IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Adebisi Ruth Adegoke
22nd January 2022

# Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

# Executive Summary

- Summary of methodologies
  - Data Collection API and Web scraping
  - Data Wrangling
  - EDA with SQL
  - EDA with data visualization
  - Building an interactive dashboard with Plotly Dash
  - Predictive Analysis

- Summary of all results
  - EDA results
  - Interactive analytics
  - Predictive Analysis

# Introduction

- Project background and context
  - Space X Falcon 9 rocket launches cost 62 million dollars; other providers cost upwards of 165 million dollars. This is because Space X can reuse the first stage. If we can predict if the first stage will land, we can determine the cost of each launch. This information can be used if another company wants to bid against space X for a rocket launch. Launch success rate may depend on orbit type, payload mass, location and proximities of the launch site.

- Problems you want to find answers
  - Determine the cost of each launch.
  - Predict if the first stage of the Falcon 9 rockets will be recovered and reused.
  - Determine the factors that influence the successful landing of launched Falcon 9 rocket.

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection methodology:
  - Data was gathered from the Space X launch API
  - Web scrap Falcon 9 historical launch records from Wikipedia
- Perform data wrangling
  - Handling null values and removing irrelevant columns
  - Create dummy variables using one-hot encoding to categorical columns
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Linear Regression, KNN, SVM and Decision Tree Classifier models were used
  - Find the best hypermeter for each classification model, find the method with the highest accuracy using the test data

# Data Collection

- Falcon 9 launch data was sourced from SpaceX REST API and web-scrapped Wiki pages
- The Space X API endpoint is a URL that begins with api.spacexdata.com/v4/
- This API will give us information about launches including information about rocket used, payload delivered, launch specifications and landing outcome
- Initialize a GET request targeting each specific endpoint from the API
- A response is returned in the JSON format, we simplify it into a data frame json_normalize function
- Web scrap HTML tables on Wikipedia using the Python BeautifulSoup package, parse the data from the tables and convert it to a Pandas data frame
- We used helper functions to target another endpoint of the API to gather more data for each rocket using their ID number

# Data Collection – SpaceX API

- User-defined functions to extract information using the ID numbers in the launch date
- Request and parse Space X launch data using GET request from https://api.spacexdata.com/v4/launches/past
- Returns Space X data in JSON
- Flatten the JSON into a data frame using json_normalize function
- Apply custom functions to the data
- Assign list to a dictionary and filter data frame to only Falcon 9 launch records
- Export to csv using .to_csv function



- https://github.com/bisiAlgoRuthM/SpaceX/blob/main/data-collection-api.ipynb

# Data Collection – SpaceX API

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

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

```
# Call getBoosterVersion
getBoosterVersion(data)
```

✓ 1m 15.1s

```
# Call getLaunchSite
getLaunchSite(data)
```

```
# Call getPayloadData
getPayloadData(data)
```

✓ 1m 16.4s

```
# Call getCoreData
getCoreData(data)
```

✓ 1m 15.5s

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

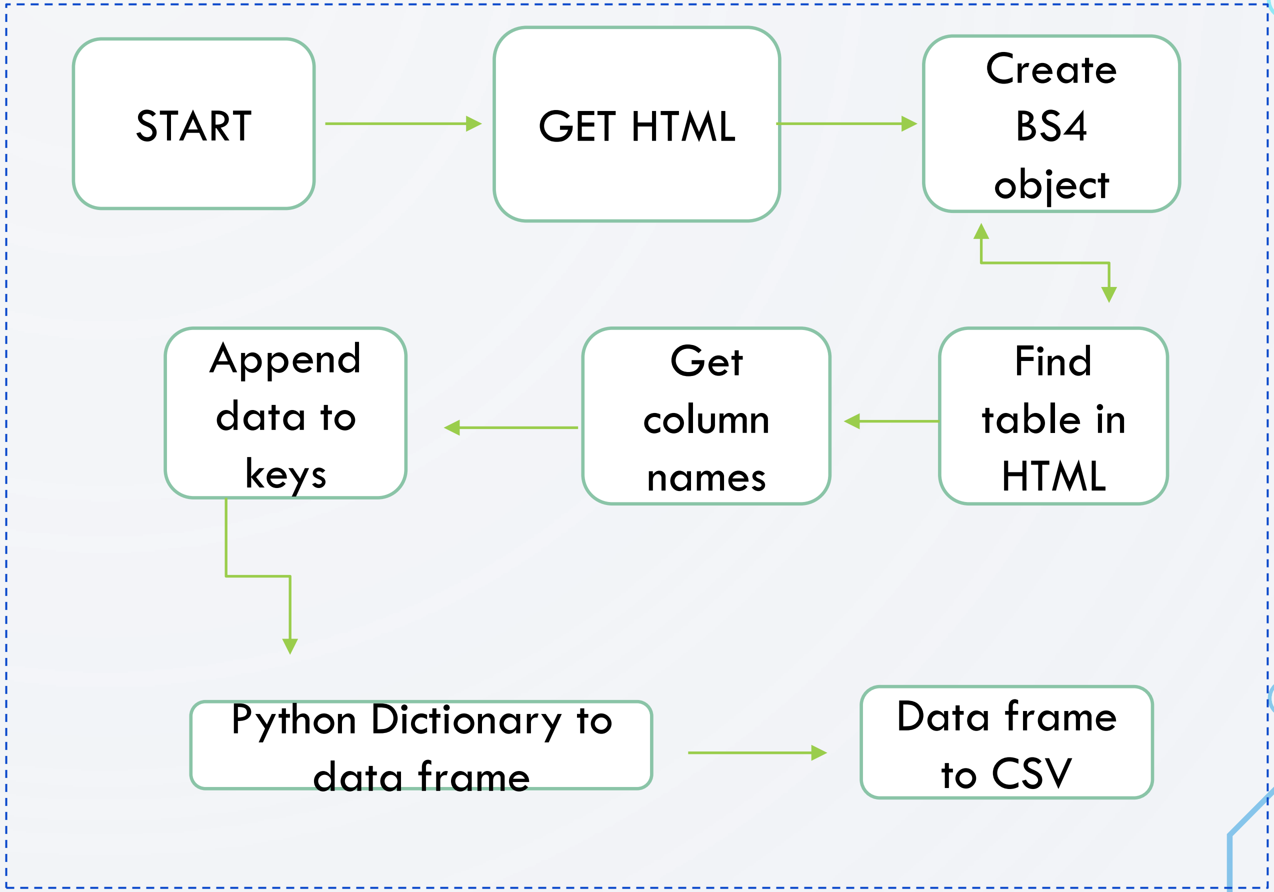
0.2s

```
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = launch_data[launch_data['BoosterVersion'] != 'Falcon 1']
```

https://github.com/bisiAlgoRuthM/SpaceX/blob/main/data-collection-api.ipynb

# Data Collection - Scraping

- Web Scrapping from Wikipedia
  - Get response from HTML from Website
  - Create a BeautifulSoup Object
  - Find table in HTML
  - Get column names and set as keys
  - Append data to keys
  - Convert dictionary to data frame and data frame to .CSV




https://github.com/bisiAlgoRuthM/SpaceX/blob/main/webscraping.ipynb

# Data Collection - Scraping

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

html_text = response.content
```

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

