IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Katelyn Downey
8/2/22

Executive Summary

Methodology

Data collection through API
Data collection with Web Scrapping
Data Wrangling
EDA with SQL
EDA with Data Visualization
Interactive Visual Analytics with Folium
Machine Learning Prediction

Result Summary

EDA results
Screenshots of Interactive analytics
Predictive analytic results

Introduction

Project Background and Context

We can use space X API to look at the success rate of each launch
Using this information we could build a plan for an alternate company that wants to compete against space X

Problems Were Trying To Find Answers To

What factors determine if the rocket will land successfully?

The interaction amongst various features that determine the success rate of a successful landing.

What operating conditions needs to be in place to ensure a successful landing program

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
    - Describe how data was collected
- Perform data wrangling
    - Describe how data was processed
- 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

Collect and web scrape data from Wikipedia

Read data as JSON

Look at available data then clean and parse data with EDA

Make Graphs that clearly explain what is shown in the data

Data Collection
SpaceX API

The get request function is used to retrieve the SpaceX API

We transform the result into JSON to make it human readable

Basic cleaning is then implemented

Finally we wrangle the data into what is needed for our use

https://github.com/katdown-code/IBM-Course-Work/blob/main/mod10/notebooks/part1.ipynb

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

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

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

```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFraame(rows)
df_rows = df_rows.replace(np.nana,PayloadMass)

data_falcon9['PayloadMass'[0]= df_rows.values
data_falcon9
```

Data Collection Scraping

Beautiful Soup is used to scrap for the Falcon9 Launch Records

We create a pandas data frame by parsing and converting the table

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

```
html_data = requests.get(static_url)
html_data.status_code
```

```
soup = BeautifulSoup(html_data.text,'html.parser')
```

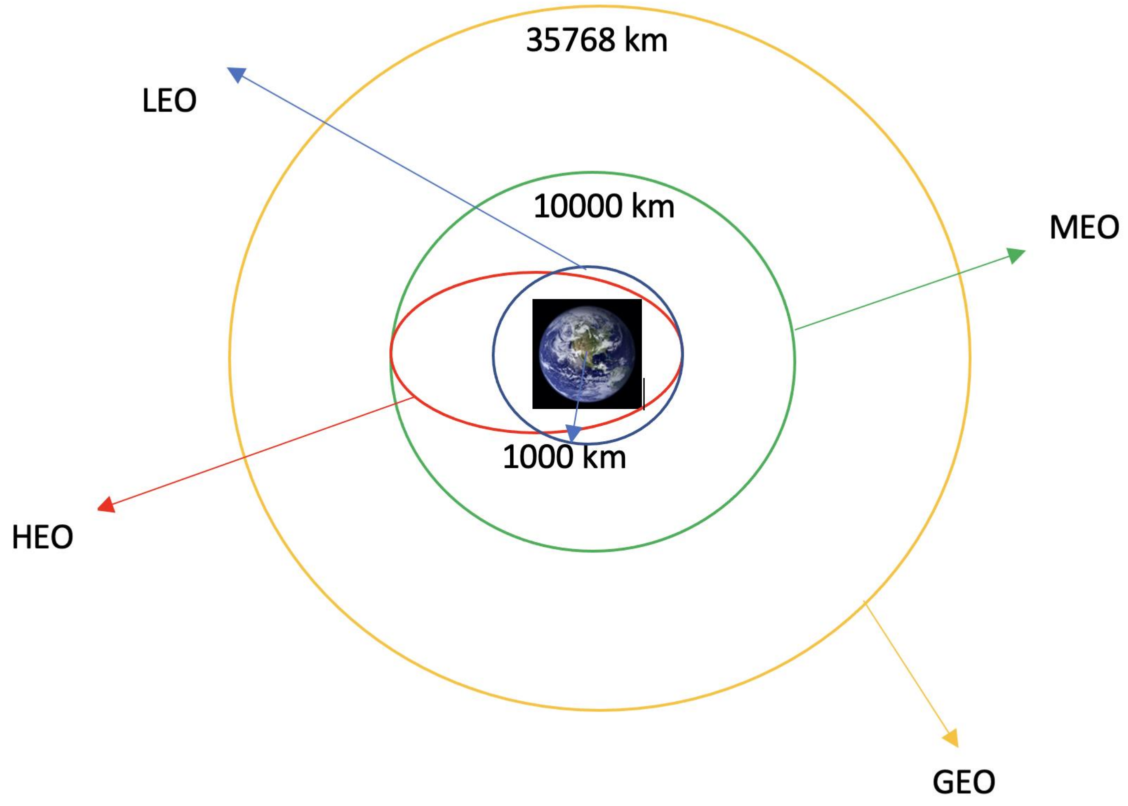
```
soup.title
```

https://github.com/katdown-code/IBM-Course-Work/blob/main/mod10/notebooks/part2-webscraping.ipynb

