IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Bradley Aidan Johnson
28/02/2024

Outline

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

Executive Summary

This data science project serves as a foundational analysis for "SpaceY," a company in the planning stage of launching missions. By examining factors contributing to the success of SpaceX launches, SpaceY aims to identify key variables that can enhance the likelihood of successful launches for their future missions.

Launch Site Selection: Analysis of SpaceX launches suggests that the choice of launch site plays a crucial role in mission success. Factors such as proximity to the equator, distance from populated areas, and regulatory considerations should be carefully evaluated when selecting launch sites to maximize success rates.

Payload Mass and Configuration: Understanding the impact of payload mass and configuration on launch outcomes is essential. Optimal payload design and mass distribution can mitigate risks associated with payload deployment and contribute to mission success. Analyzing historical data on payload characteristics and their correlation with mission outcomes can provide valuable insights for SpaceY's payload planning.

Time and Experience: Experience and time proved to be invaluable in improving the success rate of space missions. SpaceY should leverage lessons learned from SpaceX's journey to refine operational processes, enhance technological capabilities, and build a knowledgeable workforce. Investing in rigorous training programs and fostering a culture of continuous improvement can further bolster SpaceY's readiness for successful launches.

Next Steps: Moving forward, SpaceY intends to delve deeper into the analysis of historical launch data, incorporating additional factors such as weather conditions, launch vehicle reliability, and mission complexity. Collaborations with industry experts and regulatory bodies will be sought to ensure comprehensive planning and adherence to safety standards. Furthermore, SpaceY will prioritize the development of innovative technologies and operational strategies tailored to maximize mission success rates while mitigating risks.

Introduction

In the rapidly evolving landscape of space exploration, companies like "SpaceY" are emerging with ambitious plans to venture into the cosmos. As SpaceY embarks on its journey to join the ranks of spacefaring organizations, a comprehensive understanding of the factors influencing mission success is paramount. This data science project aims to provide SpaceY with actionable insights derived from the analysis of SpaceX launches, offering invaluable guidance for the planning and execution of its own space missions.

SpaceX, founded by Elon Musk in 2002, has rapidly established itself as a frontrunner in the space industry, revolutionizing space transportation with its innovative technologies and ambitious missions. Through a series of successful launches, including crewed missions to the International Space Station (ISS) and pioneering efforts in rocket reusability, SpaceX has reshaped the landscape of space exploration.

Against this backdrop, SpaceY emerges as a newcomer with aspirations to emulate SpaceX's success while carving its own path in the cosmos. With a focus on meticulous planning, technological innovation, and operational excellence, SpaceY seeks to chart a course towards achieving its spacefaring goals.

Launch Success Factors:

What are the key determinants of launch success in the context of SpaceX missions? Understanding the factors influencing mission outcomes, such as launch site selection, payload characteristics, and operational procedures, is crucial for SpaceY to optimize its own launch campaigns.

Risk Mitigation Strategies:

How can SpaceY mitigate risks associated with space missions to ensure a high likelihood of success? By analyzing historical data on mission failures and identifying common failure modes, SpaceY aims to develop robust risk mitigation strategies tailored to its unique operational requirements.

Operational Best Practices:

What lessons can SpaceY learn from SpaceX's operational practices and experiences? By examining SpaceX's journey, including successes, setbacks, and lessons learned, SpaceY seeks to glean insights into effective mission planning, execution, and continuous improvement.

By addressing these key questions, this data science project aims to equip SpaceY with the knowledge, insights, and strategic guidance necessary to navigate the complexities of space exploration and embark on a journey towards achieving its spacefaring ambitions.

Section 1

Methodology

Methodology

Executive Summary

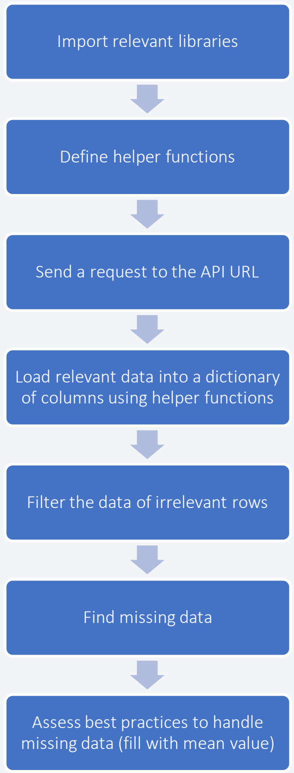
- Data collection methodology:
    - Data was collected from SpaceX directly and web scraped from related wiki pages
- Perform data wrangling
    - Data was wrangled to show landing outcome, missions by orbits and launches per site
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - 4 classification models were used and compared to improve accuracy

Data Collection

- Data was collected from the SpaceX API directly
- Data was also collected from web scraping the Wikipedia entries

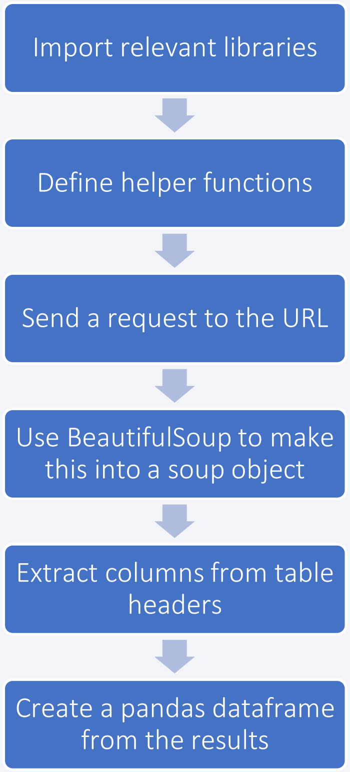
Data Collection – SpaceX API

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



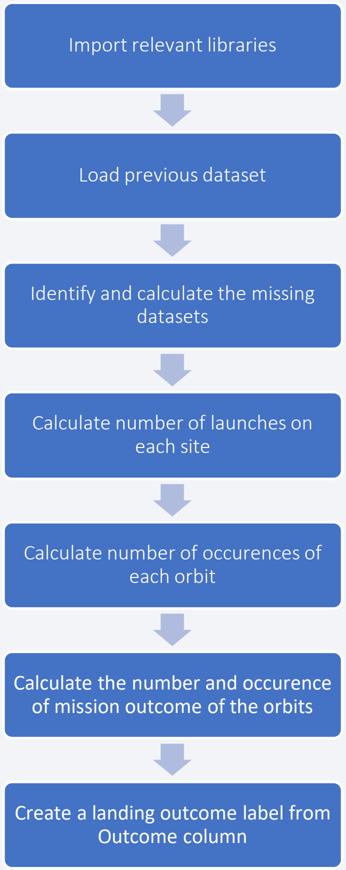
Data Collection - Scraping

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



Data Wrangling

https://github.com/bradleyaidanjohnson/SpaceY/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- We used:
- Scatter Plots
  - To see correlations in data (namely in launch sites against various factors)
- Bar Plots
  - To analyse categorical data (ie. Success rates of orbit types)
- Line Plots
  - To analyse trends over time (ie. Success rate)
- https://github.com/bradleyaidanjohnson/SpaceY/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- Queries executed:
    - Display the names of the unique launch sites in the space mission
    - Display 5 records where launch sites begin with the string 'CCA'
    - Display the total payload mass carried by boosters launched by NASA (CRS)
    - Display average payload mass carried by booster version F9 v1.1
    - List the date when the first successful landing outcome in ground pad was achieved.
    - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
    - List the total number of successful and failure mission outcomes
    - List the names of the booster versions which have carried the maximum payload mass. Use a subquery
    - List the records which will display the month names, failure landing outcomes in drone ship ,booster versions, launch site for the months in year 2015.
    - 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.

