IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mujeeb Yusuf

22-June-2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
  - Data Collection through API and Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Map Visualization with Folium
  - Interactive dashboard with Plotly Dash
  - Predictive Analysis with Machine Learning
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive Analytics
  - Predictive Analysis

Introduction

- Project background and context

  SpaceX has a competitive advantage in the rocket launch market due to its ability to reuse the first stage of its Falcon 9 rockets, which significantly reduces the launch cost compared to other providers. Predicting the landing success of the first stage can help estimate the launch cost and enable potential competitors to bid more effectively against SpaceX.

- Problem statement

  How can we predict the landing success of the first stage of a SpaceX Falcon 9 rocket based on the launch parameters and environmental conditions?

Section 1

Methodology

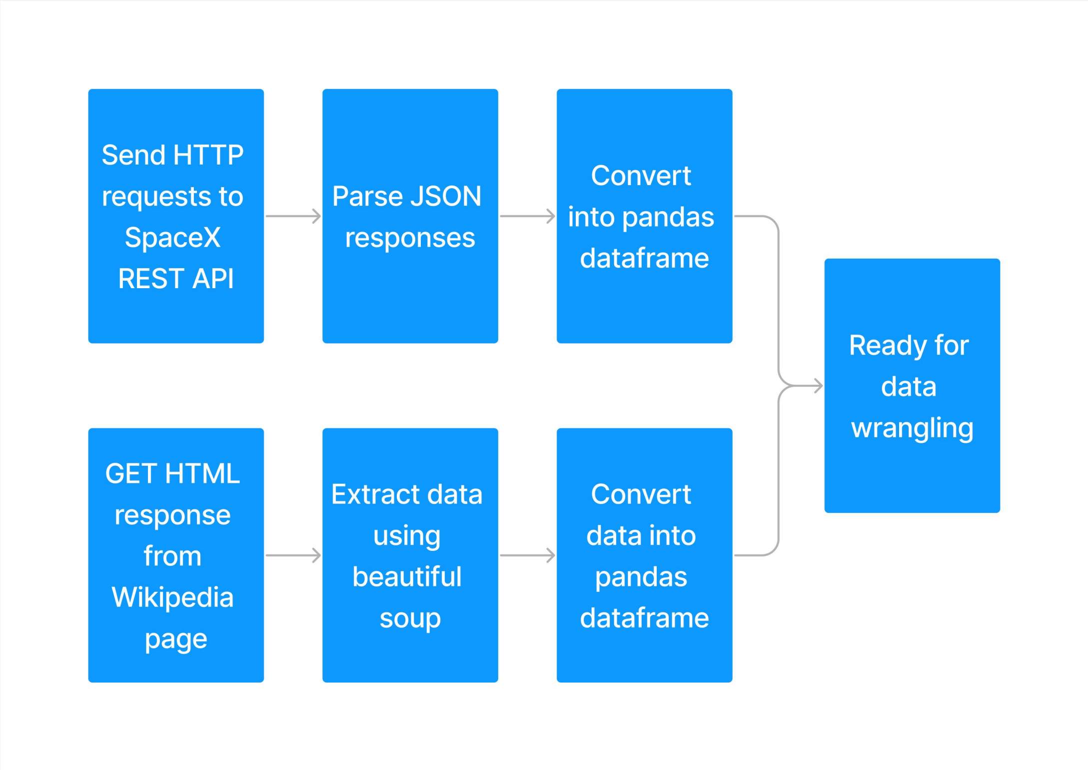
Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX REST API and Web Scraping from Wikipedia.
- Perform data wrangling
  - Transforming categorical variables into numerical vectors and removing missing values and unnecessary columns for Machine Learning.
- 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 have constructed and compared four classification models: Logistic Regression, K-Nearest Neighbors, Support Vector Machine, and Decision Tree, to find the optimal classifier for our data.

Data Collection

- We obtained the data sets from two sources:
  - The SpaceX REST API and the Wikipedia page of Falcon 9 launch records.
  - The requests library in Python was used to send HTTP requests to the API, parse the JSON responses, and convert it into a pandas dataframe.
  - BeautifulSoup library was used to scrape the HTML table from the Wikipedia page and convert it into a pandas dataframe.



Data Collection – SpaceX API

- Get response from SpaceX REST API

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

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

- Convert .json file to a pandas dataframe

```
# Use json_normalize meethod to convert the json result into a dataframe
response = requests.get(static_json_url)

data_json = response.json()

data = pd.json_normalize(data_json)
```

- Perform data wrangling

```
# Lets take a subset of our dataframe keeping only the features we want and the flight number, and date_utc.
data = data[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]

# We will remove rows with multiple cores because those are falcon rockets with 2 extra rocket boosters and rows that have m
data = data[data['cores'].map(len)==1]
data = data[data['payloads'].map(len)==1]

# Since payloads and cores are lists of size 1 we will also extract the single value in the list and replace the feature.
data['cores'] = data['cores'].map(lambda x : x[0])
data['payloads'] = data['payloads'].map(lambda x : x[0])

# We also want to convert the date_utc to a datetime datatype and then extracting the date leaving the time
data['date'] = pd.to_datetime(data['date_utc']).dt.date

# Using the date we will restrict the dates of the launches
data = data[data['date'] <= datetime.date(2020, 11, 13)]
```

from:

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

Data Collection - Scraping

- Request the Falcon9 Launch Wiki page from its URL

```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url).text
```

- Create a BeautifulSoup object from the HTML response

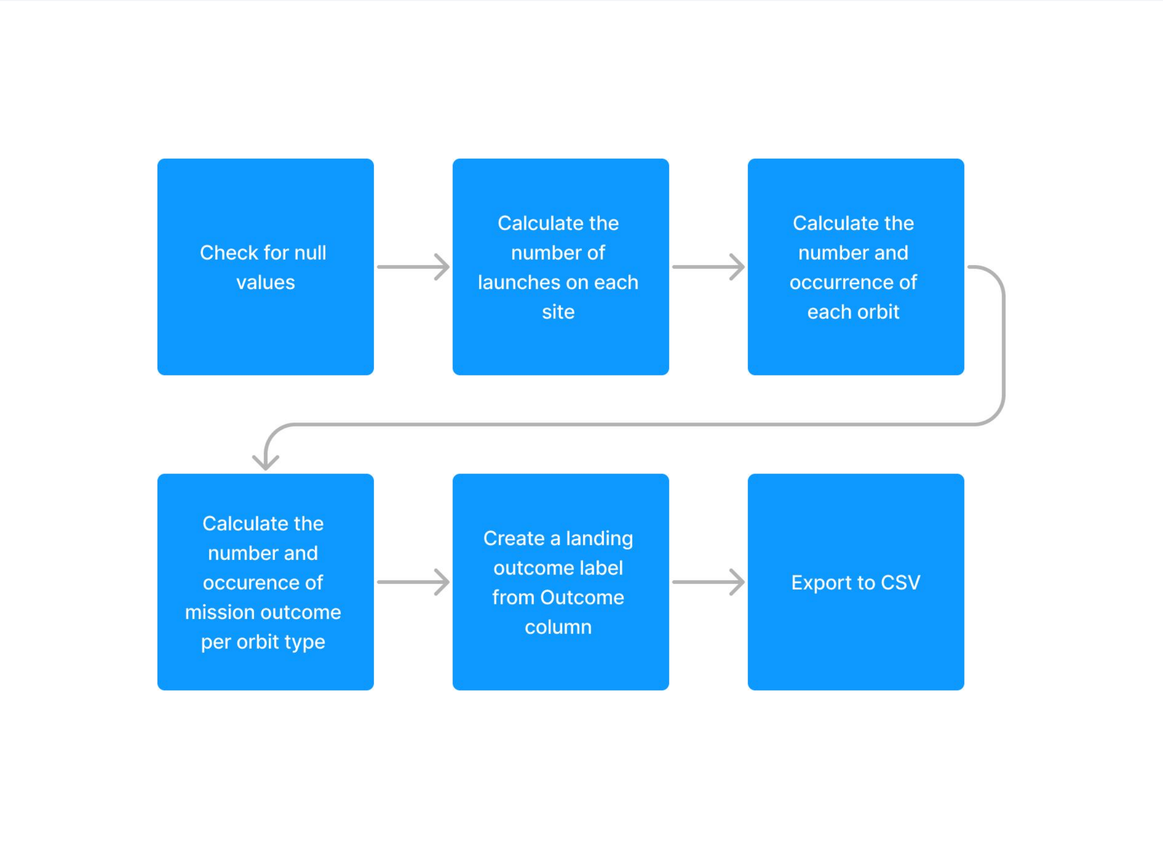
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response, 'html.parser')
```

- Extract all column/variable names from the HTML table header

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
        #if it is number save cells in a dictonary
        if flag:
            extracted_row += 1
            # Flight Number value
```