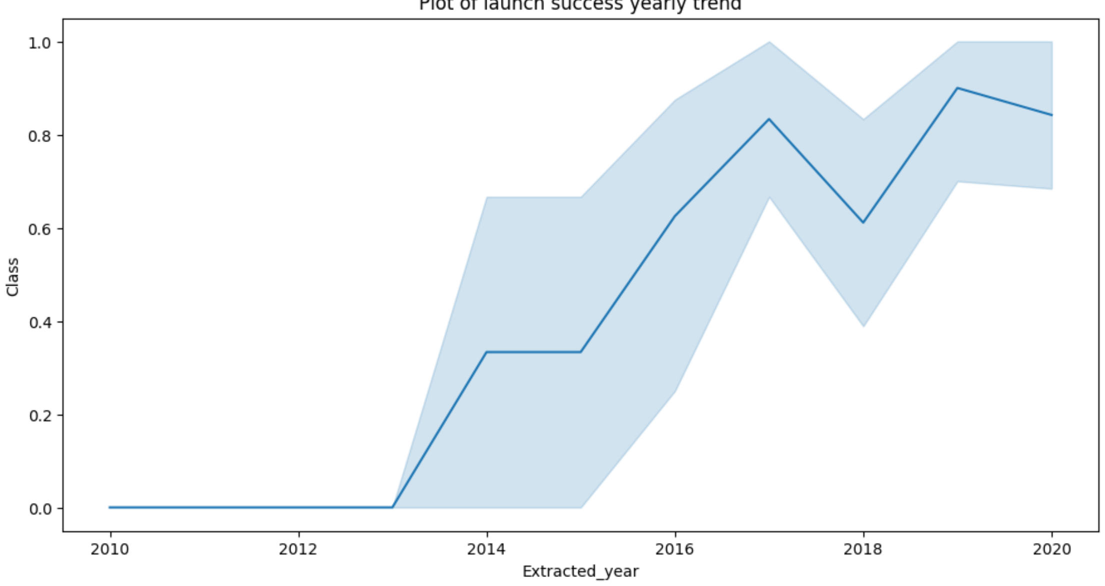
Data Wrangling

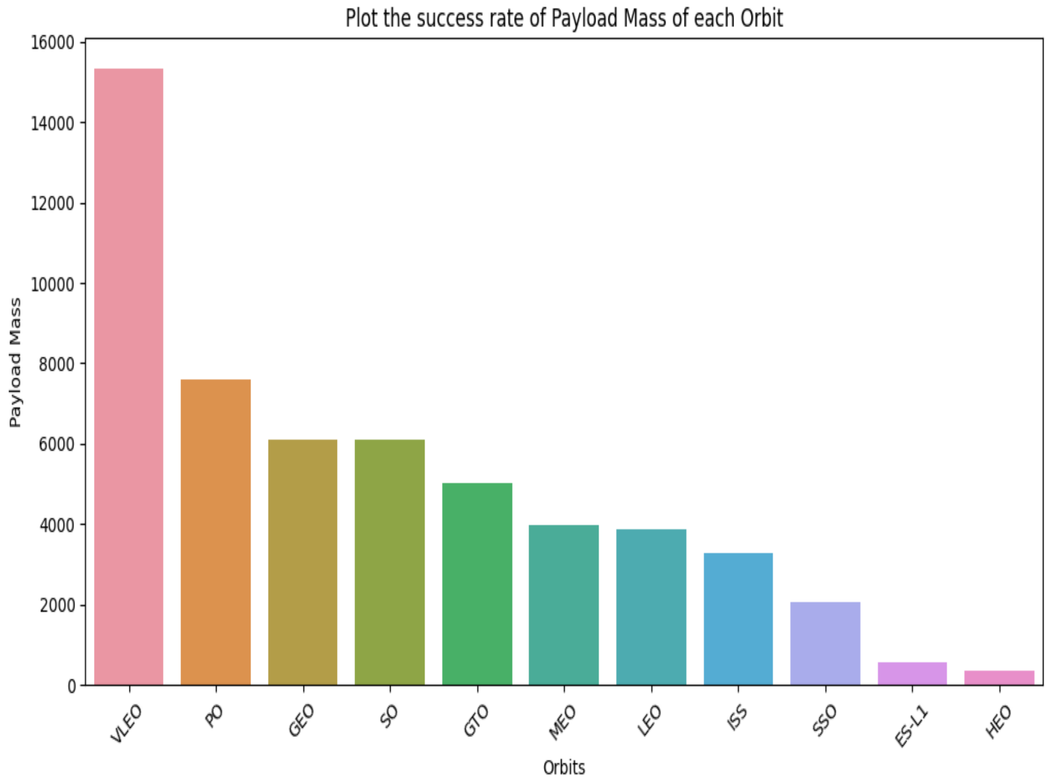
With EDA (Exploratory Data Analysis) to determine training labels we calculate the number of launches at each site and number of occurrences of each orbit

From the outcome column we can determine the landing outcomes and export them to a csv file



https://github.com/katdown-code/IBM-Course-Work/blob/main/mod10/notebooks/part3-data-wrangling.ipynb





EDA with Data Visualization

By exploring data and comparing the different columns we can build graphs that give us an idea of the story the data is telling us

https://github.com/katdown-code/IBM-Course-Work/blob/main/mod10/notebooks/part3-eda-dataviz.ipynb

EDA with SQL

We'll Use SQL for:

- Query for Booster Version, Landing Out Comes, Date, Launch Sites, and Payload Mass
- Look at where there have been successful launches and the size of their payload
- Compare the different boosters
- Compare successful and failed launches during specific periods

https://github.com/katdown-code/IBM-Course-Work/blob/main/mod10/notebooks/part4-sql.ipynb

Build an Interactive Map with Folium

We'll use circle and line markers to mark where there have been successes and failures on a folium map

We will use color to differentiate successes and failures

Using the map we will be able to determine if any locations perform better than average and answer questions like what is the ideal location for a launch site

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive Analysis (Classification)

https://github.com/katdown-code/IBM-Course-Work/blob/main/mod10/notebooks/part5-machine-learning.ipynb

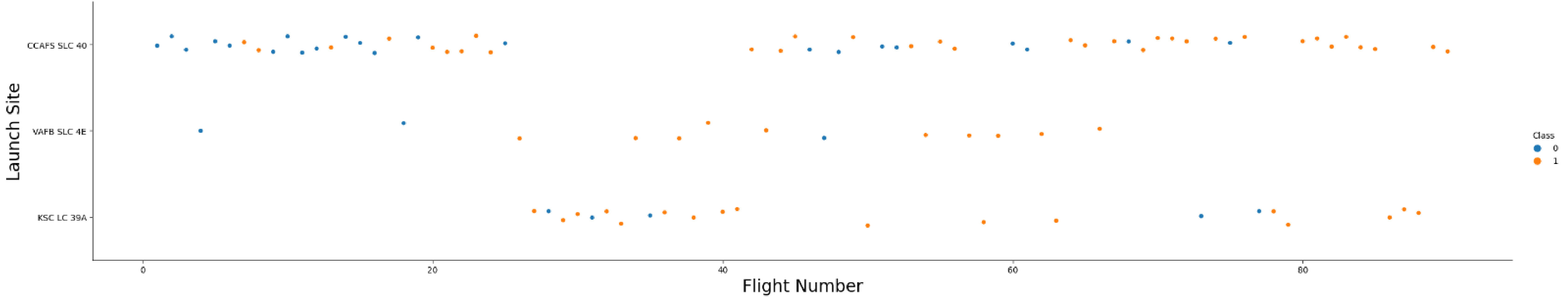
1. Collect data with Pandas and Transform with Numpy
2. Split the data randomly into test and train
3. Score accuracy of: Decision Tree, KNN, Logistic Regression, and Support Vectors
4. We find Decision Tree works best

