IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

<Name>
<Date>

# Executive Summary

- This project used data obtained from API calls and web scraping on the topic of Space X Rocket launches
- Exploratory data analysis was then performed with python and SQL
- Interactive maps and dashboards were created with folium and plotly dash
- Finally predictive analysis with machine learning in Python was performed

# Introduction

- This project aims to answer the question of what factors influence the success of a Space X rocket landing, the language used in this project is python, and a little bit of SQL

Section 1

# Methodology

# Methodology

## Executive Summary

- Performed data wrangling
- Performed exploratory data analysis (EDA) using visualization and SQL
- Performed interactive visual analytics using Folium
- Performed predictive analysis using classification models

# Data Collection

- Made Get requests to the SPACEX API
- Also performed web scraping on publicly available tables of data
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/jupyter-labs-spacex-data-collection-api%20(2).ipynb
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/jupyter-labs-webscraping%20(1).ipynb

# Data Wrangling

- Removed NaN values and replaced them with the averages for the columns using the fillna() method
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling%20(1).ipynb

# EDA with Data Visualization

- Charts comparing payload mass, flight number, launch site, and outcome (class) were used.
- These charts let us better understand the relationship between mass of payload, time (flight number) and launch sites and potentially identify any relationships
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

# EDA with SQL

- SQL queries performed:
  - Displayed unique names of launch sites
  - Displayed 5 rows where launch site started with KSC
  - Displayed total payload mass of boosters launched by NASA
  - Displayed avg payload mass of F9 1.1 boosters
  - Displayed earliest date of successful landing outcome
  - Displayed names of boosters that had success for a certain weight range
  - Displayed total number of successful and failure mission outcomes
  - Displayed boosters that have carried the max payload mass with a subquery
  - Displayed successful ground pad landing outcome records of 2017
  - Displayed in descending order the different landing outcomes between 2 dates
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/jupyter-labs-eda-sql-edx_sqllite%20(1).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
- Explain why you added those objects
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

# 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)

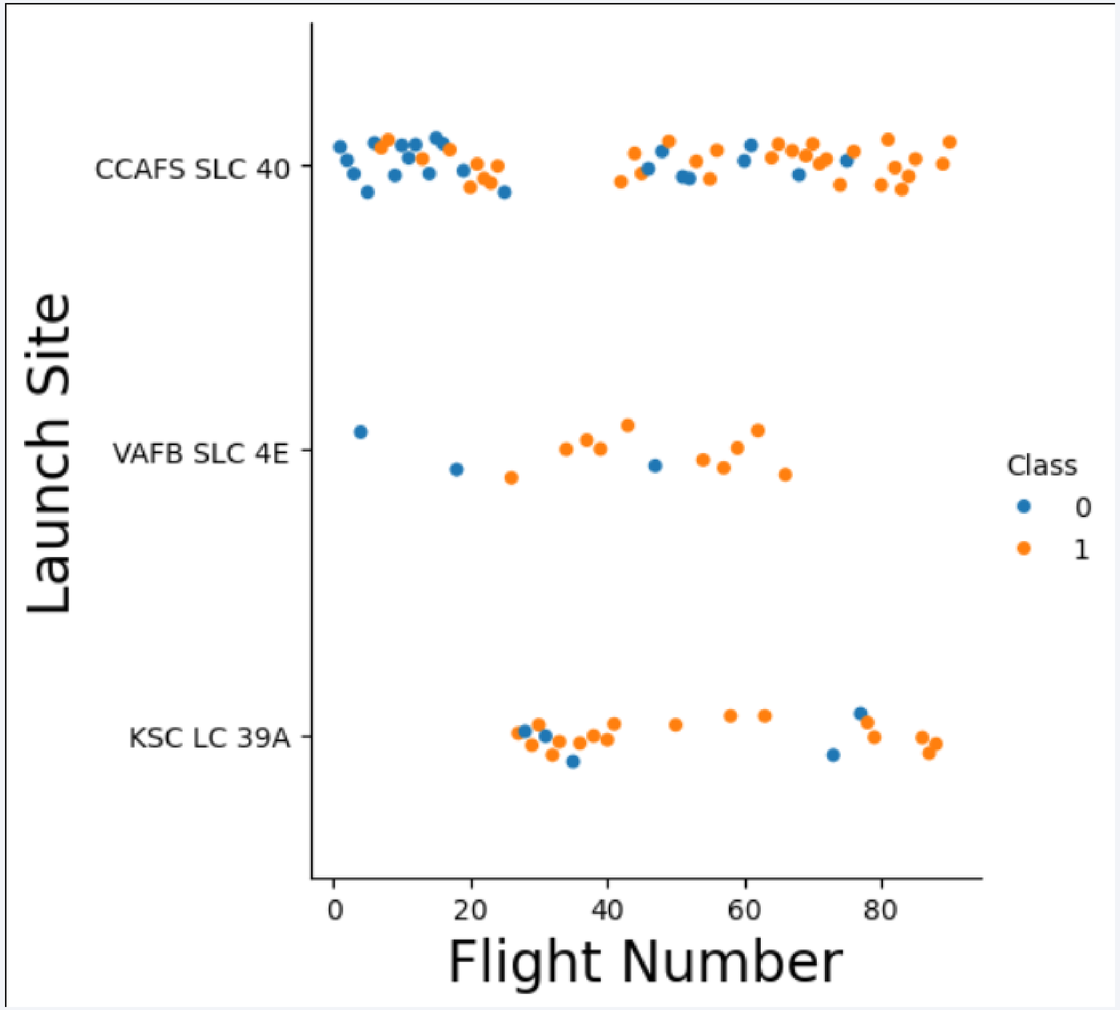
- Used a variety of methods to do predictive analysis on the data, including K nearest neighbors and decision trees. The data was split into training and testing splits for all methods used, and the method were all compared in their level of accuracy on the testing split of the data.
- https://github.com/AndrewScottFaris/IBM-Capstone/blob/main/Machine%20Learning.ipynb

