IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

David Taylor
9/17/2025

Outline

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

Executive Summary

- Collected data via SpaceX API
- Cleaned and filtered data for Falcon 9 rockets

- Summary of all results

Introduction

- Project background and context
- Determine the price of each launch
- Build a model to predict whether a launch will be successful, ie whether stage 1 of rocket will land and can be reused.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was retrieved via SpaceX API
  - Data was webscraped
- Perform data wrangling
  - Missing values were filled
  - Data was filtered
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Classification model (success/lanede

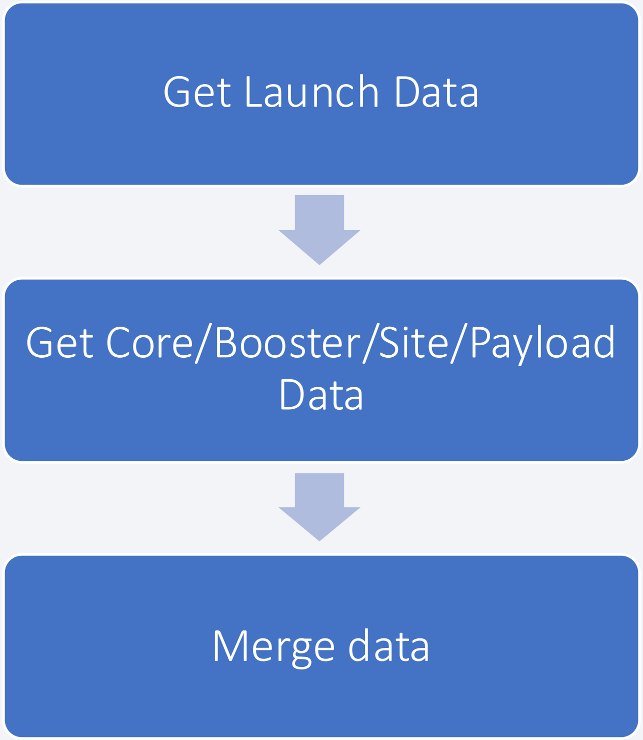
Data Collection

- Describe how data sets were collected.
    - SpaceX API
    - Webscraping
- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

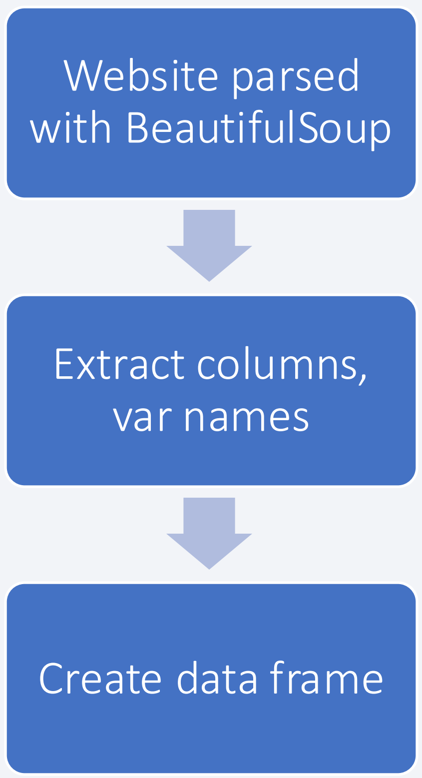
- Initial API call to get launch data
- Subsequent API calls to get Rocket Core, Booster, Launch Site and Payload data.
- Merge associated data from all calls to to a single data point

- Github link: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/jupyter-labs-spacex-data-collection-api-v2.ipynb



Data Collection - Scraping

- SpaceX Launch Wiki Link was scraped
- Table with launch data identified and parsed
- Columns and variable names extracted
- Added all data to data frame
- Github URL: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Describe how data were processed
- You need to present your data wrangling process using key phrases and flowcharts
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose

EDA with Data Visualization

- Flight Number vs Launch Site, scatterplot
- Flight Number vs Payload Mass, scatterplot
- Payload Mass vs Launch Site, scatterplot
- Average Success by Orbit, bar chart
- Flight Number vs Orbit, scatterplot
- Payload Mass vs Orbit, scatterplot
- Average Success by Year, line chart
- Github URL: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/jupyter-labs-eda-dataviz-v2%20(1).ipynb

EDA with SQL

- Names of the unique launch sites
- 5 records where launch site starts with CCA
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- Date when the first successful landing outcome in ground pad was achieved
- Names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Total number of successful and failure mission outcomes
- "booster_versions" that have carried the maximum payload mass
- Records showing month, landing outcome, booster version and site for 2015
- Landing outcomes and their totals between the date 2010-06-04 and 2017-03-20
- Github URL: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

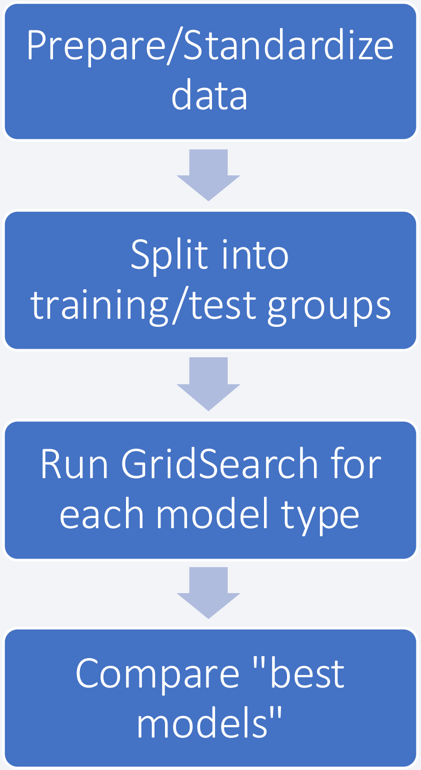
- Added markers for Launch site locations
- Github URL: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/lab-jupyter-launch-site-location-v2.ipynb

Build a Dashboard with Plotly Dash

- Add a dropdown list to enable Launch Site selection
- Add a pie chart to show the total successful launches count for all sites (or if a site is selected then show successful/failed launches
- Add a slider to select payload range (to filter the data)
- Add a scatter chart to show the correlation between payload and launch success
- Github URL: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/spacex-dash-app.py

