Rocket Launch Results

Derek Rush

12/27/23

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

- Various programing libraries were used to collect and analyze data on the SpaceX rocket launches.
- Examples of libraries used:
  - Pandas
  - Numpy
  - Datetime
- Data was cleaned and normalized so that it could be easily compared.
- Charts/graphs were created in order to create inferences.

INTRODUCTION

- Space Y is trying to determine how expensive launches will be by using data from SpaceX launches.
- The information that Space Y requires is:
  - Where are the best locations for test launches?
  - How often do we expect test launches/landings to pass?
- By using this information, Space Y will be able to determine how expensive its rocket launches will be.

Data Collection

- Data was collected from the below sources:
  - https://api.spacexdata.com/v4/launches/past
  - https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- The API data was collected using the "Requests" library.
- The Wikipedia data was collected using BeautifulSoup webscraping.

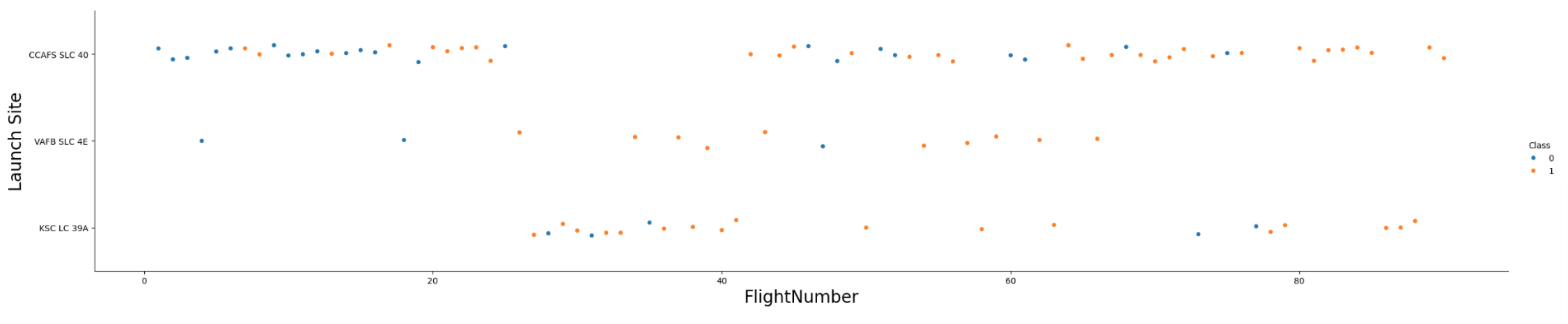
Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Wrangling

- After the data was collected, it needed Exploratory Data Analysis (EDA) was performed.
- The data needed to be cleaned/manipulated in order to provide suitable information.
  - The data that was in 'string' form was converted to numerical data.
  - Landing outcomes were the focus of the data since this is crucial to determining the price of the rocket launch.

Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization



- The above chart shows successful and unsuccessful landing outcomes by launch site and number of flights
  - Blue dots (class 0) indicate fail cases
  - Yellow dots (class 1) indicate pass cases
- Inference example – The CCAFS SLC 40 launch site is more likely to result in successful tests when the number of tests for that rocket are low.

Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- SQL was used in order to analyze data across the database.
- Queries were run for the following purposes:
  - Extract all of the Launch Site data.
  - Calculate the sum of the payloads for a specific customer.
  - Find the average payload mass for a specific booster version.
  - See which landing outcomes were successful for ground pad landings.
  - Visualize the mission and outcome success in a tabular format.
  - Ranking the landing outcomes based on how many times the outcome occurred.
  - Additional queries were run to provide helpful information.

Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Building an Interactive Map with Folium

- Circles, markers, marker clusters and lines were plotted on a map.
  - Circles were used to highlight areas were launch site were located.
  - Markers were used to determine the specific point of each launch site.
  - Marker clusters were used to denote successful (green) and unsuccessful (red) launch outcomes.
  - Lines were used to determine the distance between two points.

Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Building a Dashboard Using Plotly Dash

- Plotly Dash was used to create an interactive visuals:
  - Pie chart showing the launch sites and success rates.
  - Scatter plot showing success rate per payload mass.
- Using both visuals simultaneously allows us to make inferences for what sites and payload masses will produce successful launch outcomes.

Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis Using Machine Learning

- Four models were used in order to create a predictive model:
  - Logistical Regression
  - SVM
  - Classification Tree
  - K-Nearest Neighbor
- The models were first loaded with data and trained on a subset of the data.
- Then the models were tested to see if they could accurately predict the results of the data that was not included in the training.
- Finally, the results of each model were compared to each other to see which model should be used for predictive purposes.

Source Code: https://github.com/DerekRush98/IBM-Capstone/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Conclusion and Results

- Using Exploratory Data Analysis we can make suggestions for the SpaceY team.
    - Launch sites should be located on the southern coasts, with Florida being the first option.
    - It is best to keep the payload mass between 2k and 6k kg in order to have successful outcomes.
    - The FT booster version is the best to use in order to have a successful outcome.
- Using Predictive analysis, we can come to a decision on which model to use.
    - The Classification Tree proved to be the most accurate with 90% accuracy, so this model should be used in order to make additional launch decisions.

Appendix

SQL Example

- This is an example of the SQL code used to clean the data.

In [26]:

```
%%sql SELECT Mission_Outcome, Landing_Outcome, COUNT(Mission_Outcome), COUNT(Landing_Outcome) FROM SPACEXTABLE
GROUP BY Mission_Outcome, Landing_Outcome;
```