Section 2

# Insights drawn from EDA

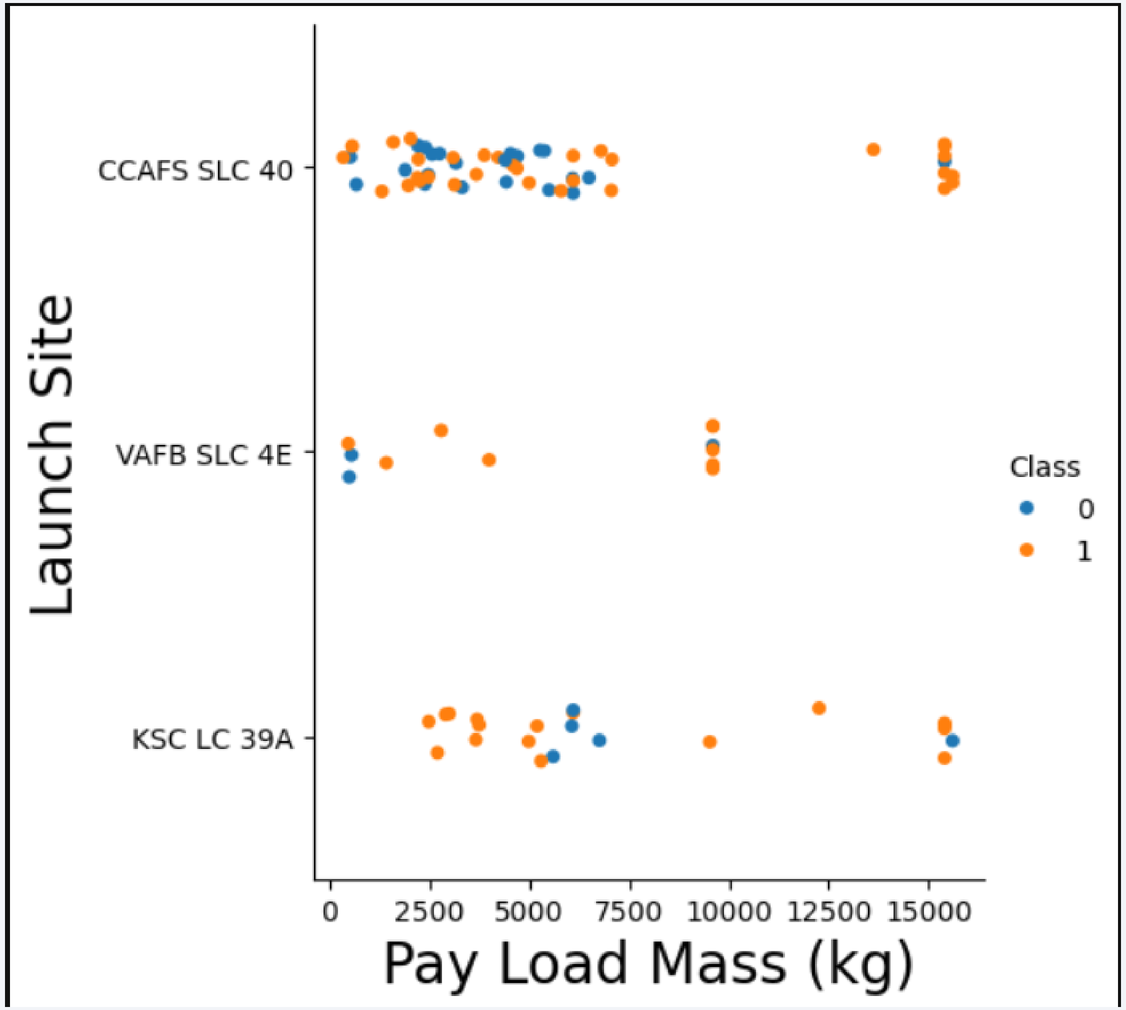
# Flight Number vs. Launch Site

- Scatter plot of Flight Number vs. Launch Site:
  - On this graph the Class variable refers to when the landing was unsuccessful (0), or successful (1)
  - We can see that for the CCAFS SLC 40 site, the early launches were often unsuccessful while the latest (highest flight number) launches were successful more frequently