```
# Assign the result to a list called
html_tables = soup.find_all('table')
```

```
launch_dict= dict.fromkeys(column_names)
```

```
# Let's initial the launch_dict with ea
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

```
headings = []
for key,values in dict(launch_dict).items():
    if key not in headings:
        headings.append(key)
    if values is None:
        del launch_dict[key]

def pad_dict_list(dict_list, padel):
    lmax = 0
    for lname in dict_list.keys():
        lmax = max(lmax, len(dict_list[lname]))
    for lname in dict_list.keys():
        ll = len(dict_list[lname])
        if  ll < lmax:
            dict_list[lname] += [padel] * (lmax - ll)
    return dict_list

pad_dict_list(launch_dict,0)

df = pd.DataFrame(launch_dict)
df.head()
```

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

spacex_web_scraped.csv

```
Flight No.,Launch site,Payload,Payload mass,Orbit,Customer,Launch outcome,Version Booster,
1,CCAFS,Dragon Spacecraft Qualification Unit,0,LEO,SpaceX,"Success
",F9 v1.0B0003.1,Failure,4 June 2010,18:45
```

# Data Wrangling

1. Filter data to only include Falcon 9 launches
2. Check for missing values
3. Calculate the number of launches on each site
4. Calculate the number and occurrence of each orbit
5. Calculate the number and occurrence of mission per orbit type
6. Create a landing label outcome for outcome column
7. Handle missing values

https://github.com/bisiAlgoRuthM/SpaceX/blob/main/spacex-Data%20wrangling.ipynb

# EDA with Data Visualization

We plotted the following types of chart :

**Cat plot** *showing the relationship between* :

- The flight Number( indicating the number of continuous launch attempts) and Payload Mass variable
- Flight Number and Launch Site

**Scatter plot** *showing the relationship between* :

- Payload Mass and Launch site
- Flight number and Orbit type
- Payload mass and Orbit type

**Bar graph** *showing:*

- Success rate of each Orbit Type

**Line plot** *showing:*

- Yearly Success rate of Falcon 9 launches



- https://github.com/bisiAlgoRuthM/SpaceX/blob/main/jupyter-labs-eda-dataviz.ipynb

# EDA with SQL

- Displaying the names of unique launch sites on the space mission
- Displaying 5 records where the launch sites begin with “CCA”
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.
- Listed the date when the first successful landing outcome on the ground pad was achieved.
- Displaying the names of the boosters which have success in drone ships and have payload mass greater than 4000kg but less than 6000kg
- Showed the total number of successful and failed mission outcomes
- Displayed names of the booster versions which have carried the maximum payload mass.
- Listing the records which display the month names, failure landing outcomes in drone ship , booster versions, launch site for the months in year 2015
- Ranked the count of successful landing outcomes between the date 04-06-2010 and 20-03-2017 in descending order



https://github.com/bisiAlgoRuthM/SpaceX/blob/main/eda-sql-sqllite.ipynb

# Build an Interactive Map with Folium

The launch success rate may depend on the proximities of a launch site, i.e., the initial position of rocket trajectories. In essence, choosing an optimal location to build a launch site can invariably impact the launch outcome

- Generate a folium map object using NASA coordinates from the dataset
- On the map, add circle and marker objects for each launch site on the site map
- Observe geographical proximities to the coast lines and the equator
- Mark the success/failed launches of each site on the map to see which launch sites have a high success rate
- Calculate the distance of a launch site to its proximities such as railways, highways, coastline and cities



https://github.com/bisiAlgoRuthM/SpaceX/blob/main/launch_site_location.ipynb

# Build a Dashboard with Plotly Dash

- To visualize we created the following dash components and plots:
  - A dropdown object to select a launch site
  - Pie chart showing the launch success rate of all sites
  - Pie chart showing the launch outcome of each site
  - A slider to select a range of payload mass
  - Scatter plot showing the Payload vs the launch outcome

https://github.com/bisiAlgoRuthM/SpaceX/blob/main/spacex_dash_app.py

# Predictive Analysis (Classification)

- Import required libraries and define a function to plot the confusion matrix
- Load data into data frame and standardize and transform the data
- Split the data int testing and training sets
- Set parameters and create classification model
- Create a GridSearchCV object and fit with the model and parameters
- Fit the object with training data
- Find the best parameters the accuracy score and plot confusion matrix
- Repeat process for all classification models
- Compare accuracy of all models

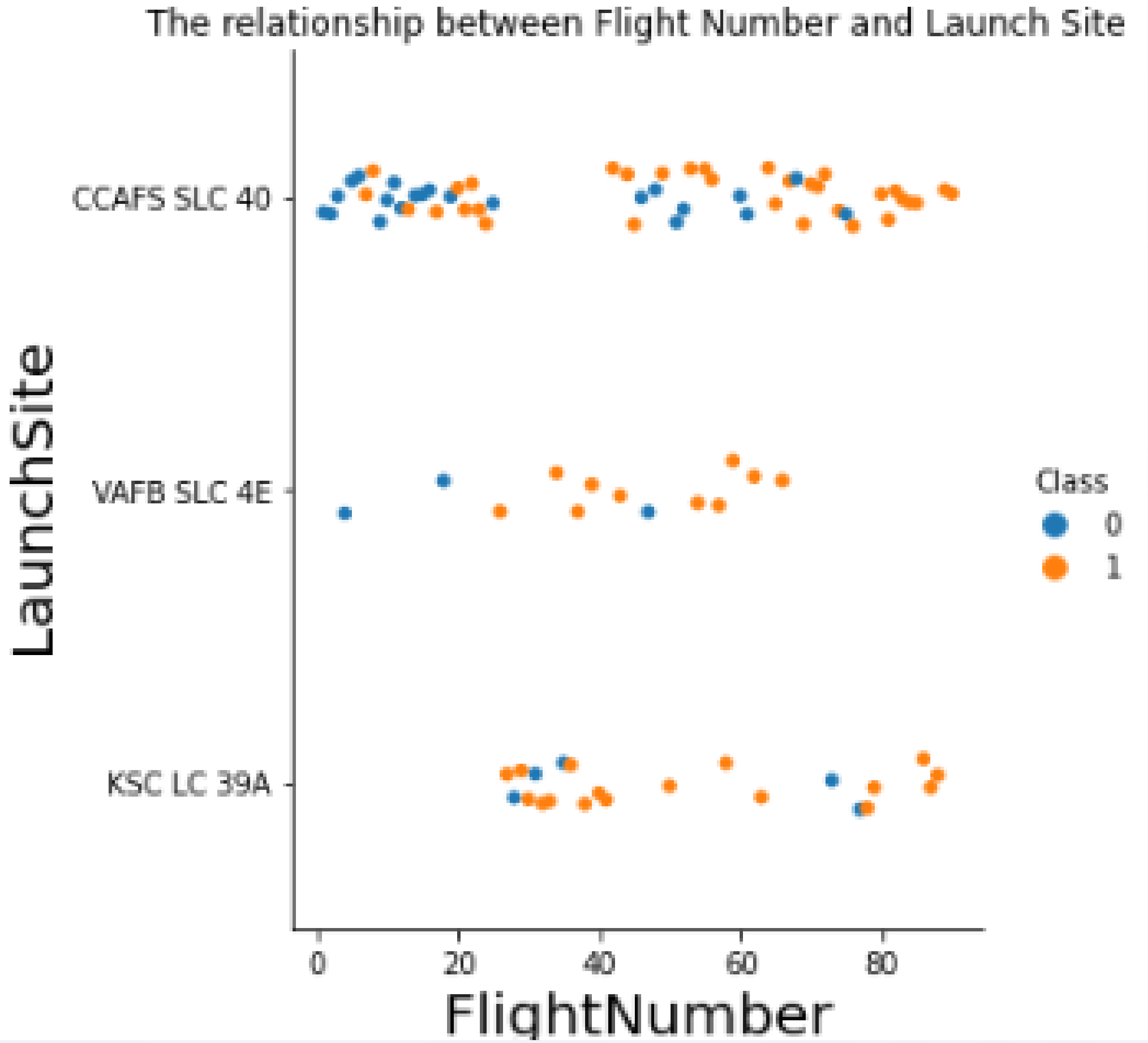
https://github.com/bisiAlgoRuthM/SpaceX/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5%20.ipynb