Results

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

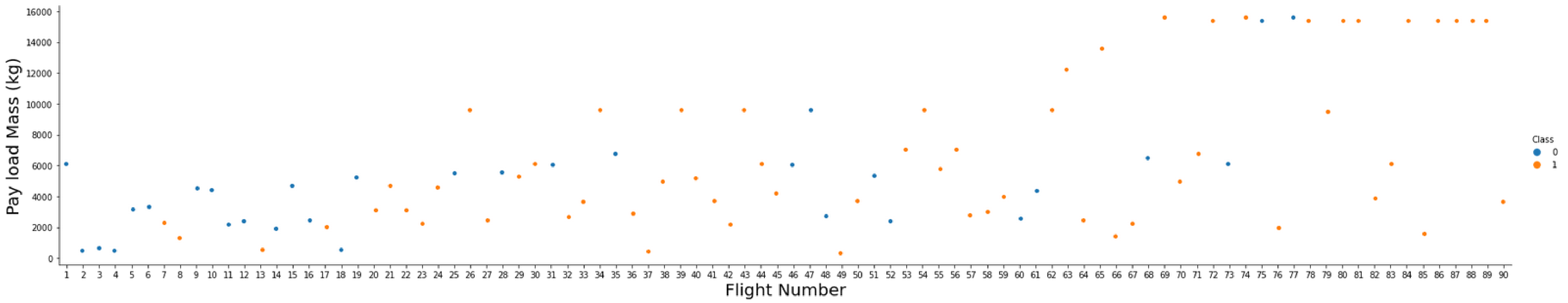
Section 2

Insights drawn from EDA



Flight Number vs. Launch Site

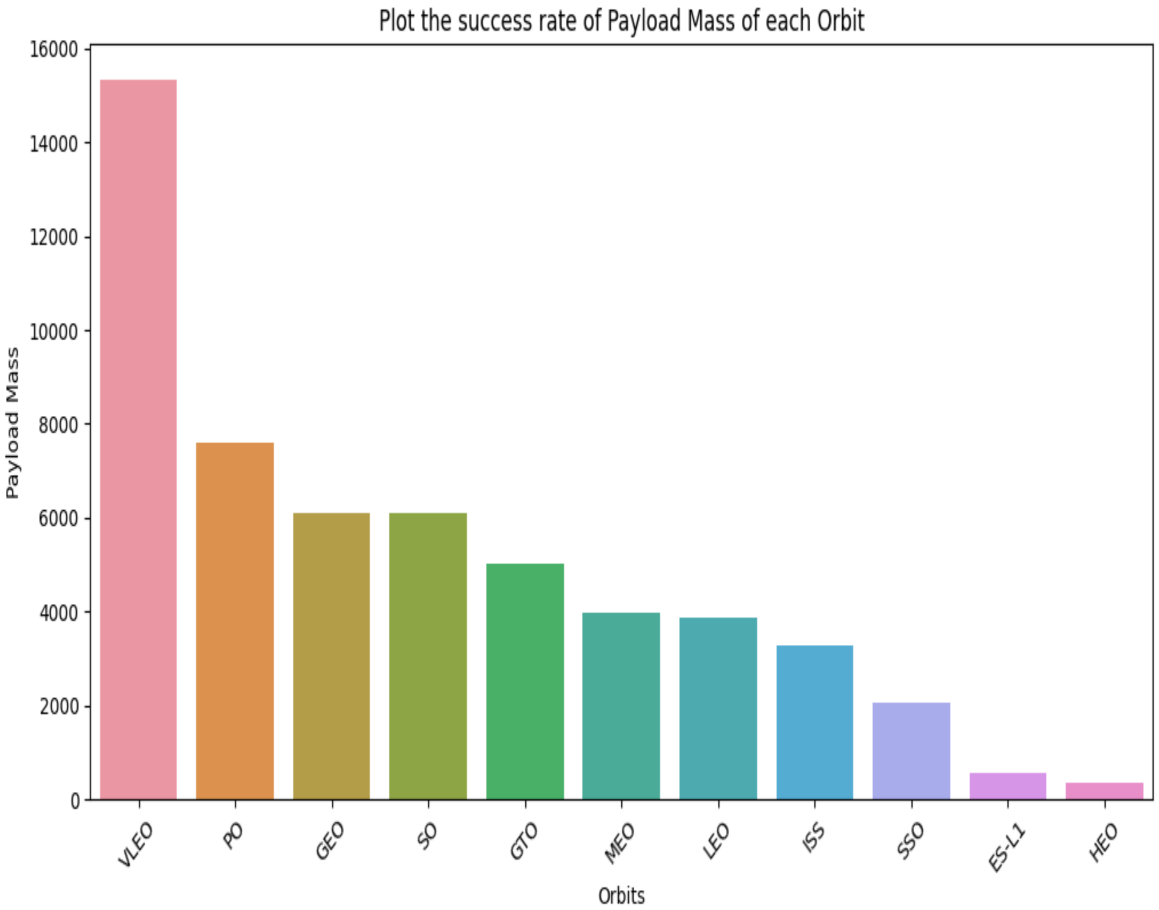
- Each dot represents the amount of Flights for each of the launch sites the farther to the right the higher the flight number
- It also shows the successes and failures with orange and blue



