IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science


Ömer Ensar DURMUŞ

14.07.2023

# Outline

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

# Executive Summary

- Summary of methodologies

- Data Collection

- Data Wrangling

- Exploratory data analysis with SQL and data visualization

- Building an interactive map with Folium

- Building a Dashboard with Plotly Dash

- Predictive Analysis

- Summary of all results

- Exploratory data analysis results

- Geospatial analytics

- Interactive dashboard

- Predictive analysis result

# Introduction

- Project background and context

  SpaceX launches Falcon 9 rockets at a cost of around$62m. This is considerably cheaper than other providers(which usually cost upwards of $165m), and much of thesavings are because SpaceX can land, and then re-usethe first stage of the rocket.

- Problems you want to find answers

  If we can make predictions on whether the first stagewill land, we can determine the cost of a launch, and usethis information to assess whether or not an alternatecompany should bid and SpaceX for a rocket launch.This project will ultimately predict if the SpaceX Falcon9 first stage will land successfully.

Section 1

# Methodology

# Methodology

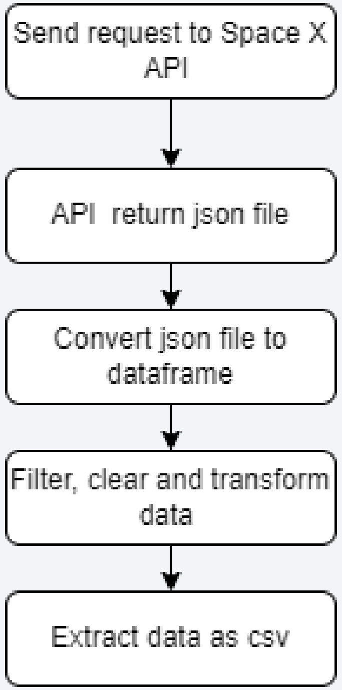
## Executive Summary

- Data collection methodology:
  - SpaceX API
  - Web scraping
- Perform data wrangling
  - Discovering, cleaning and structuring data
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Discovering, cleaning and structuring data
- Perform interactive visual analytics using Folium and Plotly Dash
  - Geospatial analytics using Folium
- Perform predictive analysis using classification models
  - Build, tune, evaluate SVM, KNN, Decision Tree and Logistic Regression classification models

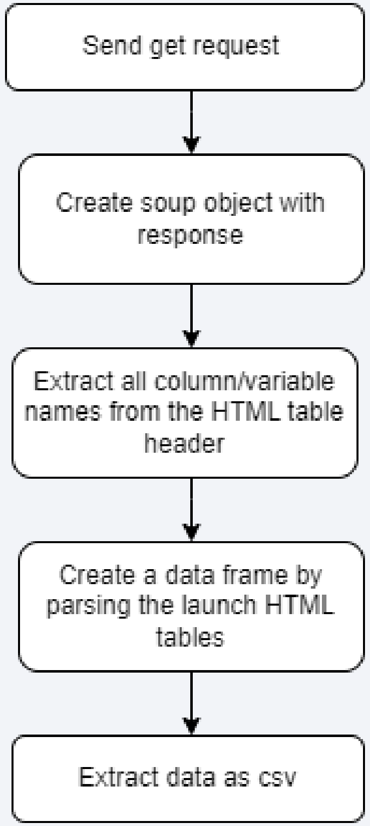
# Data Collection

- Data was obtained by sending a request via Spacex web API and web scraping from tables on the Wikipedia page

Web API



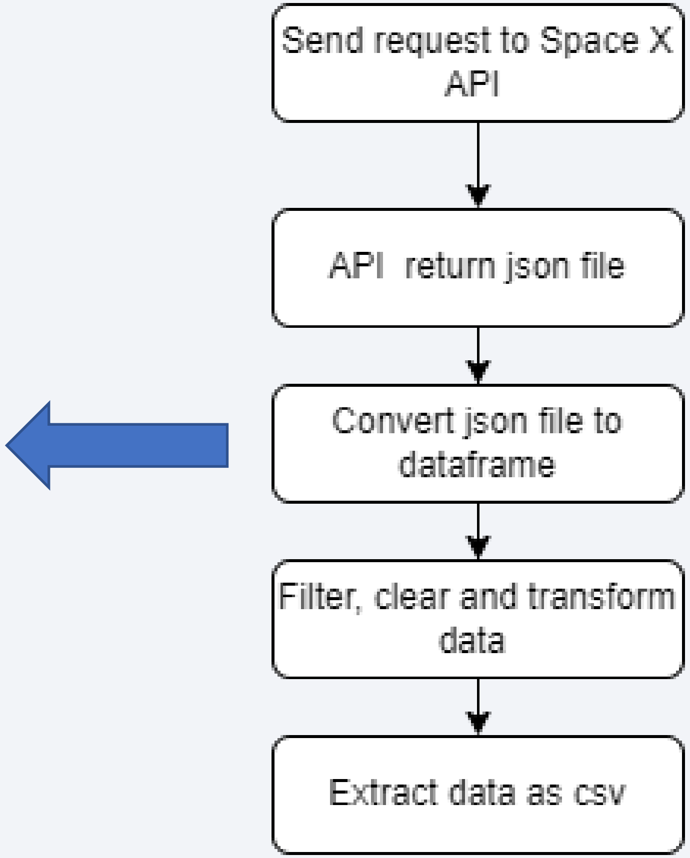

Web scrapping



# Data Collection – SpaceX API

**Output**

| | FlightNumber | Date | BoosterVersion | PayloadMass | Orbit | LaunchSite | Outcome | Flights | GridFins | Reused | Legs | LandingPad | Block | ReusedCount | Serial | Longitude | Latitude |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **4** | 1 | 2010-06-04 | Falcon 9 | NaN | LEO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0003 | -80.577366 | 28.561857 |
| **5** | 2 | 2012-05-22 | Falcon 9 | 525.0 | LEO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0005 | -80.577366 | 28.561857 |
| **6** | 3 | 2013-03-01 | Falcon 9 | 677.0 | ISS | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0007 | -80.577366 | 28.561857 |
| **7** | 4 | 2013-09-29 | Falcon 9 | 500.0 | PO | VAFB SLC 4E | False Ocean | 1 | False | False | False | None | 1.0 | 0 | B1003 | -120.610829 | 34.632093 |
| **8** | 5 | 2013-12-03 | Falcon 9 | 3170.0 | GTO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B1004 | -80.577366 | 28.561857 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **89** | 86 | 2020-09-03 | Falcon 9 | 15600.0 | VLEO | KSC LC 39A | True ASDS | 2 | True | True | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 12 | B1060 | -80.603956 | 28.608058 |
| **90** | 87 | 2020-10-06 | Falcon 9 | 15600.0 | VLEO | KSC LC 39A | True ASDS | 3 | True | True | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 13 | B1058 | -80.603956 | 28.608058 |
| **91** | 88 | 2020-10-18 | Falcon 9 | 15600.0 | VLEO | KSC LC 39A | True ASDS | 6 | True | True | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 12 | B1051 | -80.603956 | 28.608058 |
| **92** | 89 | 2020-10-24 | Falcon 9 | 15600.0 | VLEO | CCSFS SLC 40 | True ASDS | 3 | True | True | True | 5e9e3033383ecbb9e534e7cc | 5.0 | 12 | B1060 | -80.577366 | 28.561857 |
| **93** | 90 | 2020-11-05 | Falcon 9 | 3681.0 | MEO | CCSFS SLC 40 | True ASDS | 1 | True | False | True | 5e9e3032383ecb6bb234e7ca | 5.0 | 8 | B1062 | -80.577366 | 28.561857 |