Predictive Analysis (Classification)

- Separate X and Y
- Standardize X using StandardScaler
- Split into train/test (test_size = 0.2)
- GridSearchCV for the following models
  - LogisticRegression, SupportVectorMachine, DecisionTreeClassifier, KNeighbors,
- Evaluated the confusion matrix and accuracy scores for each "best model" for both training and test data
- Github URL: https://github.com/codewizard-dt/ibm-datascienc-capstone/blob/main/SpaceX-Machine-Learning-Prediction-Part-5-v1.ipynb



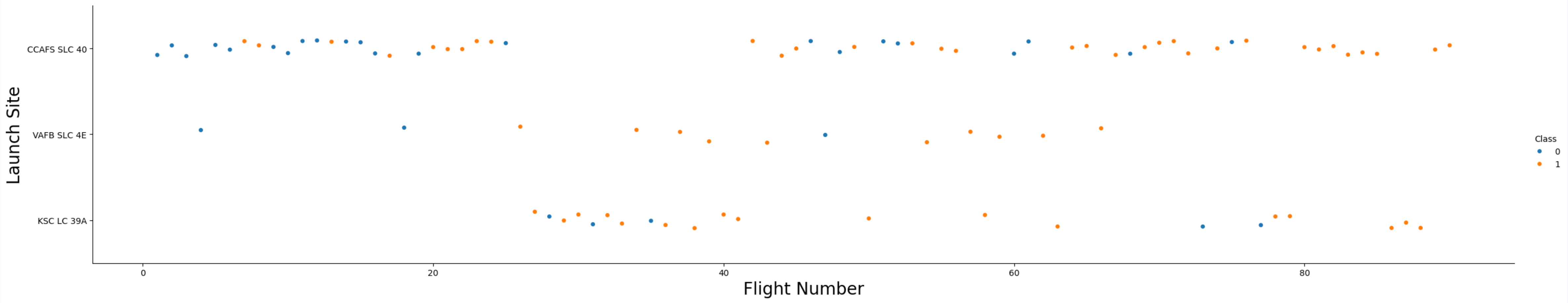
Results

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

Section 2

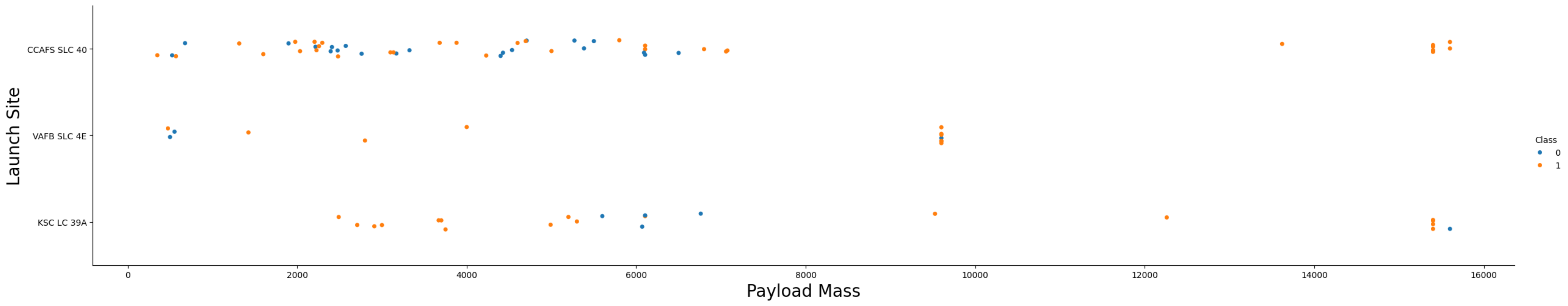
Insights drawn from EDA

Flight Number vs. Launch Site



- Blue = failure, orange = success
- Launch site 'VAFB' had higher success compared to others, but also fewer launches

