%20-%20Data%20Collection%20API%20Lab.ipynb

**API call**

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

```
response = requests.get(spacex_url)
```

**Converting Json to DataFrame**

```
# Use json_normalize method to convert the json result into a dataframe
data = pd.json_normalize(response.json())
```

**Data Cleaning**

```
# Call getLaunchSite
getLaunchSite(data)
```

```
# Call getPayloadData
getPayloadData(data)
```

```
# Call getCoreData
getCoreData(data)
```

**Data formatting**

```
data_falcon9 = df[df['BoosterVersion']!='Falcon 1']
data_falcon9.head()
```

**CSV Exportation**

```
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

# Data Collection - Scraping

- Data collection by Web Scrapping a Wikipedia article

- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%201%20-%20Data%20Collection%20with%20Web%20Scraping%20Lab.ipynb)

https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%201%20-%20Data%20Collection%20with%20Web%20Scraping%20Lab.ipynb

**Website call**

```python
# assign the response to a object
response = requests.get(static_url)

soup = BeautifulSoup(response.content, "html.parser")
```

**Converting HTML table to DF**

```python
for element in first_launch_table.find_all('th'):
    name = extract_column_from_header(element)
    if name is not None and len(name) > 0:
        column_names.append(name)
```

**Data Cleaning**

```python
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']
```

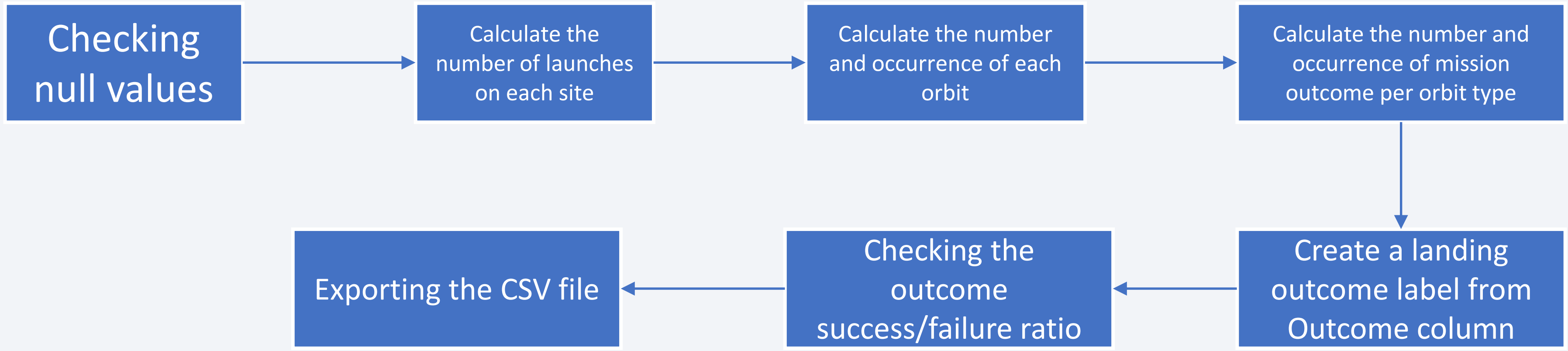
**Data formatting**

```python
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
    # get table row
    for rows in table.find_all("tr"):
```

**CSV Exportation**