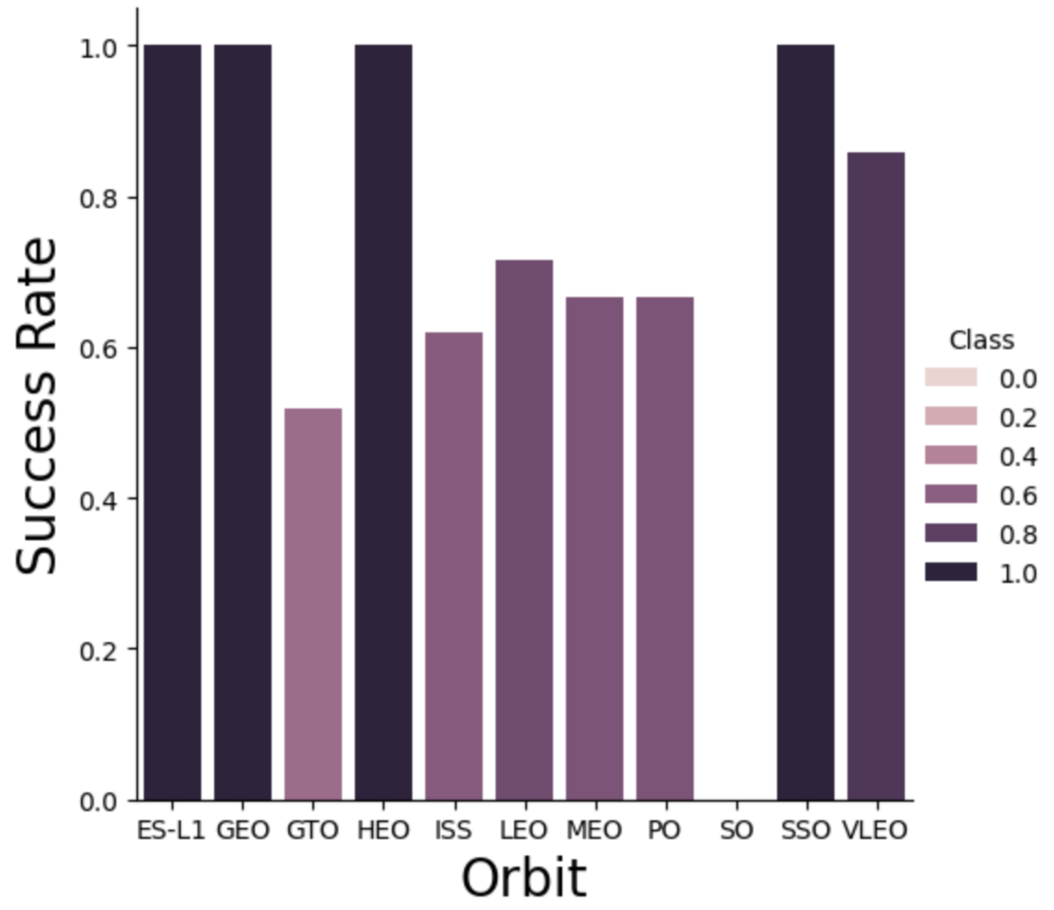
```
 * sqlite:///my_data1.db
Done.
```

Out[26]:

Mission_Outcome	Landing_Outcome	COUNT(Mission_Outcome)	COUNT(Landing_Outcome)
Failure (in flight)	Precluded (drone ship)	1	1
Success	Controlled (ocean)	5	5
Success	Failure	3	3
Success	Failure (drone ship)	5	5
Success	Failure (parachute)	2	2
Success	No attempt	20	20
Success	No attempt	1	1
Success	Success	38	38
Success	Success (drone ship)	14	14
Success	Success (ground pad)	8	8
Success	Uncontrolled (ocean)	2	2
Success	No attempt	1	1
Success (payload status unclear)	Success (ground pad)	1	1

