IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Akanksha Sinha
31st January 2025

# Outline

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

# Executive Summary

- Summary of methodologies
- Collected data using SpaceX API and web scraping.
- Processed and cleaned data, converting categorical variables to binary.
- Conducted exploratory data analysis using SQL, visualizations, and Folium maps.
- Trained four machine learning models (Logistic Regression, SVM, Decision Tree, KNN).
- Achieved 83.33% accuracy, but models over-predicted successful landings.
- More data is needed for better model performance and accuracy.

# Introduction

- Project background and context

  SpaceX offers Falcon 9 rocket launches for $62 million , while other providers charge $165 million or more . The main reason for this cost difference is that SpaceX reuses the first stage of the rocket.

  If we can predict whether the first stage will land successfully, we can estimate the cost of a launch. This information is useful for Space Y , a company looking to compete with SpaceX for rocket launches. The goal of this project is to build a machine learning model that predicts Stage 1 landing success . .

- Problems you want to find answers
  - Space Y wants us to develop a machine learning model to predict whether Stage 1 of a rocket will land successfully.
  - To achieve this, we need to analyze:
  - Key factors affecting successful landings
  - How different features interact to influence success rates
  - Ideal operating conditions for a reliable landing program.

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection:
  - Collected data from the SpaceX API and Wikipedia using web scraping.
- Data wrangling
  - Cleaned and processed the data to ensure accuracy and consistency.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Utilized Folium and Plotly Dash for dynamic and interactive data exploration
- Perform predictive analysis using classification models
  - Tuned machine learning models and assessed their performance for accuracys

# Data Collection

- The data was collected using various methods
  - Retrieved data from the SpaceX API using GET requests and converted the JSON response into a Pandas DataFrame .
  - Cleaned the data, handled missing values, and ensured consistency.
  - Scraped Falcon 9 launch records from Wikipedia using BeautifulSoup .
  - Extracted and processed data from HTML tables into a structured format.
- Key Data Columns
  - API Data: Flight Number, Date, Booster Version, Payload Mass, Orbit, Launch Site, Outcome, Flights, Reused Components, Landing Pad, etc.
  - Web Scraped Data: Flight Number, Launch Site, Payload, Orbit, Customer, Booster Version, Landing Outcome, Date, Time, etc.

# Data Collection – SpaceX API

- We retrieved data from the SpaceX API using a GET request, processed the extracted information, and performed basic data cleaning, transformation, and formatting.).

- You can find the notebook at: [GitHub Repository](https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-1-handson-Complete%20the%20Data%20Collection%20API%20Lab.ipynb)

1. Get request for rocket launch data using API

In [6]:
```python
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```python
response = requests.get(spacex_url)
```

2. Use json_normalize method to convert json result to dataframe

In [12]:
```python
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```python
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

In [30]:
```python
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

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

# Data Collection - Scraping

- We applied web scrapping to webscrap Falcon 9 launch records with BeautifulSoup
- We parsed the table and converted it into a pandas dataframe.
- The link to the notebook is https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-1-handson-Data%20Collection%20with%20Web%20Scraping%20lab.ipynb

```
# 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')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
# Use soup.title attribute
print(soup.title)
```

```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

### TASK 2: Extract all column/variable names from the HTML table header

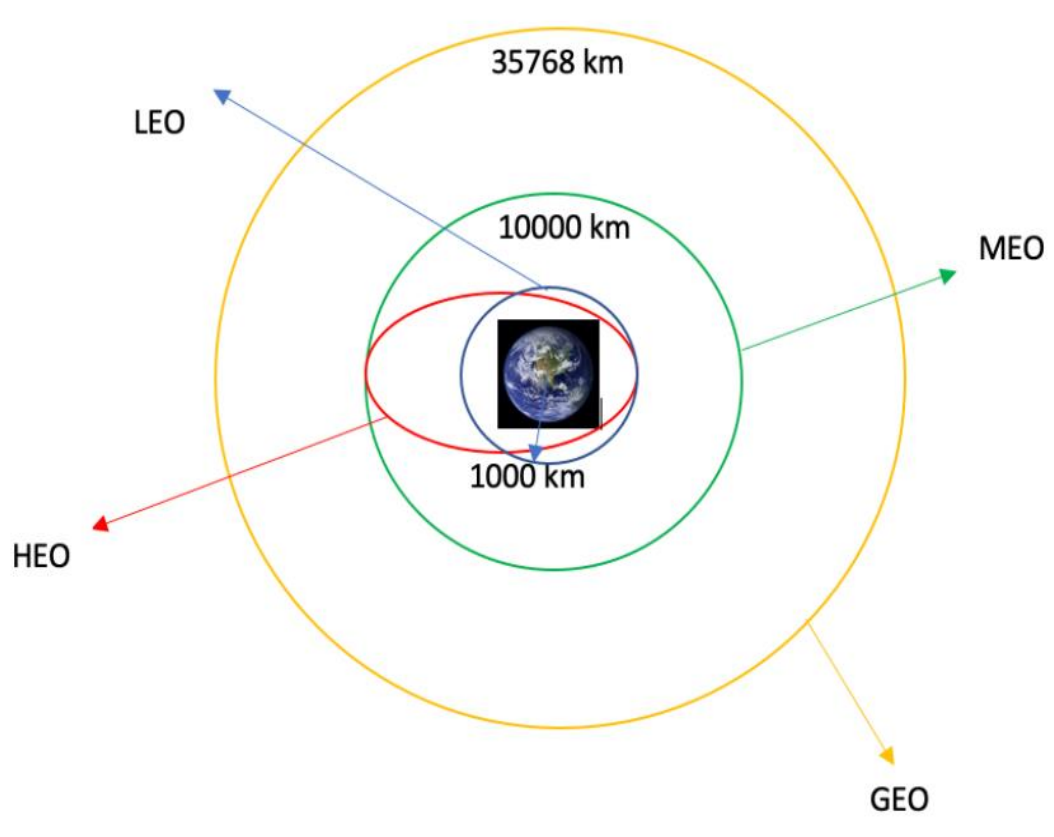
Next, we want to collect all relevant column names from the HTML table header

Let's try to find all tables on the wiki page first. If you need to refresh your memory about `BeautifulSoup`, please check the external reference link towards the end of this lab

```
# Use the find_all function in the BeautifulSoup object, with element type `table`
# Assign the result to a list called `html_tables`
html_tables = soup.find_all("table")
print(html_tables)
```

