Winning Space Race with Data Science

Pranav Anbarasu
July 30, 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection via API and web scraping
  - Data wrangling
  - EDA using SQL
  - EDA for data visualization
  - Interactive visual analytics and dashboard
  - Predictive analysis
- Summary of all results
  - Successfully collected valuable data from public data sources
  - EDA revealed which features are more useful for predicting launch outcomes
  - Predictive analytics showed which model was most accurate

Introduction

- Project background and context
  - Space X 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 Space X can reuse the first stage. Therefore, 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 space X for a rocket launch.
  - The goal is to predict if the Falcon 9 first stage will land successfully by training a machine learning model and using publicly available information
- Problems you want to find answers
  - What are the best features to use for predicting launch outcomes?
  - Where is the best location to launch from?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data obtained from SpaceX API and web scraping Wikipedia
- Perform data wrangling
  - Find patterns in the data and determine what would be ideal labels for training supervised models
  - Create a landing outcome label
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash

Methodology

Executive Summary

- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models
  - Normalize the data, split into train and test sets, fit and score using various classification models, then evaluate each model's accuracy and different combinations of hyperparameters

Data Collection

- Data were collected via the official SpaceX API
  - get request to the API url
- Decode response content to json using .json() and normalize to text in a pandas data frame using .json_normalize()
- Clean the data, check for missing values, replace missing values where necessary
- Web scrape Wikipedia for Falcon 9 launch data using BeautifulSoup package
- Extract launch data HTML table and parse to data frame

Data Collection – SpaceX API

- Use get request to SpaceX API url to get the data, clean the response data, then format and store in data frame
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20%20Applied%20Data%20Science%20Capstone/jupyter-labs-spacex-data-collection-api.ipynb

1. Get request for rocket launch data using API

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

2. Use json_normalize method to convert json result to dataframe

In [12]:
```
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

In [30]:
```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```

Data Collection - Scraping

- Use BeautifulSoup package to web scrape Falcon 9 launch data from Wikipedia
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/jupyter-labs-webscraping.ipynb

1. Apply HTTP Get method to request the Falcon 9 rocket launch page

In [4]:
```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

In [5]:
```
# use requests.get() method with the provided static_url
# assign the response to a object
html_data = requests.get(static_url)
html_data.status_code
```

Out[5]:
```
200
```

2. Create a BeautifulSoup object from the HTML response

In [6]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(html_data.text, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

In [7]:
```
# Use soup.title attribute
soup.title
```

Out[7]:
```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

3. Extract all column names from the HTML table header

In [10]:
```
column_names = []

# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided extract_column_from_header() to get a column name
# Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names

element = soup.find_all('th')
for row in range(len(element)):
    try:
        name = extract_column_from_header(element[row])
        if (name is not None and len(name) > 0):
            column_names.append(name)
    except:
        pass
```

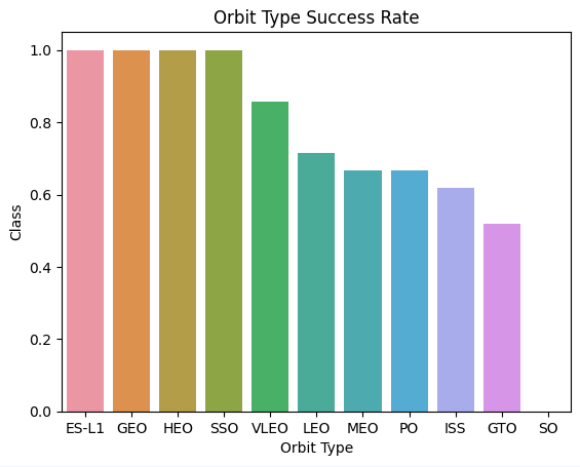
4. Create a dataframe by parsing the launch HTML tables
5. Export data to csv

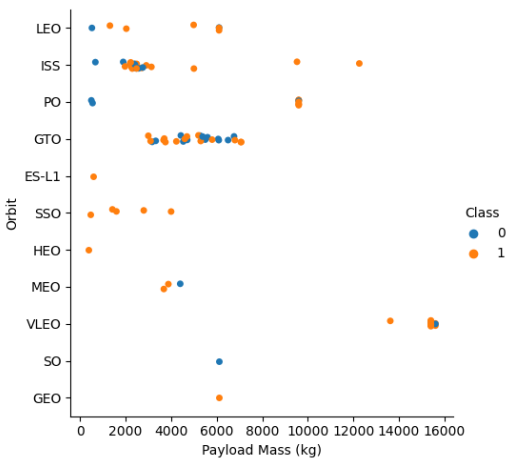
Data Wrangling

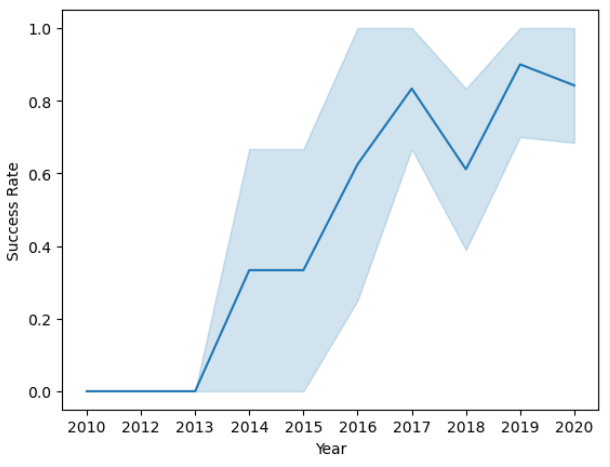
- Perform some initial EDA
- Summarize launches per site, frequency of launch to each orbit type, and frequency of launch outcome per orbit type
- Create landing outcome label using ‘Outcome’ column of data
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

- Plotted launch site vs flight number, launch site vs payload mass, orbit type success rate, orbit type vs flight number, orbit type vs payload mass, and launch success yearly trend
- To show relationships between numerical and categorical variables to determing which variables to use for future prediction
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb







EDA with SQL

- SQL queries performed
  - Names of the unique launch sites in the space mission;
  - Top 5 launch sites whose name begin with the string 'CCA';
  - Total payload mass carried by boosters launched by NASA(CRS);
  - Average payload mass carried by booster version F9 v1.1;
  - Date when the first successful landing outcome in ground pad was achieved;
  - Names of the boosters which have success in drone ship and have payload mass between 4000 and 6000 kg;
  - Total number of successful and failure mission outcomes;
  - Names of the booster versions which have carried the maximum payload mass
  - Failed landing outcomes in drone ship, their booster versions, and launch site names in year 2015
  - Rank of the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Create markers, circles, lines, and marker clusters and added to a folium map
  - Indicate launch sites using points, highlight areas of interest using circles, related events in nearby locations grouped together using marker clusters (launches at a launch site), show distance between coordinates using lines
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