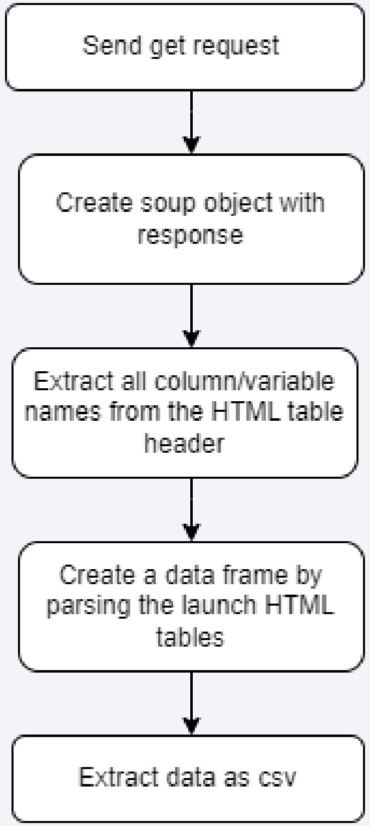



[GitHub Link]

# Data Collection - Scraping

**Output**

| | Flight No. | Launch site | Payload | Payload mass | Orbit | Customer | Launch outcome | Version Booster | Booster landing | Date | Time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | CCAFS | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success\n | F9 v1.0B0003.1 | Failure | 4 June 2010 | 18:45 |
| **1** | 2 | CCAFS | Dragon | 0 | LEO | NASA | Success | F9 v1.0B0004.1 | Failure | 8 December 2010 | 15:43 |
| **2** | 3 | CCAFS | Dragon | 525 kg | LEO | NASA | Success | F9 v1.0B0005.1 | No attempt\n | 22 May 2012 | 07:44 |
| **3** | 4 | CCAFS | SpaceX CRS-1 | 4,700 kg | LEO | NASA | Success\n | F9 v1.0B0006.1 | No attempt | 8 October 2012 | 00:35 |
| **4** | 5 | CCAFS | SpaceX CRS-2 | 4,877 kg | LEO | NASA | Success\n | F9 v1.0B0007.1 | No attempt\n | 1 March 2013 | 15:10 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **116** | 117 | CCSFS | Starlink | 15,600 kg | LEO | SpaceX | Success\n | F9 B5B1051.10 | Success | 9 May 2021 | 06:42 |
| **117** | 118 | KSC | Starlink | ~14,000 kg | LEO | SpaceX | Success\n | F9 B5B1058.8 | Success | 15 May 2021 | 22:56 |
| **118** | 119 | CCSFS | Starlink | 15,600 kg | LEO | SpaceX | Success\n | F9 B5B1063.2 | Success | 26 May 2021 | 18:59 |
| **119** | 120 | KSC | SpaceX CRS-22 | 3,328 kg | LEO | NASA | Success\n | F9 B5B1067.1 | Success | 3 June 2021 | 17:29 |
| **120** | 121 | CCSFS | SXM-8 | 7,000 kg | GTO | Sirius XM | Success\n | F9 B5 | Success | 6 June 2021 | 04:26 |




[GitHub Link]

# Data Wrangling

- First of all, exploratary data analysis was performed on the data and the unique values in the column to be estimated were examined. To make it suitable for machine learning, these values have been replaced with the number 0 and 1 representing the pass or fail state and added to the dataframe as the class column.

Import data → EDA → Define target column → Transform target column labels to number → Export data as csv

**Output**

| Outcome |
|---:|
| True ASDS |
| None None |
| True RTLS |
| False ASDS |

→

| Outcome |
|---:|
| 1 |
| 0 |
| 1 |
| 0 |

GitHub Link

# EDA with Data Visualization

1. Scatter Chart
   - Flight number vs payload mass
   - Flight number vs launch site
   - Payload vs launch site
   - Flightnumber and orbit type
   - Payload vs orbit type

Scatter charts to display the relationship between two variables and observe the nature of the relationship. The relationships observed can either be positive or negative, non-linear or linear, and/or, strong or weak.

2. Bar Chart
   - Success rate vs orbit type

Bar charts enable us to compare numerical values like integers and percentages. They use the length of each bar to represent the value of each variable.

3. Line Chart
   - Success rate vs year

Line graphs are used to track changes over different periods of time.

# EDA with SQL

- The SQL queries performed on the data set were used to:

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string 'CCA'
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display the average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome on a ground pad was achieved
6. List the names of the boosters which had success on a drone ship and a payload mass between 4000 and 6000 kg
7. List the total number of successful and failed mission outcomes
8. List the names of the booster versions which have carried the maximum payload mass
9. List the failed landing outcomes on drone ships, their booster versions, and launch site names for 2015
10. 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

GitHub Link

# Build an Interactive Map with Folium

1. Mark all launch sites on a map
    - Initialise the map using a Folium `Map` object
    - Add a `folium.Circle` and `folium.Marker` for each launch site on the launch map

2. Mark the success/failed launches for each site on a map
    - As many launches have the same coordinates, it makes sense to cluster them together.
    - Before clustering them, assign a marker colour of successful (class = 1) as green, and failed (class = 0) as red.
    - To put the launches into clusters, for each launch, add a `folium.Marker` to the `MarkerCluster()` object.
    - Create an icon as a text label, assigning the `icon_color` as the `marker_colour` determined previously.

3. Calculate the distances between a launch site to its proximities
    - To explore the proximities of launch sites, calculations of distances between points can be made using the `Lat` and `Long` values.
    - After marking a point using the `Lat` and `Long` values, create a `folium.Marker` object to show the distance.
    - To display the distance line between two points, draw a `folium.PolyLine` and add this to the map.

[GitHub Link]

# Build a Dashboard with Plotly Dash

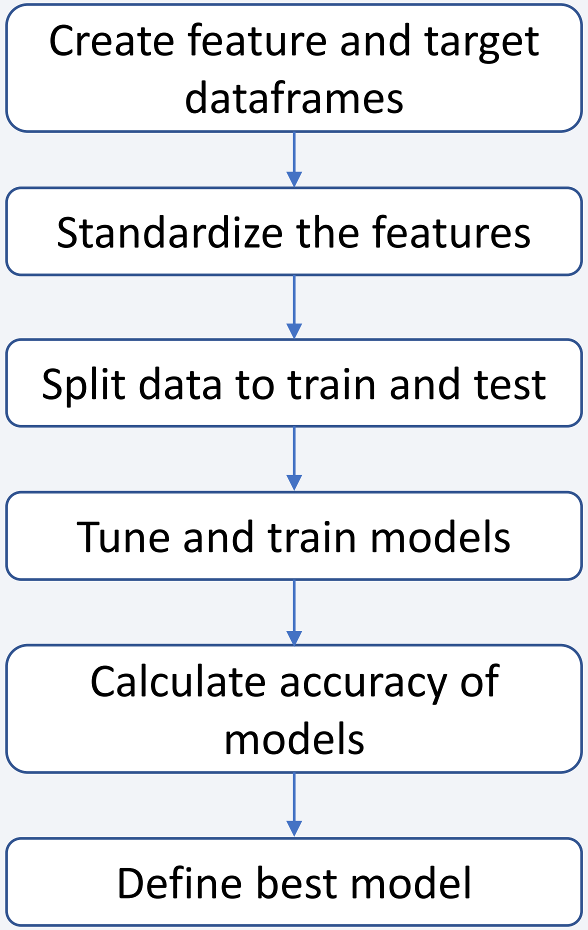
1. Pie chart showing the total successful launches per site
   - This makes it clear to see which sites are most successful
   - The chart could also be filtered to see the success/failure ratio for an individual site