```python
df.to_csv('spacex_web_scraped.csv', index=False)
```

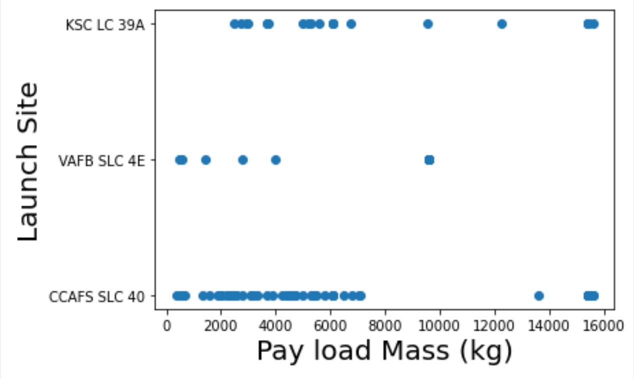
# Data Wrangling

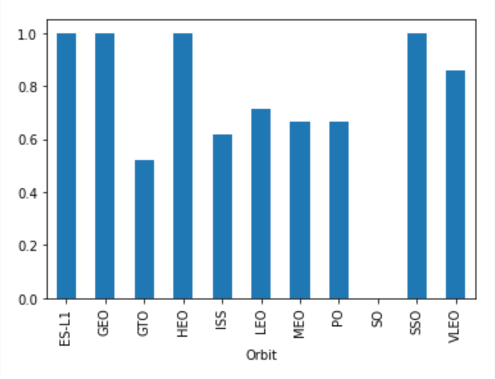


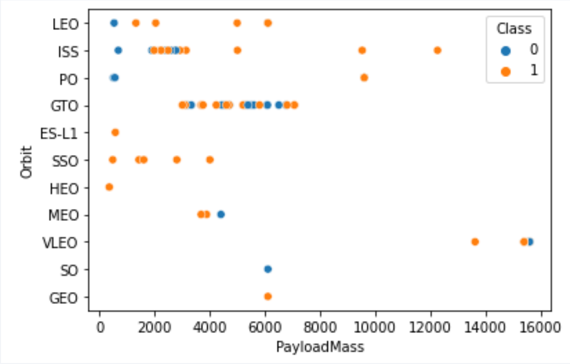

- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%201%20-%20Data%20Collection%20with%20Web%20Scraping%20Lab.ipynb)

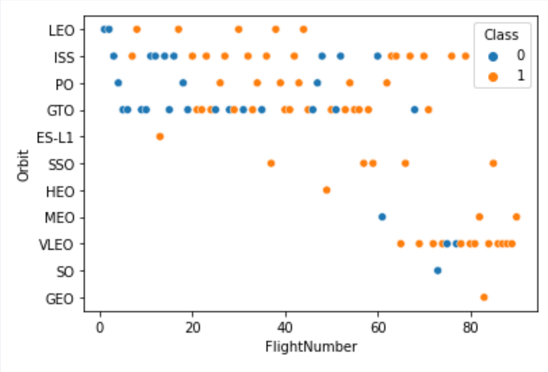
https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%201%20-%20Data%20Collection%20with%20Web%20Scraping%20Lab.ipynb

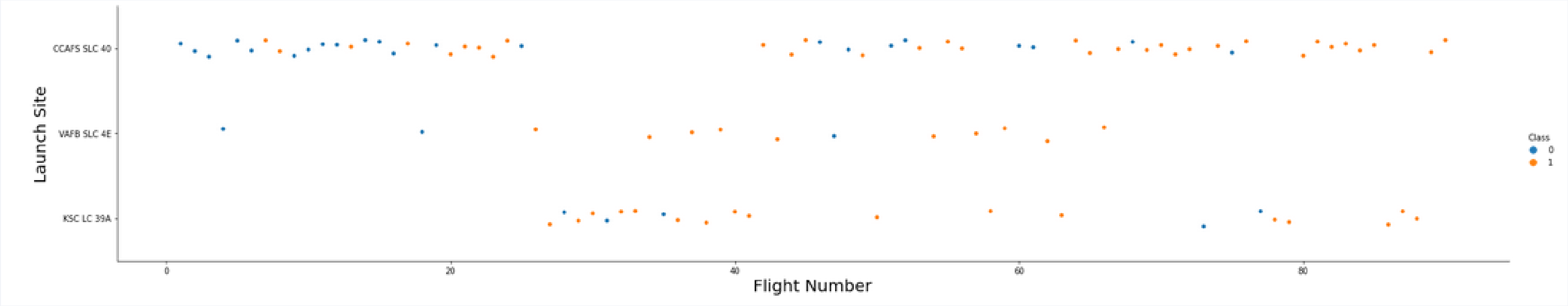
# EDA with Data Visualization

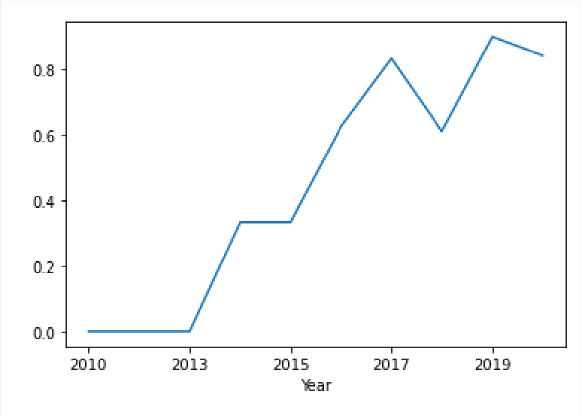



- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%202%20-%20Exploratory%20Data%20Analysis%20with%20Plots%20Lab.ipynb)

https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%202%20-%20Exploratory%20Data%20Analysis%20with%20Plots%20Lab.ipynb

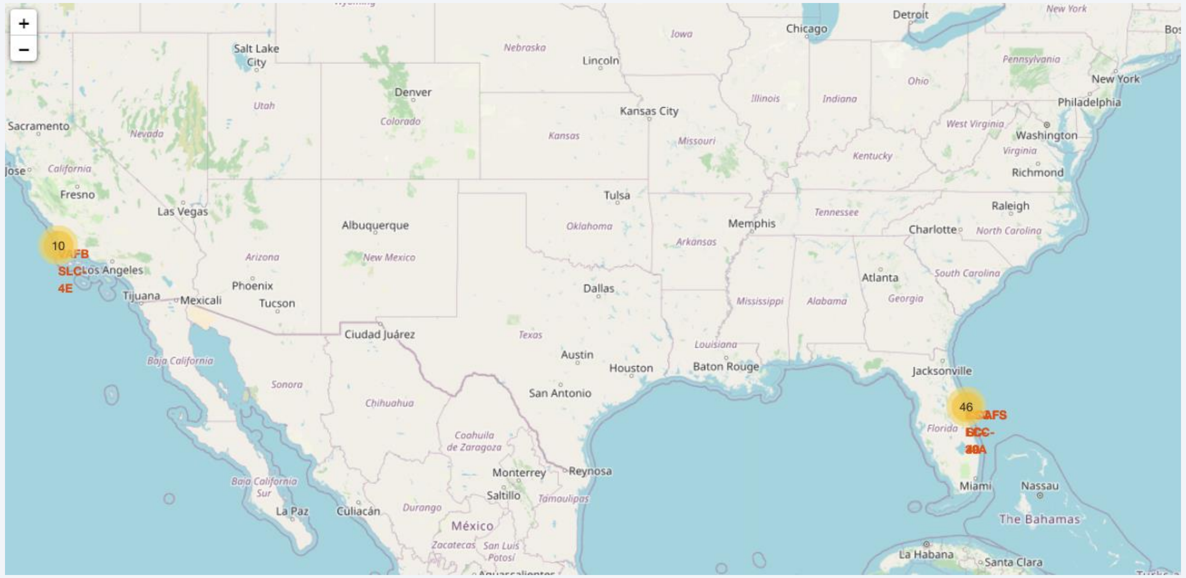
# EDA with SQL

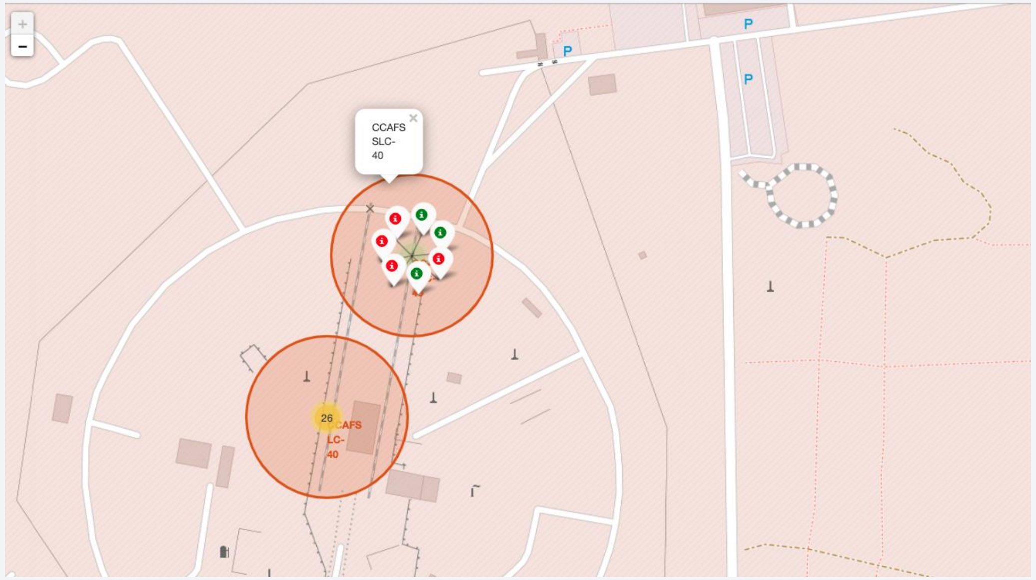
- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first successful landing outcome in ground pad was acheived.
- 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 using 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 between the date 2010-06-04 and 2017-03-20, in descending order

- Link to the repository

https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%202%20-%20Exploratory%20Data%20Analysis%20with%20SQL%20Lab.ipynb

