IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Kush Neal Bansal
01/09/2024

Outline

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

Executive Summary

- Using SpaceX data to determine cost of rockets
- Using Data analysis and Machine Learning to analyze and predict success factors for SpaceX launches and First Stage recovery
- Orbit, Launch Site, and Payload Mass are among the top variables that affect mission success and First Stage recovery
- Machine Learning models can predict success at 83% accuracy

Introduction – So you want to go to space

The Set-Up

- Space travel is very expensive ~$165M per launch
- SpaceX – recover the first stage → brings down cost to $65M
- Not 100% successful, but what if we could decrease the risk of failure?

The Problem

- What data can we pull from previous SpaceX launches to increase the probability of success?
- What factors can lead to failure? What algorithms can we use to predict success?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- 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

SpaceX REST API Calls

- Performed GET-Response requests to get various rocket data and flight outcomes
- Built functions to translate IDs to names of locations, boosters, etc.
- Normalized the data into a flat table and dealt with missing values
- Filtered for Falcon 9 launches

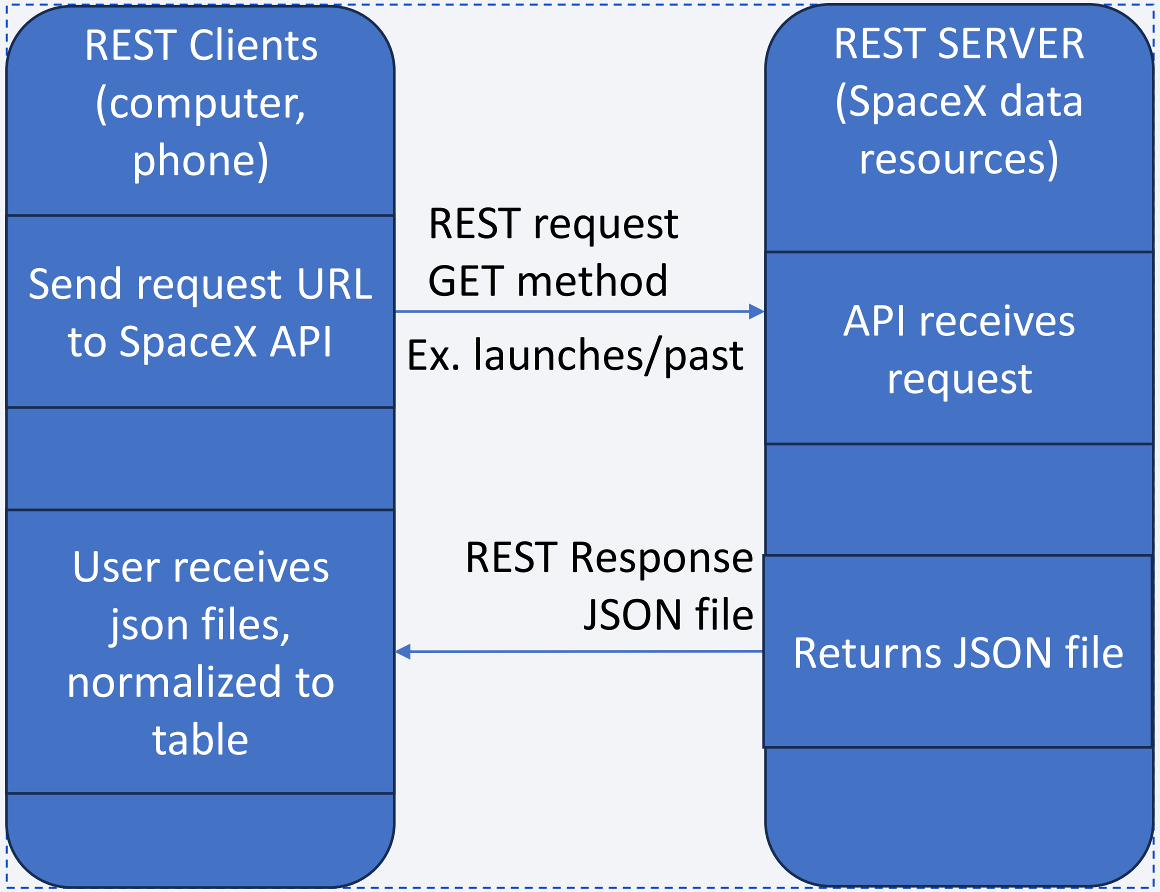
Web Scraping

- Performed GET response for Wikipedica page on Falcon 9 launches
- Parsed text with BeautifulSoup to extract columns from table headers
- Extracted data from table rows to build a Pandas DataFrame

Data Collection – SpaceX API

Data Received

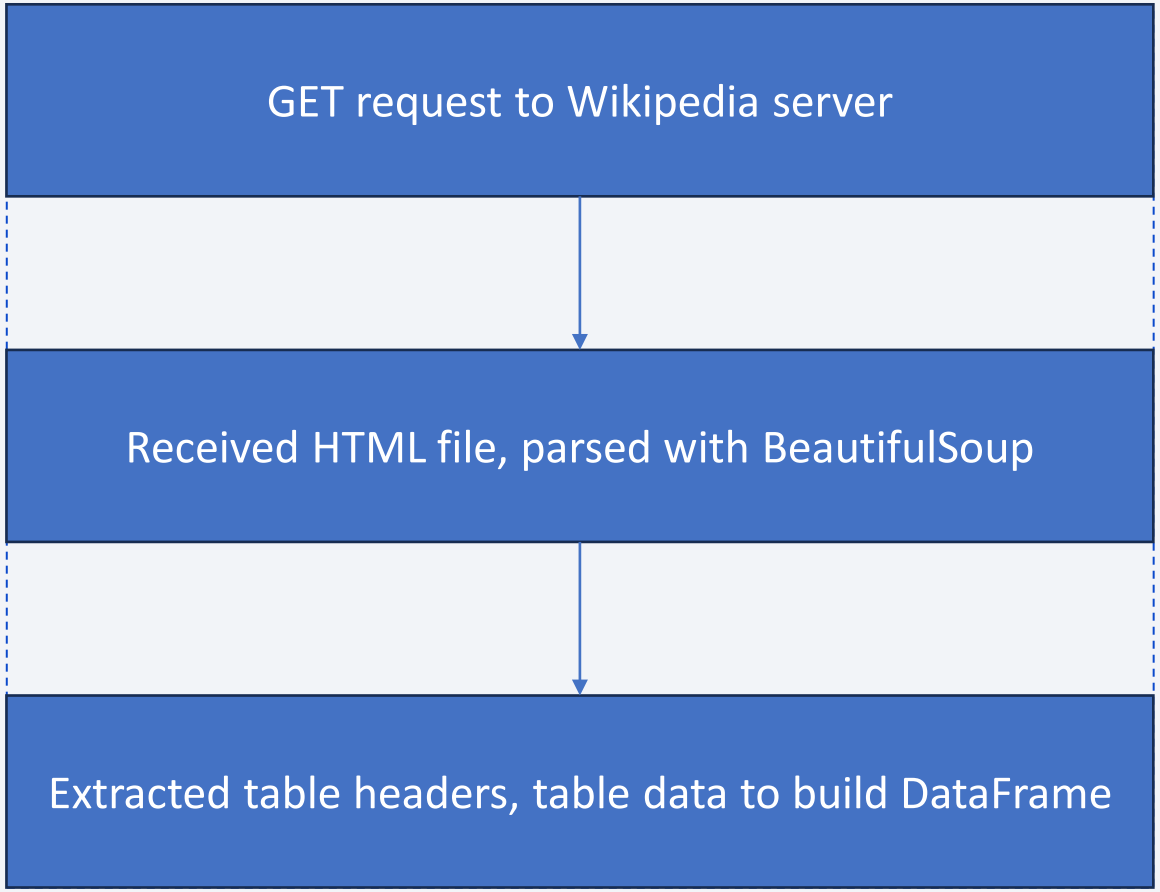
- Rocket Data
  - Booster Version
- Payload
  - Mass, Orbit
- Launchpad
  - Launch Site, Latitude, Longitude
- Cores
  - Outcomes, type of landing, flights with core, number of reuses, gridfins, legs, landing pad