```
[<table class="col-begin" role="presentation">
<tbody><tr>
<td class="col-break">
<div class="mw-heading mw-heading3"><h3 id="Rocket_configurations">Rocket configurations</h3></div>
<div class="chart noresize" style="padding-top:10px;margin-top:1em;max-width:420px;">
<div style="position:relative;min-height:320px;min-width:420px;max-width:420px;">
<div style="float:right;position:relative;min-height:240px;min-width:320px;max-width:320px;border-left:1px black solid;border
-bottom:1px black solid;">
<div style="position:absolute;left:3px;top:224px;height:15px;min-width:18px;max-width:18px;background-color:LightSteelBlue;-w
ebkit-print-color-adjust:exact;border:1px solid LightSteelBlue;border-bottom:none;overflow:hidden;" title="[[Falcon 9 v1.0]]:
2"></div>
<div style="position:absolute;left:55px;top:224px;height:15px;min-width:18px;max-width:18px;background-color:LightSteelBlue;-
webkit-print-color-adjust:exact;border:1px solid LightSteelBlue;border-bottom:none;overflow:hidden;" title="[[Falcon 9 v1.
0]]: 2"></div>
<div style="position:absolute;left:81px;top:232px;height:7px;min-width:18px;max-width:18px;background-color:LightSteelBlue;-w
ebkit-print-color-adjust:exact;border:1px solid LightSteelBlue;border-bottom:none;overflow:hidden;" title="[[Falcon 9 v1.0]]:
1"></div>
<div style="position:absolute;left:81px;top:216px;height:15px;min-width:18px;max-width:18px;background-color:SteelBlue;-webki
t-print-color-adjust:exact;border:1px solid SteelBlue;border-bottom:none;overflow:hidden;" title="[[Falcon 9 v1.1]]: 2"></div
>
<div style="position:absolute;left:107px;top:192px;height:47px;min-width:18px;max-width:18px;background-color:SteelBlue;-webk
it-print-color-adjust:exact;border:1px solid SteelBlue;border-bottom:none;overflow:hidden;" title="[[Falcon 9 v1.1]]: 6"></di
```

# Data Wrangling




- The CSV file from the first section contained the data in need of cleaning/wrangling.
- The launch sites, orbit types and mission outcomes were processed and reformatted.
- The mission outcome types were converted to a binary classification (one-hot encoding) where 1 represented the Falcon 9 first stage landing being a success and 0 represented a failure.
- The new mission outcome classification column was added to the DataFrame.
- GitHub URL (Data Wrangling):
- The link to the notebook is https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-1-handson-Data%20Wrangling.ipynb.

# EDA with Data Visualization

**The following charts were created to look at Launch Site trends**

•Scatterplot to see **mission outcome** relationship split by **Launch Site** and **Flight Number**.

•Scatterplot to see **mission outcome** relationship split by **Launch Site** and **Payload**.

**The following charts were created to look at Orbit Type trends**

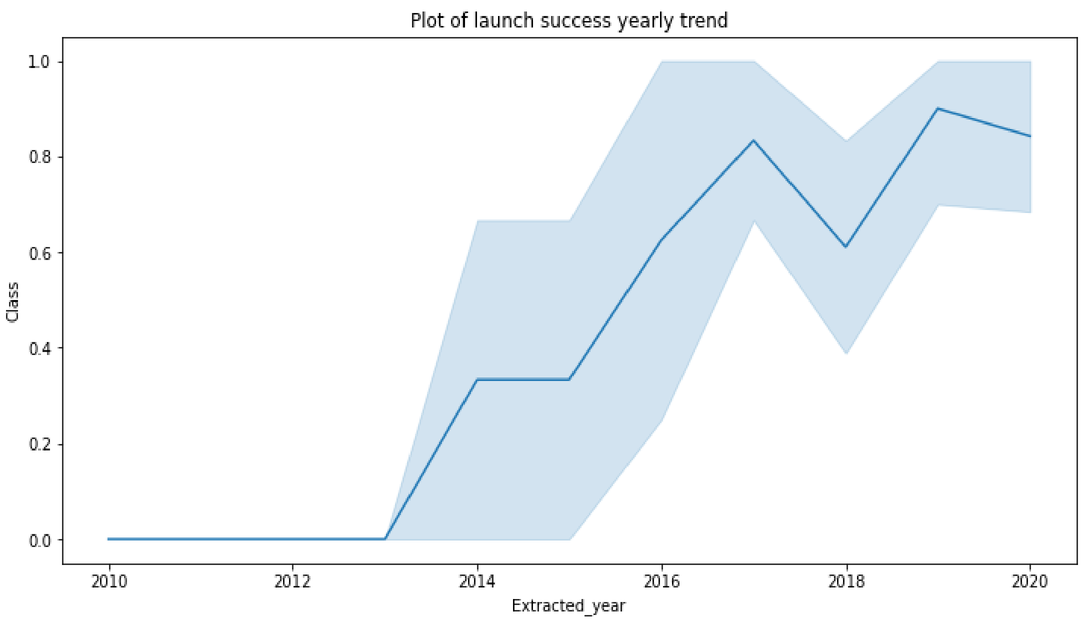
•Bar chart to see **mission outcome** relationship with **Orbit Type**.

•Scatterplot to see **mission outcome** relationship split by **Orbit Type** and **Flight Number**.

•Scatterplot to see **mission outcome** relationship split by **Orbit Type** and **Payload**.

**The following chart was created to look at trends based on time**

•Line plot to see **mission outcome** trend by **year**.




The link to the notebook is: https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-2-handson-EDA%20with%20Visualization%20Lab.ipynb

General

# EDA with SQL

- SQL queries were written to extract information about:

•Launch sites

•Payload masses

•Dates

•Booster types

•Mission outcomes

•GitHub URL (EDA with SQL):

- The link to the notebook is https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-2-handson-Complete%20the%20EDA%20with%20SQL.ipynb

# Build an Interactive Map with Folium

•Map objects were created and added to the Folium map

- Markerswere added for launch sites and for the NASA Johnson Space Center
- Circleswere added for the launch sites.
- Lineswere added to show the distance to the nearby features:
- Distance from CCAFS LC-40 to the perimeter road
- Distance from CCAFS LC-40 to the coastline
- Distance from CCAFS LC-40 to the rail line
- We 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.
- We assigned the feature launch outcomes (failure or success) to class 0 and 1.i.e., 0 for failure, and 1 for success.
- 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:
- Github URL: https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-3-handson-Interactive%20Visual%20Analytics%20with%20Folium%20lab.ipynb.

# Build a Dashboard with Plotly Dash

- The Plotly Dash dashboard included a dropdown input to select data from ‘one’ or ‘all’ launch sites to display on the pie chart and scatterplot.
- For ‘one’ launch site, the pie chart displayed the distribution of successful and failed Falcon 9 first stage landings for that site.
- For ‘all’ launch sites, the pie chart displayed the distribution of successful Falcon 9 first stage landings between the sites.