- https://github.com/bradleyaidanjohnson/SpaceY/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Folium Map Objects used:
  - Circles – to highlight launch sites
  - Markers – to provide labels for launch sites
  - MarkerClusters – to illustrate successful against unsuccessful launches at sites
  - PolyLine – to illustrate the distance between launch sites and places of interest (ie. Coastlines)

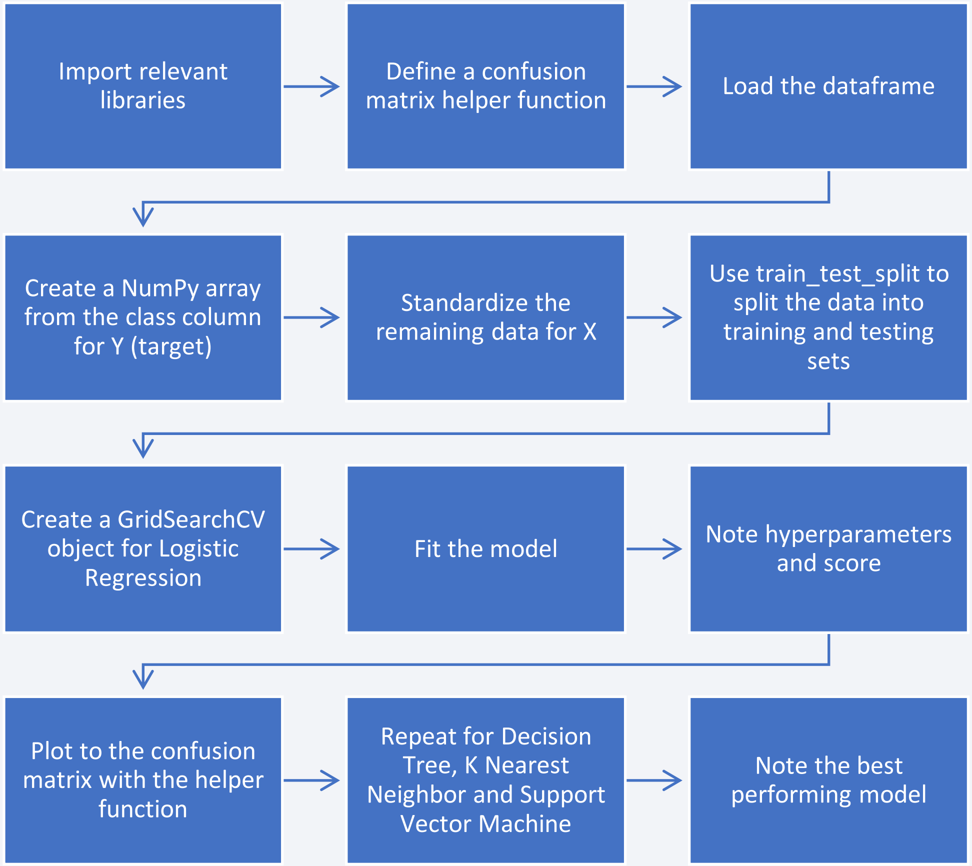
- https://github.com/bradleyaidanjohnson/SpaceY/blob/main/lab_jupyter_launch_site_location.jupyterlite%20(1).ipynb

Build a Dashboard with Plotly Dash

- We utilised the following graphing tools:
  - Pie Charts – to show launch successes by site over different time periods
  - Scatter Plots – to show launch successes by payload over different time periods
- Explain why you added those plots and interactions
- https://github.com/bradleyaidanjohnson/SpaceY/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- You need present your model development process using key phrases and flowchart
- https://github.com/bradleyaidanjohnson/SpaceY/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb



Results

We will now explore the results of these processes including:

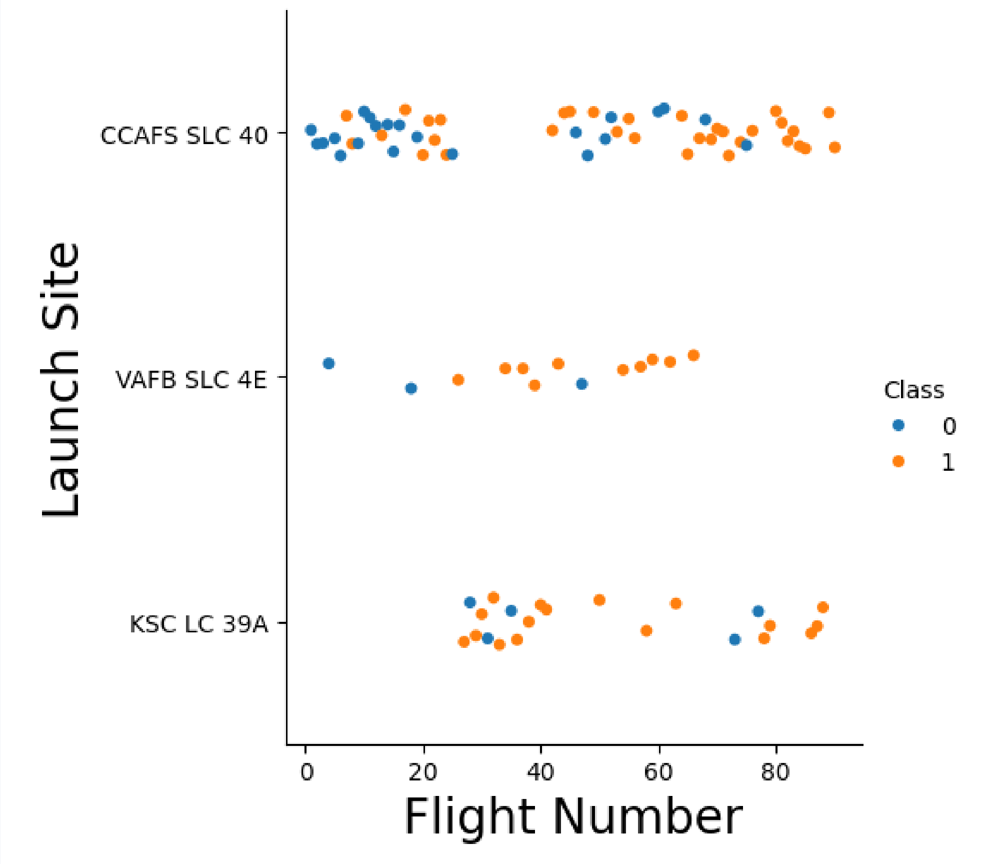
- Exploratory data analysis results
- Interactive analytics findings
- Predictive analysis results

Section 2

Insights drawn from EDA

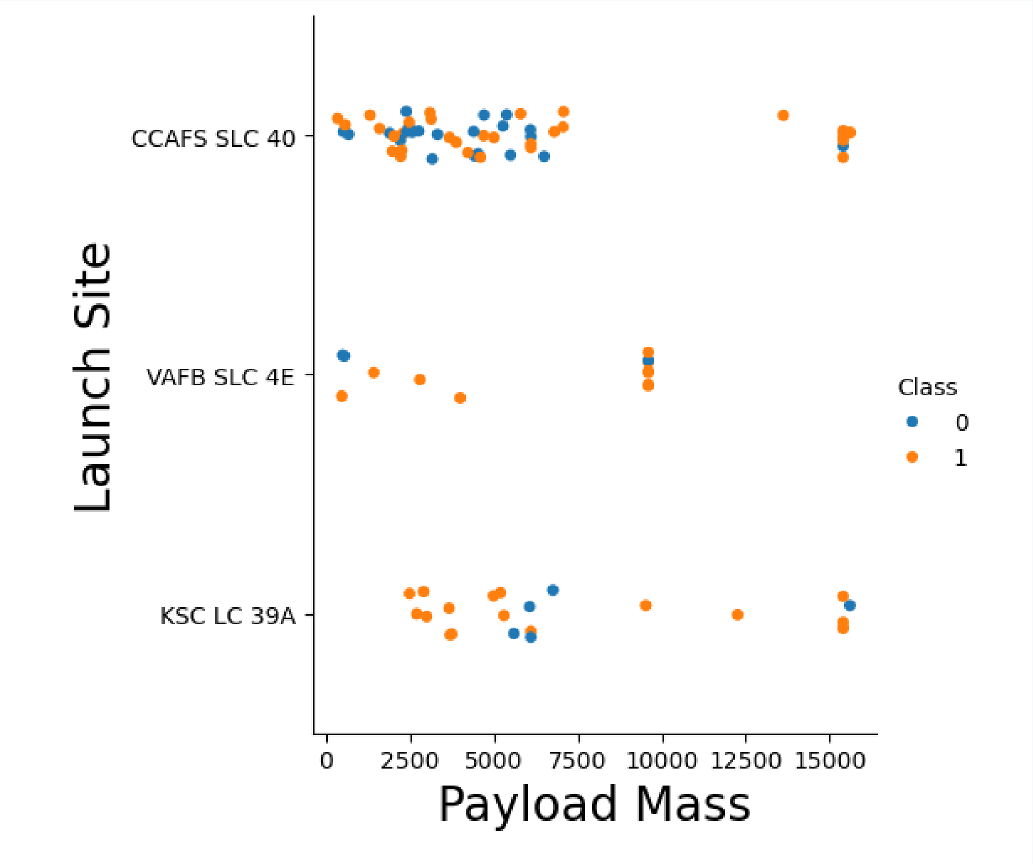
Flight Number vs. Launch Site

- Site VAFB was abandoned over time
- CCAFS is equally popular over time
- CCAFS had lower success rate than KSC



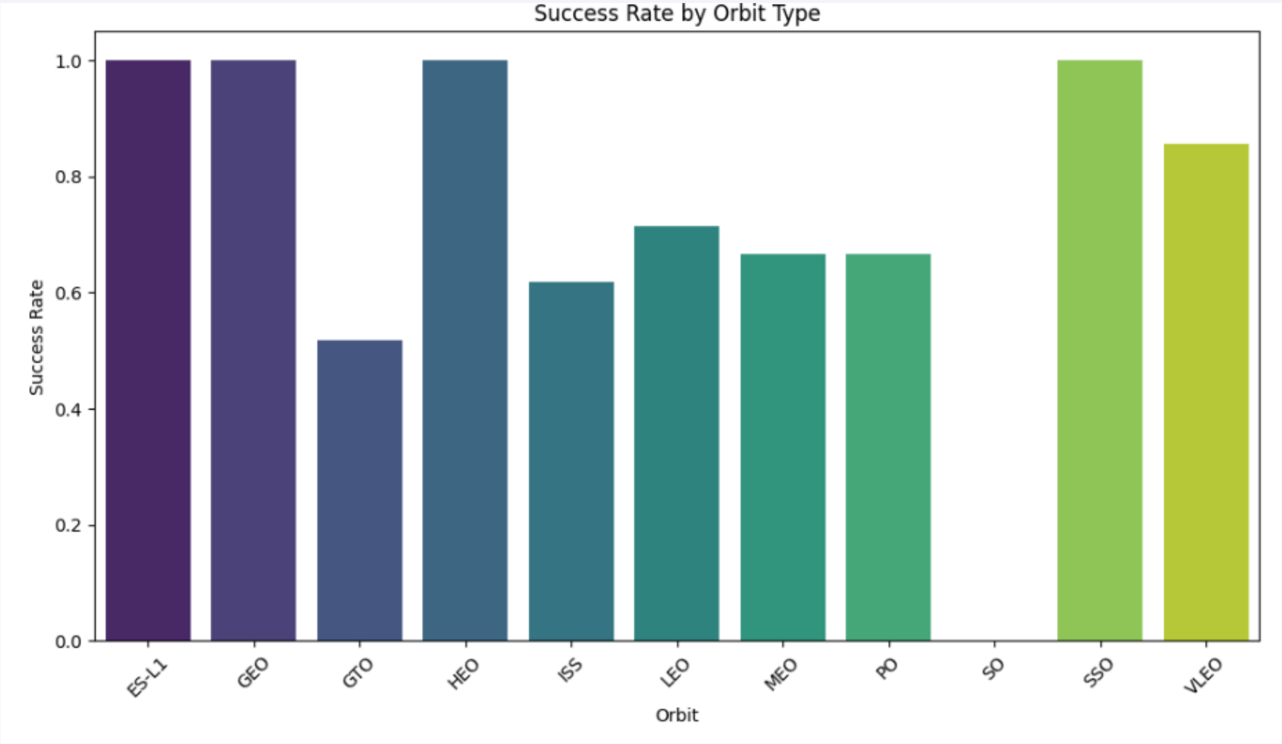
Payload vs. Launch Site

- CCAFS had a lower success rate with lower payloads
- VAFB had a high success rate
- KSC used all types of payload masses
- CCAFS mainly worked with lower payloads except for some exceptions



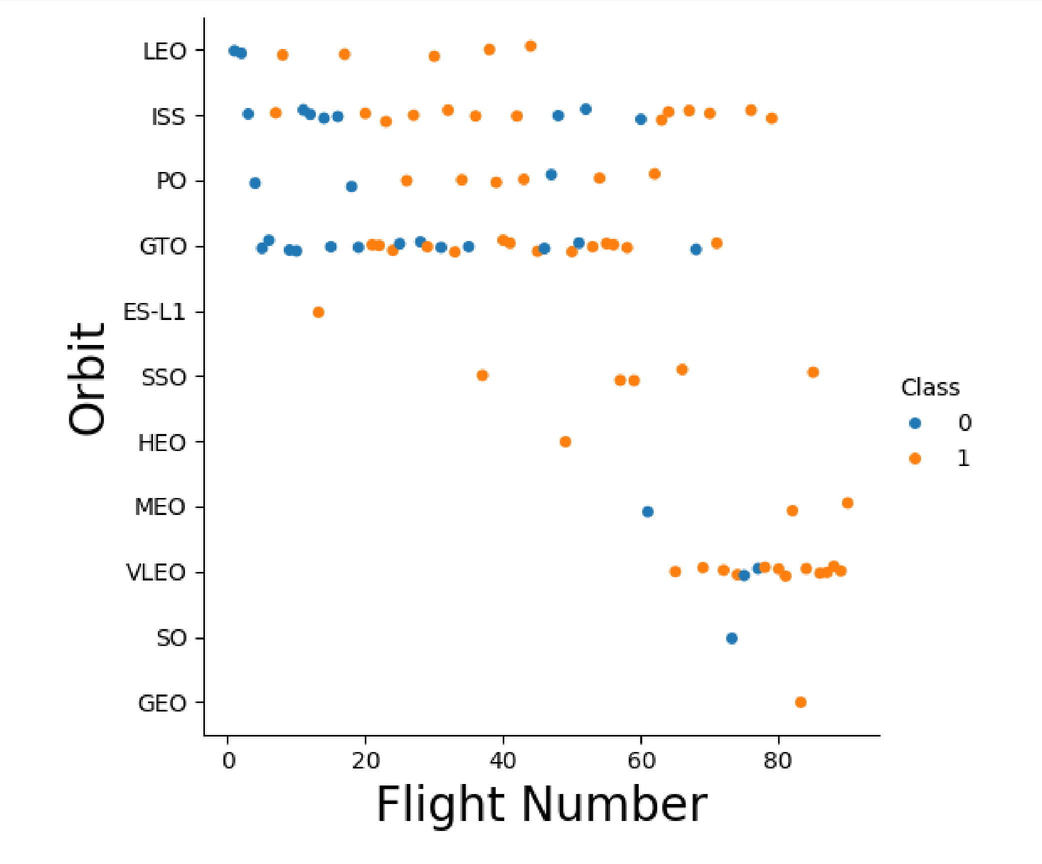
Success Rate vs. Orbit Type

- GTO/ISS/LEO/MEO/PO had middling success rates
- SO has never succeeded
- ES-L1, GEO, HEO, SSO and VLEO have high success rates



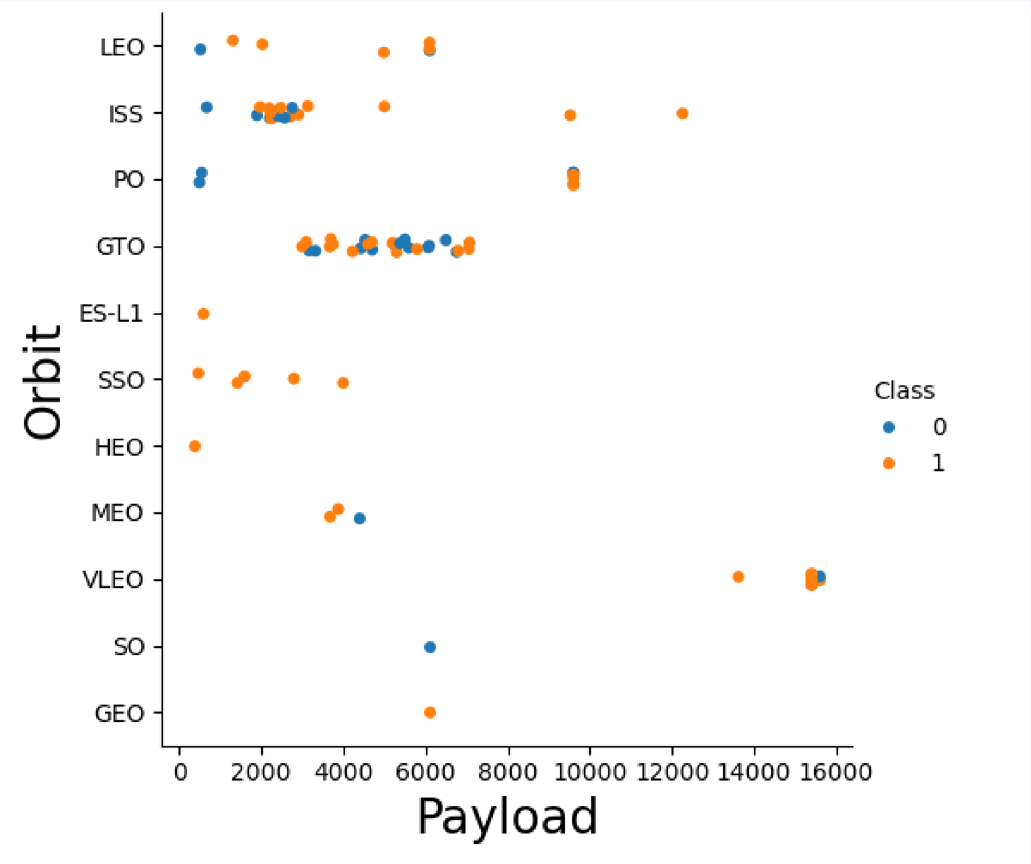
Flight Number vs. Orbit Type

- Success improved over time
- Orbit types (such as LEO/GTO) fall out of favour over time
- VLEO became incredibly popular later
- MEO/SSO are very promising
- The strongest correlation to success is time



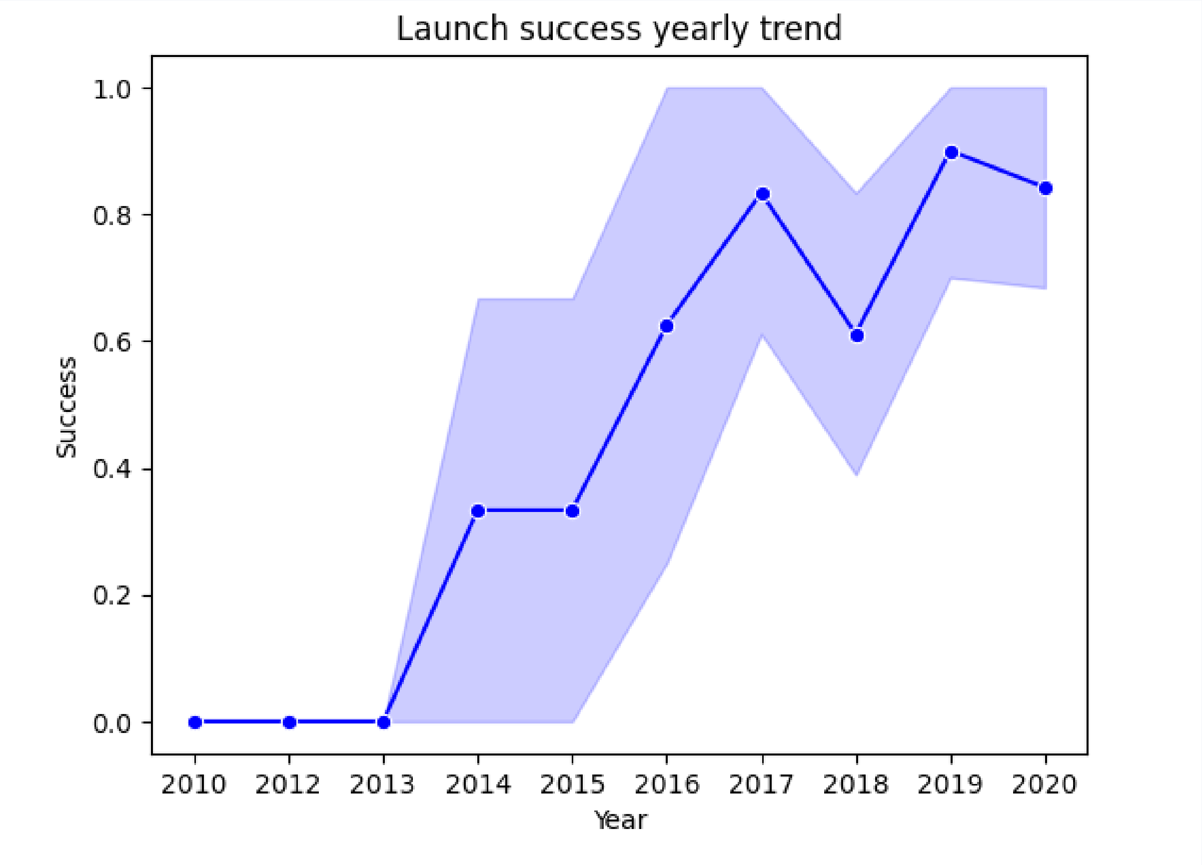
Payload vs. Orbit Type

- GTO works with mainly lower payloads and has a low success rate
- VLEO works with very large payloads to great success
- SSO works with very low payloads and has a perfect success rate
- ISS mainly works with lower payloads though has been success with average payloads



Launch Success Yearly Trend

- Since 2013 success has continued an upwards trend to the point now that launches are considerably more likely to succeed than to not.
- This shows adaptation is effective and is promising for SpaceY



All Launch Site Names

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

Launch Site Names Begin with 'CCA'

- This had the possibility of showing 2 launch sites

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

- 45596kg
- Nasa CRS total payload mass

Average Payload Mass by F9 v1.1

- 2534.6666666666665

First Successful Ground Landing Date

- 2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

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

Total Number of Successful and Failure Mission Outcomes

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

Boosters Carried Maximum Payload

- 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

• 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

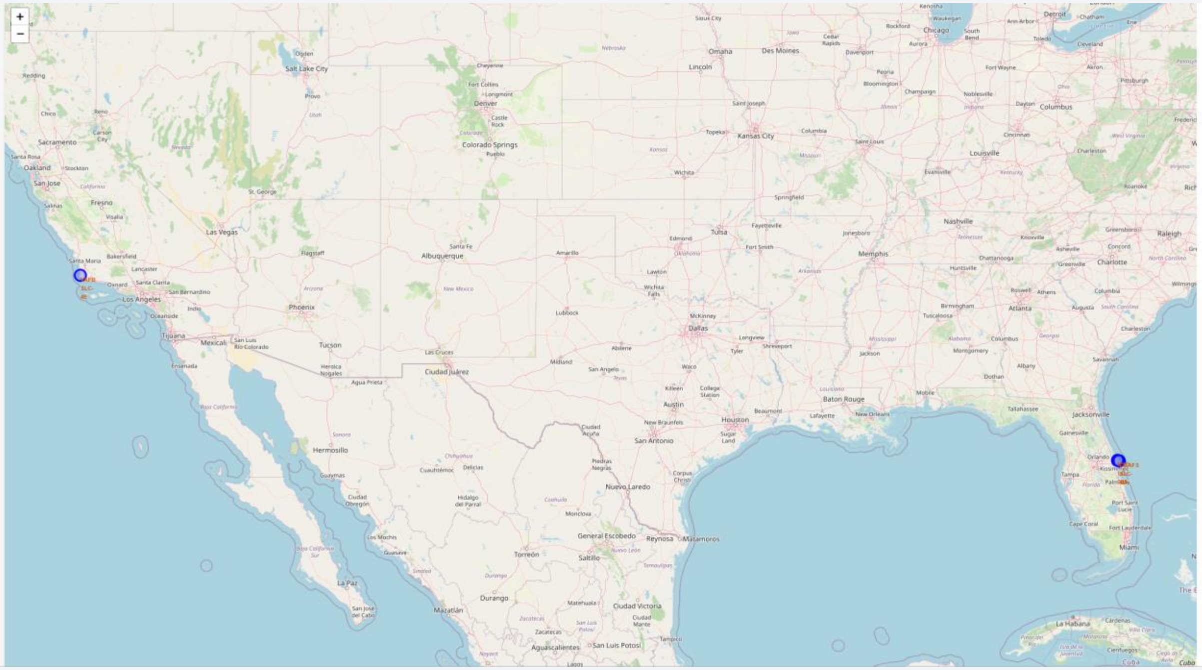
- Failure (drone ship)   5
- Success (ground pad)   3

Section 3

Launch Sites Proximities Analysis

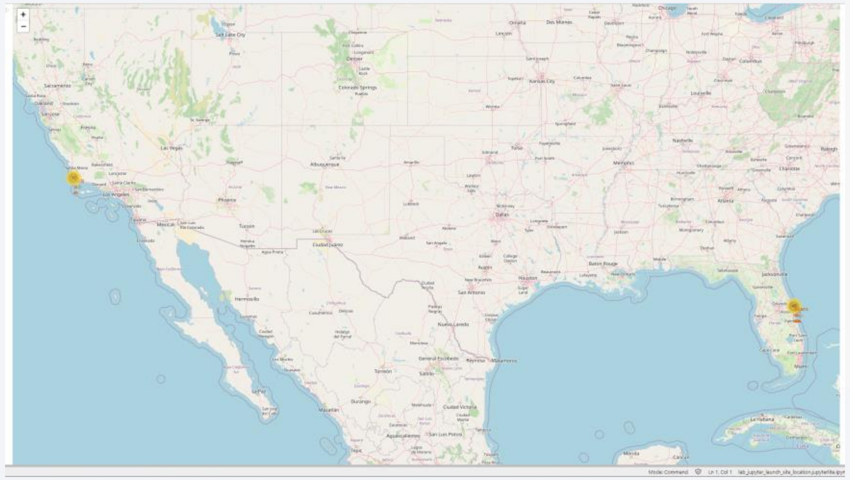
Launch Site Locations General View

- This map shows the proximity of launch sites to the equator
- Launch sites are near ocean coasts



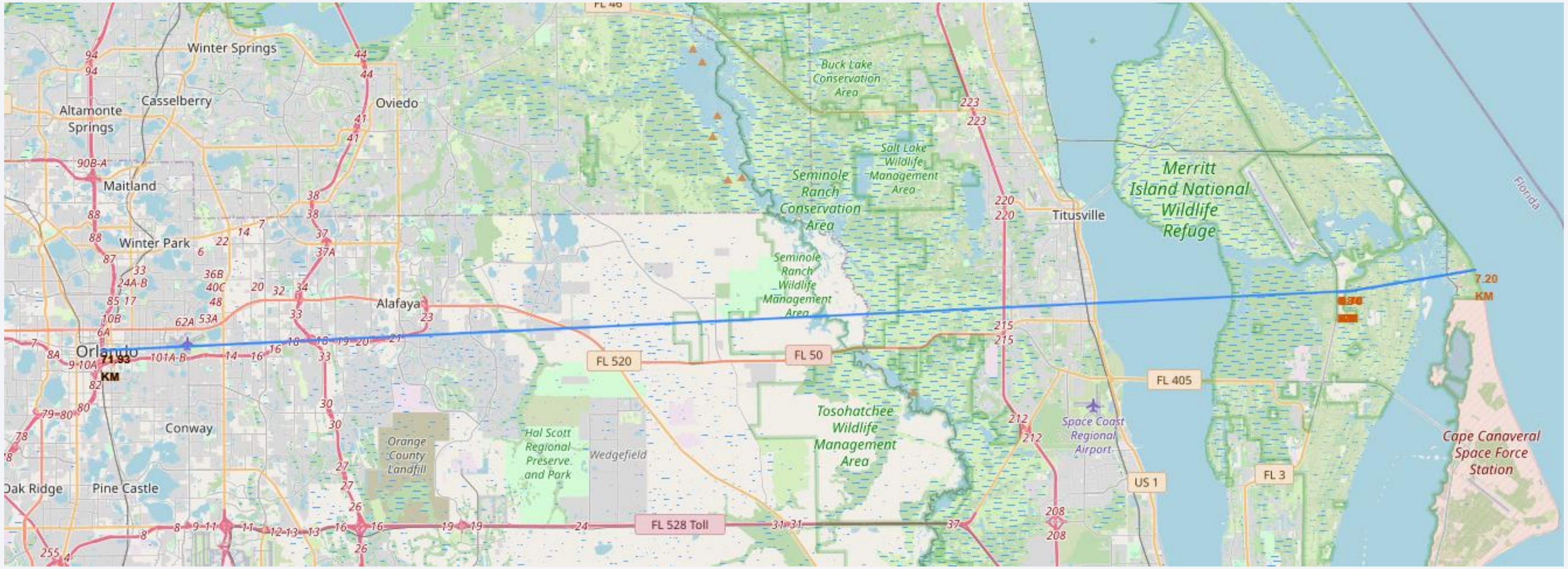
Launch Location Success Rate

- First map shows launches across the sites
- Second map showed the high success rate of launches at the KSC LC-39A site



Launch Site Locale

- The most successful launch site is extremely close to the coast, railway and highway
- It is in the vicinity of a major city. But far away enough to not interact

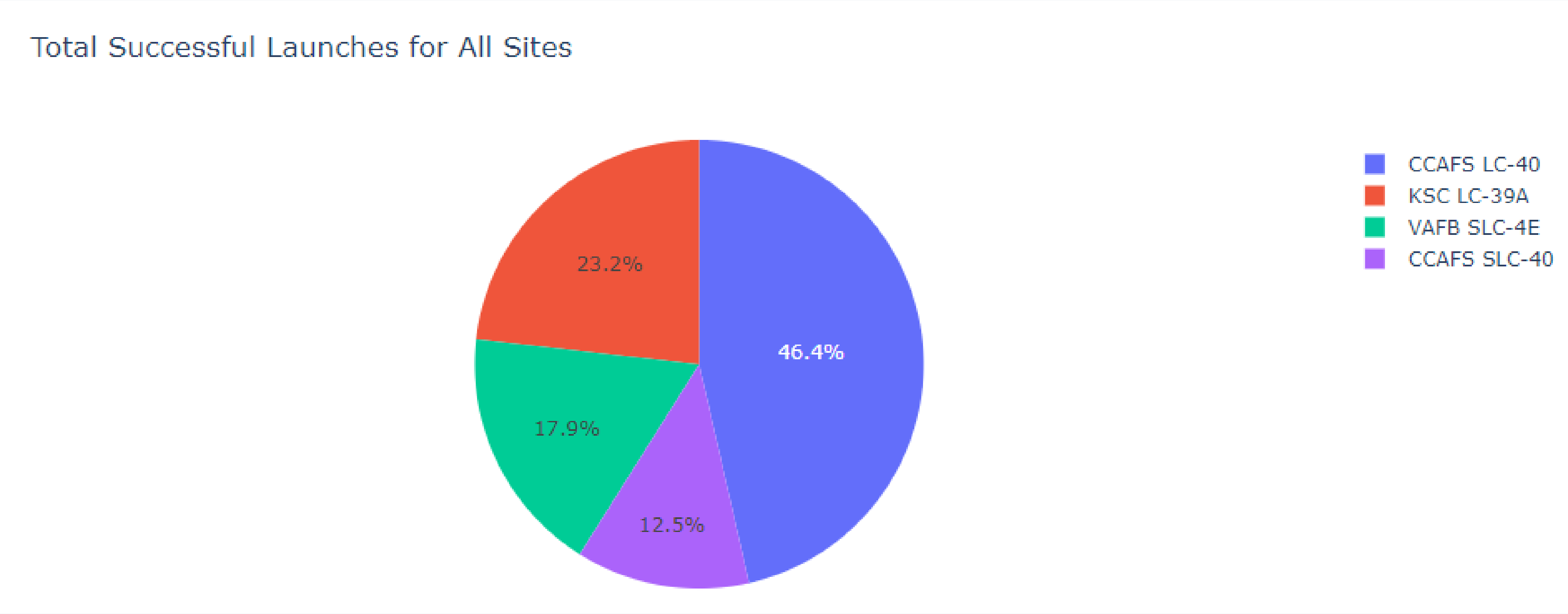


Section 4

Build a Dashboard with Plotly Dash

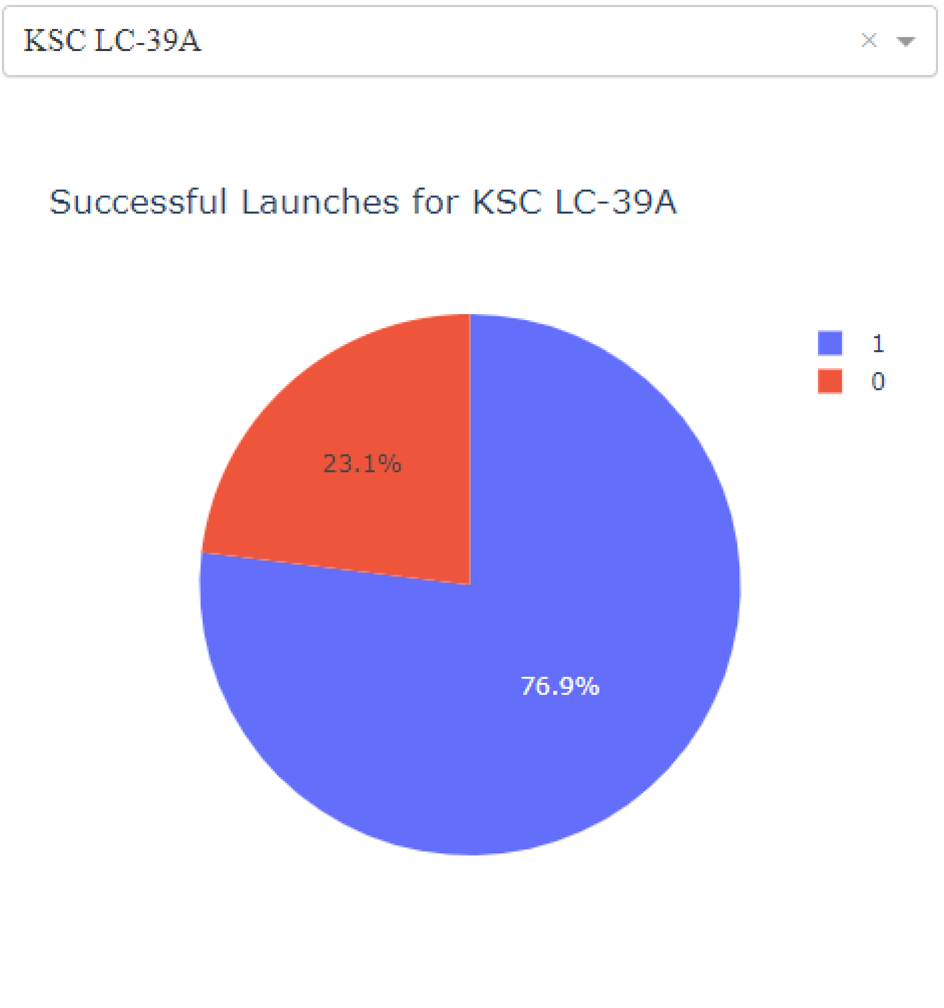
Total Successful Launches for All Sites

- This shows the most successful site is CCAFS LC-40 by a large margin



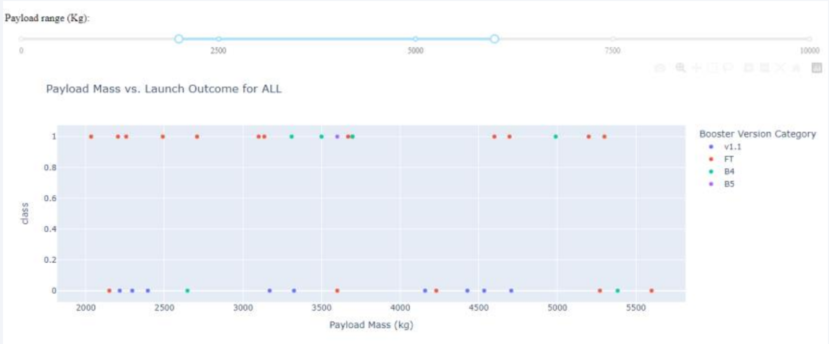
Site by Site Successful Launch Ratio

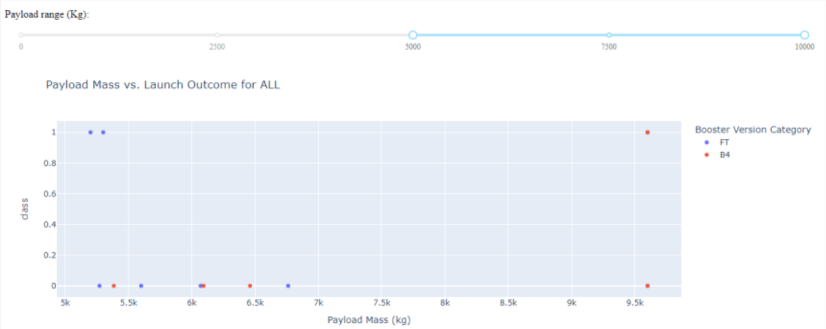
KSC LC-39A had the highest successful launch ratio

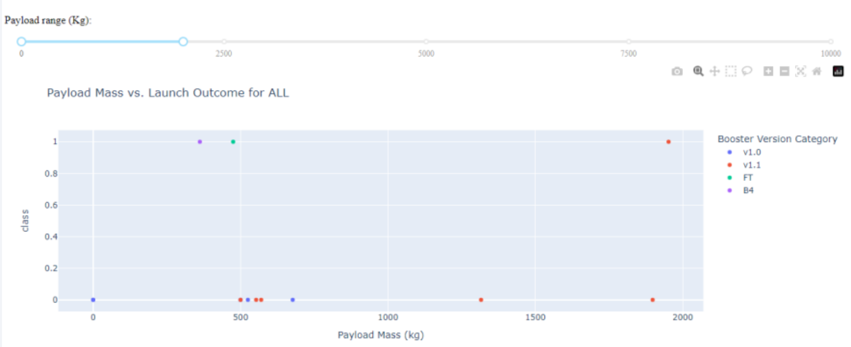


<Dashboard Screenshot 3>

- Most successful payload range:
  - 2000-6000kg
- Least Successful payload ranges:
  - 0-2000kg
  - 5-10,000kg





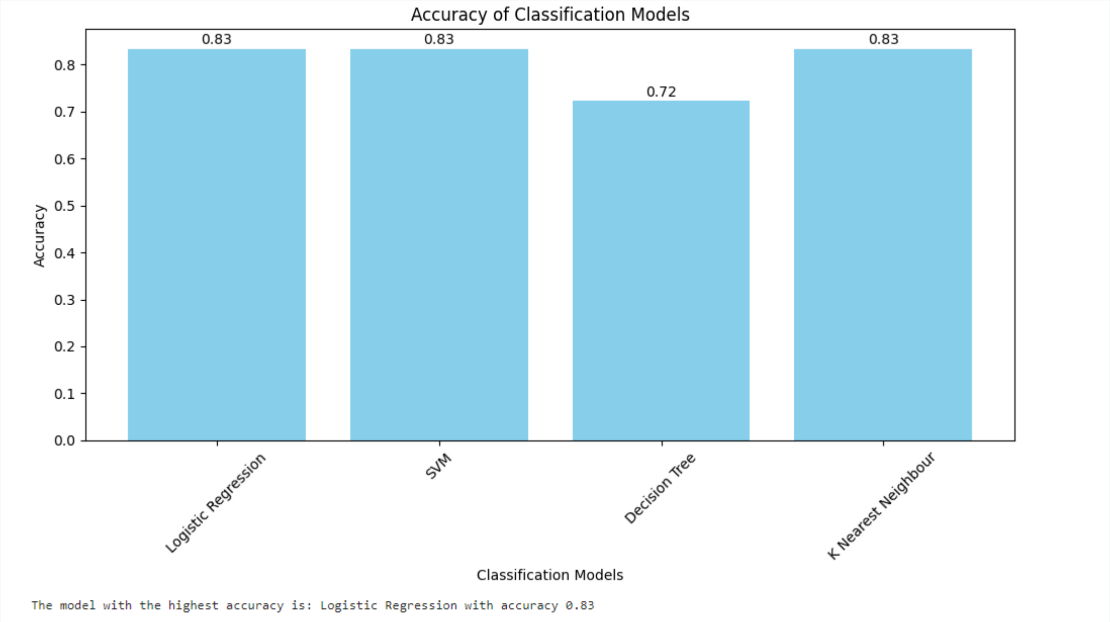


Section 5

Predictive Analysis (Classification)

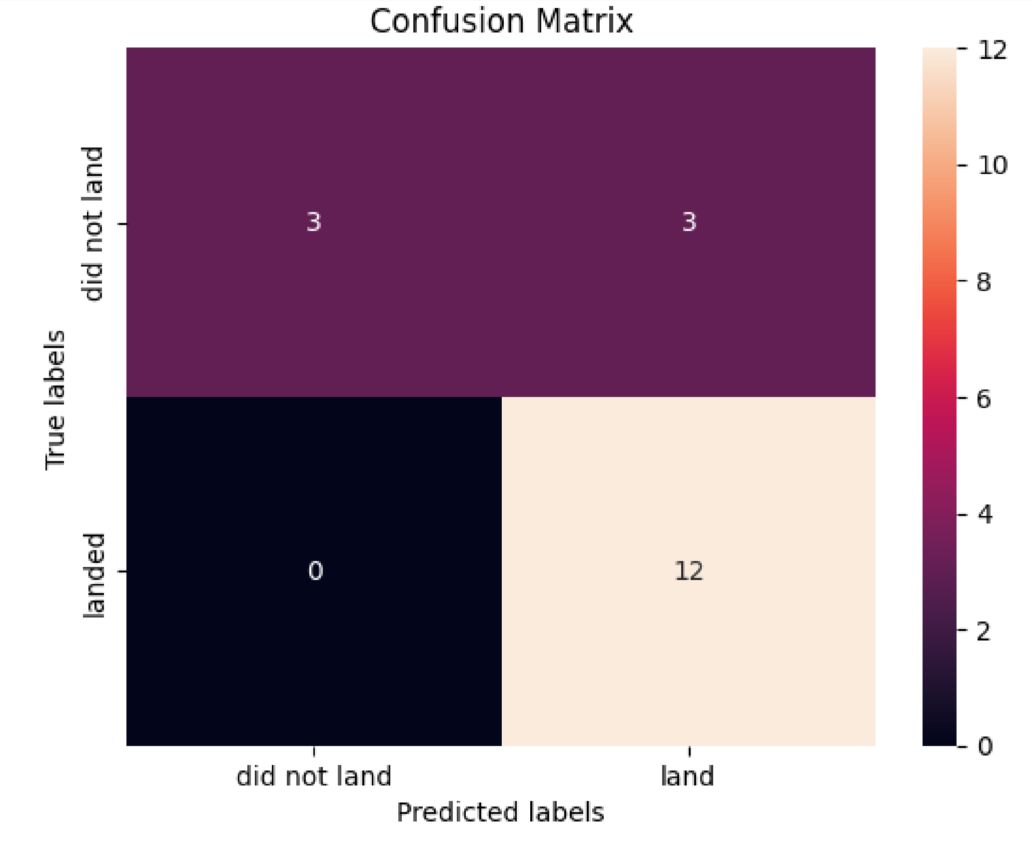
Classification Accuracy

- Whilst the model with the highest accuracy is Logistic Regression, the reality is there is no separation between any of the models except the low performing Decision Tree
- This is due to the limited test data (12). Machine Learning will always suffer with so few data points to learn/validate/test from



Confusion Matrix

- This showed the models had no issue predicting failures perfectly
- However there is an issue for false positives where the model predicted a success that in fact was a failure
- Realistically this is the worst case scenario for a model of this nature as it’s the most costly mistake
- My advice would be to limit use of the machine learning data till more data can be compiled
- Use the info as a guideline not a rule



Conclusions

- Launch sites should be near the equator, coastlines, railways and highways, not close to but in the vicinity of a major city
- Time to improve is the most reliable predictor of launch success
- ES-L1, GEO, HEO, SSO and VLEO have high success rates
- SO has never worked
- VLEO is a recent and very successful orbit type (it also works with very large payloads)
- MEO/SSO are very promising
- The site with most successes is CCAFS LC-40 by a large margin
- KSC LC-39A had the highest successful launch ratio
- The most successful payload range is 2000-6000kg
- We need more data for machine learning models

Appendix

- Useful code to iterate through launch sites for Folium

```
import folium
from folium.features import DivIcon
import html

# Initialize the map
site_map = folium.Map(location=[28.5, -80.5], zoom_start=5)  # Set initial location and zoom level

# Loop through each launch site
for x in range(1, 4):
    coordinates = [launch_sites_df.loc[x, 'Lat'], launch_sites_df.loc[x, 'Long']]
    name = launch_sites_df.loc[x, 'Launch Site']
    escaped_variable = html.escape(name)

    # Add circle marker
    folium.CircleMarker(coordinates, radius=10, color='blue', fill=True).add_to(site_map)
    # Add text marker
    folium.Marker(coordinates, icon=DivIcon(icon_size=(20,20), icon_anchor=(0,0), html=f'<div style="font-size: 12; color

# Display the map
site_map
```

- Useful addition to handle null values in web scraping

```
# Launch outcome
if row[7].string:
    launch_outcome = row[7].string.strip()
else:
    launch_outcome = None
launch_dict["Launch outcome"].append(launch_outcome)
```

Appendix

- Necessary query to make SqLite handle months

```
%%sql
SELECT
    CASE
        WHEN SUBSTR(Date, 6, 2) = '01' THEN 'January'
        WHEN SUBSTR(Date, 6, 2) = '02' THEN 'February'
        WHEN SUBSTR(Date, 6, 2) = '03' THEN 'March'
        WHEN SUBSTR(Date, 6, 2) = '04' THEN 'April'
        WHEN SUBSTR(Date, 6, 2) = '05' THEN 'May'
        WHEN SUBSTR(Date, 6, 2) = '06' THEN 'June'
        WHEN SUBSTR(Date, 6, 2) = '07' THEN 'July'
        WHEN SUBSTR(Date, 6, 2) = '08' THEN 'August'
        WHEN SUBSTR(Date, 6, 2) = '09' THEN 'September'
        WHEN SUBSTR(Date, 6, 2) = '10' THEN 'October'
        WHEN SUBSTR(Date, 6, 2) = '11' THEN 'November'
        WHEN SUBSTR(Date, 6, 2) = '12' THEN 'December'
    END AS MonthName,
    Landing_Outcome,
    Booster_Version,
    Launch_Site
FROM
    SPACEXTBL
WHERE
    SUBSTR(Date, 0, 5) = '2015'
    AND Landing_Outcome LIKE '%Failure (drone ship)%';
```

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

MonthName	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

Parameter edits to make the decision tree code work

```
parameters = {
    'criterion': ['gini', 'entropy'],
    'splitter': ['best', 'random'],
    'max_depth': [2*n for n in range(1, 10)],
    'max_features': ['auto', 'sqrt', 'log2', None],
    'min_samples_leaf': [1, 2, 4],
    'min_samples_split': [2, 5, 10]
}

tree = DecisionTreeClassifier()
tree_cv = GridSearchCV(tree, parameters, cv=10)
```

```
# Fit the GridSearchCV object to find the best parameters
tree_cv.fit(X_train, Y_train)
```

Thank you!