ARLT MATIAS

IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Arlt Matias
10/dec/2024

Outline

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

Executive Summary

- Summary of methodologies:
  - Data collection: spaceX-API and Wikipedia web-scraping
  - Data wrangling: cleaning and feature engineering
  - Perform exploratory data analysis (EDA) using visualization and SQL
  - Perform interactive visual analytics using Folium and Plotly Dash
  - Perform predictive analysis using classification models
- Summary of all results

Introduction

Project Background and Context

- SpaceX is a private aerospace manufacturer and space transportation company, known for its ambitious goals, including reducing space transportation costs and enabling the colonization of Mars. The company frequently launches various missions, including satellites, cargo for the International Space Station (ISS), and crewed flights.
- For this project, we are analyzing SpaceX launch data to uncover meaningful insights related to the success and failure of space launches. The dataset includes information on launch details, mission types, launch sites, rocket types, and outcomes.

Introduction

The goal of this analysis is to answer the following key questions:

1. What factors influence the success or failure of SpaceX launches?
   - Are there any clear patterns in mission characteristics (e.g., launch site, rocket type, payload mass) that correlate with a successful or failed mission?
2. Can we predict the success or failure of future SpaceX missions?
   - Using the historical launch data, can we create a predictive model that can forecast the outcome of future launches based on similar features?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX-API and Wikipedia web-scraping
- Perform data wrangling
  - Select useful variables, replace payload null values with mean, feature engineering dates, and success/failure in a new variable.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Create different predictor models logistic regression, SVM, decision tree, and KNN.

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

Define API connection

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
```

Check response

```
response.status_code = 200
```

Define data with .json()

```
data = response.json()
```

Normalize data

```
data = pd.json_normalize(data)
```

```
https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IMB_data_collection_and_wrangling.ip
ynb
```

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

Get wikipedia URL

```
static_url = 
"https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy
_launches&oldid=1027686922"
```

Get response

```
response = requests.get(static_url)
```

Check response

```
response.status_code
```

Parse data into soup

```
soup = BeautifulSoup(response.text, 'html.parser')
```

Get tables

```
html_tables = soup.find_all('table')
```

Get column names

```
header_row = first_launch_table.find_all('th')
column_names = [extract_column_from_header(th) for th in header_row]
```

Get information

```
Use previously defined functions to extract data and create
dataframe
```



```
https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IBM_data_collection_2_web_scrapping.ipy
```

Data Wrangling part one

Select only Falncon 9 booster

```
data_falcon9 = launch_df[launch_df['BoosterVersion'] == 'Falcon 9']
```

Check null

```
data_falcon9.isnull().sum()
```

Replace payload mass null with mean

```
data_falcon9['PayloadMass'].replace(np.nan, data_falcon9['PayloadMass'].mean())
```

Parse data into soup

```
soup = BeautifulSoup(response.text, 'html.parser')
```

Get tables

```
html_tables = soup.find_all('table')
```

Get column names

```
header_row = first_launch_table.find_all('th')
column_names = [extract_column_from_header(th) for th in header_row]
```

Get information

```
Use previously defined functions to extract data and create dataframe
```



```
https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IMB_data_collection_and_wrangling.ip
```

Data Wrangling part two - feature engineering

Check landing outcomes

```
landing_outcomes = df['Outcome'].value_counts()
```

Check bad and good outcomes

```
for i,outcome in enumerate(landing_outcomes.keys()):
    print(i,outcome)
```

Define bad outcomes

```
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
```

Define succes variable as Class

```
landing_class = [0 if outcome in bad_outcomes else 1 for outcome in df['Outcome']]
```

Add succes variable to dataframe

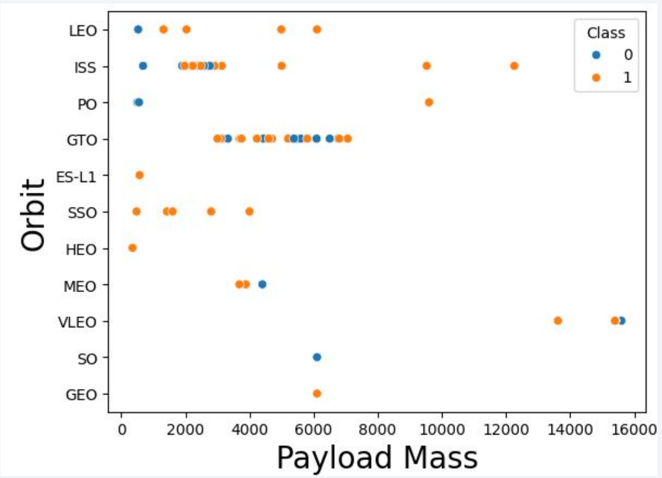
```
df['Class']=landing_class
```

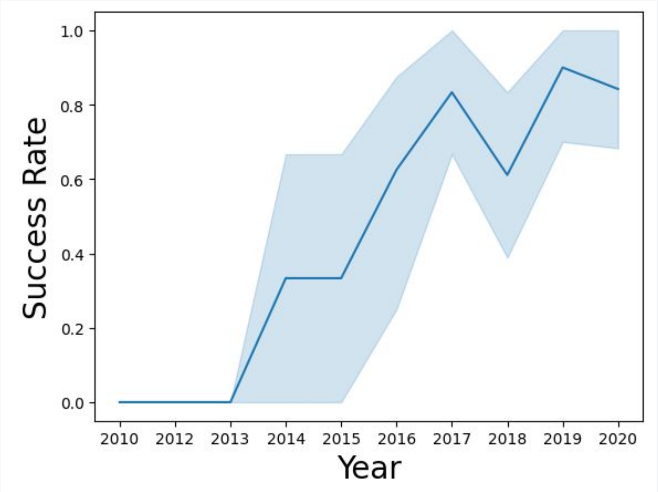

https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IBM_data_wrangling.ipynb

EDA with Data Visualization

- **Scatter plot based on success rate:**
  - Payload mass in relation to flight number:
    - Analyze if experience in regards of payload mass increased success.
  - Launch site in relation to flight number:
    - Analyze if launch site experience increased success.
  - Payload mass in relation to launch site:
    - Analyze if launch site success correlates to payload mass
  - Orbit destination in relation to flight number:
    - Analyze if experience with orbit destination increased success.
  - Orbit destination in relation to payload mass:
    - Analyze if payload mass correlates with success rate based on orbit destination.
- **Bar plot:**
  - Success rate in relation to orbit destination
- **Line chart:**
  - Success rate progression related to year of launch

Example data visualization





https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IBM_EDA_with_visualizations.ipynb

EDA with SQL

- `%sql PRAGMA table_info('SPACEXTABLE')`
- `%sql SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE`
- `%sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5`
- `%sql SELECT SUM(PAYLOAD_MASS__KG_) AS 'Total payload mass carried by boosters launched by NASA (CRS)' FROM SPACEXTABLE WHERE Customer LIKE '%NASA%'`
- `%sql SELECT AVG(PAYLOAD_MASS__KG_) AS 'Average payload mass carried by booster version F9 v1.1' FROM SPACEXTABLE WHERE Booster_Version LIKE '%F9 v1.1%'`
- `%sql SELECT MIN(Date) AS 'Date of first succesful landing outcome in ground pad' FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%Success (ground pad)%'`
- `%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000`
- `%sql SELECT Mission_Outcome, COUNT(*) AS 'Total' FROM SPACEXTABLE GROUP BY Mission_Outcome`
- `%sql SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)`
- `%sql SELECT substr(Date, 6, 2) AS Month,Booster_Version,Launch_Site,Landing_Outcome FROM SPACEXTABLE WHERE substr(Date, 1, 4) = '2015' AND Landing_Outcome = 'Failure (drone ship)'`
- `%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS COUNT FROM SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY COUNT(Landing_Outcome) DESC`

https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IBM_EDA_with_SQL.ipynb

Build an Interactive Map with Folium

- Created a folium United States of America map object.
- Map objects:
  - Marker object for all launch sites with circle child -> Show launch sites
  - Marker cluster for each flight in each launch site -> illustrate flights with colors based on success or failure.
  - Mouse position object to show latitude and longitude in map -> function: user may calculate relative distances.

https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IBM_Interactive_Visual_Analytics_with_Folium.ipynb

Build a Dashboard with Plotly Dash

- Dropdown Menu: Added a dropdown menu to view information for all launch sites or filter by individual launch sites.
- Pie Chart:
  - When "All Sites" is selected, the pie chart displays the distribution of successful launches by launch site.
  - When a specific launch site is selected, the pie chart shows the success vs. failure percentage for that site.
- Scatter Plot:
  - Displays the relationship between payload mass and launch success.
  - Includes a payload mass slider to dynamically adjust the scatter plot based on the selected range.

https://github.com/MatArlt/Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Import databases to variables X and Y