Section 2

# Insights drawn from EDA

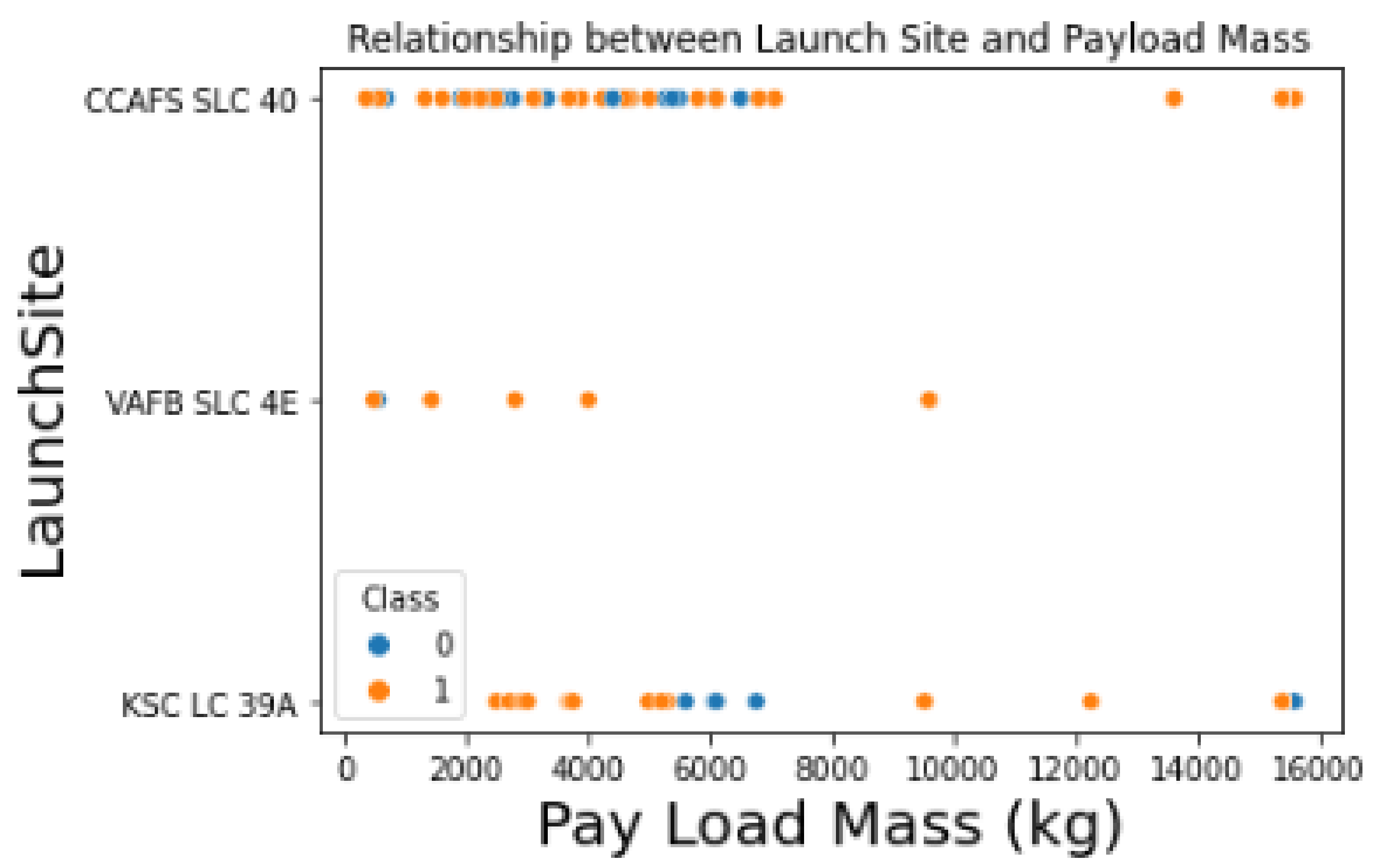
# Flight Number vs. Launch Site

- CCAFS SLC 40 has the highest number of launches
- The launches have recorded higher successes in recovering the first stage over time