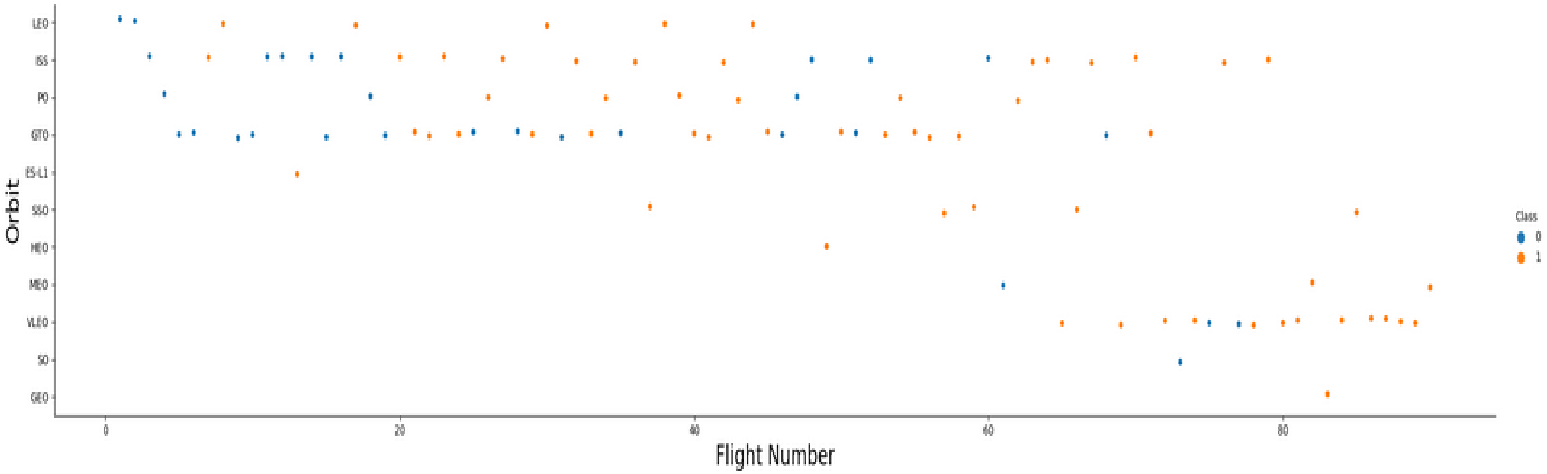
Payload vs. Launch Site

- Using the same kind of graph we can see that a launch with a higher flight number is more likely to success with a large payload then a lower flight number