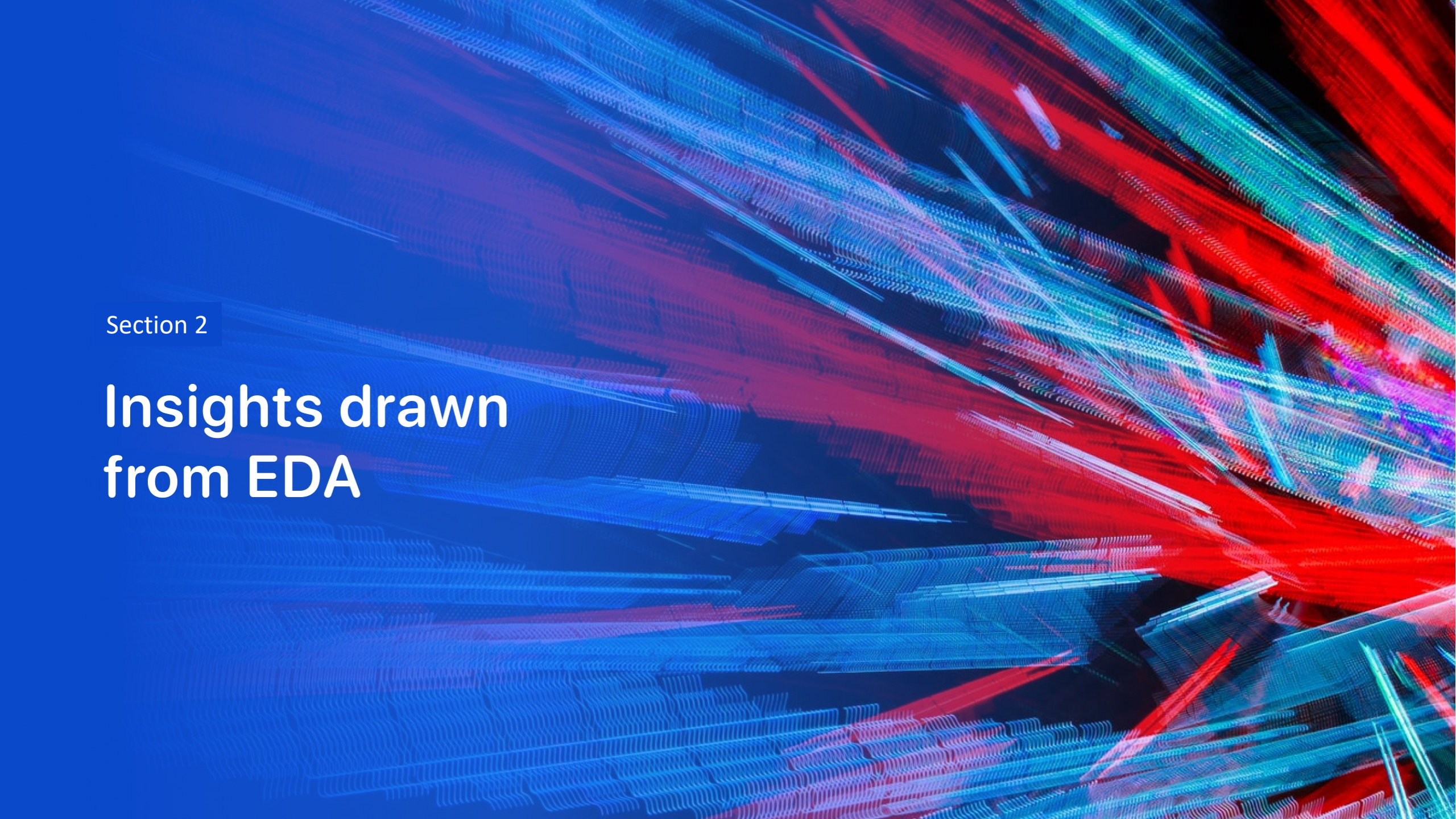
- Interactive dashboard shows data about launches by site and booster
- Pie charts show total launches across all sites, proportion of successful and failed launches per site
- Scatter plot shows launch outcome vs payload mass for different boosters
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/spacex_dash_app.py

Predictive Analysis (Classification)

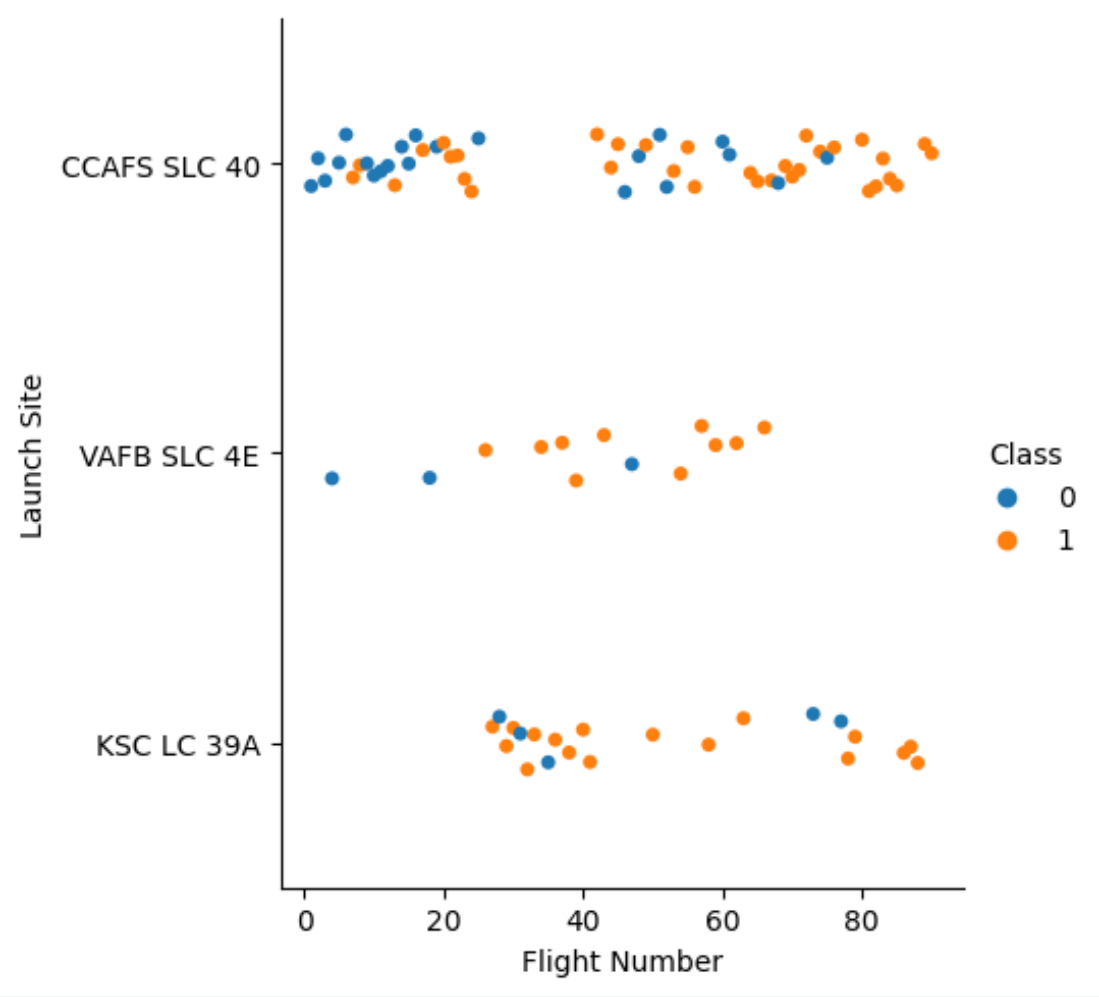
- Load the data to arary and split into train and test sets (80/20)
- Logistic regression, support vector machine, decision tree, and k-nearest neighbors classification models were built
- Use GridSearchCV to find best parameters for each model
- Fit and score each model
- Compare accuracy and best score, best parameter set for each model
- https://github.com/pranavanba/ibm-data-science/blob/main/Course%2010%20-%20Applied%20Data%20Science%20Capstone/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

