Final Capstone Presentation

Bohdan Ledwij

27 December 2021

OUTLINE

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

EXECUTIVE SUMMARY

Space X vs Space Y

- Data
  - Data for Space X was gathered via API
  - Data for Space Y was scraped from Wikipedia
- Analysis
  - Data was classified into successful and unsuccessful landings
  - Data was explored via Folium Maps, various visualizations and dashboards
  - Onehot encoding was leveraged to change variable data into binary

Machine Learning

- Leveraged Logistic Regression, Support Vector Machine, Decision Tree Classifier and K Nearest Neighbours.
- Achieved approx. 83% accuracy with these models
- All models predicted higher proportions of false positives

INTRODUCTION

- We will predict if the Falcon 9 first stage will land successfully!
- SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars
- other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.
- If we can determine if the first stage will land, we can determine the cost of a launch.
- This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

METHODOLOGY

Data Sources

- SpaceX Public API
- SpaceX Wikipedia

Data Wrangling

- Data classified into successful landings (true) and unsuccessful (false)

Exploratory Data Analysis (EDA)

- Leveraged SQL and Visualizations
- Leveraged Plotly Dash and Folium to implement interactive and dynamic geographical analytics
- Leveraged Classification models to undertake predictive analytics, leveraging GridSearch CV for fine tuning.

Methodology #2

Space X API

1. API Request
2. Json File & Data Lists
3. Json Normalize
4. Obtain relevant data &Cast Dictionary to DF
5. Filter Data (Falcon 9)
6. Substitute missing PayloadMass values with mean

Wikipedia Webscraping

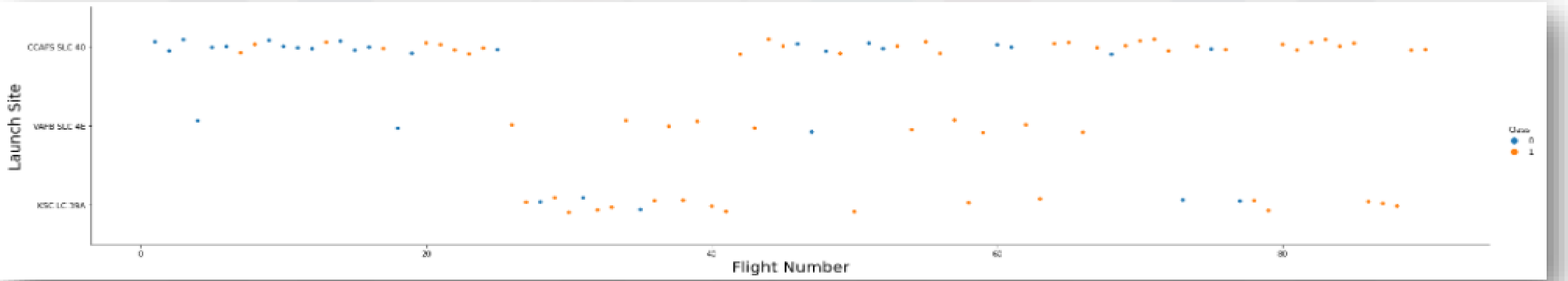
1. Request Wiki html
2. BeautifulSoup html5lib Parser
3. Find launch info html table
4. Create dictionary
5. Iterate to extract data to dictionary
6. Cast dictionary to DF