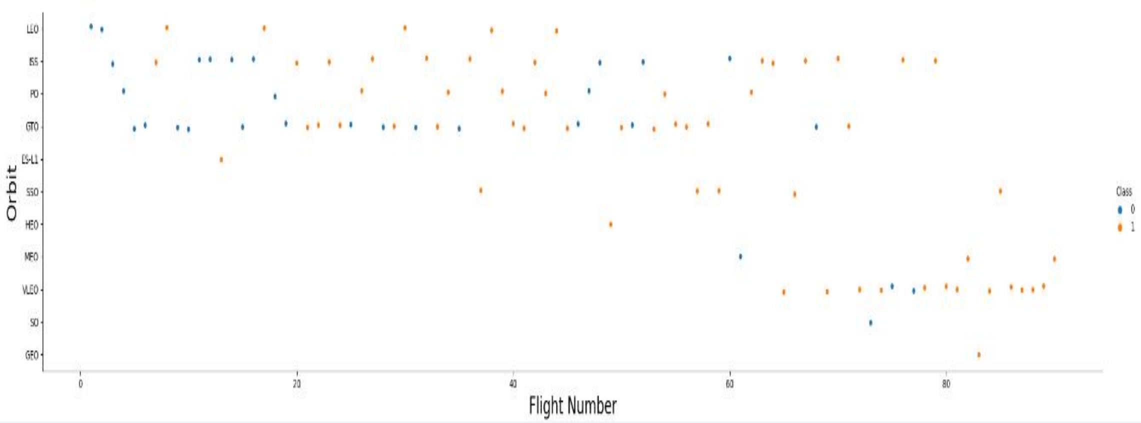
from: https://github.com/Mujeeby/Applied-Data-Science-Capstone-SpaceX-Project/blob/main/jupyter-labs-webscraping.ipynb

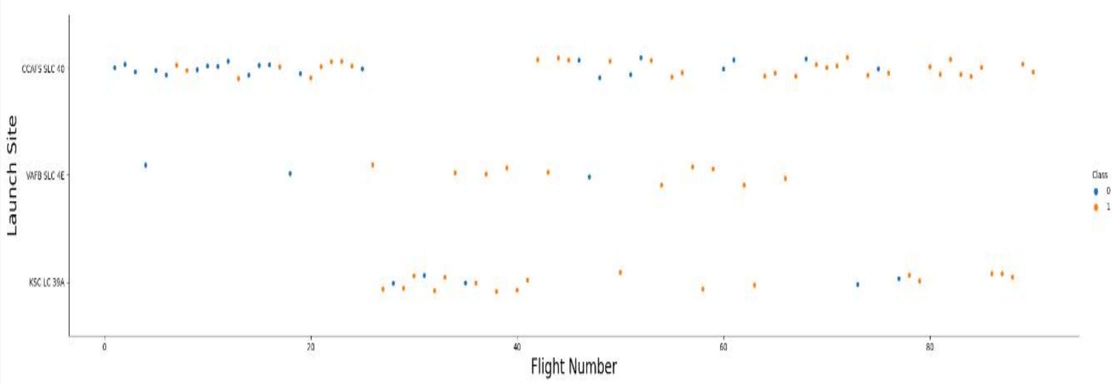
Data Wrangling

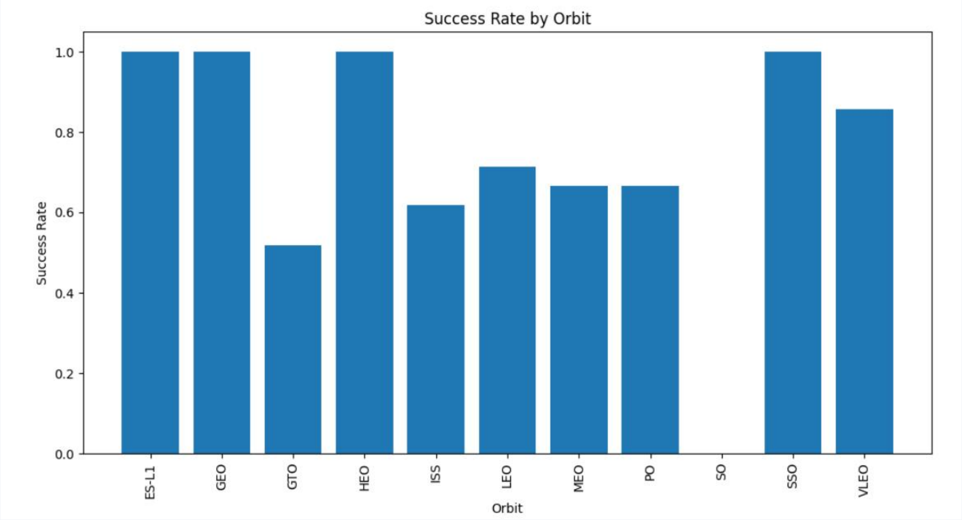


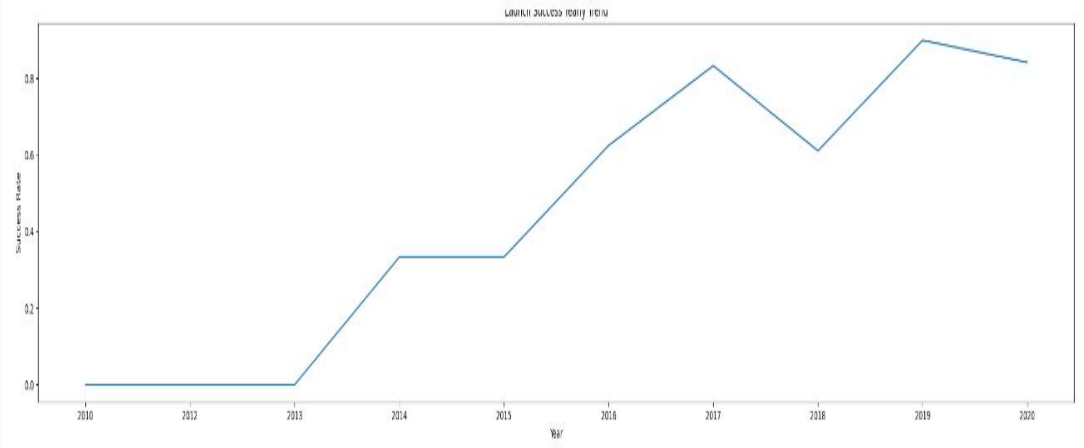
from:
https://github.com/Mujeeby/Applied-Data-Science-Capstone-SpaceX-Project/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization









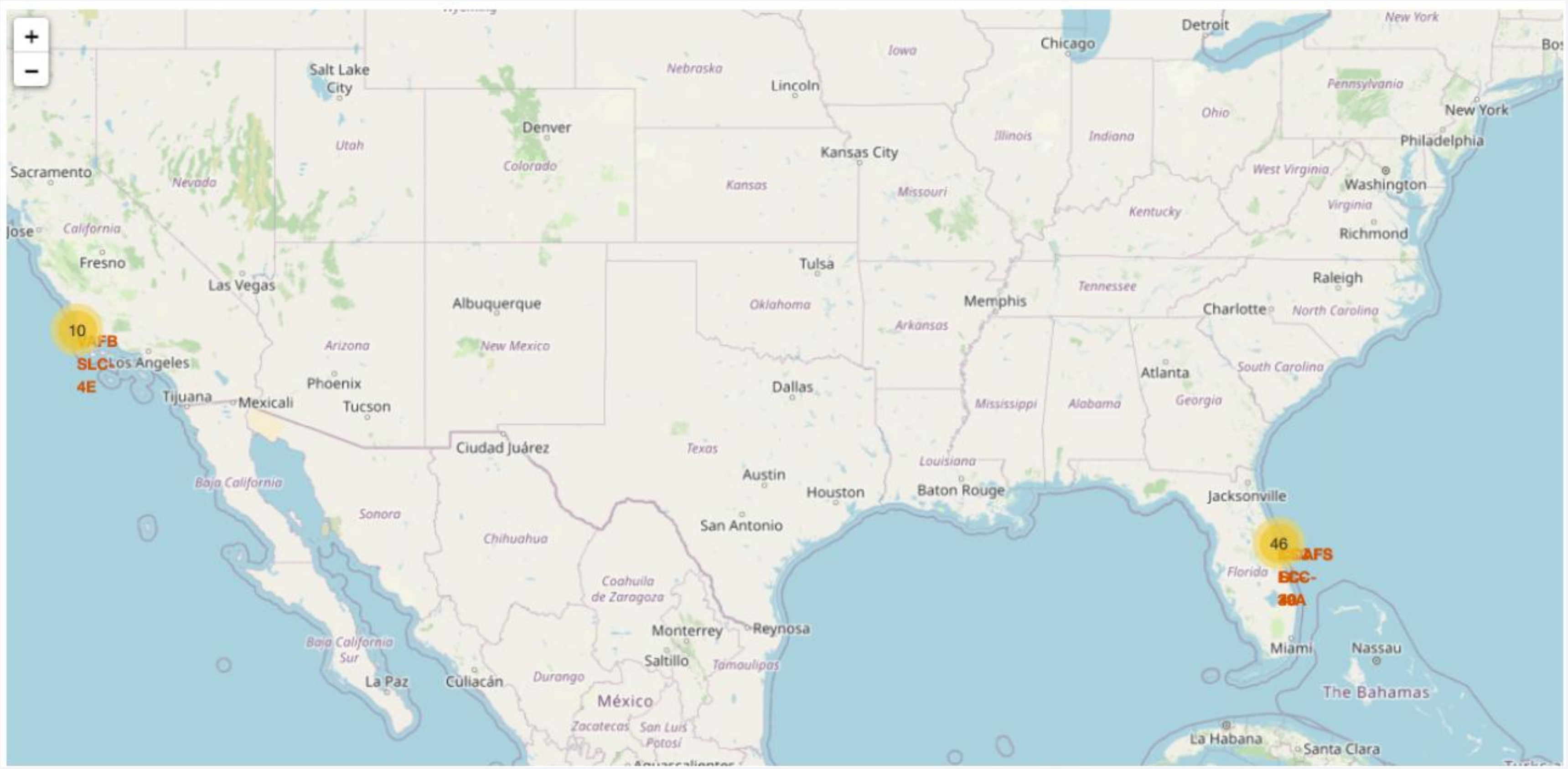
https://github.com/Mujeeby/Applied-Data-Science-Capstone-SpaceX-Project/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- The following SQL queries were performed:
  - 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 succesful 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. Use a subquery
  - 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.
  - Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

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

Build an Interactive Map with Folium



- Map markers have been added to the map with aim to finding an optimal location for building a launch site

https://github.com/Mujeeby/Applied-Data-Science-Capstone-SpaceX-Project/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- We built an interactive dashboard with Plotly dash that enables the user to explore and manipulate the data according to their requirements.
- We created pie charts to visualize the proportion of launches by different sites.
- We then plotted scatter plots to examine the correlation between Outcome and Payload Mass (Kg) for the various booster versions.

https://github.com/Mujeeby/Applied-Data-Science-Capstone-SpaceX-Project/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Model Development

- Import the dataset using NumPy and Pandas
- Preprocess the data and then partition into training and test sets
- Select the appropriate ML method
- Pass the hyperparameters and estimators to GridSearchCV and train it on the data.

Model Evaluation

- Measure the accuracy for each model
- Retrieve the optimal hyperparameters for each algorithm.
- Visualize the confusion matrix.

Model Optimization

- Apply Feature Engineering and Algorithm Tuning

Identify the Optimal Model

- The model with the highest accuracy score will be the most suitable model.

https://github.com/Mujeeby/Applied-Data-Science-Capstone-SpaceX-Project/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