2. Scatter graph to show the correlation between outcome (success or not) and payload mass (kg)
   - This could be filtered by ranges of payload masses
   - It could also be filtered by booster version

GitHub Link

# Predictive Analysis (Classification)

- To prepare the dataset for model development:
  - Load dataset
  - Perform necessary data transformations (standardise and pre-process)
  - Split data into training and test data sets
  - Decide which type of machine learning model are most appropriate
- For each chosen model:
  - Create a **GridSearchCV** object and a dictionary of parameters
  - Fit the object to the parameters
  - Use the training data set to train the model and find best parameters
- For each chosen model:
  - Review the accuracy scores for all chosen algorithms
  - The model with the highest accuracy score is determined as the best performing model



# Results

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

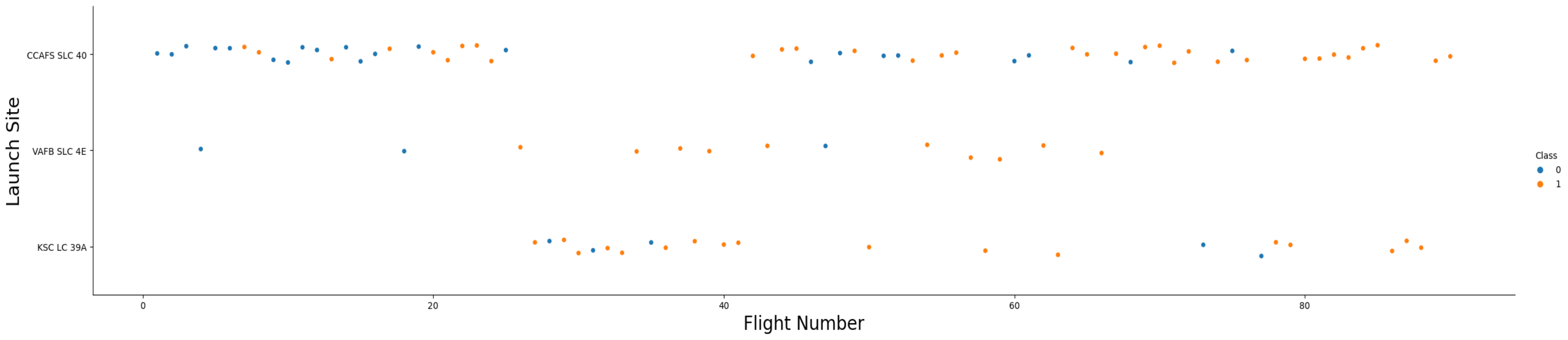
Section 2

# Insights drawn from EDA

# Flight Number vs. Launch Site

**The scatter plot of Launch Site vs. Flight Number shows that:**

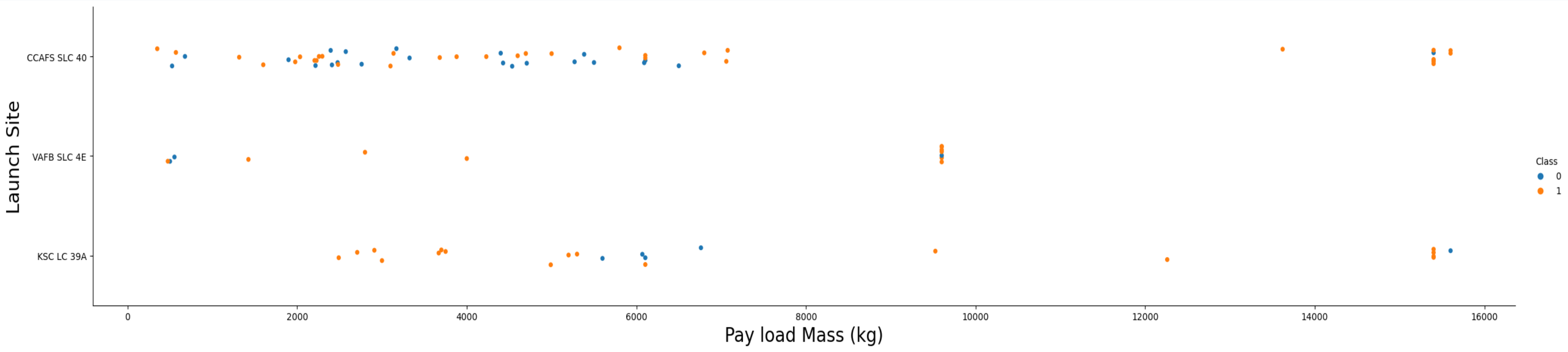
- As the number of flights increases, the rate of success at a launch site increases.
- Most of the early flights (flight numbers < 30) were launched from CCAFS SLC 40, and were generally unsuccessful.
- The flights from VAFB SLC 4E also show this trend, that earlier flights were less successful.
- Above a flight number of around 30, there are significantly more successful landings (Class = 1).



# Payload vs. Launch Site

**The scatter plot of Launch Site vs. Payload Mass shows that:**

- Above a payload mass of around 7000 kg, there are very few unsuccessful landings, but there is also far less data for these heavier launches.
- There is no clear correlation between payload mass and success rate for a given launch site.
- All sites launched a variety of payload masses, with most of the launches from CCAFS SLC 40 being comparatively lighter payloads



