APPLIED DATA SCIENCE CAPSTONE

IBM

OUTLINE

- Executive Sammary
- Introduction
- Methodology
- Results
- Conclusion

EXECUTIVE SUMMARY

- Our strategy involved utilizing both API integration and web scraping methods to gather data. After obtaining the data, we utilized various Python techniques to thoroughly process and clean it. We then used SQL queries to extract relevant information from the refined dataset. Early insights were obtained through systematic data visualization and trend analysis. To complete our analytical framework, we utilized supervised machine learning models to predict the success of landing events.
- By conducting thorough data analysis, we uncovered clear patterns and correlations among variables that directly impact the success of landing events. Using these insights, we built and trained a predictive model that showed significant ability to accurately predict the likelihood of a successful landing event. Importantly, the model achieved an impressive accuracy rate of 83%, highlighting its effectiveness in delivering dependable predictions within this field.

INTRODUCTION

- SpaceX's dedication to reusable rockets has notably decreased the expenses associated with space travel by concentrating on retrieving the initial phase of the rocket. Preserving and reusing costly components from this phase is crucial for reducing costs directly. Evaluating the success rate of these retrieval events provides a significant metric for assessing efficiency and cost-effectiveness in SpaceX's innovative approach. This project focuses on predicting the success of the first phase retrieval event, providing predictive insights to improve decision-making within the space industry.

METHODOLOGY

- How data was collected
- How data was processed
- Exploratory data analysis using visualization and SQL
- Interactive visualizing with Folium and Plotly Dash
- Predictive analysis using classification models

