IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

Context

- SpaceY cost advantages depends primarily on its ability to reuse the first stage
- Predicting if a first stage is going to land safely is critical

Objectives:

- Find key characteristics of that share boosters with successful recoveries
- Given the characteristics of new rocket determine if its going to be recovered

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Directly from SpaceX API
  - By web scrapping Wikipedia data
- Perform data wrangling
  - Lightly analysis of the data and creation of a binary value for the outcome of the recovery
- 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

Data was collected from two sources:

SpaceX API:

Request calls using python request library

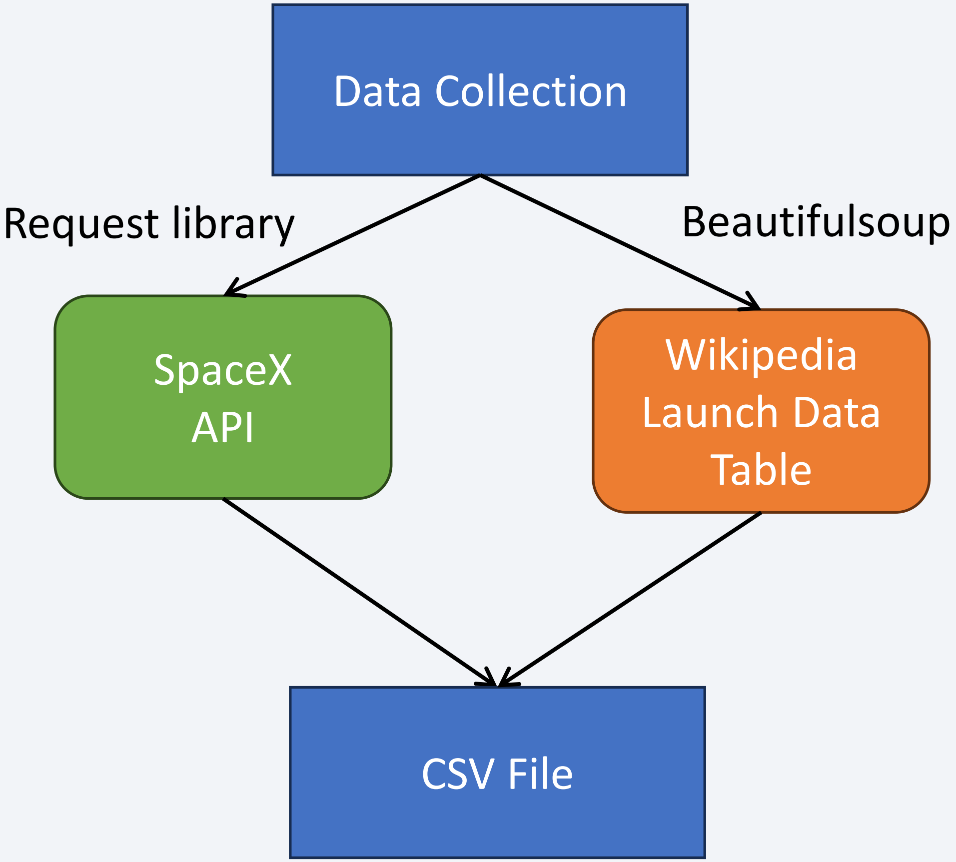
Data obtained: Place of launch, booster version, etc.

Wiki webscapping

Requesting wiki .json with requests

And scrapping with beautifulsoup

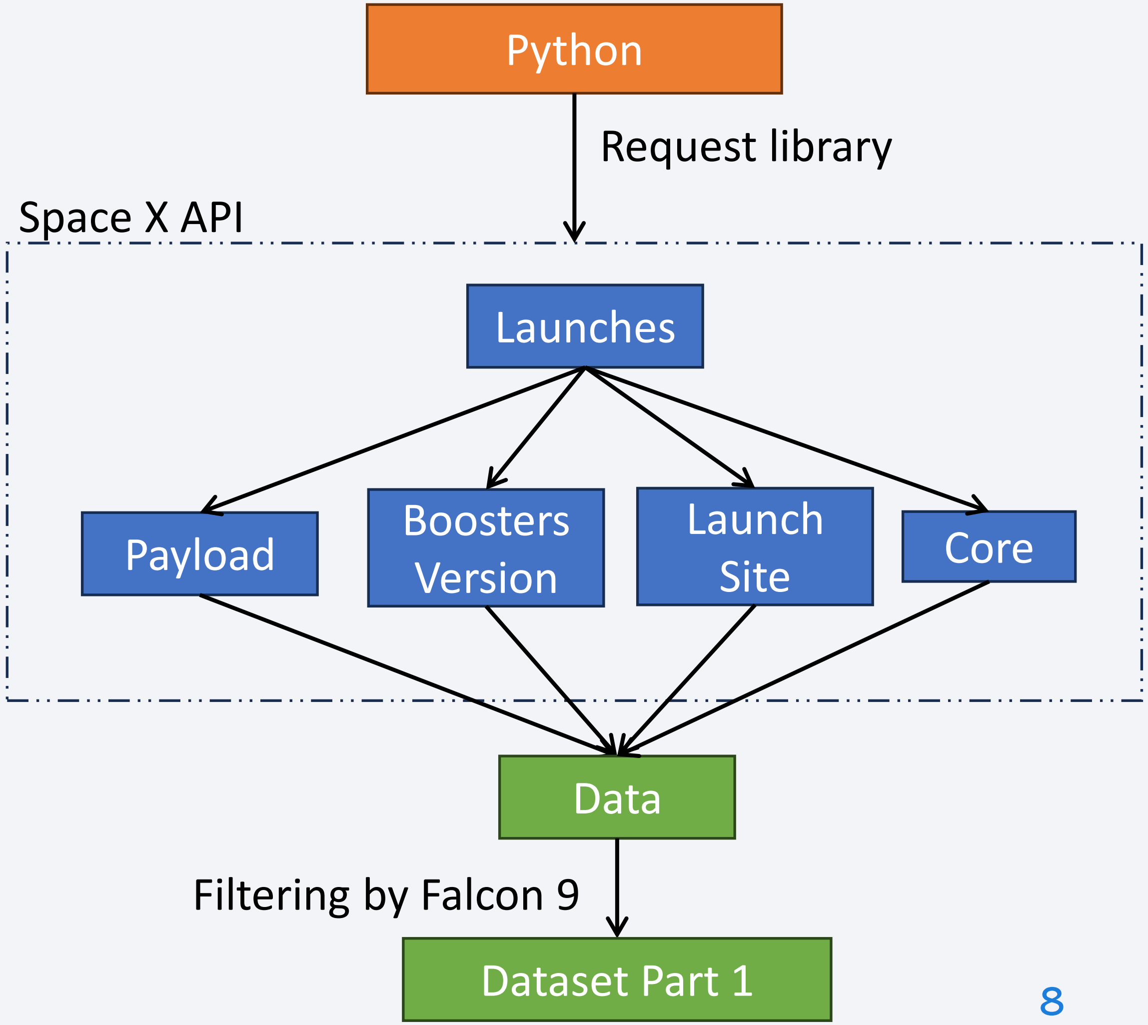
Data obtained: date, successful or not, ect.