```
X = independent variables,  Y = data['Class'].to_numpy()
```

Scale and transform X

```
transform = preprocessing.StandardScaler()
X = transform.fit_transform(X)
```

Divide train and test data

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=2)
```

Create prediction models for logistic, KNN, decision tree, and SVM

```
Create parameters, create model, create GridSearchCV, fit model, test tuned parameters and accuracy.
Parameters = {parameters}
model = model()
model_cv = GridSearchCV(moderl, parameters, cv=10)
model_cv.fit(X_train, Y_train)
model_cv.best_params_, model_cv.best_score_
```

Test accuracy

```
model_cv.score(X_test, Y_test)
```

Create confusion matrix

```
yhat=svm_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```

Compare accuracy scores and F1 metrics

```
from sklearn.metrics import accuracy_score, f1_score
```


https://github.com/MatArlt/Data-Science-Capstone/blob/main/Capstone_project_Data_Science_IMB_data_collection_and_wrangling.ipynb

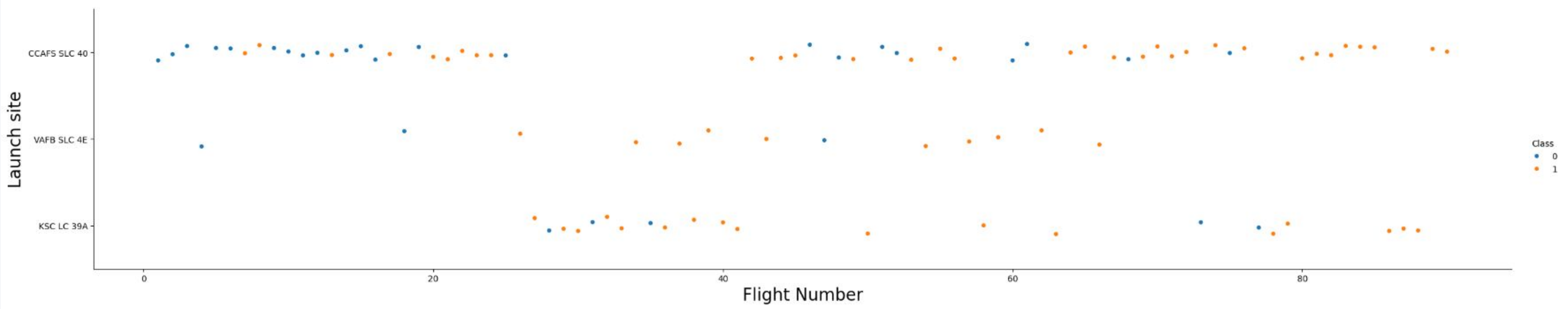
Results

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

Section 2

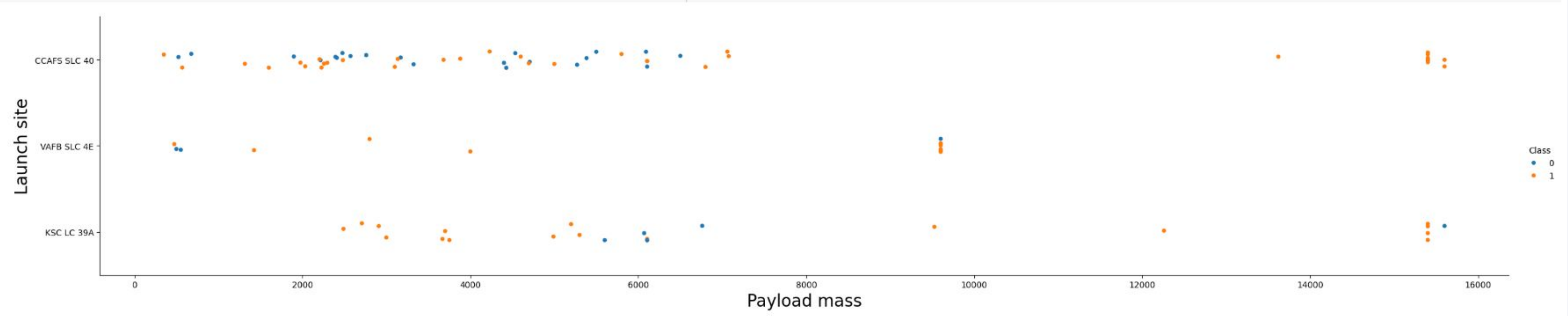
Insights drawn from EDA

Flight Number vs. Launch Site



- Launch site related to flight number with success/failure encoded in yellow/blue respectively
- We clearly observe a trend towards improvement with flight number progression.

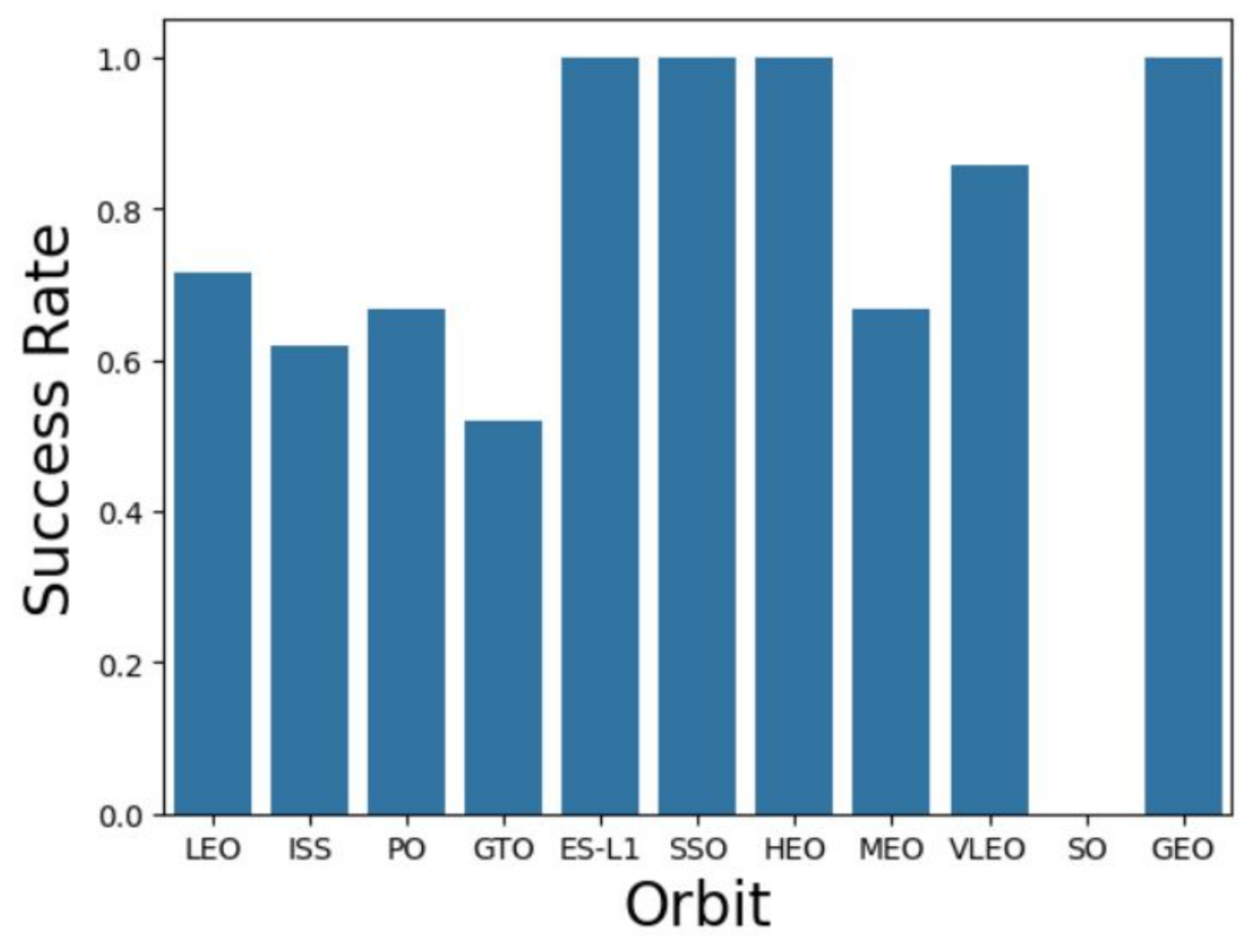
Payload vs. Launch Site



- Launch site related to payload mass with success/failure encoded in yellow/blue respectively
- Higher payload mass tends to be related to improved success rate, predominantly so in launch site CCAFS-SLC 40

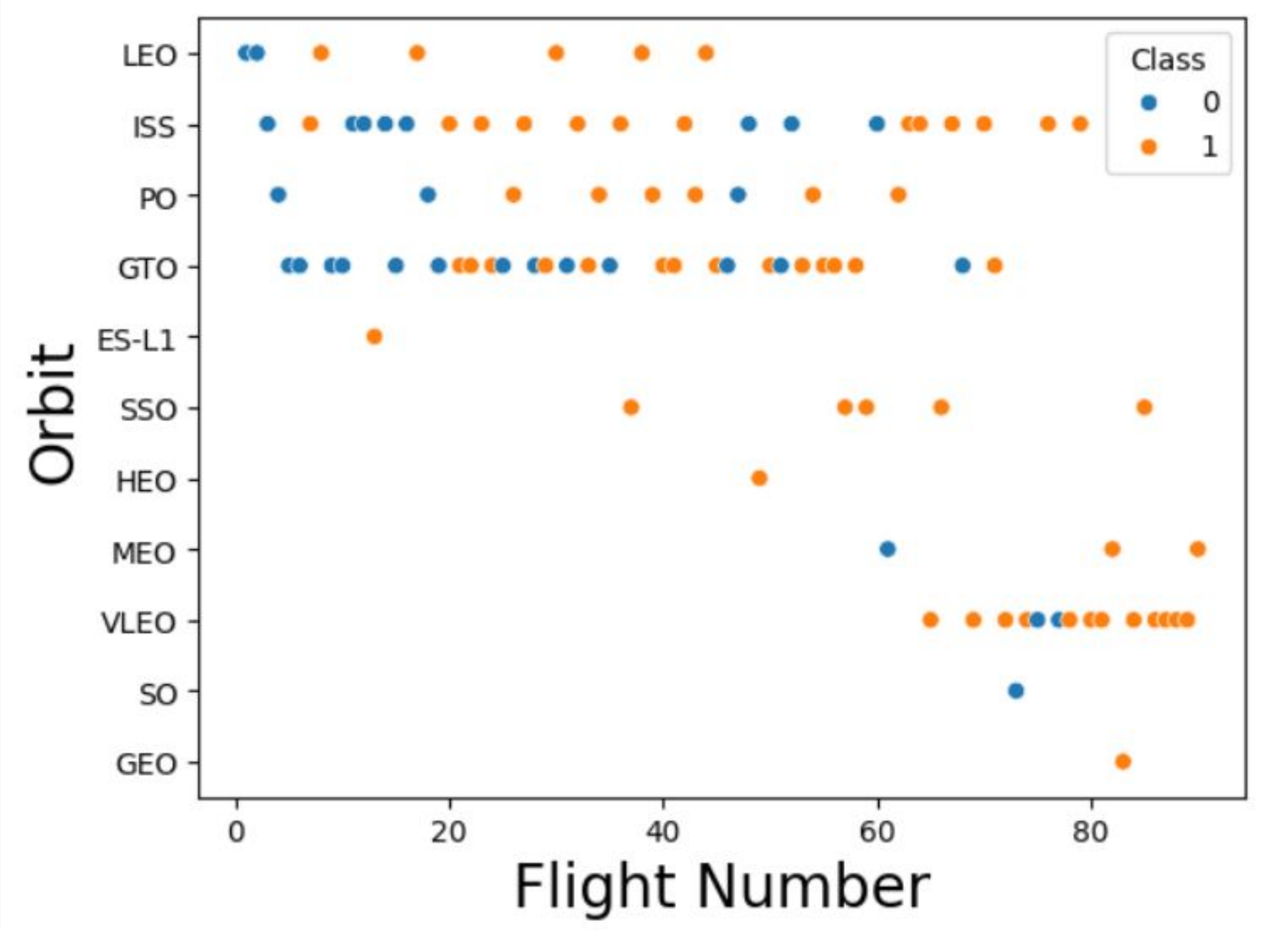
Success Rate vs. Orbit Type

- High success rate (100%):
