IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science


James Hughey

09/16/2024

# Outline

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

# Executive Summary

- Used the Data Science Life Cycle
  - Business Understanding
  - Data Mining
  - Data Cleaning
  - Data Exploration
  - Feature Engineering
  - Predictive Modeling
  - Data Visualizations

- Summary of all results
  - Used various machine learning models
  - Was able to accurately predict the outcome of most landings

# Introduction

- SpaceX advertises Falcon 9 rocket launches cost of 62 million dollars
- Other providers cost upward of 165 million dollars each
- Much of the savings is because SpaceX can reuse the first stage
- If we can determine if the first stage will land, we can determine the cost of a launch
- This information can be used if an alternate company wants to bid against SpaceX for a rocket launch
- Problems you want to find answers for
    - Success of the first stage landing
    - Cost of launch

Section 1

# Methodology

# Methodology

## Executive Summary

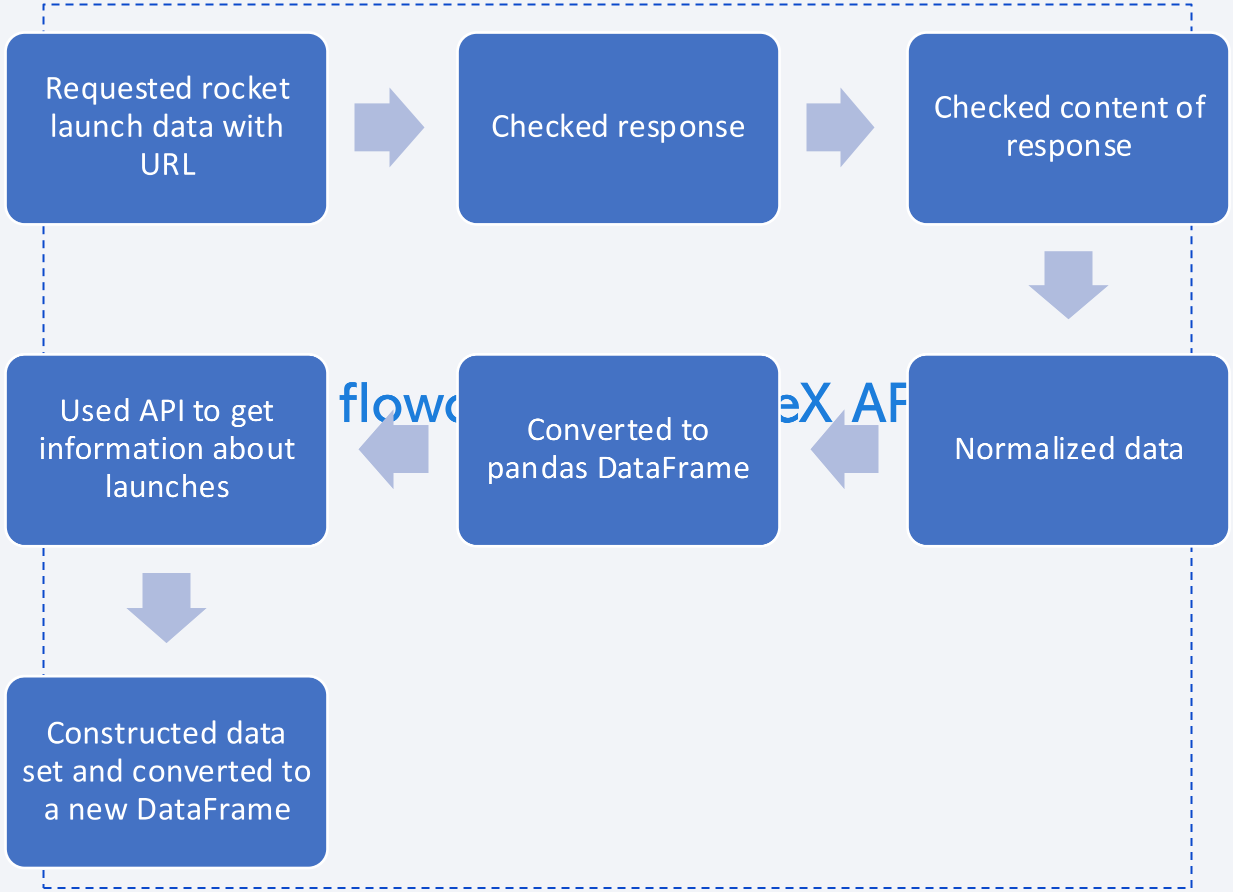
- Data collection methodology:
    - Data was collected utilizing the SpaceX API
- Perform data wrangling
    - Missing data was replaced with mean values
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - How to build, tune, evaluate classification models

# Data Collection

- Describe how data sets were collected.
- Data was collected using SpaceX API
  - Collected via get request
- Abstracted the names from various columns
  - Booster Name
  - Rockets
  - Landing Sites
  - Cores

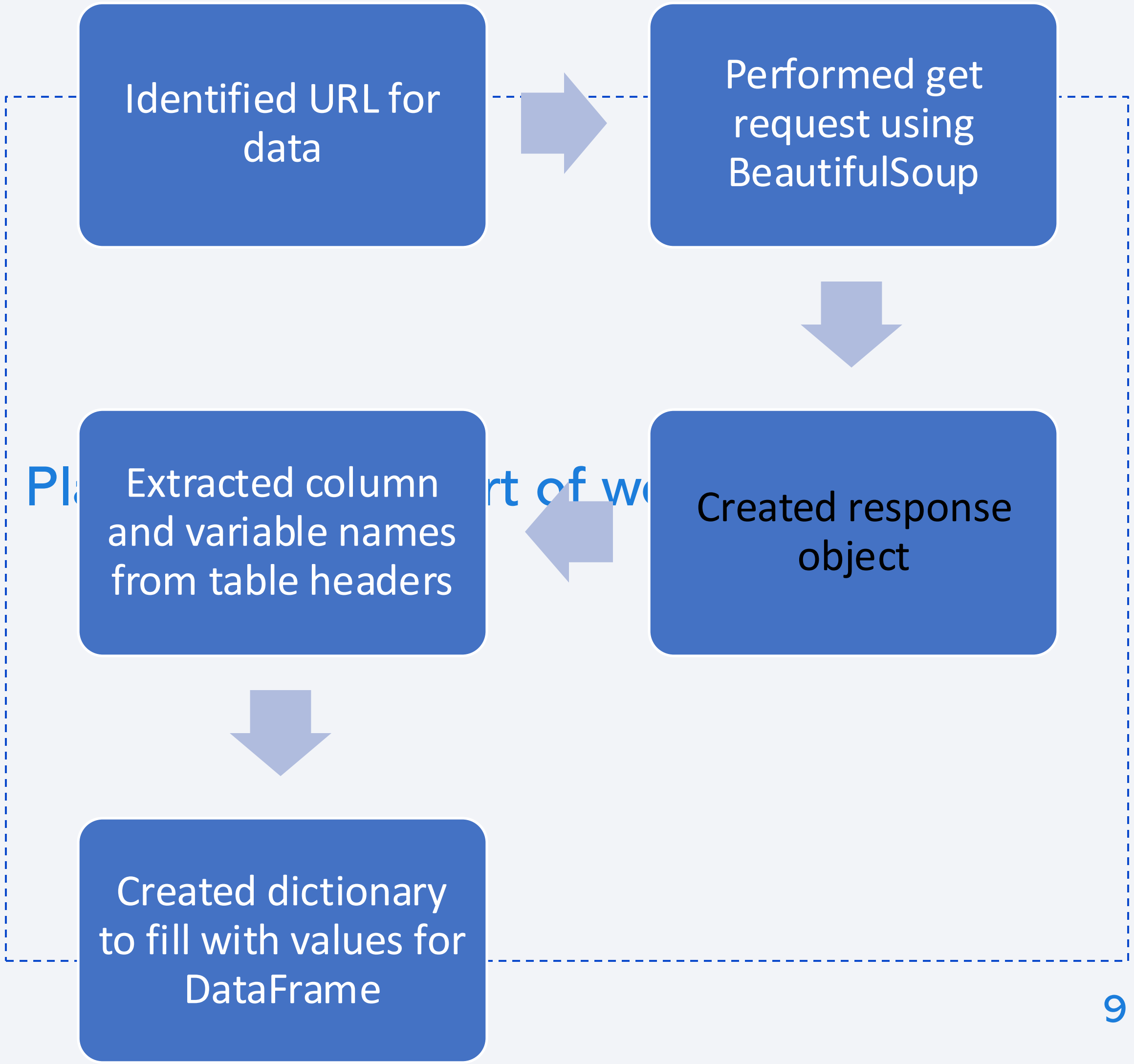
# Data Collection – SpaceX API

- URL of Jupyter notebook
    - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/jupyter-labs-spacex-data-collection-api.ipynb



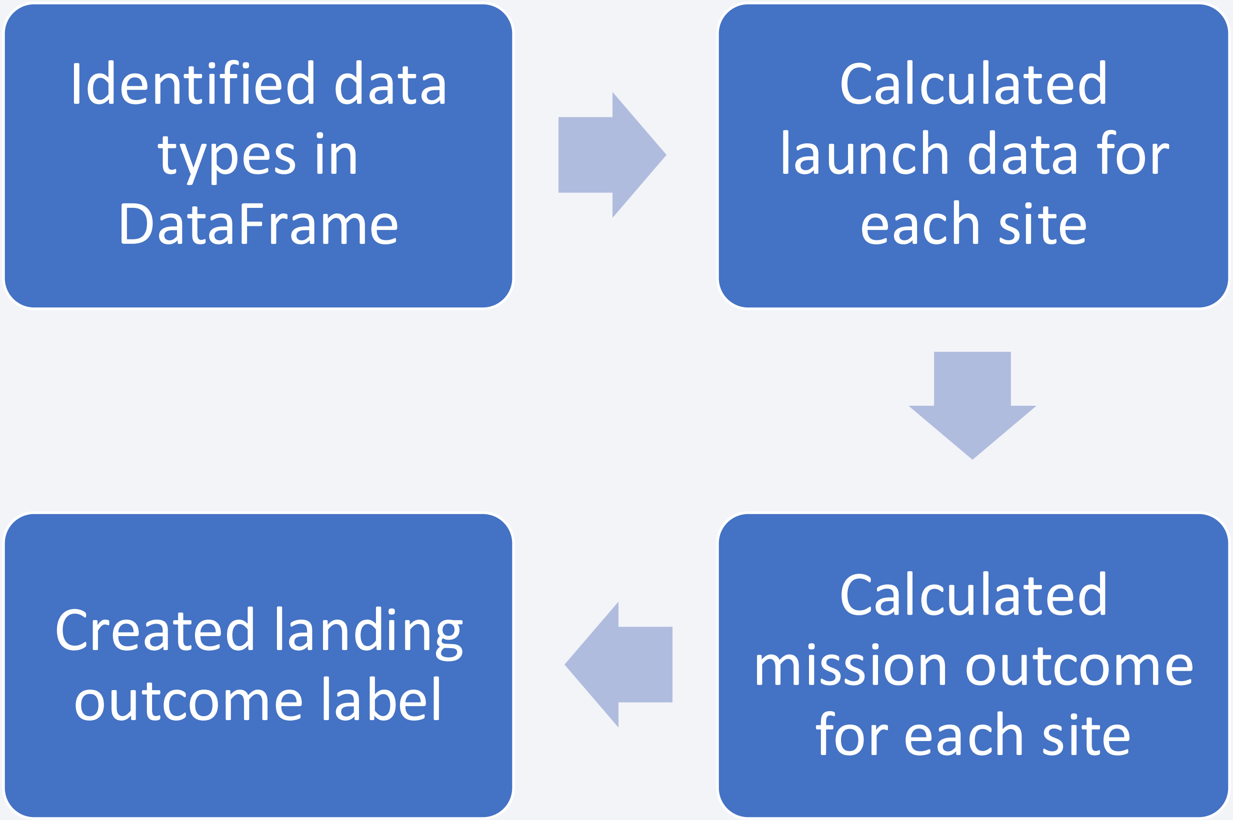
# Data Collection - Scraping

- URL of Web Scraping Jupyter notebook
  - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/jupyter-labs-webscraping.ipynb



# Data Wrangling

- URL of Data Wrangling Jupyter notebook
  - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/labs-jupyter-spacex-Data%20wrangling.ipynb



# EDA with Data Visualization

- Utilized scatter point charts to compare relationships between features
  - Flight Number vs Launch Site
  - Payload mass vs Launch Site
  - Flight Number vs Orbit Type
  - Payload mass vs Orbit Type
- Plotted Bar chart to compare success rate of each orbit type
- Visualized yearly launch trends with line graph
- Feature engineered relative data and used one hot encoding for categorical values
- Notebook URL - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/edadataviz.ipynb

# EDA with SQL

- Some SQL queries performed:
    - Display the names of the unique launch sites
    - Display 5 records where launch sites begin with the string 'CCA'
    - Display the total payload mass carried by boosters
    - Display average payload mass
    - List the date when the first successful landing outcome
    - List the total number of successful and failure mission outcomes
- GitHub URL - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/jupyter-labs-eda-sql-coursera_sqllite.ipynb

# Build an Interactive Map with Folium

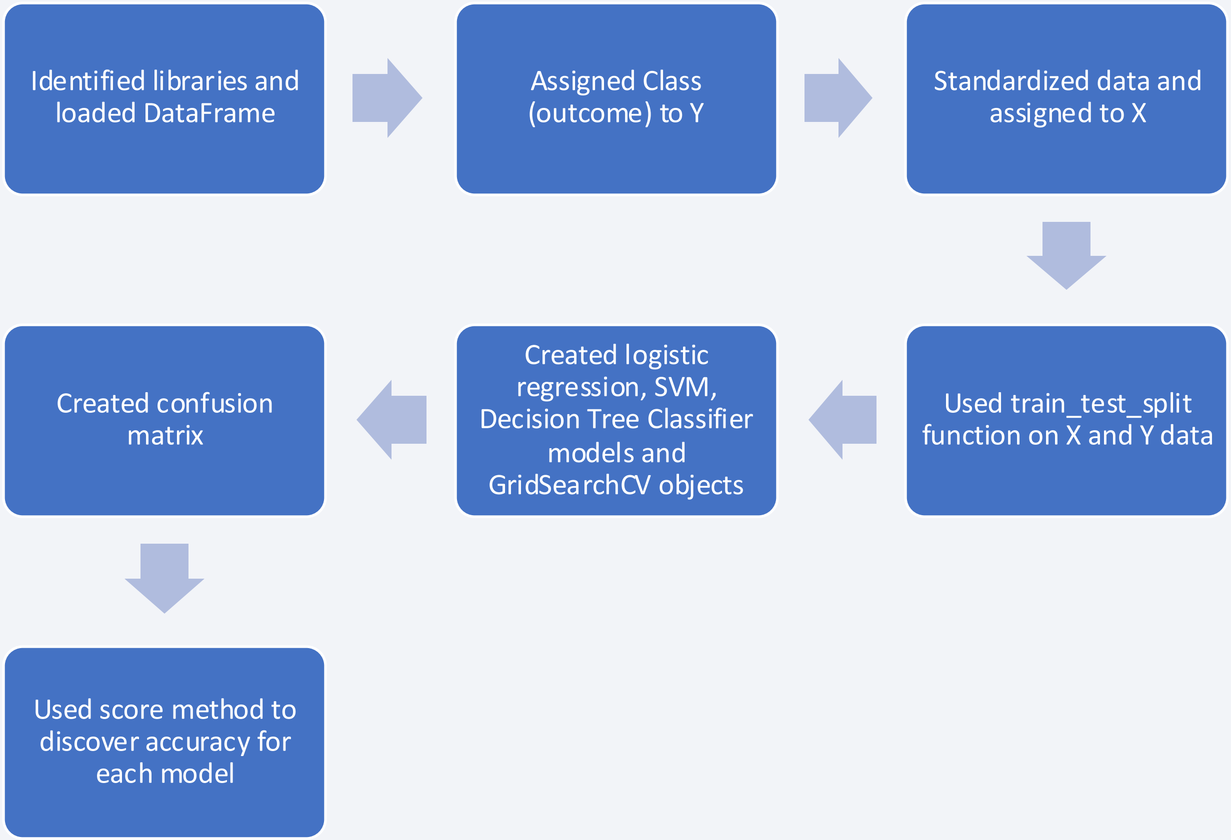
- Used circle to create zoom location
- Added circles for each launch site to see each location
- Added markers to identify names of launch sites
- Added lines to identify distances from landmarks
  - Highways
  - Cties
  - Coastline
- GitHub URL- https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/lab_jupyter_launch_site_location.ipynb

# Build a Dashboard with Plotly Dash

- Attempted to add dropdown menus, pie chart, etc
- I was unable to. None of the visualizations ever showed (as with the original lab).
- GitHub URL - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/80d0a12af259758356d932cb10c8b7c4572f55c5/spacex_dash_app.py

# Predictive Analysis (Classification)

- Evaluated several models
  - Logistic Regression
  - SVM
  - Decision Tree Classifier
- Fit using GridSearchCV
- SVM had highest accuracy
- GitHub URL - https://github.com/Jhughey24/IBM-Data-Science-Capstone-Project/blob/2eb46b7d7fac1f55e6e83692c67974945dde6b04/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



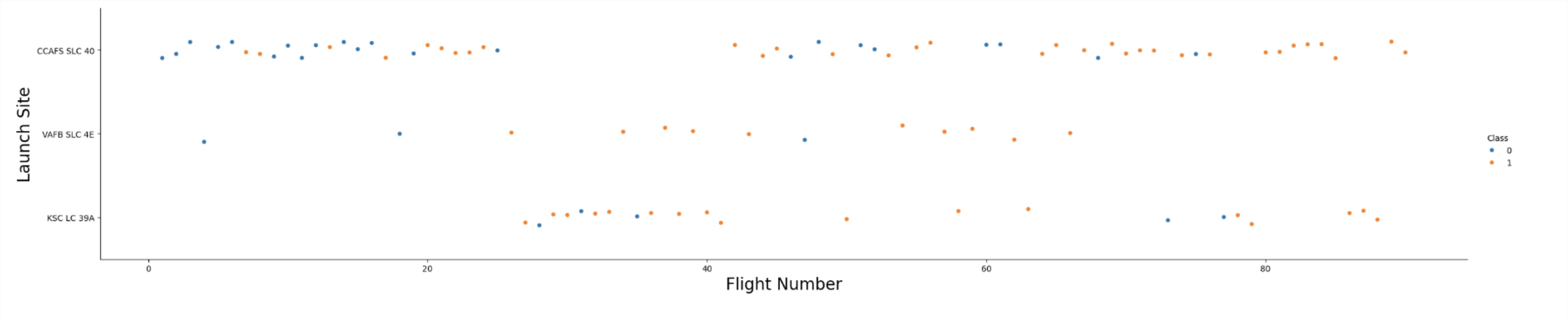
# Results

- Exploratory data analysis results
  - Created outcomes for each type of launch
  - Listed outcome with each launch site
- Dash did not work. Do not have the screenshots
- Predictive analysis results show that SVM had the highest accuracy and would allow us to predict the landing outcomes the best.

Section 2

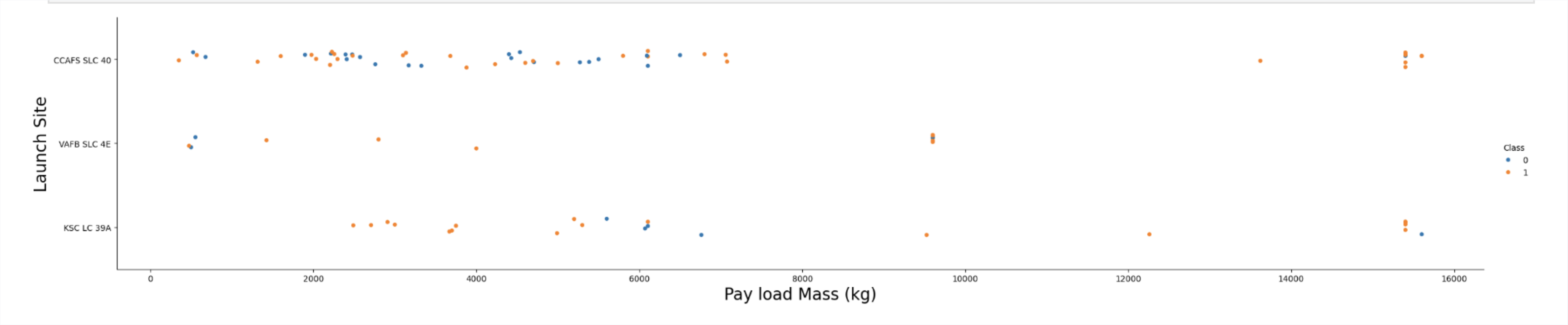
# Insights drawn from EDA

# Flight Number vs. Launch Site




- Launches became more successful with more flights

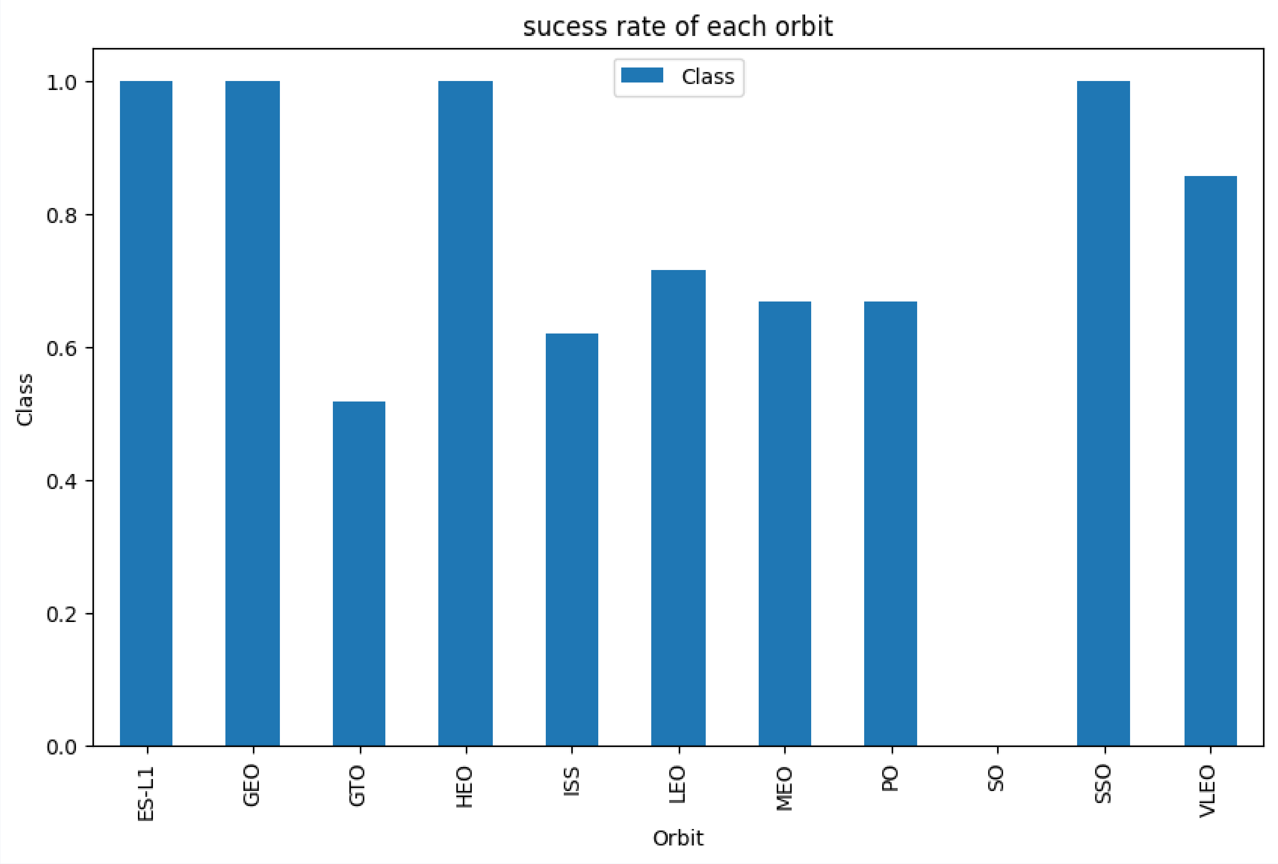
# Payload vs. Launch Site



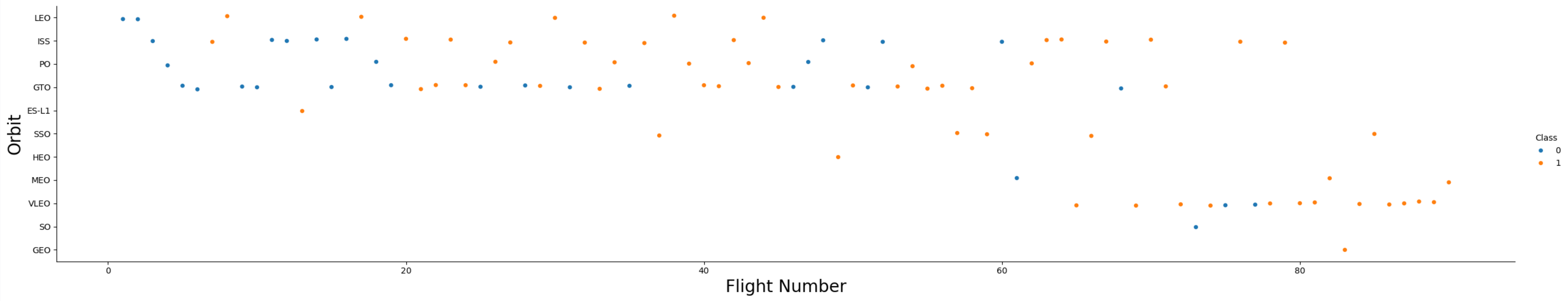

- VAFB-SLC launchsite - no rockets launched for heavypayload mass (greater than 10000)
- KSC LC 39A did better with smalled payloads
- Mixe

# Success Rate vs. Orbit Type

- Half orbits had moderate success rates
- The other half was generally successful

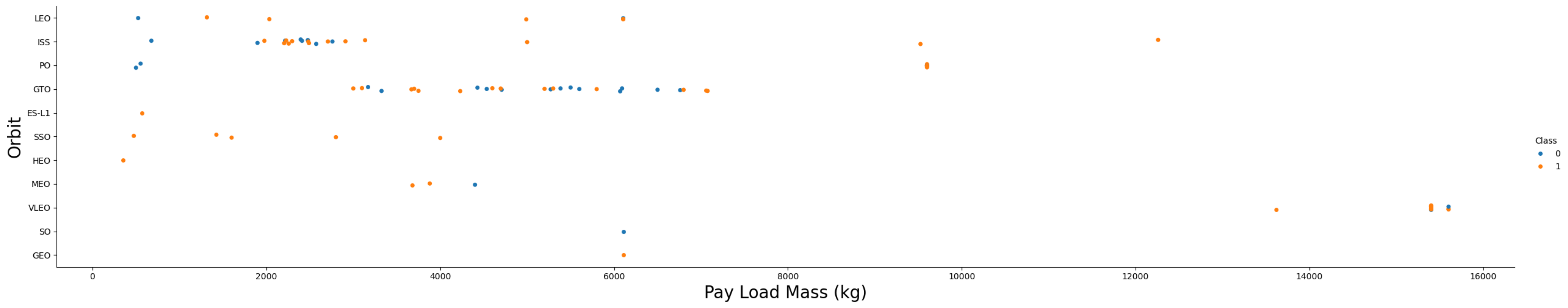


# Flight Number vs. Orbit Type




- In the LEO orbit, success seems to be related to the number of flights
- Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

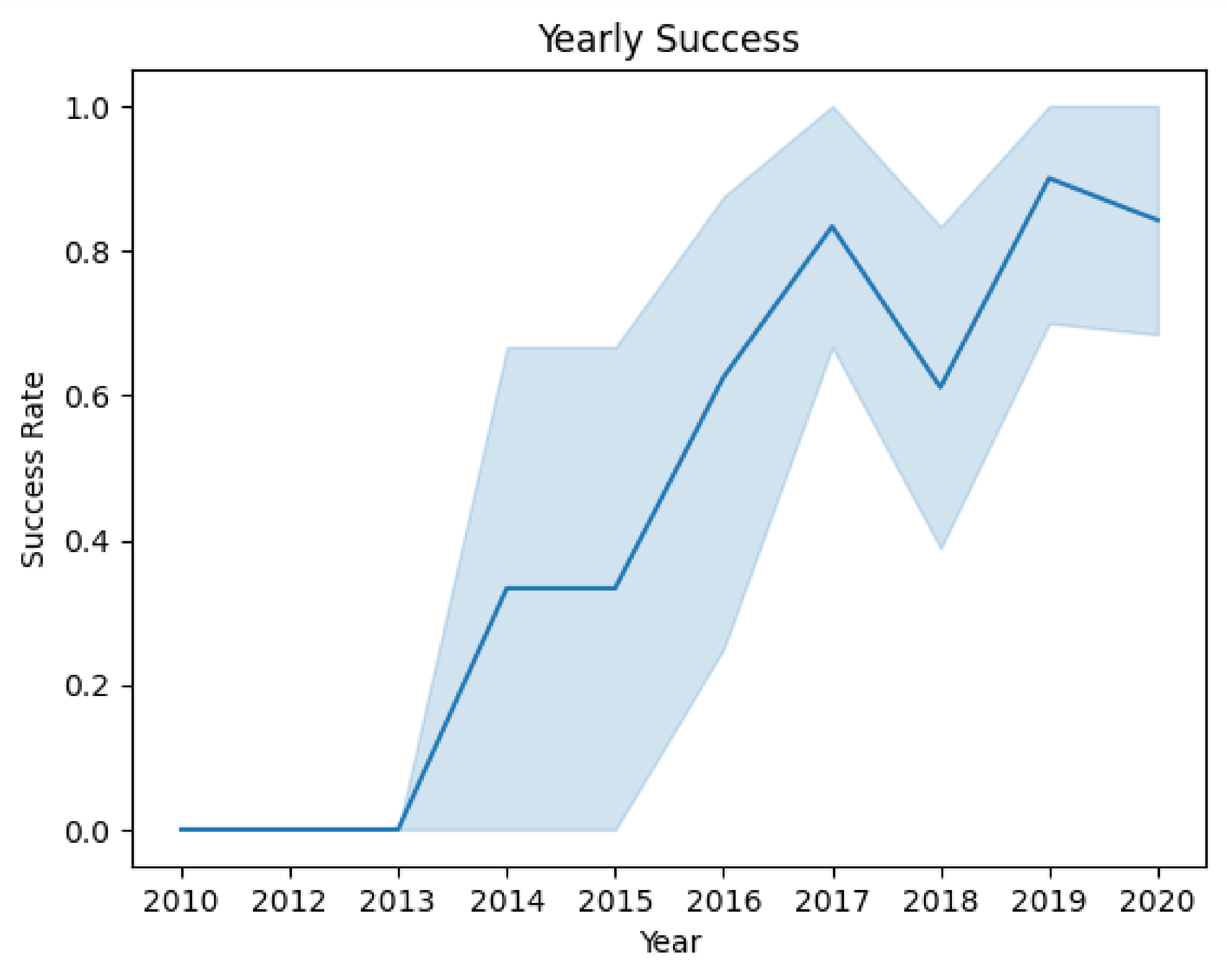
# Payload vs. Orbit Type




- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

# Launch Success Yearly Trend

- You can observe that the success rate since 2013 kept increasing till 2020



# All Launch Site Names

- Names of the unique launch sites
- Listed with Lattitude and Longitude coordinates used for plotting

| | Launch Site | Lat | Long |
|---|---|---|---|
| **0** | CCAFS LC-40 | 28.562302 | -80.577356 |
| **1** | CCAFS SLC-40 | 28.563197 | -80.576820 |
| **2** | KSC LC-39A | 28.573255 | -80.646895 |
| **3** | VAFB SLC-4E | 34.632834 | -120.610745 |

# Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Query used
  - **%sql** `SELECT "Launch_Site" from SPACEXTBL where "Launch_Site" LIKE 'CCA%' LIMIT 5;`
- Present your query result with a short explanation here
  - I did not select the whole row and missed some the data
  - I only retrieved the launch site - CCAFS LC-40

# Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Query used: `%sql SELECT SUM("PAYLOAD_MASS__KG_") from SPACEXTBL where Customer = 'NASA (CRS)'`
- `Result:` 45596 KG

# Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Quesy used: **`%sql`** `SELECT AVG("PAYLOAD_MASS__KG_") from SPACEXTBL where "Booster_Version" = 'F9 v1.1'`
- `Result:` 2928.4 KG

# First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Query used: `%sql SELECT min(Date) from SPACEXTBL where "Mission_Outcome" = 'Success';`
- Result: 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
- Query used: %sql SELECT "Booster_Version" from SPACEXTBL where "Mission_Outcome" = 'Success' & "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;
- Result:

# Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Query used: `%sql SELECT (SELECT COUNT("Mission_Outcome") from SPACEXTBL where "Mission_Outcome" like '%Success%') as Success, (SELECT COUNT("Mission_Outcome") from SPACEXTBL where "Mission_Outcome" like '%Failure%') as Failure;`
- Result: Success: 100 – Failure: 1

# Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Query used: %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
- Query:

%%sql

SELECT Date, "Landing_Outcome","Booster_Version","Launch_Site" from SPACEXTBL as 'Failure' where "Landing_Outcome" like '%drone ship%' and substr(Date, 6,2) and substr(Date,0,5)='2015'

| Date | Landing_Outcome | Booster_Version | Launch_Site |
|---|---|---|---|
| 2015-01-10 | Failure (drone ship) | F9 v1.1 B1012 | CCAFS LC-40 |
| 2015-04-14 | Failure (drone ship) | F9 v1.1 B1015 | CCAFS LC-40 |
| 2015-06-28 | Precluded (drone ship) | F9 v1.1 B1018 | 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
- Query used: **%%sql**
- **SELECT** Date, "Landing_Outcome" **from** SPACEXTBL **where** Date **between** '2010-06-04' **AND** '2017-03-20' **ORDER BY** DATE **DESC**

| Date | Landing_Outcome |
|---|---|
| 2017-03-16 | No attempt |
| 2017-02-19 | Success (ground pad) |
| 2017-01-14 | Success (drone ship) |
| 2016-08-14 | Success (drone ship) |
| 2016-07-18 | Success (ground pad) |
| 2016-06-15 | Failure (drone ship) |
| 2016-05-27 | Success (drone ship) |
| 2016-05-06 | Success (drone ship) |
| 2016-04-08 | Success (drone ship) |
| 2016-03-04 | Failure (drone ship) |
| 2016-01-17 | Failure (drone ship) |
| 2015-12-22 | Success (ground pad) |
| 2015-06-28 | Precluded (drone ship) |
| 2015-04-27 | No attempt |
| 2015-04-14 | Failure (drone ship) |
| 2015-03-02 | No attempt |
| 2015-02-11 | Controlled (ocean) |
| 2015-01-10 | Failure (drone ship) |
| 2014-09-21 | Uncontrolled (ocean) |
| 2014-09-07 | No attempt |
| 2014-08-05 | No attempt |
| 2014-07-14 | Controlled (ocean) |
| 2014-04-18 | Controlled (ocean) |
| 2014-01-06 | No attempt |
| 2013-12-03 | No attempt |
| 2013-09-29 | Uncontrolled (ocean) |
| 2013-03-01 | No attempt |
| 2012-10-08 | No attempt |
| 2012-05-22 | No attempt |
| 2010-12-08 | Failure (parachute) |
| 2010-06-04 | Failure (parachute) |

Section 3

# Launch Sites Proximities Analysis

# Folium Map Screenshots

- Due to an unknown error I was not able to make the visualizations work.
- I completed to code to the best of my ability but nothing showed up on the skeleton dashboard

Section 4

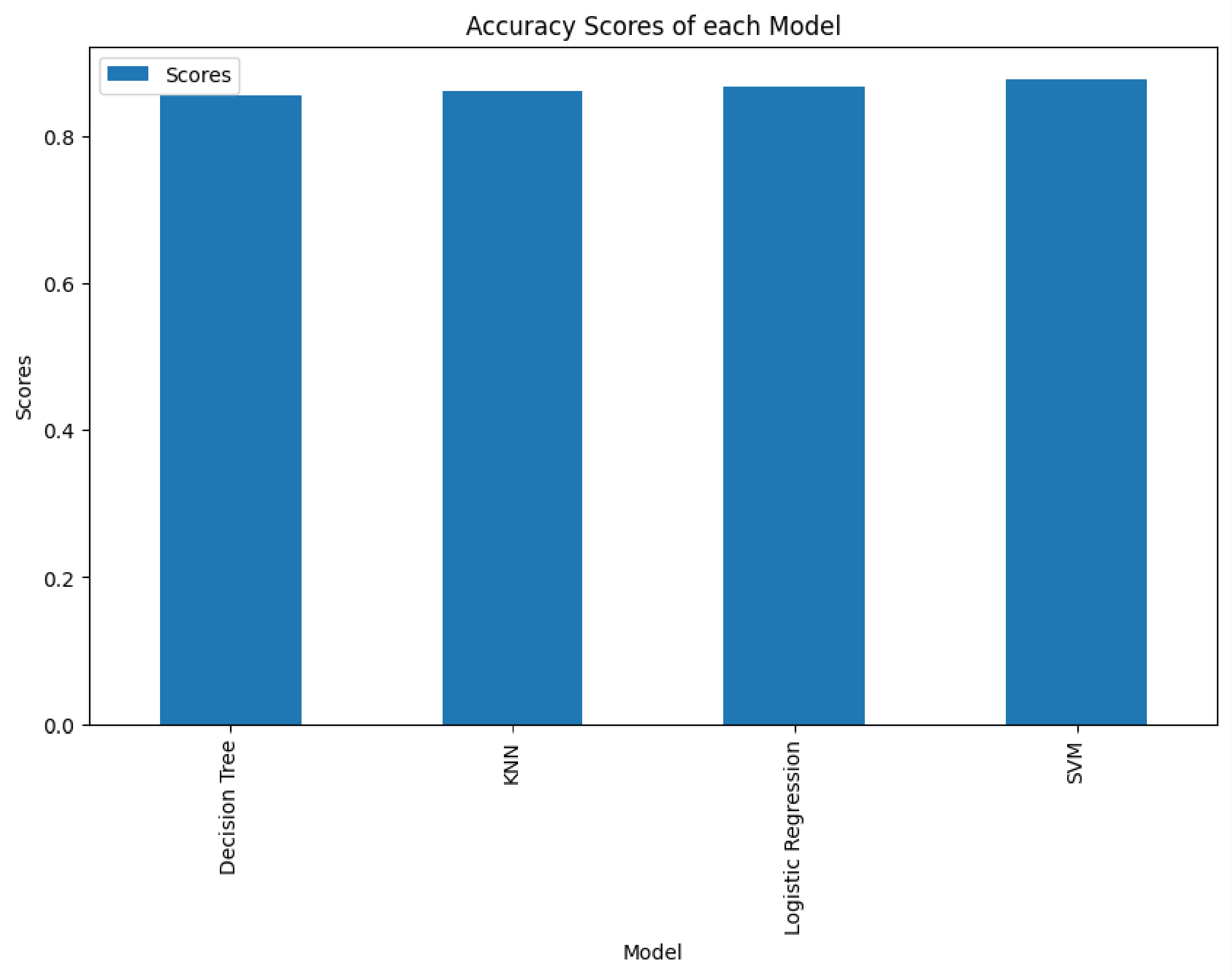
# Build a Dashboard with Plotly Dash

Section 5

# Predictive Analysis (Classification)

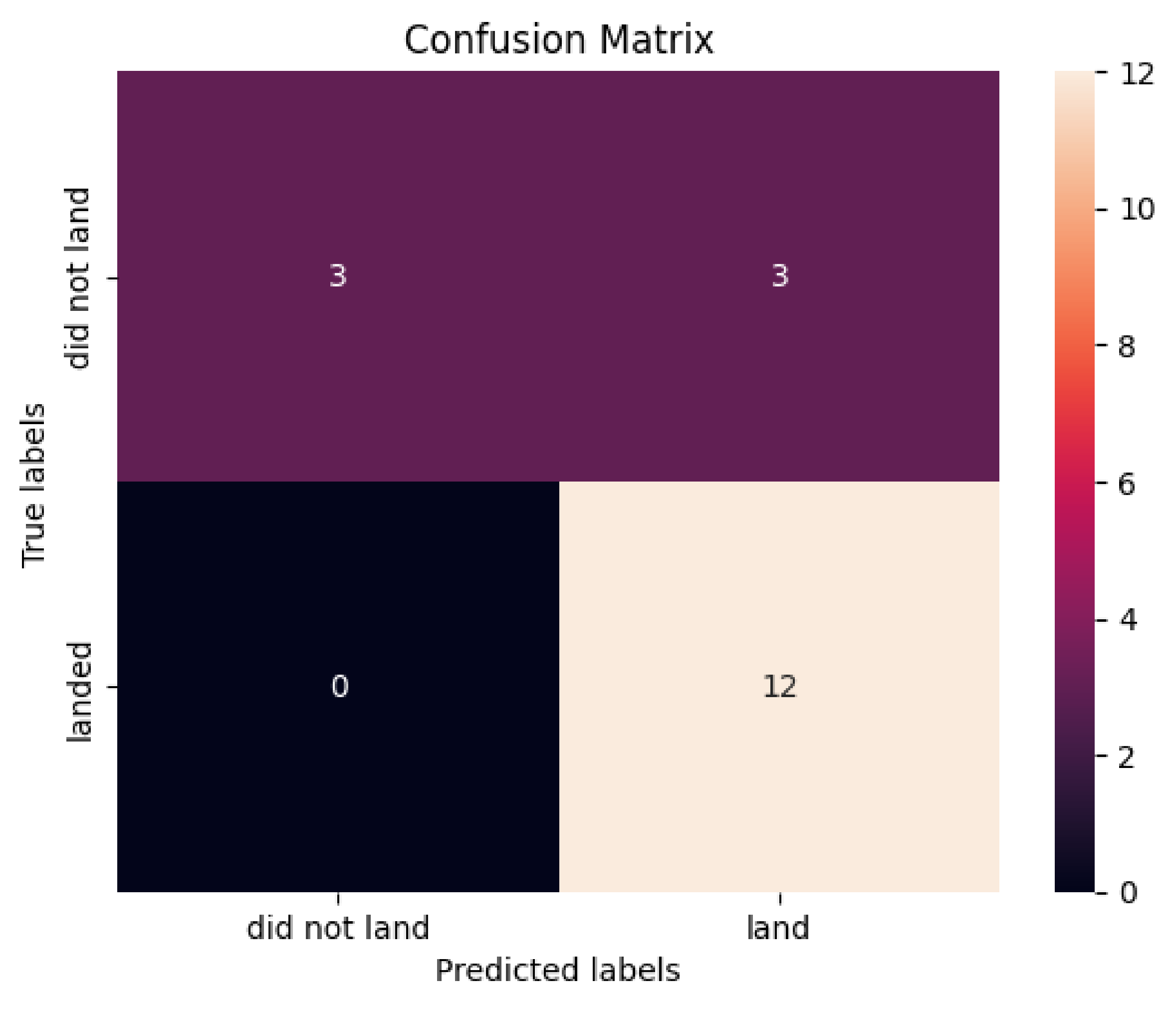
# Classification Accuracy

- As observed, the models have fairly the same amount of accuracy
- The SVM model has a slightly higher accuracy score



# Confusion Matrix

- Confusion Matrix
- Positively predicted 12 successful landings
- Incorrectly predicted 3 failed landings as successful
- Correctly predicted 3 failed landings
- Did not predict failed landings as successful landings



# Conclusions

- Landings generally get better as more flights are launched
- Heavier payloads can lead to failed landings
- Some launch sites are more successfulthan others
- The best model to predict successful landings is with SVM machine

# Appendix

- None

# Thank you!