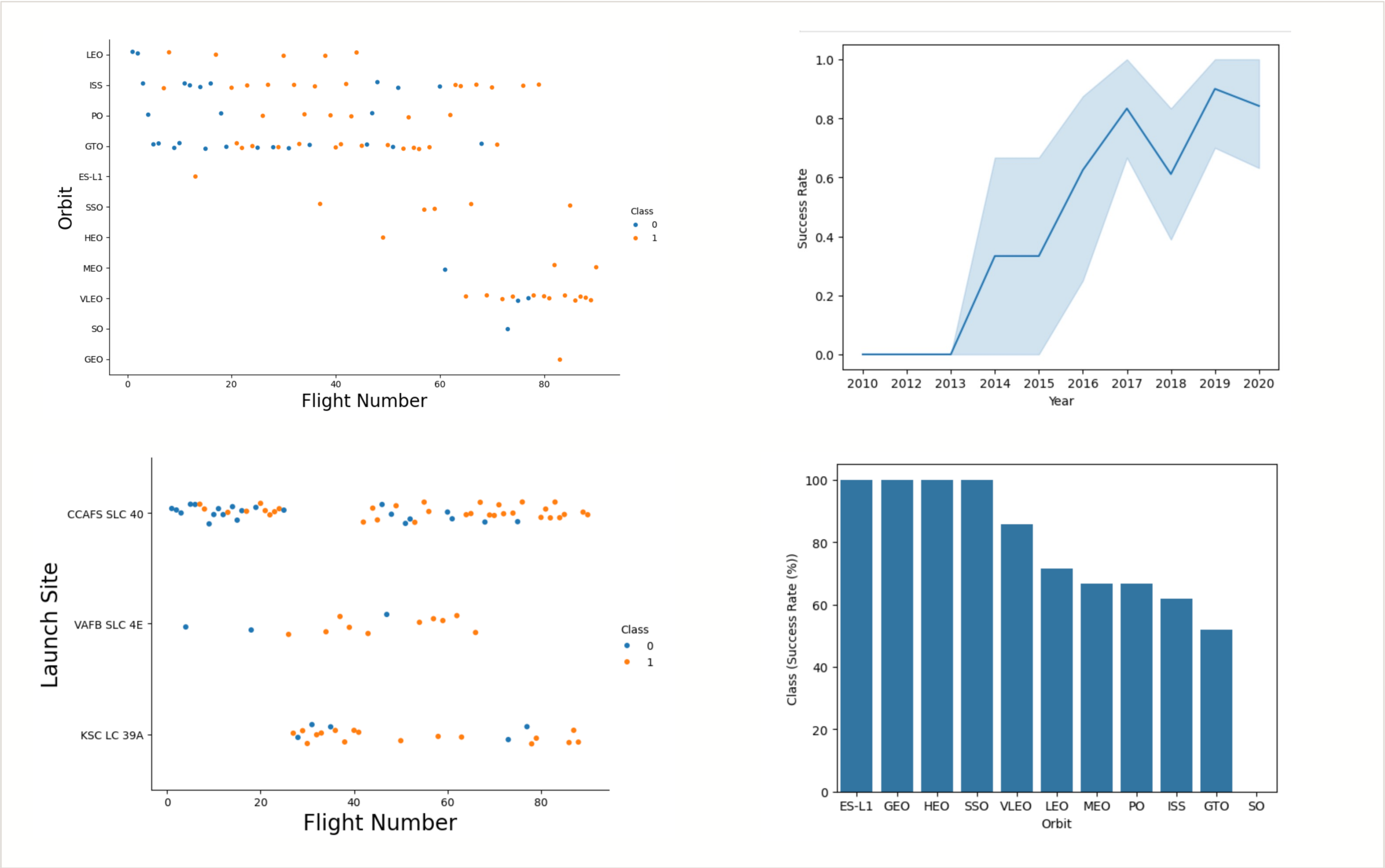
DATA COLLECTION

- Data was first collected using SpaceX API (a RESTful API) by making a get request to theSpaceX API. This was done by first defining a series helper functions that would help in the use of the API to extract information using identification numbers in the launch data and then requesting rocket launch data from the SpaceX API url.
- SpaceX launch data was requested and parsed using the GET request and then decoded the response content as aJson result which was then converted into a Pandas data frame.
- performed web scraping to collect Falcon 9 historical launch records from Wikipedia page

SPACEX API

- Define auxiliary function to parse the data
- Retrieve data from the REST API using the method GET
- Parse the data with the previously built auxiliary functions
- Store the data in PANDAS DataFrame

DATA VISUALIZATION



SQL

- Displaying the names of the launch sites.
- Displaying 5 records where launch sites begin with the string 'CCA'.
- Displaying the total payload mass carried by booster launched by NASA (CRS).
- Displaying the average payload mass carried by booster version F9 v1.1.
- Listing the date when the first successful landing outcome in ground pad was achieved.
- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
- Listing the total number of successful and failure mission outcomes.
- Listing the names of the booster_versions which have carried the maximum payload mass.

INTERACTIVE MAP WITH FOLIUM

- We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- We assigned the feature launch outcomes (failure or success) to class 0 and 1.i.e., 0for failure, and 1 for success.
- Using the color-labeled marker clusters, we identified which launch sites have relatively high success rate.

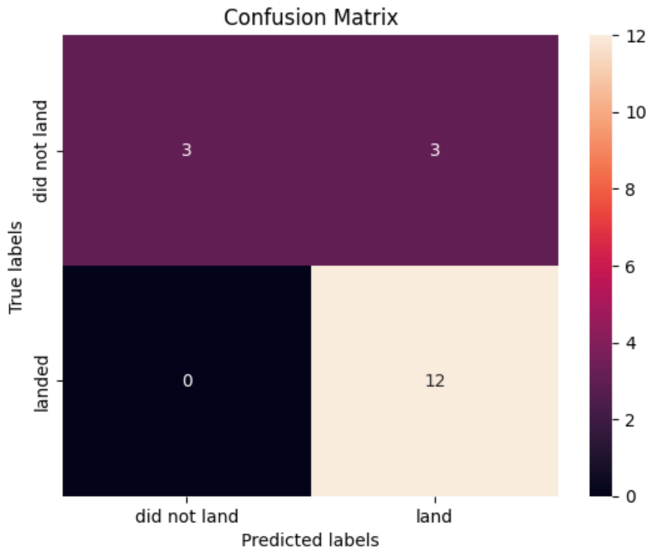
DASHBOARD WITH PLOTLY DASH

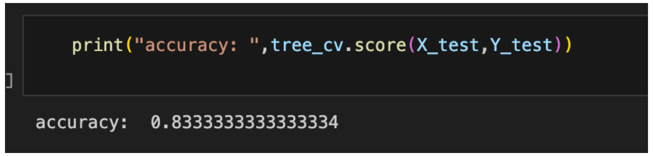
- We built an interactive dashboard with Plotly dash.
  We plotted pie charts showing the total launches by a certain sites.
- We plotted scatter graph showing the relationship with Outcome and PayloadMass (Kg) for the different booster version.

CLASSIFICATION

- We loaded the data using numpy and pandas, transformed the data, split our data into training and testing sets.
- Tune different hyperparameters using GridSearchCV.
- We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.

RESULTS





```
print("accuracy: ",tree_cv.score(X_test,Y_test))

accuracy:  0.8333333333333334
```

CONCLUSION

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate started to increase in 2013 till 2020.