https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%201%20Notebook.ipynb

RESULTS

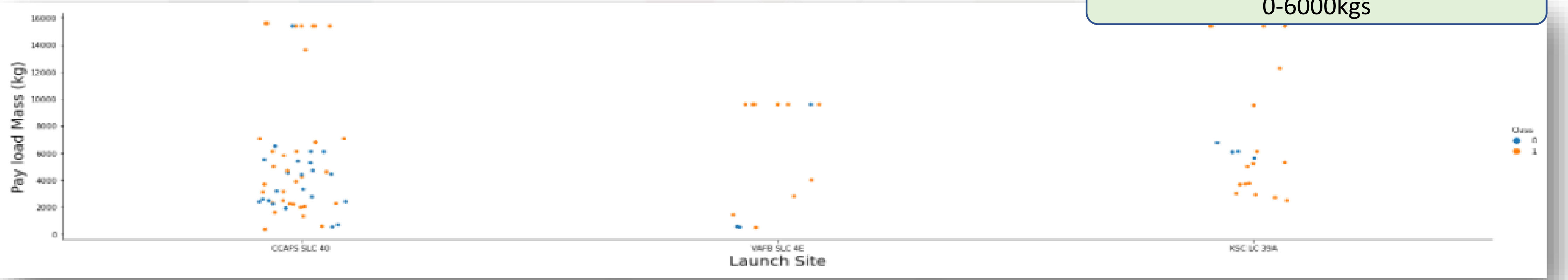
Flight number versus Launch Site

of successful attempts appears to improve as more launches are conducted



Pay Load versus Launch Site

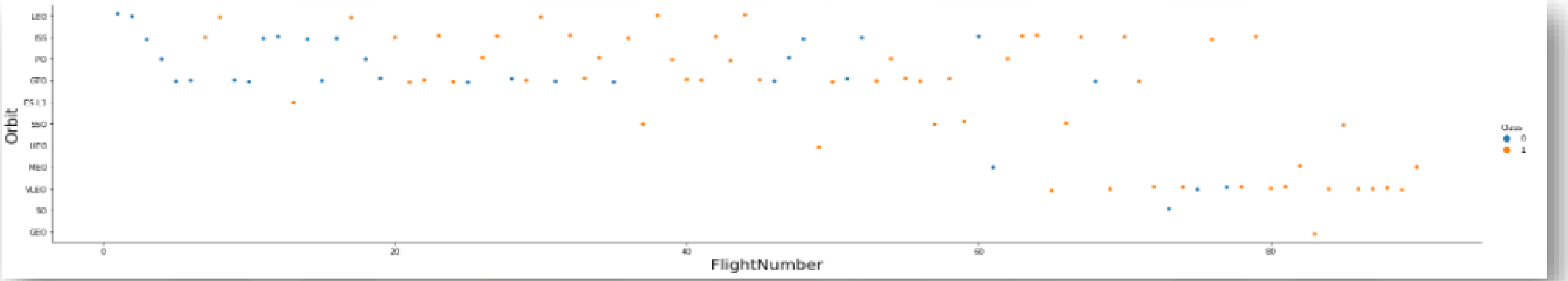
Most payloads are between 0-6000kgs



RESULTS

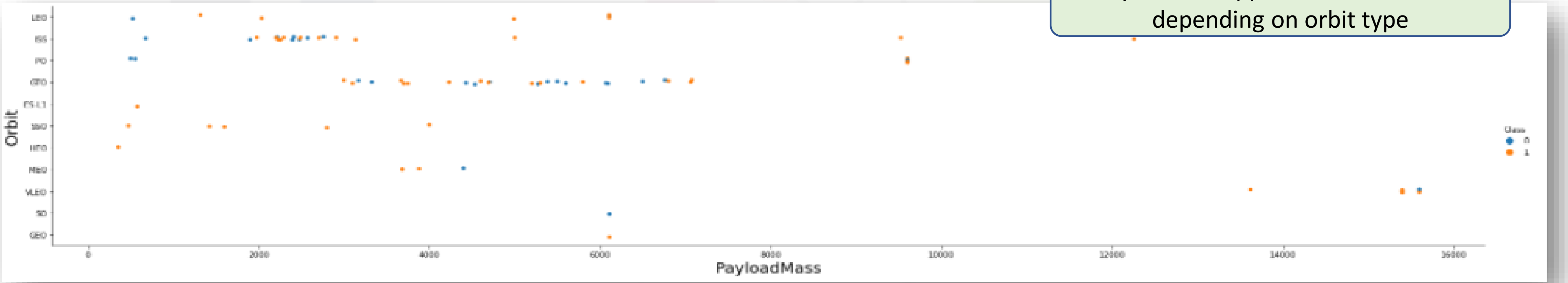
Flight number versus Orbit