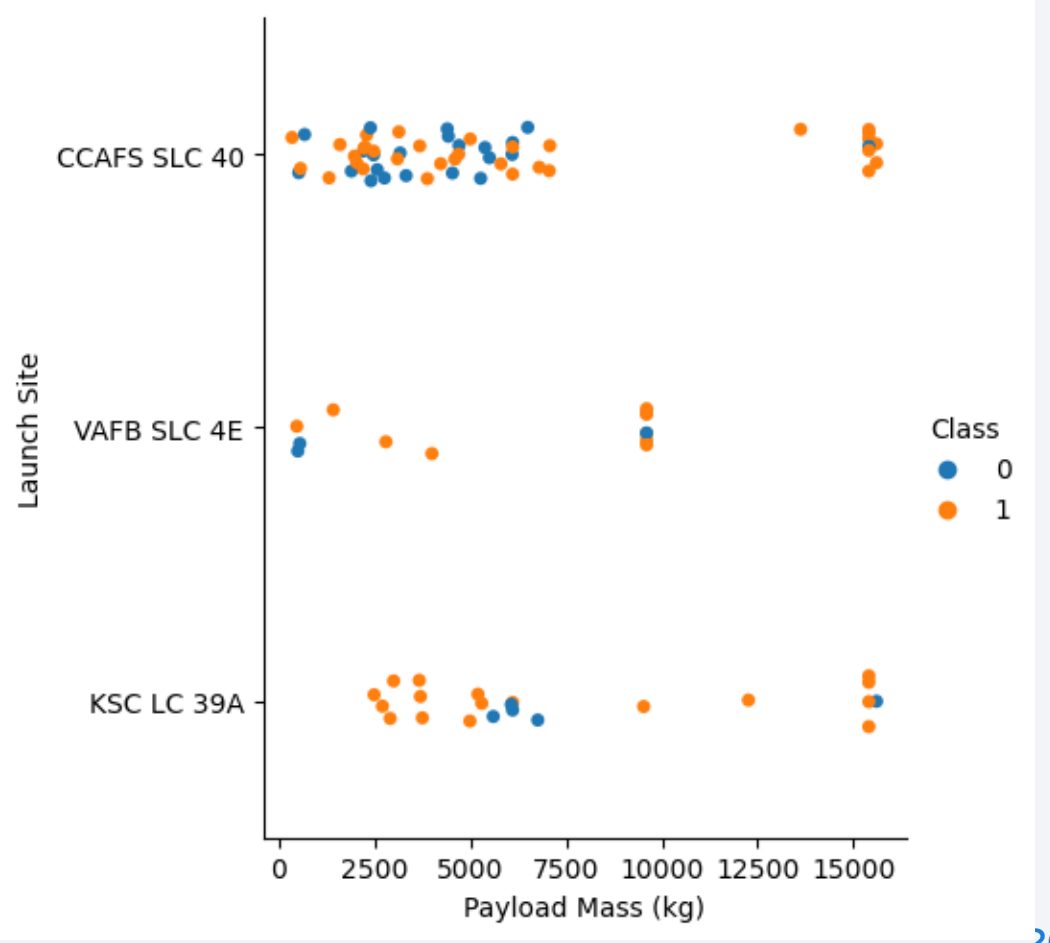
Flight Number vs. Launch Site



- The success rate appears to be greater as the flight number increases
- All launch sites tends to have more successful launches at higher flight numbers

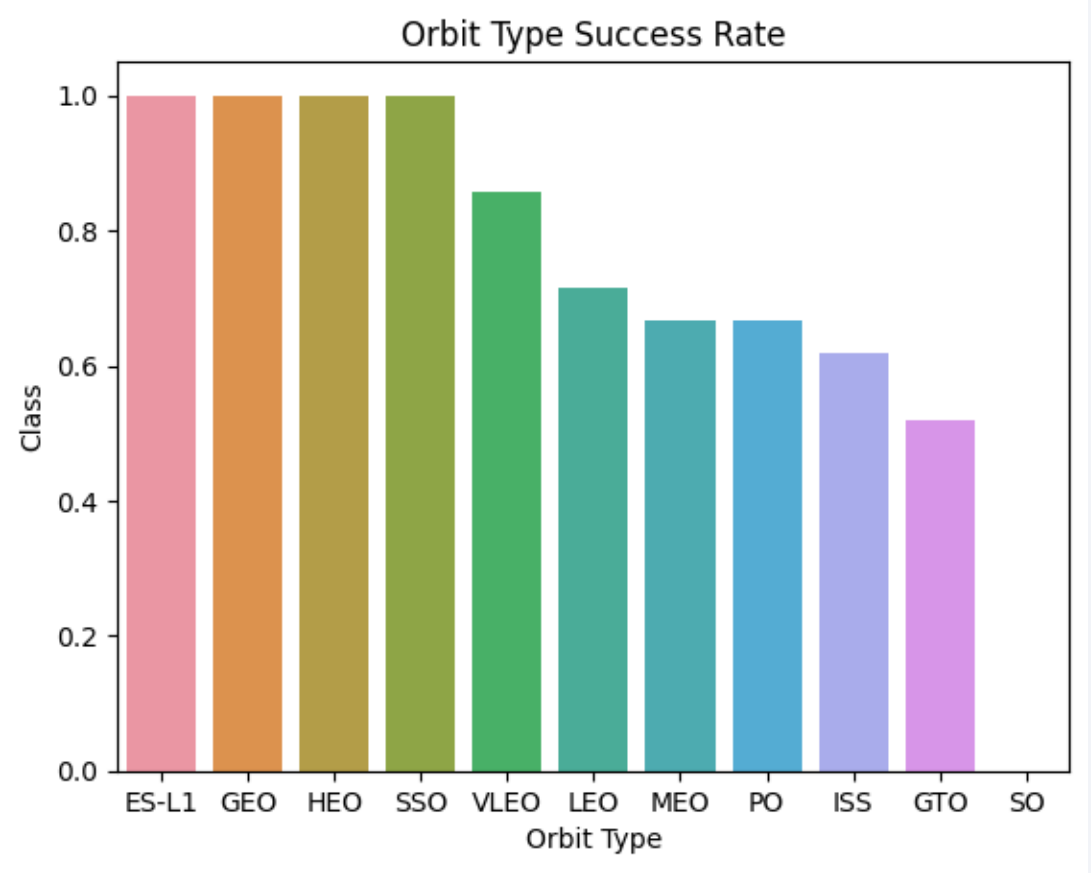
Payload vs. Launch Site

- For the VAFB-SLC launch site there are no rockets launched for heavy payload masses (>10k kg)
- Payload mass <10k kg have no significant difference between successful and failed launches at CCAFS launch site