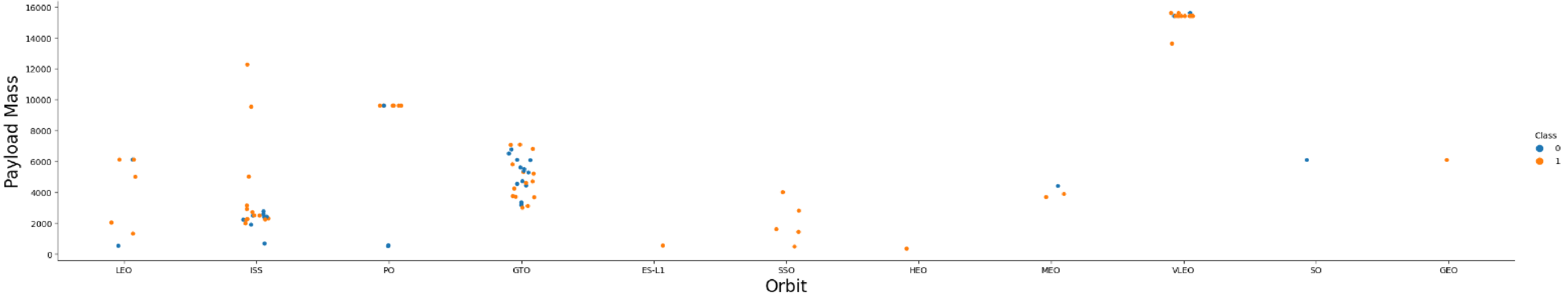
Success Rate vs. Orbit Type



Flight Number vs. Orbit Type



Using a scatter plot we can compare the success rate of flights in each orbit

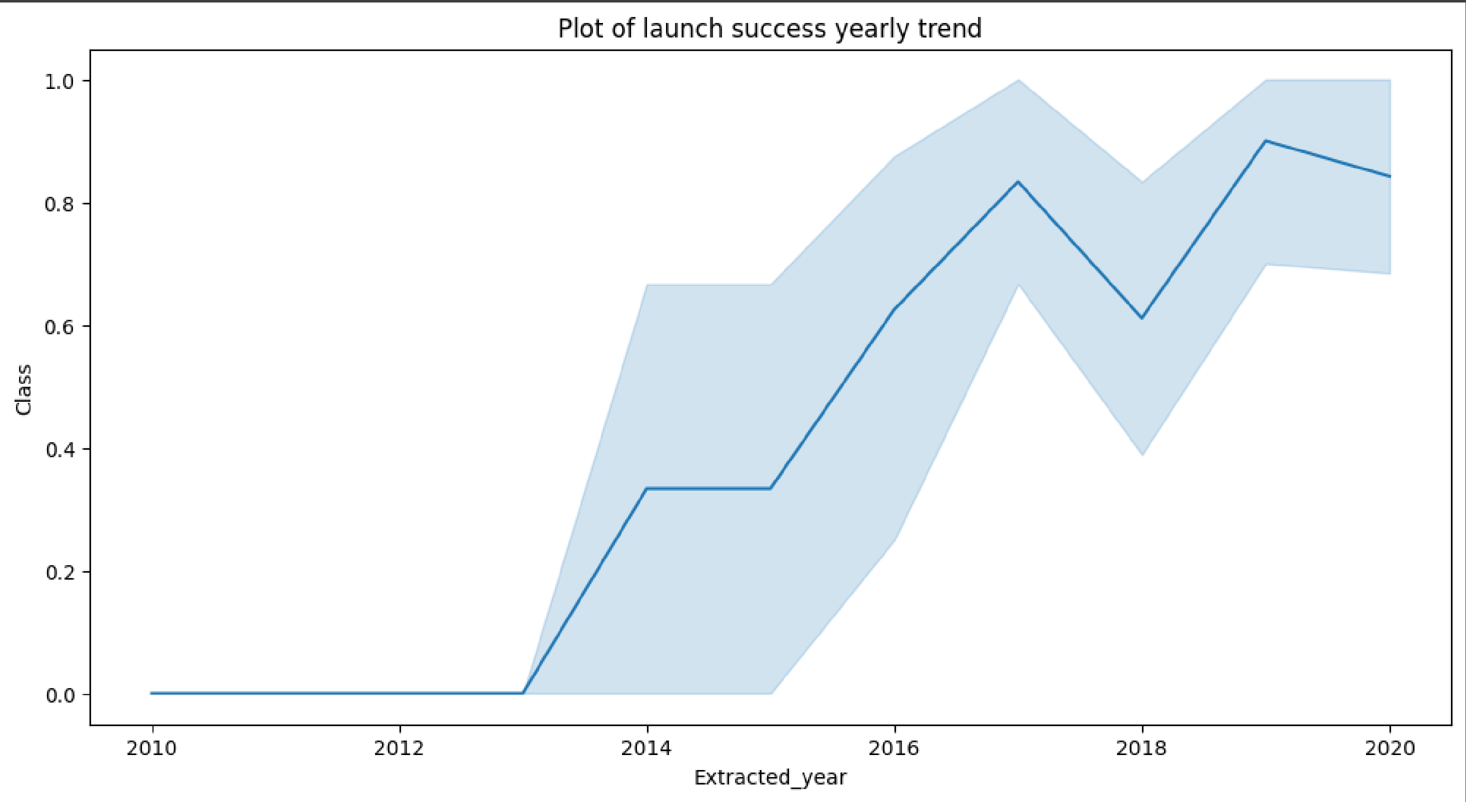


Payload vs. Orbit Type

With this chart we can see how the launches payload will affect its success at various orbits

Launch Success Yearly Trend

We can see that there has been a steep improvement of launches over the last decade



All Launch Site Names

Using a simple SQL query we can get all the launch sites

```
%%sql
SELECT DISTINCT launch_site FROM SpaceX
```

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

Launch Site Names Begin with 'CCA'

```
%%sql
SELECT DISTINCT launch_site FROM SpaceX
WHERE launch_site like 'CCA%'
```

launch_site
CCAFS LC-40
CCAFS SLC-40

We can even be more specific and get only the launch sites that meet our criteria

Total Payload Mass

```
%%sql
SELECT sum(payload_mass__kg_) FROM SpaceX
WHERE customer = 'NASA (CRS)'
```

22007

We can calculate the total payload carried by NASA to be 2207 kg

Average Payload Mass by F9 v1.1

```
%%sql
SELECT avg(payload_mass__kg_) FROM SpaceX
WHERE booster_version = 'F9 v1.1'
```

3676

We can query the average payload carried by the F9 v1.1 and see it is 3676

First Successful Ground Landing Date

We can see the first successful ground pad landing was in 2017

```
%%sql
SELECT DATE FROM SpaceX
where landing__outcome = 'Success (ground pad)'
ORDER BY DATE ASC
LIMIT 1
```

2017-01-05

Successful Drone Ship Landing with Payload between 4000 and 6000

We can make a where statement with multiple requirements to find the specific boosters

```
%%sql
SELECT booster_version FROM SpaceX
WHERE landing__outcome = 'Success (drone ship)'
    and payload_mass__kg_ > 4000
    and payload_mass__kg_ < 6000
```

F9 FT B1022
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

We can use GROUP BY to see how many times there is a failure or success

```
%%sql
SELECT landing__outcome,
    COUNT(landing__outcome) as "count"
FROM SpaceX
GROUP BY landing__outcome
```

landing__outcome	count
Controlled (ocean)	1
Failure	1
Failure (drone ship)	2
Failure (parachute)	2
No attempt	12
Success	18
Success (drone ship)	5
Success (ground pad)	4

In total there are
28 successes / controlled
17 failure / no attempt

Boosters Carried Maximum Payload

the names of the booster versions which have carried the maximum payload mass

Using subqueries

```
%%sql
SELECT DISTINCT booster_version FROM SpaceX
WHERE payload_mass__kg_ = (SELECT max(payload_mass__kg_) FROM SpaceX)
```

booster_version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1058.3
F9 B5 B1060.2