# Build an Interactive Map with Folium



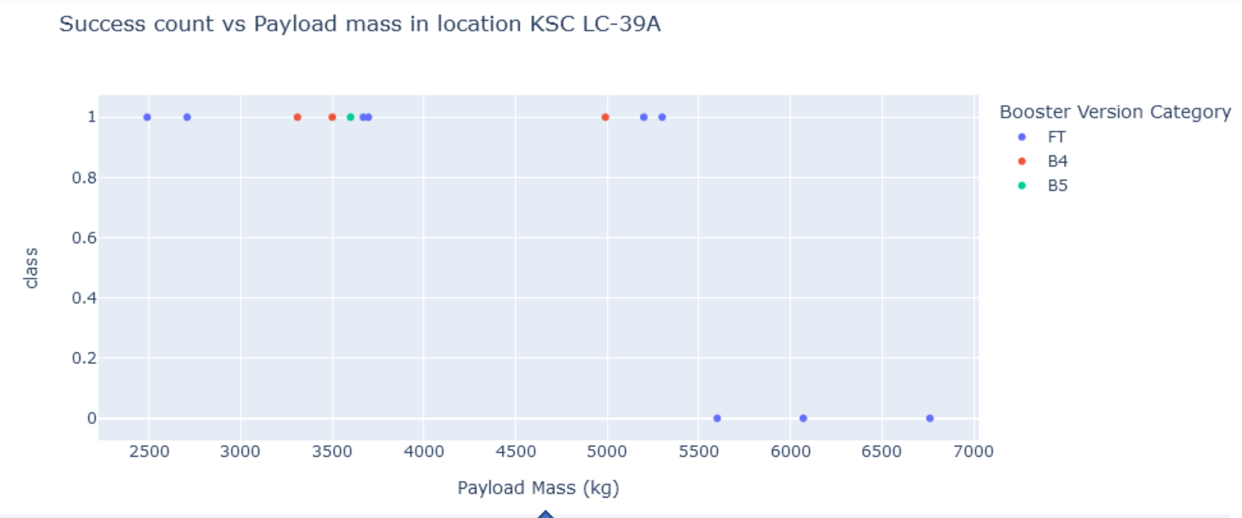




Markers were added to the sites to see the proportion of success for each one, and the Circles describes the launching sites, in order to have an overview of each site

- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%203%20-%20Interactive%20Visual%20Analytics%20with%20Folium%20Lab.ipynb)

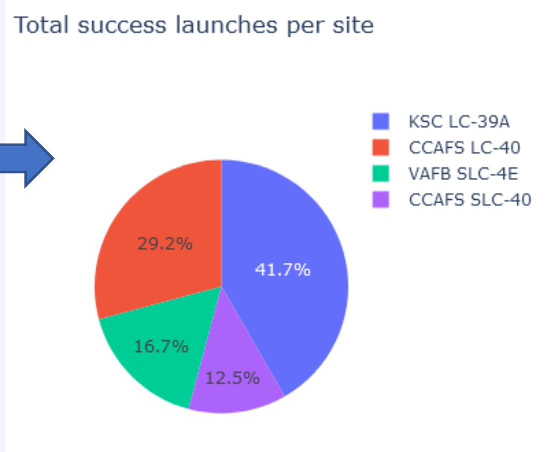
https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%203%20-%20Interactive%20Visual%20Analytics%20with%20Folium%20Lab.ipynb

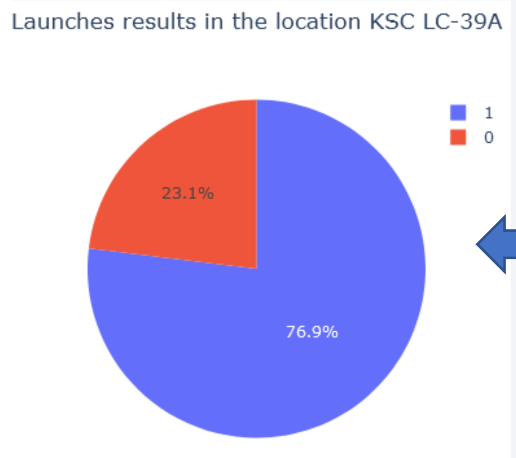
# Build a Dashboard with Plotly Dash




Payload mass and booster version for a location or for all locations to have an overview of the impact of each feature on the outcome

Seeing the most frequent site with success



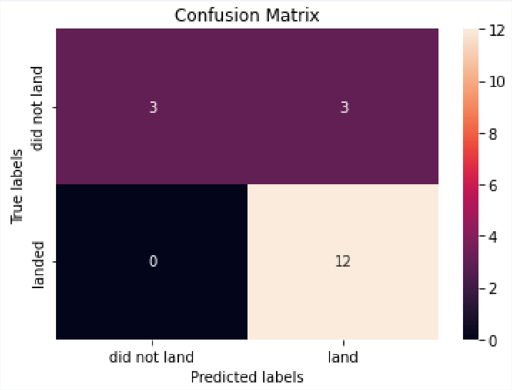

Getting the proportion of success for each location

- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/spacex_dash_app.py)

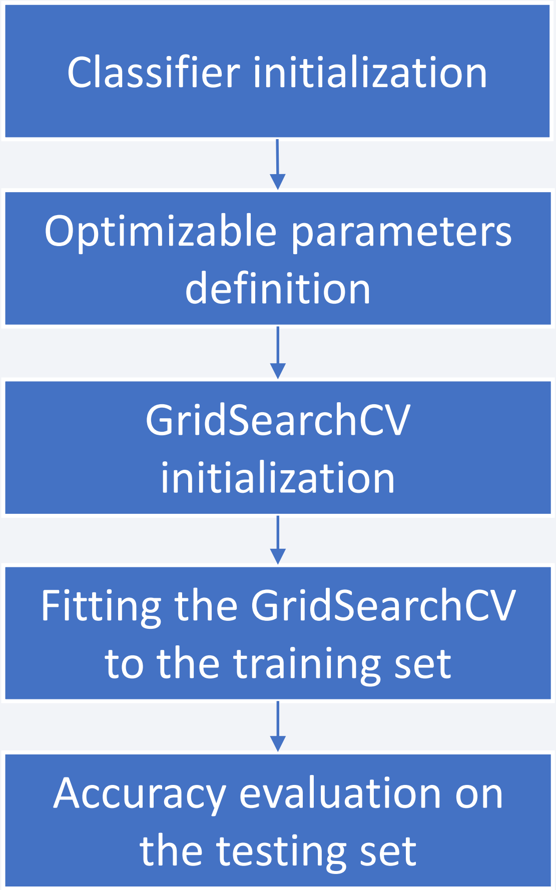
https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/spacex_dash_app.py

# Predictive Analysis (Classification)

- Several models studied
  - Logistic Regression, SVM, KNN, and Decision Tree
- A confusion matrix built on the training test is displayed for each model for to increase readability




- [Link to the repository](https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%204%20-%20Machine%20Learning%20Prediction%20lab.ipynb)




https://github.com/eliaccess/Applied-Data-Science-Capstone/blob/master/Week%204%20-%20Machine%20Learning%20Prediction%20lab.ipynb

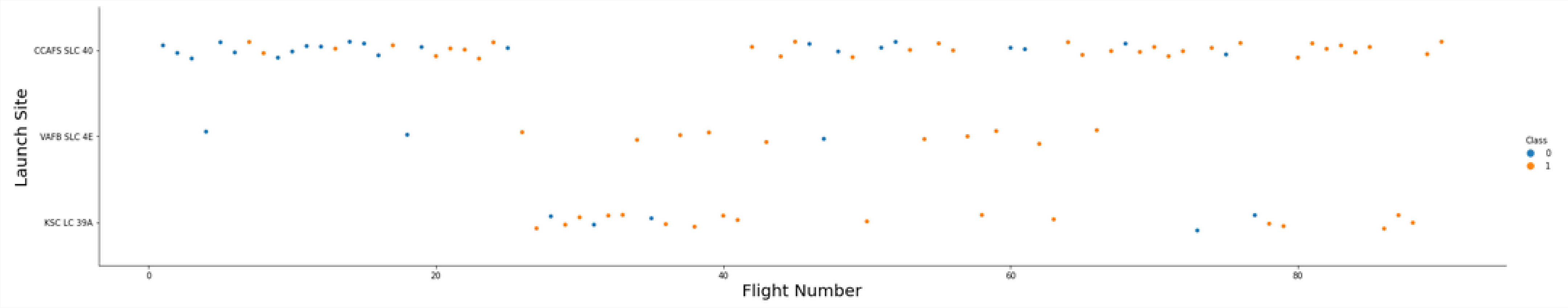
# Results

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

Section 2

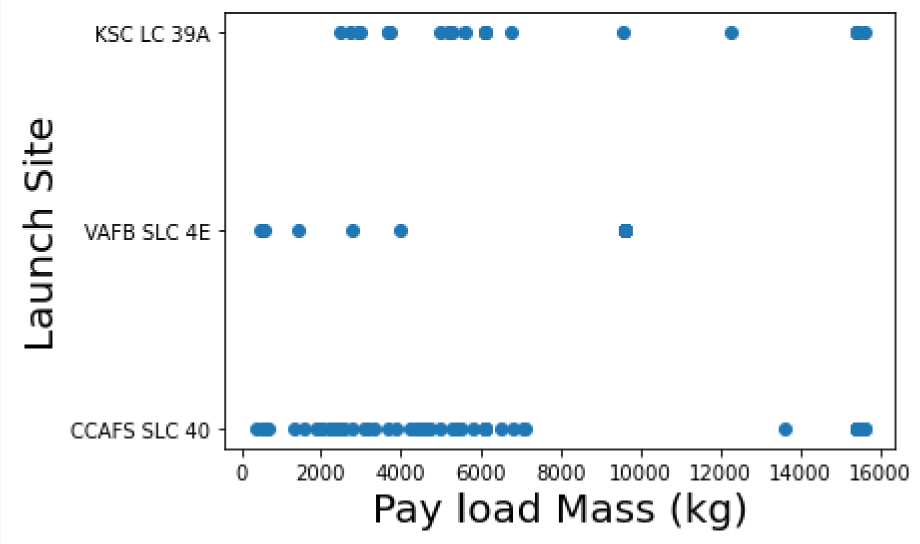
# Insights drawn from EDA

# Flight Number vs. Launch Site



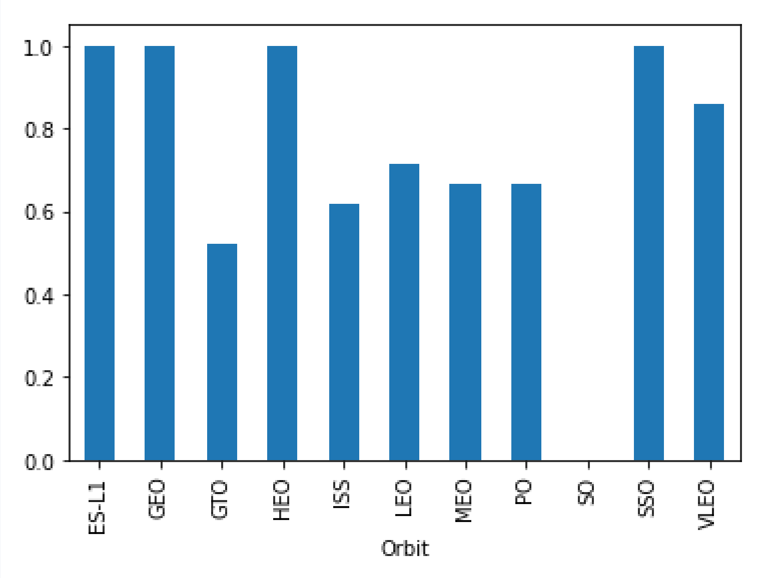

CCAFS SLC 40 seems to be the one with the most successful outcomes.

# Payload vs. Launch Site



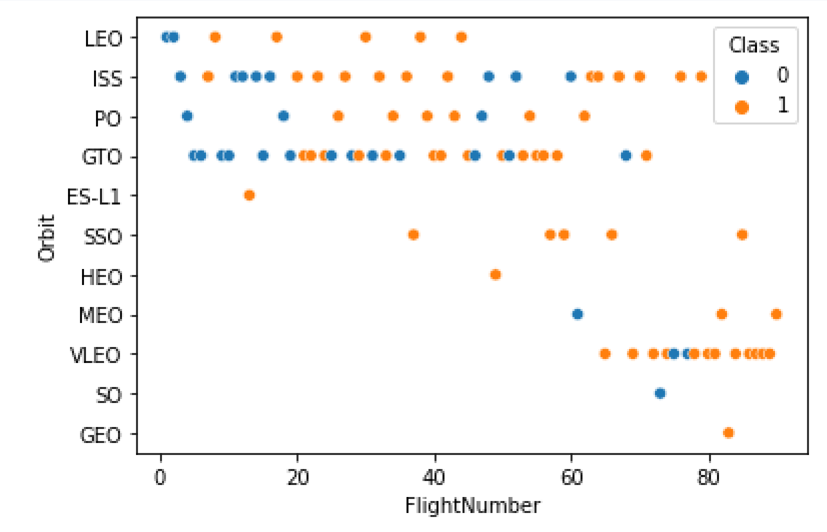

CCAFS SLC 40 have a highly dense launches with a lighter payload mass in average.

# Success Rate vs. Orbit Type



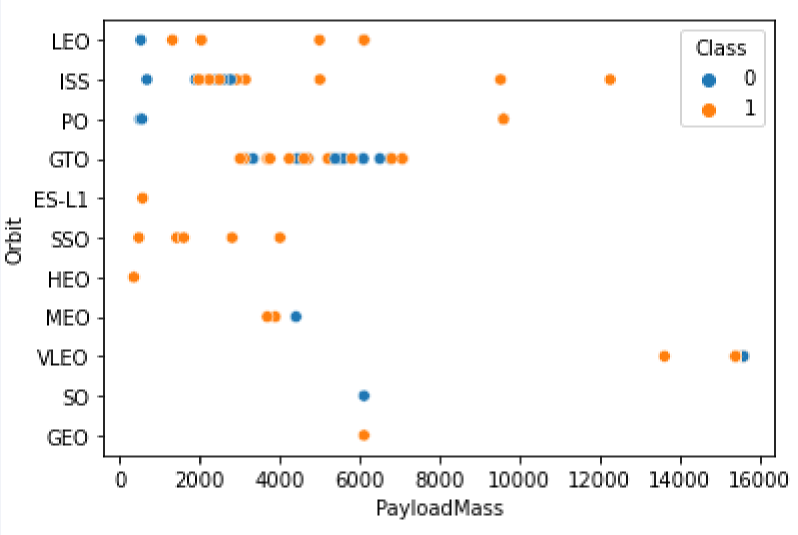

The ES-L1, GEO, HEO and SSO orbits has the highest landing outcome success rate.

# Flight Number vs. Orbit Type



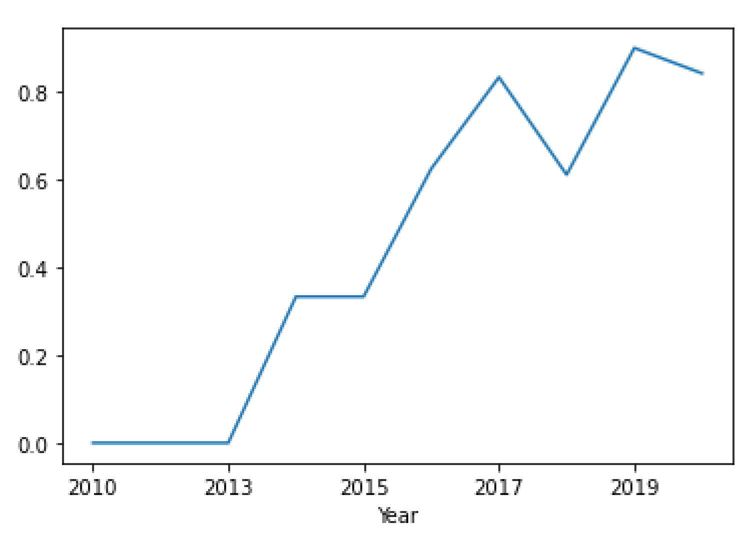

VLEO is the orbit that had the most rockets launched to during the last years

# Payload vs. Orbit Type




ISS has a cluster of launches with a payload of 2000 to 4000, and GTO has one from 3000 to 8000.

# Launch Success Yearly Trend




The results are increasing with time in average, even though some years are a bit lower, like in 2018.

# All Launch Site Names