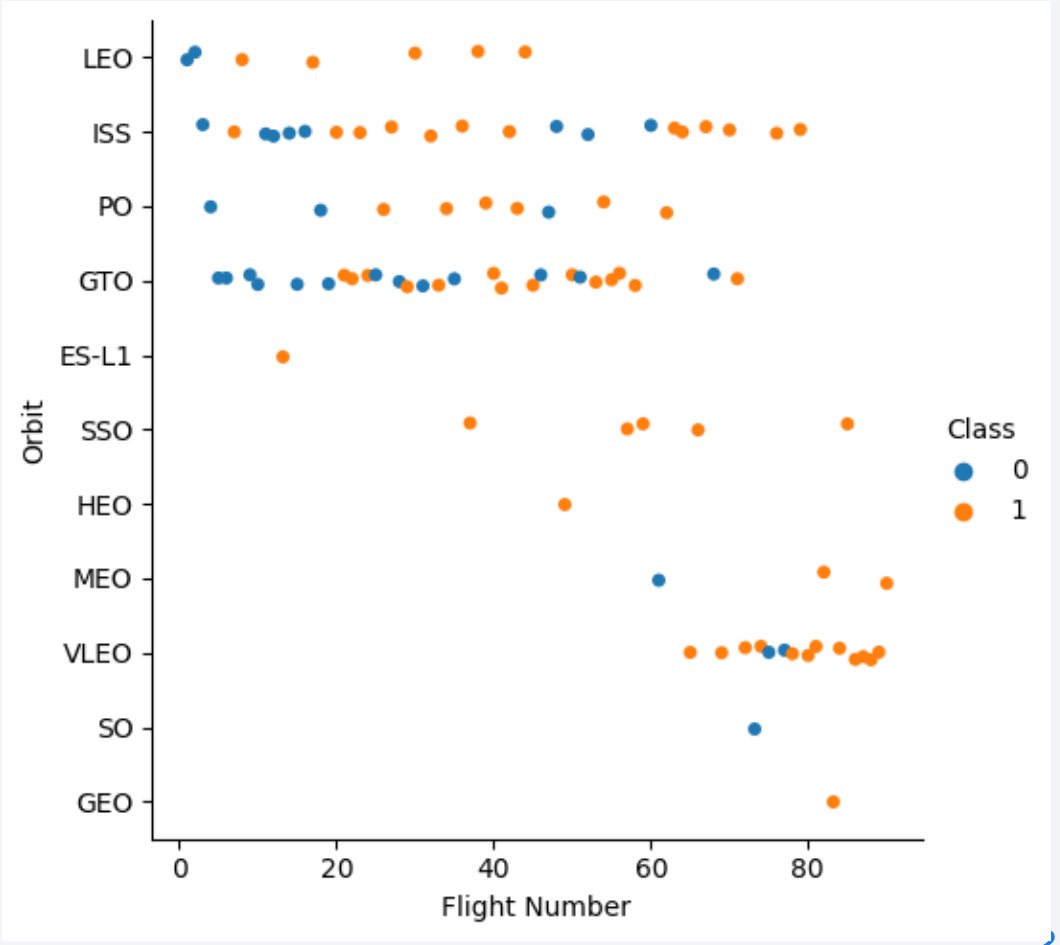
Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO had highest success rates
- SO had lowest success rate



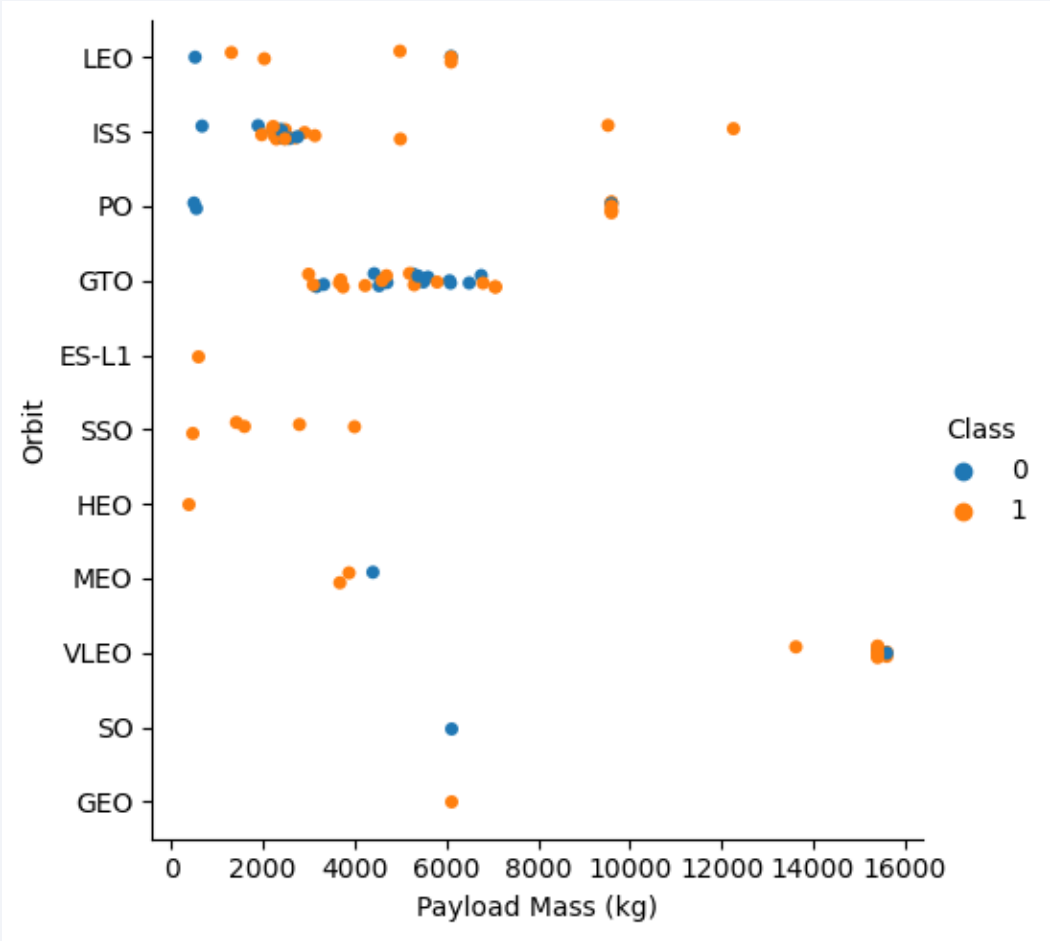
Flight Number vs. Orbit Type

- In LEO and PO orbit the success appears related to the number of flights
- No visible relationship between flight number and success in GTO orbit