Launch orbit changes over time which correlates with an increase in success



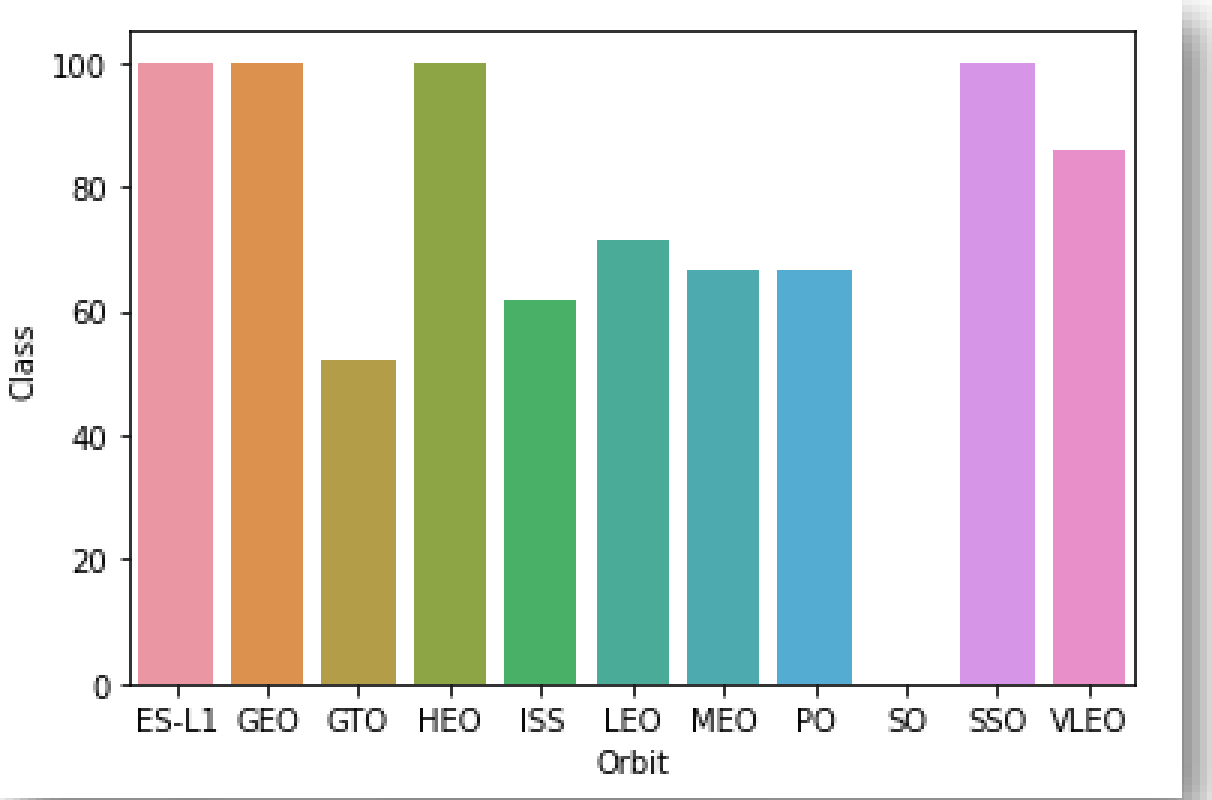
Pay Load versus Orbit

Payload ass appears to be different depending on orbit type



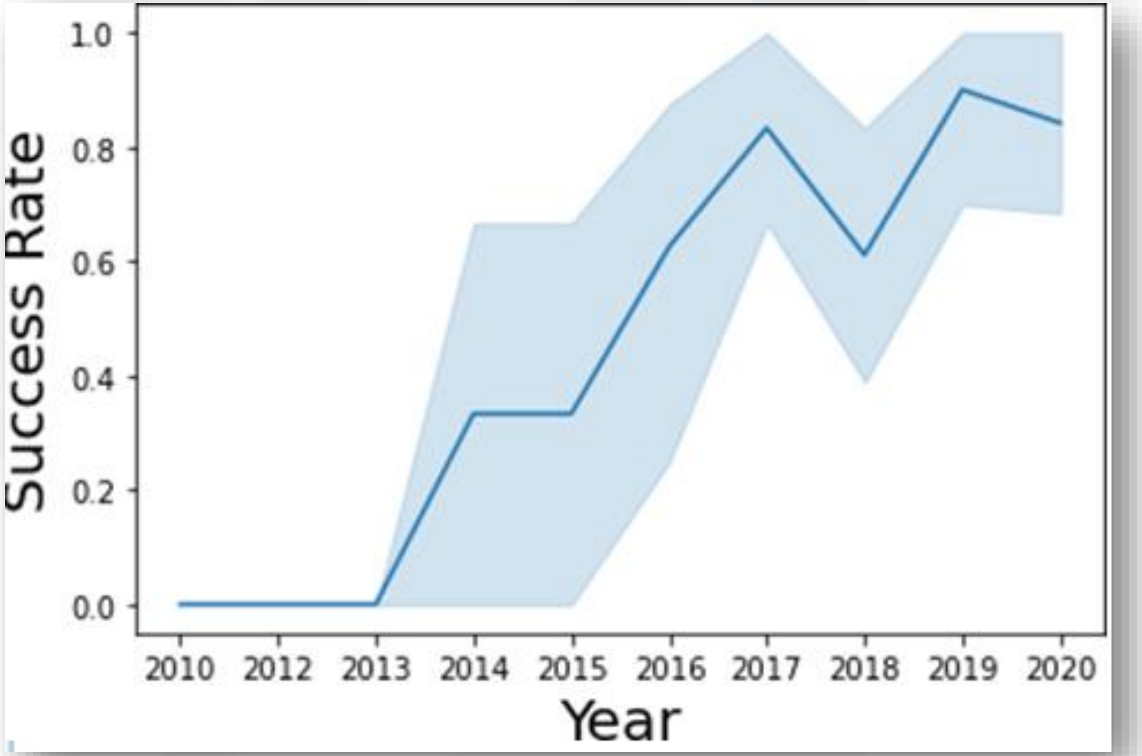
RESULTS

- 100% success rate
  - SSO (5)
  - ES-L1 (1)
  - GEO (1)
  - HEO (1)
- VLEO has mid 80's success rate off 14 attempts
- SO has 0% success rate off 1 attempt
- GTO has approximately 50% success rate off the largest sample of 27.