Data Collection - Scraping

Data Received

- Launch Site
- Payload
- Payload Mass
- Orbit
- Customer
- Launch Outcome
- Version Booster
- Booster Landing
- Date and Time



Data Wrangling

Created landing outcome label to track which launches landed successfully → determined success rate of 67%

Landing Outcome	Meaning
True ASDS	Successful (drone ship)
False ASDS	Failure (drone ship)
True RTLS	Successful (ground pad)
False RTLS	Failure (ground pad)
True Ocean	Successful (ocean)
False Ocean	Failure (ocean)
None None	Failure to land
None ASDS	Failure to land

Launch Sites	Names
CCAFS SLC 40	Cape Canaveral Space
KSC LC 39A	Kennedy Space Center
VAFB SLC 4E	Vandenberg Air Force Base

EDA with Data Visualization

- Charts built primarily generated correlation between:
  - Launch Site, Flight Number, Payload Mass, Orbit
- Additional charts show:
  - Success Rate over Time
  - Success Rate by Orbit

- Correlation Charts
  - How do two variables relate
  - Generates visible trends
- Success Rate Graphs
  - Visualize what has worked and what has not

EDA with SQL

SQL queries are more readable and understandable

Can provide data analysis and summaries

Query Summary

- Unique Launch Sites
- Cape Canaveral records
- Payloads by NASA
- Booster F9 v1.1 Payload
- First successful ground pad landing
- Boosters with drone ship success
- Mission Outcome Statistics
- Boosters that carried Max payloads
- 2015 Drone Ship landing failures
- Ranked landing outcomes (6/4/2010 – 3/20/2017)

Build an Interactive Map with Folium

Folium allows users to visualize geospatial data

Folium Elements

- Circles – provides a circle of where the coordinates are
- Marker – provides a pin with name of location
- MarkerCluster – provides number of markers in one area if they are close together

Build a Dashboard with Plotly Dash

Plotly Dash allows you to build shareable, interactive dashboards

- Dropdown menu to select all launch sites and drill down to specific launch site
- Pie Chart
  - All Landing Sites – provides success rate of each landing site from total successes
  - Filtered landing site – provides percent of success and failures at specific landing site
- Payload Mass vs Launch Outcome Correlation
  - Provides booster versions that are interactable
  - Provides Payload slider to adjust Payload Mass
  - Filters to specific landing site when dashboard is changed

Predictive Analysis (Classification)

- Classification Methods
  - Logistic Regression – similar to linear regression; used for discrete target field to create decision boundaries
  - Support Vector Machines – classifies by finding a separator
  - Decision Trees – map out possible decision paths and probabilities
  - K-Nearest Neighbor – classifies data points based on similarity to other data points
- Refinement with hyperparameters
  - GridSearchCV – uses different combinations of hyperparameters to train and evaluate model
- Evaluation
  - Accuracy Score – percent of labels that exactly match
  - Confusion Matrix – matrix that shows what the model guessed correctly and incorrectly

Train-Test-Split: Split the data into training and testing data

Training – Train each model on the training data

Test – Test how well the model predicts the values

Evaluate – use statistical scores to determine accuracy and compare models

Section 2

Insights drawn from EDA

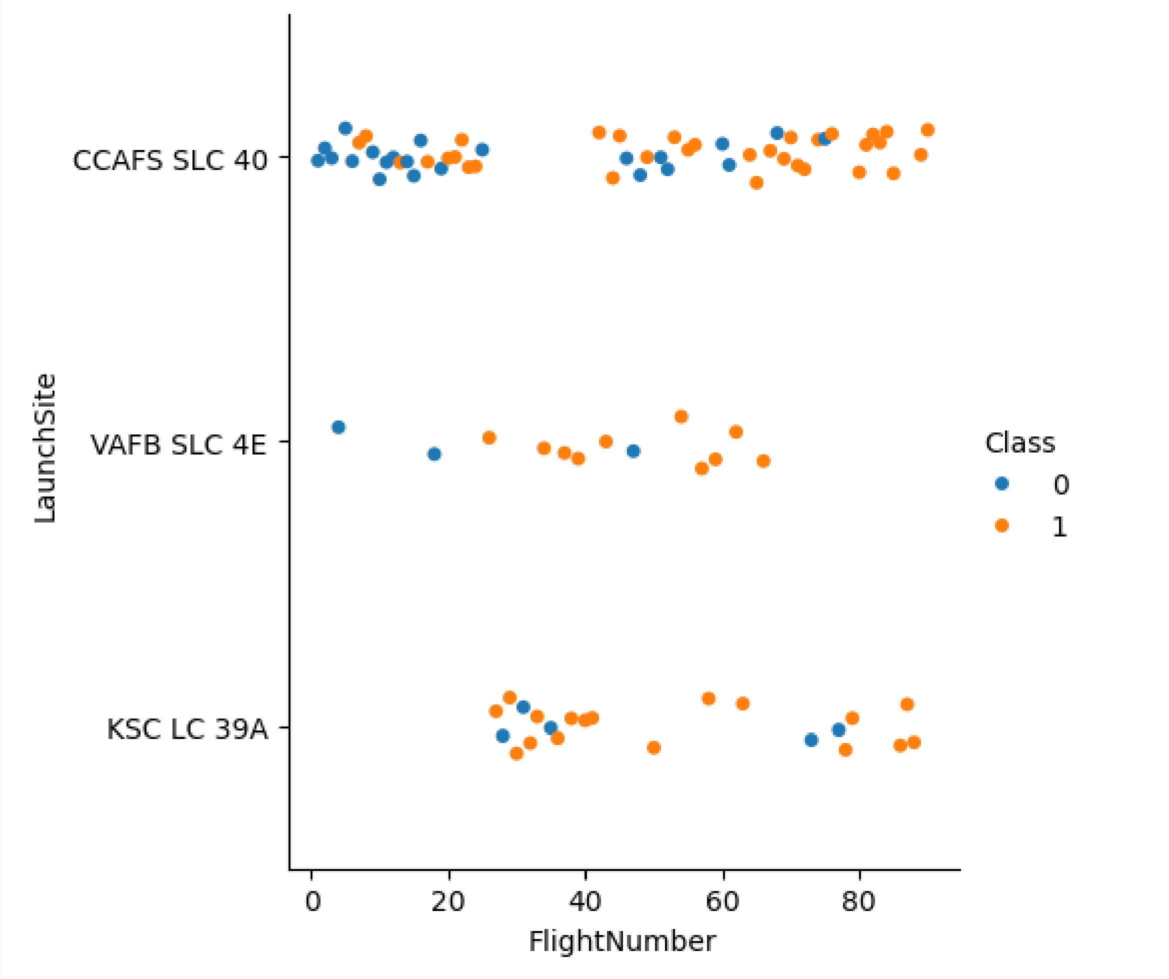
Flight Number vs. Launch Site

Correlates flight number and launch site

Class represents landing success (1) and landing failure (0)

Conclusions

- Higher success rate the higher the flight number
- Cape Canaveral is the most popular launch site