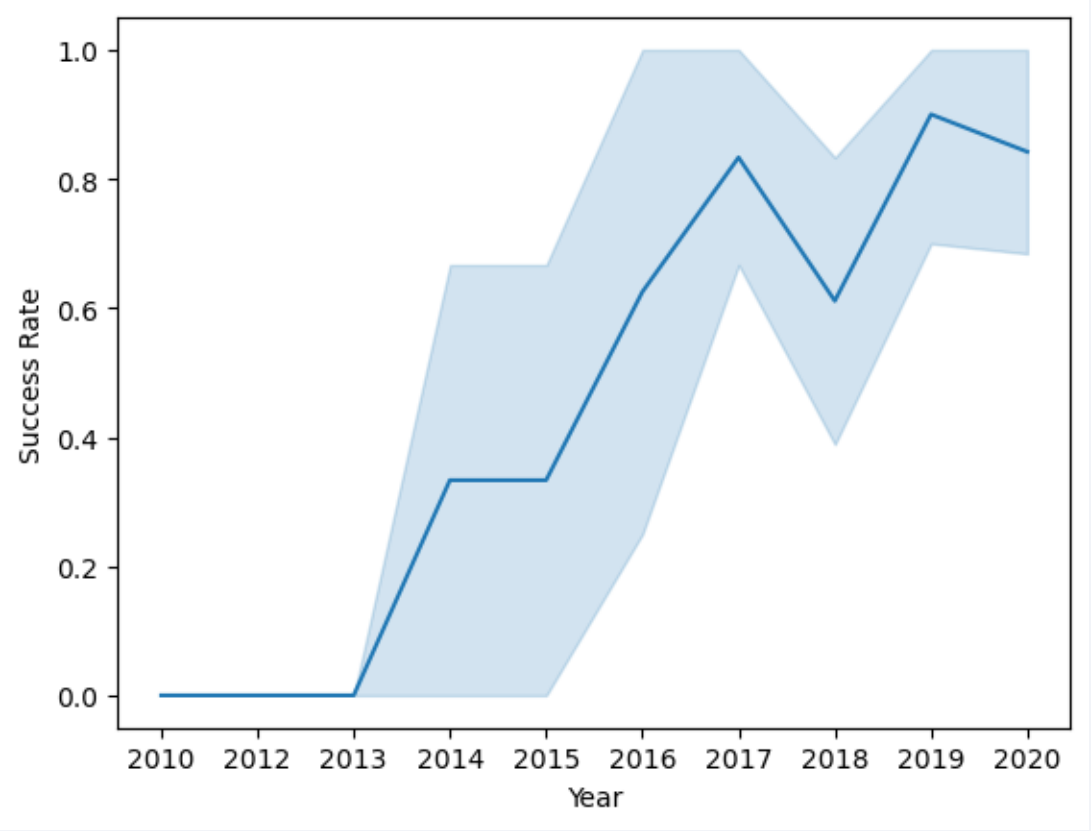
Payload vs. Orbit Type

- With heavy payloads the successful landing or positive landing rate are more for LEO, ISS, and PI
- For GTO we cannot distinguish as there is a seemingly even mix of positive and negative class



Launch Success Yearly Trend

- Success rate has increased overall from 2013 to 2020



All Launch Site Names

- Unique launch sites
- select distinct "Launch_Site" from SPACEXTABLE

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

Launch Site Names Begin with 'CCA'

- select * from SPACEXTABLE where "Launch_Site" like "CCA%" limit 5

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Customer" like "NASA (CRS)"

sum(PAYLOAD_MASS__KG_)
45596.0

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Booster_Version" like "F9 v1.1%”

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- select min("Date") from SPACEXTABLE where "Landing_Outcome" like "Success (ground pad)”

min("Date")
01/08/2018

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- select "Booster_Version" from SPACEXTABLE where "Landing_Outcome" like "Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Success
  - select count("Mission_Outcome") from SPACEXTABLE where "Mission_Outcome" like "Success%”

count("Mission_Outcome")
100

- Failure
  - select count("Mission_Outcome") from SPACEXTABLE where "Mission_Outcome" like "Failure%”

count("Mission_Outcome")
1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- select "Booster_Version", "PAYLOAD_MASS__KG_" from SPACEXTABLE where PAYLOAD_MASS__KG_ == (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE) order by "Booster_Version"

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600.0
F9 B5 B1048.5	15600.0
F9 B5 B1049.4	15600.0
F9 B5 B1049.5	15600.0
F9 B5 B1049.7	15600.0
F9 B5 B1051.3	15600.0
F9 B5 B1051.4	15600.0
F9 B5 B1051.6	15600.0
F9 B5 B1056.4	15600.0
F9 B5 B1058.3	15600.0
F9 B5 B1060.2	15600.0
F9 B5 B1060.3	15600.0

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- select substr(Date, 4, 2) as month, "Landing_Outcome", "Booster_Version", "Launch_Site" from SPACEXTABLE where "Landing_Outcome" like "Failure (drone ship)%" and substr(Date,7,4)='2015'

month	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- 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
- select "Landing_Outcome", count(*) from SPACEXTABLE where "Date" between '04/06/2010' and '20/03/2017' group by "Landing_Outcome" order by count(*) desc

Landing_Outcome	count(*)
Success	20
No attempt	9
Success (drone ship)	8
Success (ground pad)	7
Failure (drone ship)	3
Failure	3
Failure (parachute)	2
Controlled (ocean)	2
No attempt	1