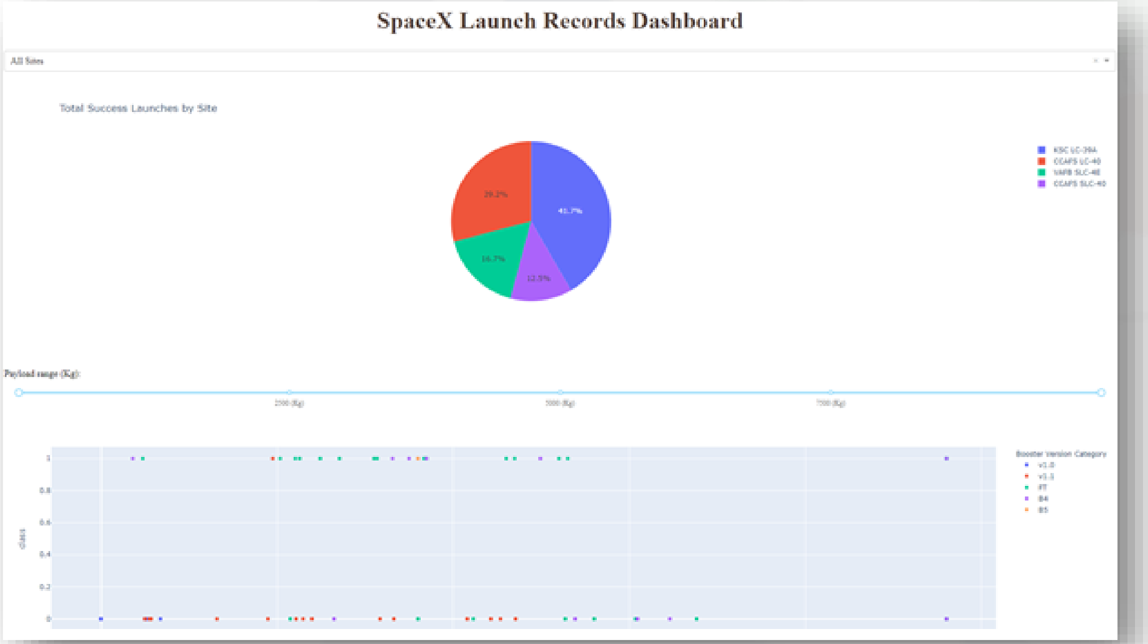
RESULTS

- There is a general increase in success rate
- Confidence interval of 95%
- Latest data shows success rate is approximately 80%