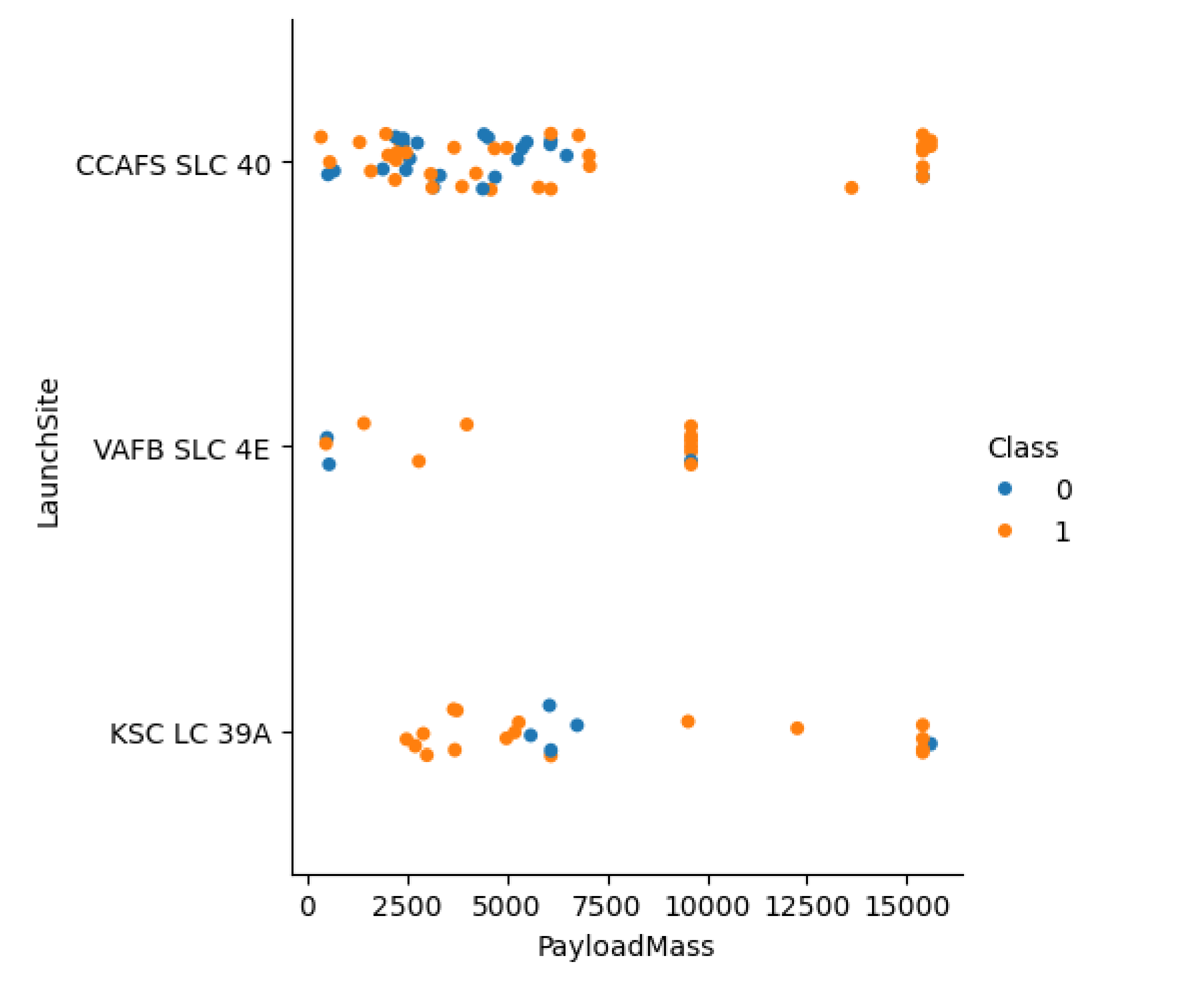


Payload vs. Launch Site

Correlates Payload Mass to Launch Site

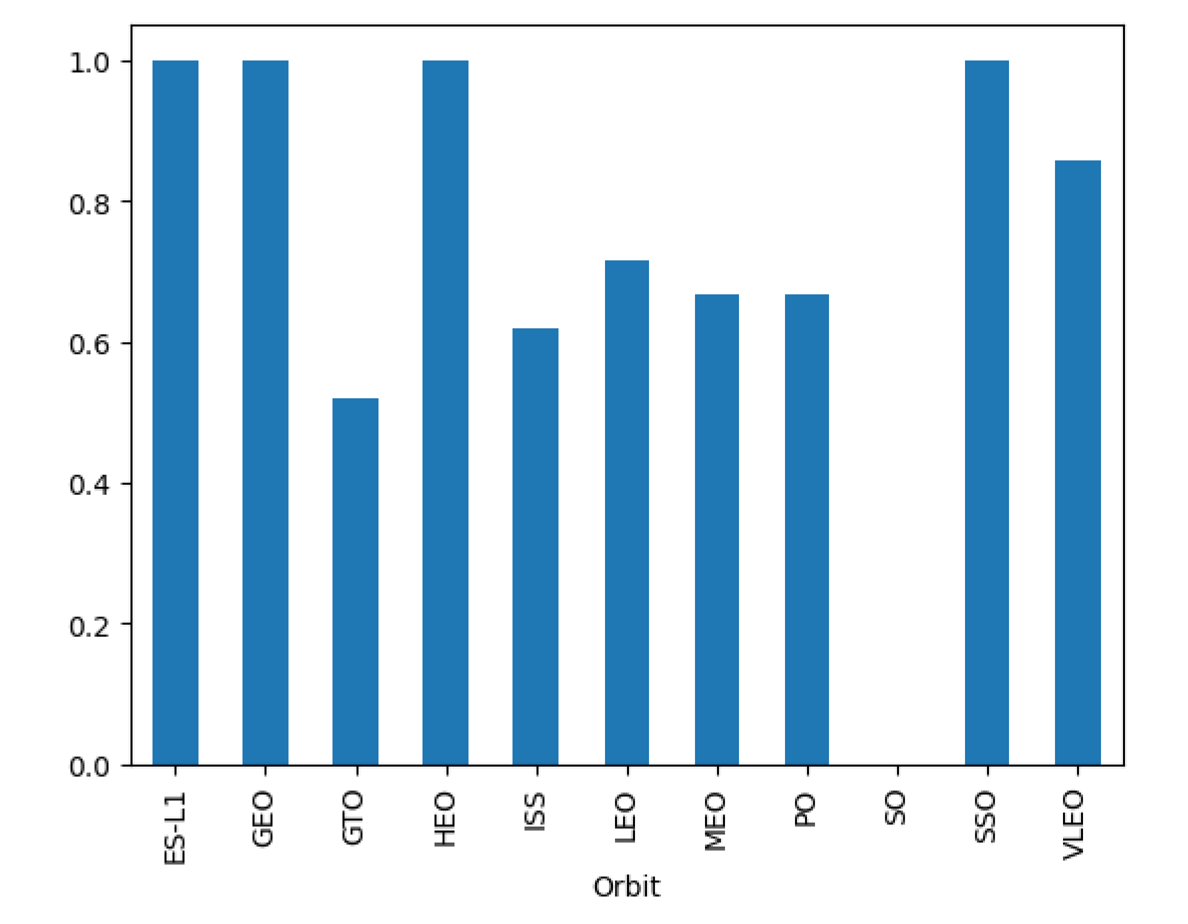
Conclusions

- Cape Canaveral launches with at least 12,500kg payloads tend to succeed
- Vandenberg does not exceed 10,000kg payloads
- Vandenberg launches with 10,000kg payloads tend to succeed



Success Rate vs. Orbit Type

Orbit	Explanation
ES-L1	Lagrange Points
GEO	Geosynchronous Orbit
GTO	Geostationary Orbit
HEO	Highly Eliptical Orbit
ISS	Space Station
LEO	Low Earth Orbit
MEO	Medium Earth Orbit
PO	Polar Orbit
SO	Solar Orbit
SSO	Sun-synchronous Orbit
VLEO	Very Low Earth Orbit