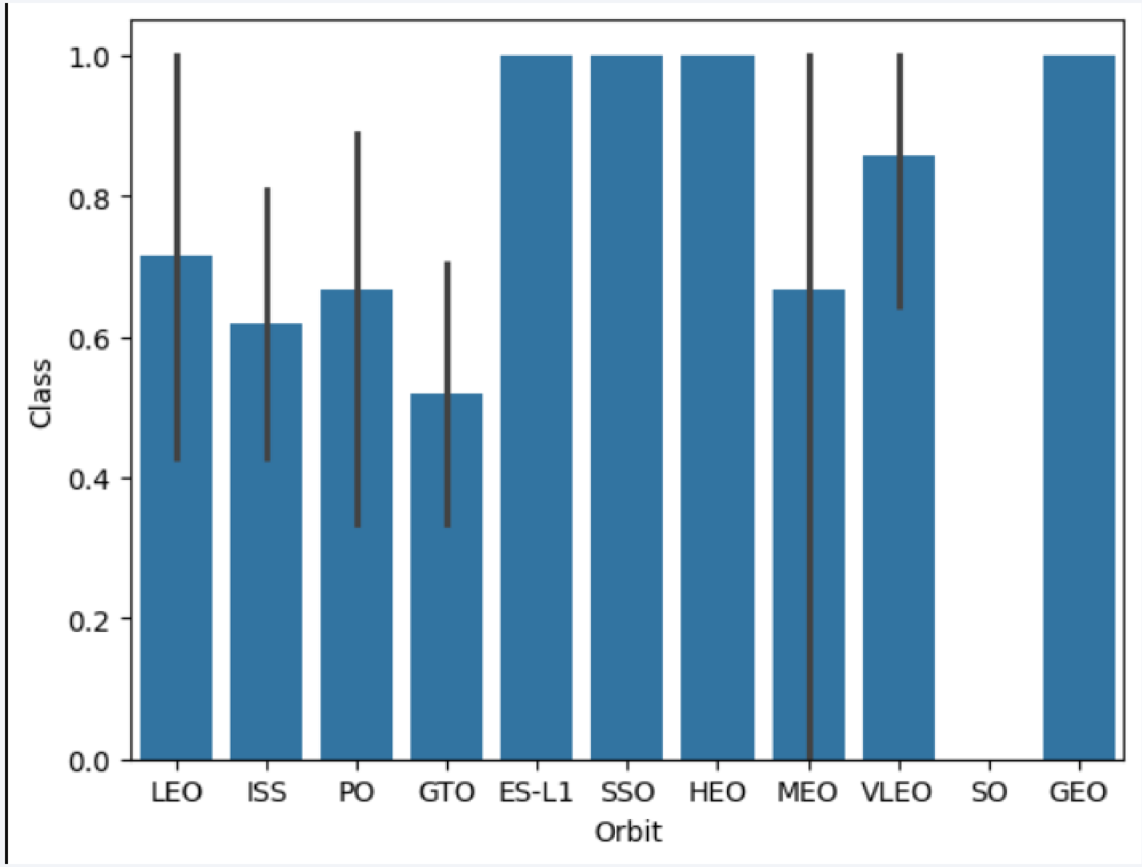
# Payload vs. Launch Site

- Scatter plot of Payload vs. Launch Site:
  - Again, here a class of 0 means unsuccessful and a class of 1 means successful
  - One insight we can see from this graph is that for the VAFB SLC 4E launch site, there were no payload masses greater than 10000 kg.