DASHBOARD

- Screen shot of SpaceX Launch Records Dashboards created leveraging Plotly Dash



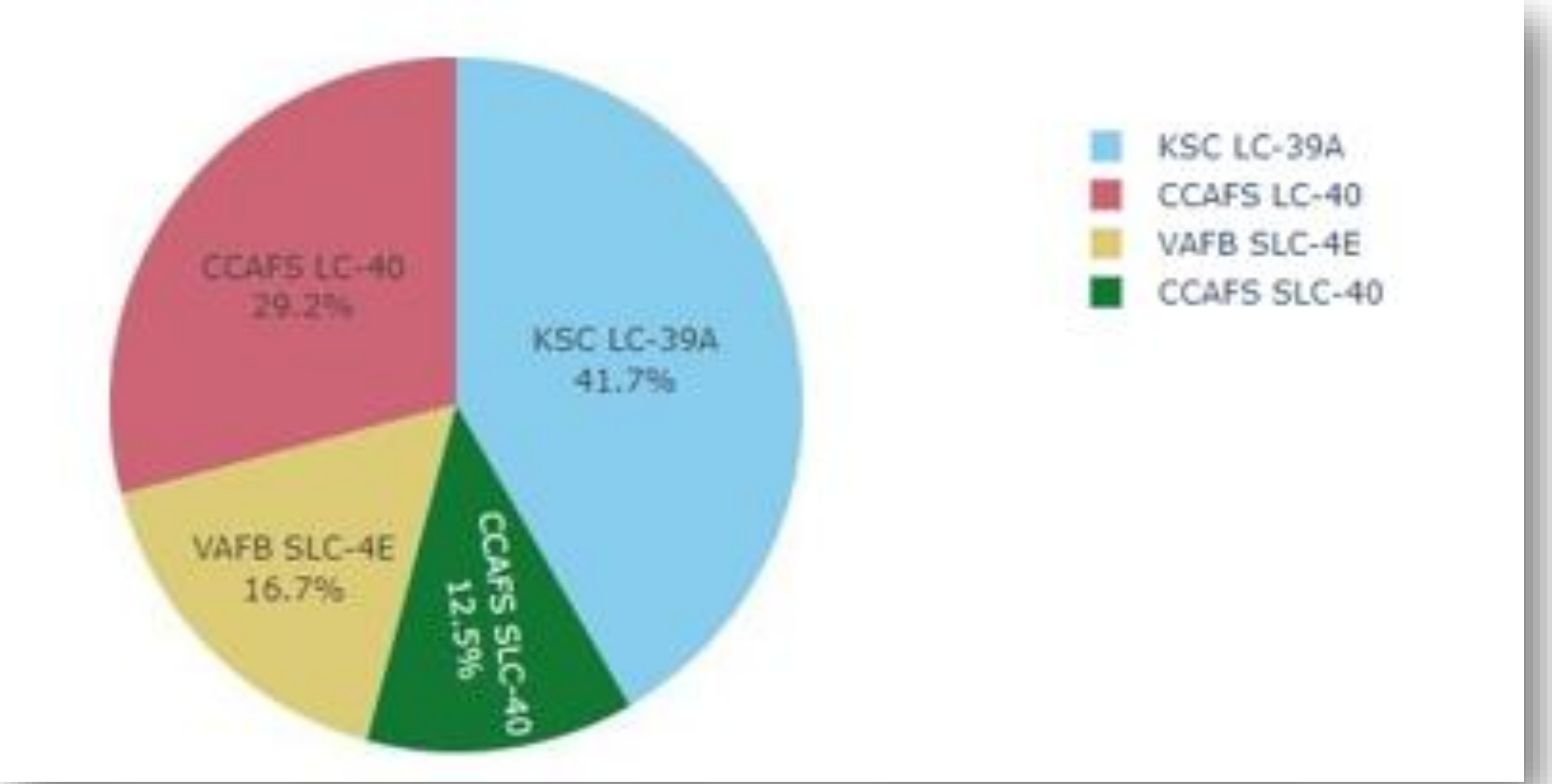
- Screen shot of code to achieve functional app

```
# Import required libraries
import pandas as pd
import plotly.graph_objects as go
import dash
import dash_html_components as html
import dash_core_components as dcc
from dash.dependencies import Input, Output

# Read the airline data into pandas dataframe
airline_data =  pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-DV0101EN-SkillsNetwork/Data%20Files/airline_data.csv',
                            encoding = "ISO-8859-1",
                            dtype={'Div1Airport': str, 'Div1TailNum': str,
                                   'Div2Airport': str, 'Div2TailNum': str})
# Create a dash application
app = dash.Dash(__name__)

app.layout = html.Div(children=[ html.H1('Airline Performance Dashboard',
                                style={'textAlign': 'center', 'color': '#503D36',
                                'font-size': 40}),
                                html.Div(["Input Year: ", dcc.Input(id='input-year', value='2010',
                                type='number', style={'height':'50px', 'font-size': 35}),],
                                style={'font-size': 40}),
                                html.Br(),
                                html.Br(),
                                html.Div(dcc.Graph(id='line-plot')),
                                ])

# add callback decorator
@app.callback( Output(component_id='line-plot', component_property='figure'),
               Input(component_id='input-year', component_property='value'))

# Add computation to callback function and return graph
def get_graph(entered_year):
    # Select 2019 data
    df =  airline_data[airline_data['Year']==int(entered_year)]

    # Group the data by Month and compute average over arrival delay time.
    line_data = df.groupby('Month')['ArrDelay'].mean().reset_index()

    fig = go.Figure(data=go.Scatter(x=line_data['Month'], y=line_data['ArrDelay'], mode='lines', marker=dict(color='green')))
    fig.update_layout(title='Month vs Average Flight Delay Time', xaxis_title='Month', yaxis_title='ArrDelay')
    return fig

# Run the app
if __name__ == '__main__':
    app.run_server()
```

https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%203%20Dashboard%20App.ipynb