Conclusions

Average Success Rate by Orbit Type

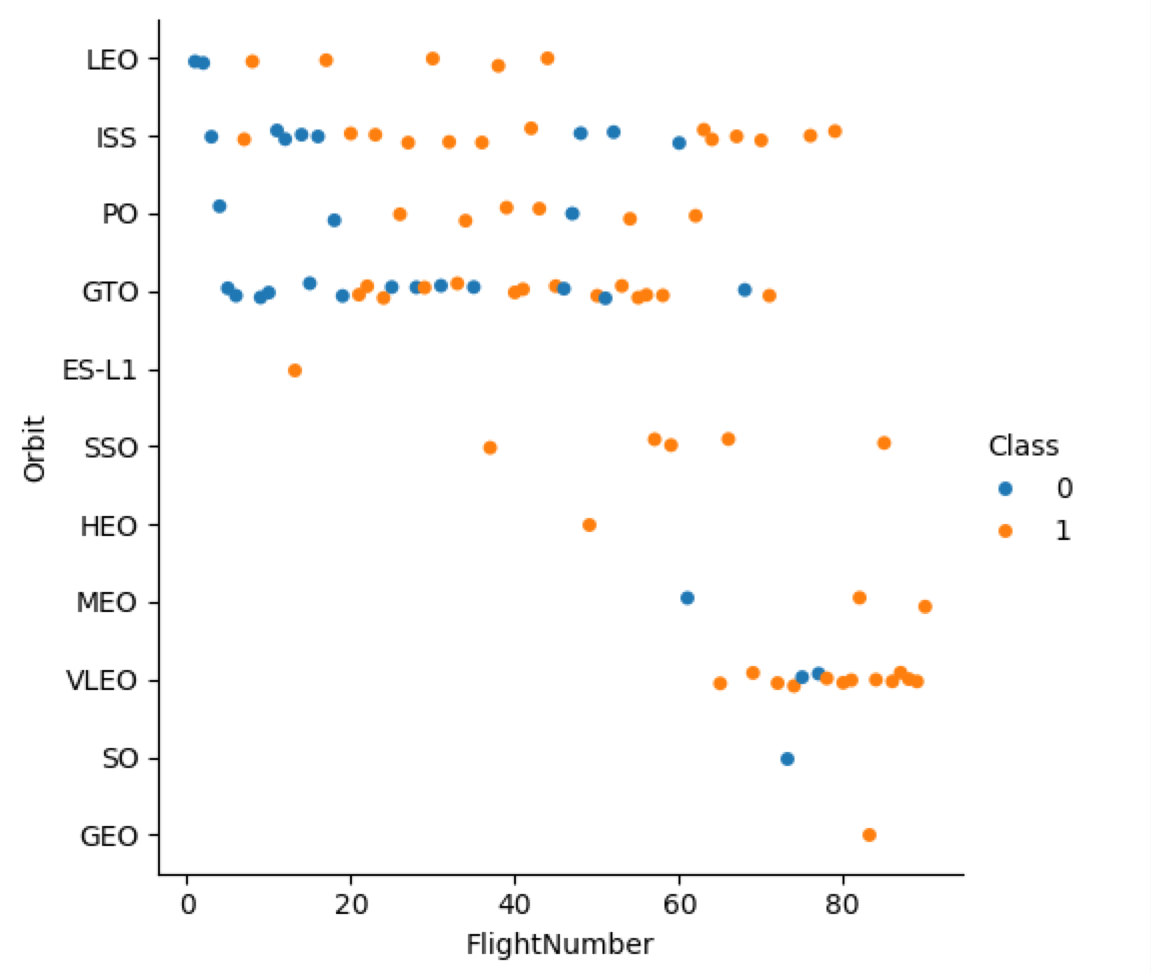
ES-L1, GEO, HEO, and SSO are successful

Flight Number vs. Orbit Type

Correlates flight number and orbit type

Conclusions

- Fewer failures as flight number increases
- VLEO are more popular

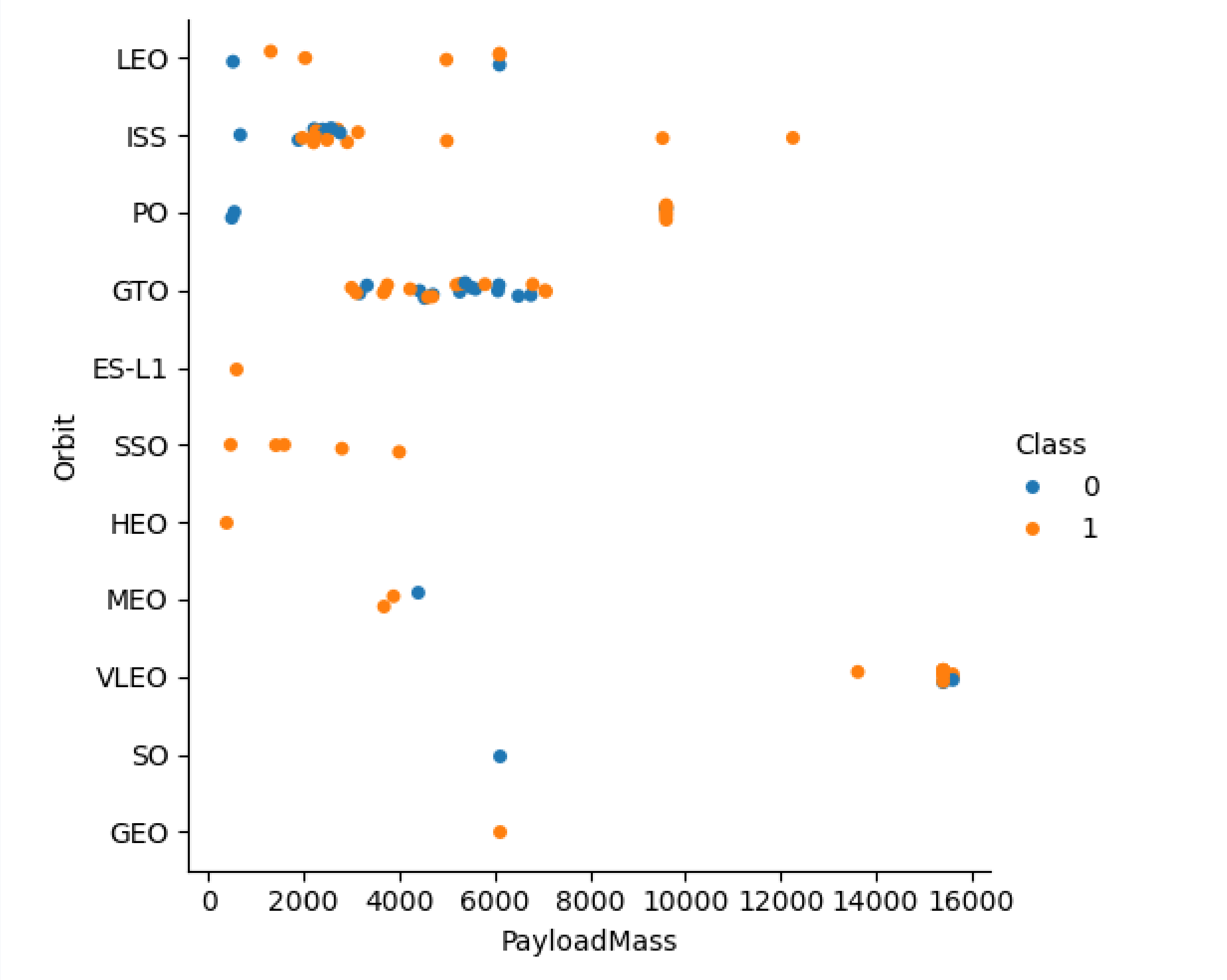


Payload vs. Orbit Type

Correlation between payload mass and orbit type

Conclusions

- VLEO payloads are greater than 13,000kg
- GTO payloads are between 3,000 and 8,000kg
- SSO payloads are less than 6,000kg