Data Collection – SpaceX API

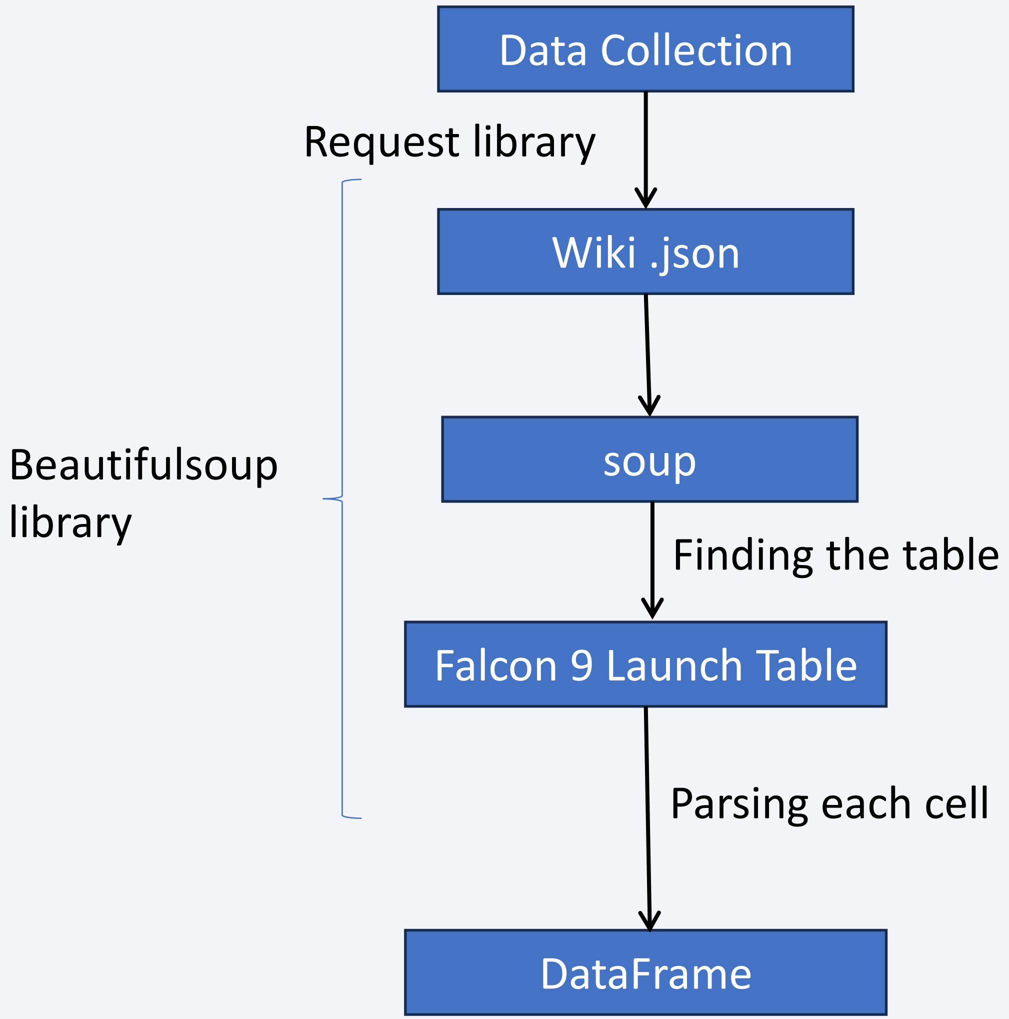
- Data collected in multiple sources references through ids

- https://github.com/RLarrubia/SpaceX_ML_project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- Adquireing the .json file from request to wiki web pag
- Creating a soup with BeautifulSoup library
- Parsing the content of Falcon 9 Launching Data into a DataFrame
- https://github.com/RLarrubia/SpaceX_ML_project/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- 1. Analysing missing values
- 2. Analysing Data Types
- 3. Studying occurrence of values such as orbit and launching sites
- 4. Creating a binary feature to be predicted for the outcome

- https://github.com/RLarrubia/SpaceX_ML_project/blob/main/labs-jupyter-spacex-Data%20wrangling-v2.ipynb

EDA with Data Visualization

- Charts used:
  - Scatter plots: to show relationship between to numerical variables and discover patterns
  - Bar plots: to show the success rate for different categories (launch sites)
  - Line plot: to represent the trend of success rate over time

- https://github.com/RLarrubia/SpaceX_ML_project/blob/main/jupyter-labs-eda-dataviz-v2.ipynb

EDA with SQL

- Converting the previous data into a database and accessing through sql in python with sql magic command
- Used to explore the data:
  - Retrieve Unique Launch Sites:
  - Filter Launches by Site Name Prefix ('CCA’)
  - Calculate Total Payload Mass
  - Identify First Successful Ground Pad Landing
  - Rank Landing Outcomes Between Two Dates

https://github.com/RLarrubia/SpaceX_ML_project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Data displayed on maps to see visually:

- **Markers** for launches in each location
- **Circles** to mark launching sites
- **Lines** for distances between launching sites and and relevant points

- https://github.com/RLarrubia/SpaceX_ML_project/blob/main/lab-jupyter-launch-site-location-v2.ipynb

Build a Dashboard with Plotly Dash

- Pie charts:
    - To show the percentage of flight in each location site
    - The successful rate in each location
- Scatter plots:
    - To show success rate with the mass of the launcher and with the booster version for different payload ranges

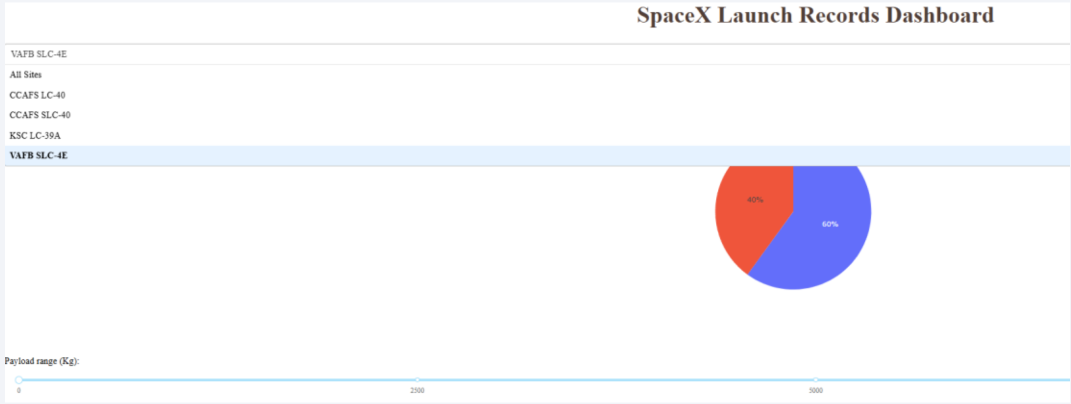
https://github.com/RLarrubia/SpaceX_ML_project/blob/main/spacex_dash_app.py