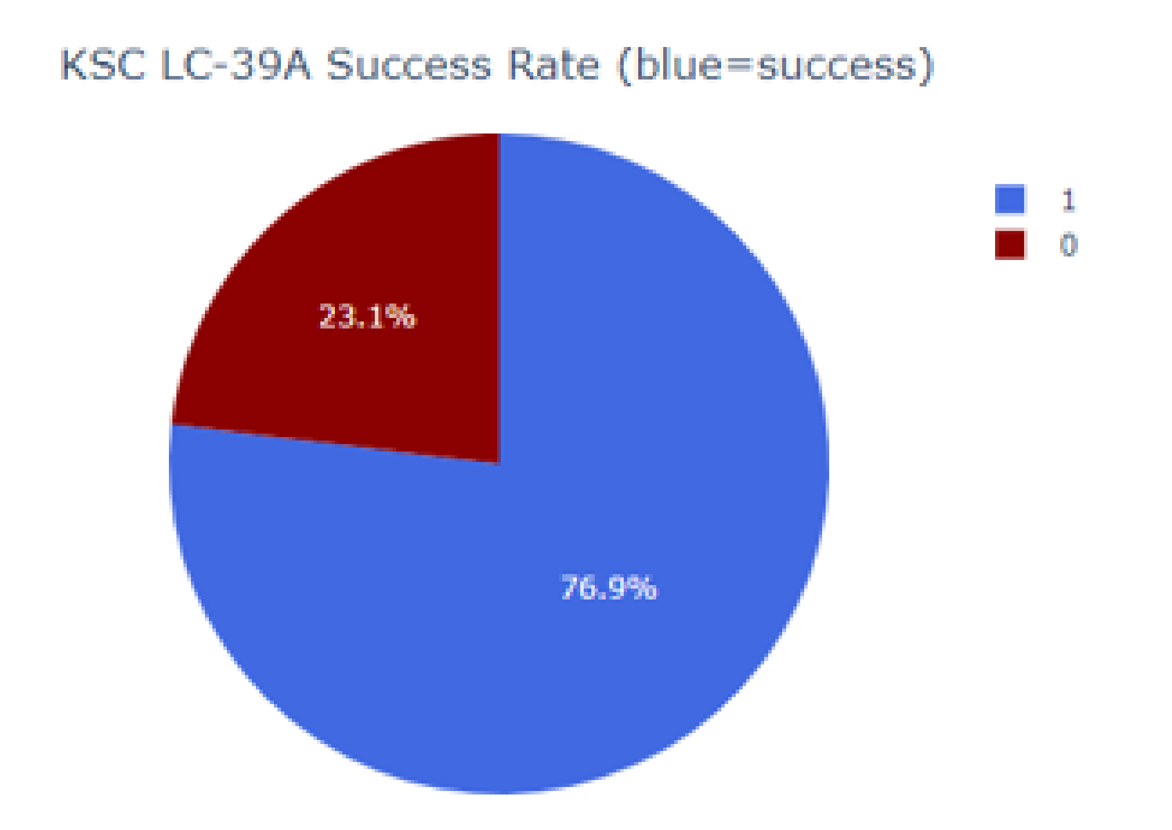
DASHBOARD TAB 1

- Distribution of successful landings across launch sites.
- CCAFS and KSC have the same amount of successful landings
- VAFB has the smallest number of successful landings.

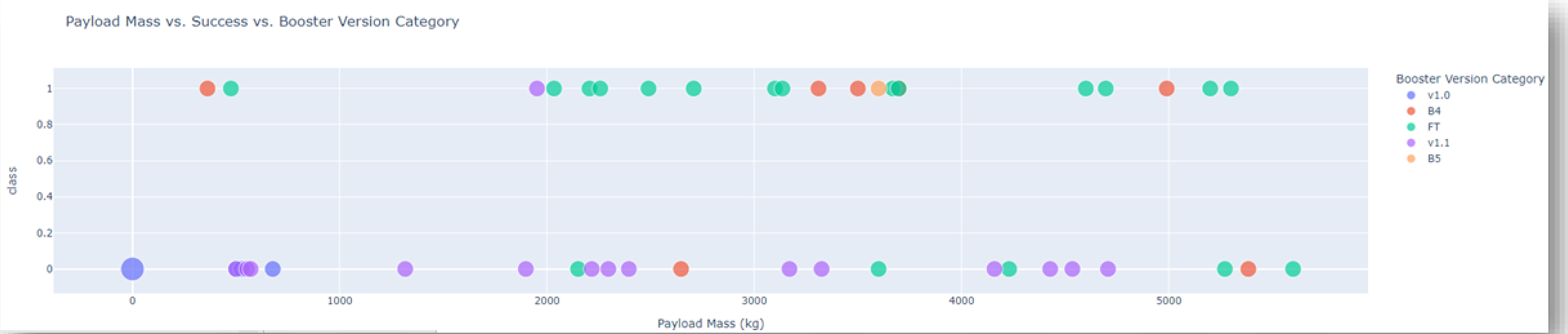


DASHBOARD TAB 2

- KSC LC-39A has greatest success rate
    - 10 successful landings
    - 3 failed landings.