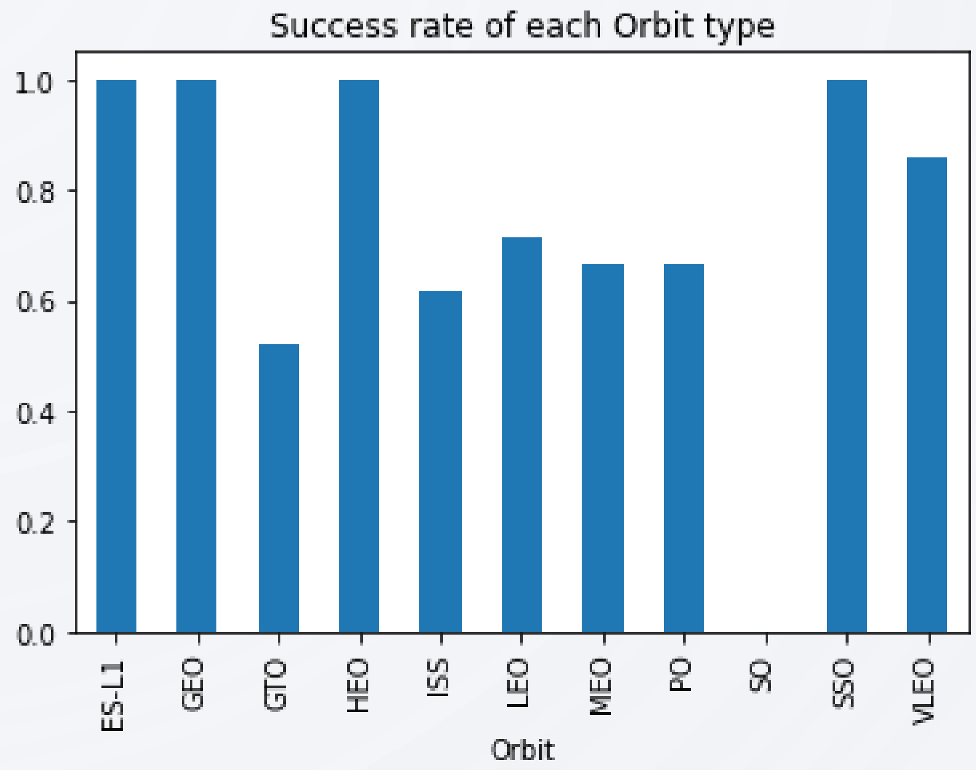
# Payload vs. Launch Site

- No rockets were launches at VAFB SLC 4E with a heavy payload mass(above 10,000kg)
- Other sites launched rockets with both heavier and lighter payload mass



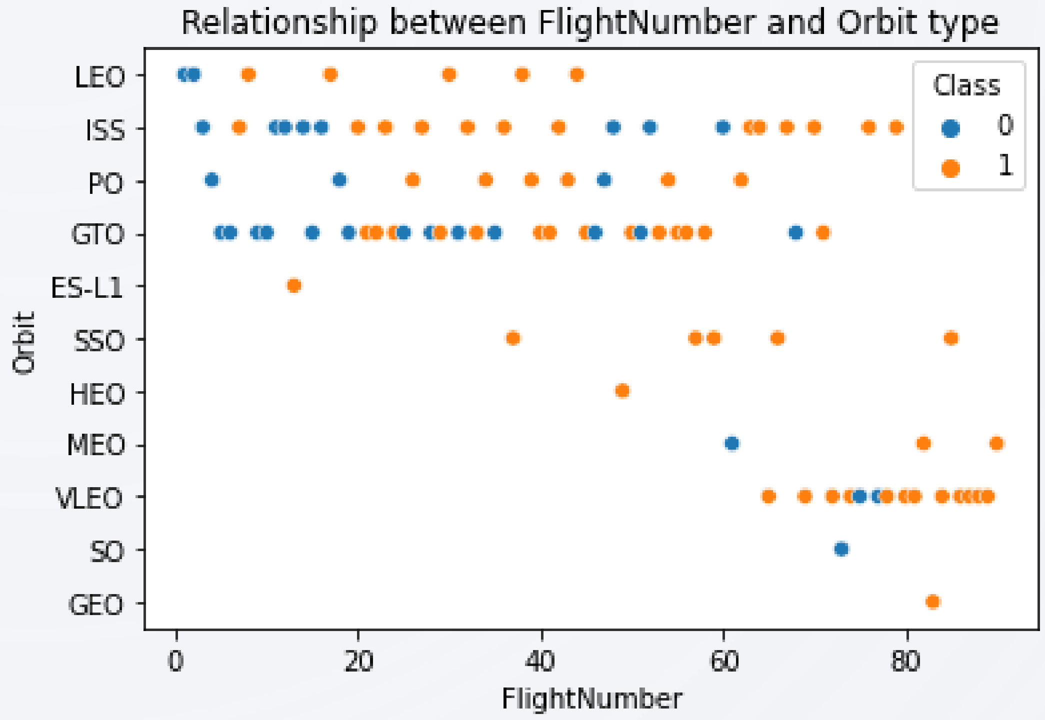
# Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO and VLEO have high success rate of landing
- SO orbit had no success in landing



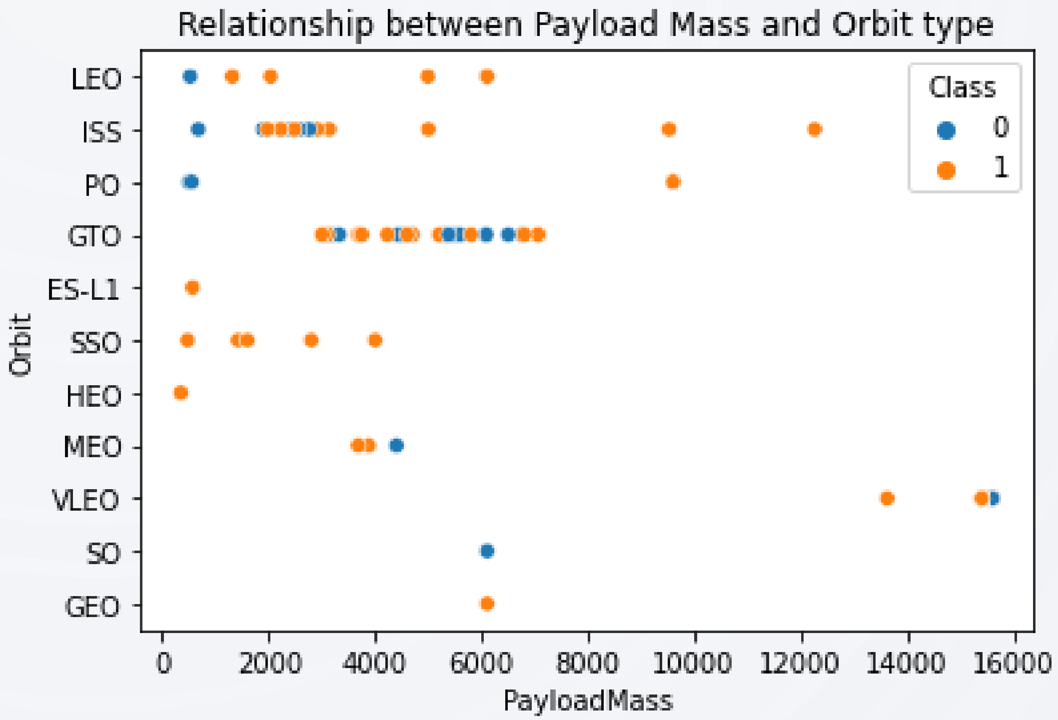
# Flight Number vs. Orbit Type

- LEO orbit success appears to increase with successive flights
- In general, there is no definitive relationship between the flight number and orbit type