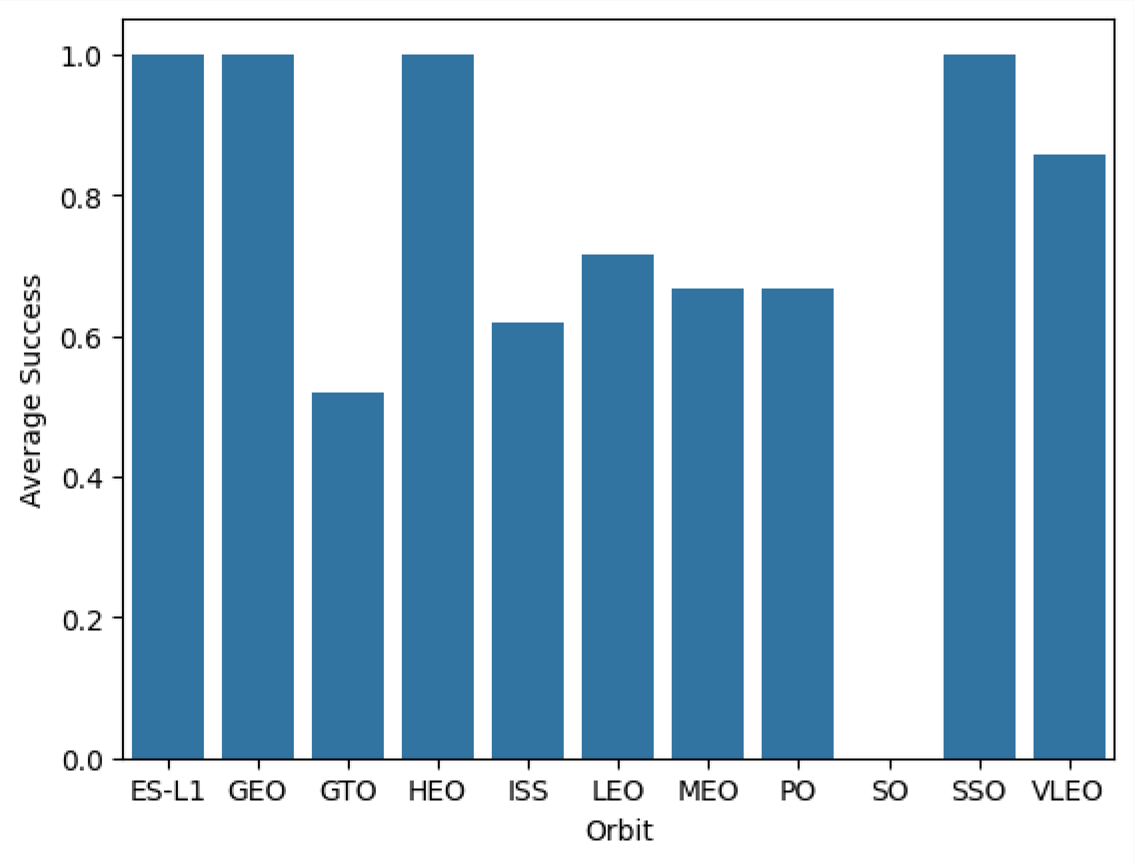
Payload vs. Launch Site



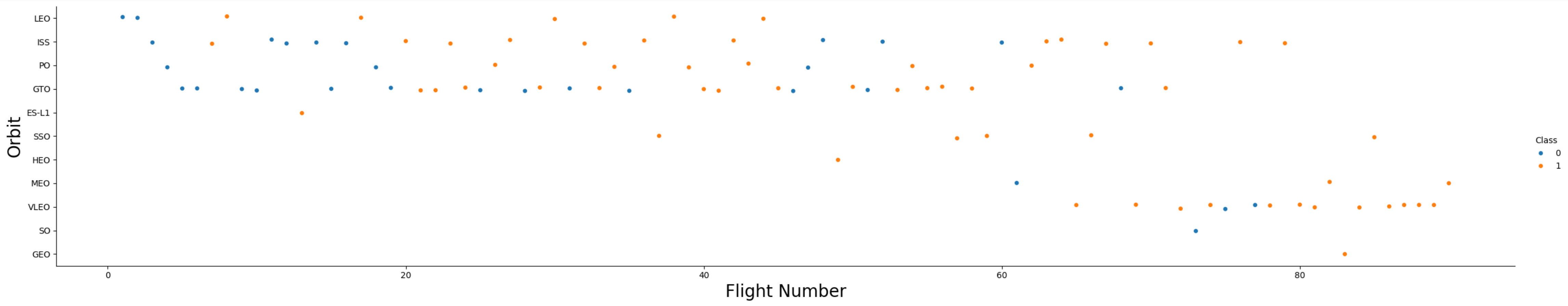
- Blue = failure, orange = success
- Only 2 out of 3 launch sites had large payloads
- Large payloads had higher success rates

Success Rate vs. Orbit Type

- SO orbit had all failures
- ES-L1, GEO, HEO, SSO had all successes
- 6 out of 11 orbits had mixed success

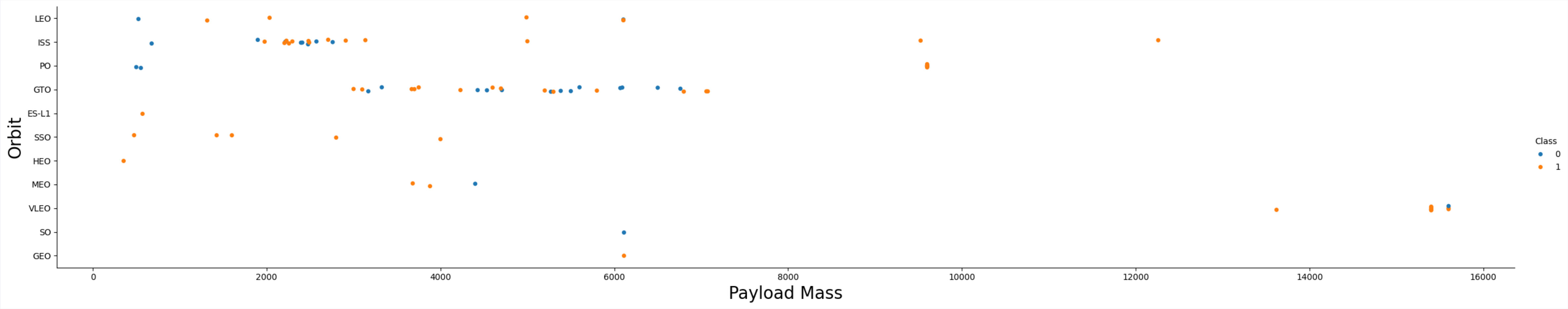


Flight Number vs. Orbit Type



- Blue = failure, orange = success
- Earlier flights were less successful, while later flights more successful
- VLEO, MEO, HEO, SSO, GEO were only tried for later flights