DASHBOARD TAB 3

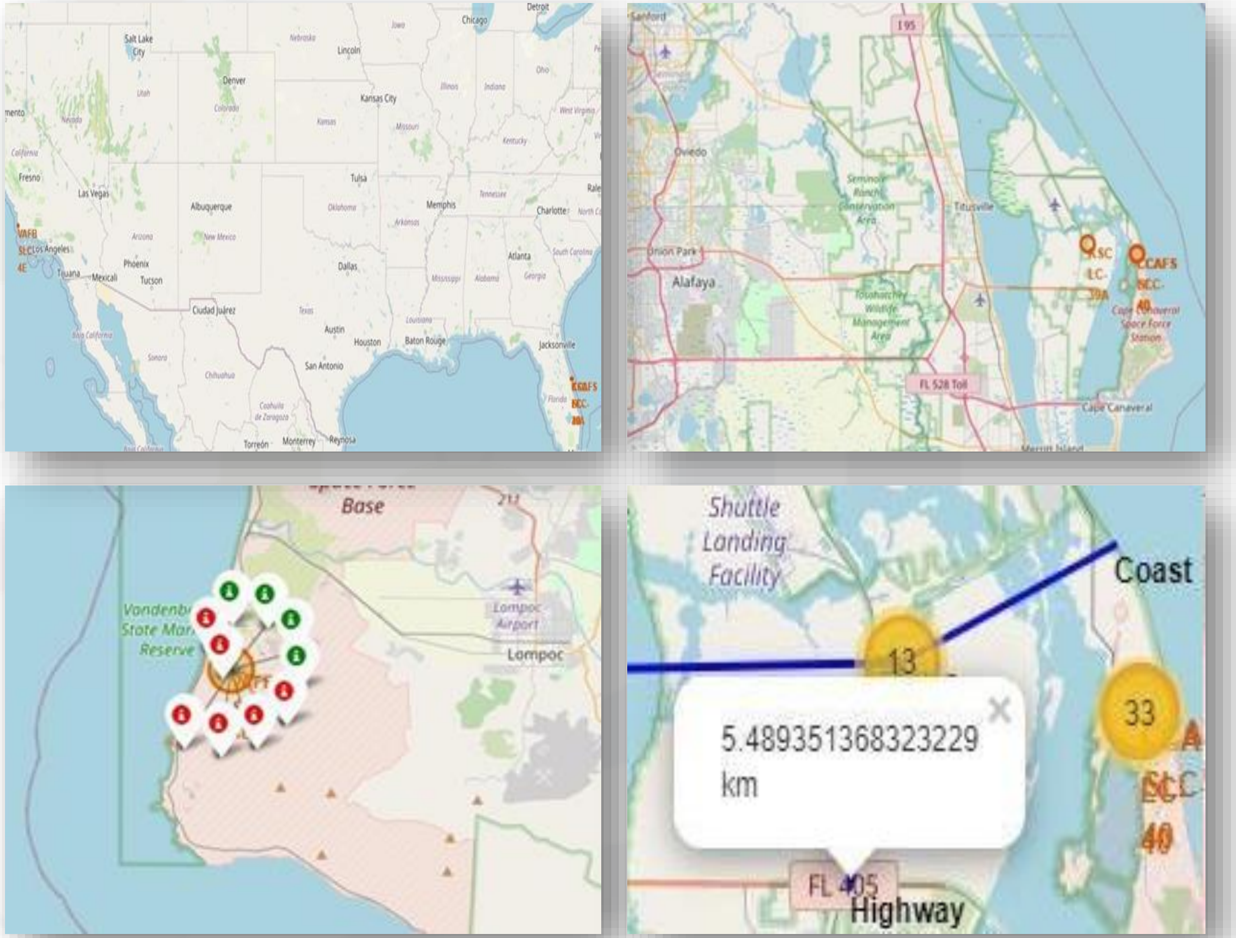


- Class indicates 1 for successful landing, 0 for failure.
- Scatter plot is colored by booster version category
- Size of dot represents the number of launches number of launches

DISCUSSION

- Top images display launch sites
- Bottom left image displays clusters to display successful landings in green and unsuccessful in red.
- Observing proximity of surrounding features, launch sites are close to railways and highways for supply chain purposes and close to oceans in the event of failure.