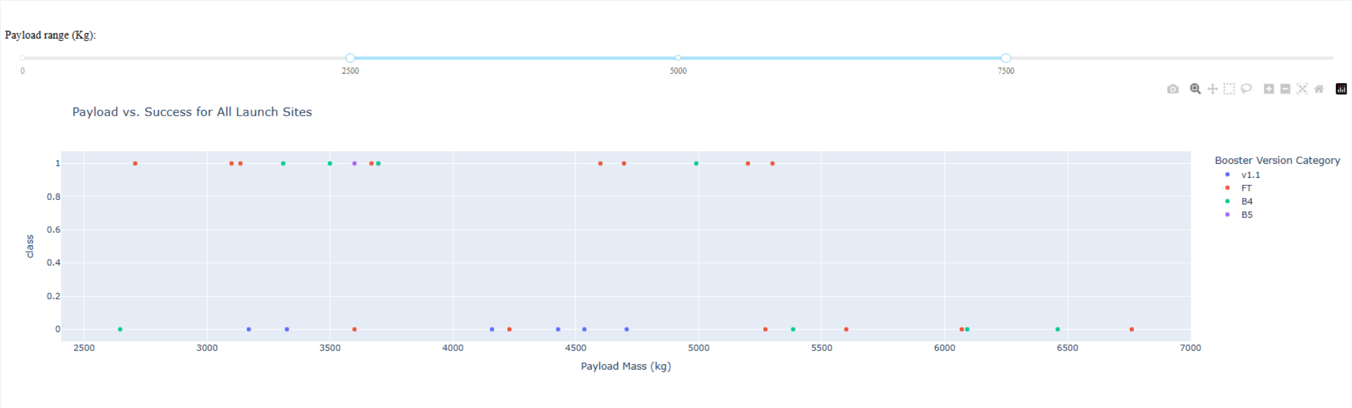
Predictive Analysis (Classification)

- Using ML classification models to predict the outcome of a launch given a series of properties of the launcher and the mission
- First splitting 80% of the data for train
- Using GridShearch to tune the hyperparameters with crossvalidation in 10 folds
- Tried different models:
  - Logistic regression
  - SVM
  - Decision Trees
  - KNN
- Using accuracy to determine best model
- https://github.com/RLarrubia/SpaceX_ML_project/blob/main/SpaceX-Machine-Learning-Prediction-Part-5-v1.ipynb

Results

- Most landings occur in Cabe Canaveral Launch Site with more success rate
- All models tried and tuned achieved similar results of accuracy over the test set around 83%

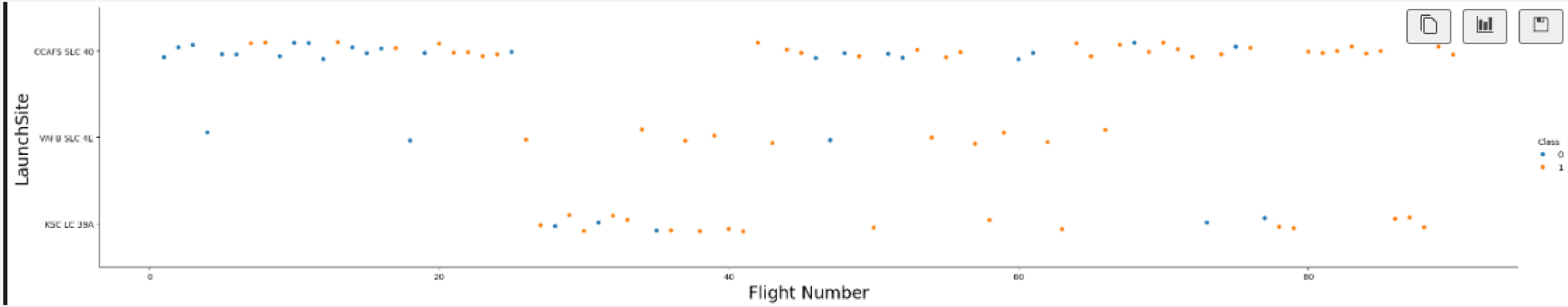




Section 2

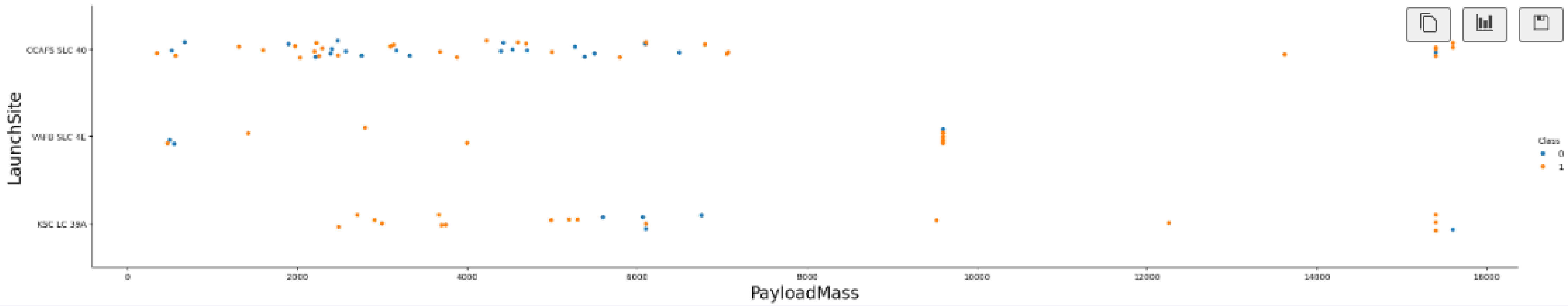
Insights drawn from EDA

Flight Number vs. Launch Site



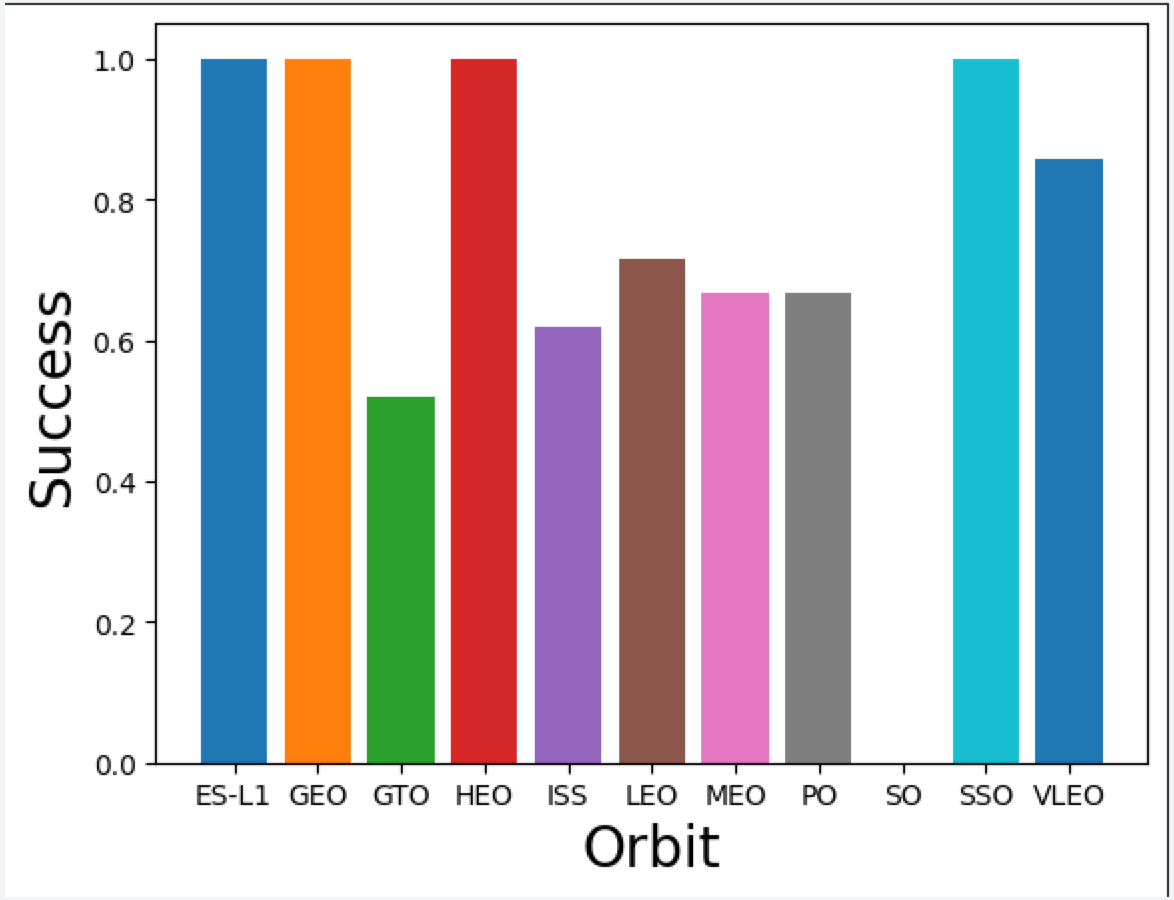
- Most in Cabe Canaberal with more success rate
- Rest less launches and less success rate, similar between them