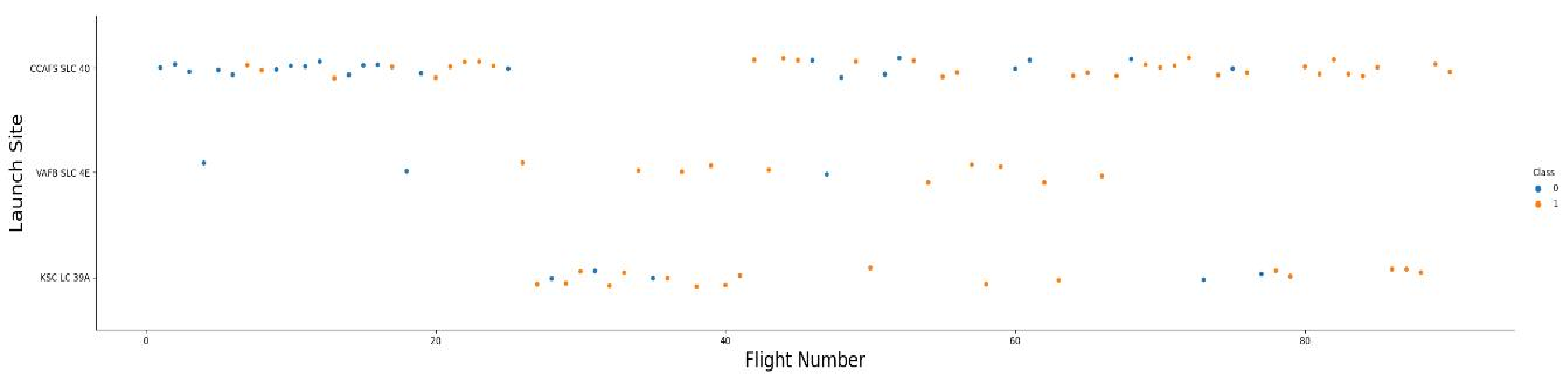
Results

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

Section 2

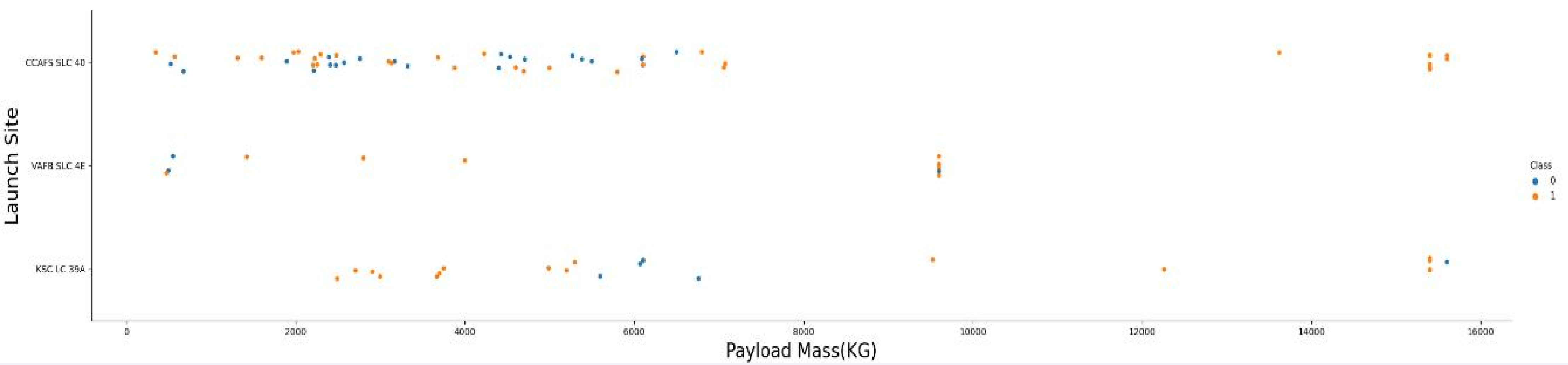
Insights drawn from EDA

Flight Number vs. Launch Site



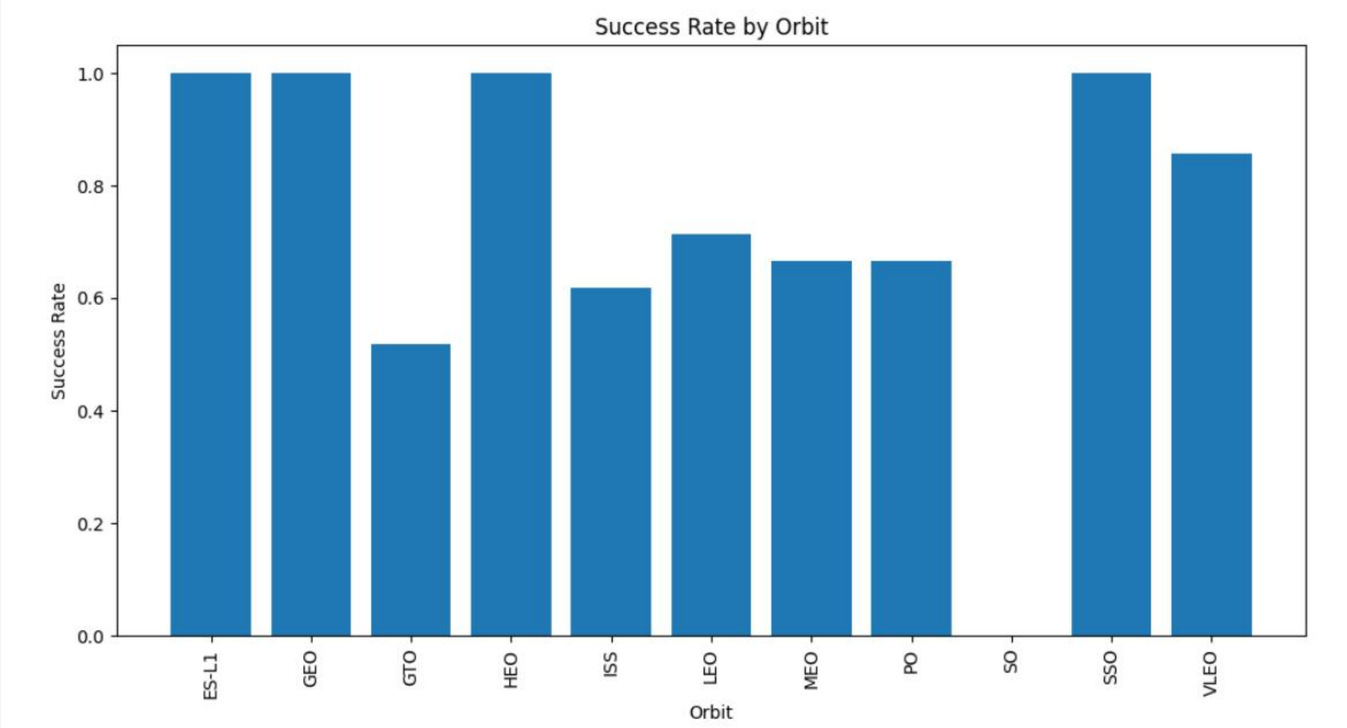
- This scatter plot indicates that the higher the number of flights from the launch site, the more likely the success rate is. However, site CCAFS SLC40 deviates from this trend.

Payload vs. Launch Site



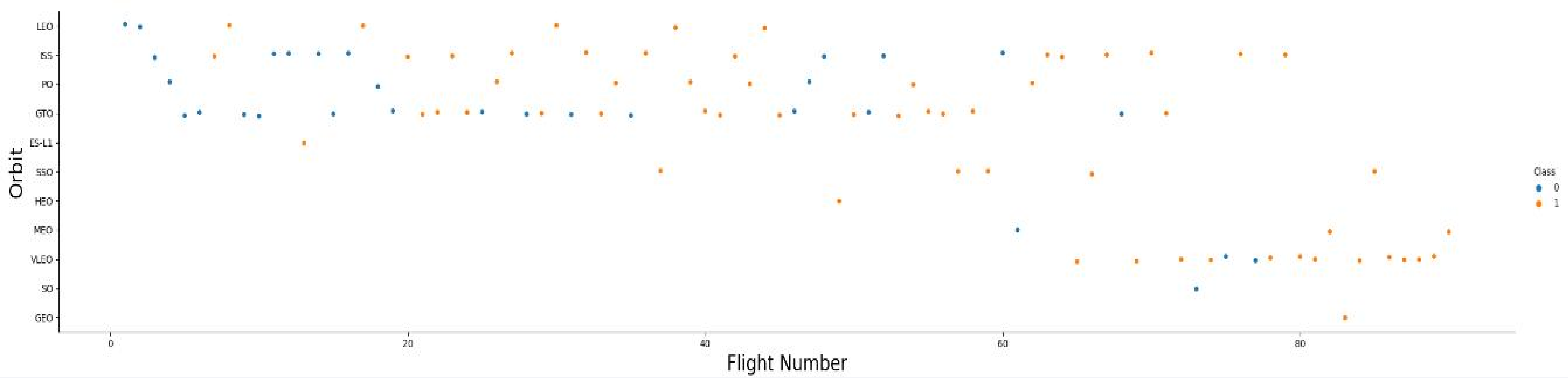
- This scatter plot reveals that the success rate increases significantly when the payload mass is above 7000kg. However, there is no evident relationship between the launch site and the payload mass for the success rate.