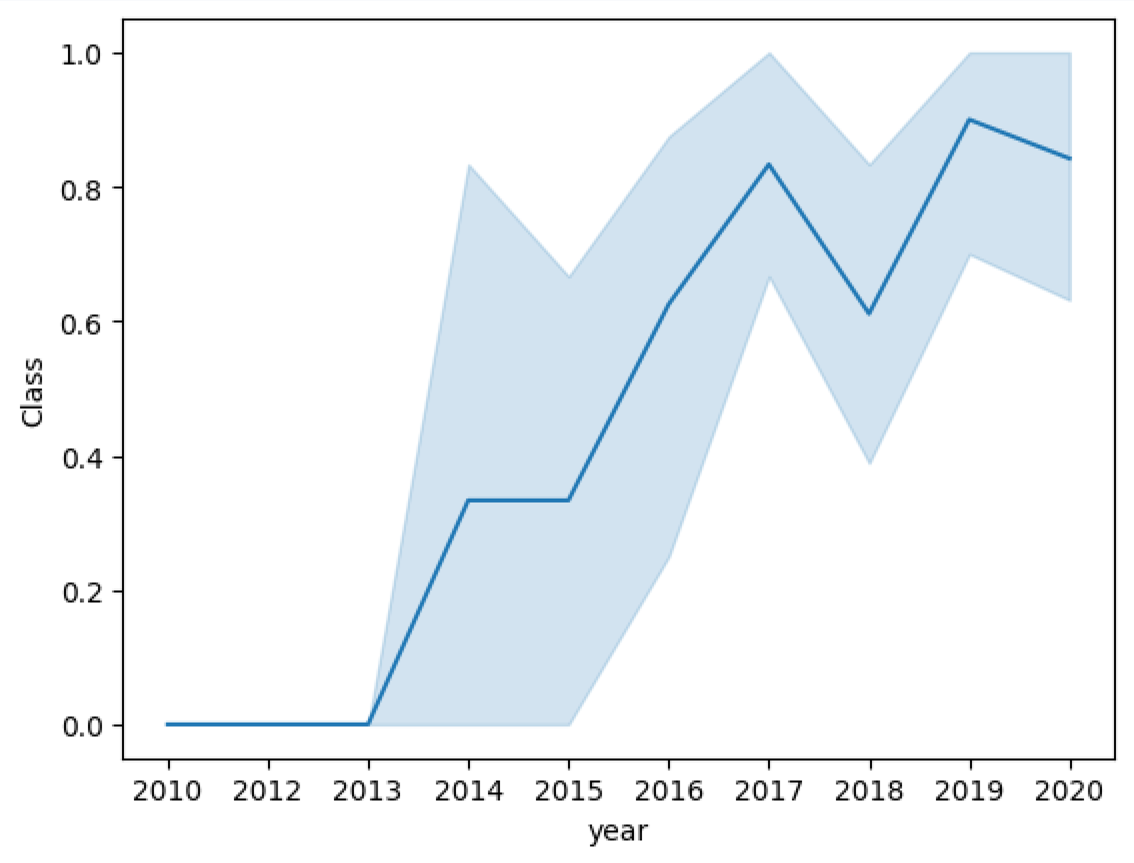
Payload vs. Orbit Type



- Blue = failure, orange = success
- VLEO had only large payloads, while MEO, SSO, HEO had relatively small payloads
- GTO had a mix of success/failures for both small and large payloads
- LEO/SS had more success with larger payloads

Launch Success Yearly Trend

- No success until 2013
- 2013-2015 shows increasing success but large variance
- Around 2017 success leveled out and remained above 60%



All Launch Site Names

- Find the names of the unique launch sites
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40
- There are 4 unique launch sites

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Using LIKE "%" and LIMIT to get the correct records

```
%sql select * from SPACEXTABLE where launch_site like 'CCA%' limit 5
```

✓ 0.0s Python

Running query in 'sqlite:///my_data1.db'

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

- Calculate the total payload carried by boosters from NASA
- Using SUM and WHERE to correctly aggregate
- 45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Using AVG and WHERE to correctly aggregate
- 2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Using
  - where landing_outcome = 'Success (ground pad)'
  - order by Date ASC limit 1
- 2015-12-22

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
  - F9 FT B1022
  - F9 FT B1026
  - F9 FT B1021.2
  - F9 FT B1031.2
- Using
  - distinct Booster_Version
  - where landing_outcome = 'Success (drone ship)' and payload_mass__kg_ between 4000 and 6000

Total Number of Successful and Failure Mission Outcomes

- Calculate the 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

- List the names of the booster which have carried the maximum payload mass
    - 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
- Using subquery to get the max payload, then get distinct booster version

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Using WHERE landing_outcome = 'Failure (drone ship)' and substr(Date,0,5)='2015'

month	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

- 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
- Using
  - Count (landing_outcome) as total
  - Group by landing_outcome

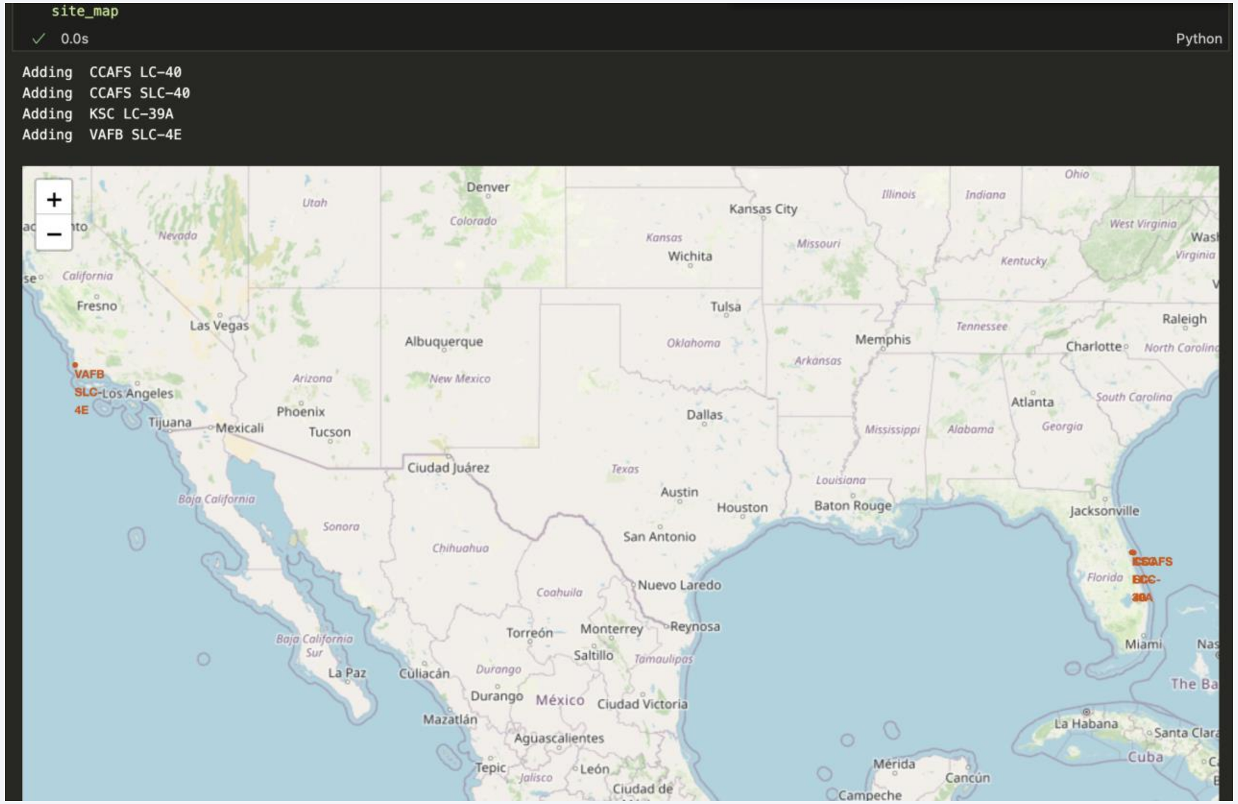
Landing_Outcome	total
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