Section 3
Launch Sites
Proximities Analysis

All Launch Sites

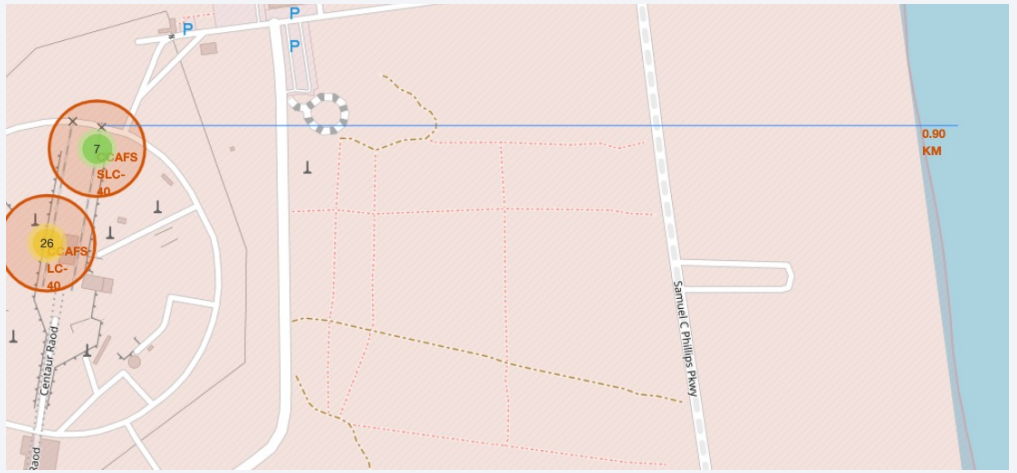
- All launch sites are on the coasts of the United States, located near roads and railroads

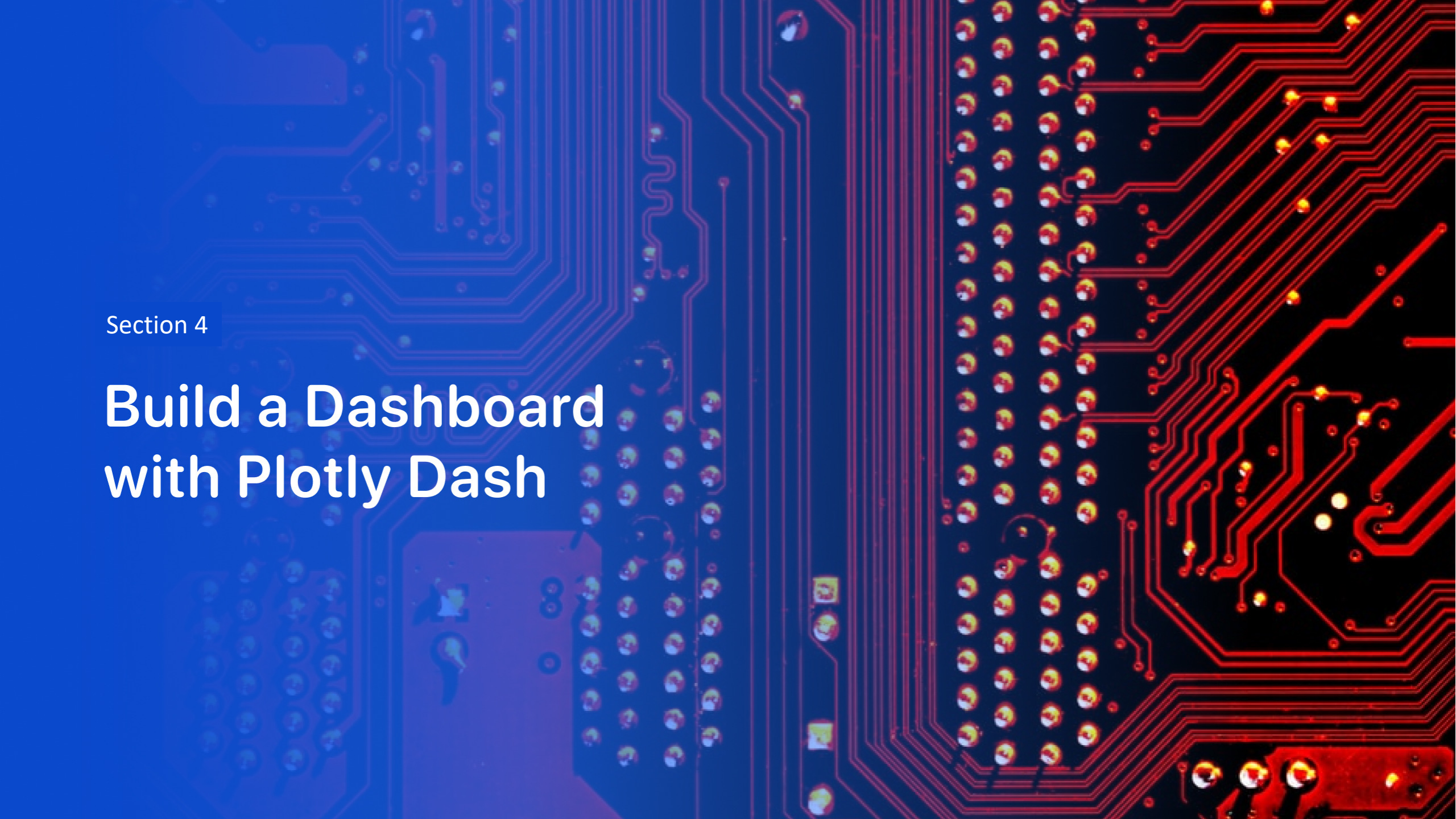
Launch Outcome by Site

- Example of KSCLC-39A launch site with marker cluster of launch outcomes

Launch Site and Proximities

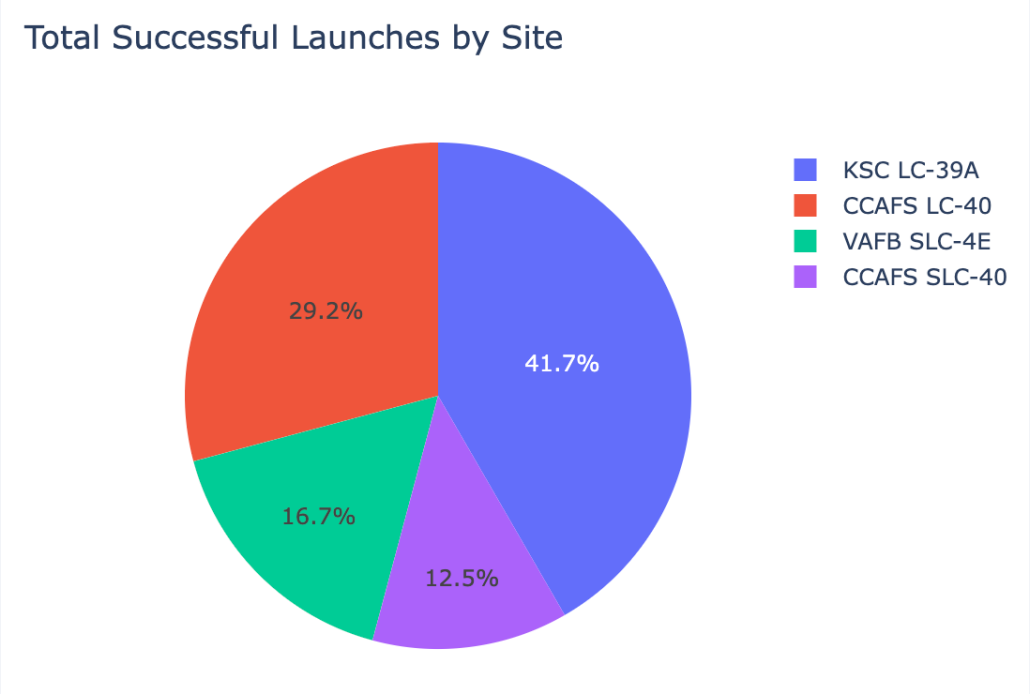
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? No
- Do launch sites keep certain distance away from cities? No
- Ex: CCAFSLC-40 is near roads and away from inhabited areas, and 0.9 km to the sea