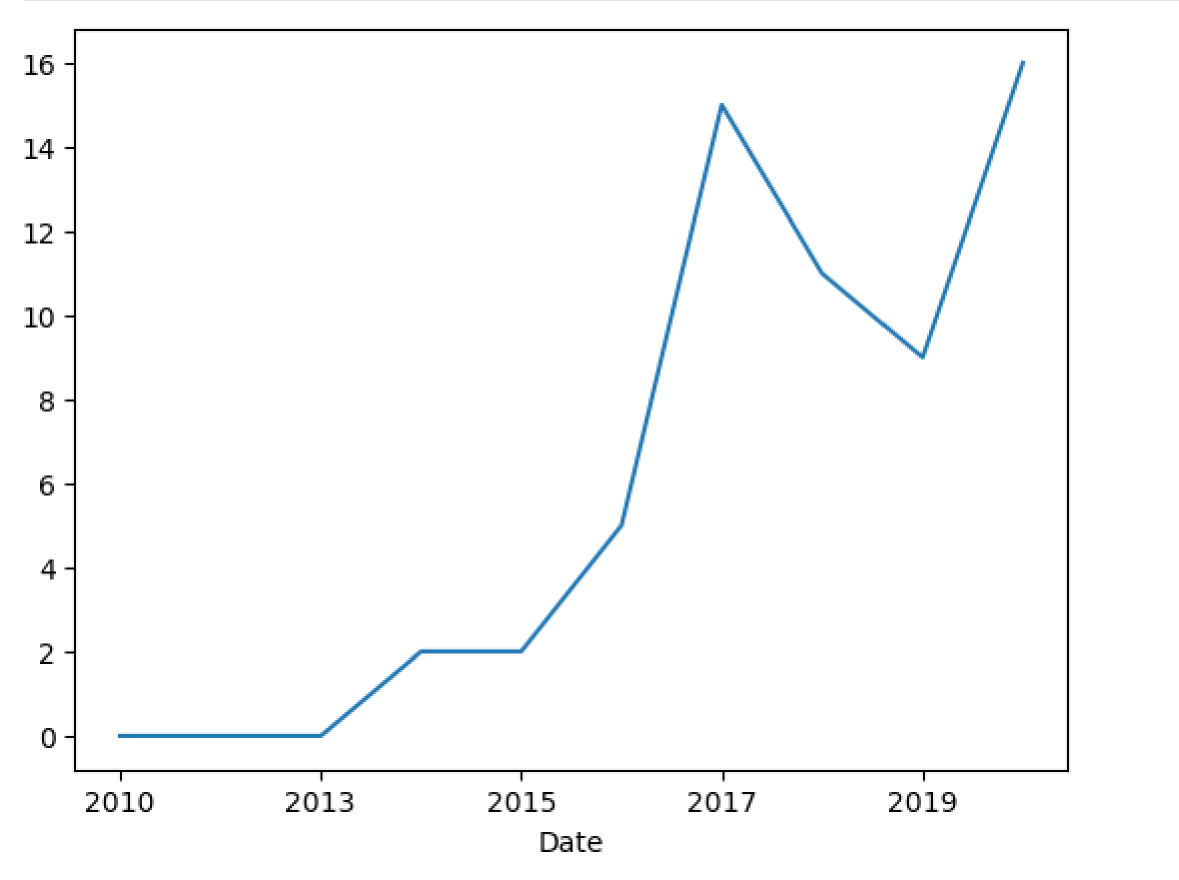


Launch Success Yearly Trend

Line chart of launch success over time

Conclusions

- Exponentially growing success
- Slight dip from 2017-2019 but could recover



Common SQL Terminology

Query – a line of code that does something to a table

%sql – a command that allows user to execute SQL queries in Python

SELECT – picks columns from the data table

FROM – data table you are pulling from (in this case, SPACEXTABLE)

WHERE – filters data table

GROUP BY – group like rows by a specific column

ORDER BY – orders final table by a specific column (DESC or ASC)

All Launch Site Names

Query

```
%sql SELECT DISTINCT Launch_Site
FROM SPACEXTABLE
```

Explanation:

Pulls names of launch sites from the SPACEXTABLE table

Distinct prevents any duplication

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

Launch Site Names Begin with 'CCA'

Query

%sql SELECT *

FROM SPACEXTABLE

WHERE Launch_Site LIKE “%CCA%

LIMIT 5

Explanation

Pulls all rows (*) from the SPACEXTABLE table

Filters data to find Launch_Site rows that have CCA in them; % signs let text come before or after CCA

Limit statement decreases the results to just the first five rows

Total Payload Mass

Query

%sql SELECT Booster_Version, sum(PAYLOAD_MASS__KG_)

FROM SPACEXTABLE

WHERE Customer = "NASA (CRS)"

GROUP BY Booster_Version

ORDER BY 2

Explanation

Groups data by Booster_Version and sums Payload column based on its Booster_Version

Filters to NASA customers only

Orders by the summed column

Booster_Version	sum(PAYLOAD_MASS__KG_)
F9 v1.0 B0006	500
F9 v1.0 B0007	677
F9 v1.1 B1015	1898
F9 v1.1 B1018	1952
F9 B5 B1059.2	1977
F9 FT B1035.2	2205
F9 v1.1 B1010	2216
F9 FT B1025.1	2257
F9 B5 B1056.2	2268
F9 v1.1	2296
F9 v1.1 B1012	2395
F9 FT B1031.1	2490
F9 B5B1056.1	2495
F9 B5B1050	2500
F9 B4 B1039.2	2647
F9 B4 B1045.2	2697
F9 FT B1035.1	2708
F9 B5 B1058.4	2972
F9 FT B1021.1	3136
F9 B4 B1039.1	3310

Average Payload Mass by F9 v1.1

Query

```
%sql SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Booster_Version LIKE "F9 v1.1%"
```

Explanation

Selects the average of the Payload from the table

Only averages data with Booster_Version F9 v1.1

Result

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

Query

```
%sql SELECT min(Date)
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE "%ground pad%"
```

Explanation

Selects the earliest date from the table

Filters data to Landing_Outcome having “ground pad” in the data

Result

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Query

```
%sql SELECT Booster_Version
FROM SPACEXTABLE
WHERE (PAYLOAD_MASS__KG_ > 4000 and
PAYLOAD_MASS__KG_ < 6000)
and Landing_Outcome LIKE '%drone ship%’
and Landing_Outcome LIKE 'Success%'
```

Explanation

Returns Booster_Version from the table

Filters for Payload between 4,000kg and 6,000kg

Filters Landing Outcome to a success and drone ship

Result

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

Total Number of Successful and Failure Mission Outcomes

Query

%sql SELECT Mission_Outcome, count(Mission_Outcome)

FROM SPACEXTABLE

GROUP BY Mission_Outcome

Explanation

Groups by possible mission outcomes

Selects number of mission outcome events

Result

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

Boosters Carried Maximum Payload

Query

```
%sql SELECT distinct Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ in (
        SELECT MAX(PAYLOAD_MASS__KG_)
        FROM SPACEXTABLE)
```

Explanation

Selects Booster_Version without any repeats from table

Uses sub-query to get highest maximum payload

Filters to only select boosters that have pulled maximum payload

Result

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

Query

%sql SELECT substr(Date,6,2) AS 'Month Name', Landing_Outcome, Booster_Version, Launch_Site

FROM SPACEXTABLE

WHERE Landing_Outcome = 'Failure (drone ship)' AND substr(Date,0,5)='2015'

Explanation

Filters to failed drone ship landing outcomes that occurred in 2015

Selects month it occurred, landing outcome, booster, and launch site

Result

Month Name	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

Query

%sql SELECT Landing_Outcome, COUNT(Landing_Outcome)