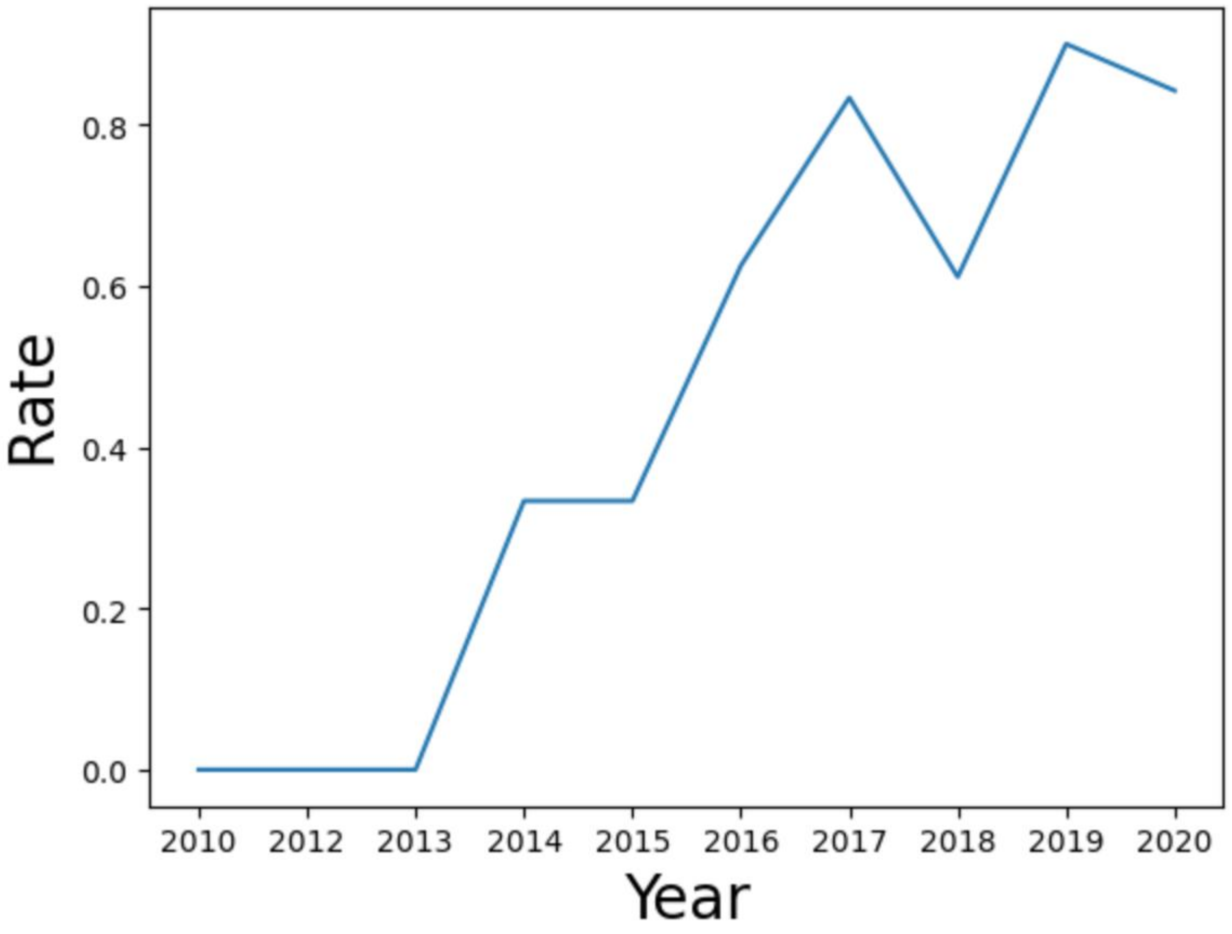
Success Rate by Orbit

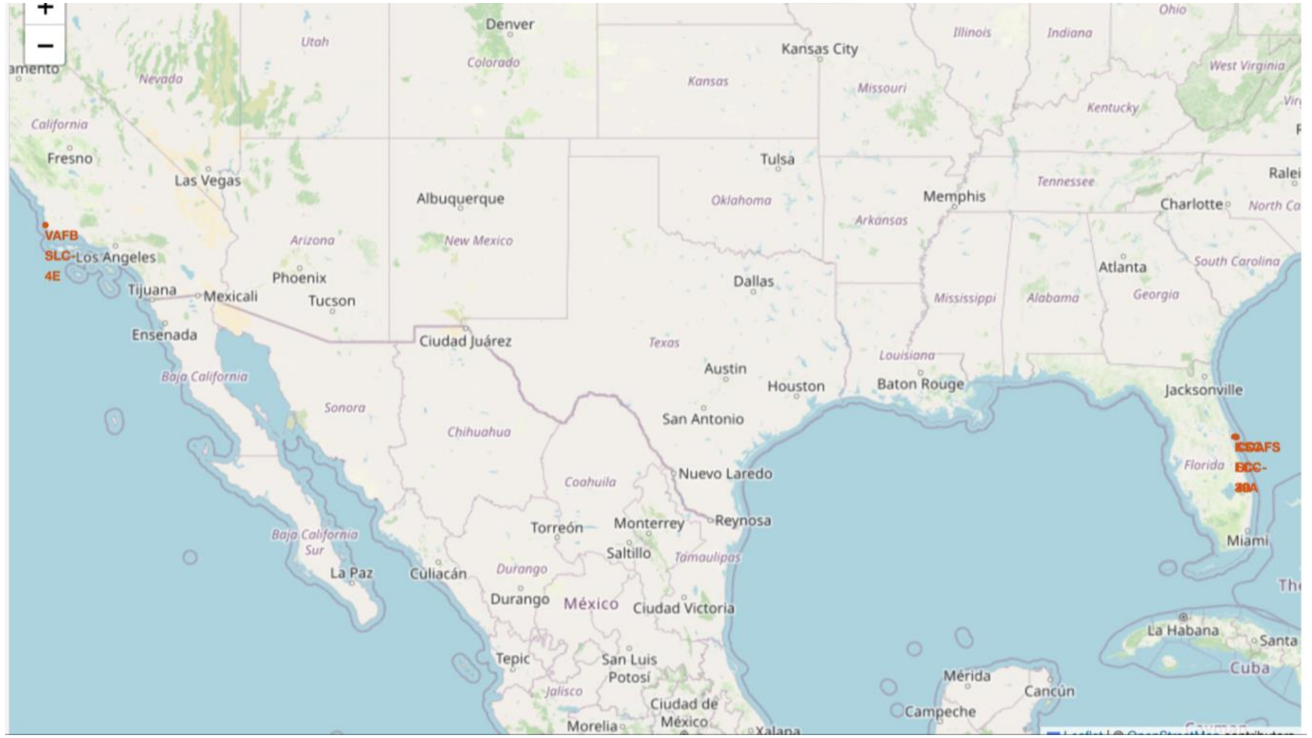
- This bar chart shows how orbit impacts success rate of the launches.
- ES-L1, GEO, HEO and SSO all have a 100% success rate.