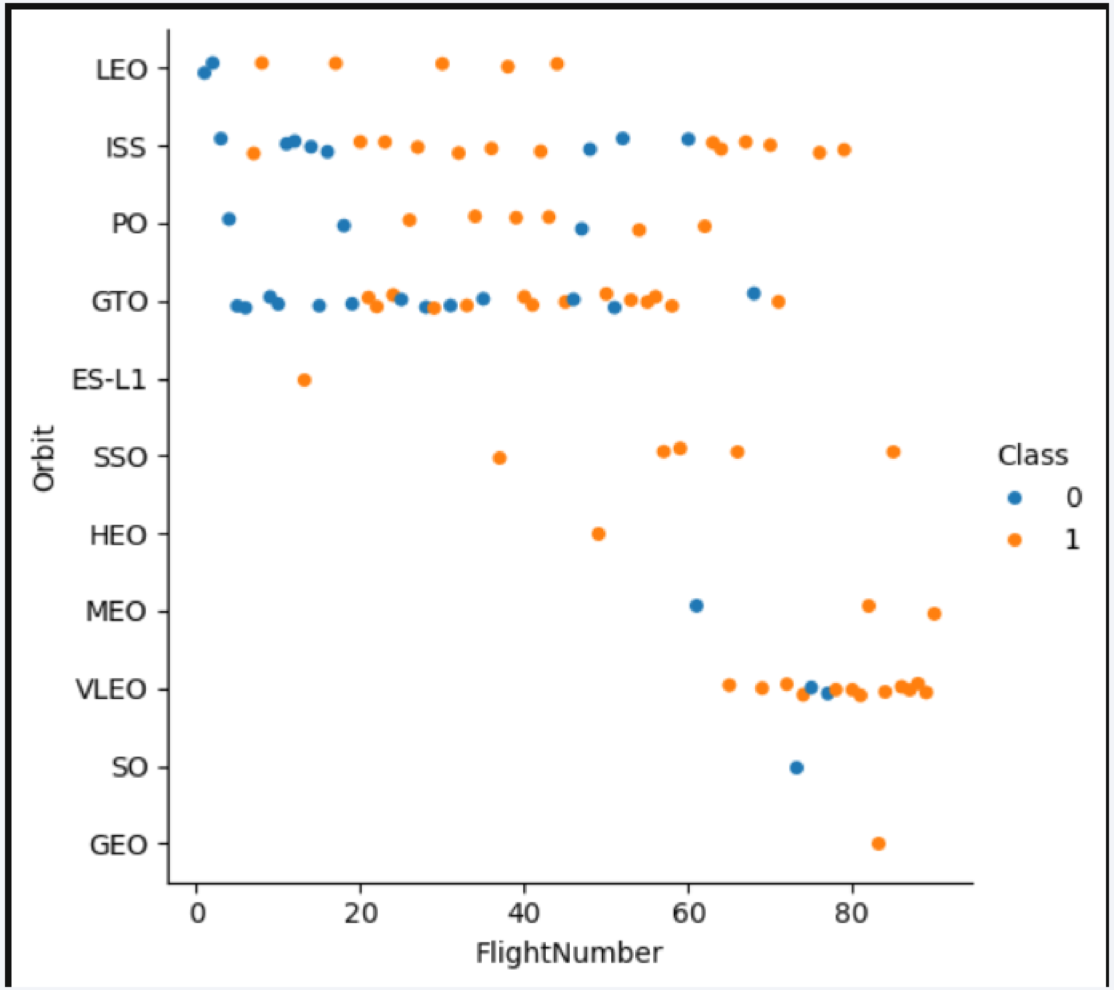
# Success Rate vs. Orbit Type

- Bar chart for the success rate of each orbit type:
  - Here the success rate can be seen on the y axis, with a class value of 1.0 denoting a 100% success rate, and a value of 0.0 denoting a 0% success rate
  - From this we can see that the ES-L1, SSO, HEO, and GEO had the highest success rates