Section 4
Build a Dashboard with Plotly Dash

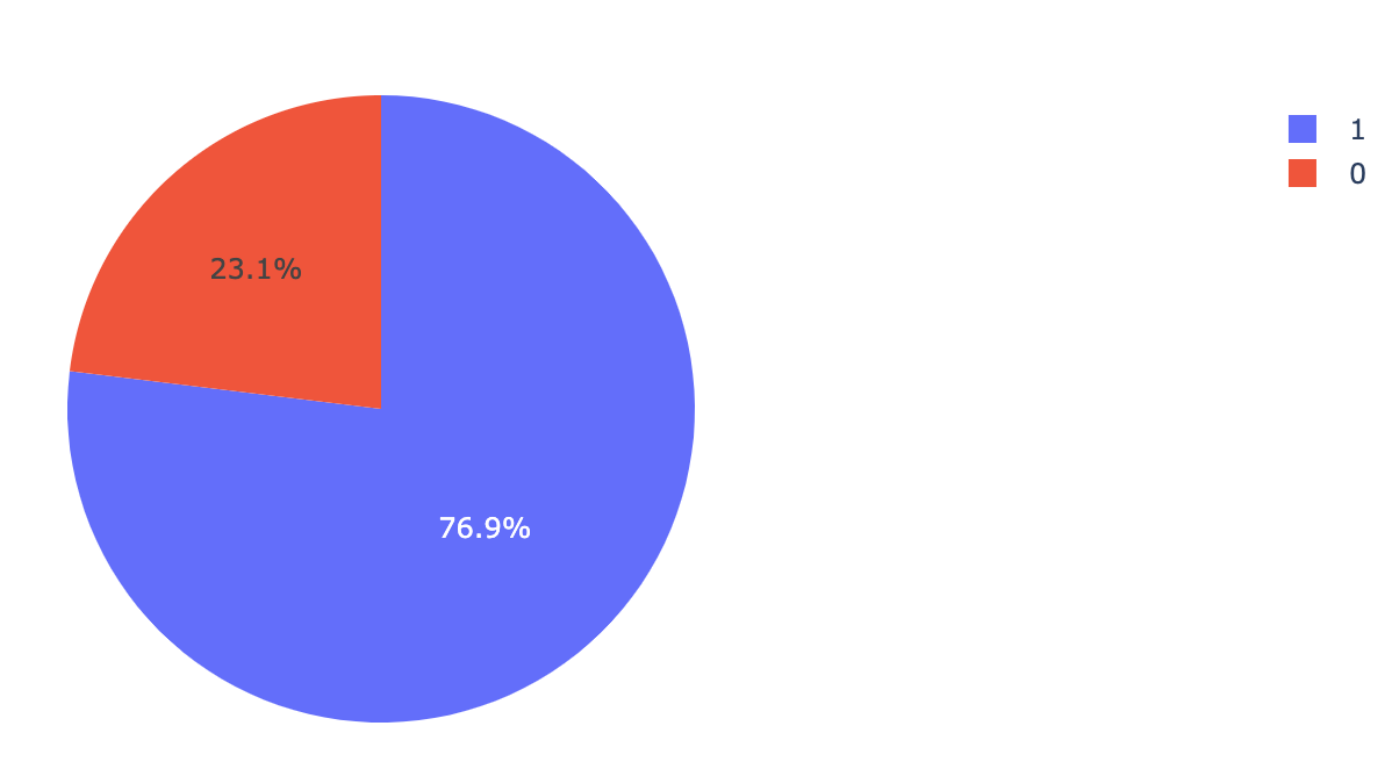
Proportion of Successful Launches Across All Sites

- KSC LS-39A has the highest number of successful launches as compared to the other 3 launch sites



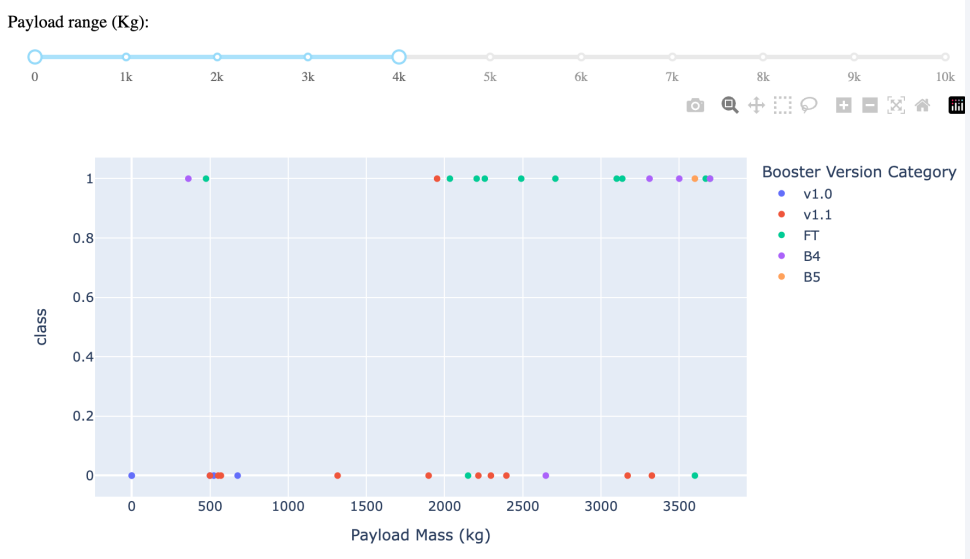
Launch Success Ratio for KSC LC-39A

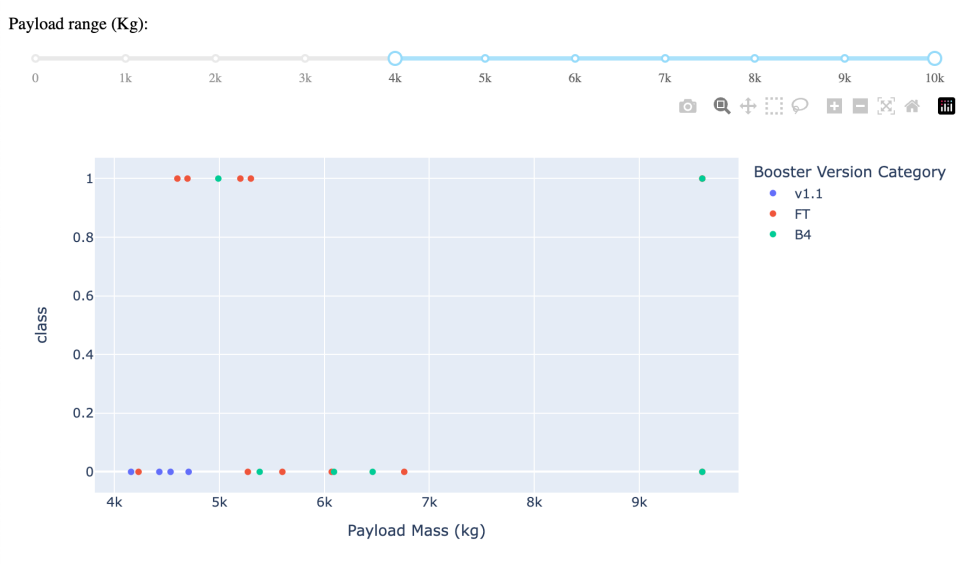
- Launch site with highest launch success ratio is KSC LC-39A