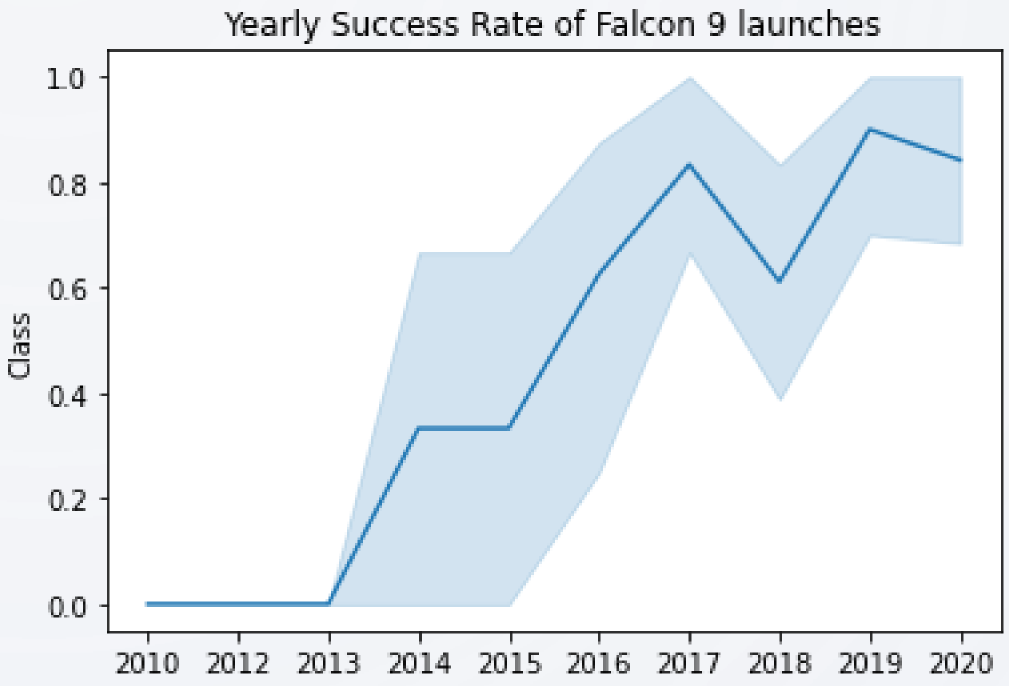
# Payload vs. Orbit Type

- Polar, LEO and ISS have a higher positive landing rate with heavy payloads
- GTO has about positive and negative landing outcomes within a similar payload mass range



# Launch Success Yearly Trend

- The success rate has progressively improved from 2013 to 2020
- There was no success from 2010 to 2012



# All Launch Site Names

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

- We used SQL to select the unique entries in the Launch_Site column in the database. The output is displayed above.

# Launch Site Names Begin with 'CCA'

| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome |
|---|---|---|---|---|---|---|---|---|
| 04-06-2010 | 18:45:00 | F9 v1.0 B0003 | CCAFS LC-40 | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success |
| 08-12-2010 | 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 |
| 22-05-2012 | 07:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success |
| 08-10-2012 | 00:35:00 | F9 v1.0 B0006 | CCAFS LC-40 | SpaceX CRS-1 | 500 | LEO (ISS) | NASA (CRS) | Success |
| 01-03-2013 | 15:10:00 | F9 v1.0 B0007 | CCAFS LC-40 | SpaceX CRS-2 | 677 | LEO (ISS) | NASA (CRS) | Success |

- This displays the first 5 launches from CCAFS LC-40 launch site

# Total Payload Mass

| TotalPayloadMassNASA(CRS) |
| --- |
| 45596 |

- The total payload carried by boosters from NASA is 45596
- Select the sum of payload mass column where the customer is NASA from the SpaceX database

# Average Payload Mass by F9 v1.1

| AVGPayloadMassF9v1.1 |
| --- |
| 2928.4 |

- Query the average of the Payload mass column where the Booster version is F9v1.1 in the database

# First Successful Ground Landing Date

| FirstSuccessfulLanding |
| --- |
| 02-03-2019 |

- Find the minimum date when landing outcome is a success in the database
- The first successful landing outcome on a ground pad was on 2nd March 2019

# Successful Drone Ship Landing with Payload between 4000 and 6000

| Booster_Version |
|---|
| F9 FT B1021.1 |
| F9 FT B1022 |
| F9 FT B1023.1 |
| F9 FT B1026 |
| F9 FT B1029.1 |
| F9 FT B1021.2 |
| F9 FT B1029.2 |
| F9 FT B1036.1 |
| F9 FT B1038.1 |
| F9 B4 B1041.1 |
| F9 FT B1031.2 |
| F9 B4 B1042.1 |
| F9 B4 B1045.1 |
| F9 B5 B1046.1 |

- These are the14 boosters that have had success in drone ship and have payload mass greater than 4000 but less than 6000

# Total Number of Successful and Failure Mission Outcomes

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

- The total number of fully successful mission outcomes is 100
- There is a case where the mission was successful but the payload status is unclear
- The failure mission outcomes is a total of 1

# Boosters Carried Maximum Payload

| 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 |

- These 12 boosters carried maximum payload
- Select the boosters where the payload mass is the maximum of all payload masses in the database

# 2015 Launch Records

| month | year | Landing _Outcome | Booster_Version | Launch_Site |
|---|---|---|---|---|
| 01 | 2015 | Failure (drone ship) | F9 v1.1 B1012 | CCAFS LC-40 |
| 04 | 2015 | Failure (drone ship) | F9 v1.1 B1015 | CCAFS LC-40 |

- Two Falcon 9 drone ships failed to crash in 2015
- Both ships were launched at CCAFS LC-40
- They were launched in January and April respectively
- The booster versions for the drone ships that failed to land are F9 v1.1 B1015 and F9 v1.1 B1012

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

| Landing _Outcome | rank |
|---|---|
| Success (ground pad) | 6 |
| Success (drone ship) | 8 |
| Success | 20 |

- Select distinct landing outcomes, count the frequency of each outcome that is a Success and filter the query to only include records between the date 2010-06-04 and 2017-03-20, in descending order

Section 3

# Launch Sites Proximities Analysis

# Space X Launch Sites


From the map generated, we can observe that :