FROM SPACEXTABLE

WHERE (Date > '2010-06-04' and Date < '2017-03-20')

GROUP BY 1

ORDER BY 2

Explanation

Filters table to fall in between 6/4/2010 – 3/20/2017

Ranks landing_outcome from lowest to highest

Groups by landing_outcome

Result

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

Section 3

Launch Sites Proximities Analysis

Launch Site Locations

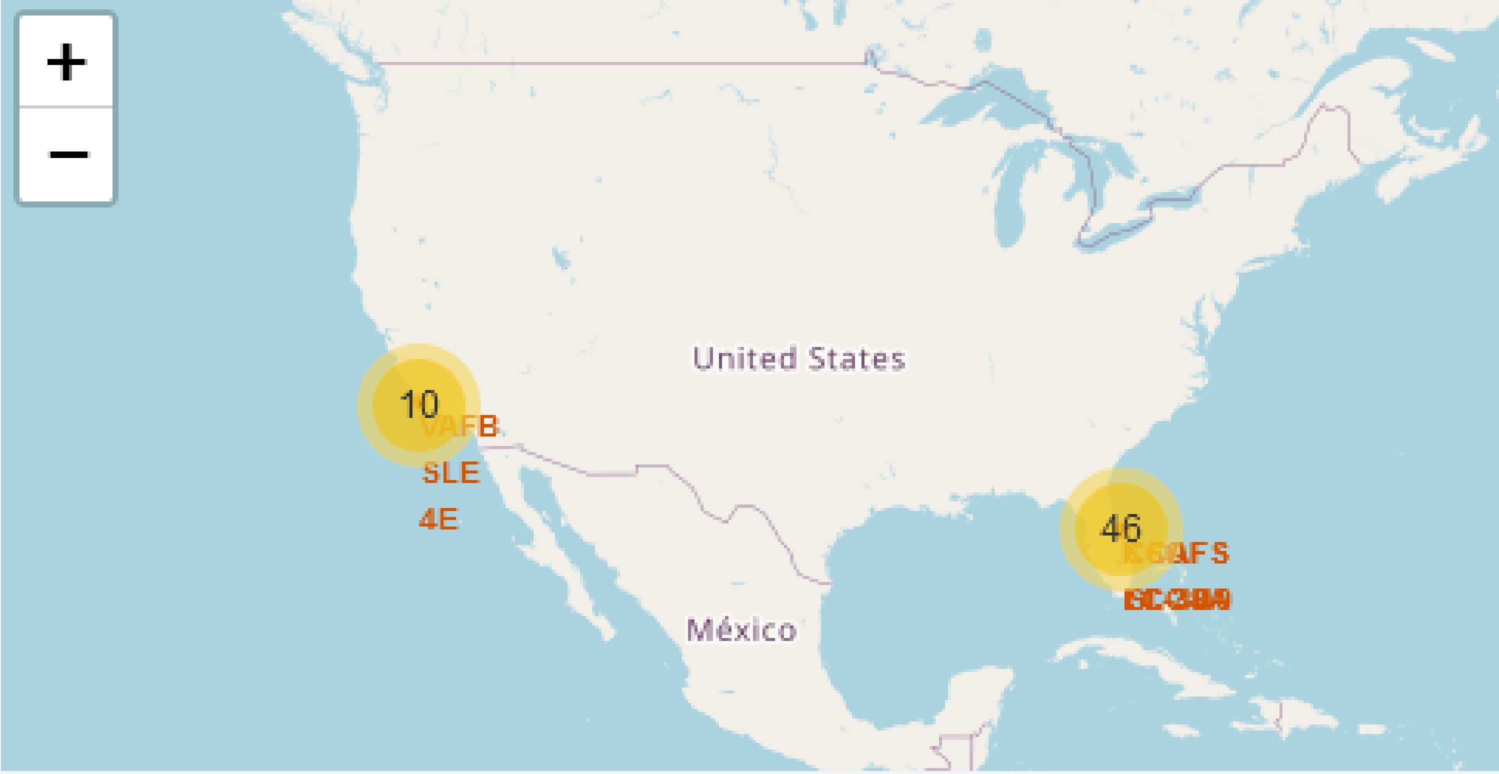
Map shows Launch Site locations

Launch sites are all coastal

Florida contains multiple launch sites

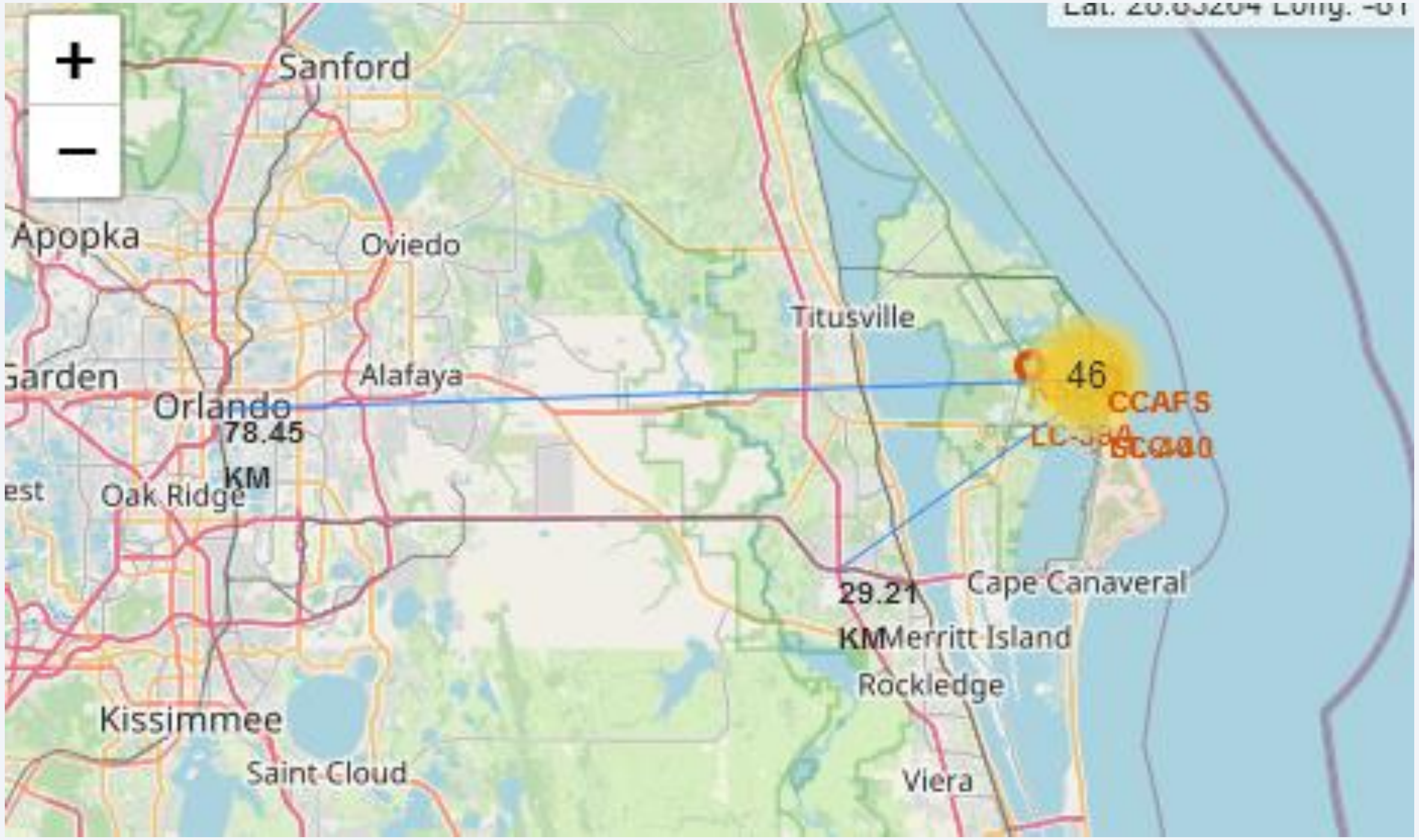
Volume of Launches by Location

Most launches take place in Florida



Space Station Proximity

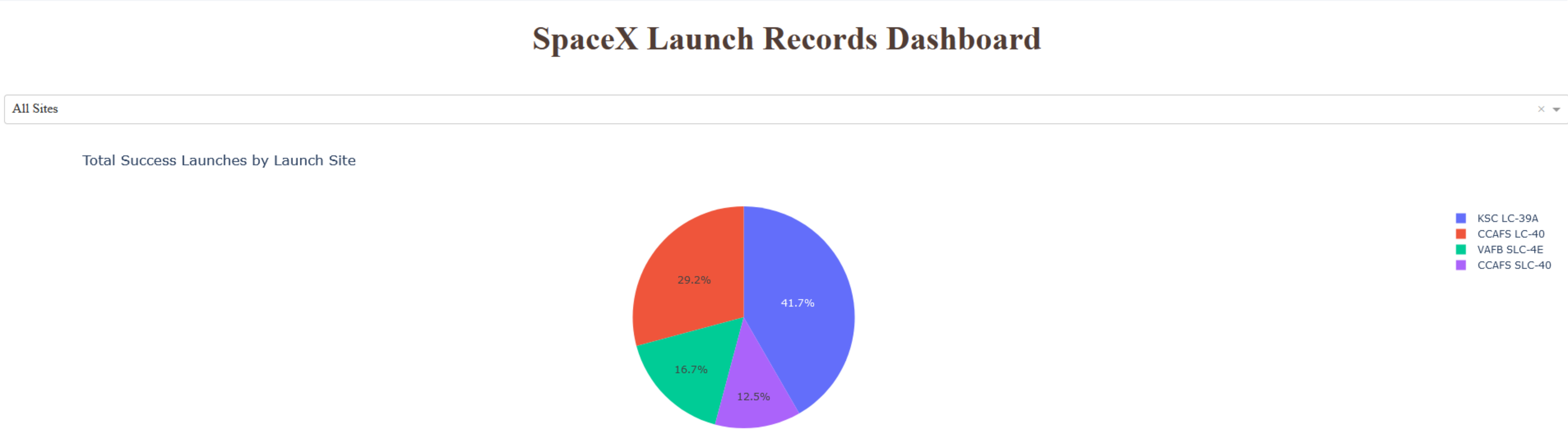
- 78.5KM from Orlando
- 29.2KM from freeway
- On the coast
    - Water landings
    - Prevent civilian injuries