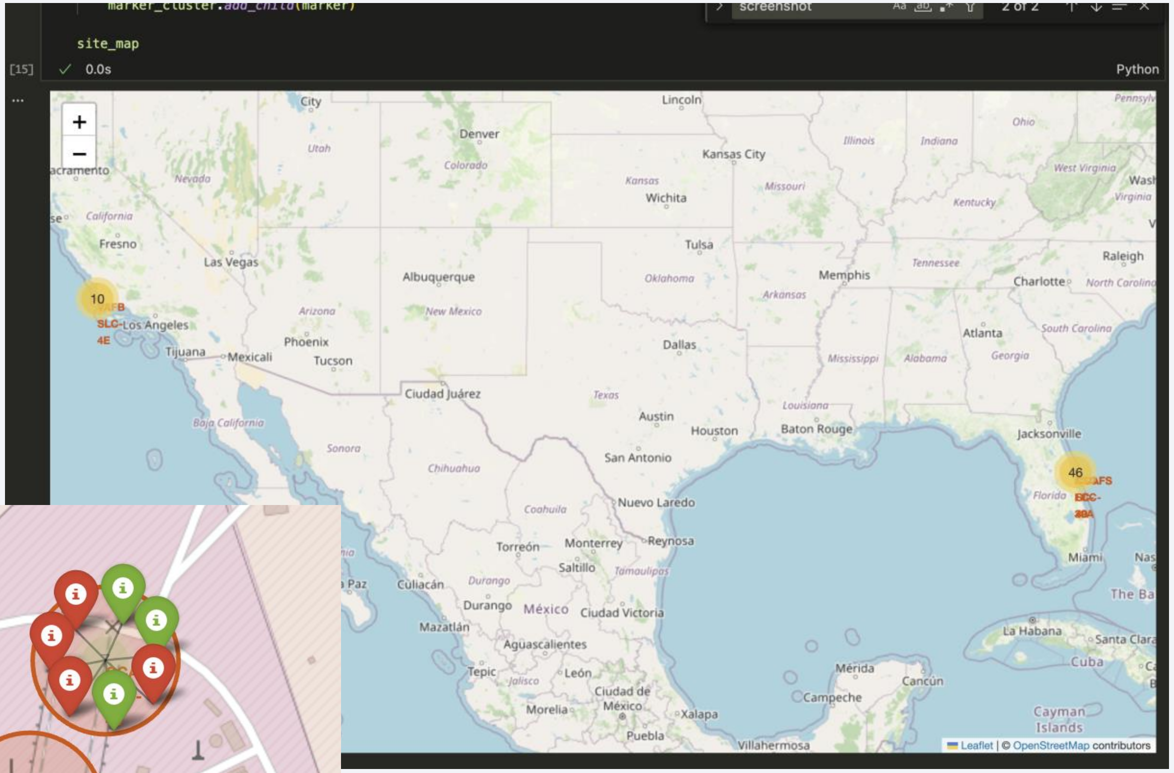
Launch Site Locations

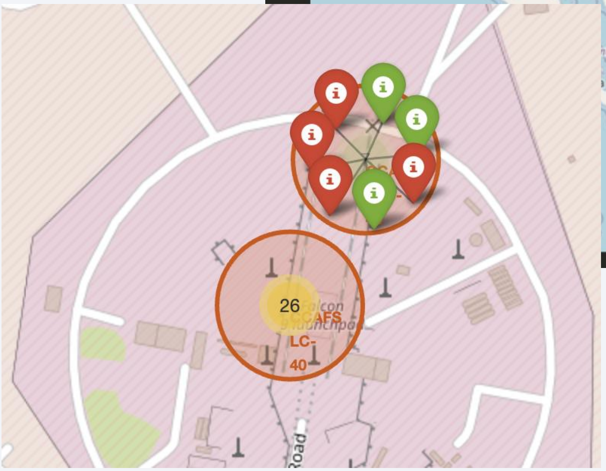
- Each marker is a launch site location



Launch Outcomes

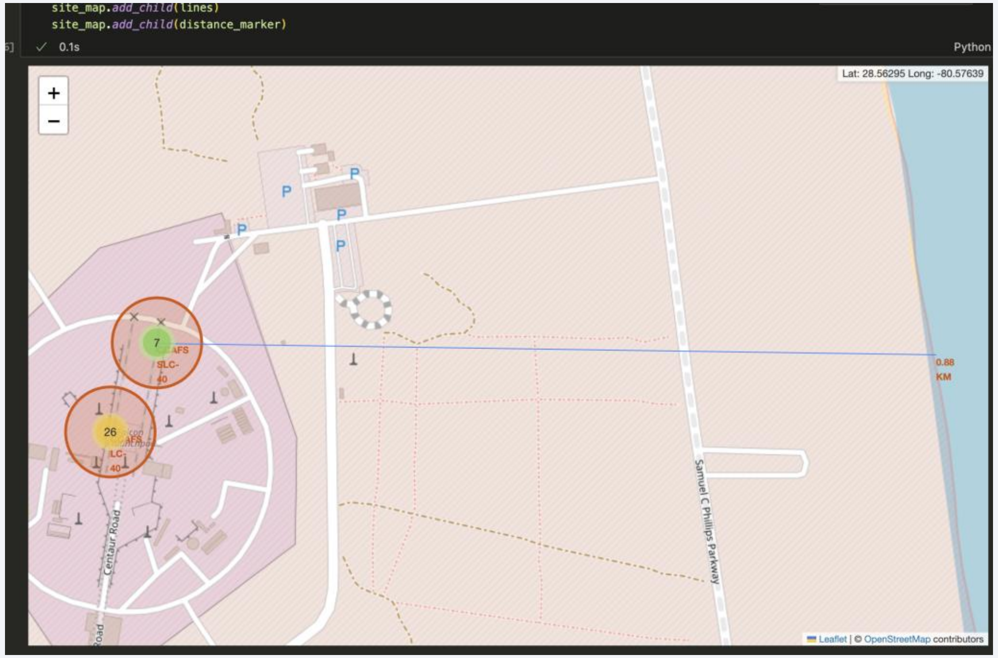
- Each launch site is marked
- There is a Cluster of markings for each site that shows the outcomes of each individual launch