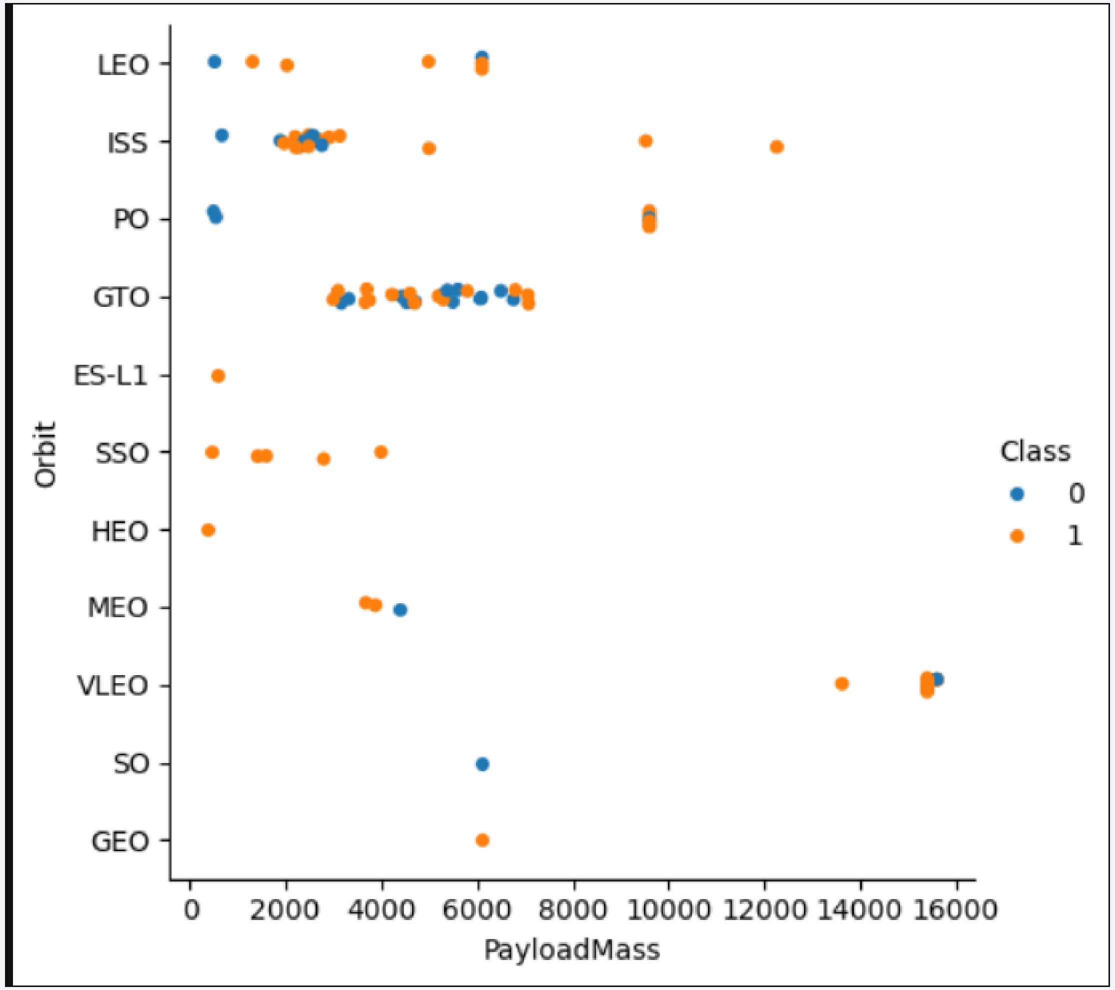
# Flight Number vs. Orbit Type

- Scatter point of Flight number vs. Orbit type:
  - Here we can see the different flight numbers for launches done with different orbit types
  - From this graph we can understand that LEO has a low average flight number and was generally done further in the past compared to VLEO for example.



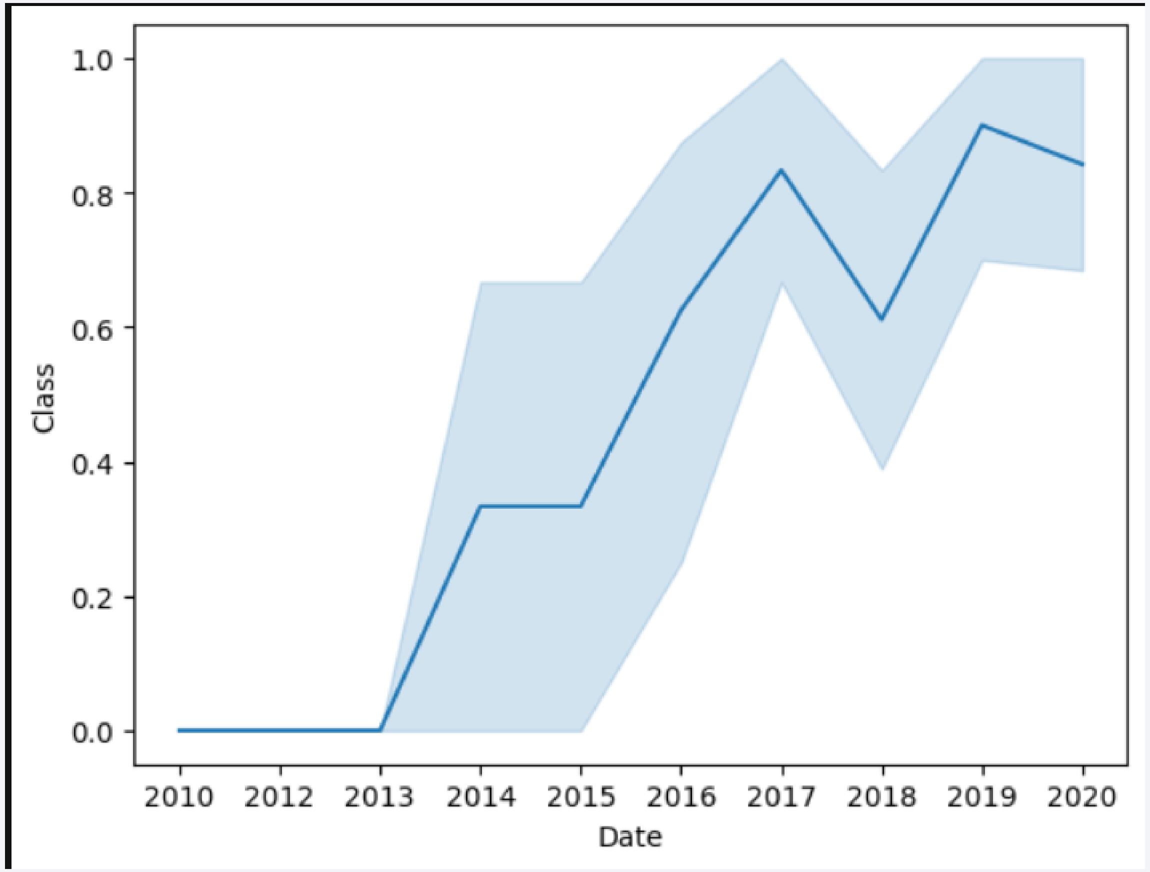
# Payload vs. Orbit Type

- Scatter point of payload vs. orbit type:
  - This graph plots the relationship between the Payload Mass and the orbit type
  - From this we can understand that SSO orbit types are only used when the payload is relatively light, as an example.