Section 4

Build a Dashboard with Plotly Dash

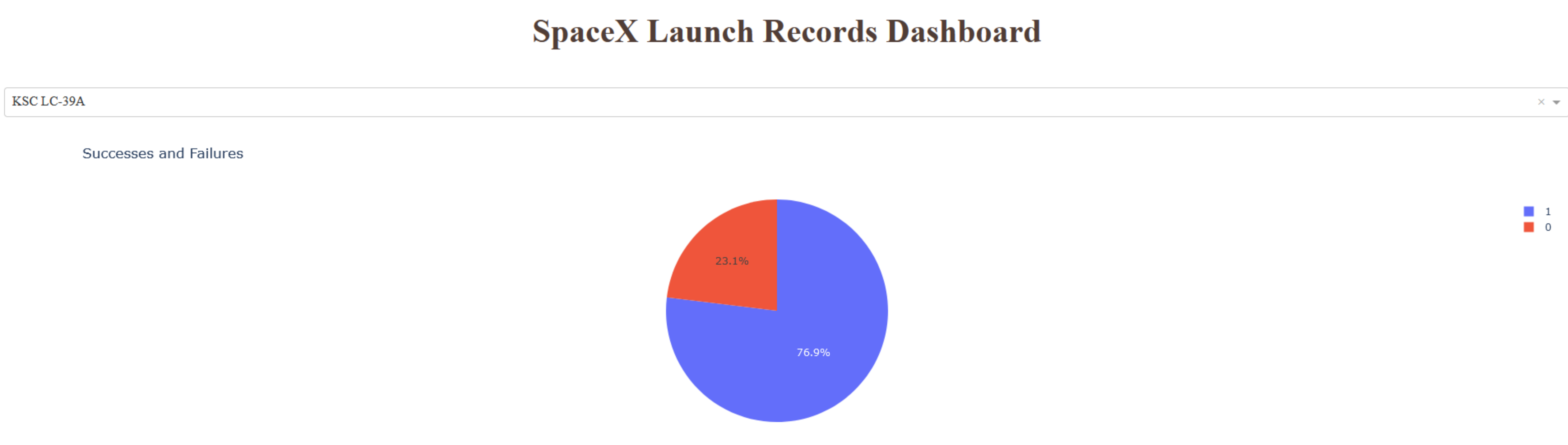
Successful Launches by Launch Site



Total Successful Launches by Launch Site

Of the successful launches, 41.7% launch from Kennedy Space Center

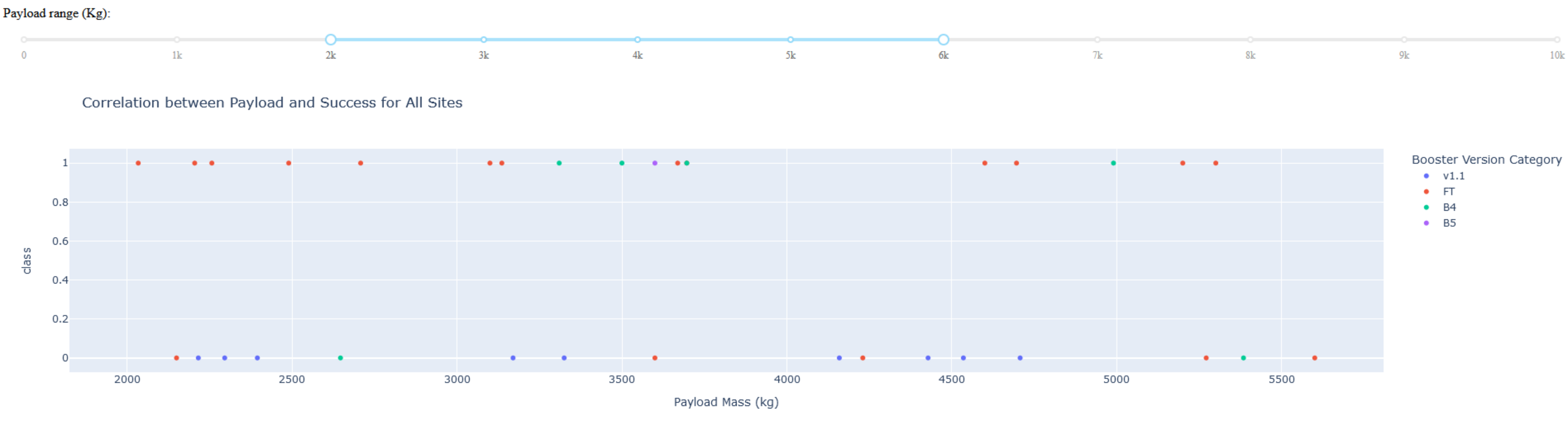
Kennedy Space Center Launch Success



The success and failure rate of all launches at Kennedy Space Center

76.9% of launches from Kennedy Space Center succeed

Correlation between Payload and Success for All Sites



Provides success and failure values for all sites between a payload range

Provides which booster categories were attached to each launch

The FT Booster Category seems to have a high success rate

The graph can also filter by launch site

Section 5

Predictive Analysis (Classification)