2015 Launch Records

List the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%%sql
SELECT booster_version,launch_site,landing__outcome,DATE FROM SpaceX
WHERE landing__outcome = 'Failure (drone ship)' and YEAR(DATE) = 2015
```

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

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

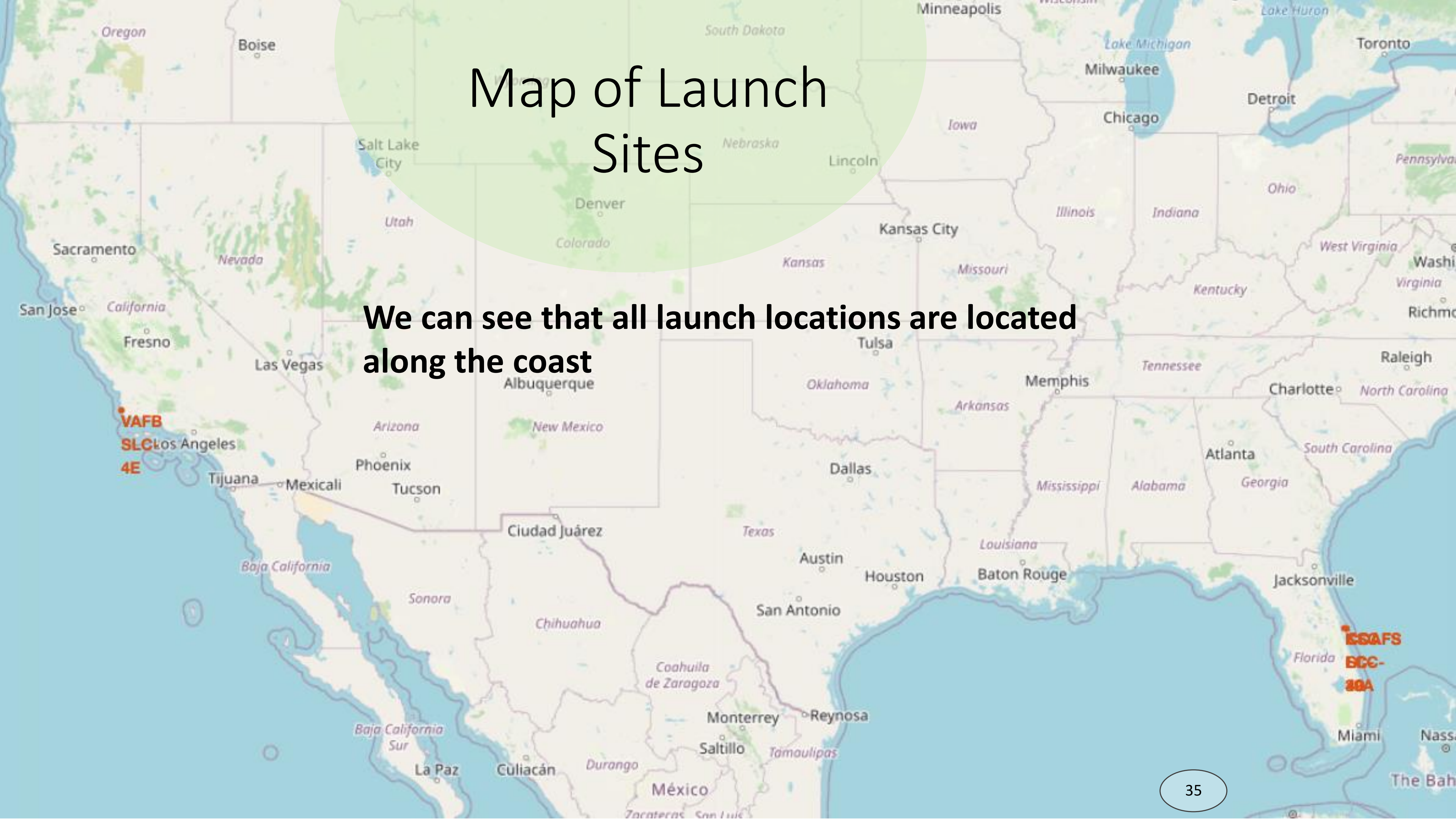
```
%%sql
SELECT landing__outcome, count(landing__outcome) as "count" FROM SpaceX
WHERE DATE <= '2017-03-20' and DATE >= '2010-06-04'
GROUP BY landing__outcome
```

landing__outcome	count
Controlled (ocean)	1
Failure (drone ship)	2
Failure (parachute)	1
No attempt	7
Success (drone ship)	2
Success (ground pad)	2

Section 3

Launch Sites Proximities Analysis

Map of Launch Sites

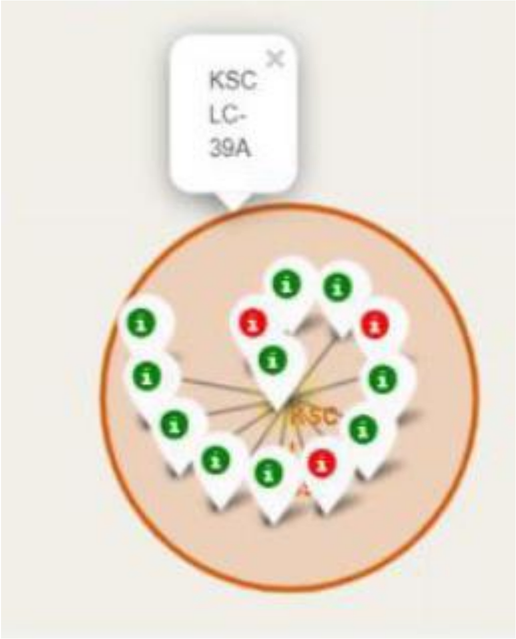


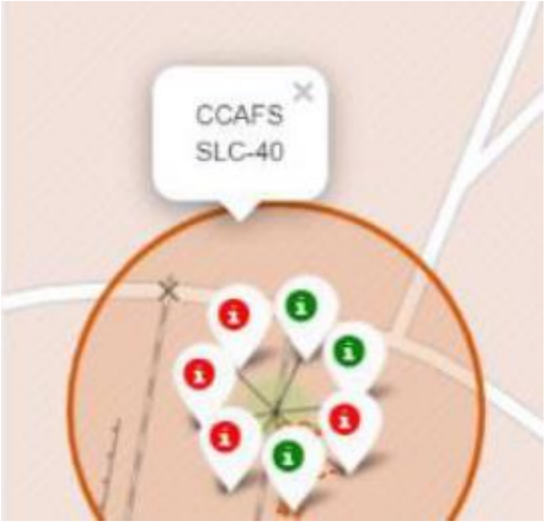