  - ES-L1, SSO, HEO and GEO
- Moderate-high success rate (>60%):
  - LEO, ISS, PO and VLEO
- Less than 50% success rate:
  - GTO, and SO



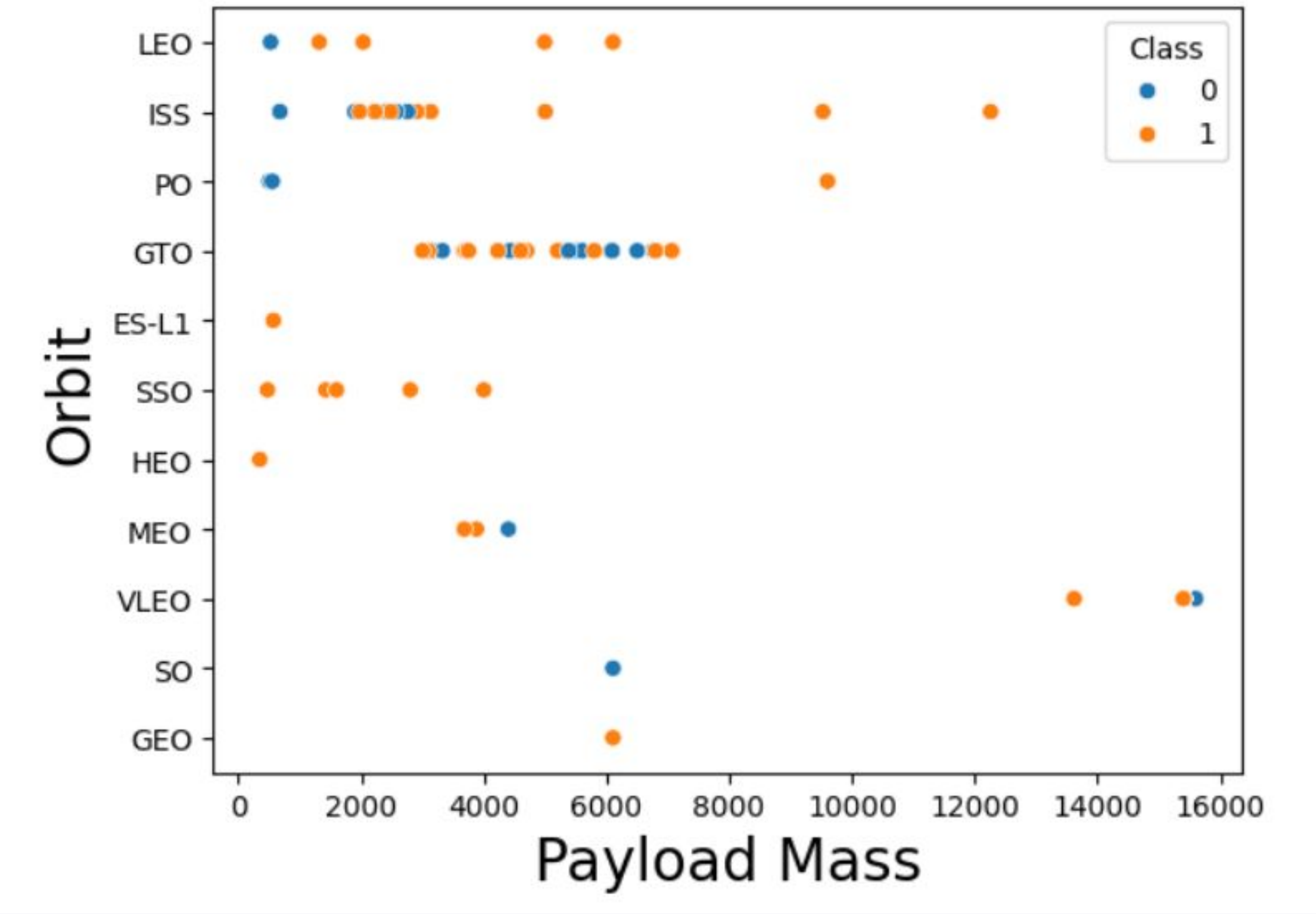
Flight Number vs. Orbit Type

- We continue to observe an overall improvement through the progression in flight number.
- We recognize very few failures after flight 60.



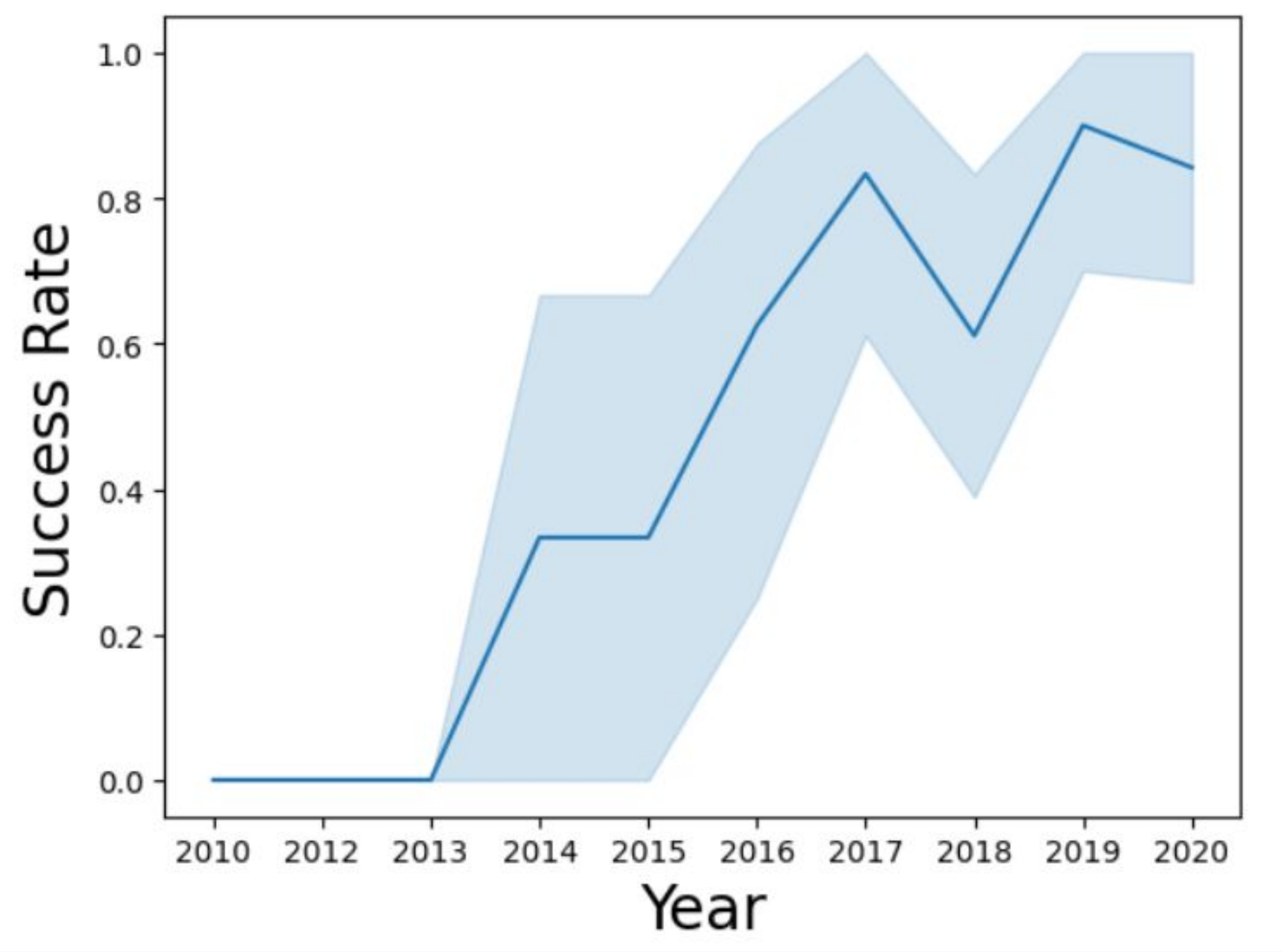
Payload vs. Orbit Type

- Most launches have a payload mass between 500 and 8000.
- Every launch, but one was successful with payloads above 8000.



Launch Success Yearly Trend

- Clear improvement in success rate through the years.
- There was a drawback in the year 2018, but continued to improve after that, reaching an average success rate of approximately 90% in the last years.



All Launch Site Names

- CCAFS LC-40
- VAFB SLC-4
- EKSC LC-39
- ACCAFS SLC-40