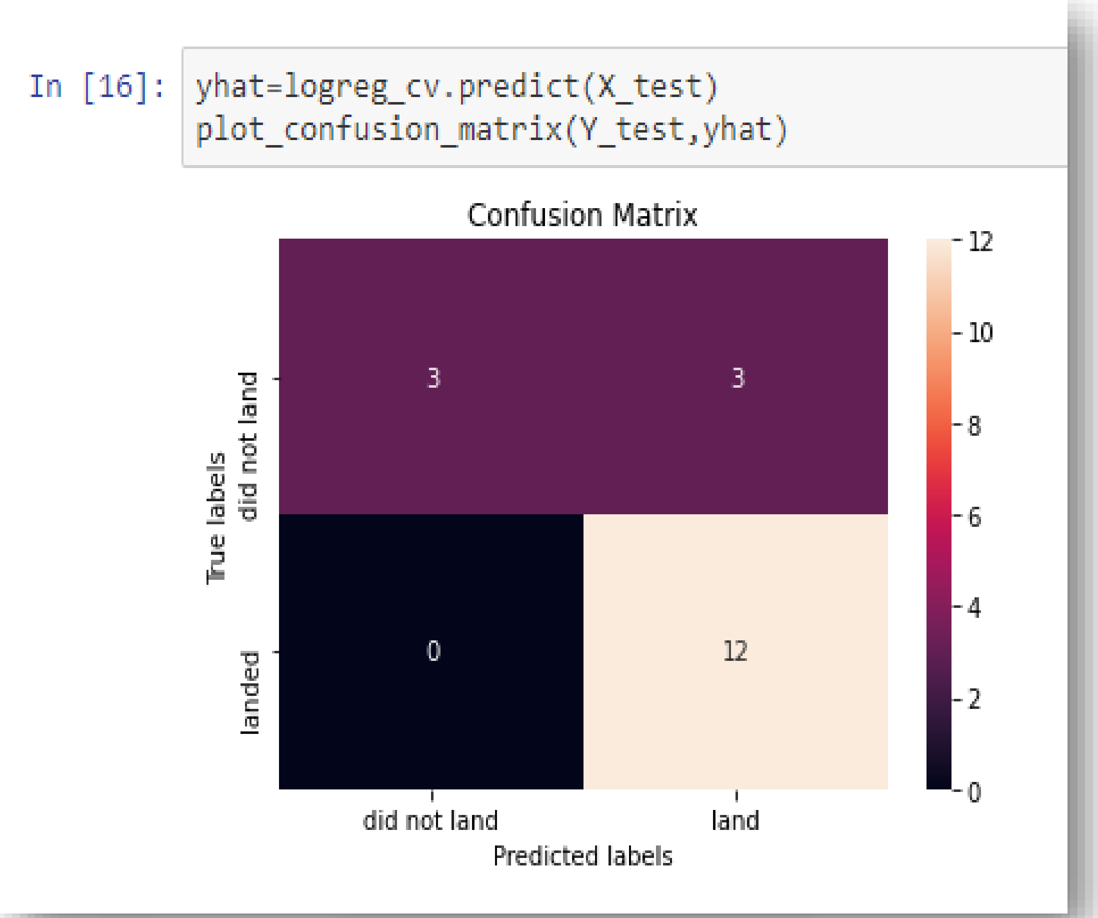
https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%203%20Dashboard%20App.ipynb



DISCUSSION

- All models performed similarly
- The models over predict successful landings.
- correctly predicted 12 successful landings
- Correctly predicted 3 unsuccessful landings
- predicted 3 false positive successful landings

https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%204%20Notebook.ipynb



OVERALL FINDINGS & IMPLICATIONS

Findings

- Overtime, the success rate of launches increased
- Differing orbit types yielded differing success rates
- Launch locations factored the supply chain and safety
- Low number of launch attempts for different orbit types

Implications

- Without greater data available, the ML algorithm will continue to present have false positives
- These false positives could lead to disaster costing lives and dollars
- Low number of launch attempts for different orbit types means that it is difficult to know true success rate as this is a significant variable

CONCLUSION

- The objective of this data science project was to create an ML model with the goal of predicting successful stage 1 landings, saving approx. US$100MM on failed landings.
- Data was collected via API and Webscraping
- Data was stored in a DB2 SQL DB
- A dynamic dashboard was created leveraging Plotly Dash
- An ML model with 83% accuracy was created leveraging the available data
- For space exploration, this % is low and thus more data should be acquired so that the ML model operates with greater accuracy

APPENDIX

All relevant github:

Notebooks

- https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%202%20Notebook%20b.ipynb
- https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%203%20Dashboard%20App.ipynb
- https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%203%20Dashboard%20App.ipynb
- https://github.com/Bodi23/Coursera_Capstone/blob/main/Capstone%20SpaceX%20-%20Week%204%20Notebook.ipynb

PDF of Presentation

- https://github.com/Bodi23/Coursera_Capstone/blob/main/capstone-story-template-Bohdan_Ledwij.pdf