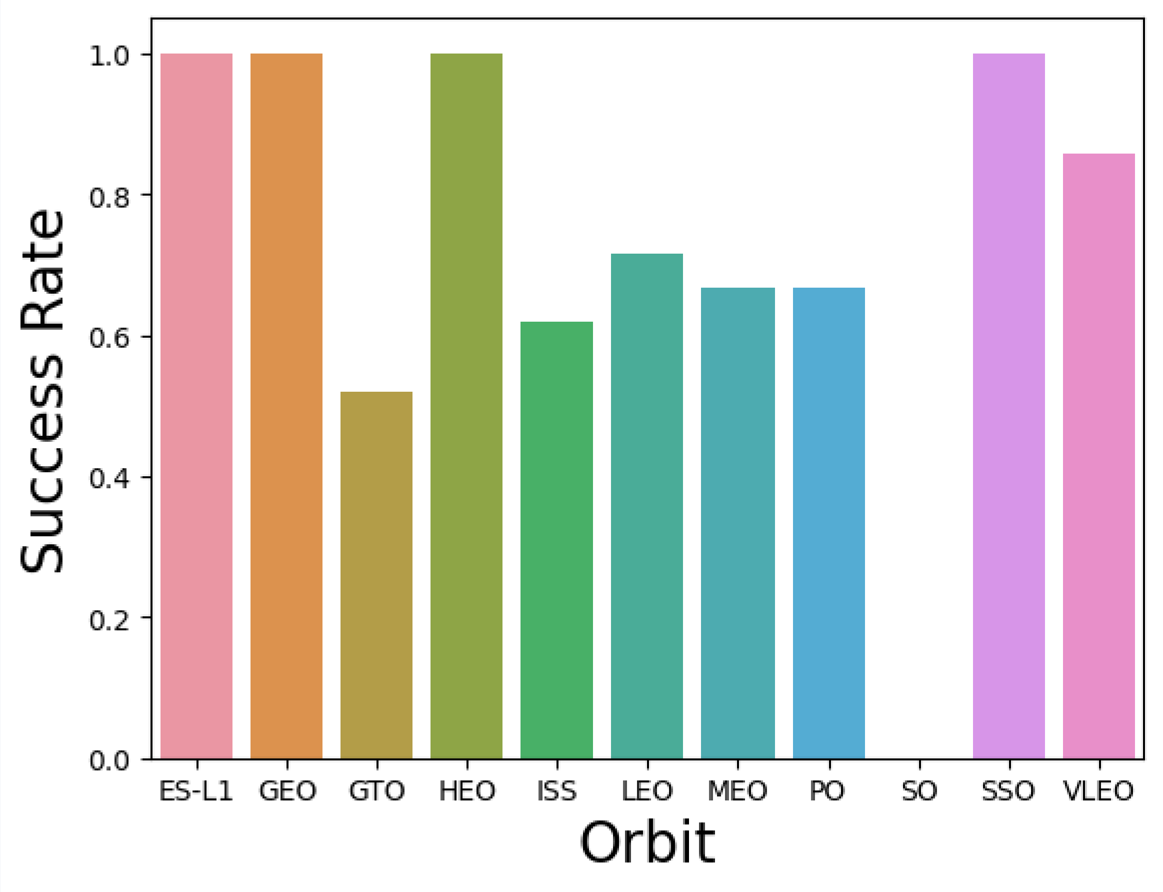
# Success Rate vs. Orbit Type

The bar chart of Success Rate vs. Orbit Type shows that the following orbits have the highest (100%) success rate:

- ES-L1 (Earth-Sun First Lagrangian Point)
- GEO (Geostationary Orbit)
- HEO (High Earth Orbit)
- SSO (Sun-synchronous Orbit)

The orbit with the lowest (0%) success rate is:

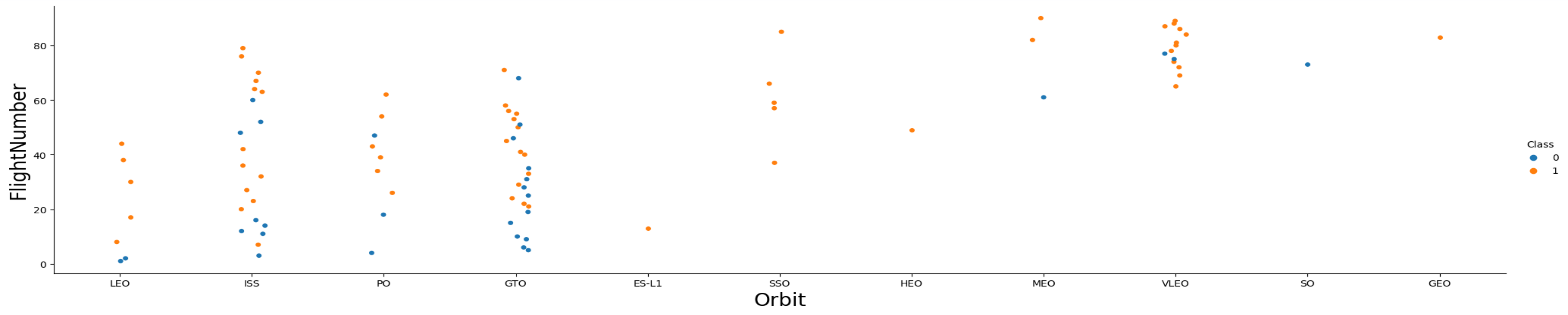
- SO (Heliocentric Orbit)



# Flight Number vs. Orbit Type

This scatter plot of Orbit Type vs. Flight number shows a few useful things that the previous plots did not, such as:

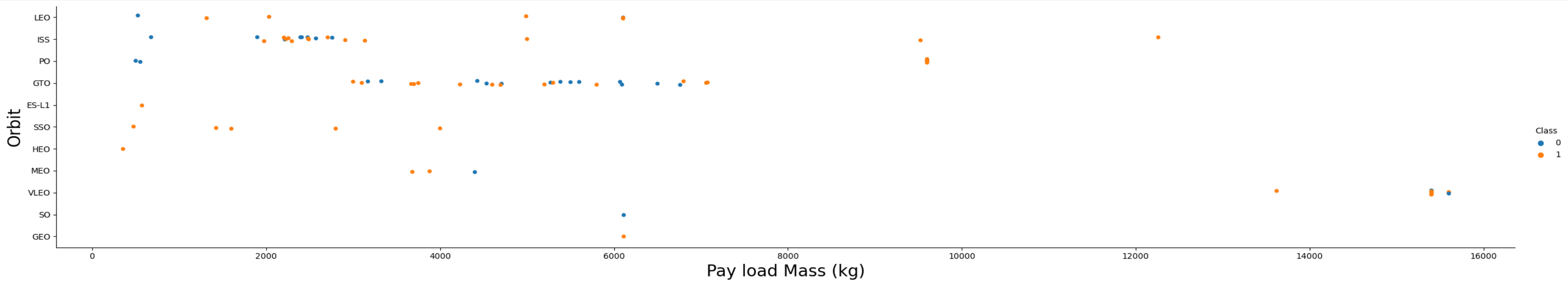
- The 100% success rate of GEO, HEO, and ES-L1 orbits can be explained by only having 1 flight into the respective orbits.
- The 100% success rate in SSO is more impressive, with 5 successful flights.
- There is little relationship between Flight Number and Success Rate for GTO.
- Generally, as Flight Number increases, the success rate increases. This is most extreme for LEO, where unsuccessful landings only occurred for the low flight numbers (early launches).



# Payload vs. Orbit Type

**This scatter plot of Orbit Type vs. Payload Mass shows that:**

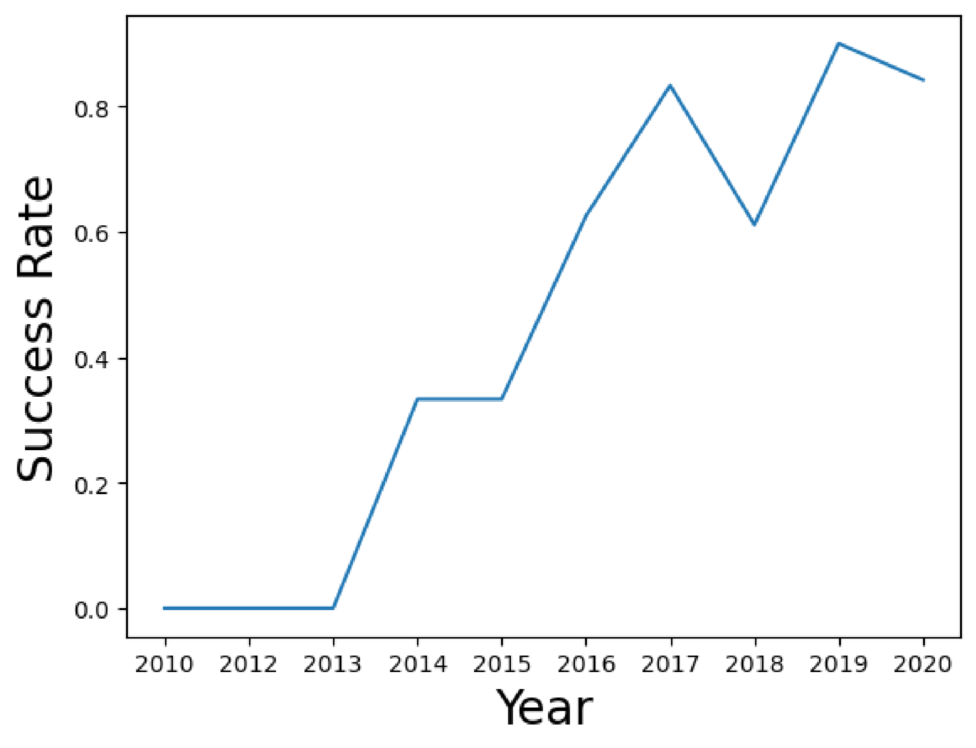
- The following orbit types have more success with heavy payloads:
  - PO
  - ISS
  - LEO
- For GTO, the relationship between payload mass and success rate is unclear.
- VLEO (Very Low Earth Orbit) launches are associated with heavier payloads, which makes intuitive sense.