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`

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

- Calculate the total payload carried by boosters from NASA

Total payload mass carried by boosters launched by NASA (CRS)
107010

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

Average payload mass carried by booster version F9 v1.1
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

Date of first succesful landing outcome in ground pad
2015-12-22

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

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 (not to be confused with landing outcome, mission outcome refers to payload successful delivery.
- Total successful missions: 100
- Failure missions: 1

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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

-

Booster_Version
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

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

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

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

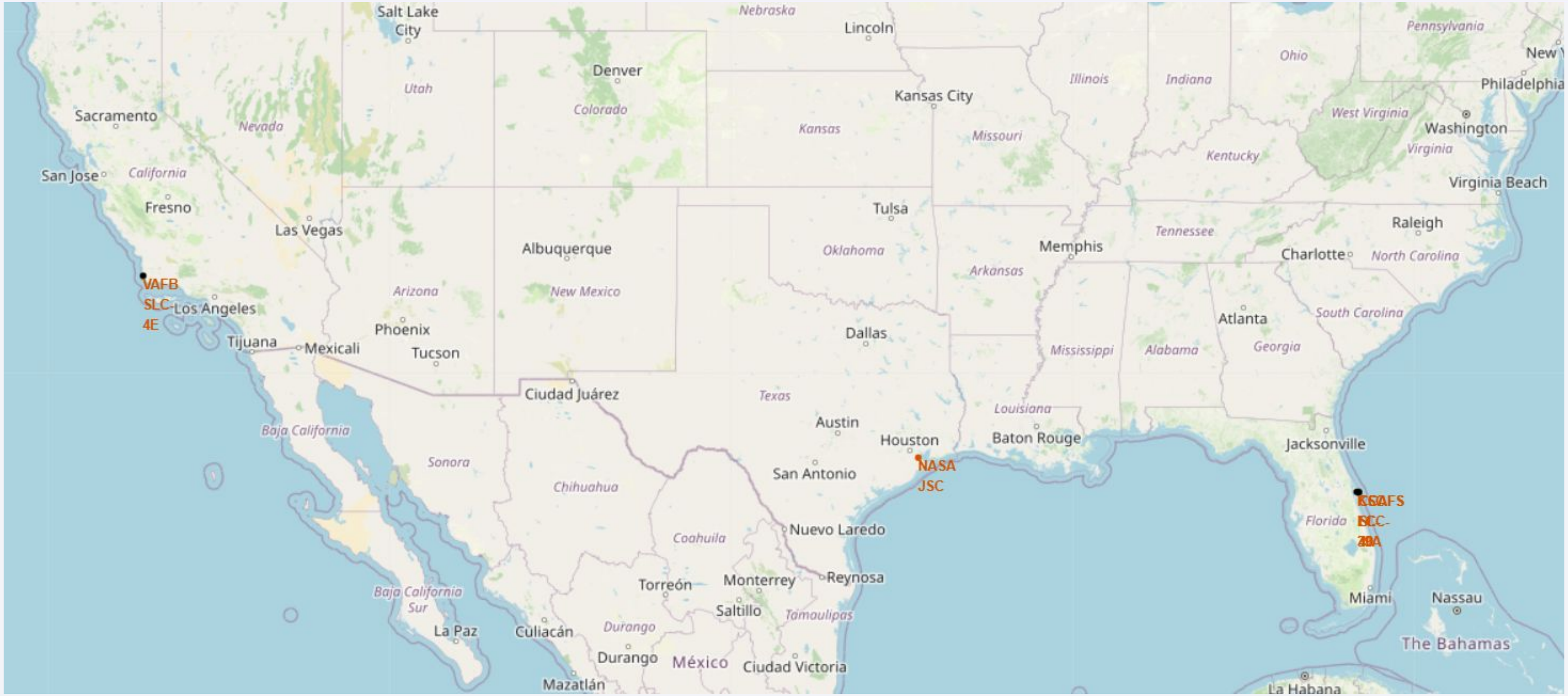
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

Landing_Outcome	COUNT
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

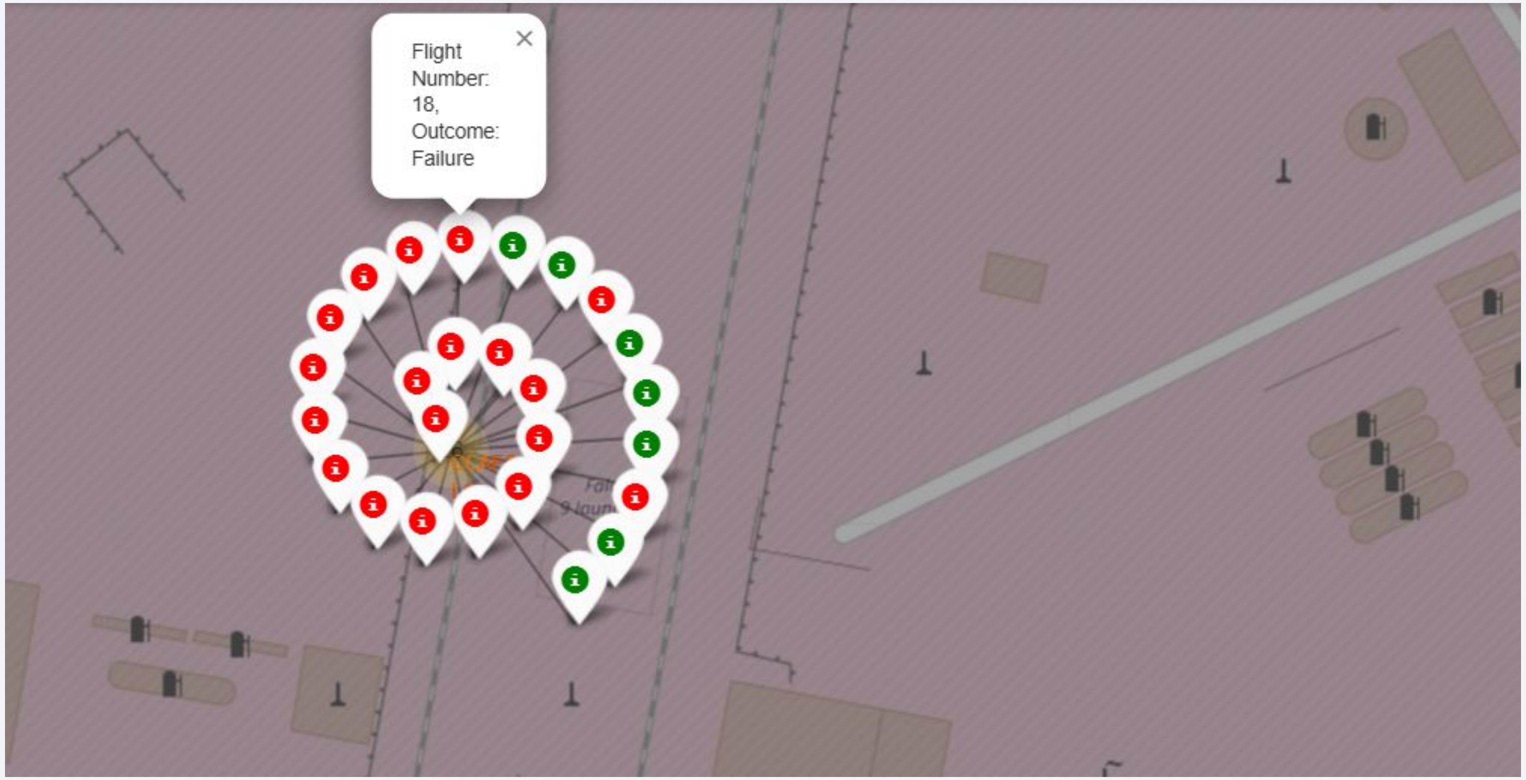
Launch Sites Proximities Analysis

Map of the United States of America with launch sites



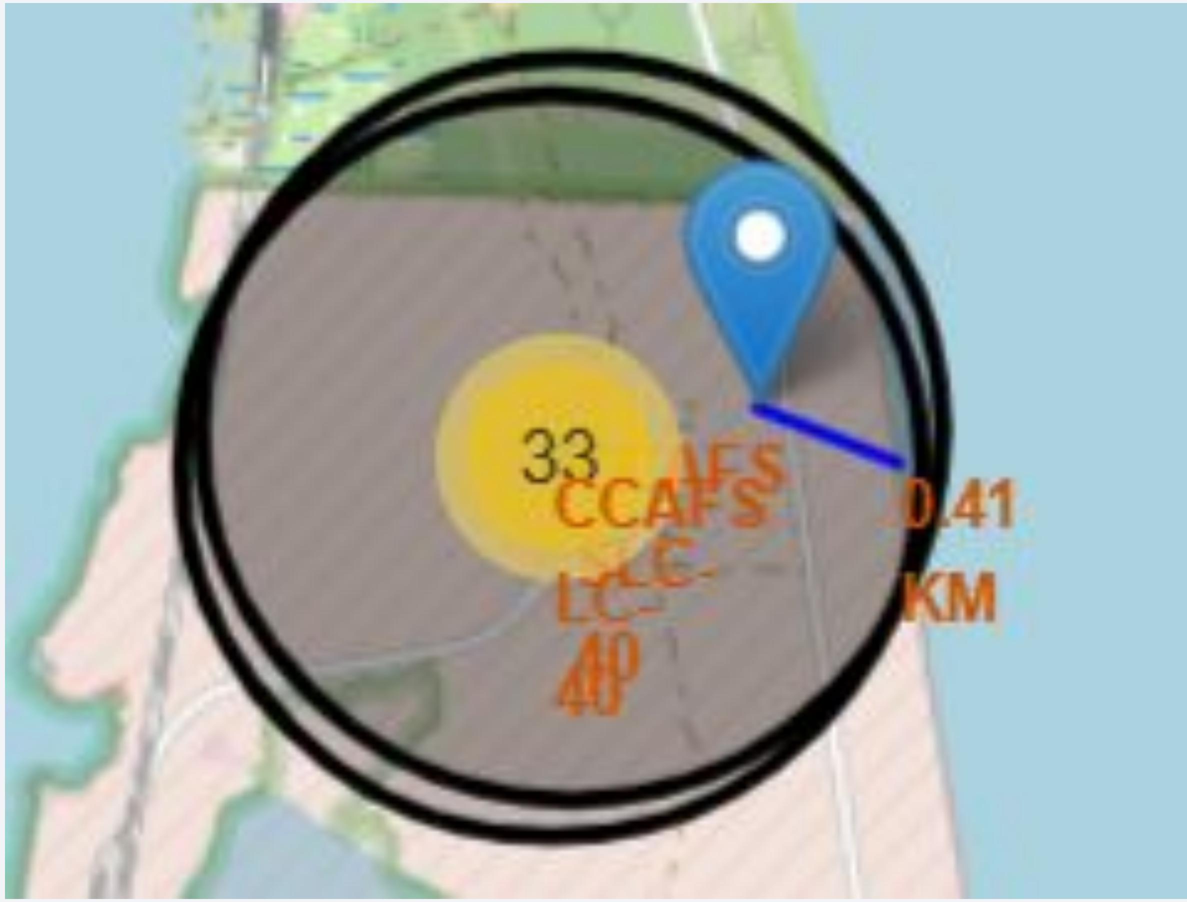
- There are four launch sites: Houston, California and two in Florida.

Launch outcomes



- Map mark clusters showing each flight and encoded by outcome (green success, red failure)

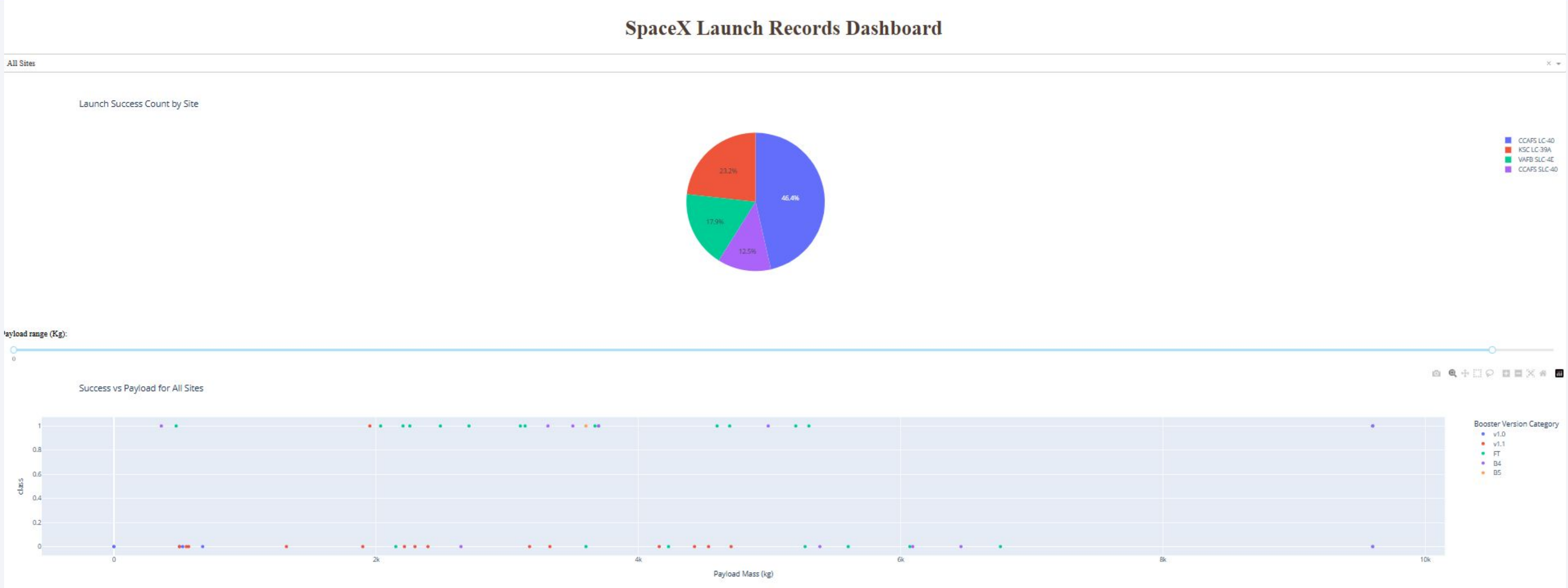
Example of markes with proximities: Coastline



Section 4

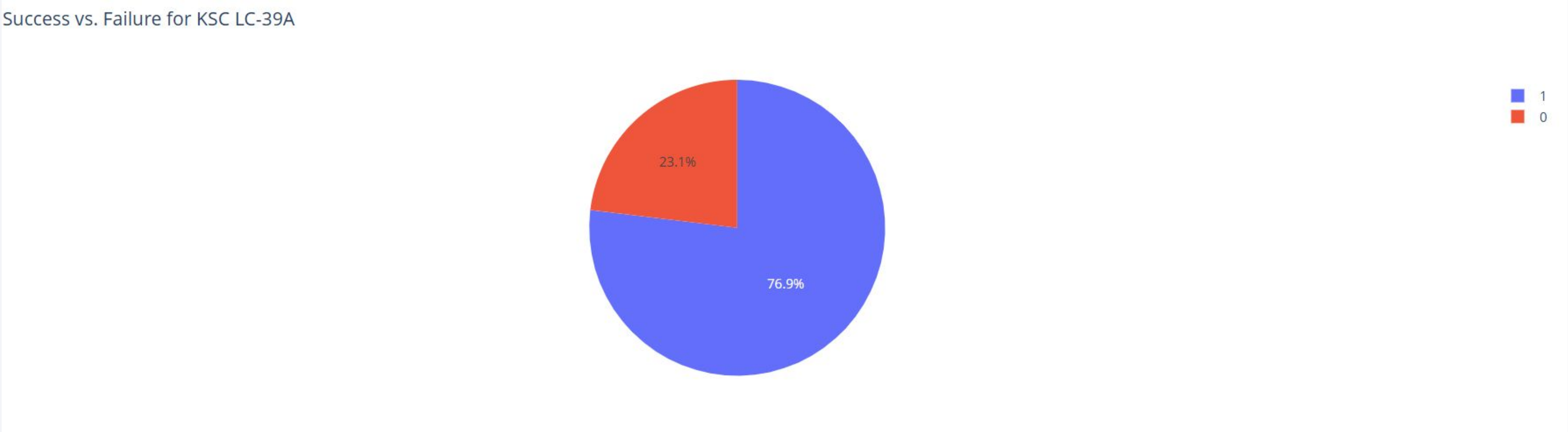
Build a Dashboard with Plotly Dash