Success Rate vs. Orbit Type



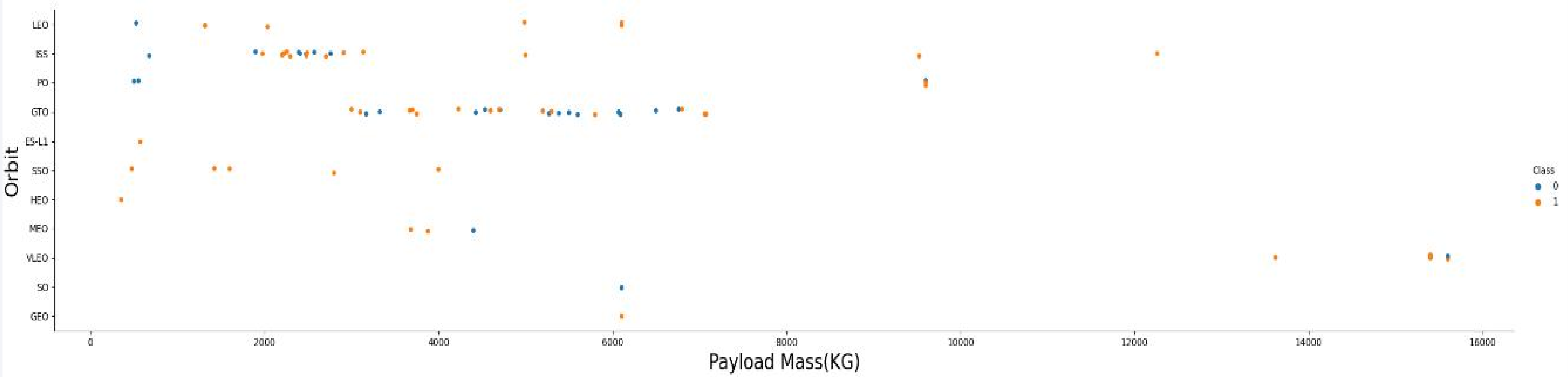
- This figure illustrates the potential impact of the orbits on the landing outcomes as some orbits have 100% success rate such as SSO, HEO, GEO and ES-L1 while SO orbit yielded 0% rate of success.
- However, further analysis reveals that some of these orbits have only one observation such as GEO, SO, HEO and ES-L1 which means this data requires more samples to see patterns or trends before we make any inference.

Flight Number vs. Orbit Type



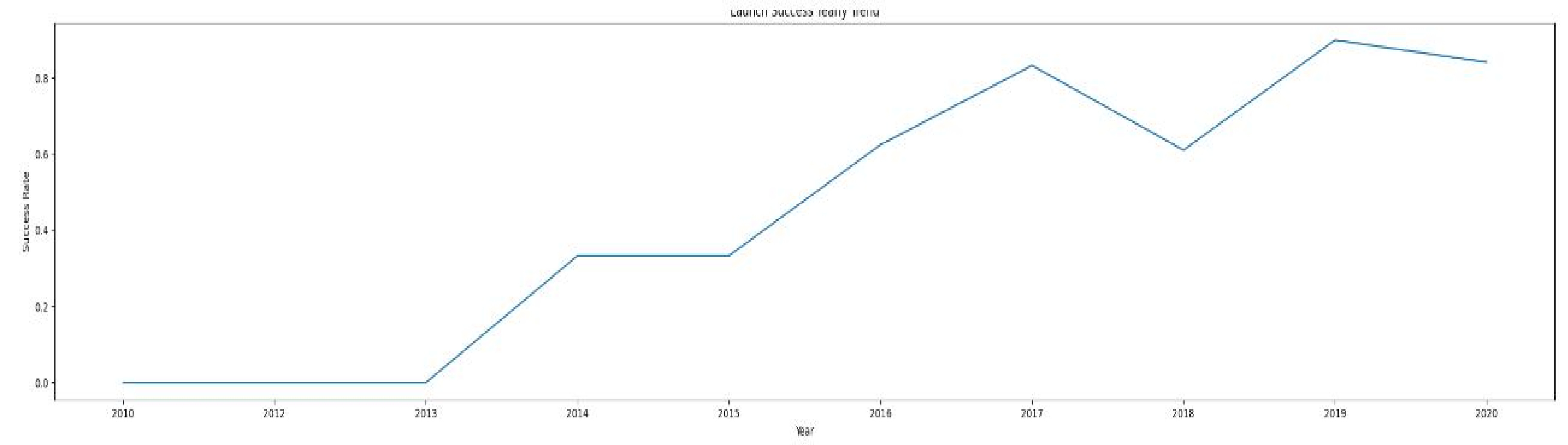
- This scatter plot indicates that overall, the higher the flight number on each orbit, the higher the success rate (particularly LEO orbit) except for GTO orbit which shows no correlation between both variables. Orbit that only has one observation should also be omitted from the above statement as it requires more data.

Payload vs. Orbit Type



- Higher payload mass has positive effect on LEO, ISS and P0 orbit. However, it has negative effect on MEO and VLEO orbit. GTO orbit seems to show no relationship between the variables. Meanwhile, again, SO, GEO and HEO orbit require more data to see any patterns or trends.

Launch Success Yearly Trend



- This figure clearly shows an increasing trend from the year 2013 to 2020. If this trend persists for the following years, the success rate will gradually increase until reaching 100% success rate.

All Launch Site Names

- %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL

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

Launch Site Names Begin with 'CCA'

- %sql 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_Outc
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (paracl
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (paracl
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No atte
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No atte
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No atte

Total Payload Mass

- %%sql

```
    SELECT Customer, SUM("PAYLOAD_MASS__KG_") AS
"TOTAL_PAYLOAD_MASS" FROM SPACEXTBL
    WHERE Customer = 'NASA (CRS)'
```

Customer	TOTAL_PAYLOAD_MASS
NASA (CRS)	45596.0

Average Payload Mass by F9 v1.1

- %%sql

```
SELECT Booster_Version, AVG("PAYLOAD_MASS__KG_") AS
"AVERAGE_PAYLOAD_MASS" FROM SPACEXTBL
WHERE "Booster_Version" = 'F9 v1.1'
```

Booster_Version	AVERAGE_PAYLOAD_MASS
F9 v1.1	2928.4

First Successful Ground Landing Date

- %%sql

```
SELECT MIN(Date), "Landing_Outcome" FROM SPACEXTBL
WHERE "Landing_Outcome" = 'Success (ground pad)'
```

MIN(Date)	Landing_Outcome
01/08/2018	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

- %%sql

  SELECT "Booster_Version", "PAYLOAD_MASS__KG_", "Landing_Outcome" FROM SPACEXTBL

  WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000

Booster_Version	PAYLOAD_MASS__KG_	Landing_Outcome
F9 FT B1022	4696.0	Success (drone ship)
F9 FT B1026	4600.0	Success (drone ship)
F9 FT B1021.2	5300.0	Success (drone ship)
F9 FT B1031.2	5200.0	Success (drone ship)

Total Number of Successful and Failure Mission Outcomes

- %%sql

```
SELECT "Mission_Outcome", COUNT(*) AS Total FROM SPACEXTBL
WHERE "Mission_Outcome" LIKE '%Success%' OR "Mission_Outcome" LIKE '%Fail%'
GROUP BY "Mission_Outcome"
```

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

Boosters Carried Maximum Payload

- %%sql

  SELECT "Booster_Version", "PAYLOAD_MASS__KG_" FROM SPACEXTBL

  WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTBL)

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600.0
F9 B5 B1049.4	15600.0
F9 B5 B1051.3	15600.0
F9 B5 B1056.4	15600.0
F9 B5 B1048.5	15600.0
F9 B5 B1051.4	15600.0
F9 B5 B1049.5	15600.0
F9 B5 B1060.2	15600.0
F9 B5 B1058.3	15600.0
F9 B5 B1051.6	15600.0
F9 B5 B1060.3	15600.0
F9 B5 B1049.7	15600.0