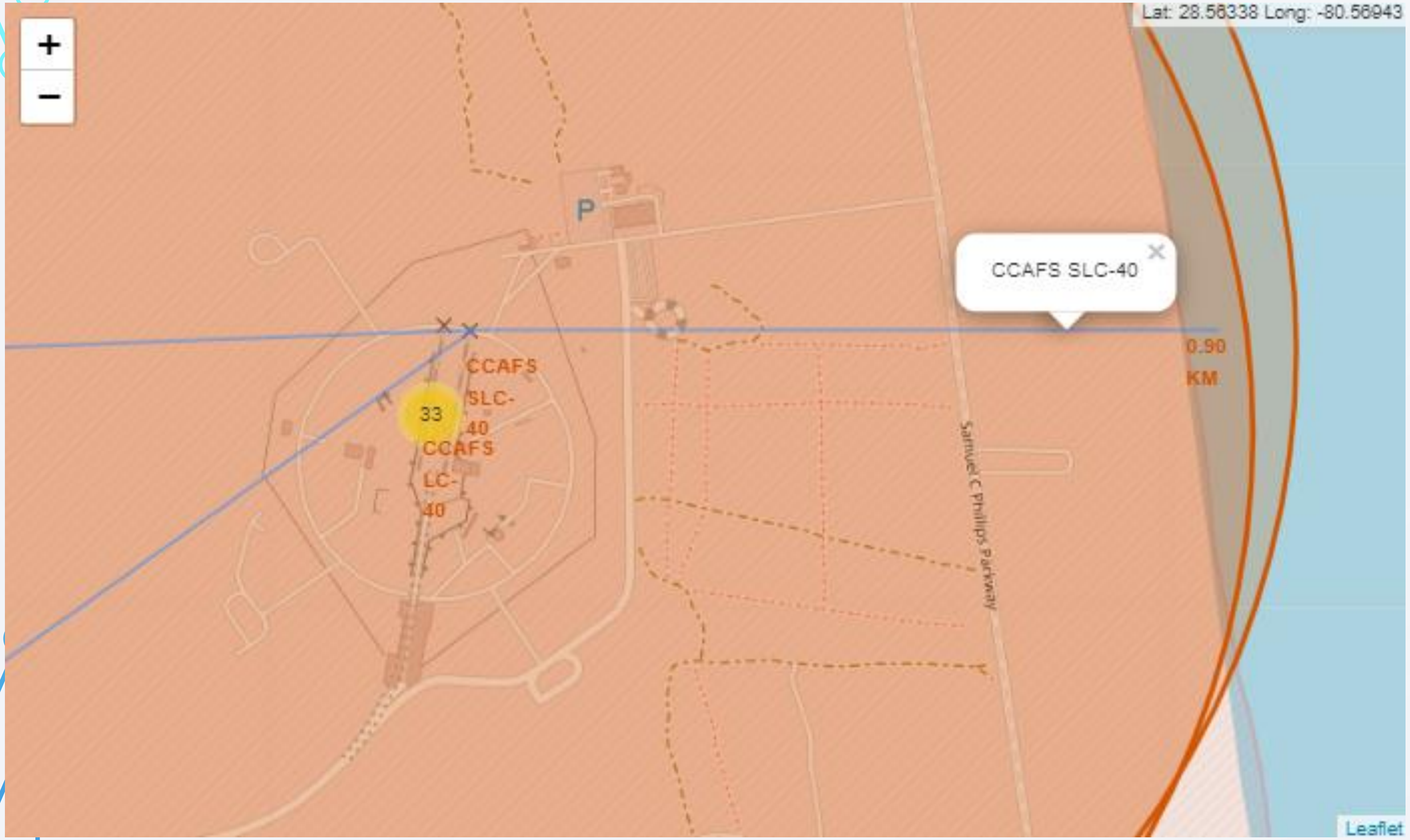
- Launch sites are all in proximity to the coast
- Launch sites are far from the equator line

# Success or failed landings for each launch site


| Launch Site | Failed landings | Successful landings |
|---|---|---|
| CCAFS SLC-40 | 4 | 3 |
| CCAFS LC-40 | 19 | 7 |
| KSC LC-39A | 3 | 10 |
| VAFB SLC-4E | 6 | 4 |

- Green labels represent positive landing outcomes and red labels, vice versa

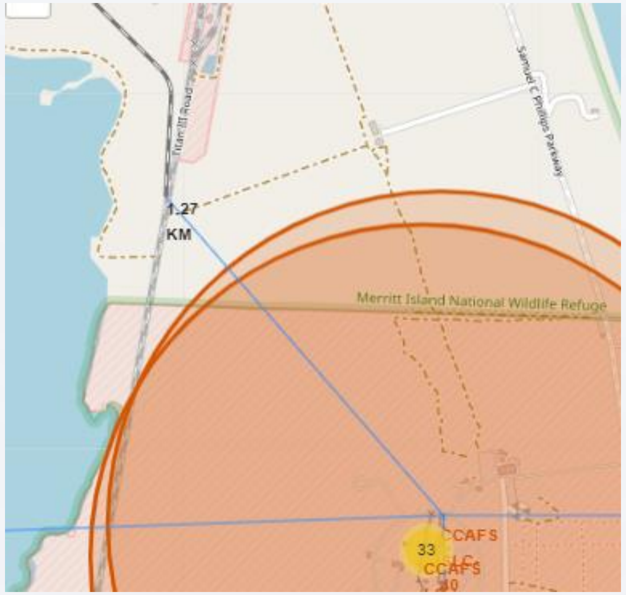
# Distance between CCAFS SLC-40 and the nearest coastline




- The distance between this launch site and the nearest coastline is 0.9km
- This puts the launch site in close proximity to the coastline

# Distance between a launch to its proximities



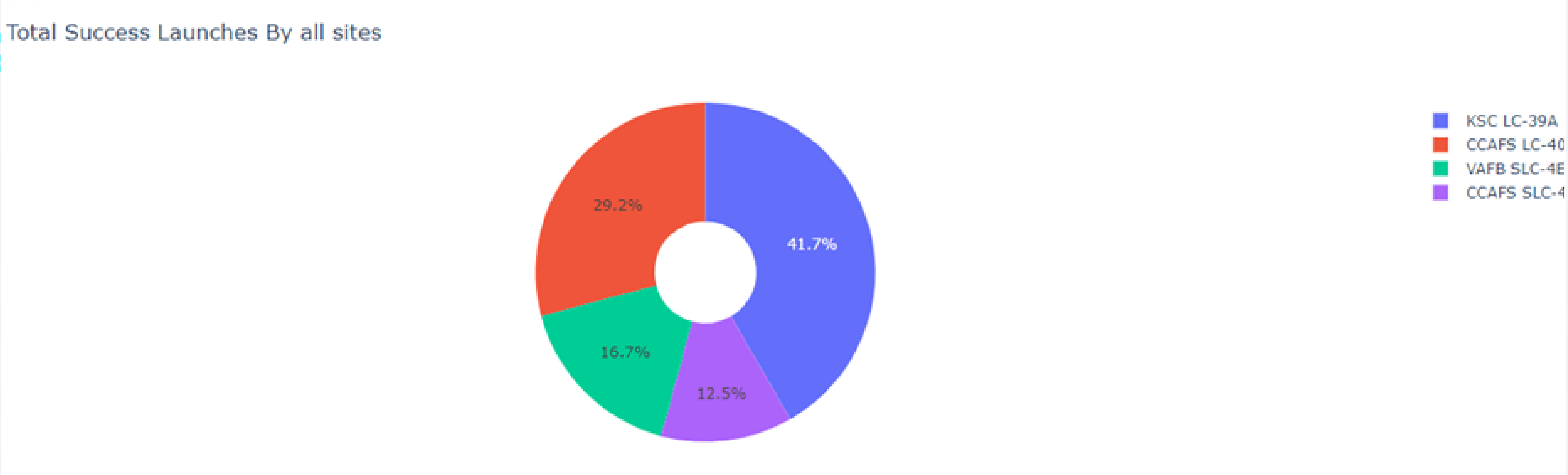



- Florida city is 78.45 km away from CCAFS SLC-40.
- NASA Railroad is 21.3km from CCAFS SLC-40
- Samuel C Philips Parkway, a highway, is 7.0km away from CCAFS SLC-40