We can see that all launch locations are located along the coast

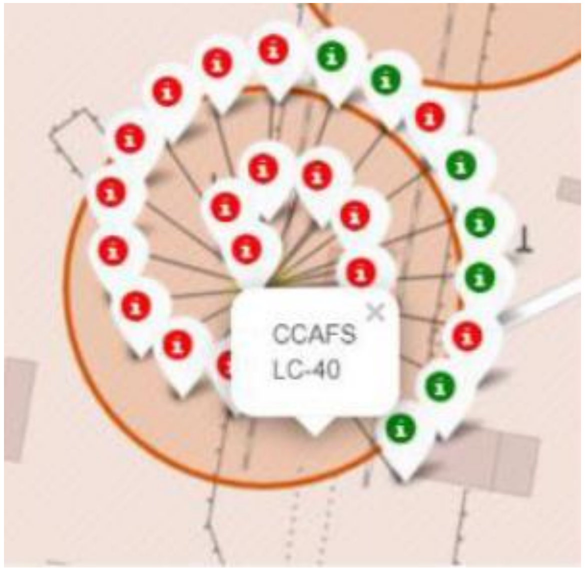
Folium Map Launch Outcomes

California Launches

Florida Launches







We can see that here are less locations in California for launches

Folium Map Launch Placement

Are Launches in close proximity to?:

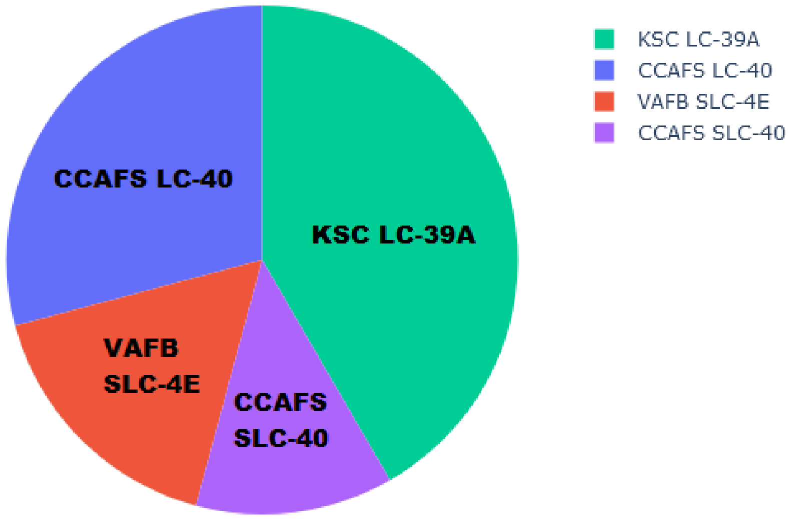
- Railways? - No
- Highways? - No
- Coastlines? - Yes
- A certain Distance from Cities- Yes

Closest to a Launch location

Section 4

Build a Dashboard with Plotly Dash

Total percentage of successful launches per launch site

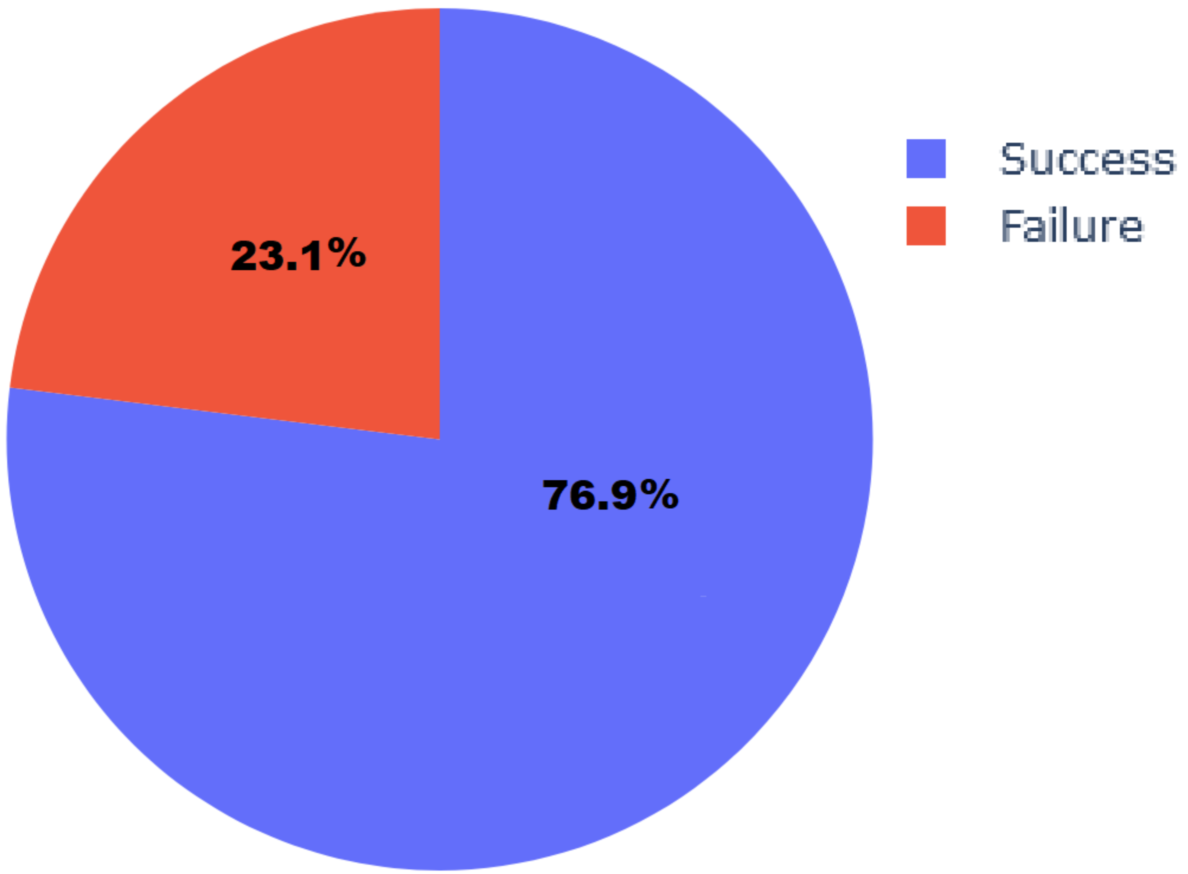


Dash Board Graph of successful launches

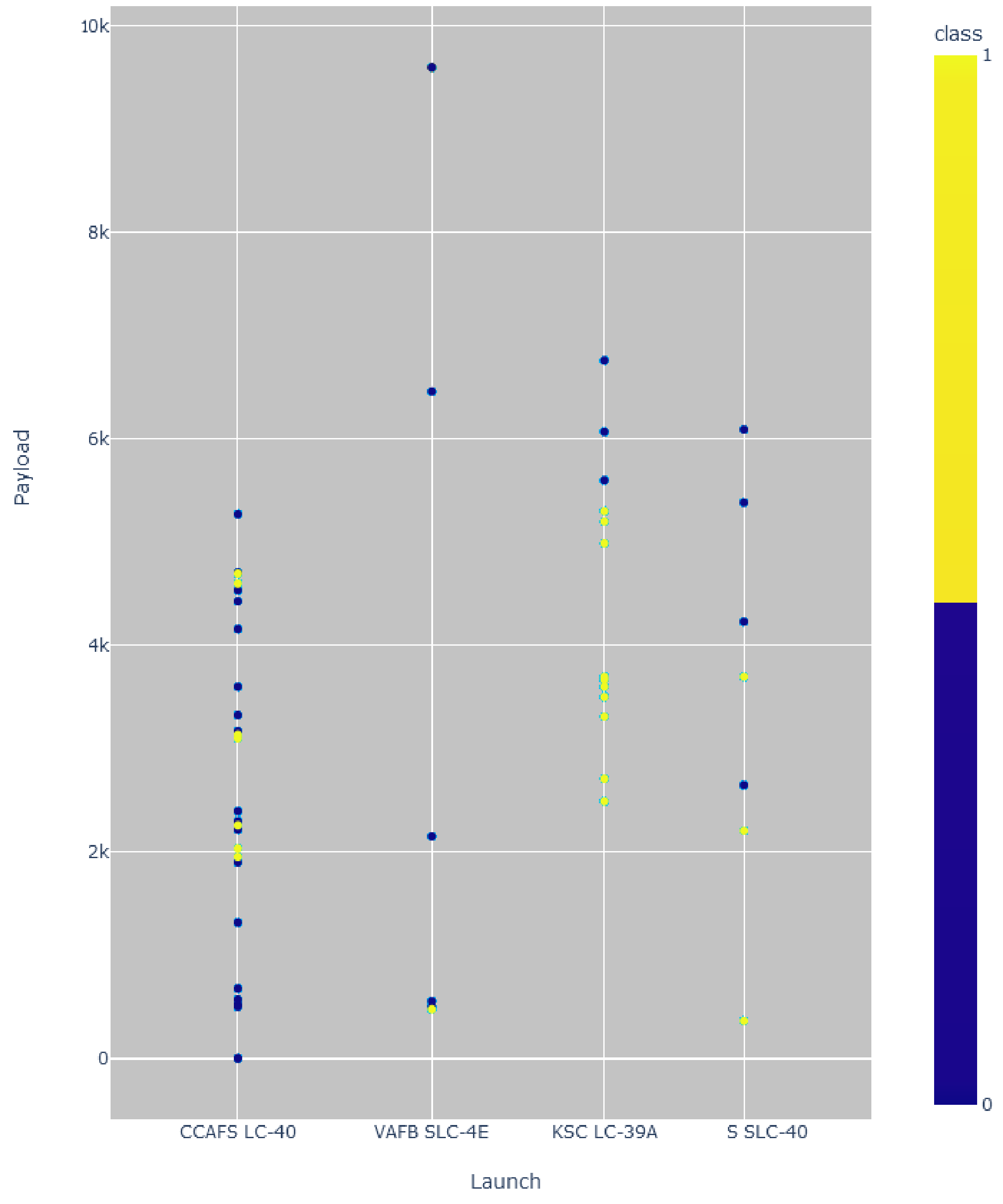
Here we see that the KSC LC-39A has the majority of the successful launches with nearly half

We also see CCAFS SLC-40 have the least successes

Rate of Success in the launch site with the most success