2015 Launch Records

- %%sql

SELECT substr(Date, 4, 2) AS "Month_Names", Date, "Landing_Outcome", "Booster_Version", "Launch_Site"

FROM SPACEXTBL

WHERE "Landing_Outcome" LIKE '%Failure%' AND substr(Date,7,4)='2015'

Month_Names	Date	Landing_Outcome	Booster_Version	Launch_Site
10	01/10/2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	14/04/2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- %%sql

  SELECT "Landing_Outcome", COUNT("Landing_Outcome") AS "Count" FROM SPACEXTBL

  WHERE "Landing_Outcome" LIKE '%Success%' AND Date BETWEEN '2010-06-04' AND '2017-03-20'

  GROUP BY "Landing_Outcome"

  ORDER BY "Count" DESC;

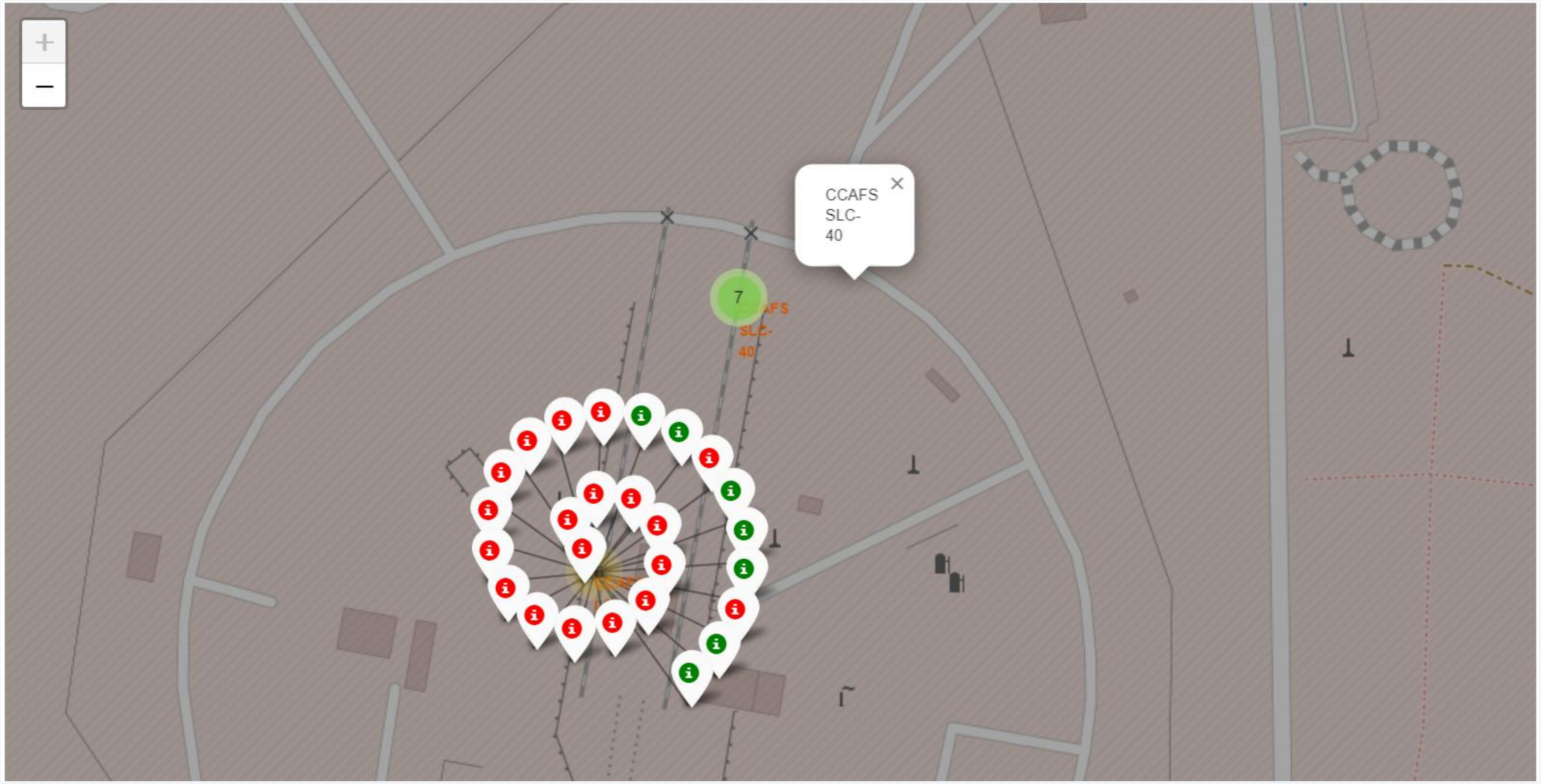
Section 3

Launch Sites Proximities Analysis

Location of all Launch Sites marked on map

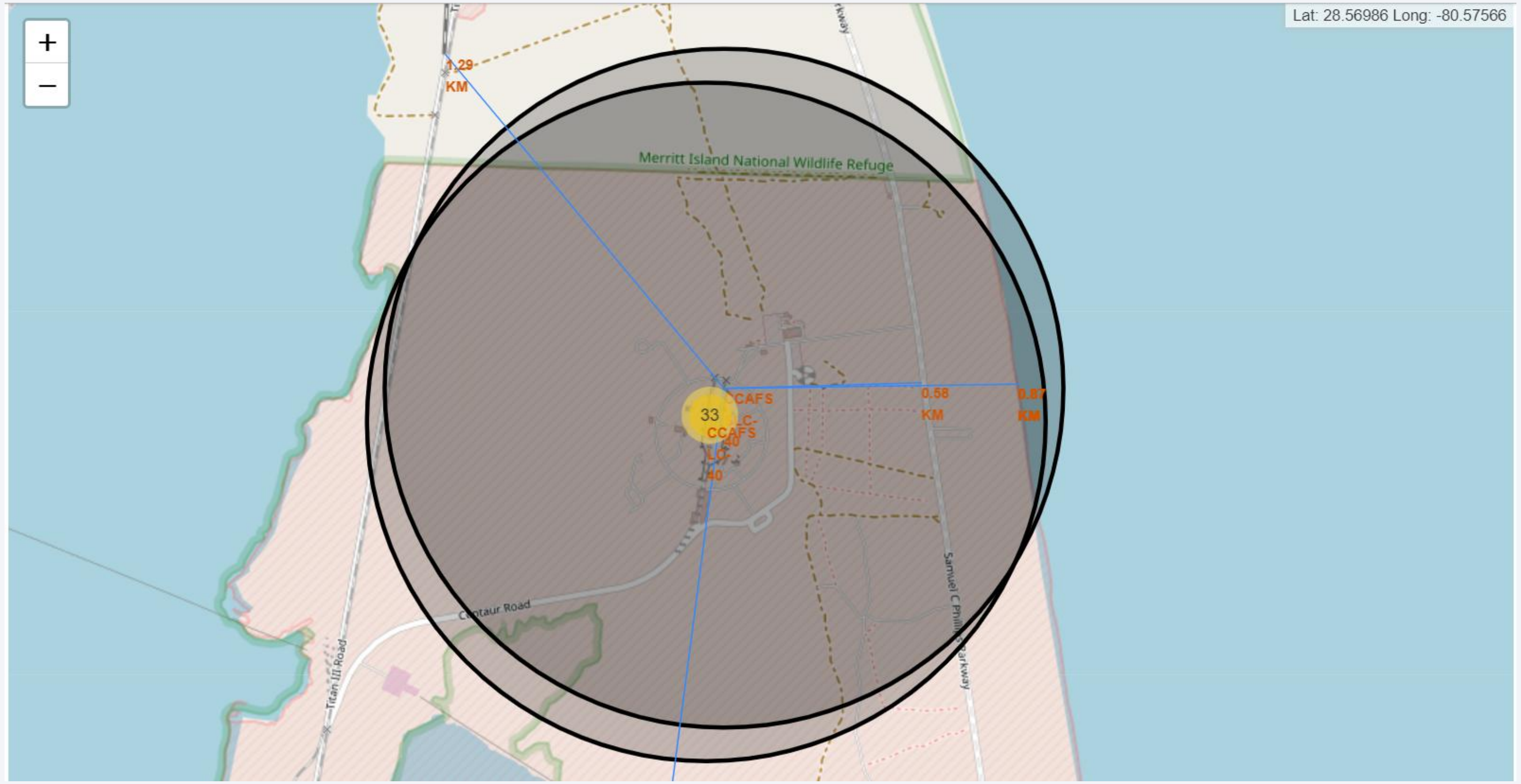
- The United States is the only country that hosts the launch sites of SpaceX.