# Launch Success Yearly Trend

**The line chart of yearly average success rate shows that:**

- Between 2010 and 2013, all landings were unsuccessful (as the success rate is 0).
- After 2013, the success rate generally increased, despite small dips in 2018 and 2020.
- After 2016, there was always a greater than 50% chance of success.



# All Launch Site Names

**Find the names of the unique launch sites:**

- The word **UNIQUE** returns only unique values from the **LAUNCH_SITE** column of the **SPACEXTBL** table.

```
%%sql
SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

**Unique launch sites:**

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

# Launch Site Names Begin with 'CCA'

## Find 5 records where launch sites begin with `CCA`

- **`LIMIT 5`** fetches only 5 records, and the **`LIKE`** keyword is used with the wild card **`'CCA%'`** to retrieve string values beginning 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 att |
| 10/08/2012 | 0:35:00 | F9 v1.0 B0006 | CCAFS LC-40 | SpaceX CRS-1 | 500.0 | LEO (ISS) | NASA (CRS) | Success | No att |
| 03/01/2013 | 15:10:00 | F9 v1.0 B0007 | CCAFS LC-40 | SpaceX CRS-2 | 677.0 | LEO (ISS) | NASA (CRS) | Success | No att |

# Total Payload Mass

## Calculate the total payload carried by boosters from NASA

- The **SUM** keyword is used to calculate the total of the **LAUNCH** column, and the **SUM** keyword (and the associated condition) filters the results to only boosters from NASA (CRS).

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) AS total_payload_mass
FROM SPACEXTBL
WHERE Customer LIKE 'NASA (CRS)%';
```

| Total payload mass: |
| --- |
| 48213.0 |

# Average Payload Mass by F9 v1.1

## Calculate the average payload mass carried by booster version F9 v1.1

- The **AVG** keyword is used to calculate the average of the **PAYLOAD_MASS__KG_** column, and the **WHERE** keyword (and the associated condition) filters the results to only the F9 v1.1 booster version.

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) AS avarage_payload_mass
FROM SPACEXTBL
WHERE Booster_Version LIKE 'F9 v1.1%';
```

**Average Payload Mass:**
2534.66

# First Successful Ground Landing Date

## Find the dates of the first successful landing outcome on ground pad

- The **`MIN`** keyword is used to calculate the minimum of the **`DATE`** column, i.e. the first date, and the **`WHERE`** keyword (and the associated condition) filters the results to only the successful ground pad landings.

```
%%sql
SELECT MIN(Date) AS first_succesful_date
FROM SPACEXTBL
WHERE Landing_Outcome LIKE 'Success (ground pad)';
```

| First Successful Date |
| --- |
| 01/08/2018 |

# 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

- The **WHERE** keyword is used to filter the results to include only those that satisfy both conditions in the brackets (as the **AND** keyword is also used). The **BETWEEN** keyword allows for 4000 < x < 6000 values to be selected.

```
%%sql
SELECT Booster_Version
FROM SPACEXTBL
WHERE "Landing_Outcome" = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000;
```

| **Booster Version** |
|---|
| F9 FT B1022 |
| F9 FT B1026 |
| F9 FT B1021.2 |
| F9 FT B1031.2 |

# Total Number of Successful and Failure Mission Outcomes

## Calculate the total number of successful and failure mission outcomes

- The **COUNT** keyword is used to calculate the total number of mission outcomes, and the **GROUPBY** keyword is also used to group these results by the type of mission outcome.

```
%%sql
SELECT
    CASE
        WHEN "Mission_Outcome" LIKE 'Success%' THEN 'Success'
        WHEN "Mission_Outcome" LIKE 'Failure%' THEN 'Failure'
    END AS "Mission_Outcome_Output",
    COUNT(*) AS total_outcomes
FROM SPACEXTBL
GROUP BY Mission_Outcome_Output;
```

| Mission Outcome Output | Total Outcomes |
|---|---|
| Failure | 1 |
| Success | 100 |

# Boosters Carried Maximum Payload

**List the names of the booster which have carried the maximum payload mass**

- A subquery is used here. The **SELECT** statement within the brackets finds the maximum payload, and this value is used in the **WHERE** condition. The **DISTINCT** keyword is then used to retrieve only distinct /unique booster versions.

```
%%sql
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 |

# 2015 Launch Records

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

- The **WHERE** keyword is used to filter the results for only failed landing outcomes, **AND** only for the year of 2015. **CASE** and **WHEN** keyword define the month name based on **'Date'** column.

```
%%sql
SELECT
    CASE
        WHEN substr("Date", 4, 2) = '01' THEN 'January'
        WHEN substr("Date", 4, 2) = '02' THEN 'February'
        WHEN substr("Date", 4, 2) = '03' THEN 'March'
        WHEN substr("Date", 4, 2) = '04' THEN 'April'
        WHEN substr("Date", 4, 2) = '05' THEN 'May'
        WHEN substr("Date", 4, 2) = '06' THEN 'June'
        WHEN substr("Date", 4, 2) = '07' THEN 'July'
        WHEN substr("Date", 4, 2) = '08' THEN 'August'
        WHEN substr("Date", 4, 2) = '09' THEN 'September'
        WHEN substr("Date", 4, 2) = '10' THEN 'October'
        WHEN substr("Date", 4, 2) = '11' THEN 'November'
        WHEN substr("Date", 4, 2) = '12' THEN 'December'
    END AS Month_Name,
    "Landing_Outcome",
    "Booster_Version",
    "Launch_Site"
FROM SPACEXTBL
WHERE substr("Date", 7, 4) = '2015'
    AND "Landing_Outcome" = 'Failure (drone ship)';
```

| Month Name | Landing Outcome | Booster Version | Launch Site |
|---|---|---|---|
| October | Failure (drone ship) | F9 v1.1 B1012 | CCAFS LC-40 |
| April | Failure (drone ship) | F9 v1.1 B1015 | CCAFS LC-40 |

# 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**

- The **WHERE** keyword is used with the **BETWEEN** keyword to filter the results to dates only within those specified. The results are then grouped and ordered, using the keywords **GROUP BY** and **ORDER BY**, respectively, where **DESC** is used to specify the descending order.

```
%%sql
SELECT Landing_Outcome, COUNT(*) AS outcome_count
FROM SPACEXTBL
WHERE substr("Date", 7, 4) || '-' || substr("Date", 1, 2) || '-' || substr("Date", 4, 2)
BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY outcome_count DESC;
```