Section 4

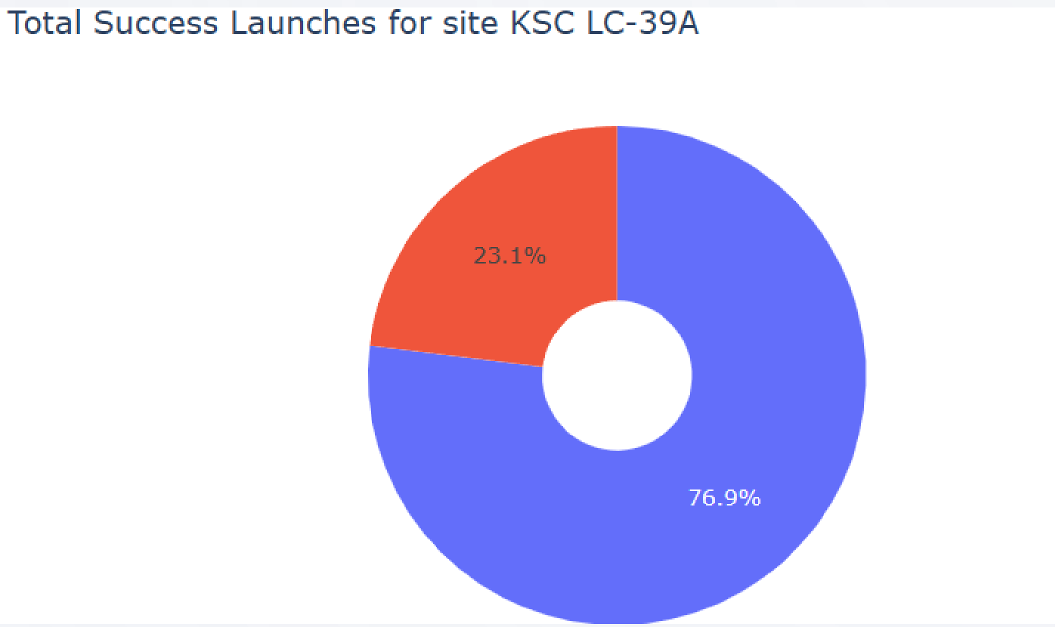
# Build a Dashboard with Plotly Dash

# Launch Success Sites in a Pie chart




- KSC LC 39A has recorded the highest success rate

# Launch Site KSC LC-39A



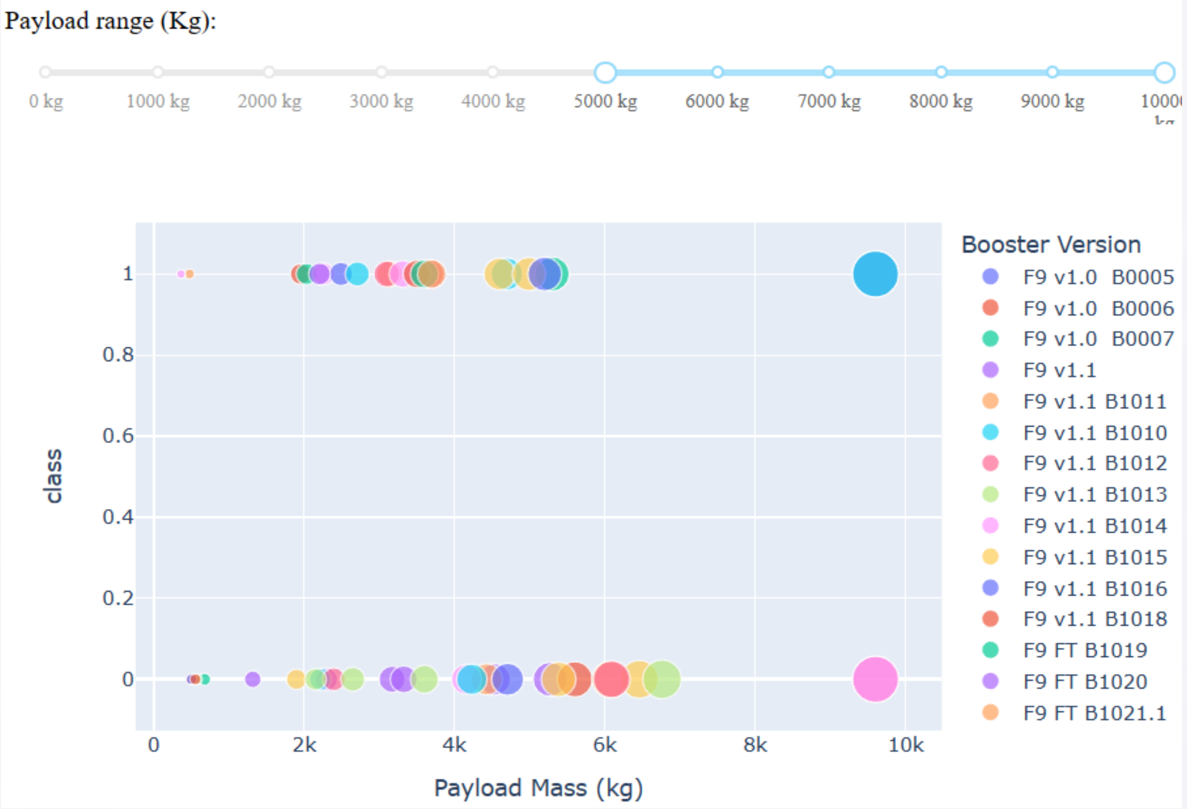


- KSC LC-39A has the highest launch success ratio
- KSC LC-39A had the most successful launches (76.9%) with 10 booster versions
- With payload < 4000kg, the landing was always successful for this launch site

# Payload VS Launch Outcome for All sites


- Payload mass ranging from 2000kg to 3700kg have the highest success rate
- Lighter payload masses (<2000kg) have a low success rate
- Heavy payload masses can land. However, they still have a relatively low chance of successful landing