- The input slider is used to filter the payload masses for the scatterplot.
- The scatterplot displayed the distribution of Falcon 9 first stage landings split by payload mass, mission outcome and by booster version category.
- The link to the notebook is https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/spacex_dash_app.py

# Predictive Analysis (Classification)

- The dataset was split into training and testing sets.
- The following machine learning models were trained on the training data set:
- Logistic Regression
- SVM (Support Vector Machine)
- Decision Tree
- KNN (k-Nearest Neighbors)
- Hyper-parameters were evaluated using GridSearchCV() and the best was selected using the best_paramsmethod.
- Using the best hyper-parameters, each of the four models were scored on accuracy by using the testing data set.
- The link to the notebook is https://github.com/akanksha-s1nha/Data-Science-Capstone-SpaceX/blob/main/Module-4-handson-Machine%20Learning%20Prediction%20lab.ipynb

# Results

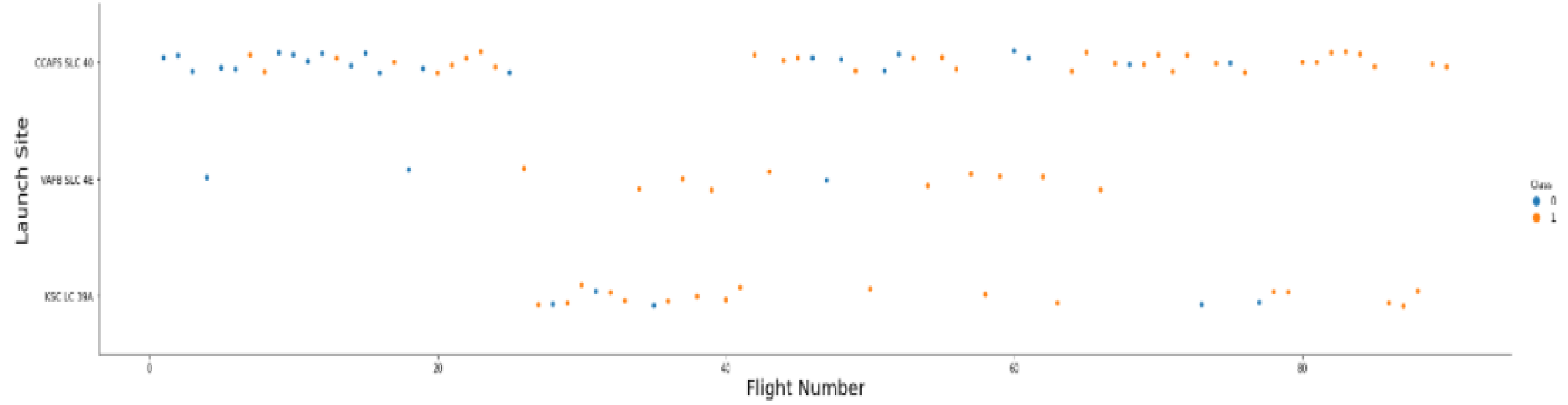
- Insights Drawn from EDA (Exploratory Data Analysis)
  - Exploratory Data Analysis – Data Visualizations
  - Exploratory Data Analysis – SQL Queries
- Launch Sites Proximities Analysis
  - Interactive Folium Maps (Screenshots)
- Build a Dashboard with Plotly Dash
  - Interactive Plotly Dash Dashboard (Screenshots)
- Predictive Analysis (Classification)
  - Predictive Analysis (Classification) – Machine Learning

Section 2

# Insights drawn from EDA

# Flight Number vs. Launch Site

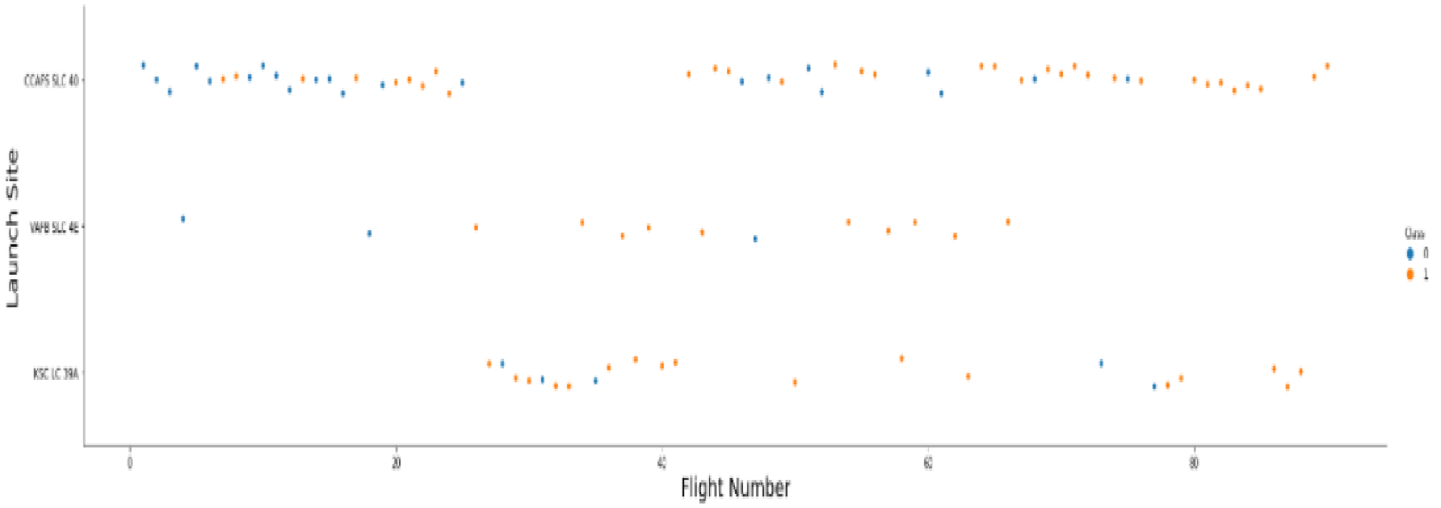
- Success rate varied noticeably with launch site.
- Successful Falcon 9 first stage landings appear to become more prevalent as the flight number increases.




- Falcon 9 first stage failed landings are indicated by the ‘0’ Class (● red markers) and successful landings by the ‘1’ Class (● green markers).

# Payload vs. Launch Site

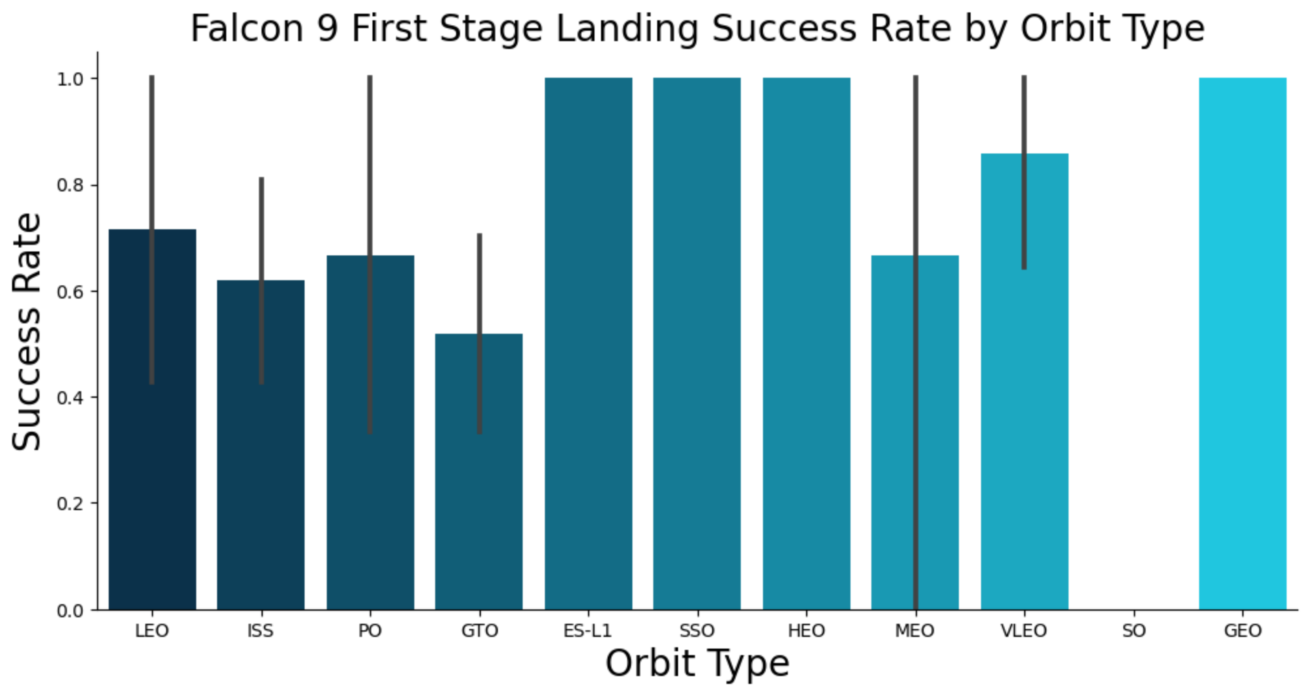
- For the CCAFS SLC 40 launch site, the payload mass and the landing outcome appear to not be strongly correlated.
- The failed landings at the KSC LC 39A launch site are mostly grouped around a narrow band of payload masses.




- Falcon 9 first stage failed landings are indicated by the ‘0’ Class (● red markers) and successful landings by the ‘1’ Class (● green markers).

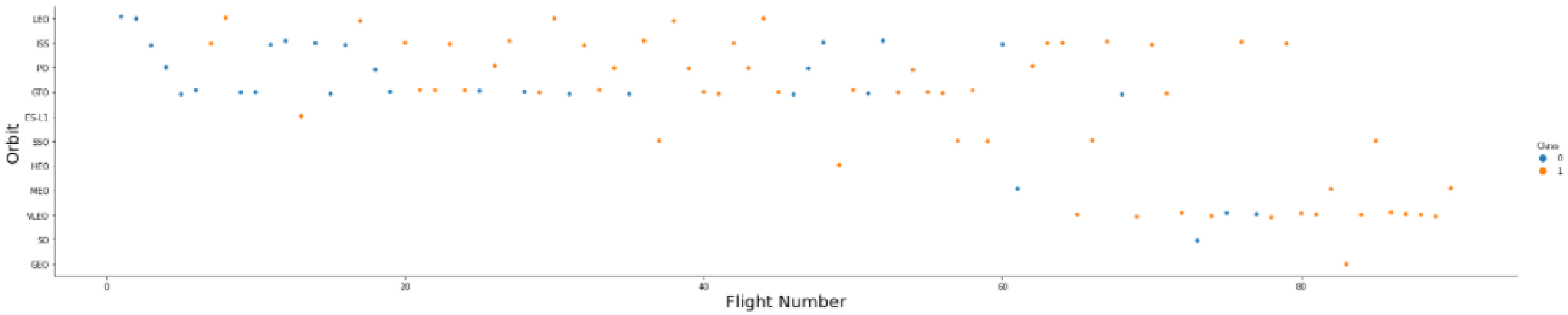
# Success Rate vs. Orbit Type

- There were no failed first-stage landings for ES-L1, SSO, HEO, and GEO orbits. Additionally, SO orbits did not have any successful first-stage landings.



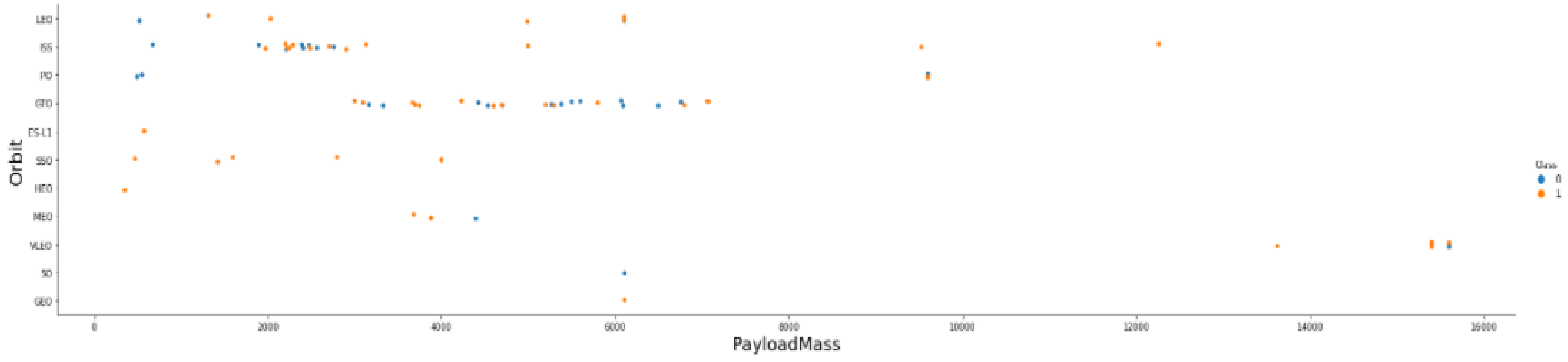
# Flight Number vs. Orbit Type

- The plot below illustrates the relationship between flight number and orbit type. It shows that for LEO orbits, success appears to be linked to the number of flights, while for GTO orbits, no clear correlation exists between flight number and orbit type.
- Additionally, there is a positive correlation between the flight number and success rate, indicating that higher flight numbers tend to be associated with better success rates.



# Payload vs. Orbit Type

- Certain orbit types exhibited higher success rates compared to others, while the success rate did not show a clear correlation with payload mass.However, when considering heavier payloads, successful landings were more frequent for PO, LEO, and ISS orbits.

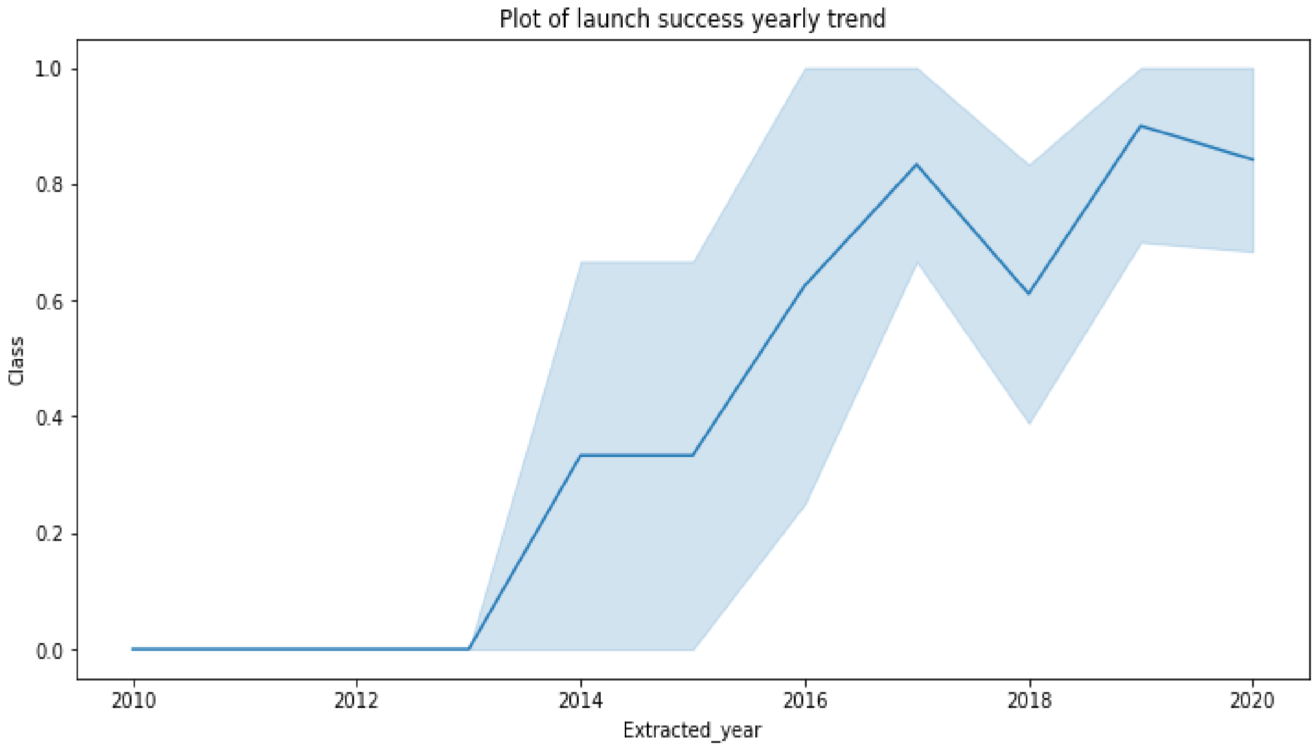





General

# Launch Success Yearly Trend

- The success rate of the Falcon 9 first stage landings has increased significantly over the selected interval of years.



# All Launch Site Names

- We used the key word **DISTINCT** to show only unique launch sites from the SpaceX data.

Display the names of the unique launch sites in the space mission

In [10]:
```
task_1 = '''
        SELECT DISTINCT LaunchSite
        FROM SpaceX
'''
create_pandas_df(task_1, database=conn)
```

Out[10]:

| | launchsite |
|---|---|
| 0 | KSC LC-39A |
| 1 | CCAFS LC-40 |
| 2 | CCAFS SLC-40 |
| 3 | VAFB SLC-4E |

# Launch Site Names Begin with 'CCA'

- **Task:** Find 5 records with launch sites that begin with `CCA`.
- **Query:** SELECT * FROM `SPACEXDATASET` WHERE `launch_site` LIKE 'CCA%' LIMIT 5;
- **Result:**

| 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 |

- **Explanation:** This is a fairly straightforward sampling mechanism used to gain a sense of the data contained in the database table.

- We used the query above to display 5 records where launch sites begin with `CCA`

# Total Payload Mass

- **Question:** What is the total payload carried by boosters from NASA?
- **Query:** `SELECT sum(`payload_mass__kg_`) AS "Total Payload Mass (kg)" FROM `SPACEXDATASET` WHERE `customer` LIKE '%NASA (CRS)%';`
- **Result:**
- **Explanation:** The total payload carried by boosters from NASA is 48,213 kg.

# Average Payload Mass by F9 v1.1

- **Question:** What is the average payload mass carried by booster version F9 v1.1?

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

In [13]:

```
task_4 = '''
        SELECT AVG(PayloadMassKG) AS Avg_PayloadMass
        FROM SpaceX
        WHERE BoosterVersion = 'F9 v1.1'
        '''
create_pandas_df(task_4, database=conn)
```

Out[13]:

| | avg_payloadmass |
|---|---|
| 0 | 2928.4 |

# First Successful Ground Landing Date

- We observed that the dates of the first successful landing outcome on ground pad was 22nd December 2015

In [14]:
```
task_5 = '''
        SELECT MIN(Date) AS FirstSuccessfull_landing_date
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Success (ground pad)'
        '''
create_pandas_df(task_5, database=conn)
```

Out[14]:

| | firstsuccessfull_landing_date |
|---|---|
| 0 | 2015-12-22 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

```
In [15]: task_6 = '''
                 SELECT BoosterVersion
                 FROM SpaceX
                 WHERE LandingOutcome = 'Success (drone ship)'
                     AND PayloadMassKG > 4000
                     AND PayloadMassKG < 6000
                 '''
         create_pandas_df(task_6, database=conn)
```

Out[15]:

| | boosterversion |
|---|---|
| 0 | F9 FT B1022 |
| 1 | F9 FT B1026 |
| 2 | F9 FT B1021.2 |
| 3 | F9 FT B1031.2 |

- We used the **WHERE** clause to filter for boosters which have successfully landed on drone ship and applied the **AND** condition to determine successful landing with payload mass greater than 4000 but less than 6000
- The four booster versions that have successfully landed on drone ship with a payload mass greater than 4,000 kg but less than 6,000 kg are listed above.

# Total Number of Successful and Failure Mission Outcomes

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

Out[16]:

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

In [17]:

```
%sql SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
```

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

Out[17]:

| COUNT(*) |
|---|
| 101 |

In [18]:

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

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

Out[18]:

| Booster_Version |
|---|
| F9 B5 B1048.4 |
| F9 B5 B1049.4 |

- We used wildcard like '%' to filter for **WHERE** MissionOutcome was a success or a failure.

# Boosters Carried Maximum Payload

- We determined the booster that have carried the maximum payload using a subquery in the **WHERE** clause and the **MAX()** function.

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [17]:
```
task_8 = '''
        SELECT BoosterVersion, PayloadMassKG
        FROM SpaceX
        WHERE PayloadMassKG = (
                                SELECT MAX(PayloadMassKG)
                                FROM SpaceX
                                )
        ORDER BY BoosterVersion
        '''
create_pandas_df(task_8, database=conn)
```

Out[17]:

| | boosterversion | payloadmasskg |
|---|---|---|
| 0 | F9 B5 B1048.4 | 15600 |
| 1 | F9 B5 B1048.5 | 15600 |
| 2 | F9 B5 B1049.4 | 15600 |
| 3 | F9 B5 B1049.5 | 15600 |
| 4 | F9 B5 B1049.7 | 15600 |
| 5 | F9 B5 B1051.3 | 15600 |
| 6 | F9 B5 B1051.4 | 15600 |
| 7 | F9 B5 B1051.6 | 15600 |
| 8 | F9 B5 B1056.4 | 15600 |
| 9 | F9 B5 B1058.3 | 15600 |
| 10 | F9 B5 B1060.2 | 15600 |
| 11 | F9 B5 B1060.3 | 15600 |

# 2015 Launch Records

- We used a combinations of the **WHERE** clause, **LIKE**, **AND**, and **BETWEEN** conditions to filter for failed landing outcomes in drone ship, their booster versions, and launch site names for year 2015

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

In [18]:
```
task_9 = '''
        SELECT BoosterVersion, LaunchSite, LandingOutcome
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Failure (drone ship)'
            AND Date BETWEEN '2015-01-01' AND '2015-12-31'
        '''
create_pandas_df(task_9, database=conn)
```

Out[18]:

| | boosterversion | launchsite | landingoutcome |
|---|---|---|---|
| 0 | F9 v1.1 B1012 | CCAFS LC-40 | Failure (drone ship) |
| 1 | F9 v1.1 B1015 | CCAFS LC-40 | Failure (drone ship) |

# 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))

In [19]:

```
task_10 = '''
        SELECT LandingOutcome, COUNT(LandingOutcome)
        FROM SpaceX
        WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY LandingOutcome
        ORDER BY COUNT(LandingOutcome) DESC
        '''
create_pandas_df(task_10, database=conn)
```

Out[19]:

| | landingoutcome | count |
|---|---|---|
| 0 | No attempt | 10 |
| 1 | Success (drone ship) | 6 |
| 2 | Failure (drone ship) | 5 |
| 3 | Success (ground pad) | 5 |
| 4 | Controlled (ocean) | 3 |
| 5 | Uncontrolled (ocean) | 2 |
| 6 | Precluded (drone ship) | 1 |
| 7 | Failure (parachute) | 1 |

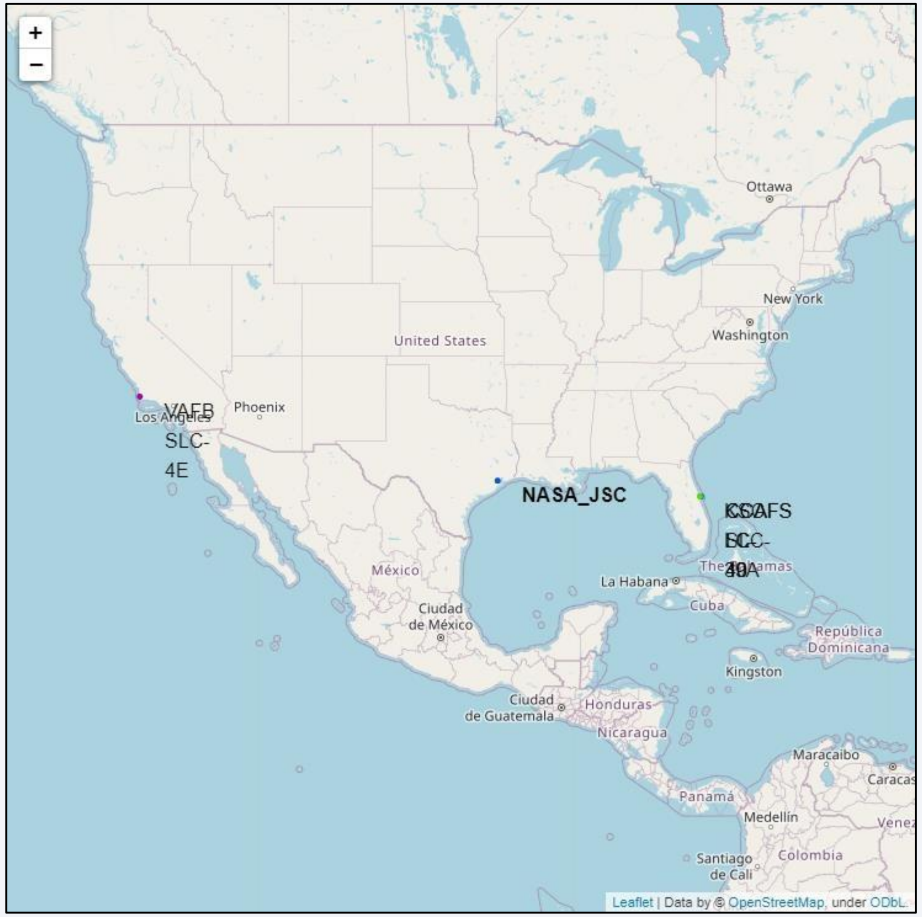
- We selected Landing outcomes and the **COUNT** of landing outcomes from the data and used the **WHERE** clause to filter for landing outcomes **BETWEEN** 2010-06-04 to 2010-03-20.
- We applied the **GROUP BY** clause to group the landing outcomes and the **ORDER BY** clause to order the grouped landing outcome in descending order.

Section 4

# Launch Sites Proximities Analysis

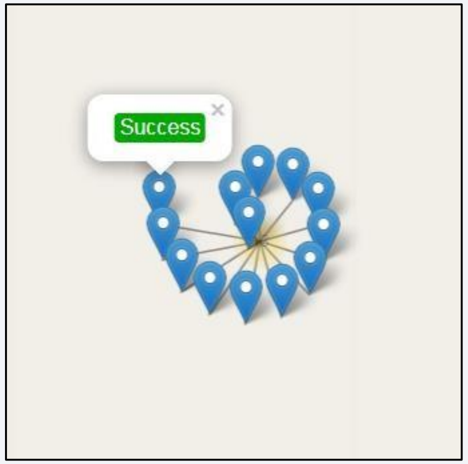
# All launch sites global map markers

- California, USA
  - VAFB SLC-4E | Vandenberg Air Force Base Space Launch Complex 4E
- Florida, USA
  - KSC LC-39A | Kennedy Space Center Launch Complex 39A
  - CCAFS LC-40 | Cape Canaveral Air Force Station Launch Complex 40
  - CCAFS SLC-40 | Cape Canaveral Air Force Station Space Launch Complex 40
  - *Note: CCAFS LC-40 and CCAFS SLC-40 in the data refer to the same launch site

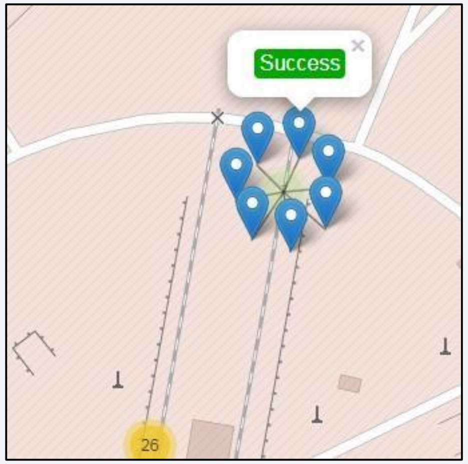


# Markers showing launch sites with color labels

- The markers display the mission outcomes (Success/Failure) for Falcon 9 first stage landings. They are grouped on the map to be associated with the geographical coordinates for the launch site.
- A sense of a launch site's success rate for Falcon 9 first stage landings can be gleaned from the relative number of green success markers to red failure markers.

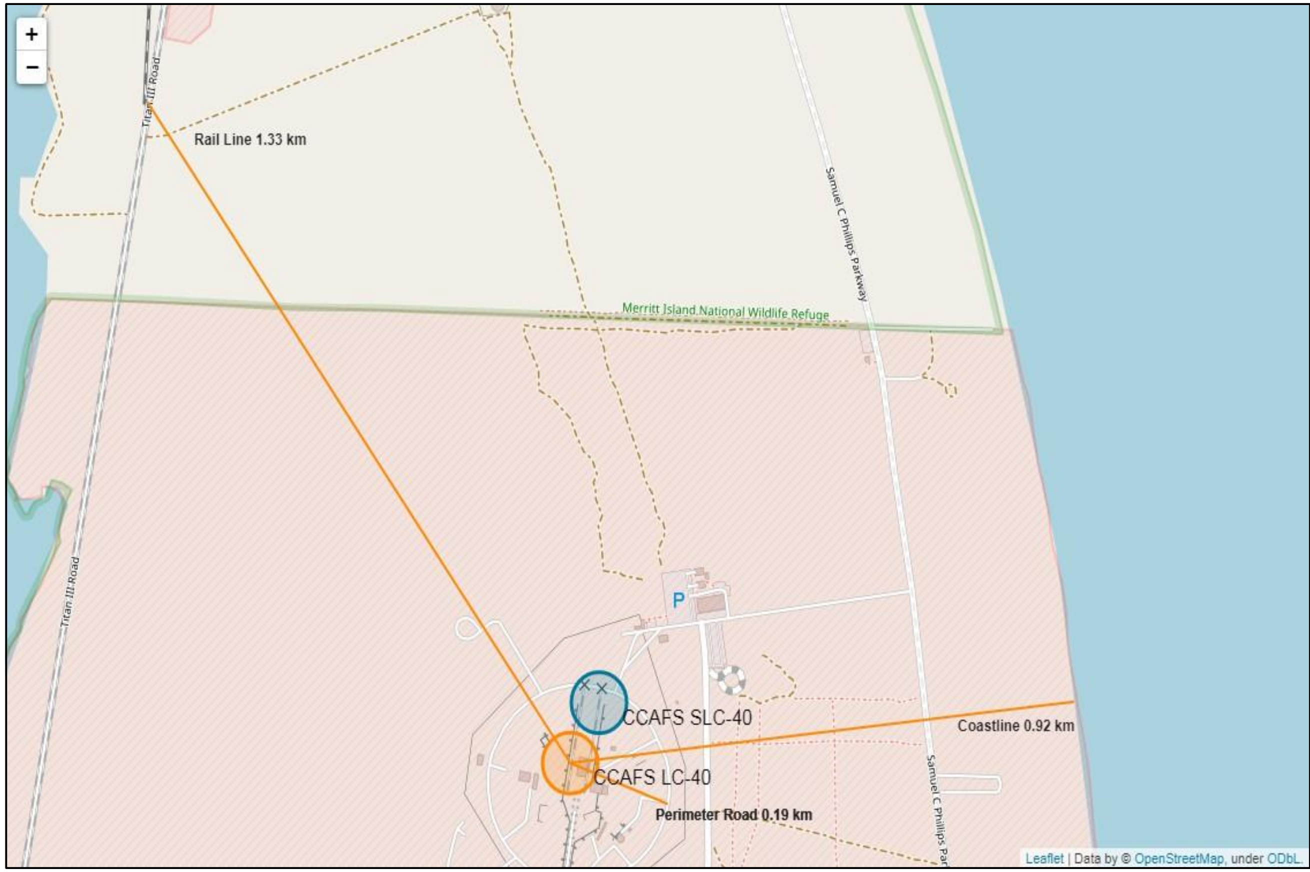







# Launch Site distance to landmarks

- The CCAFS LC-40 and CCAFS SLC-40 launch sites have coordinates that are close to being, but are not exactly, right on top of each other.
- The perimeter road around CCAFS LC-40 is 0.19 km away from the launch site coordinates.
- The coastline is 0.92 km away from CCAFS LC-40.
- The rail line is 1.33 km away from CCAFS LC-40.



Section 5

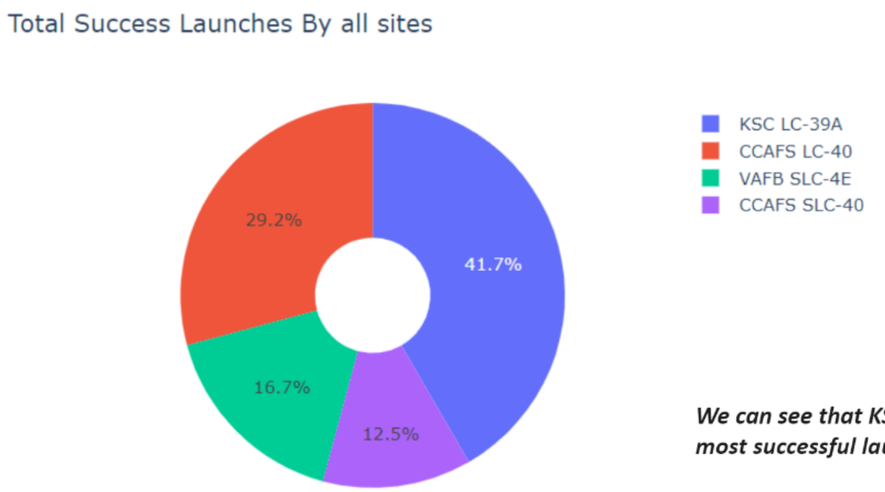
# Build a Dashboard with Plotly Dash

# Pie chart showing the success percentage achieved by each launch site

- The dropdown menu allowed the
- selection of one or all launch sites.

- With all launch sites selected, the pie chart displayed the distribution of successful Falcon 9 first stage landing outcomes between the different launch sites.

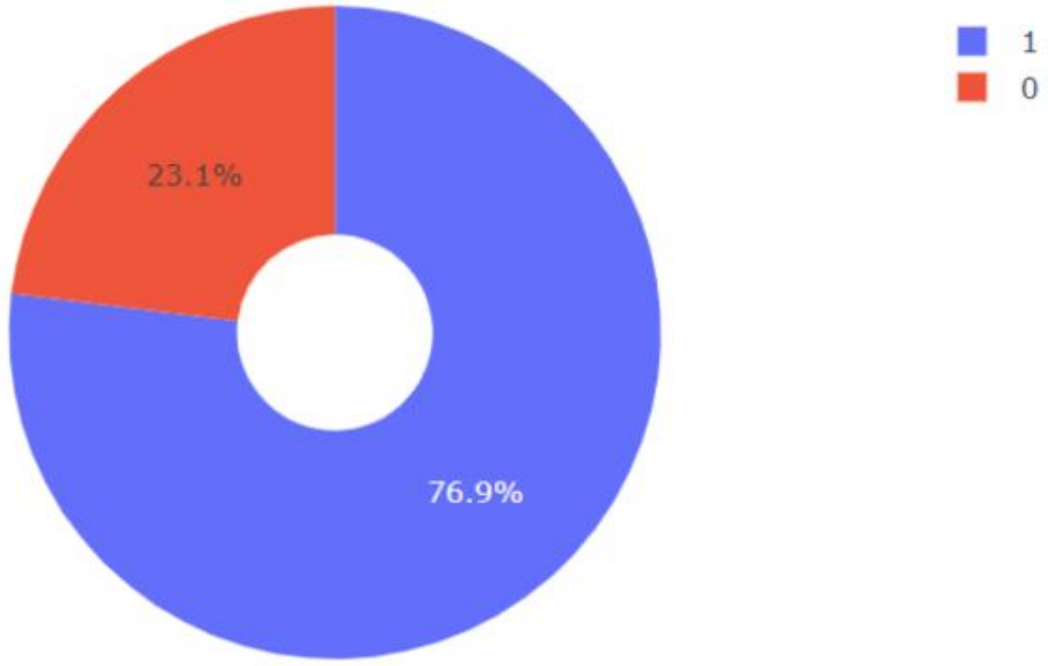
- The greatest share of successful Falcon 9 first stage landing outcomes (at 41.7% of the total) occurred at KSC LC-39A.




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

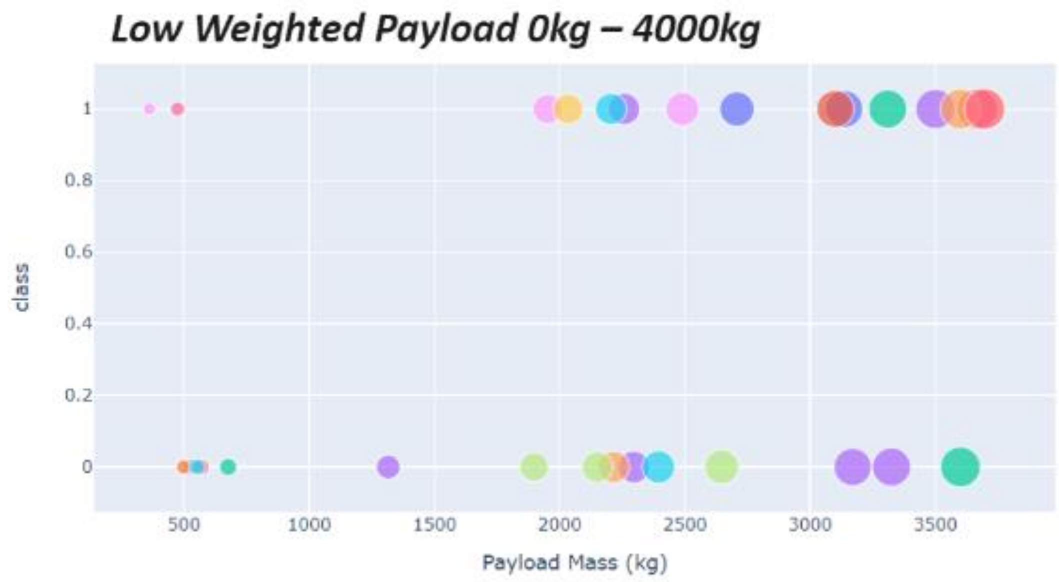
# Pie chart showing the Launch site with the highest launch success ratio

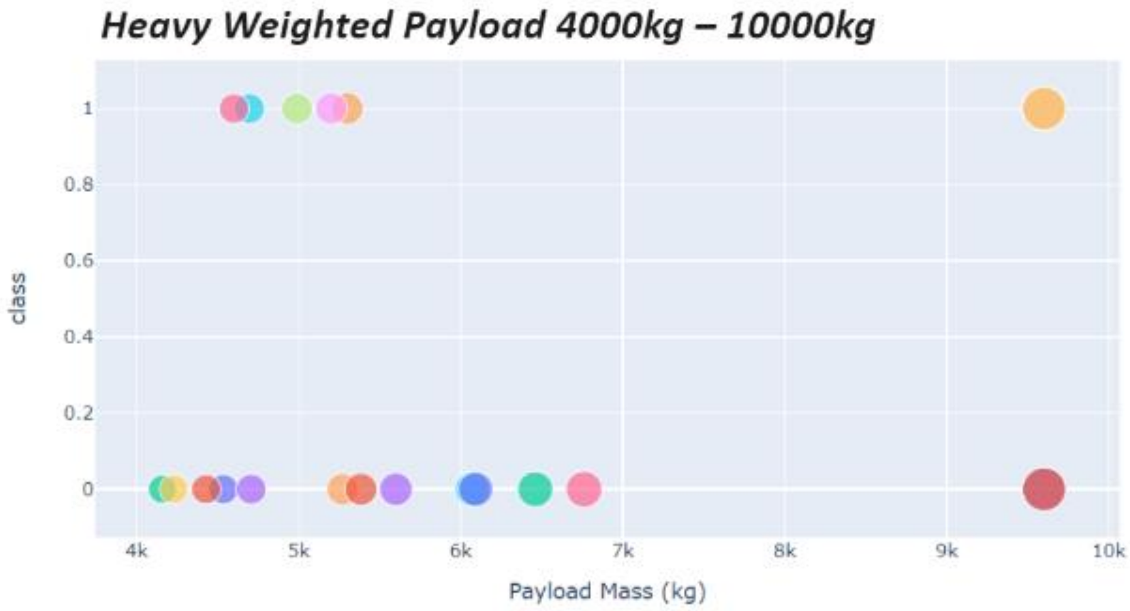
- Falcon 9 first stage failed landings are indicated by the ‘0’ Class (blue wedge in the pie chart) and successful landings by the ‘1’ Class (red wedge in the pie chart).
- CCAFS SLC-40 was the launch site that had the highest Falcon 9 first stage landing success rate (42.9%).




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

# Scatter plot of Payload vs Launch Outcome for all sites, with different payload selected in the range slider



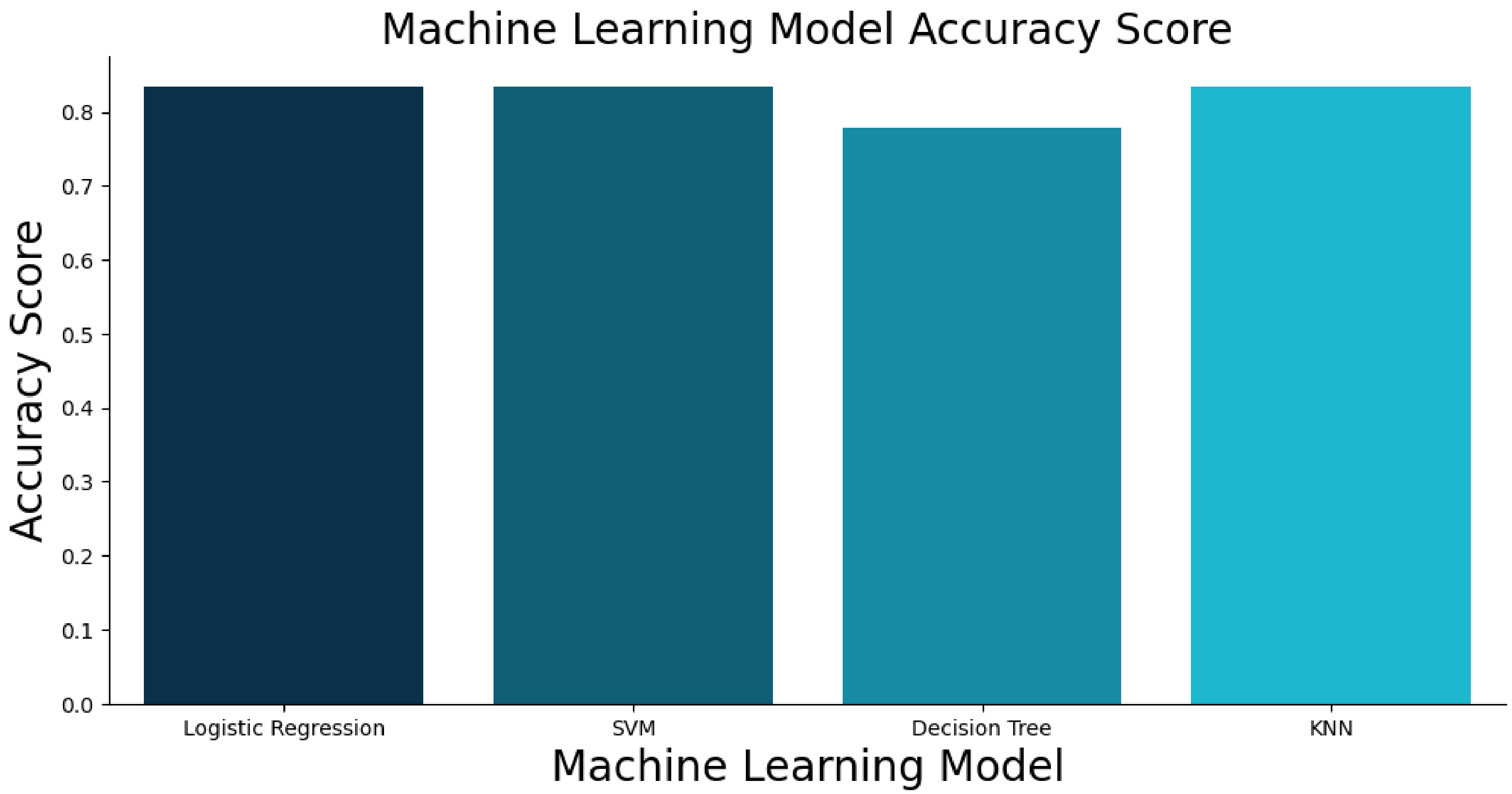

*We can see the success rates for low weighted payloads is higher than the heavy weighted payloads*

Section 6

# Predictive Analysis (Classification)

# Classification Accuracy

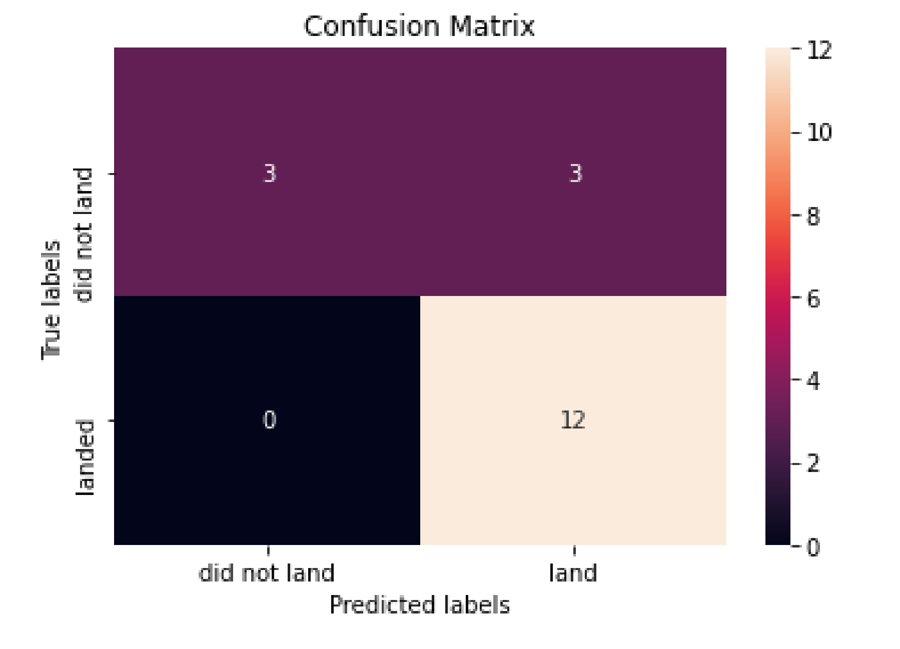
- All models performed equally well except for the Decision Tree model which performed poorly relative to the other models



# Confusion Matrix

- Shown here is the confusion matrix for
- the Logistic Regression model.
- Confusion matrices can be read as:

| | |
|---|---|
| True Negative | False Positive |
| False Negative | True Positive |




- Prediction Breakdown:
  - 12 True Positives and 3 True Negatives
  - 3 False Positives and 0 False Negatives

# Conclusions

- In conclusion, we can observe that the greater the number of flights at a launch site, the higher the success rate.
- The launch success rate steadily increased from 2013 to 2020.
- Orbits such as ES-L1, GEO, HEO, SSO, and VLEO had the highest success rates. KSC LC-39A recorded the most successful launches among all sites.
- The Decision Tree classifier emerged as the most effective machine learning algorithm for predicting landing outcomes.
- SpaceX's Falcon 9 first stage landing outcomes have shown significant improvement, reflecting an overall trend of enhanced performance and greater success with more launches.
- Machine learning models can be effectively utilized to predict future Falcon 9 first stage landing outcomes, ensuring continued progress in launch success.

# Thank you!