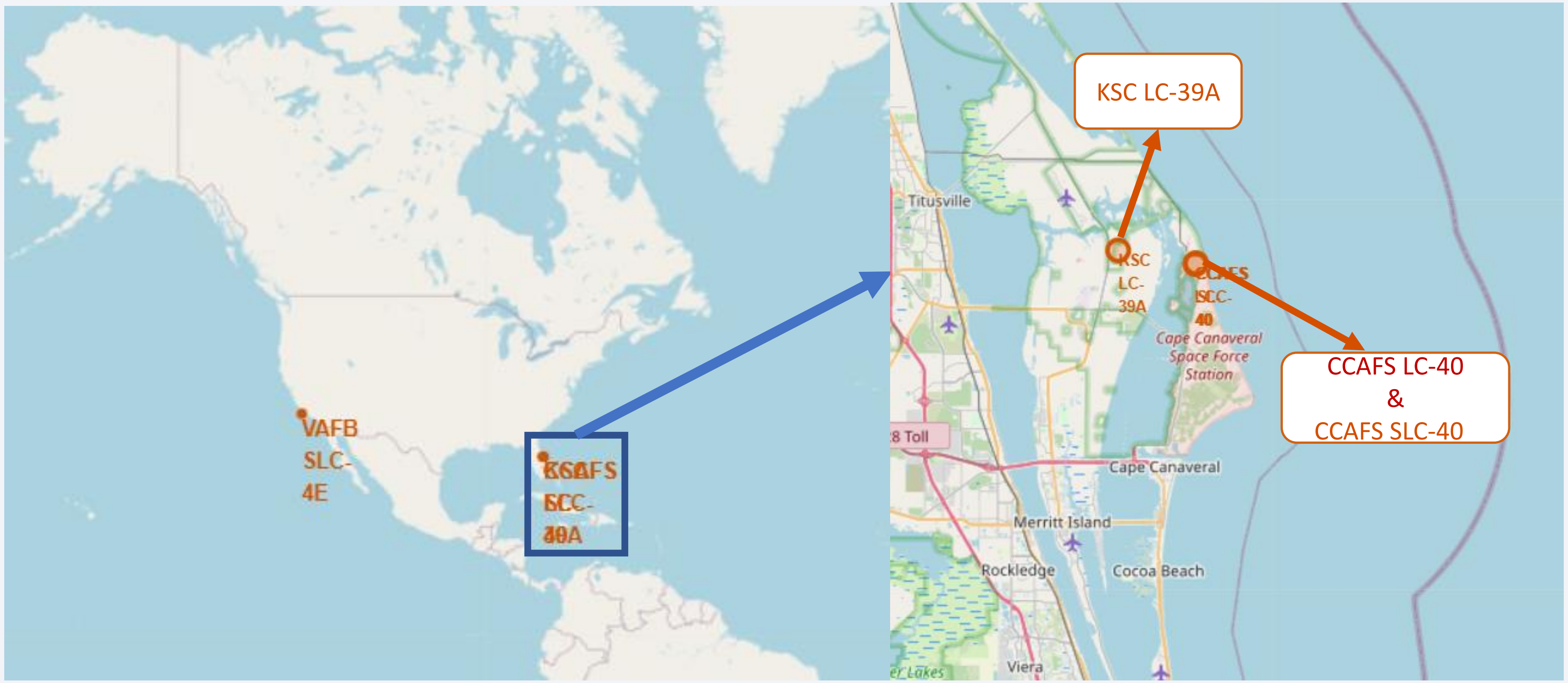
| Landing Outcome | Outcome Count |
|---|---|
| No attempt | 9 |
| Failure (drone ship) | 5 |
| Success (drone ship) | 4 |
| Controlled (ocean) | 3 |
| Uncontrolled (ocean) | 2 |
| Success (ground pad) | 2 |
| Failure (parachute) | 2 |
| Precluded (drone ship) | 1 |

Section 3

# Launch Sites Proximities Analysis

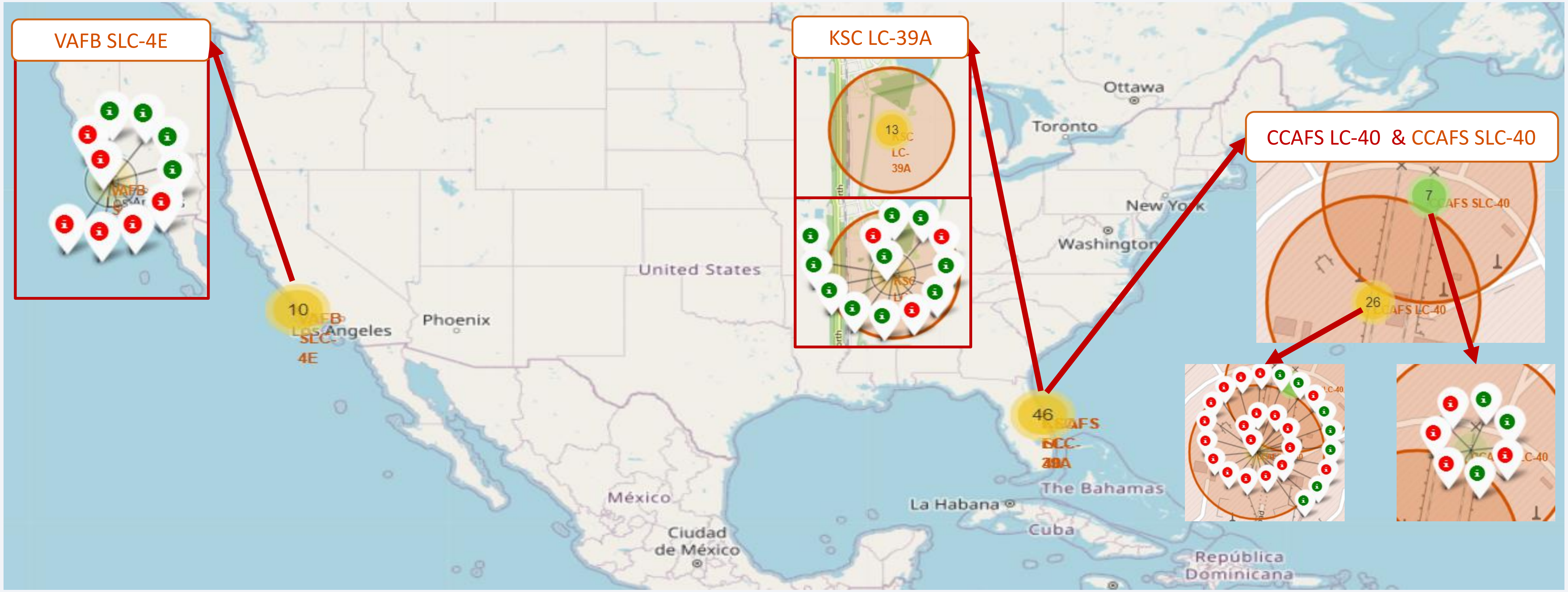
# Launch Sites Locations

- All SpaceX launch sites are on coasts of the United States of America, specifically Florida and California.