# Launch Success Yearly Trend

- Line chart of yearly average success rate:
  - From this chart we can observe an unsteady increase in the yearly average success rate since 2013



# All Launch Site Names

- The names of the unique launch sites:
- This query pulls all the distinct launch site names from the dataset SPACEXTABLE

```
%sql SELECT DISTINCT launch_site FROM SPACEXTABLE
```

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

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

# Launch Site Names Begin with 'KSC'

- 5 records where launch sites' names start with `KSC`
- This query pulls 5 rows of the data set where the "launch_site" column begins with KSC

```
%sql SELECT * FROM SPACEXTABLE WHERE launch_site LIKE 'KSC%' LIMIT 5
```

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

| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_Outcome |
|---|---|---|---|---|---|---|---|---|---|
| 2017-02-19 | 14:39:00 | F9 FT B1031.1 | KSC LC-39A | SpaceX CRS-10 | 2490 | LEO (ISS) | NASA (CRS) | Success | Success (ground pad) |
| 2017-03-16 | 6:00:00 | F9 FT B1030 | KSC LC-39A | EchoStar 23 | 5600 | GTO | EchoStar | Success | No attempt |
| 2017-03-30 | 22:27:00 | F9 FT B1021.2 | KSC LC-39A | SES-10 | 5300 | GTO | SES | Success | Success (drone ship) |
| 2017-05-01 | 11:15:00 | F9 FT B1032.1 | KSC LC-39A | NROL-76 | 5300 | LEO | NRO | Success | Success (ground pad) |
| 2017-05-15 | 23:21:00 | F9 FT B1034 | KSC LC-39A | Inmarsat-5 F4 | 6070 | GTO | Inmarsat | Success | No attempt |

# Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- This query sums the numbers in the payload mass column for all the rows where the customer is NASA.

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer LIKE 'NASA (CRS)'

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

| SUM(PAYLOAD_MASS__KG_) |
|---|
| 45596 |

# Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- This query calculates the average payload mass where the booster version in F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1'
```

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

| AVG(PAYLOAD_MASS__KG_) |
|---|
| 2928.4 |

# First Successful Ground Landing Date

- This query looks at all the entries where the mission outcome is success and returns that value in the date column which is the lowest out of that set of entries

```
%sql SELECT MIN(DATE) FROM SPACEXTABLE WHERE MISSION_Outcome LIKE 'Success'
```

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

| MIN(DATE) |
|---|
| 2010-06-04 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- (Note: the > 4000 is cutoff here but I still did it)
- This query finds the booster version where there was a success on a ground pad and the payload mass was between 4000 and 6000.

```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Success (ground pad)' AND PAYLOAD_MASS__KG_ < 6000
```

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

| Booster_Version |
|---|
| F9 FT B1032.1 |
| F9 B4 B1040.1 |
| F9 B4 B1043.1 |

# Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- This query counts the number of different outcomes for each mission, and it does this by using the group by statement

**List the total number of successful and failure mission outcomes**

```
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

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

| Mission_Outcome | TOTAL_NUMBER |
|---|---|
| Failure (in flight) | 1 |
| Success | 98 |
| Success | 1 |
| Success (payload status unclear) | 1 |

# Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- This query finds all the distinct boosters that have carried a mass equal to the maximum mass in the dataset

```
%sql SELECT Distinct(BOOSTER_VERSION) FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL
```

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

| 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 |

# 2017 Launch Records

- List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017
- This query uses a where clause and a substring to pull the year of 2017 from the date column

```
%sql SELECT * FROM SPACEXTABLE WHERE substr(Date,0,5) = '2017' AND Landing_Outcome LIKE 'Success (Ground Pad)'
```

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

| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_Outcome |
|---|---|---|---|---|---|---|---|---|---|
| 2017-02-19 | 14:39:00 | F9 FT B1031.1 | KSC LC-39A | SpaceX CRS-10 | 2490 | LEO (ISS) | NASA (CRS) | Success | Success (ground pad) |
| 2017-05-01 | 11:15:00 | F9 FT B1032.1 | KSC LC-39A | NROL-76 | 5300 | LEO | NRO | Success | Success (ground pad) |
| 2017-06-03 | 21:07:00 | F9 FT B1035.1 | KSC LC-39A | SpaceX CRS-11 | 2708 | LEO (ISS) | NASA (CRS) | Success | Success (ground pad) |
| 2017-08-14 | 16:31:00 | F9 B4 B1039.1 | KSC LC-39A | SpaceX CRS-12 | 3310 | LEO (ISS) | NASA (CRS) | Success | Success (ground pad) |
| 2017-09-07 | 14:00:00 | F9 B4 B1040.1 | KSC LC-39A | Boeing X-37B OTV-5 | 4990 | LEO | U.S. Air Force | Success | Success (ground pad) |
| 2017-12-15 | 15:36:00 | F9 FT B1035.2 | CCAFS SLC-40 | SpaceX CRS-13 | 2205 | LEO (ISS) | NASA (CRS) | Success | Success (ground pad) |

# 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
- This query pulls all the counts of the different landing outcomes between the 2 specified dates and then sorts them in descending order using DESC()

```
%sql SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTBL WHERE Date > 2010-06-04 GROUP BY Landing_Outcome ORDE
```