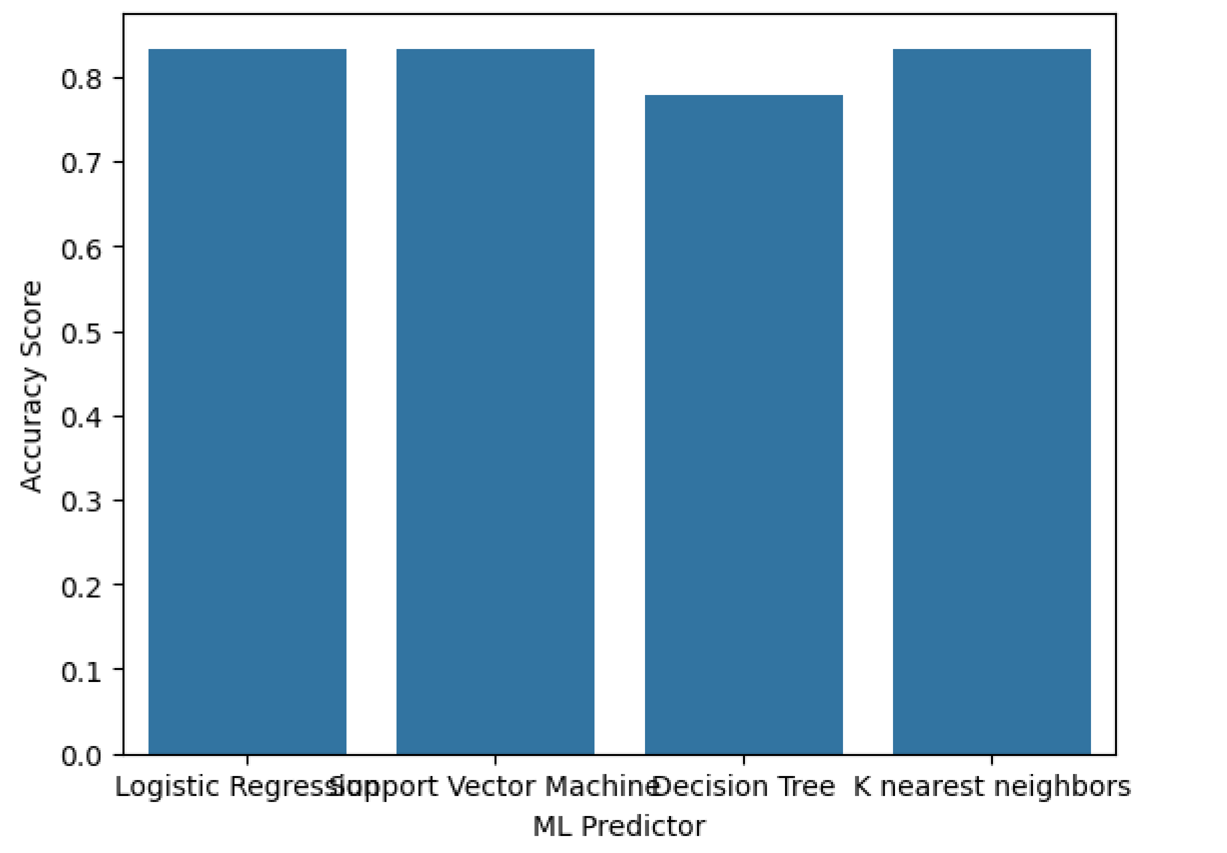
Classification Accuracy

Accuracy Scores are similar across all four models

Models have been optimized by passing various parameters to produce best model

No test stands out as the best meaning any classification system can be used

Decision Tree accuracy fluctuates slightly and therefore should be eliminated

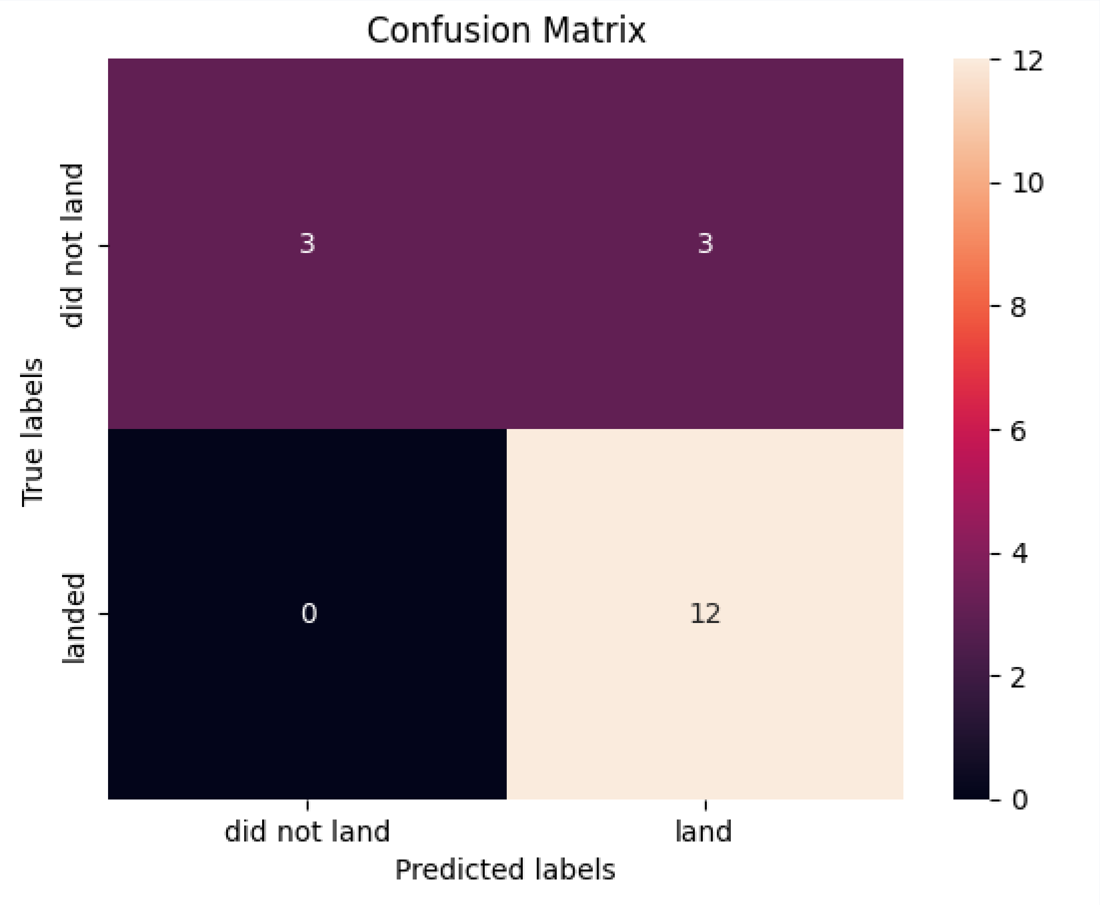


Confusion Matrix

The Confusion Matrix is similar across remaining three models

Models all correctly predict “Land” but do have some False Positives

This 83% Accuracy can be useful in deciding whether future launches are viable



Conclusions

- Payload, Orbit Type, and Launch Site play a huge role in determining mission success and Landing Outcome
- It is possible that Booster Version could play a greater role, but needs to be analyzed further
- Launch Sites are required to be on the coast with some distance from cities
- Several classification Machine Learning methods can be used to determine success and failure of missions, but are not 100% accurate

Thank you!

Appendix A: GitHub Notebook Links

- Collecting Data: https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb
- Webscraping: https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/jupyter-labs-webscraping.ipynb
- Data Wrangling: https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb
- Exploratory Data Analysis (SQL): https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb
- Exploratory Data Analysis (Visualization): https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb
- Folium: https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb
- Dashly Dashboard: https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/spacex_dash_app.py
- ML Predictions: https://github.com/bleepbloop1213/AppliedDataScienceCapstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb