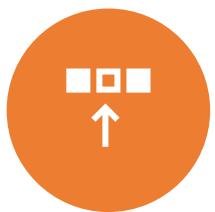


Winning Space Race with Data Science

Lawrence Jacobs
February 10, 2024



Outline



EXECUTIVE
SUMMARY



INTRODUCTION



METHODOLOGY



RESULTS



CONCLUSION



APPENDIX

Executive Summary

- Summary of methodologies
- Used SpaceX API
- WIKI WebScraping
- Python Visual Library
- EDA SQL
- Folium
- Summary of all results
- SpaceY can successfully use the SpaceX model to launch and return Rockets



Introduction

- SpaceY was inquiring about the SpaceX approach to launch and reuse rockets
- The goal of this research is to determine if the SpaceX had a good or bad success rate on their launch attempts and what the parameters of each launch was in terms of success or failure

Section 1

Methodology

Methodology

6

Executive Summary

Data collection methodology:

- Data was collected using the SpaceX database

Perform data wrangling

- Data was processed using Python analytical tools and Machine Learning

Performed exploratory data analysis (EDA) using visualization and SQL

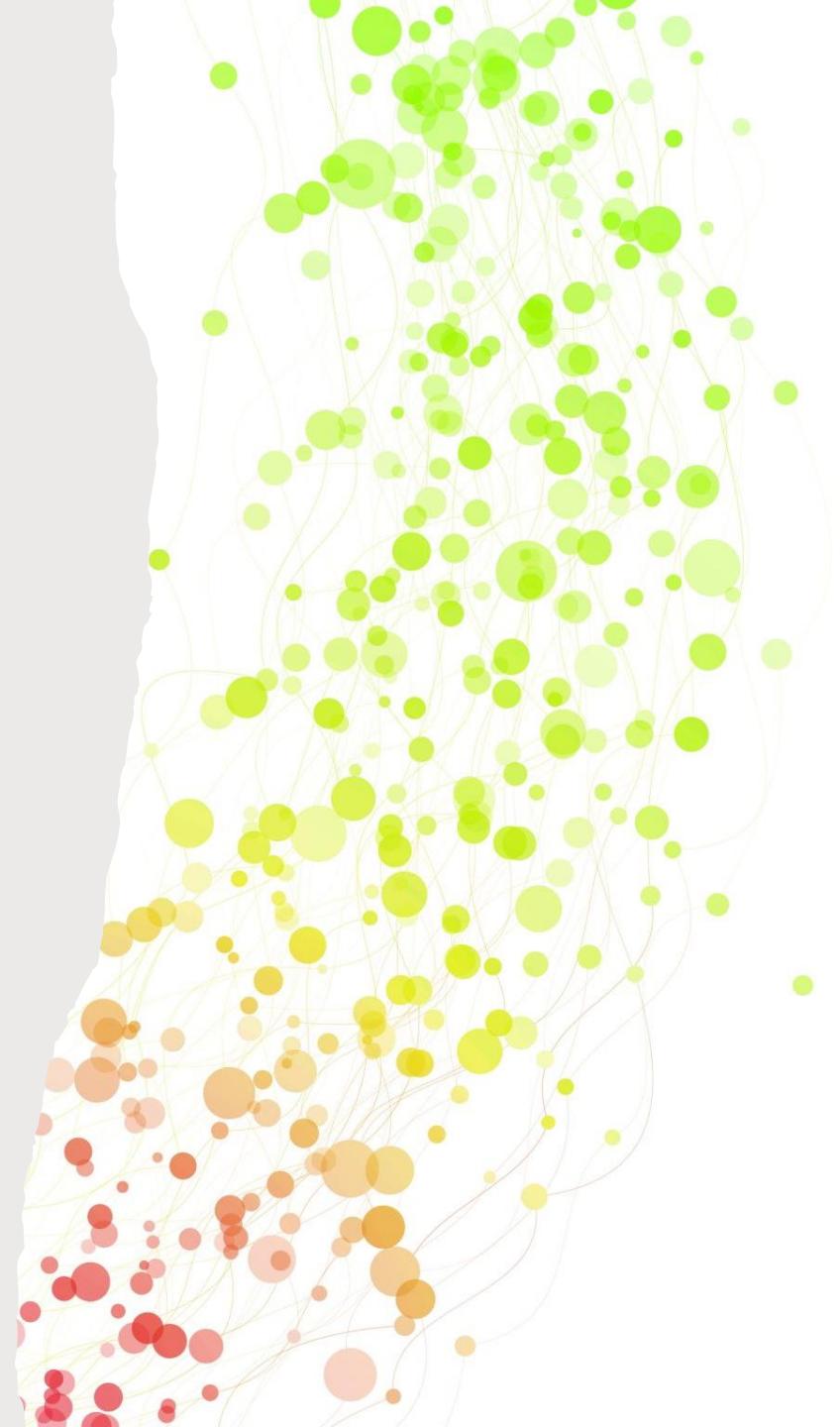
Performed interactive visual analytics using Folium and Plotly Dash

Performed predictive analysis using classification models

- How to build, tune, evaluate classification models

Data Collection

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



Data Collection – SpaceX API

Present your data collection with SpaceX REST calls using key phrases and flowcharts

Add the GitHub URL of the completed SpaceX API calls notebook ([must include completed code cell and outcome cell](#)), as an external reference and peer-review purpose

Place your flowchart of SpaceX API calls here

Data Collection - Scraping

We used the SpaceX Api call

Cleaned the data using Panda and Numpy

Created a Dataframe that we could manipulate

<https://github.com/kingHannibal66/SpaceY/blob/main/jupyter-labs-spacex-data-collection-api.ipynb>

<https://github.com/kingHannibal66/SpaceY/blob/main/jupyter-labs-spacex-data-collection-api.ipynb>

Data Wrangling

- Using the Dataset we
- Calculated the number of launches from each SpaceX site
- Calculated the number of missions and the outcome of those missions
- <https://github.com/kingHannibal66/SpaceY/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb>

EDA with Data Visualization

- Used Matplotlib to chart the differences in Payload mass, Lauchsite, Flightnumber, Orbit of launch, and success or failure of mission
- <https://github.com/kingHannibal66/SpaceY/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb>



EDA with SQL

- Some of the SQL queries performed are
- Create table. to display SpaceX information
- Select sum. to display the Payloadmass
- Select avg. to average the payload mass carried by each mission
- Select count. to display the success or failure of the missions
- https://github.com/kingHannibal66/SpaceY/blob/main/jupyter-labs-eda-sql-coursera_sqlite.ipynb



Build an Interactive Map with Folium

- Used circles show the launch sites and successful/ failed launches
- Lines to show distance from major cities and highways to SpaceX launchsites
- https://github.com/kingHannibal66/SpaceY/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- Created a SpaceX launch dashboard that filters all launches with the success and failure rate of each launch vehicle and site and a payload mass plot
- https://github.com/kingHannibal66/SpaceY/blob/main/spacex_dash_app.py



Predictive Analysis (Classification)

- Used Machine learning(ML) prediction to create , train, and test 3 different ML algorithms to find the best fit for the data.
- The algorithms used where KNN, Decision Tree Classifier, and Logistic regression.
- https://github.com/kingHannibal66/SpaceY/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

- Decision Tree Classifier had the best outcome at 87%

The screenshot shows a GitHub browser interface with a tab for a JupyterLite notebook. The notebook title is "SpaceY / SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb". The content of the notebook includes:

```
In [41]: algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_}
bestalgorithm = max(algorithms, key=algorithms.get)
print('Best Algorithm is ',bestalgorithm,'with a score of ',algorithms[bestalgorithm])
if bestalgorithm == 'Tree':
    print('Best Params is : ',tree_cv.best_params_)
if bestalgorithm == 'KNN':
    print('Best Params is : ',knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best Params is : ',logreg_cv.best_params_)
```

Below the code, there are sections for "Authors" and "Change Log".

Authors: Pratiksha Verma

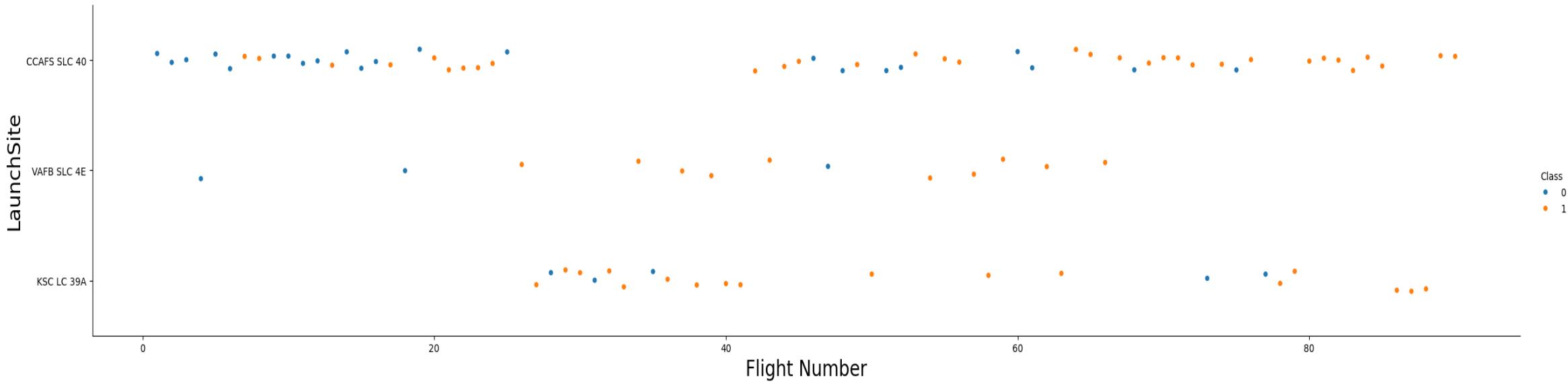
Change Log:

Date (YYYY-MM-DD)	Version	Changed By	Change Description

The background of the slide features a complex, abstract digital visualization. It consists of numerous thin, glowing lines that create a sense of depth and motion. The lines are primarily blue and red, with some green and purple highlights. They form a grid-like structure that curves and twists across the frame, resembling a three-dimensional space or a network of data points. The overall effect is futuristic and dynamic.

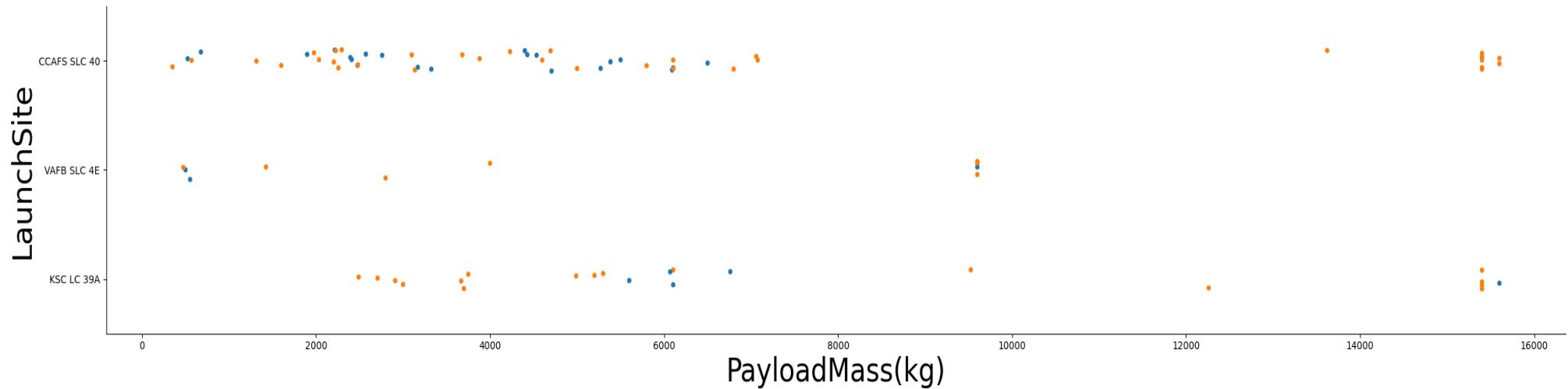
Section 2

Insights drawn from EDA



Flight Number vs. Launch Site

- 0 = Failure(blue)
- 1= Success(Orange)

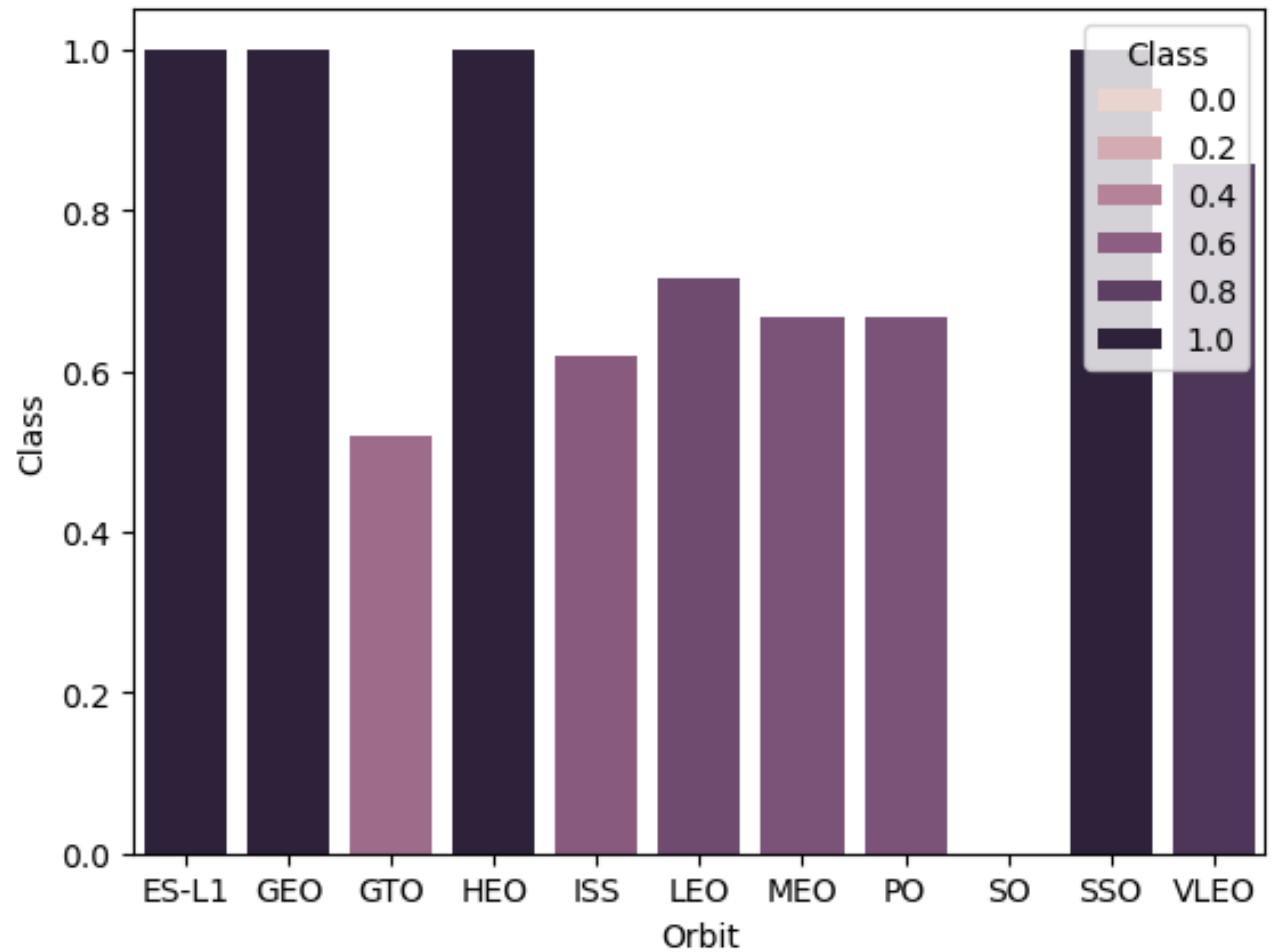


Payload vs. Launch Site

- No launches for a payload greater than 10,000kg

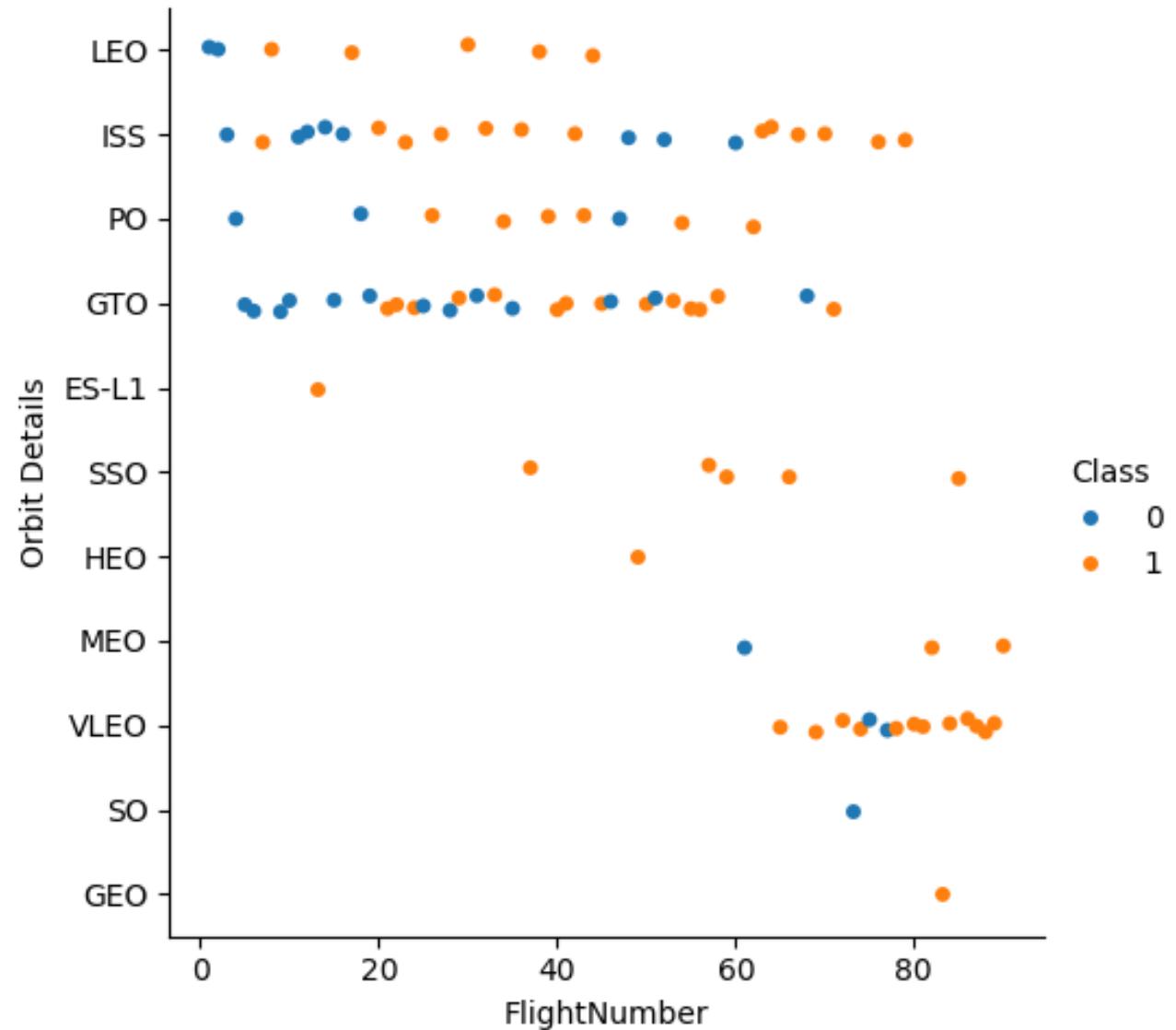
Success Rate vs. Orbit Type

- A graph showing which orbits have the highest success rate



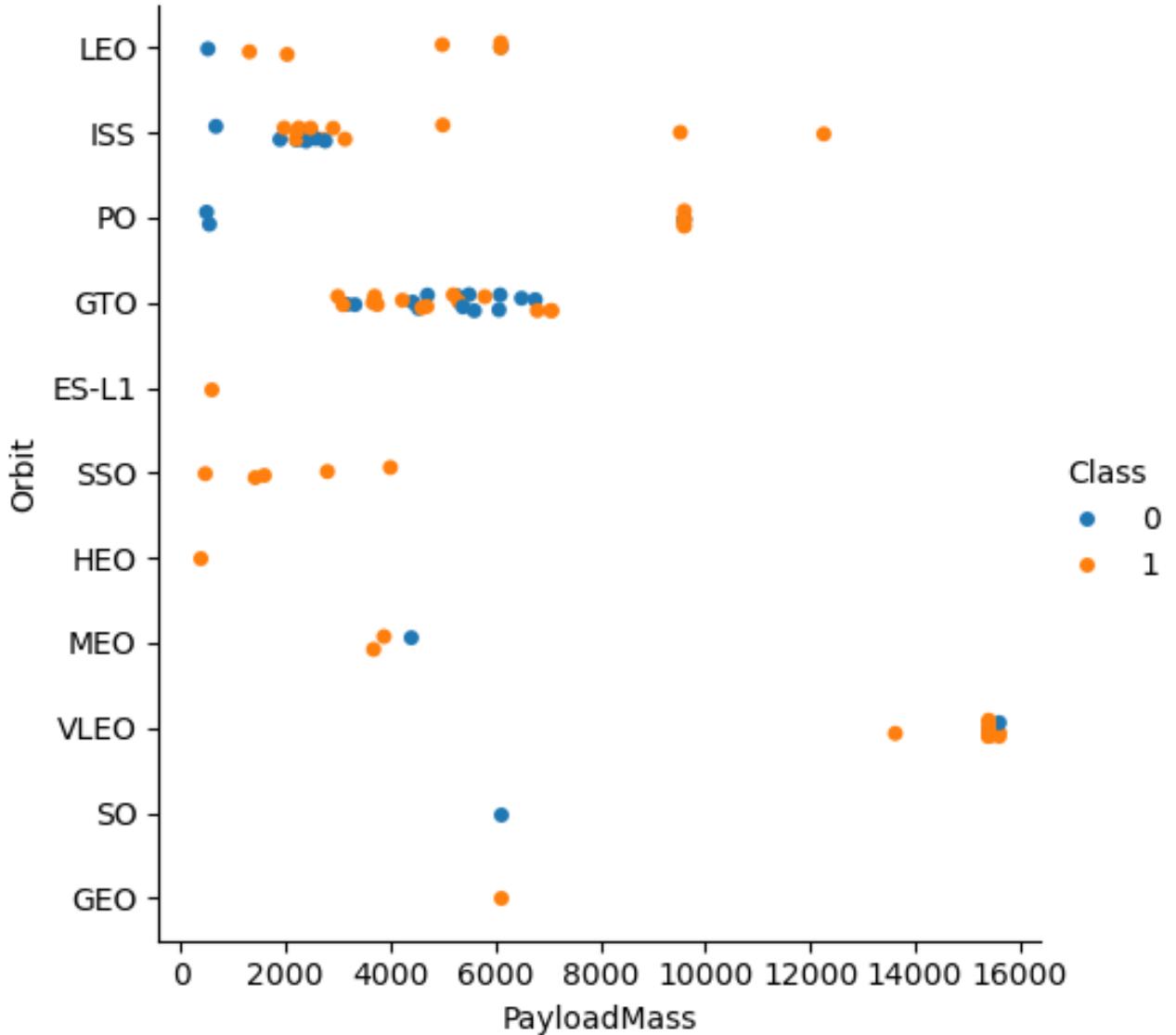
Flight Number vs. Orbit Type

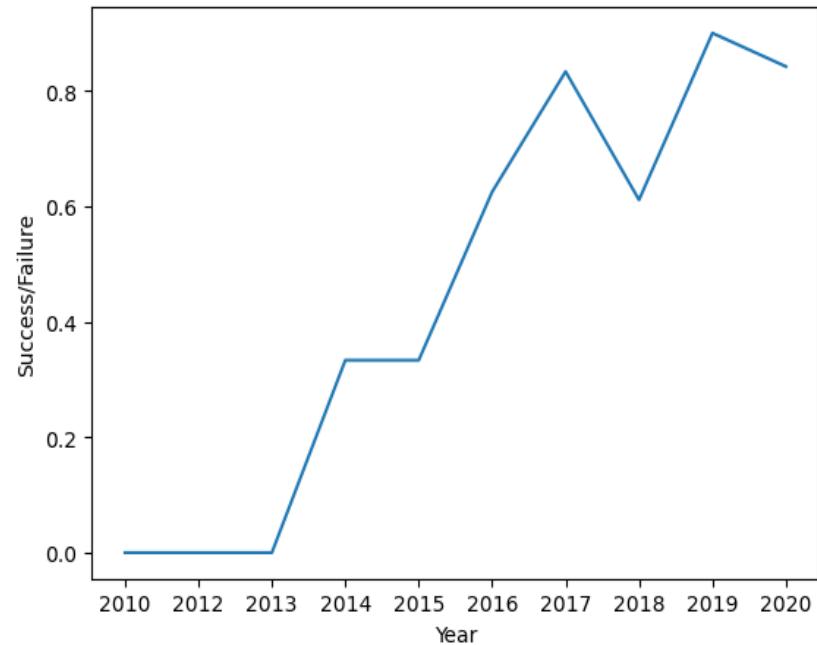
- You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



Payload vs. Orbit Type

- How payload mass and orbit correlate





Launch Success Yearly Trend

- SpaceX has been more successful with launches every year

All Launch Site Names

- SELECT DISTINCT(Launch_Site)
From SPACEXTBL
- Selected distinct launch sites
- Results:

CCAFS LC-40

VAFB SLC-4E

KSC LC-39A

CCAFS SLC-40

main SpaceY / jupyter-labs-eda-sql-coursera_sqlite.ipynb ↑ Top

Preview Code Blame 985 lines (985 loc) · 26.6 KB Code 55% faster with GitHub Copilot Raw ⌂ ⌄ ⌅ ⌆

Done.

Out[65]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYOUT_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

Launch Site Names
Begin with 'CCA'

- SELECT * FROM SPACEXTBL
- WHERE (Launch_Site) LIKE 'CCA%' LIMIT 5;
- 5 CCA Launch sites



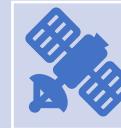
Total Payload Mass



```
select sum(PAYLOAD_MASS__KG_)  
as payloadmass from SPACEXTBL
```



```
where Customer='NASA (CRS)';
```



Total payload mass was 45596kg

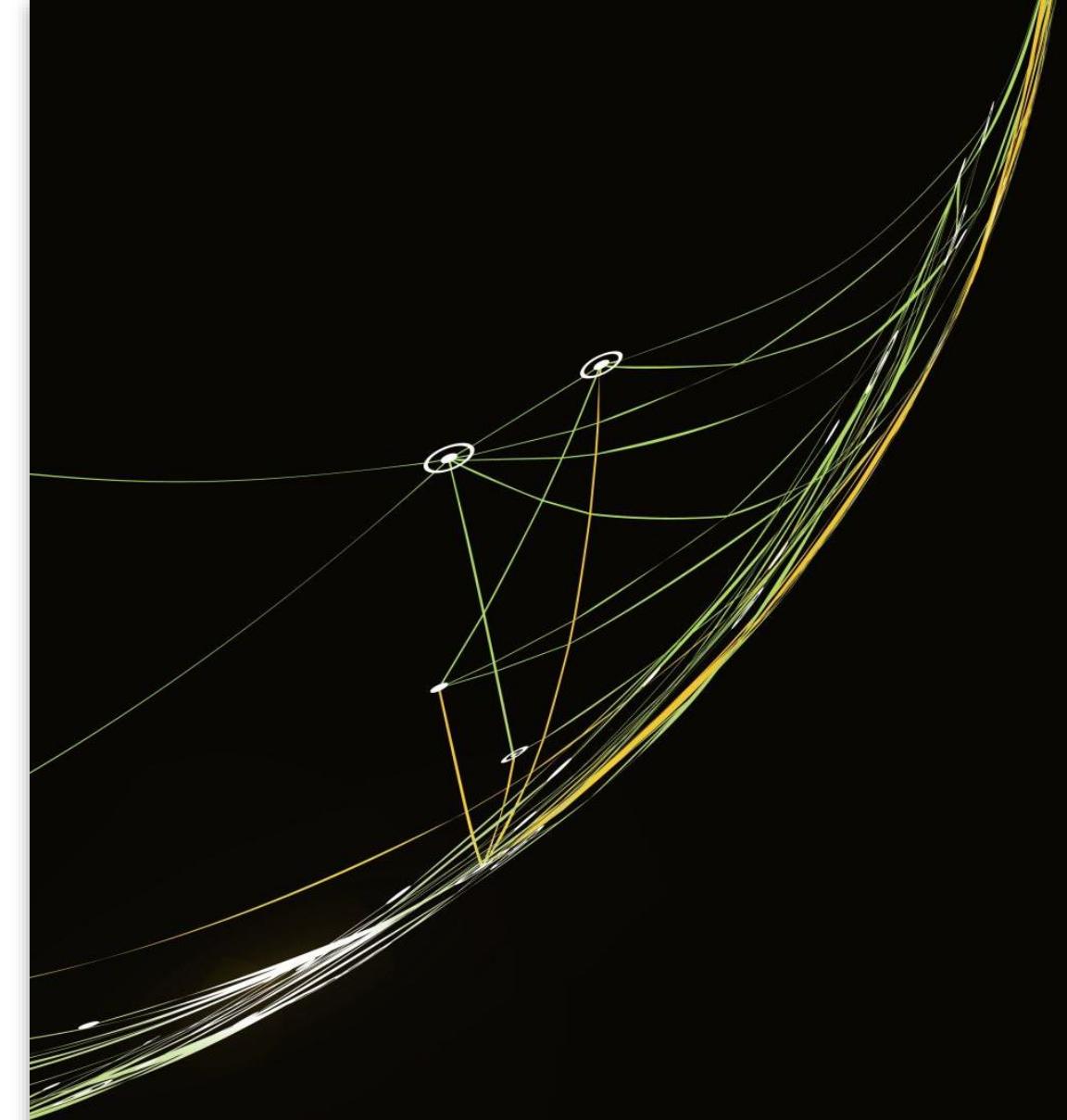


Average Payload Mass by F9 v1.1

- select avg(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTBL
- where Booster_Version = 'F9 v1.1';
- Avg Payload mass = 2928.4

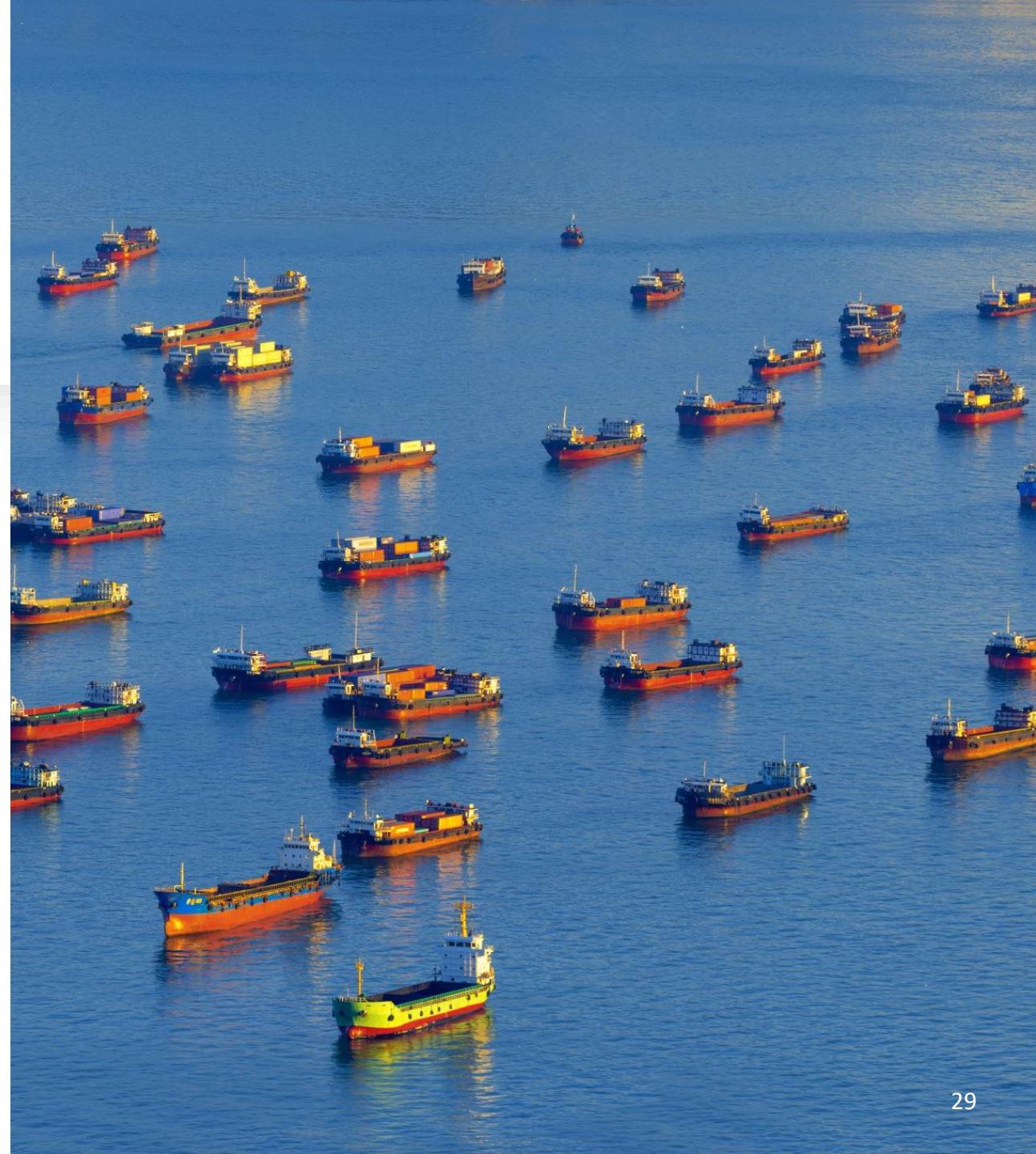
First Successful Ground Landing Date

- sql select min(DATE) from SPACEXTBL
- where Landing_Outcome= 'Success(ground pad)';
- NA



Successful Drone Ship Landing with Payload between 4000 and 6000

- sql select Booster_Version from SPACEXTBL
- where Landing_Outcome ='Success(drone ship)' and PAYLOAD_MASS_KG_>4000 and PAYLOAD_MASS_KG_<6000;
- NA



Total Number of Successful and Failure Mission Outcomes

```
sql select count  
(MISSION_OUTCOME) as  
missionoutcomes from SPACEXTBL
```

```
group by Mission_Outcome  
='Success' or Mission_Outcome =  
'Failure'
```

Successful missions = 98

Failed missions =3

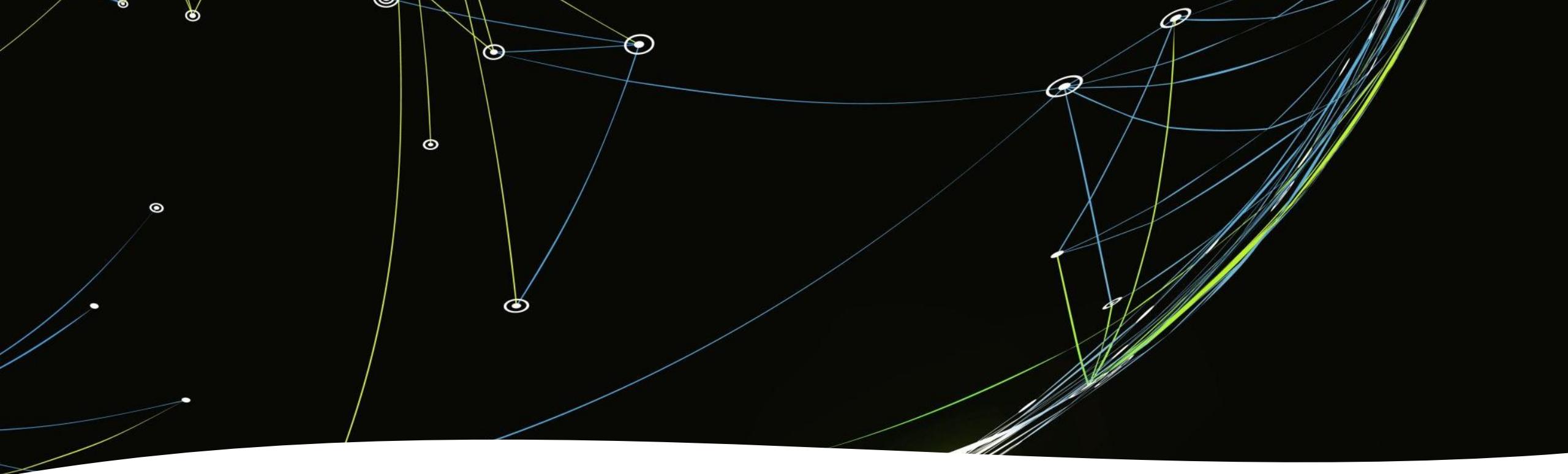
Boosters Carried Maximum Payload

- sql select BOOSTER_VERSION as boosterversion from SPACEXTBL
- where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTBL);
- boosterversion
- 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

- sql select 'Booster_Version','Launch_Site' from SPACEXTABLE
- where 'Landing_Outcome' = 'Failure(drone ship)'
- and substr(Date,1,4) = '2015';
- NA





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

- sql select 'LANDING_OUTCOME', count(*) as 'count' from SPACEXTBL
- where substr(Date,1,4) || substr(Date,6,2) || substr(date,9,2)
- between '20100604' and '20170320'
- group by 'Landing_Outcome' order by 'count' Desc;
- Landing Outcome = 31

The background of the slide is a photograph taken from space at night. It shows the curvature of the Earth's horizon against a dark blue sky. Numerous glowing yellow and white points represent city lights, concentrated in coastal and urban areas. In the upper right quadrant, there are bright green and yellow bands of light, likely the Aurora Borealis or Australis. The overall atmosphere is dark and mysterious.

Section 3

Launch Sites Proximities Analysis

Jupyter Notebook Viewer X JupyterLite X Peer Review: Submit your Work X ds-capstone-template-coursera X

https://nbviewer.org/github/kingHannibal66/SpaceY/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

YouTube Coinbase Coursera | Online C... SQL Server Functions SQL Keywords Refe... Keyboard shortcuts... 350 Excel Functions...

+ -

Launch Site Analysis

- It's important to note that all the launch sites are near the coastline. Most likely to have rockets launch over unpopulated areas.

35

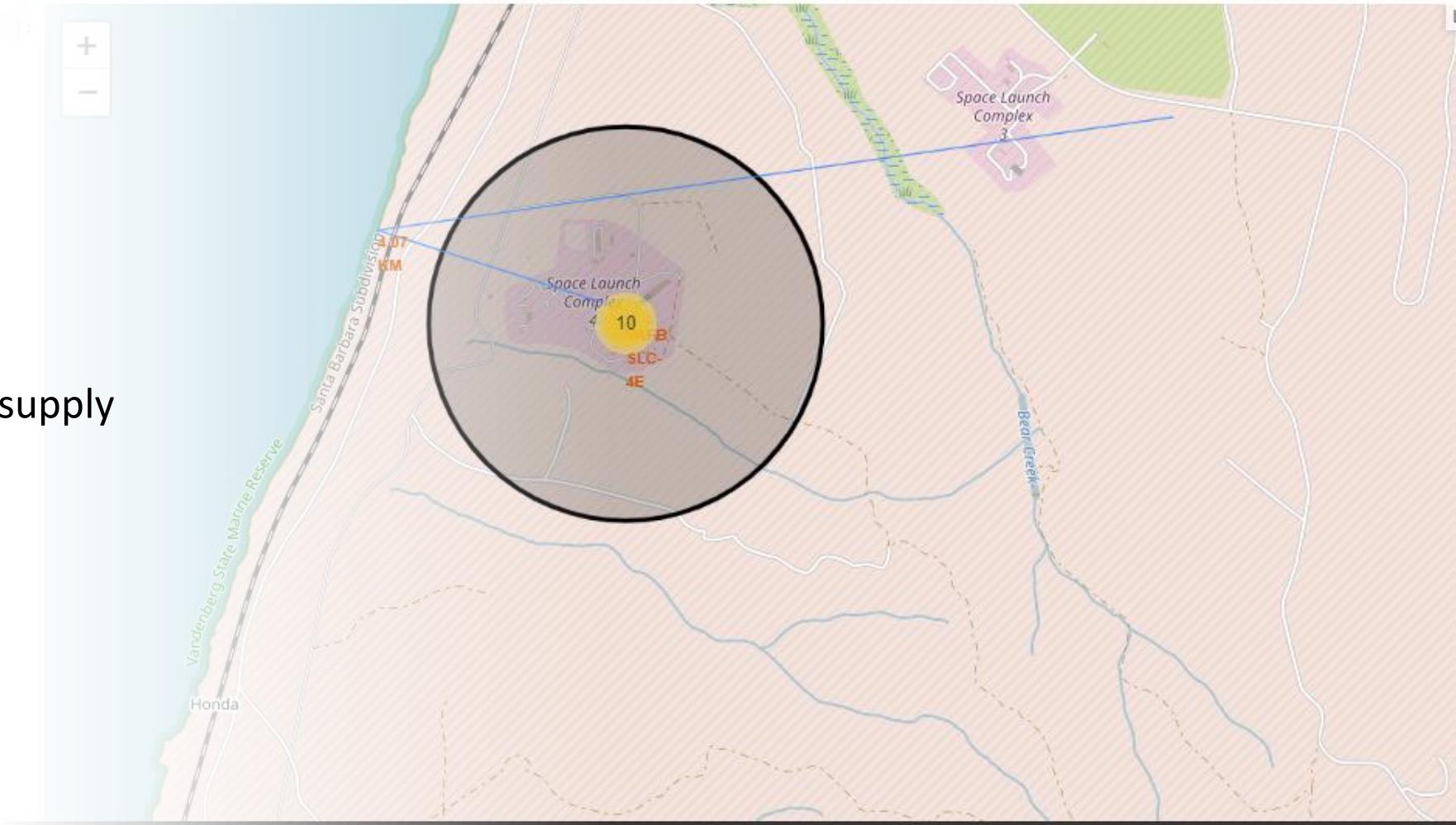
Launch Site Success/Failure

- Green = Launch success
- Red = Launch failure



Launch Site Area

- Launch sites are always in proximity of railways and highways presumably for supply and logistic reasons





Section 4

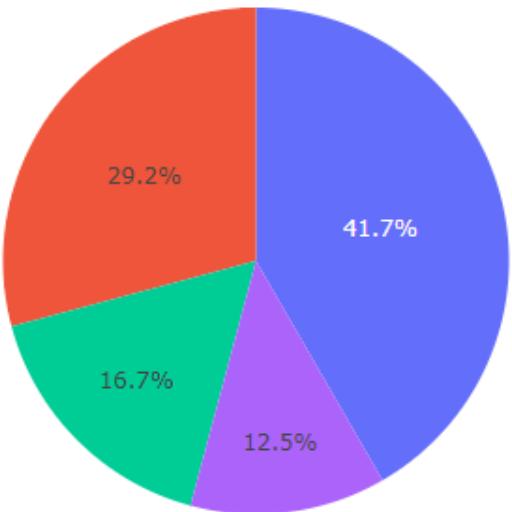
Build a Dashboard with Plotly Dash

Launch Success Dashboard

for all Sites

- Launch success for all sites

SpaceX Launch Records Dashboard

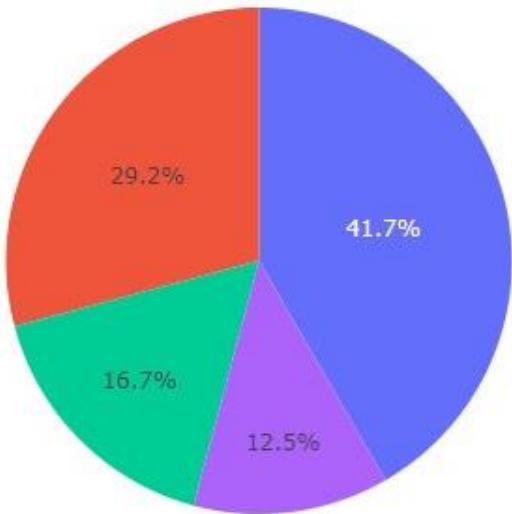


Launch Success Ratio

SpaceX Launch Records Dashboard

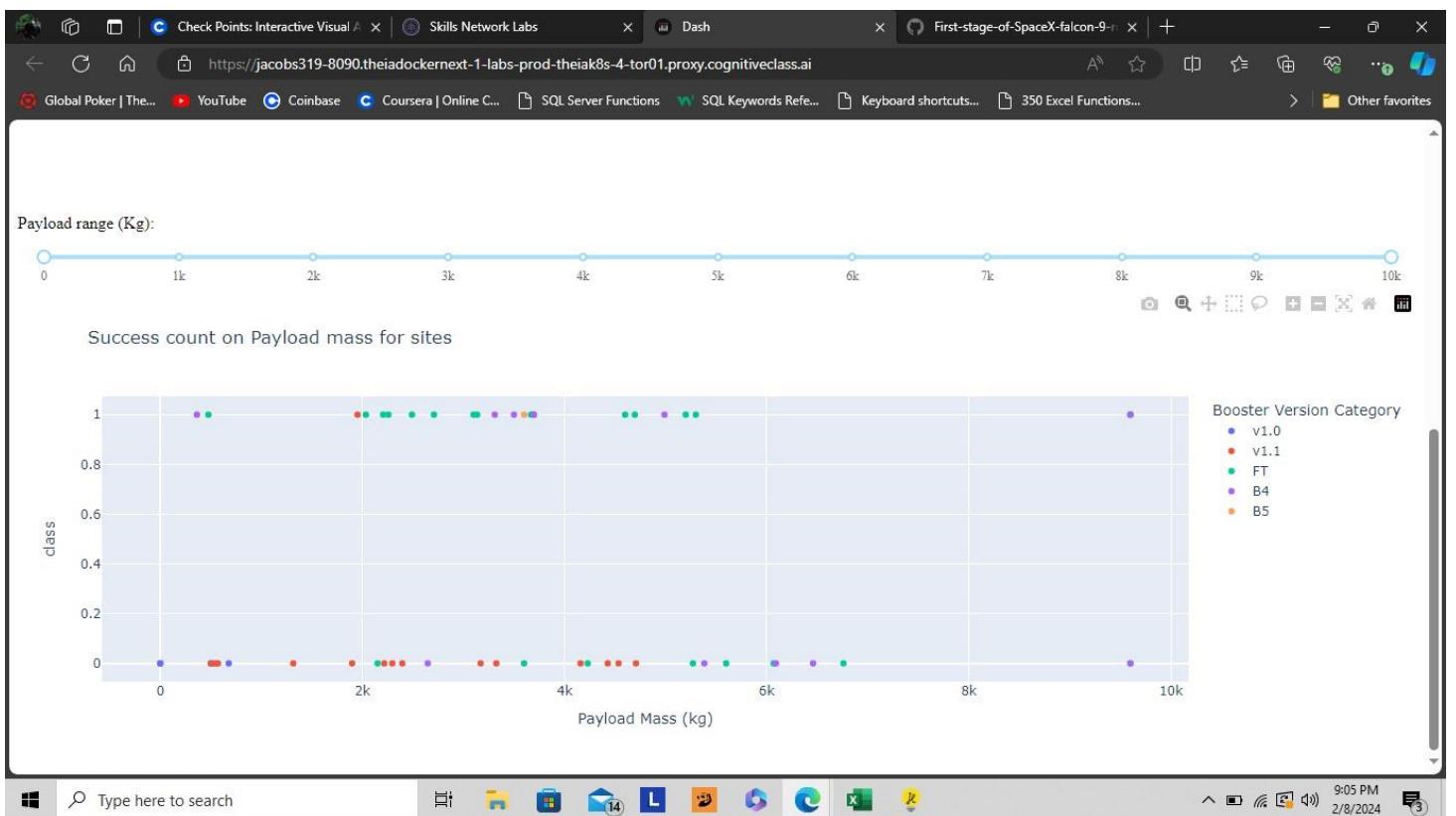
for all Sites

- Launch site with the highest ratio is KSC LC-40



Payload Launch vs Outcome

- Anything over 8000k booster version B4 is best suited
- For 8000k and under booster version FT is best suited



The background of the slide features a dynamic, abstract design. It consists of several thick, curved lines that transition from a bright yellow at the top right to a deep blue at the bottom left. These curves are set against a lighter blue background, creating a sense of motion and depth. In the lower right quadrant, there is a vertical column of white space where the text is placed.

Section 5

Predictive Analysis (Classification)

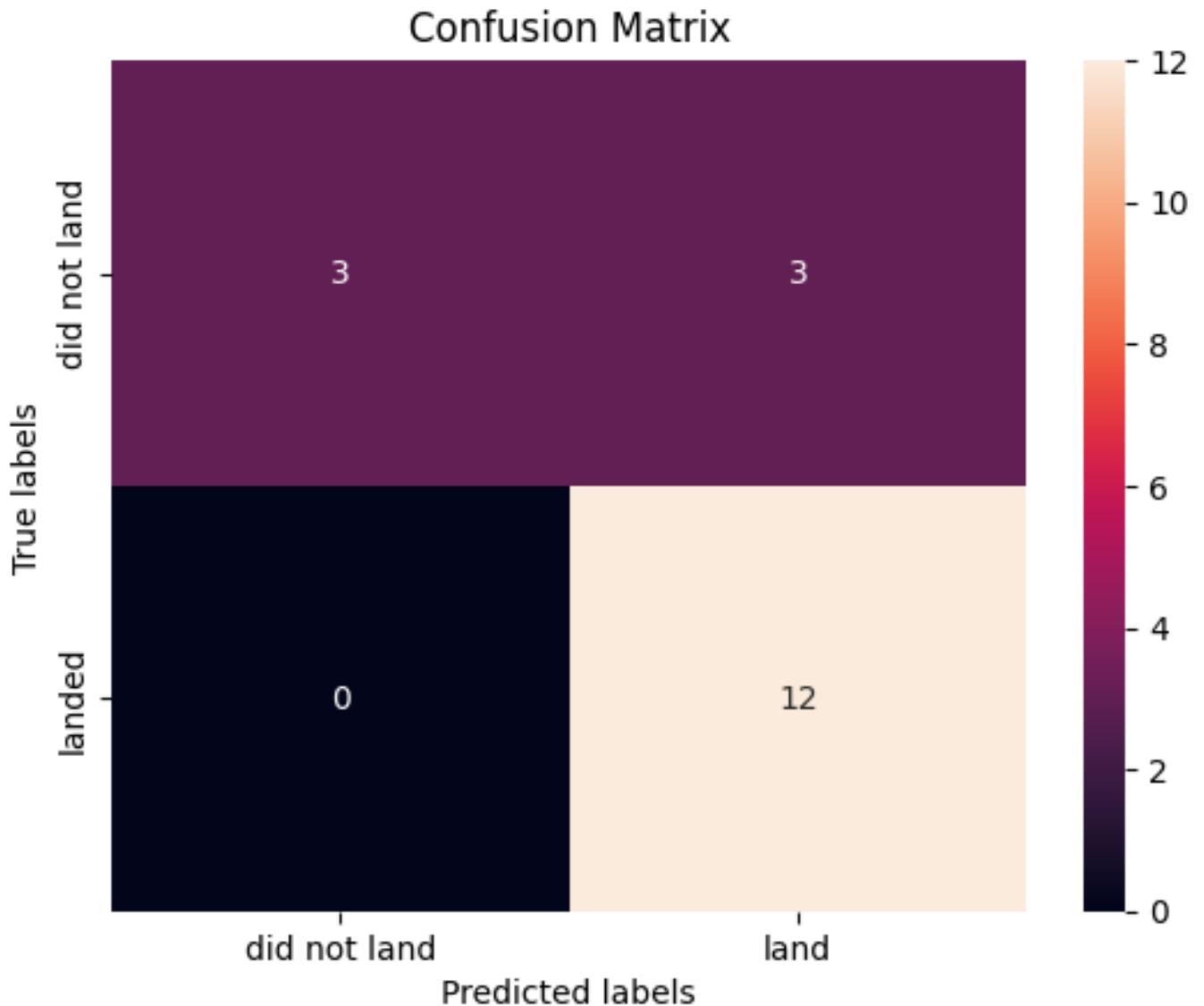


Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart
- Find which model has the highest classification accuracy

Confusion Matrix

- The Decision Tree had the best accuracy at 87%



Conclusions

SpaceX has a very high success rate

Depending on payload and orbit our rate goes up even higher

By duplicating the Data SpaceY should be able to successfully launch our vehicles

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!