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

| Landing_Outcome | Outcome_Count |
|---|---|
| Success | 38 |
| No attempt | 21 |
| Success (drone ship) | 14 |
| Success (ground pad) | 9 |
| Failure (drone ship) | 5 |
| Controlled (ocean) | 5 |
| Failure | 3 |
| Uncontrolled (ocean) | 2 |
| Failure (parachute) | 2 |
| Precluded (drone ship) | 1 |
| No attempt | 1 |

Section 3

# Launch Sites Proximities Analysis

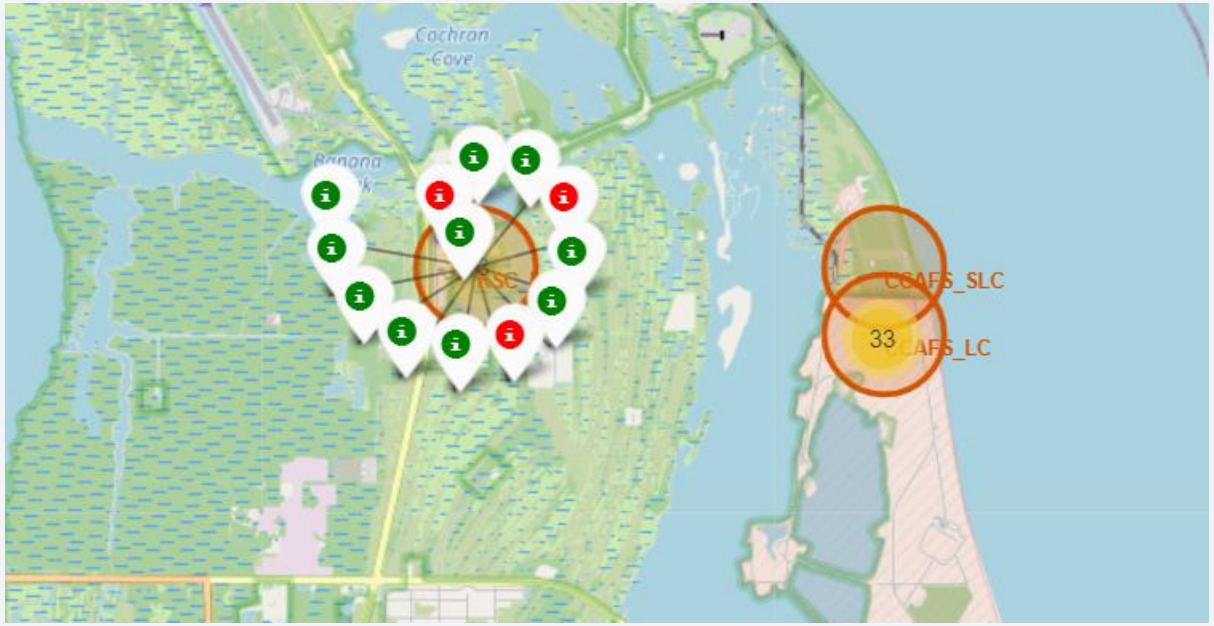
# Map of Launch Sites

- Here is a map of the launch sites


- Here we can see that they are all along the coast and all in the southern half of the US

# Map Color Coded with Success/Failure for a Launch site

- Here we can see the interactive map object, where if you click on a launch site, the successes and failures are displayed in a circle outward



# <Folium Map Screenshot 3>

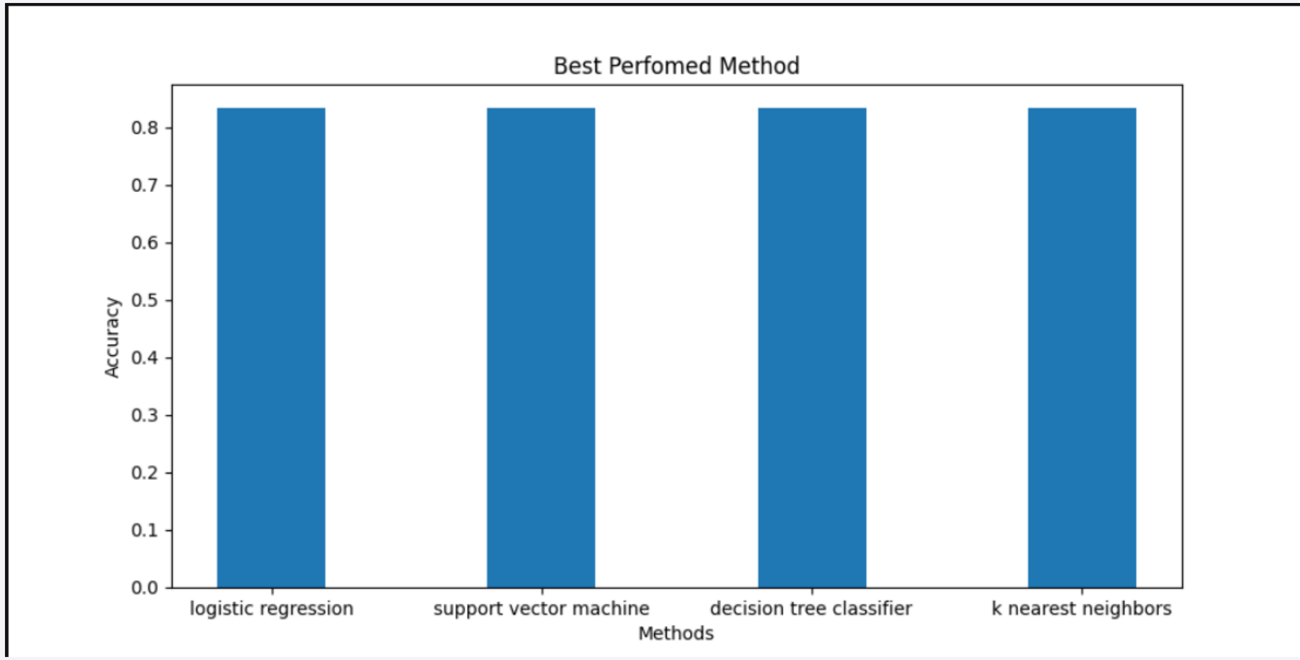
- Here is a map showing the distances from one launch site to the coast and to the nearest highway