# Payload VS Launch Outcome

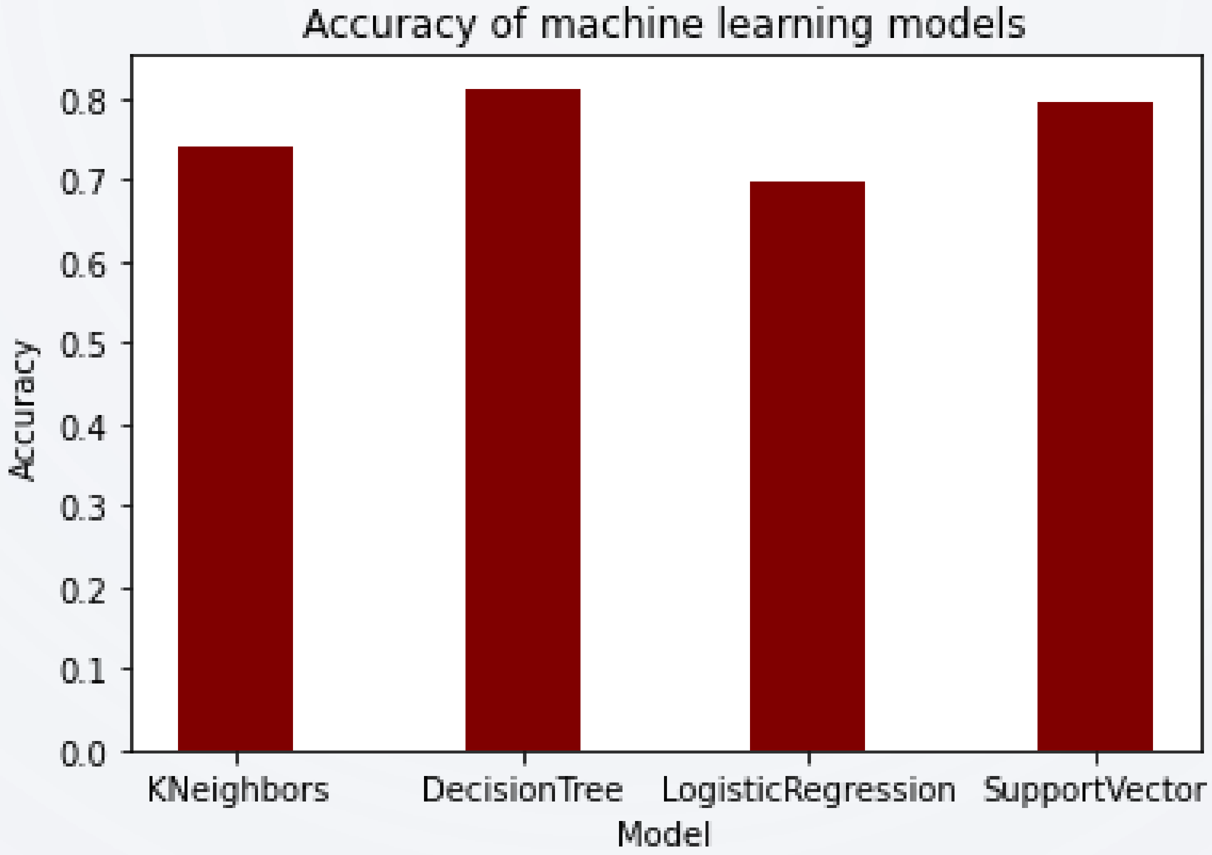



- The heaviest payload was successfully launched with booster version F9 B4 B104.1 and a payload mass of 9600kg

Section 5

# Predictive Analysis (Classification)

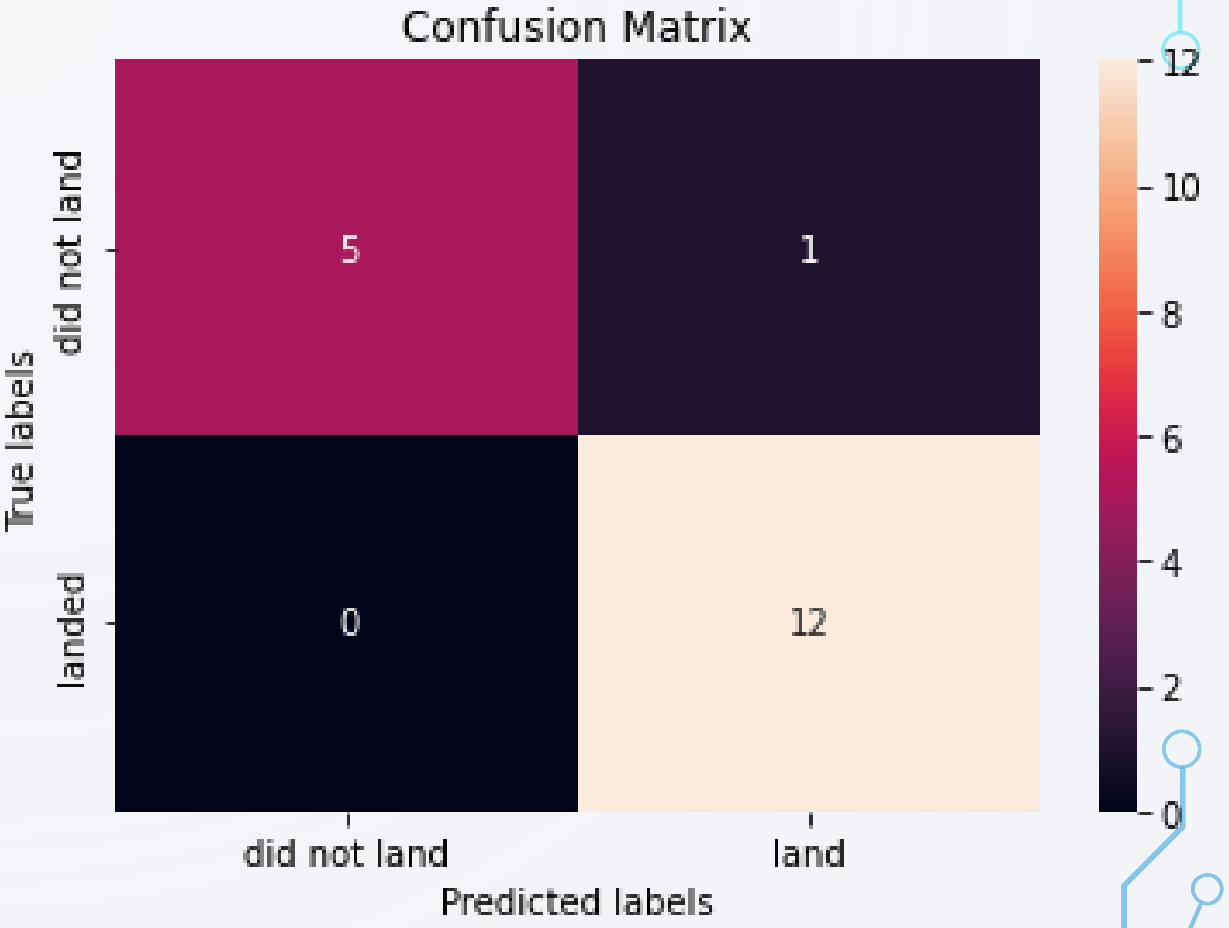
# Classification Accuracy




- Decision Tree model has the highest accuracy of classifying the outcomes
- Support vector machine also has a high prediction accuracy on the test data

# Confusion Matrix


Decision tree




Support Vector Machine

# Confusion Matrix


Logistic Regression


K Nearest Neighbor

# Conclusions

- Decision Tree model performed best in terms of predicting landing outcomes in this dataset
- The success rate of Space X launches has increased progressively over time
- KSC LC 39A has the most successful launches from all sites
- Booster version F9 B4 B1041.1 has the heaviest successful launch
- Although there were cases where there were successful launches at different payload masses, 2000kg and 3660kg is the most successful range
- Space X launch sites are usually situated close to the coastlines but far from city centers

# Thank you!