Success Rate by Year

- This line chart shows that the success of the launches has increased since 2010.
- The success rate was the highest at around 90% success rate in 2019.



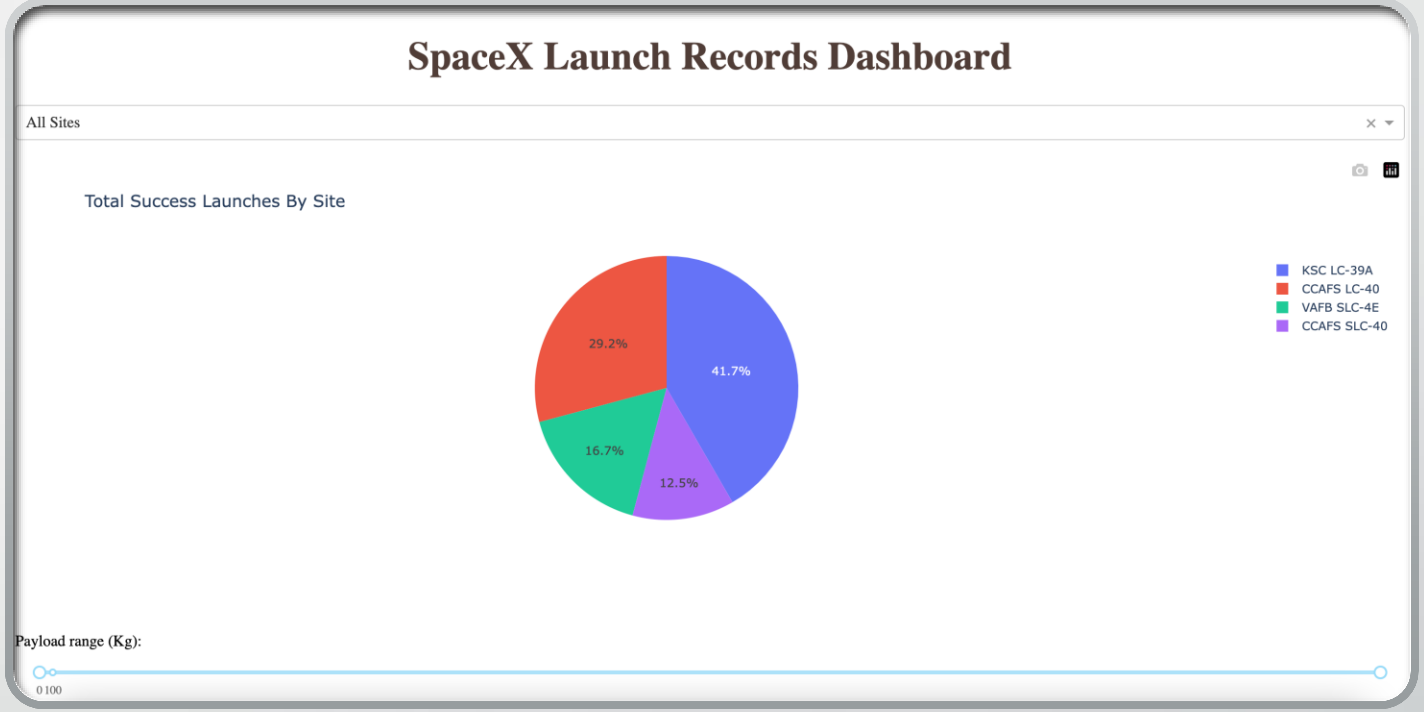


Folium Example

The red dots/text on this map show how folium is used to create interactive maps with popups. The two locations plotted on the map are where SpaceX has its launch sites.