Section 4

# Predictive Analysis (Classification)

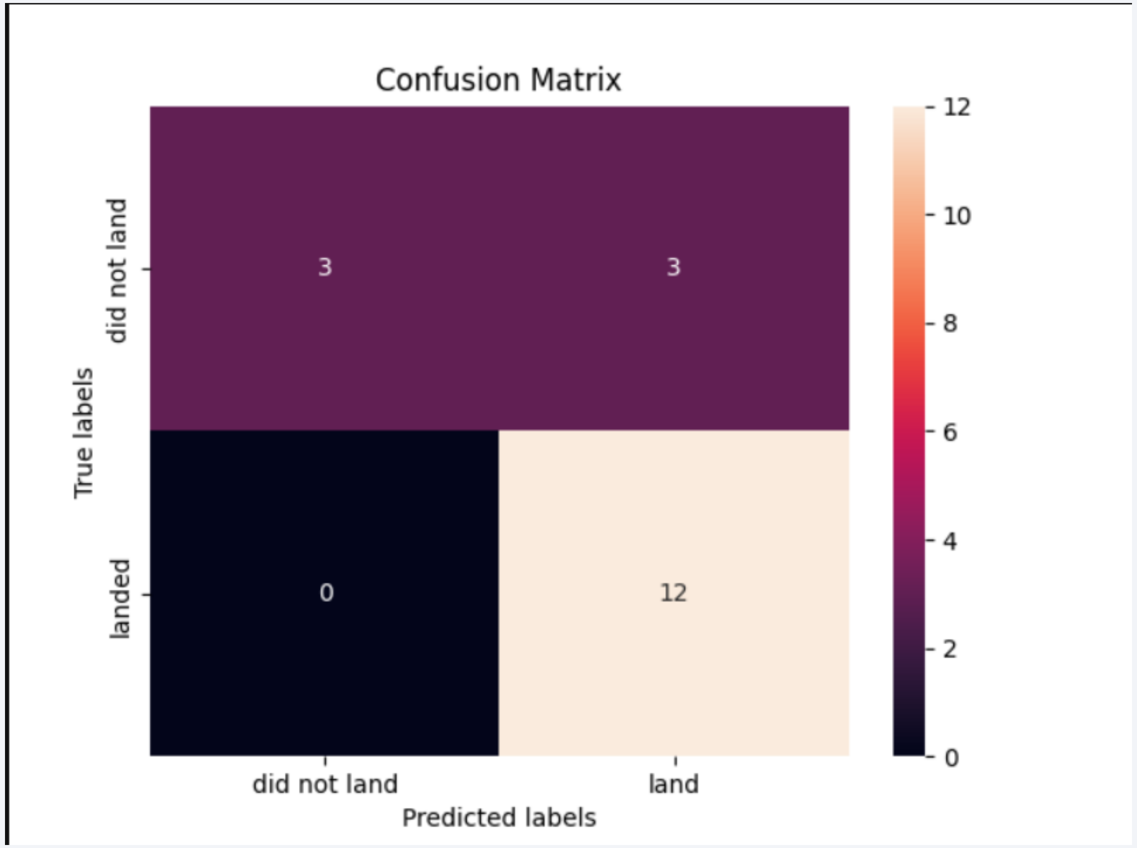
# Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart
- All models used had the same accuracy, 0.833



# Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation
- All models performed with equal accuracy and had identical confusion matrices.
- The models all have issues with false positives, i.e. they predict successful landings when in reality the landings are unsuccessful.



# Conclusions

- It is interesting that all the models used had identical accuracies and confusion matrices. This is likely due to the relatively small size of the data set, since there are less than 100 entries
- From the exploratory data analysis, it seems that the success rate has been increasing over time as more flights are flown. This is to be expected, as more experience should help prevent future failures

# Thank you!