```
SELECT UNIQUE(launch_site)
FROM SPACEXTBL;
```

| launch_site |
|---|
| CCAFS LC-40 |
| CCAFS SLC-40 |
| KSC LC-39A |
| VAFB SLC-4E |

The UNIQUE() function allows us to avoid results redundancy.

# Launch Site Names Begin with 'CCA'

```
SELECT *
FROM SPACEXTBL
WHERE launch_site LIKE '%CCA%'
LIMIT 5;
```

| 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 | 07:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 2012-10-08 | 00:35:00 | 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 |

The LIKE allows to get results that are like CCA, with strings that can replace the ‘%’ signs.

The LIMIT parameter limits the result of the query to 5 rows.

# Total Payload Mass

```
SELECT SUM(payload_mass__kg_) AS "Total Payload Mass"
FROM SPACEXTBL
WHERE customer LIKE 'NASA (CRS)';
```

| Total Payload Mass |
|---|
| 45596 |

The SUM() function allows us to get the sum of all payload mass of every rows returned by the rest of the query.

# Average Payload Mass by F9 v1.1

```
SELECT AVG(payload_mass__kg_) AS "Average Payload Mass"
FROM SPACEXTBL
WHERE booster_version = 'F9 v1.1';
```

| Average Payload Mass |
|---|
| 2928 |

The AVG() function allows us to get the average of all payload mass of every rows returned by the rest of the query, with a filter to get the 'F9 v1.1' model.

# First Successful Ground Landing Date

```sql
SELECT MIN(date) AS "First successful launch"
FROM SPACEXTBL
WHERE landing__outcome LIKE '%Success (ground pad)%';
```

| First successful launch |
|---|
| 2015-12-22 |

The MIN() function allows us to get the minimum value of the dates that were filtered to get only the ones with a successful outcome.

# Successful Drone Ship Landing with Payload between 4000 and 6000

```
SELECT booster_version
FROM SPACEXTBL
WHERE landing__outcome LIKE '%Success (drone ship)%'
AND payload_mass__kg_ BETWEEN 4000 AND 6000;
```

| booster_version |
|---|
| F9 FT B1022 |
| F9 FT B1026 |
| F9 FT B1021.2 |
| F9 FT B1031.2 |

The BETWEEN parameter limits the results to the rows with a payload mass between  4000 and 6000.

# Total Number of Successful and Failure Mission Outcomes

```
SELECT COUNT(*)
FROM SPACEXTBL
WHERE mission_outcome LIKE '%Success%'
OR mission_outcome LIKE '%Failure%';
```

| 1 |
| --- |
| 101 |

The OR parameter allows to include the two values in the result.

# Boosters Carried Maximum Payload

```
SELECT booster_version
FROM SPACEXTBL
WHERE payload_mass__kg_ = (
    SELECT MAX(payload_mass__kg_) FROM SPACEXTBL);
```

| 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 |

The subquery gets the maximum payload value, and the main one returns the rows with that payload value.

# 2015 Launch Records

```sql
SELECT date, booster_version, launch_site, landing__outcome
FROM SPACEXTBL
WHERE landing__outcome LIKE '%Failure (drone ship)%'
AND YEAR(date) = 2015;
```

| DATE | booster_version | launch_site | landing__outcome |
|---|---|---|---|
| 2015-01-10 | F9 v1.1 B1012 | CCAFS LC-40 | Failure (drone ship) |
| 2015-04-14 | F9 v1.1 B1015 | CCAFS LC-40 | Failure (drone ship) |

The YEAR() functions returns the year contained in the ‘date’ field as an integer, so it is easier to compare to the ‘2015’ value.

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

```
SELECT landing__outcome, COUNT(*) AS "Amount of launches"
FROM SPACEXTBL
WHERE date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY landing__outcome
ORDER BY "Amount of launches" DESC;
```

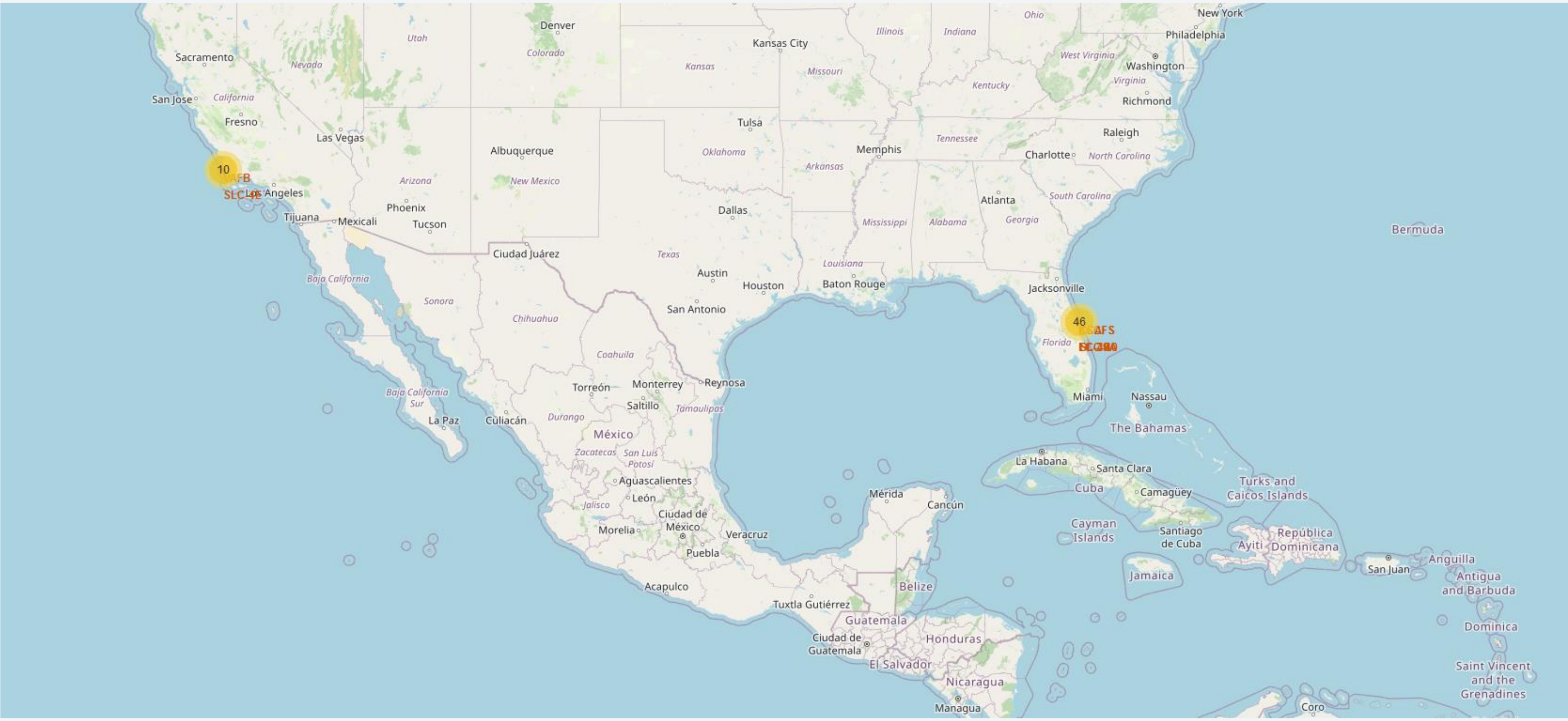
| landing__outcome | Amount of launches |
|---|---|
| No attempt | 10 |
| Failure (drone ship) | 5 |
| Success (drone ship) | 5 |
| Controlled (ocean) | 3 |
| Success (ground pad) | 3 |
| Failure (parachute) | 2 |
| Uncontrolled (ocean) | 2 |
| Precluded (drone ship) | 1 |

The GROUP BY groups the results to get data for all the values of the field, and the COUNT() gets the occurrences (amount of rows) of each one.

Section 3

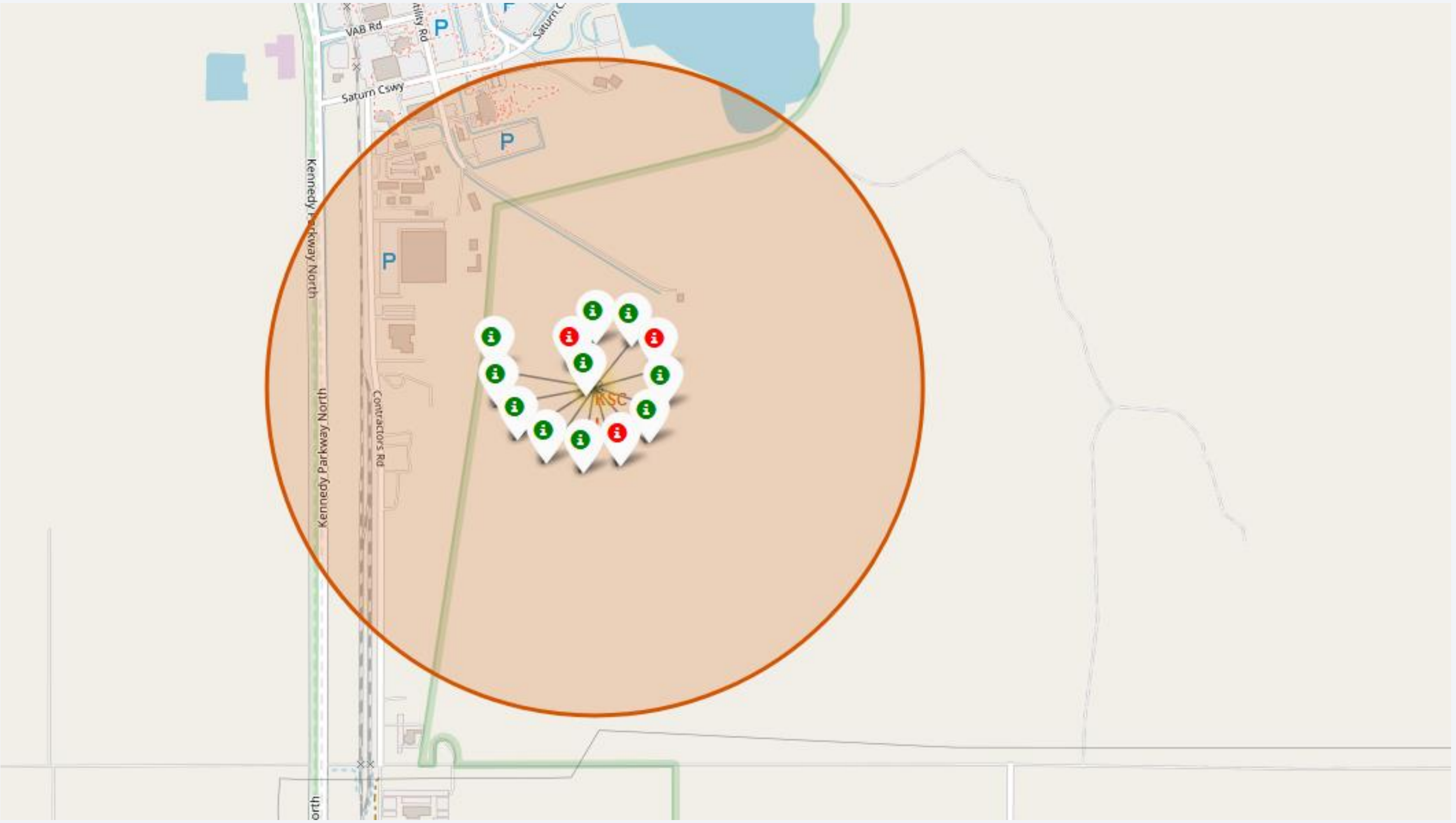
# Launch Sites Proximities Analysis

# All Launch Sites Locations



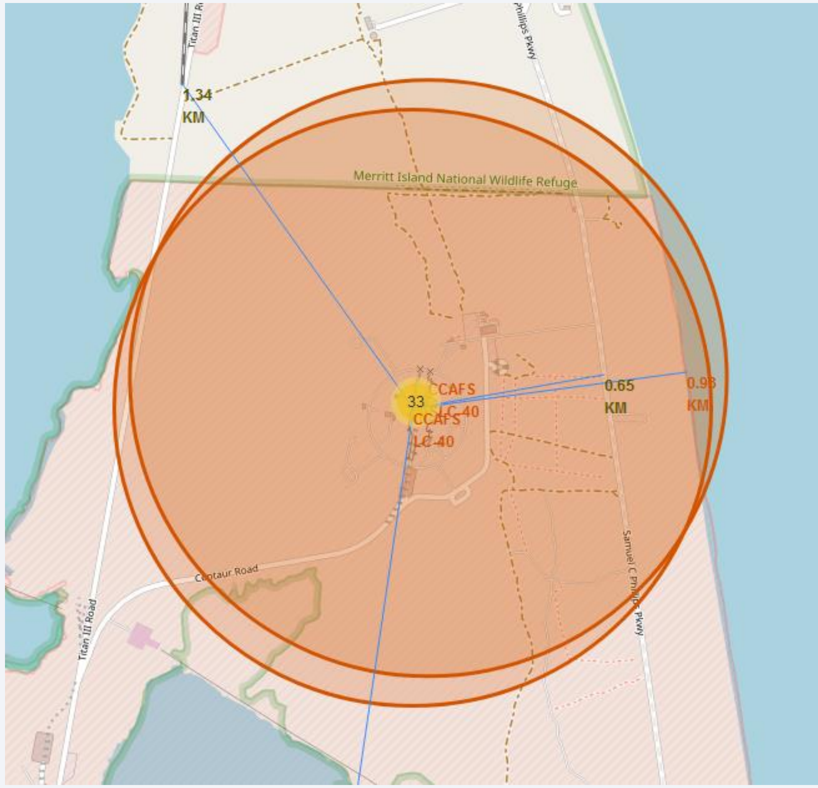

Two clusters of sites are noticeable on the map : one near Los Angeles, and the most import near Orlando

# Launch Sites Outcomes




For each launch point, the success outcomes are marked in green and the failure in red when clicked on.