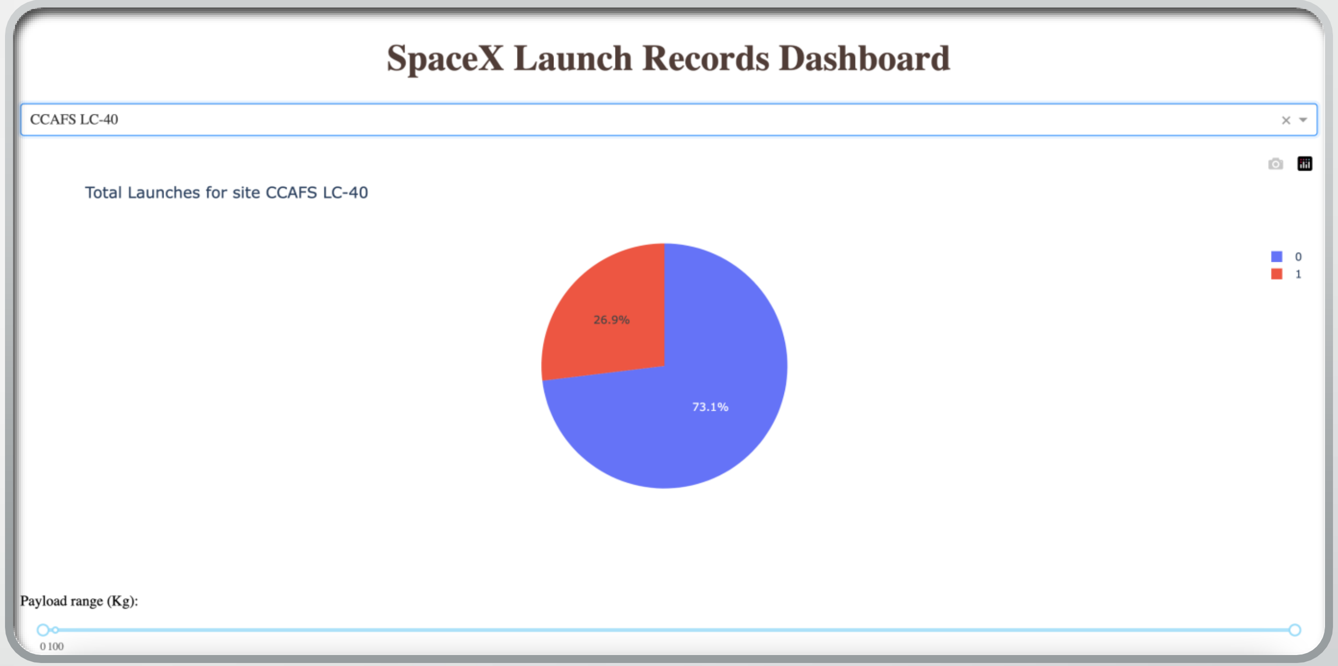
Example of the Plotly Dashboard

- The pie chart shows success by launch site. We can adjust the visual by clicking on the dropdown that currently says "All Sites". The next slide will show an example of how we can change the chart.



Dashboard Dropdown Example

- This pie chart is showing the successful/unsuccessful split for the CCAFS LC-40 launch site. There are additional choices from the dropdown menu that you can choose in order to see this same split for the rest of the launch sites.



Example of Plotly Scatterplot

- This scatterplot shows the successful/unsuccessful split by payload for each booster version.
- There is a slider above the graph that allows the user to adjust the range of the payload. The next slide will show how the graph changes when the slider is changed.

Dashboard Slider Example

- This is the same chart from the previous slide, but with a narrowed range on the slider.
- This graph shows the successful/unsuccessful split by booster version while also limiting the payload mass to 3k – 6k kg.