SpaceX Launch Records Dashboard

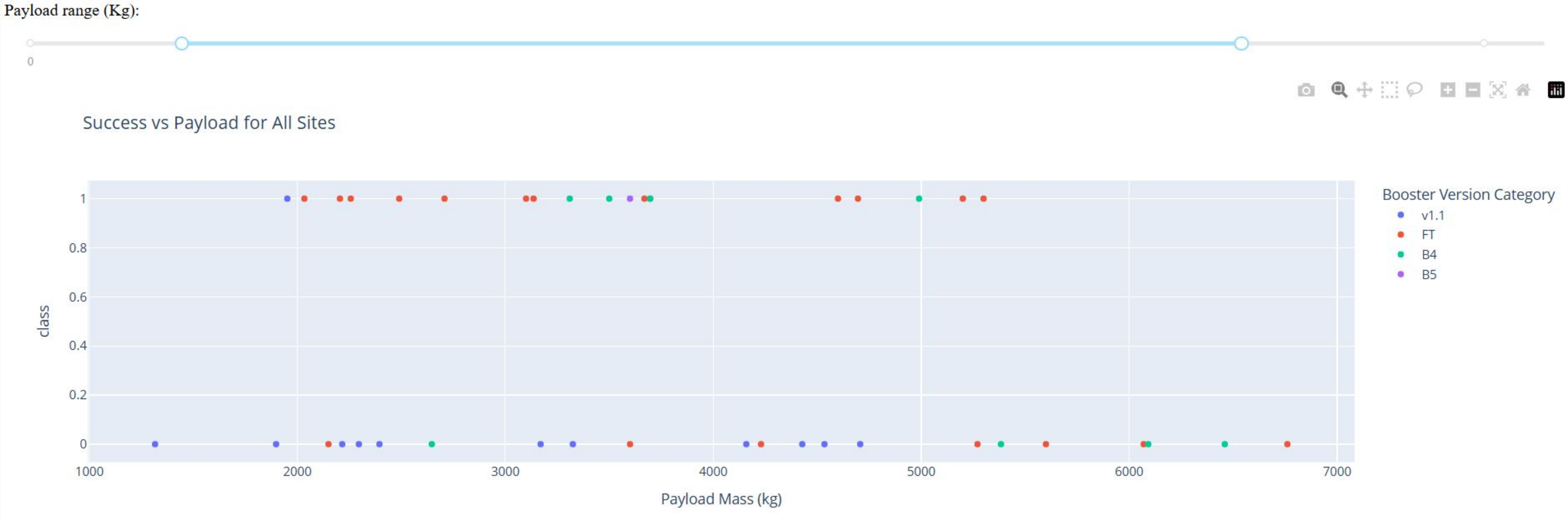


- Dropdown to filter through launch sites
- Pie chart: if all sites selected show sum percentage of successful launches, if specific site, shows percentage of success and failure.
- Scatter plot: payload mass vs success in regards to booster version.
- Slider to filter through payload masses

Piechart for the launch site with highest launch success ratio



Success vs payload mass

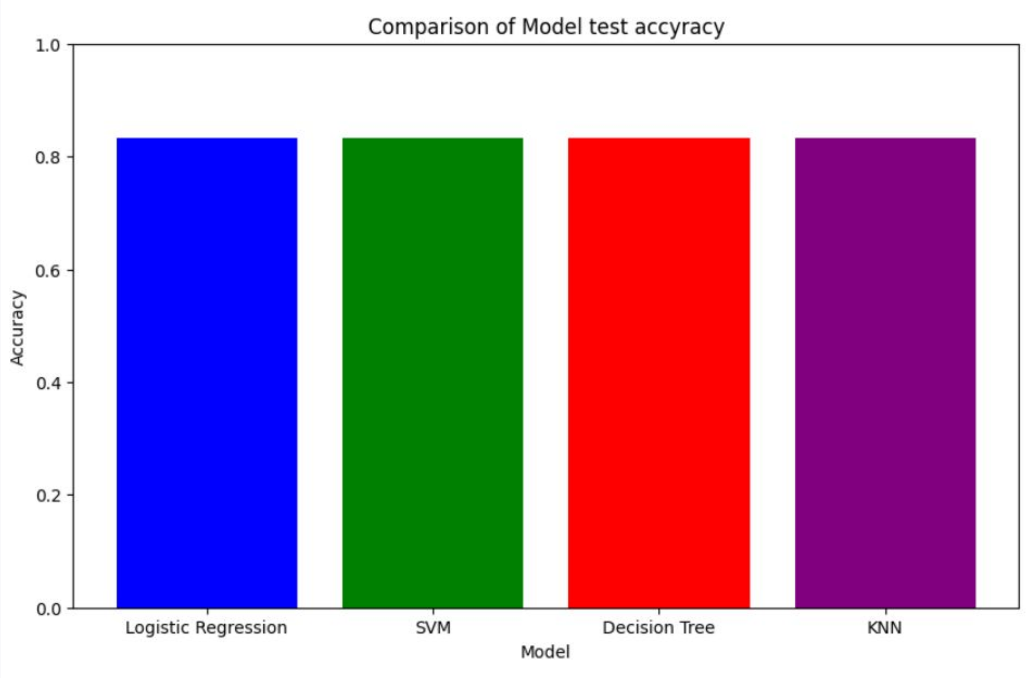


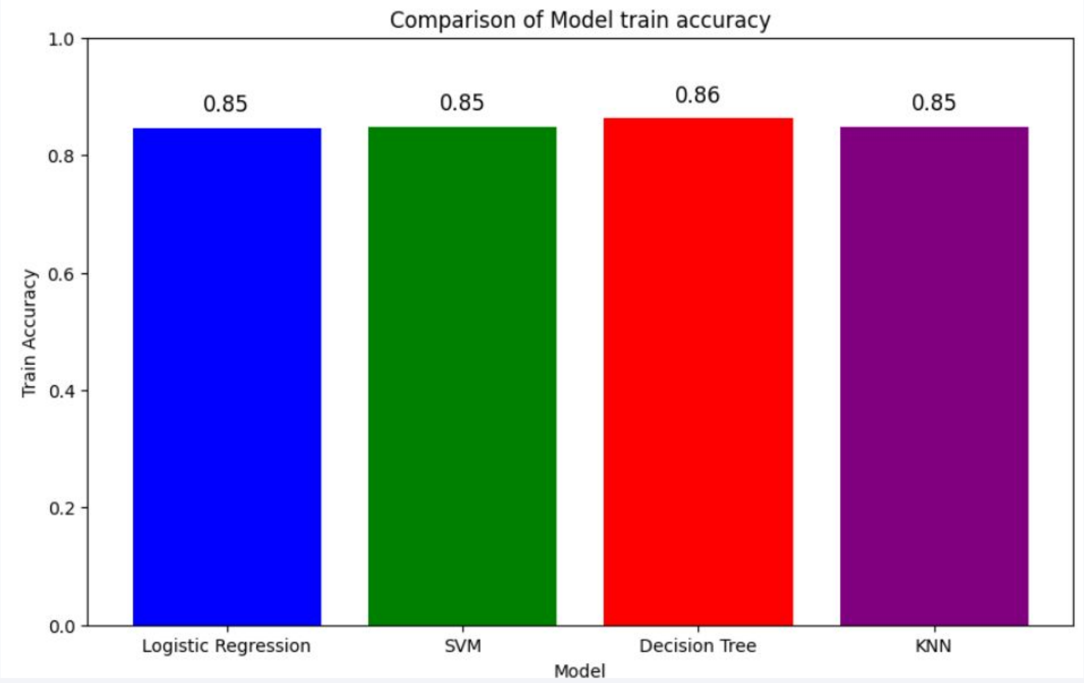
- FT booster version seems to have the most successful rate in the payload range evaluated.

Section 5

Predictive Analysis (Classification)

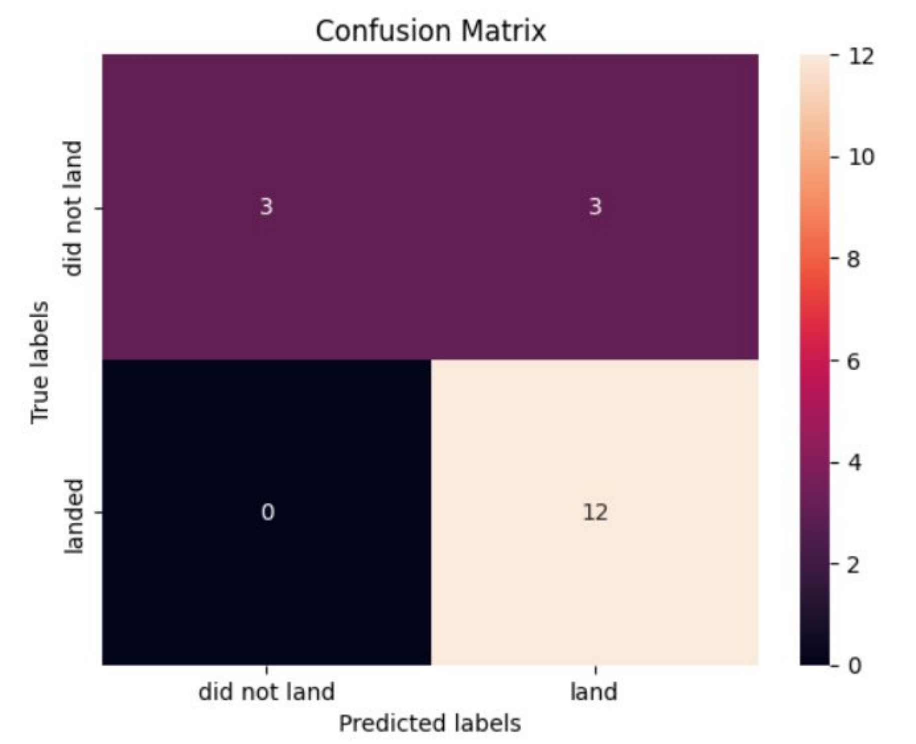
Classification Accuracy





- Logistic regression, SVM and KNN and decision tree share the same test accuracy
- This might be due to small n in the test data.
- Decission tree shows slightly better train accuracy.
- A bigger sample should be analyzed to observe it this trend is significant.

Confusion Matrix



- Confusion matrix for best model:
  - Good True positive prediction
  - Model lacks prediction for failure outcome with 50% false negatives.

Conclusions

- Flight number progression as a measure of time and experience shows a significant increase in success rate.
- This is particulary evident with flights to orbits ES-L1, HEO, VLEO
- KSC LC-39A had the most successful rate of any sites.
- CCAFS C40 had the most successful amount of launches.
- While test accuracy is practically equal for the different models, decision tree shows a slightly better prediction accuracy in train data.
- A bigger sample is needed to determine if this trend persists, and if its, indeed, the best prediction model.

Thank you!

ARLT MATIAS