# Proximities to infrastructures





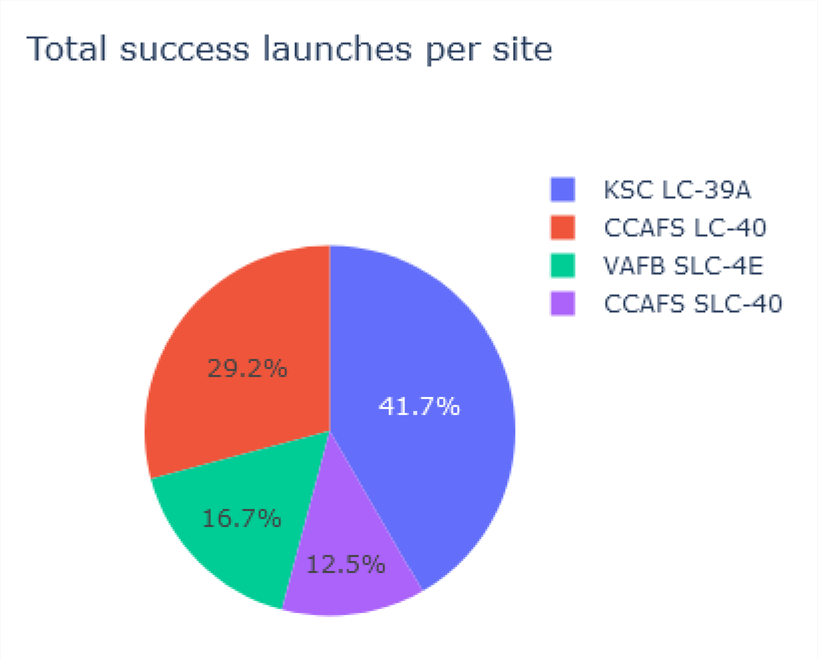
Distances are shown, for example here the site is at a minimum distance of :

- 1.34 km from a railway
- 0.98 km from a coast
- 0.65 km from a highway
- 18.03 km from a city

Section 4

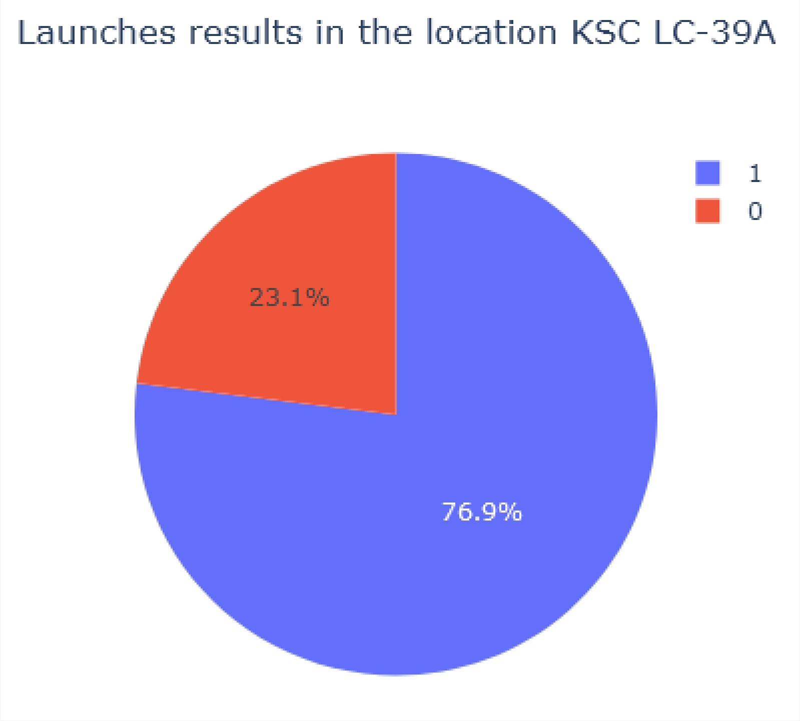
# Build a Dashboard with Plotly Dash

# Launch Success Count for all Sites




The site with the most success launches is KSC LC-39A (41.7% of the successfully landed launches), whereas CCAFS SLC-40 is the one with the smallest amount.

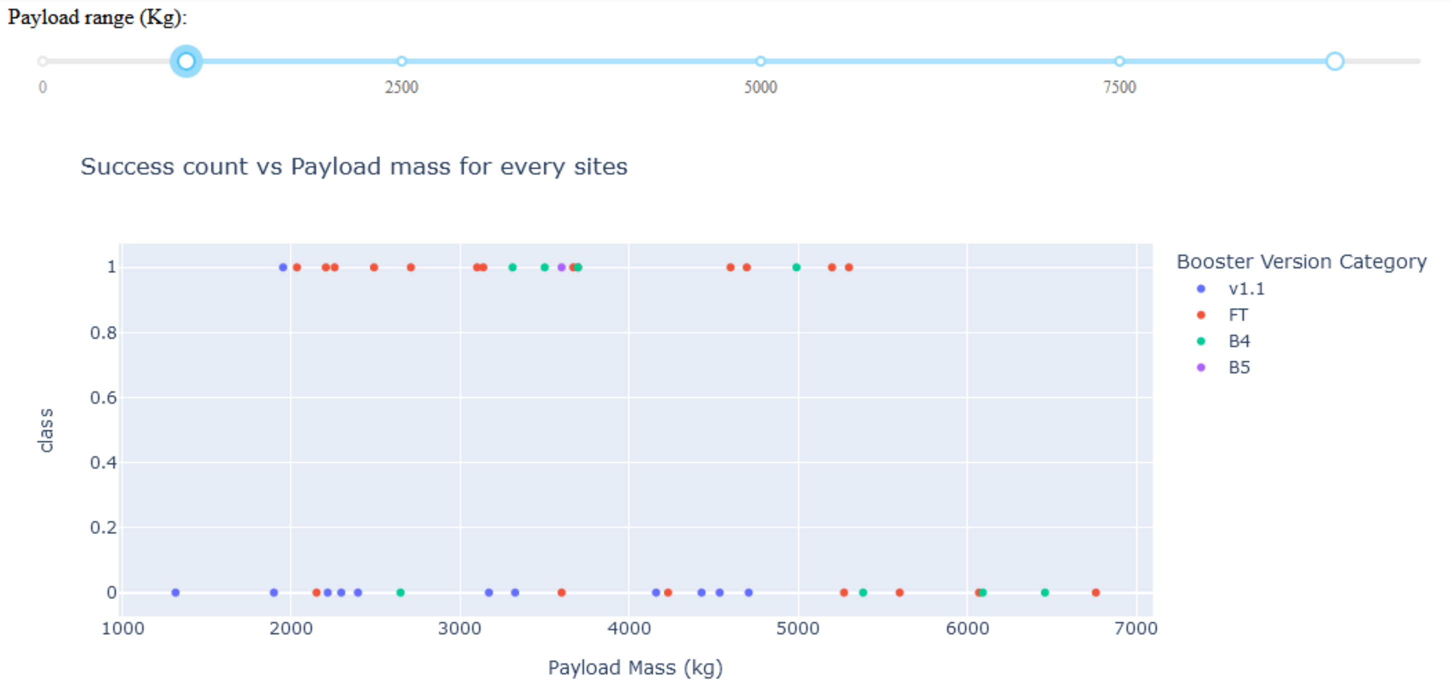
# Highest Launch Success Ratio Site




The site with the most success launches is ratio KSC LC-39A (76.9%).

It was the most represented on the last slide too, so this site seems to have a noticeable correlation with the outcome result.

# Payload vs. Launch Outcome for all Sites




For the Payload range from 1000 to 8500, the results range goes from 1300 to less than 7000 kg of payload.

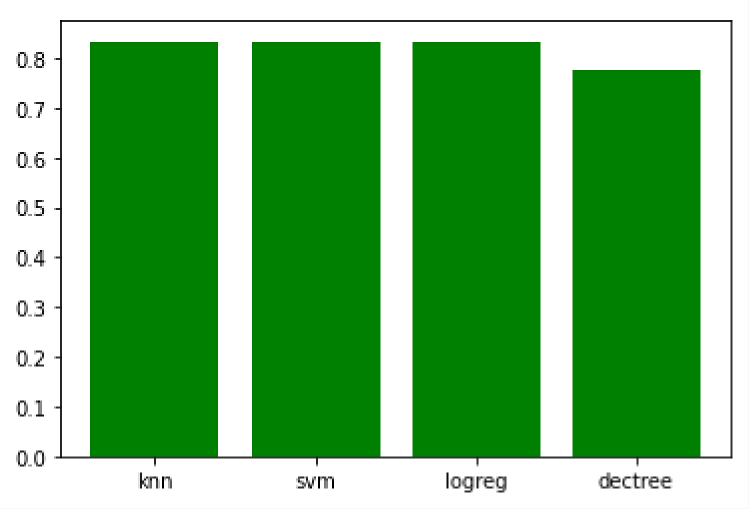