Successful/failed launches marked on the Map



- The green marker shows successful launches and the red markers shows failed launches.

Launch Sites distance to Landmarks

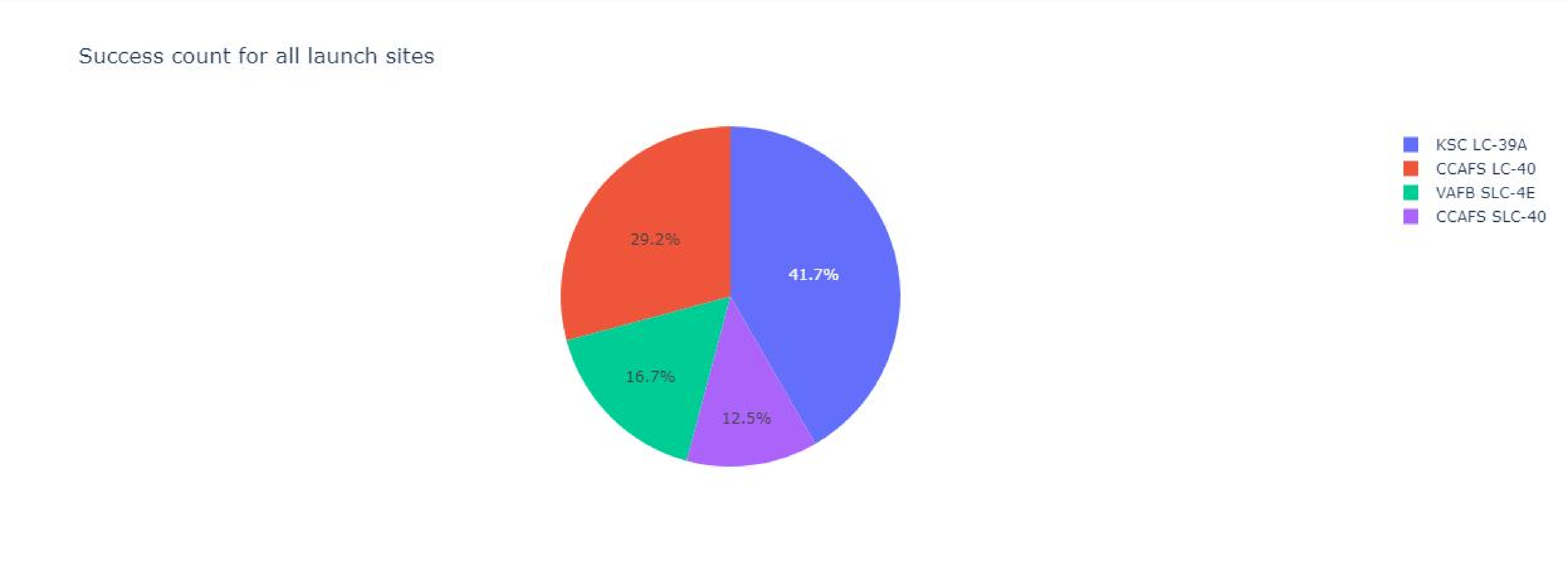


- We use the blue line to indicate how far the launch sites are to the nearest landmarks.