Distance To Coast

- A marking that shows a spot on the coast
- Plus a line that shows the distance between the launch site and the coast

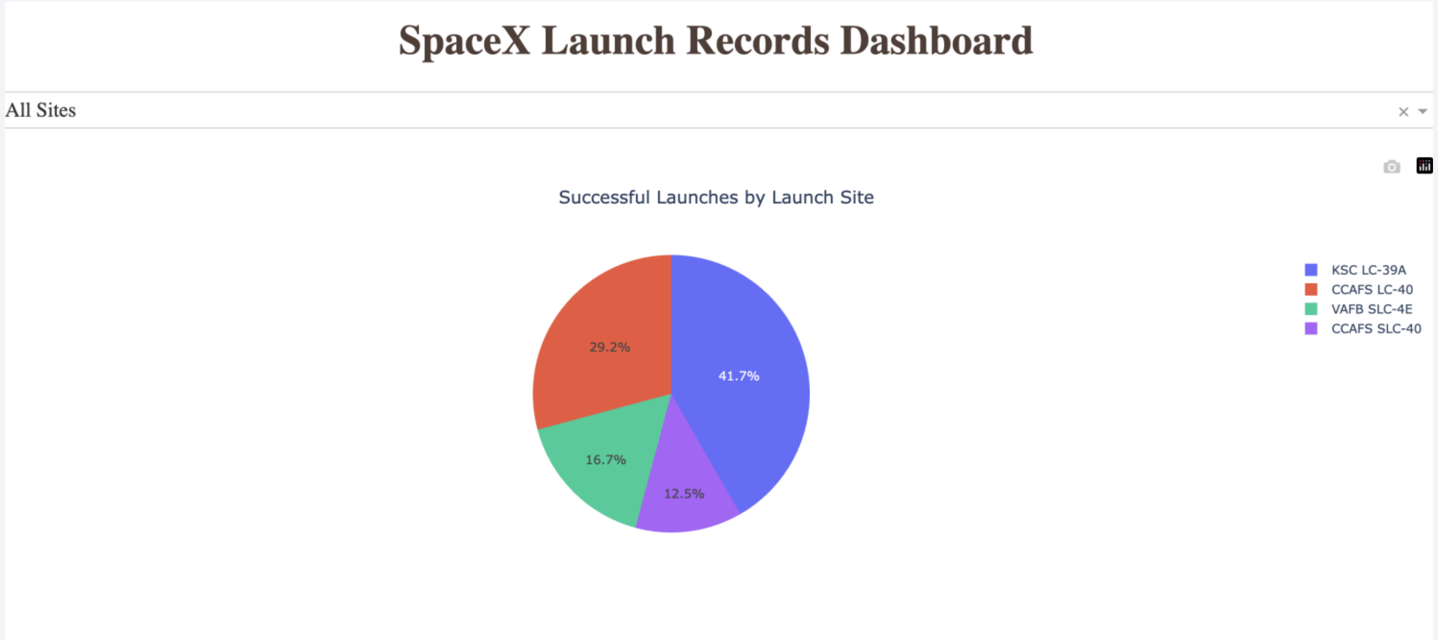


Section 4

Build a Dashboard with Plotly Dash

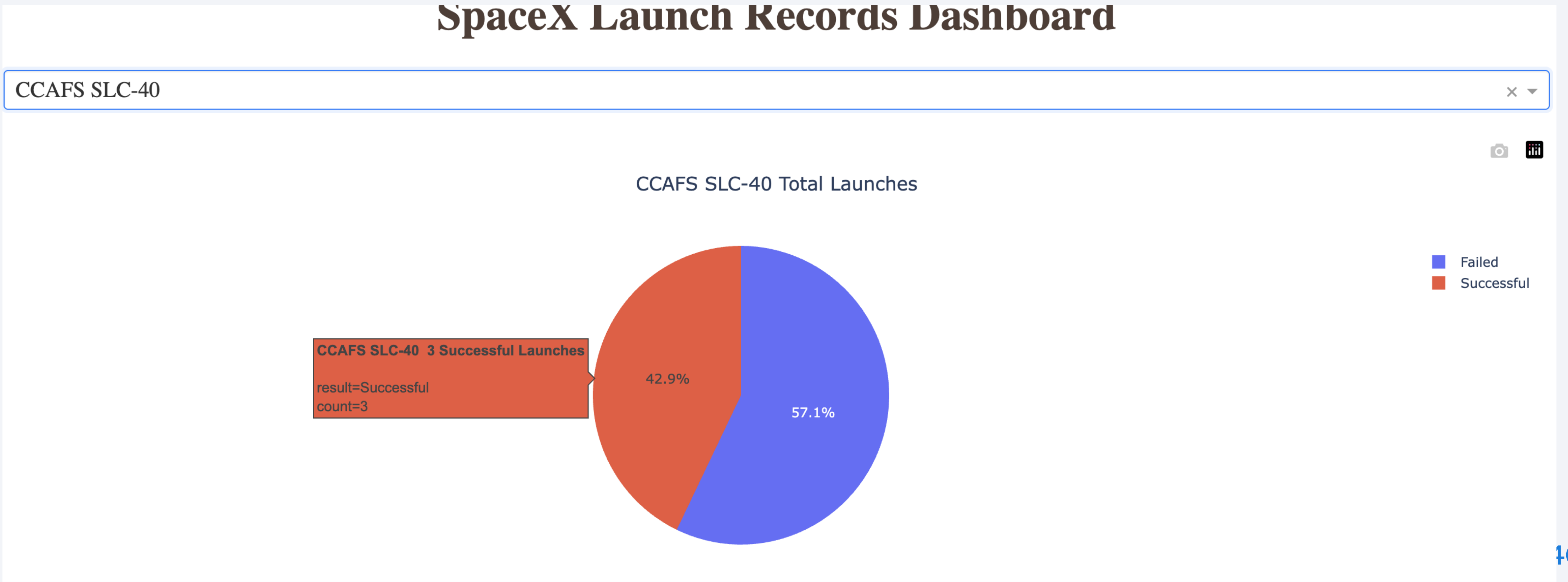
<Dashboard Screenshot 1>

- The dropdown has "All Sites" selected
- The pie chart shows Successful launches for all sites