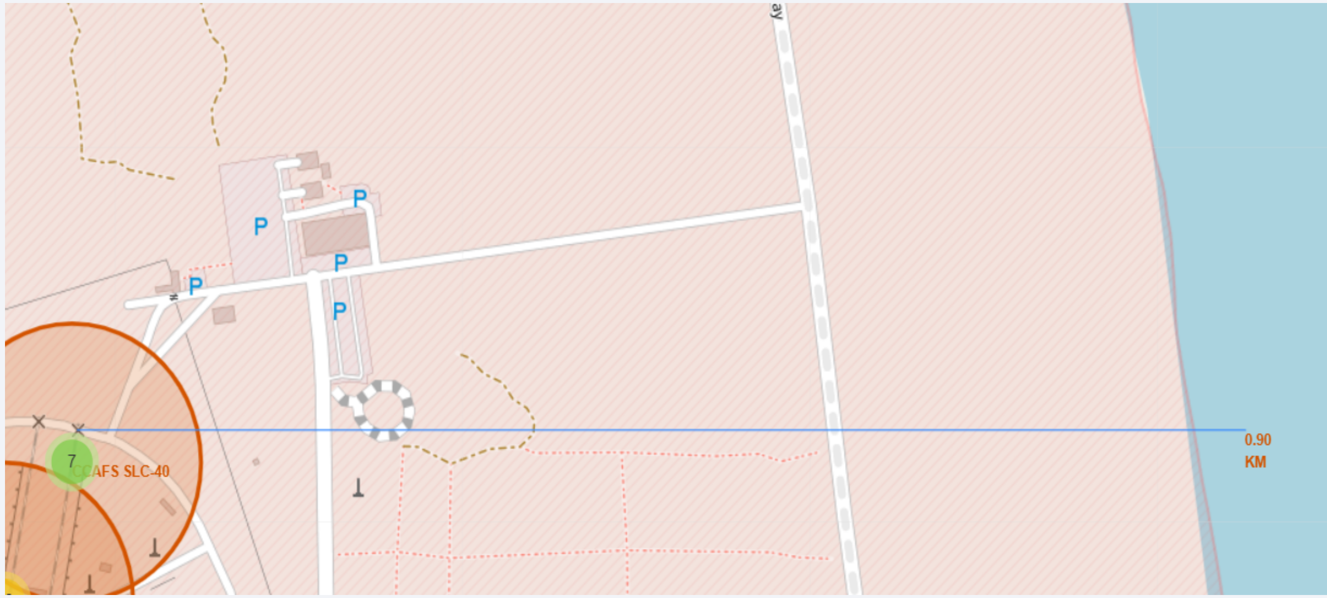
# Success/Failed Launches For Each Site

- Launches have been grouped into clusters, and annotated with green icons for successful launches, and red icons for failed launches.



# PROXIMITIES OF LAUNCH SITES TO OTHER POINTS OF INTEREST

- Using the CCAFS SLC-40 launch site as an example site, we can understand more about the placement of launch sites.





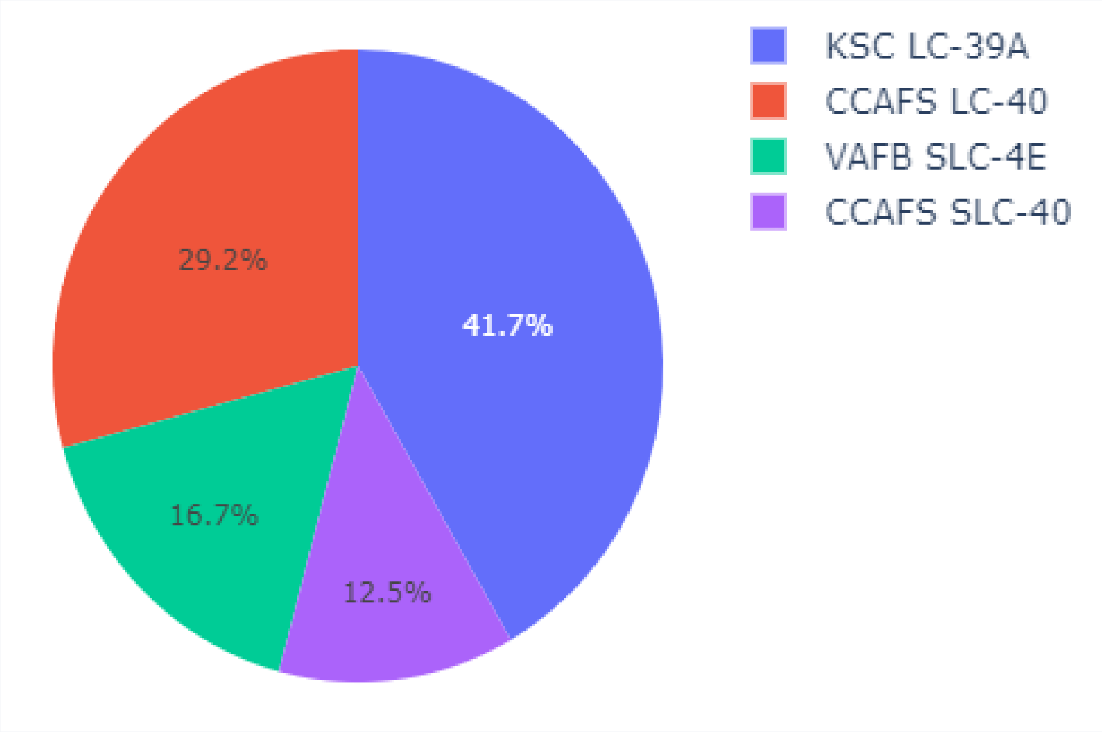
- The coastline is only 0.90 km due East.
- The nearest highway is only 0.59km away.
- The nearest railway is only 1.29 km away.
- The nearest city is 78.66 km away.

Section 4

# Build a Dashboard with Plotly Dash

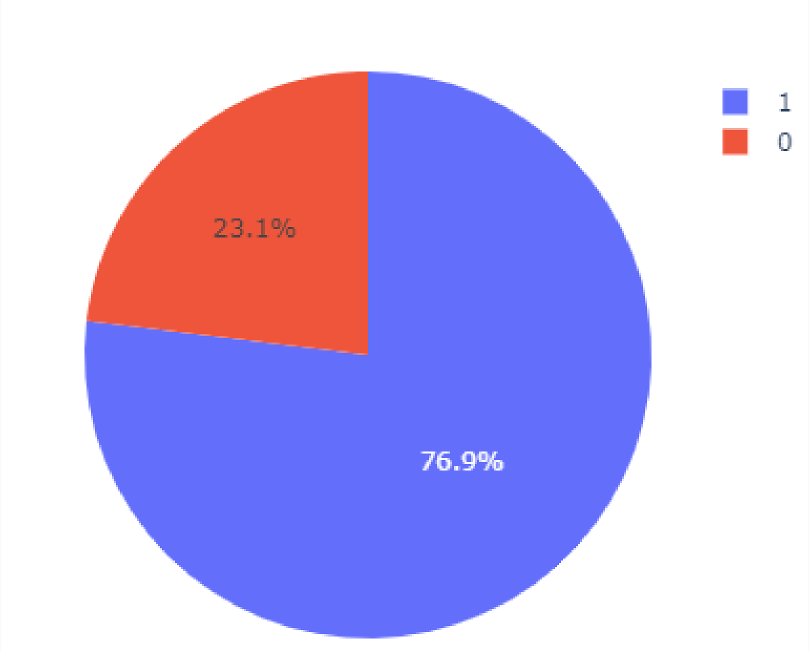
# Total Success Launches by Sites

The launch site KSC LC-39 A had the most successful launches, with 41.7% of the total successful launches



# Total Success Launches Rate for Site KSC LC-39 A

The launch site KSC LC-39 A also had the highest rate of successful launches, with a 76.9% success rate.



# Outcome vs. Payload Scatter Plot


- Plotting the launch outcome vs. payload for all sites shows a gap around 4000 kg, so it makes sense to split the data into 3 ranges:
  - 0 – 4000 kg (low payloads)
  - 4000 kg-8000 kg (mid payloads)
  - 8000 – 10000 kg (massive payloads)
- Some booster types (v1.0 and B5) do not start with massive and mid loads
- It is seen that only FT and B4 booster versions are used for massive loads.

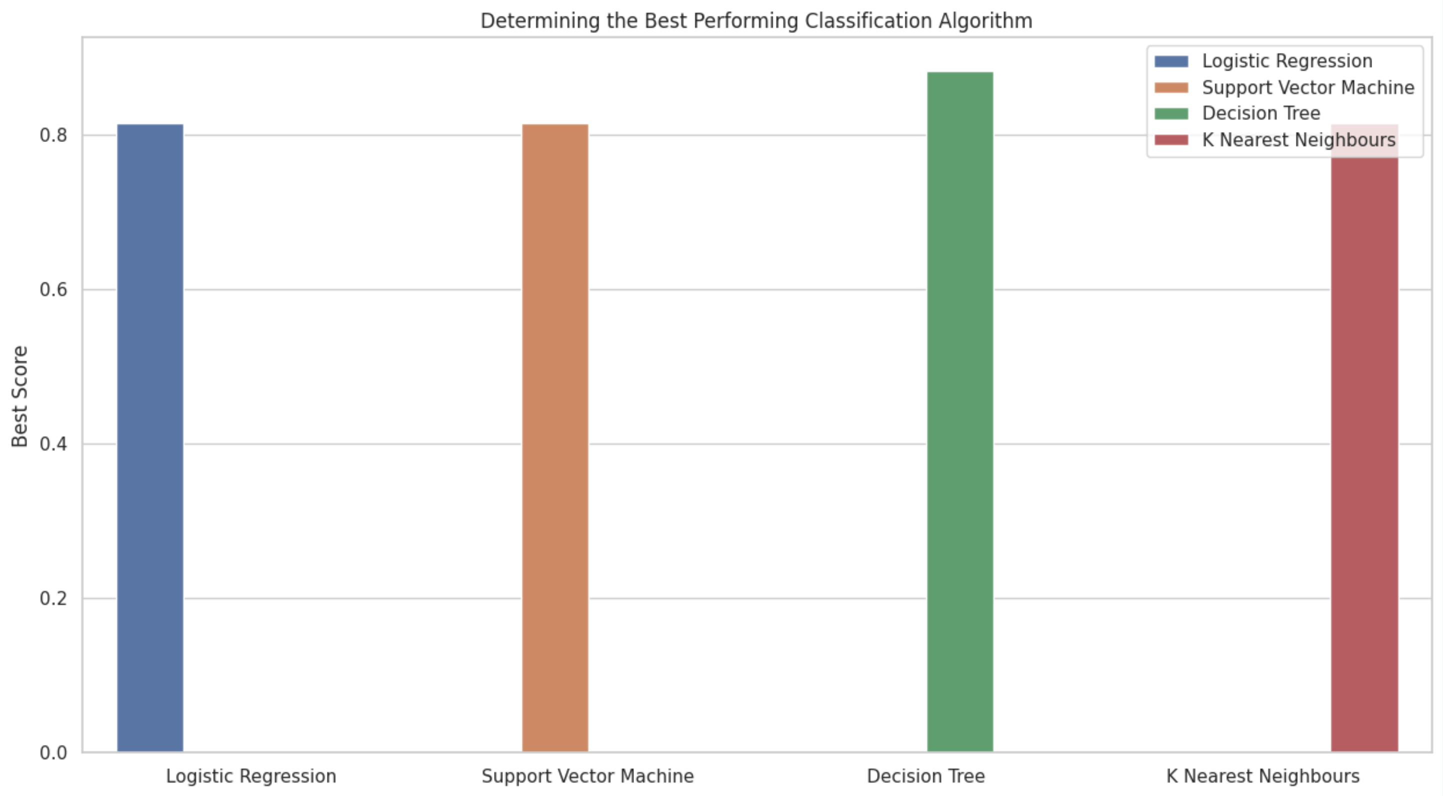
Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

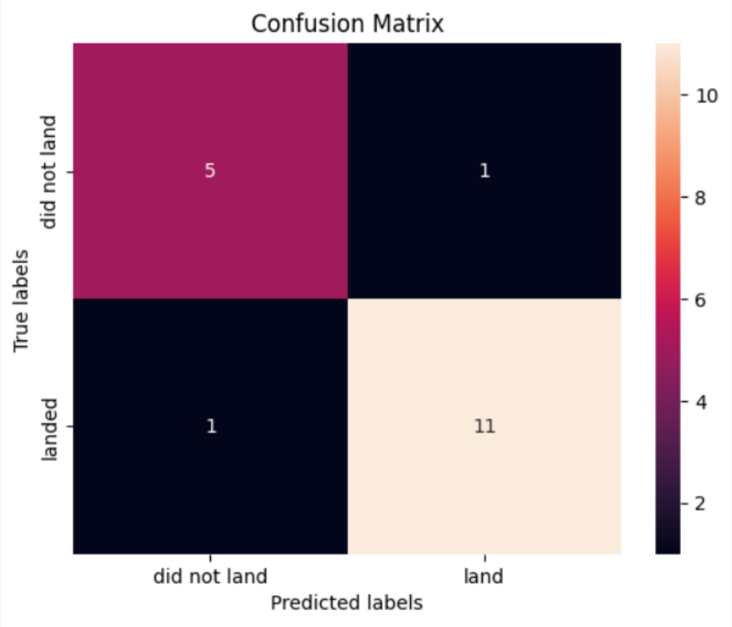
**Plotting the Accuracy Score and Best Score for each classification algorithm produces the following result:**

- The **Decision Tree** model has the highest classification accuracy



# Confusion Matrix

- As shown previously, best performing classification model is the **Decision Tree** model, with an accuracy of 88.88%.
- This is explained by the confusion matrix, which shows only 2 out of 18 total results classified incorrectly (a false positive, shown in the top-right corner and false negative bottom-right corner).
- The other 17 results are correctly classified (5 did not land, 11 did land).



# Conclusions

- As the number of flights increases, the rate of success at a launch site increases, with most early flights being unsuccessful. I.e. with more experience, the success rate increases.
  - Between 2010 and 2013, all landings were unsuccessful (as the success rate is 0).
  - After 2013, the success rate generally increased, despite small dips in 2018 and 2020.
  - After 2016, there was always a greater than 50% chance of success.
- Orbit types ES-L1, GEO, HEO, and SSO, have the highest (100%) success rate.
  - The 100% success rate of GEO, HEO, and ES-L1 orbits can be explained by only having 1 flight into the respective orbits.
  - The 100% success rate in SSO is more impressive, with 5 successful flights.
  - The orbit types PO, ISS, and LEO, have more success with heavy payloads:
  - VLEO (Very Low Earth Orbit) launches are associated with heavier payloads, which makes intuitive sense.
- The launch site **KSC LC-39 A** had the most successful launches, with 41.7% of the total successful launches, and also the highest rate of successful launches, with a 76.9% success rate.
- The best performing classification model is the Decision Tree model, with an accuracy of 94.44%.

# Appendix

Appendix can be found at the link below.

- https://github.com/omerensar13/IBM_Data_Science_Exercise.git

# Thank you!