Looking closer at the real success rate of KSC LC-39A we can see it has 76.9% rate of success



Scatter plot of Payload vs Launch Outcome for all sites

Here we can see the ranges in which each launch site preforms best and worst

Section 5

Predictive Analysis (Classification)

Find the method performs best:

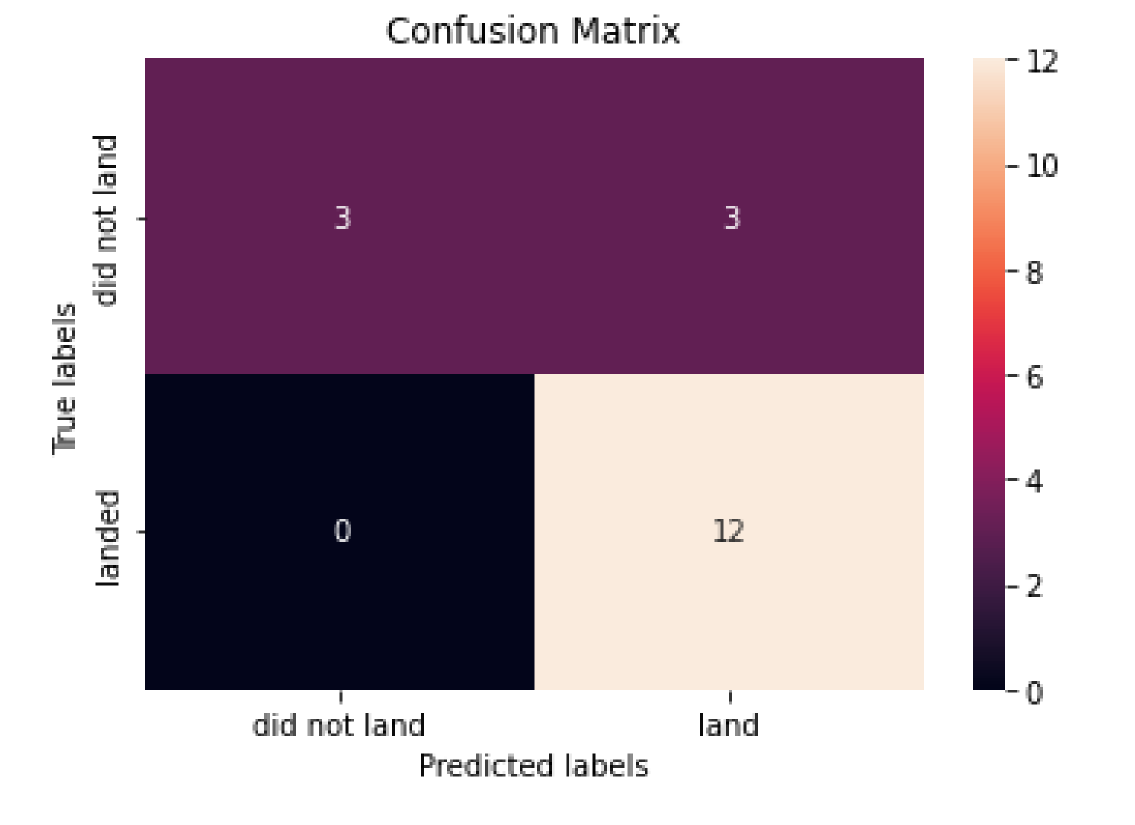
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8732142857142856
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 5, 'splitter': 'random'}
```

Classification Accuracy

The decision tree classifier is the model with the highest classification accuracy

Confusion Matrix



Shows the confusion matrix of the best performing model

Conclusions

We can conclude that:

• 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 Decision tree classifier is the best machine learning algorithm for this task.

Thank you!