Section 4

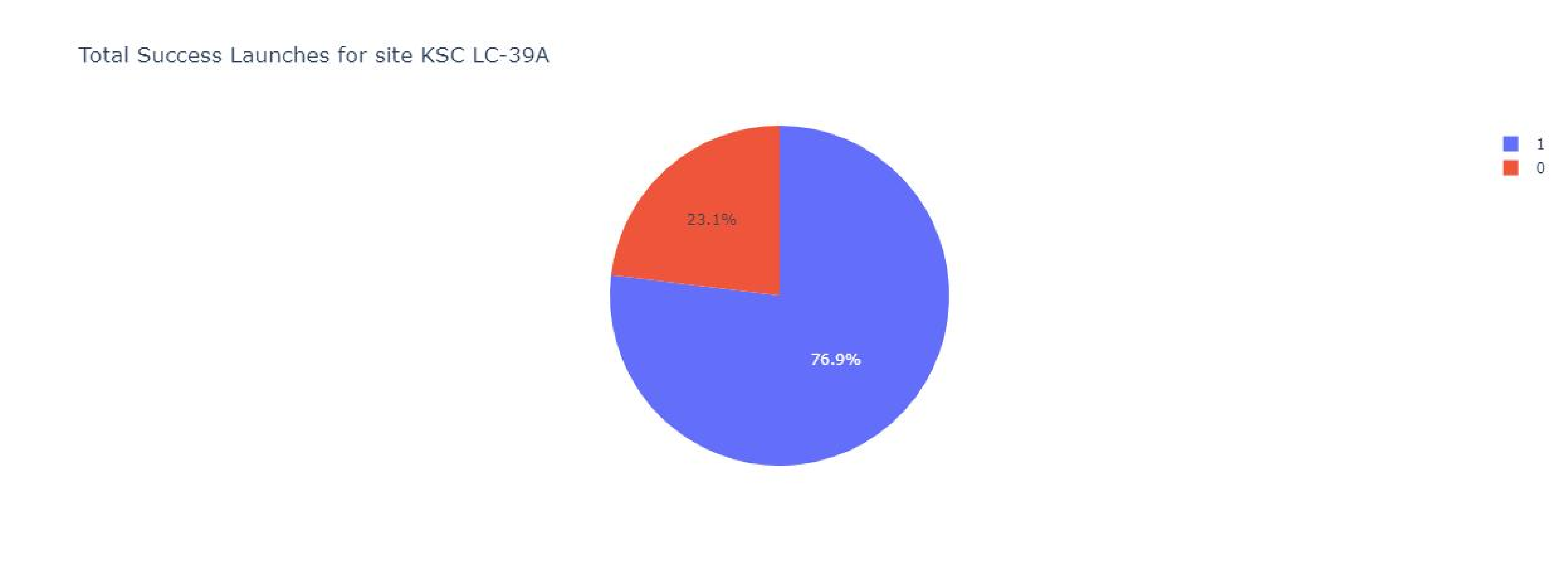
Build a Dashboard with Plotly Dash

Total success launches by all sites



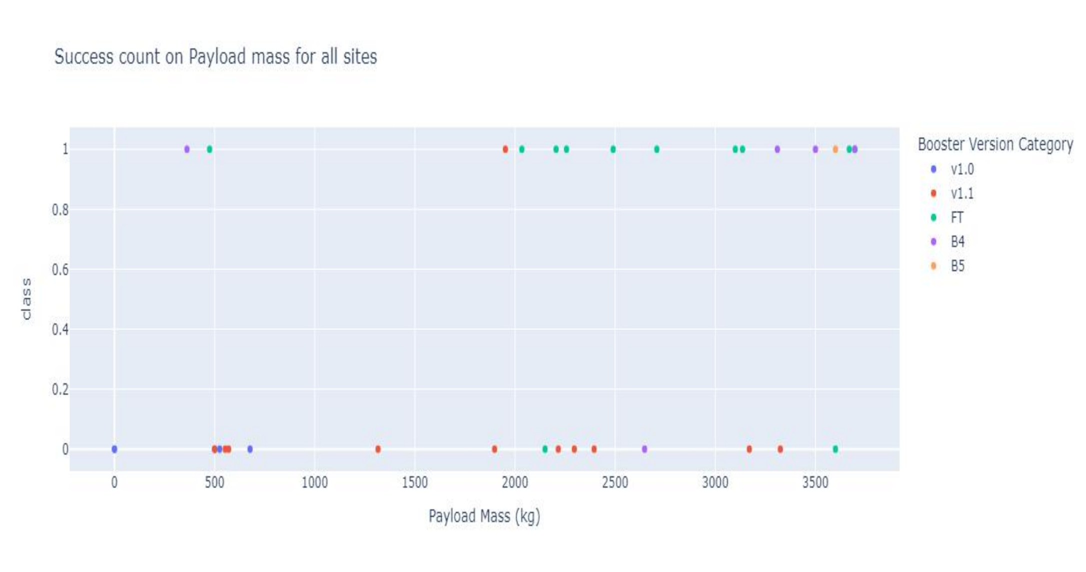
- We can see that KSC LC-39A had the most successful launches from all the sites

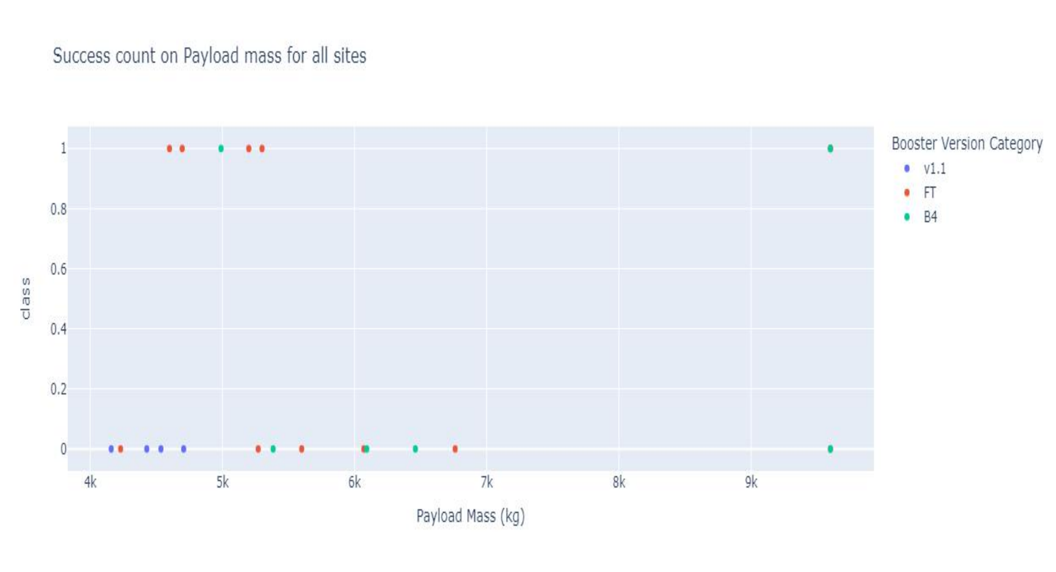
Success rate by site



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

Payload vs Launch Outcome





- We can all see that the success rate drops as the payload weight increases.

Section 5

Predictive Analysis (Classification)

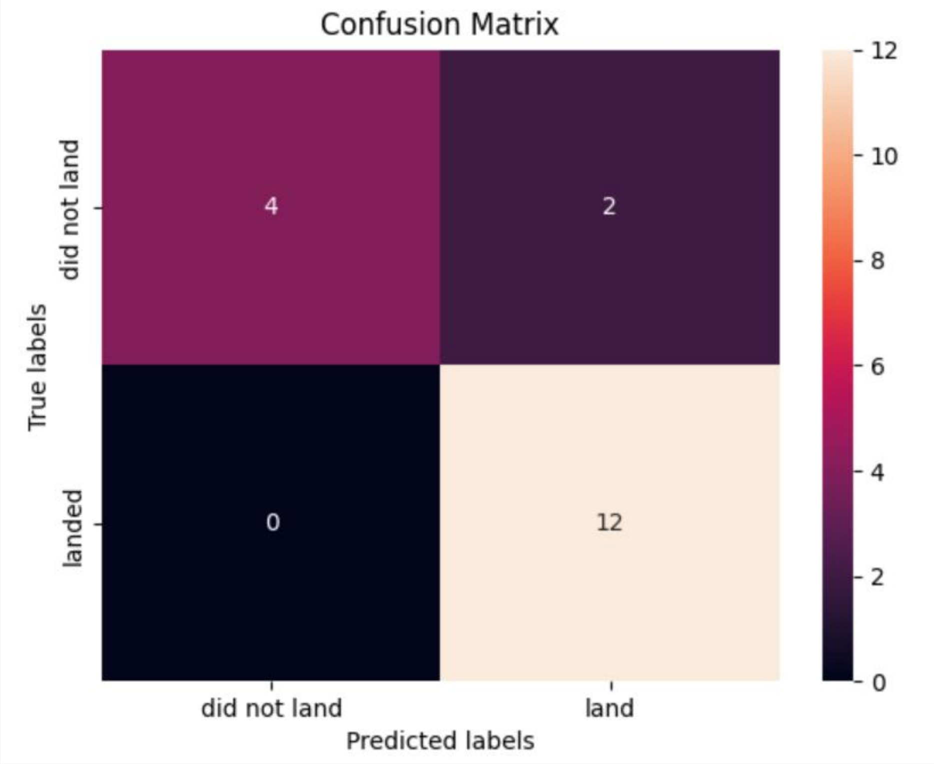
Classification Accuracy

```
predictors = [knn_cv, svm_cv, logreg_cv, tree_cv]
best_predictor = None
best_result = 0

for predictor in predictors:
    score = predictor.score(X_test, Y_test)

    if score > best_result:
        best_result = score
        best_predictor = predictor

best_predictor
```

```
GridSearchCV(cv=10, estimator=DecisionTreeClassifier(),
             param_grid={'criterion': ['gini', 'entropy'],
                         'max_depth': [2, 4, 6, 8, 10, 12, 14, 16, 18],
                         'max_features': ['auto', 'sqrt'],
                         'min_samples_leaf': [1, 2, 4],
                         'min_samples_split': [2, 5, 10],
                         'splitter': ['best', 'random']})
```

- The code shows that the Decision Tree Classifier has the best performance among the classifiers, based on the accuracy metric.

Confusion Matrix



- The Decision Tree Classifier Confusion Matrix

Conclusions

- The Decision Tree Classifier is the best in terms of prediction accuracy for this dataset.
- The low weighted payloads (which define as 4000kg and below) performed better than the heavy weighted payloads.
- The success rate for SpaceX launches has been rising steadily since 2013, and it is expected to reach perfection in the near future.
- KSC LC-39A is the most reliable launch site, with a success rate of 76.9%.
- SSO orbit is the most successful orbit type, with a 100% success rate and more than one occurrence.

Thank you!