The V1.1 Booster tends to fail more.

Section 5

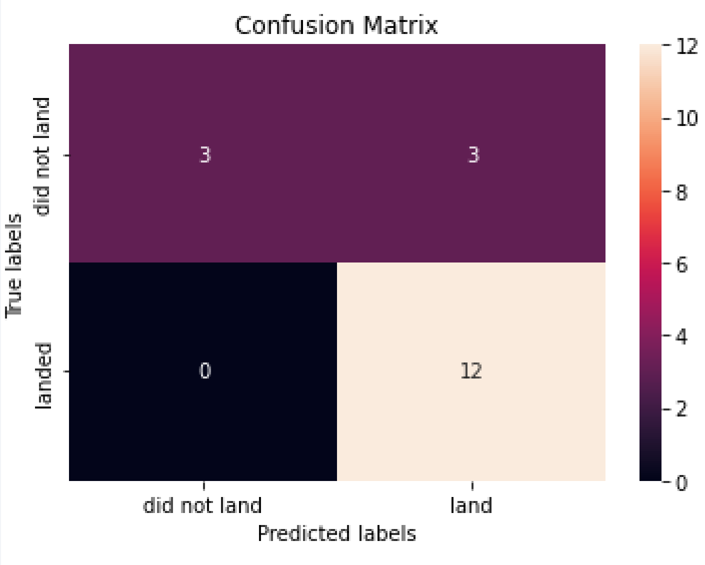
# Predictive Analysis (Classification)

# Classification Accuracy

The KNN, SVM, and Logistic Regression has an equivalent accuracy of 83.34% on the testing set with the optimized parameters.



# Confusion Matrix




All the successfully landed launches have been well classified, whereas 3 of the 6 failures have been classified as success.

# Conclusions

- The lighter the payloads are, the higher the success rate is.
- The results are getting better with time, the rate of success may still increase in the next coming years.
- KSC LC-39A launch site has the best overall success rate (76.9%).
- ES-L1, GEO, HEO and especially SSO are the orbits with the smallest failure rate.
- KNN, SVM, and Logistic Regression are the best models from the ones tested.

# Appendix

- A testing datasets have been created and exported to CSV format
- All the code files are available on my GitHub for more information

# Thank you!