Payload vs. Launch Outcome Across All Sites

- Payload vs. Launch Outcome for all sites, with different payload selected in the range slider
- Success rate for lighter payloads is higher than that of heavier payloads



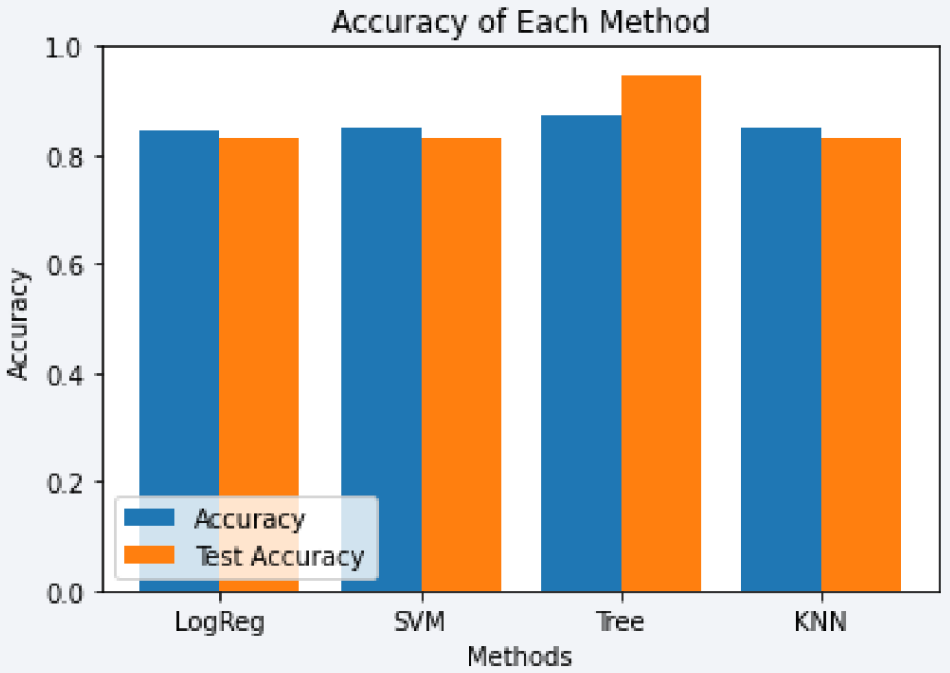


Section 5

Predictive Analysis (Classification)

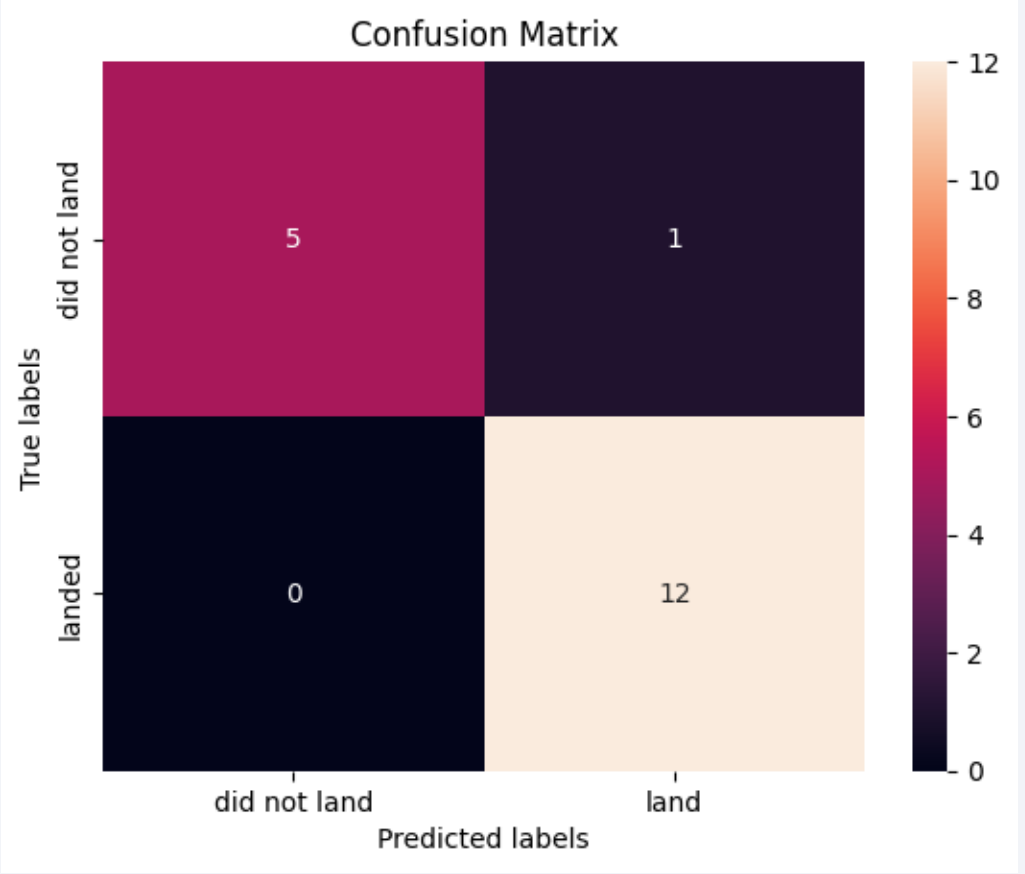
Classification Accuracy

- Decision tree classifier had highest classification accuracy at >88%



Confusion Matrix

- Decision tree can distinguish between the different classes without any significant issues (low false positives, no false negatives)



Conclusions

- Best launch site is KS CLC-39A (most number of successful launch outcomes)
- Lighter payloads correlate to more successful launches, as do more consecutive launches (increased flight numbers)
    - Payload mass isn't the only deciding factor, as heavier payloads above 7000 kg are also able to launch successfully more often than they fail at various launch sites
- Launch outcomes have become more successful over time, year over year
- Decision tree classifier can accurately predict successful launch outcomes, leading to better competing bids and potentially increased profits for SpaceX competitors

Appendix

n/a

Thank you!