Payload vs. Launch Site



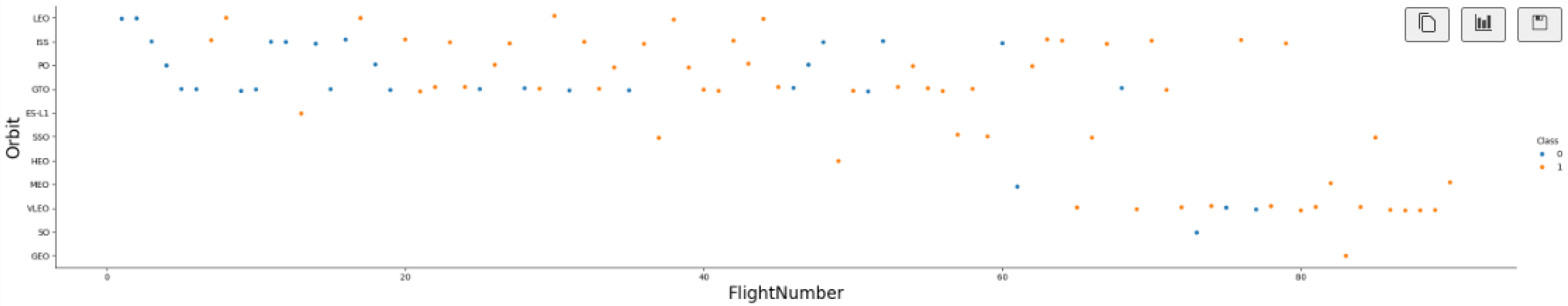
- No heavy launches on VAFB
- Canaveral Cape has a significant launches for light launches with diverse success rate
- KSC has high success rate for lightweight launches

Success Rate vs. Orbit Type



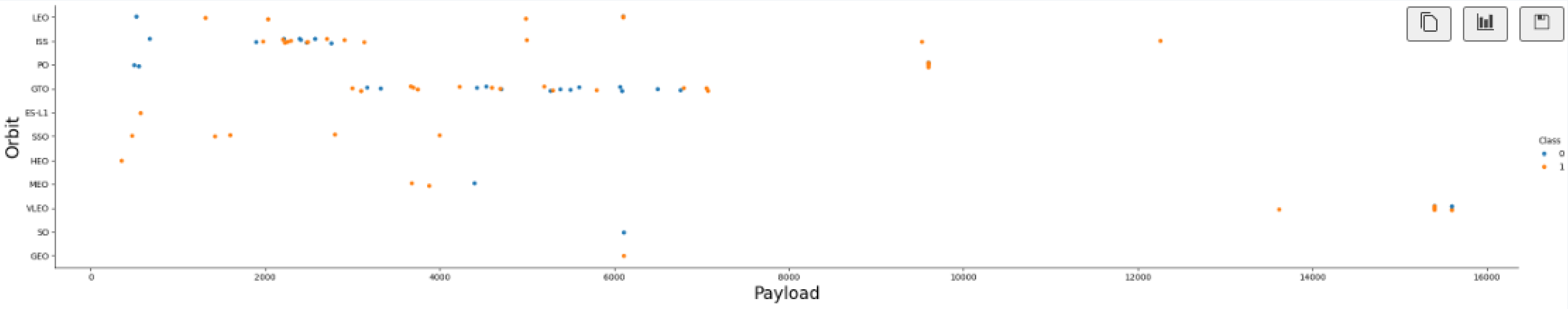
- High Success rate in ES-L1, GEO, HEO, SSO and VLEO

Flight Number vs. Orbit Type



- LEO orbit gets higher success rate with time (Flight Number)
- Others like GTO do not show a clear pattern

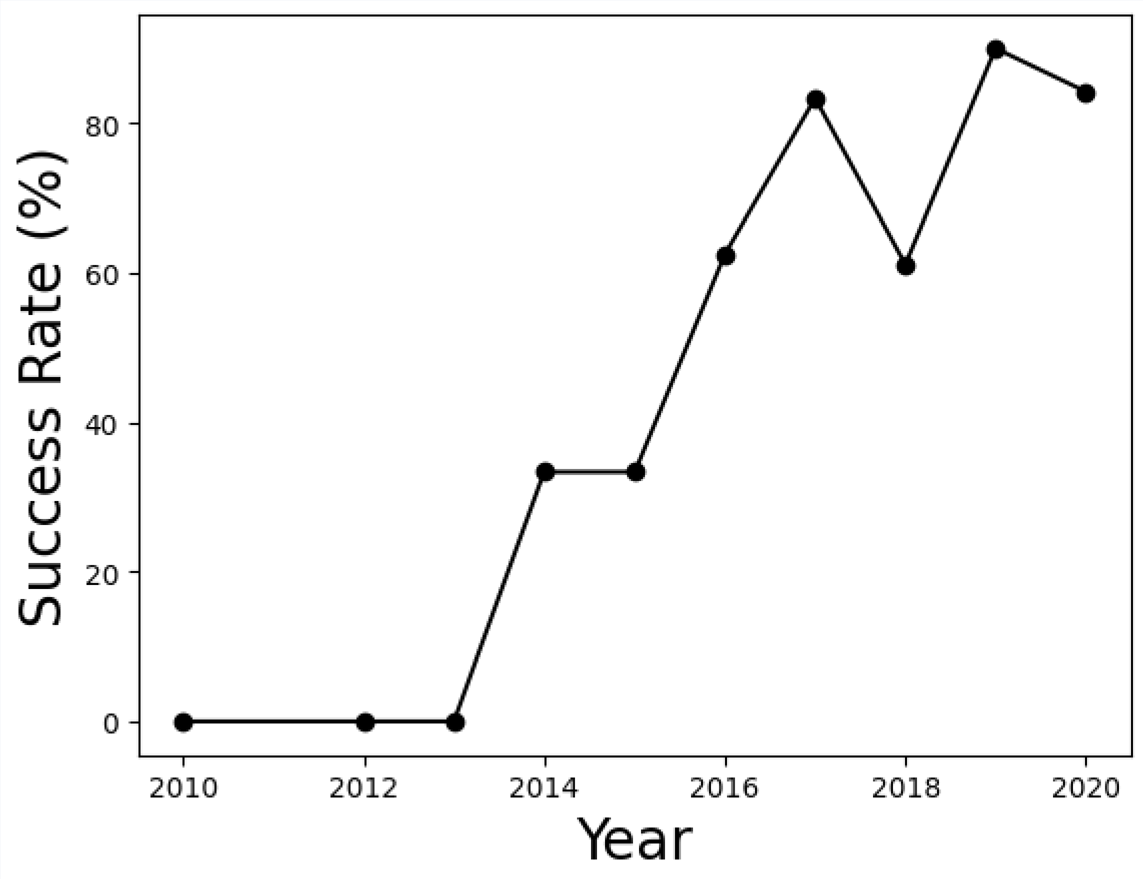
Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- For others there is not a clear pattern