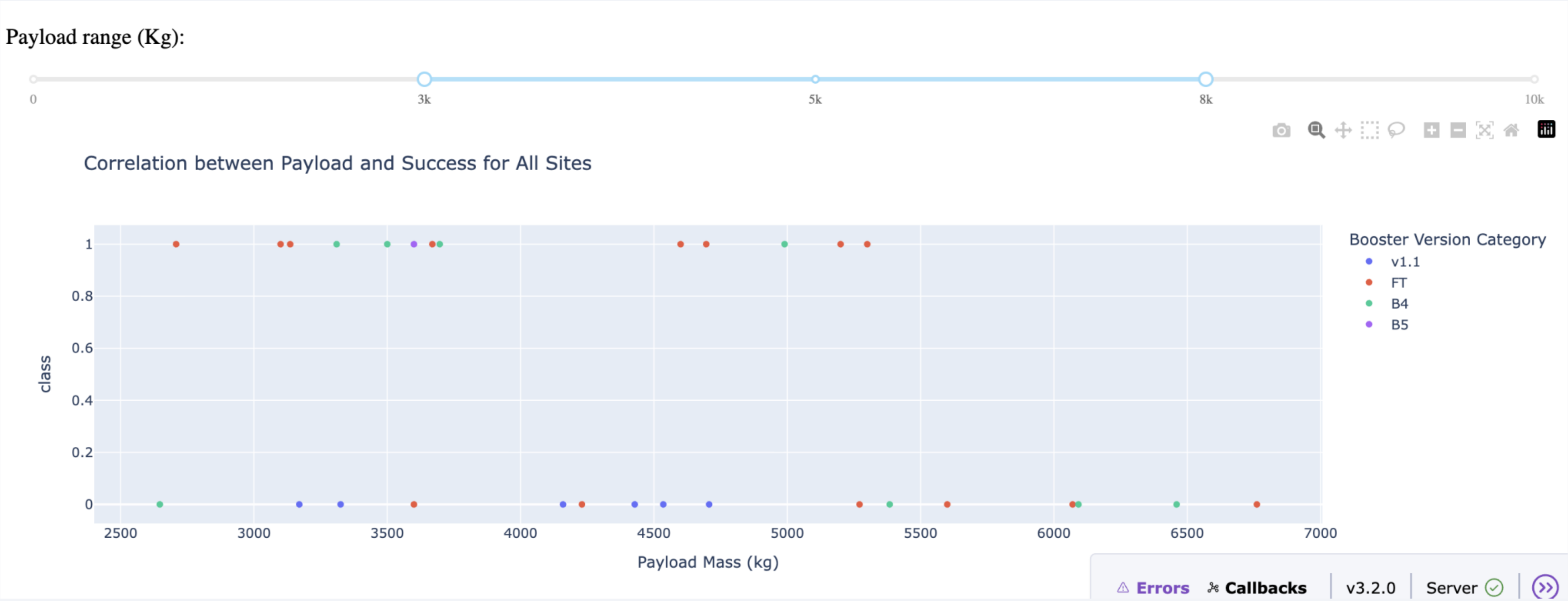
Most successful launch site

- CCAFS SLC-40 has a success rate of 42.9%



Correlation between Payload and Success for All Sites

- Showing payload > 3 and < 8

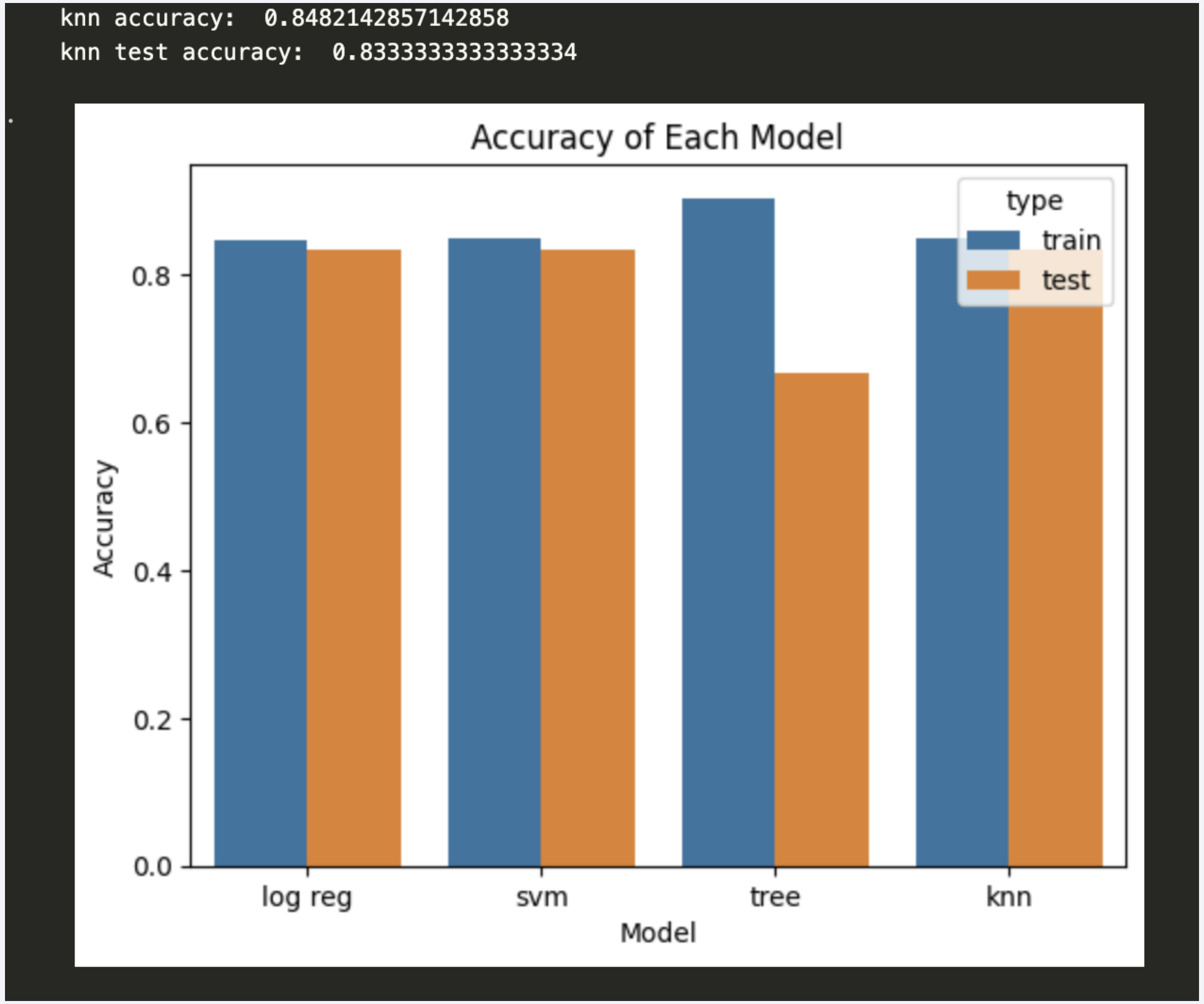


Section 5

Predictive Analysis (Classification)

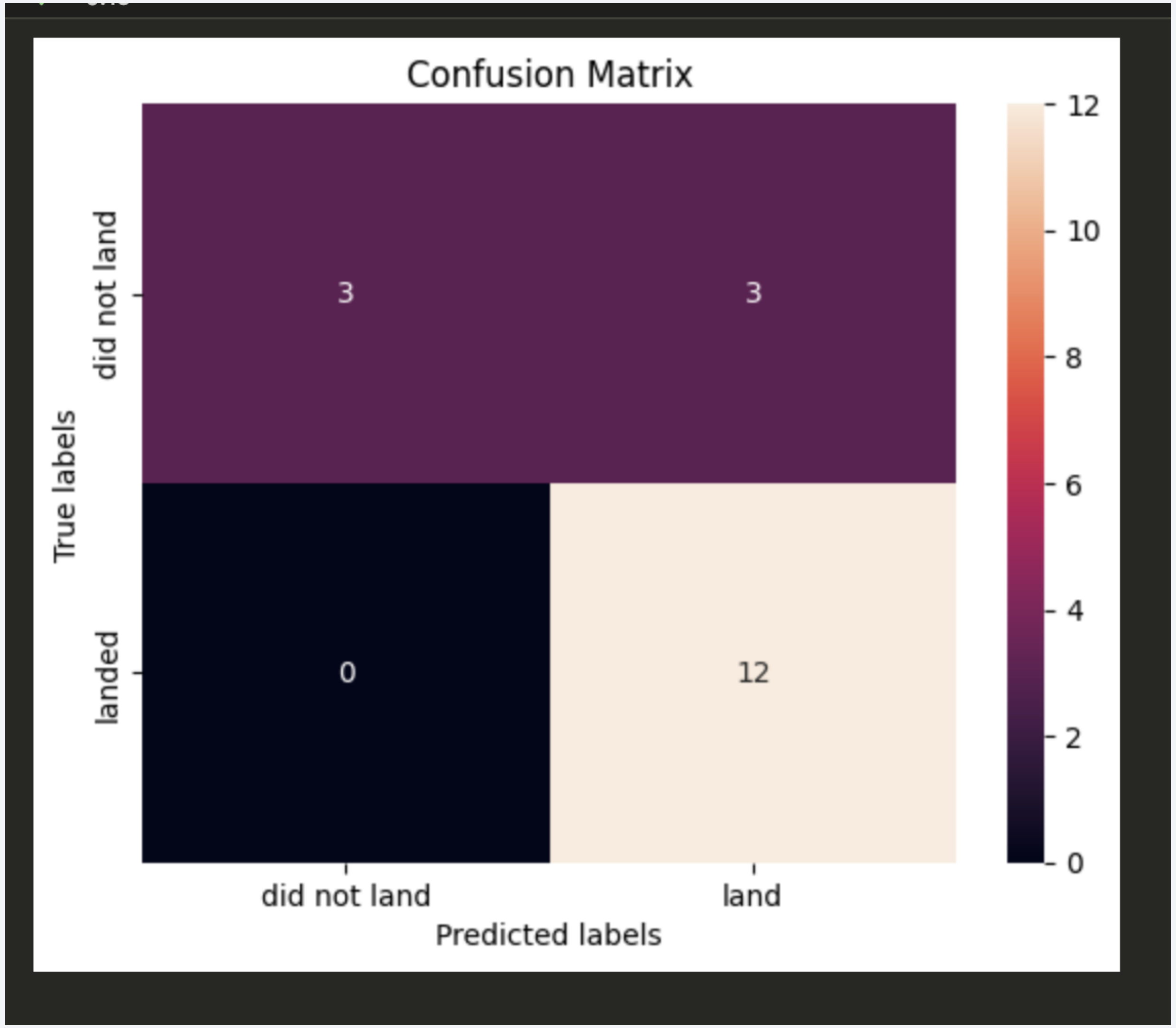
Classification Accuracy

- Tree classifier had highest accuracy with training data but not with test data
- All other models performed the same



Confusion Matrix

- Problem with false positives (predict 'land' but actually 'did not land')



Conclusions

- Most of the models were pretty good at predicting.
- Tree was bad
- Problem with false positives

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!