Launch Success Yearly Trend

- The success rate has improved significantly over time
- Not having achived any success before 2013



All Launch Site Names

Codes for the different launch sites:

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

Launch Site Names Begin with 'CCA'

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

Total Payload Mass

Total_Payload
619967

Average Payload Mass by F9 v1.1

Total_Payload_Mass_F9
2534.6666666666665

First Successful Ground Landing Date

First_Successful_Ground_Pad_Landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 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

Mission_Outcome	Total
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

Month_Name	Landing_Outcome	Booster_Version	Launch_Site
January	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

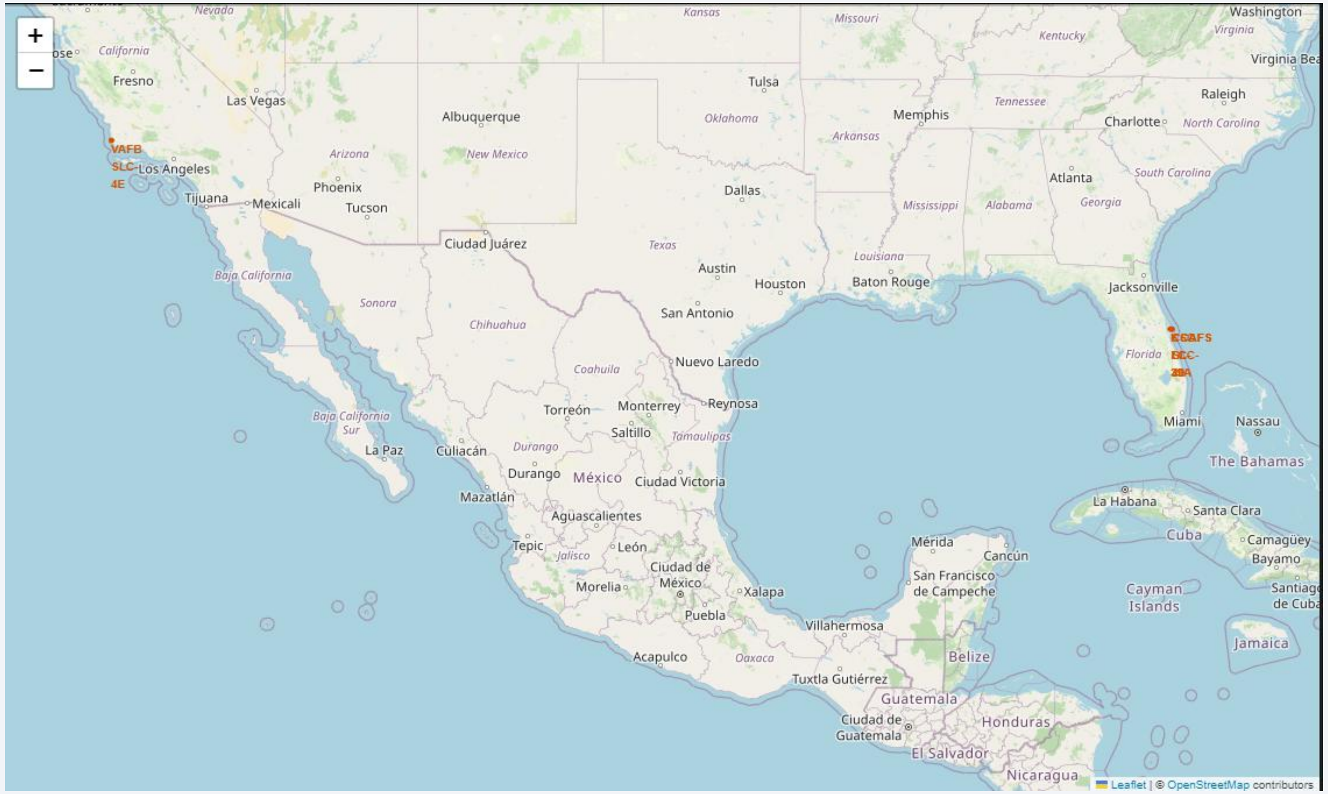
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

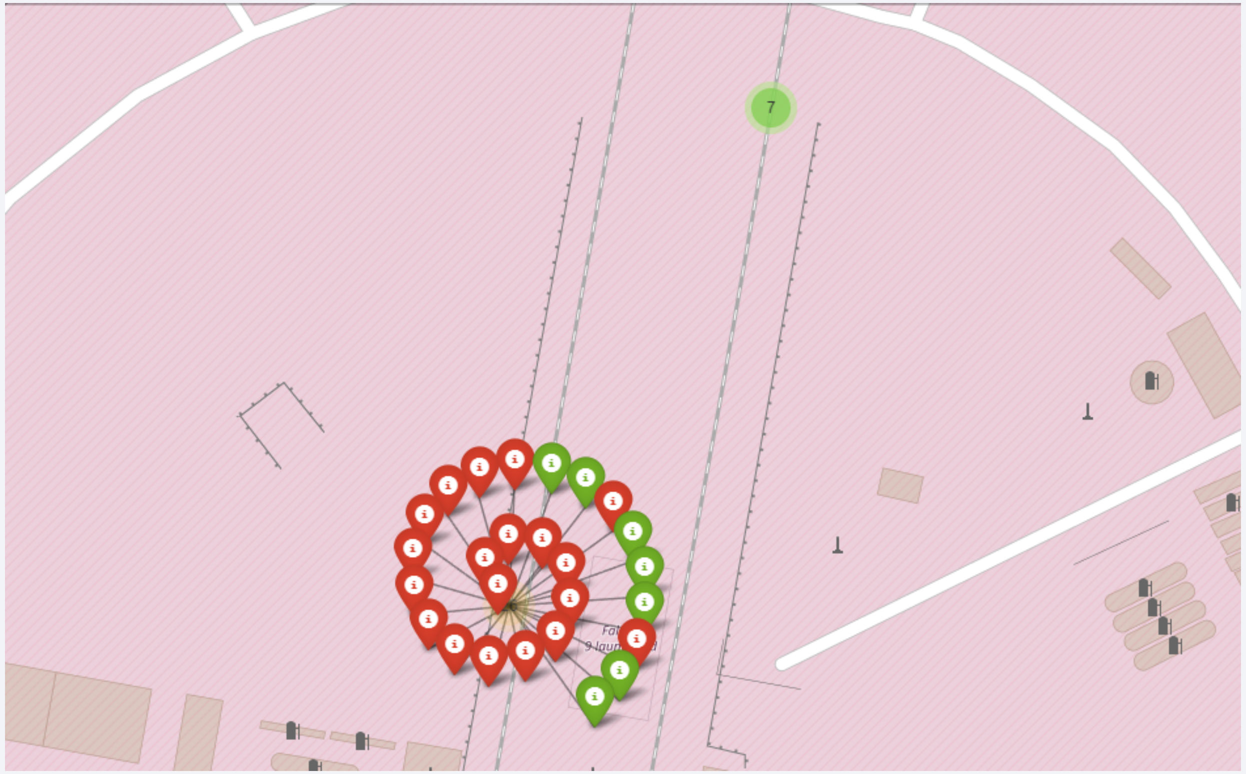
Location of the different launch sites

- 3 in Florida
- 1 in California
- Close to the equator (in the US)
- Close to the coast
- In both coast for different orbits



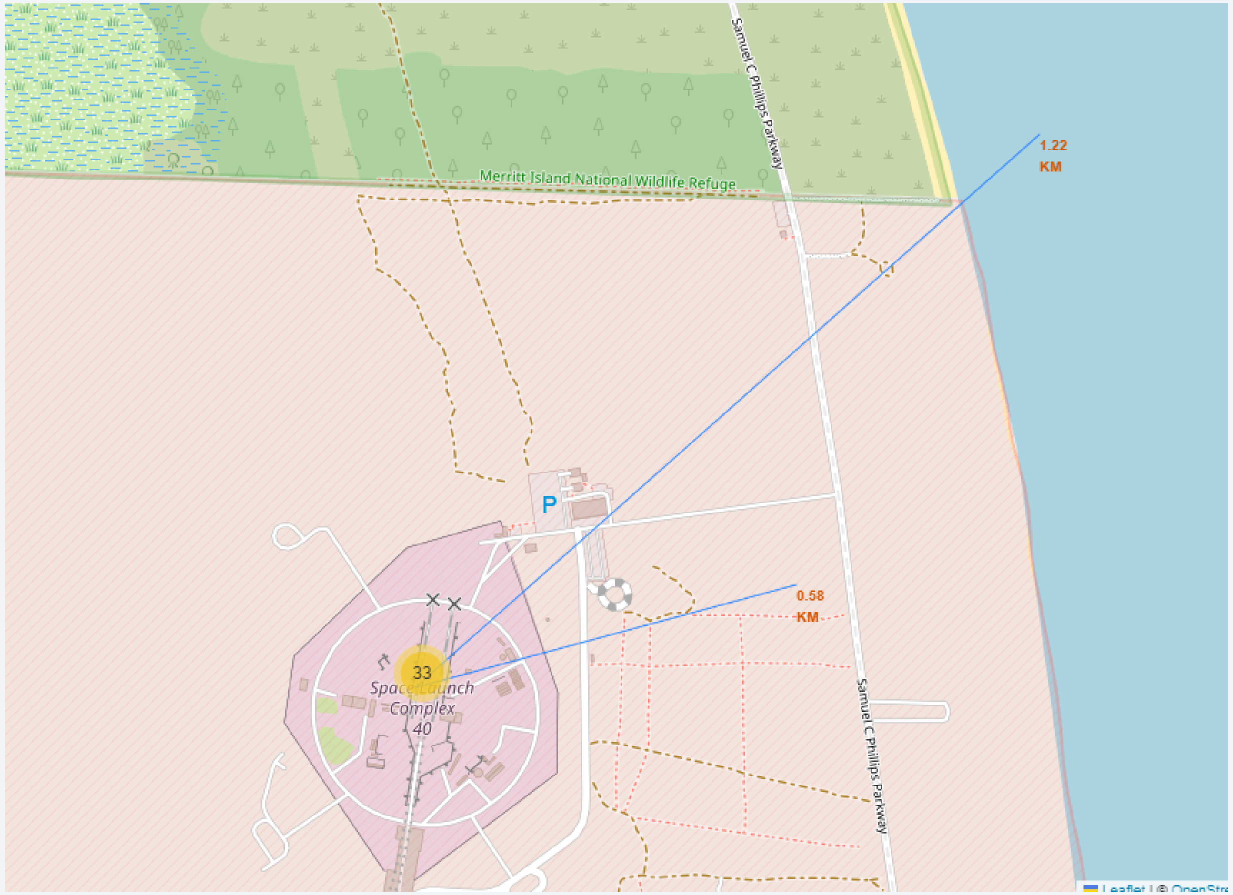
Successful missions in Canaveral Cape

- Successful landings in green
- Unscucessful in red
- Clusters grouping launches



Coast proximity to Canaveral Cape

- Canaveral Cape around 1km from the coastline

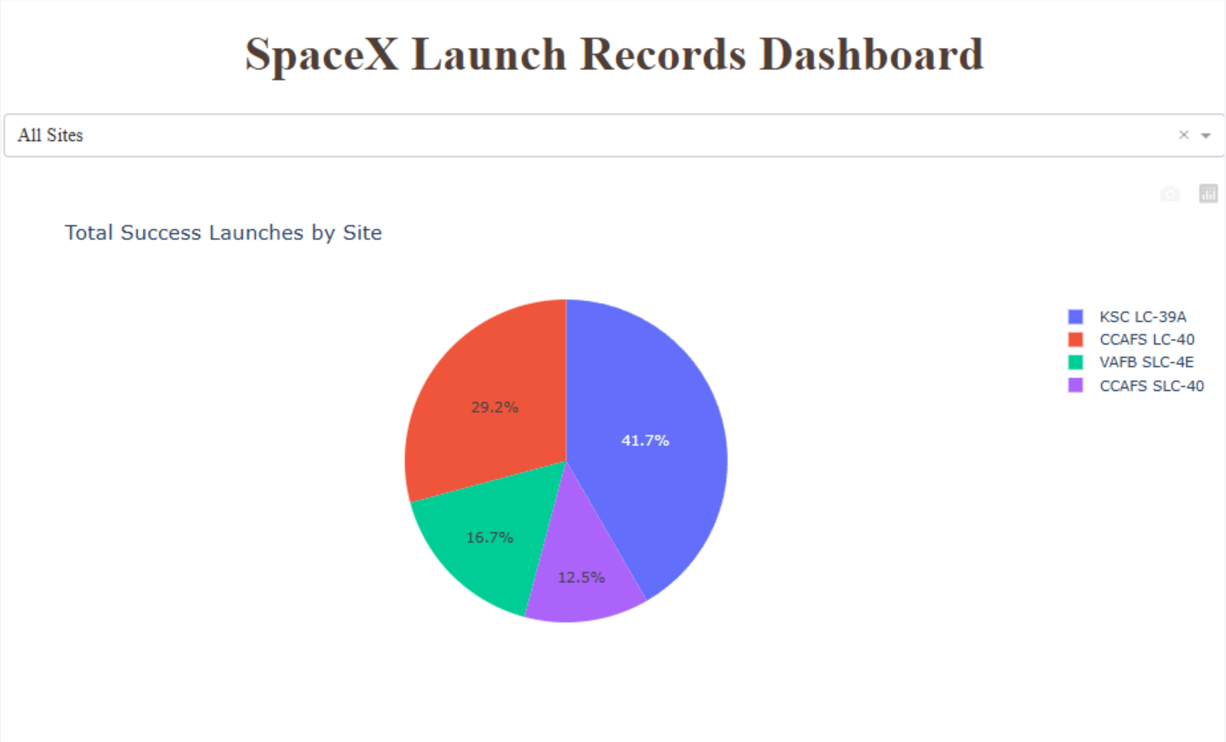


Section 4

Build a Dashboard with Plotly Dash

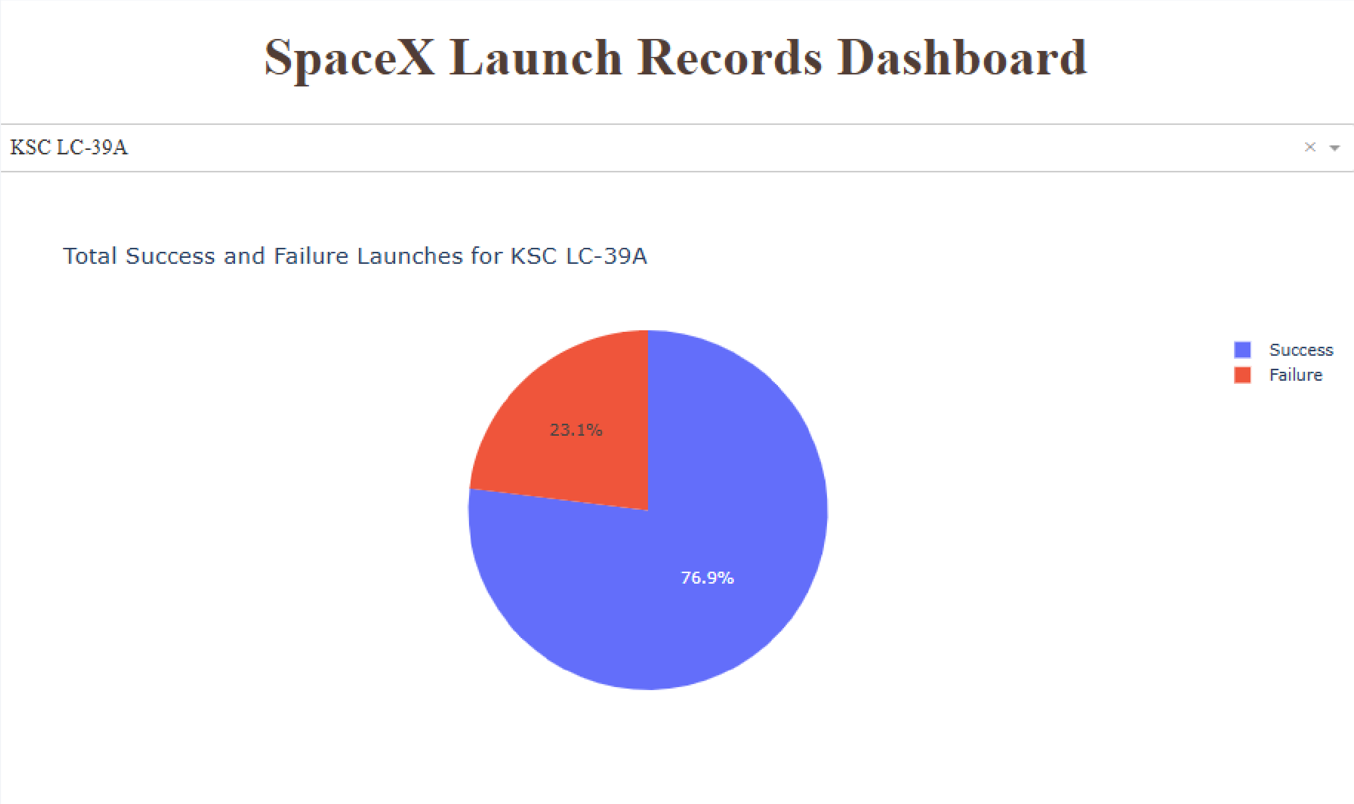
Total Success Launches by Site

- Canaveral Cape has the highest success landing returns rates in its 3 facilities
- Highest in KSC LC 39A

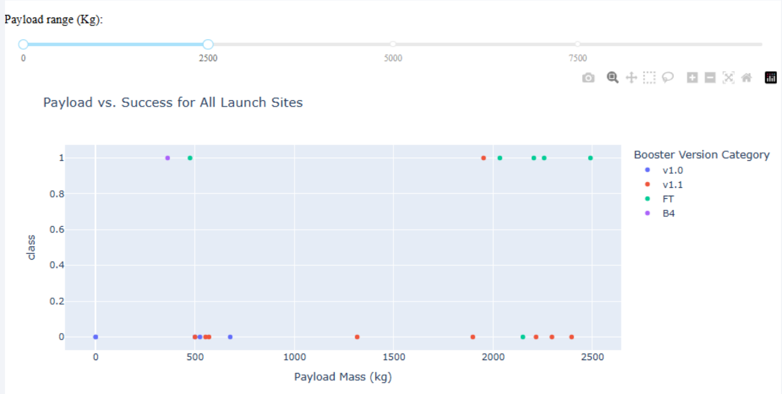


Success distribution in KSC LC-39A

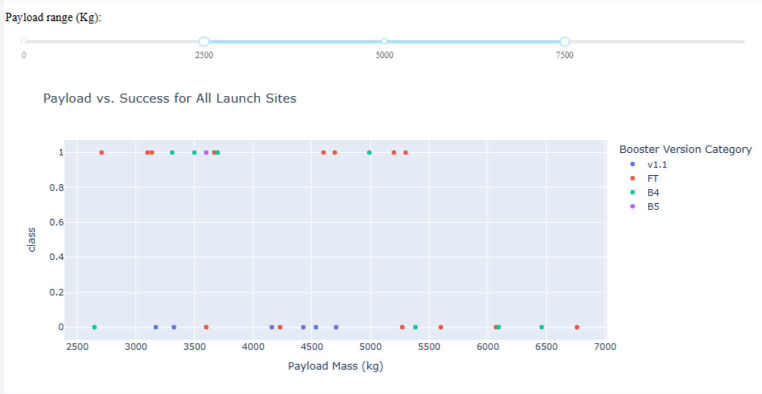
- Around 80% of success rate



Successful returns with payload



Lightweight launches



Medium weight launches

Heavyweight launches

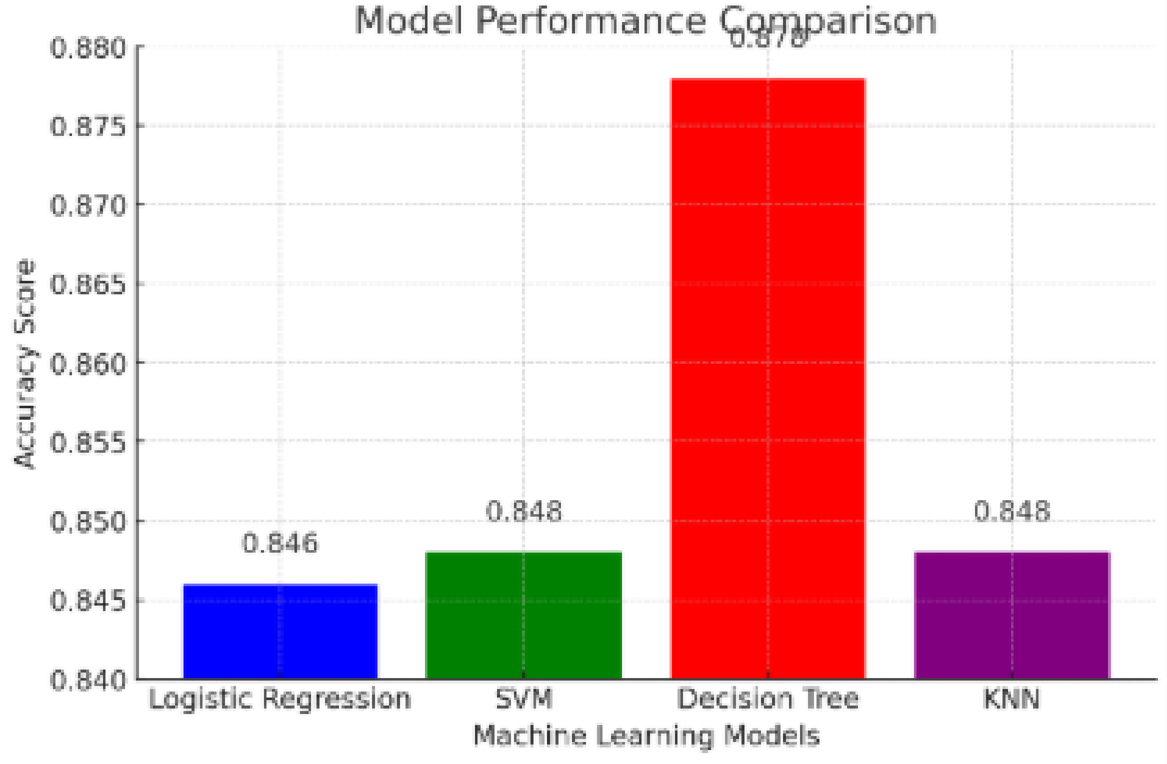
- Light and medium weight launches are more common
- Heavy weight can only be performed with a few boosters
- Correlation is unclear between weight and successful return of booster stage

Section 5

Predictive Analysis (Classification)

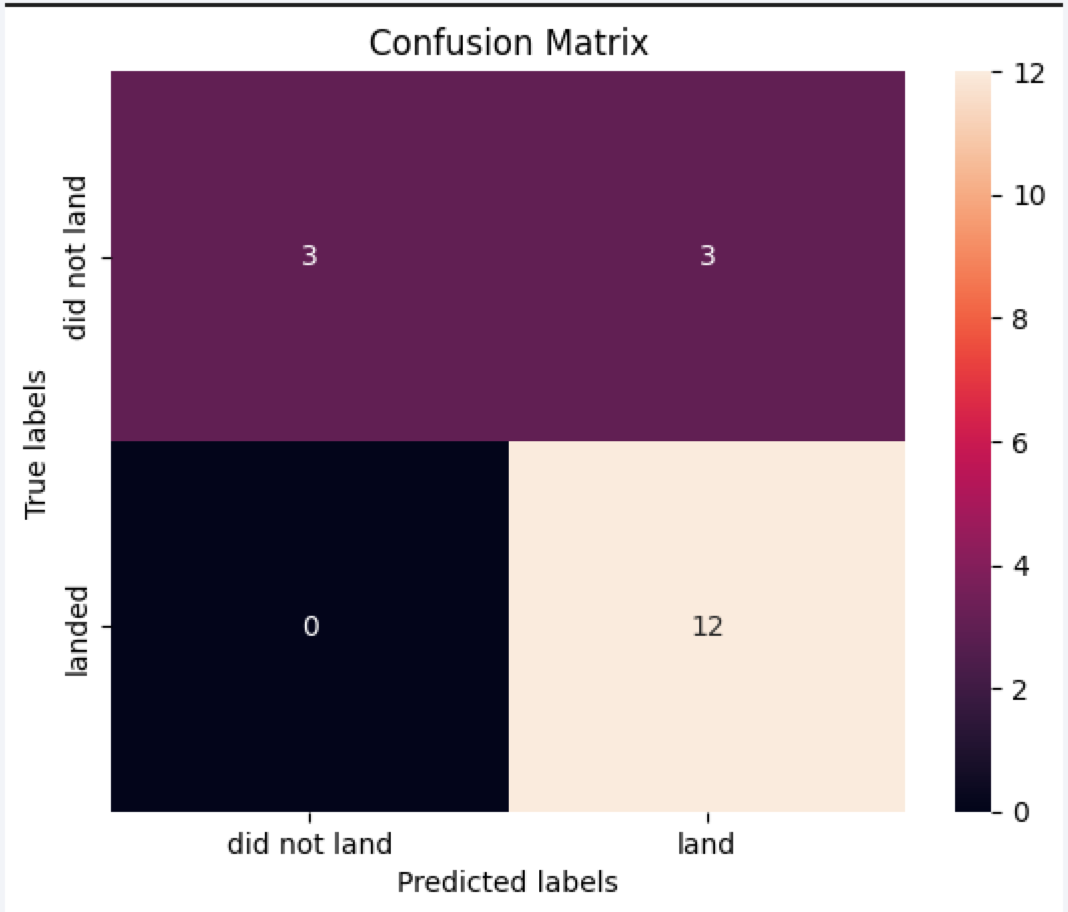
Classification Accuracy

- Best model Decision tree with score of 83% on test set and 88% on test set
- All models achieved the same accuracy on train set 83%



Confusion Matrix

- 3 True negatives
- 12 True positives
- 3 False positives
- 0 False negatives



Conclusions

- Florida Launching sites are more popular probably due to the fact that most orbit are prograde and launching from the east benefits earths rotation
- East side launches are more probable to be returned
- ML classification models can be built to predict the return of the booster with an accuracy of 83%

Appendix

- import matplotlib.pyplot as plt
- models = ['Logistic Regression', 'SVM', 'Decision Tree', 'KNN’]
- scores = [0.846, 0.848, 0.878, 0.848] # Create bar chart plt.figure(figsize=(8,5))
- plt.bar(models, scores, color=['blue', 'green', 'red', 'purple’])
- plt.xlabel("Machine Learning Models")
- plt.ylabel("Accuracy Score")
- plt.title("Model Performance Comparison")
- plt.ylim(0.84, 0.88) # Adjust y-axis range for better visibility
- for i, v in enumerate(scores): plt.text(i, v + 0.002, str(v), ha='center', fontsize=12)
- plt.show()

Thank you!