IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

THAO NGUYEN
06 Apr 2025

Outline

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

Executive Summary

- Summary of methodologies
  - Collect data of SpaceX launches
  - EDA data to find out impacted features that can be used for modelling
  - Test different models & parameters to choose which one results the best result (highest accuracy)
- Summary of all results
  - The features using: Booster Version, Payload Mass, Orbit, Launch Site
  - Decision Tree is the winning model with highest accuracy (86.25%) with parameters as: *{'criterion': 'gini', 'max_depth': 8, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 10, 'splitter': 'best'}*

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.
- Problems you want to find answers
  - If we can predict how 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

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Using request library to crawl data of SpaceX launches on public source
- Perform data wrangling
  - Data was preprocessed & transform & split into train & test set
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Using Grid Search to iterate through different models with multiple parameters to find the best results

Data Collection

- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts
    - Crawl data online using request library & public API (*https://api.spacexdata.com/v4/launches/past*)
    - Preprocess data with correct format & filter needed data
    - Data Scraping
    - Data Wraling

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

1- Request and parse the SpaceX launch data using the GET request

2- Filter the dataframe to only include Falcon 9 launches

3- Dealing with Missing Values

4- GitHub notebook:

(https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/jupyter-labs-spacex-data-collection-api.ipynb)

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

1- Request the Falcon9 Launch Wiki page from its URL

(https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922)

2- Extract all column/variable names from the HTML table header

3- Create a data frame by parsing the launch HTML tables

4- GitHub URL:

(https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/jupyter-labs-webscraping.ipynb)

Data Wrangling

- Describe how data were processed
- You need to present your data wrangling process using key phrases and flowcharts
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose
  - Load the collected data
  - Perform exploratory Data Analysis and determine Training Labels
  - Calculate the number of launches for each site
  - Calculate the number and occurrence of each orbit
  - Calculate the number and occurence of mission outcome of the orbits
  - Create a landing outcome label from Outcome column
  - GitHub URL: *https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb*

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
  - Using Scatter Plot to visualize relationship between: Flight Number & Launch Site, Payload Mass & Launch Site, Success Rate of each Orbit Type, FlightNumber and Orbit type, Payload Mass and Orbit type, etc
  - Using Line Chart to visualize yearly success trend
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose
  - GitHub URL: https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/edadataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
    - Display: names of the unique launch sites in the space mission / 5 records where launch sites 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
    - List: the date when the first succesful landing outcome in ground pad was achieved/ names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000/ total number of successful and failure mission outcomes / names of the booster_versions which have carried the maximum payload mass / records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
    - 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.
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose
    - https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
  - 1: Mark all launch sites on a map
  - 2: Mark the success/failed launches for each site on the map
  - 3: Calculate the distances between a launch site to its proximities
- Explain why you added those objects
  - To find some geographical patterns about launch sites that could be impacted on the success rate
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose
  - https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
  - Pie Chart to show Success & Fail count for each Launch Site
  - Scatter Chart to show Relationship between Payload Mass & Success launch, for each Booster version, can filter for details of Payload Range
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose
  - https://github.com/thaonguyen2601/coursera-final-assignment/blob/main/spacex_dash_app_NHTT.py

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- Perform exploratory Data Analysis and determine Training Labels
    - Create a column for the class
    - Standardize the data
    - Split into training data and test data
    - Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
    - Find the method performs best using test data
- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

Results

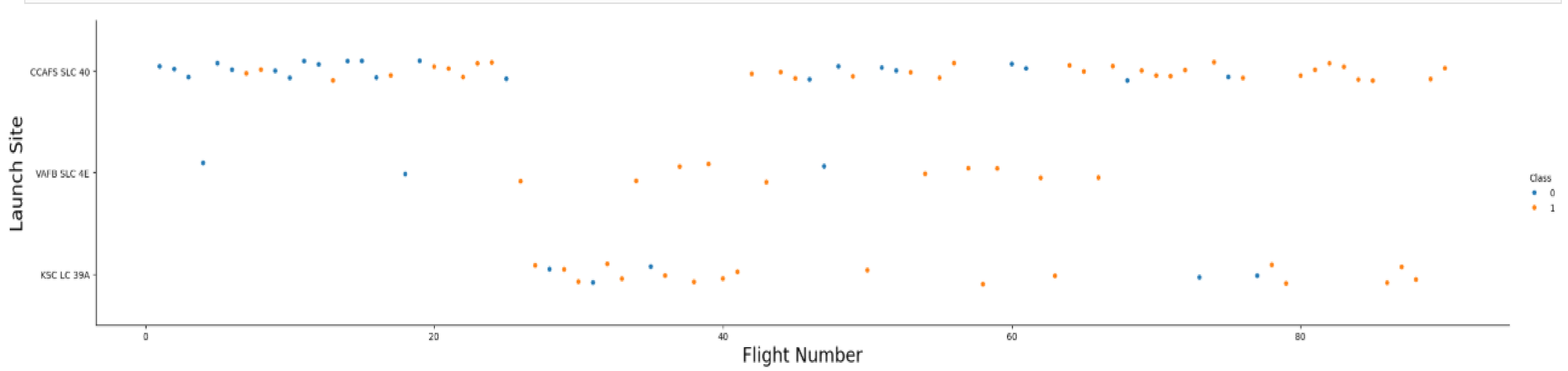
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

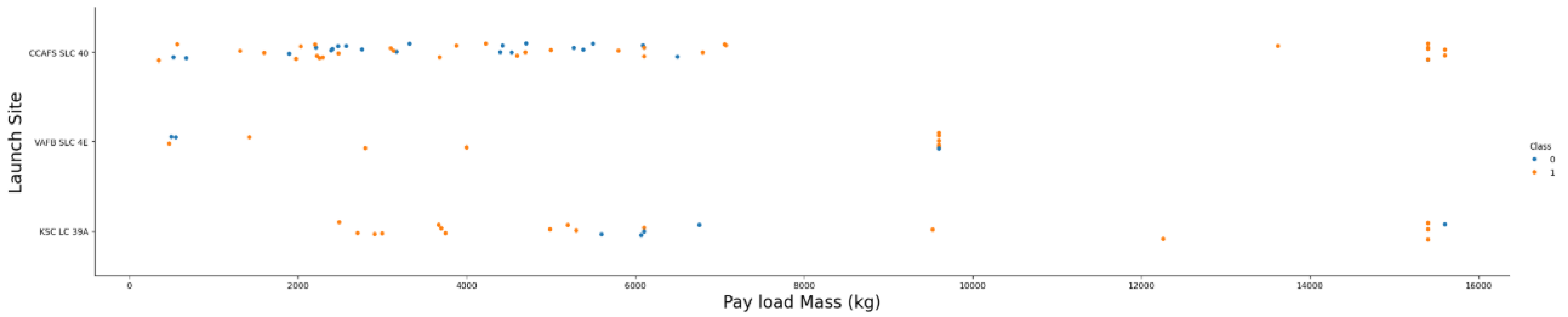
Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site
- Show the screenshot of the scatter plot with explanations
  - Not seeing any significant correlation between Flight Number & Launch Site



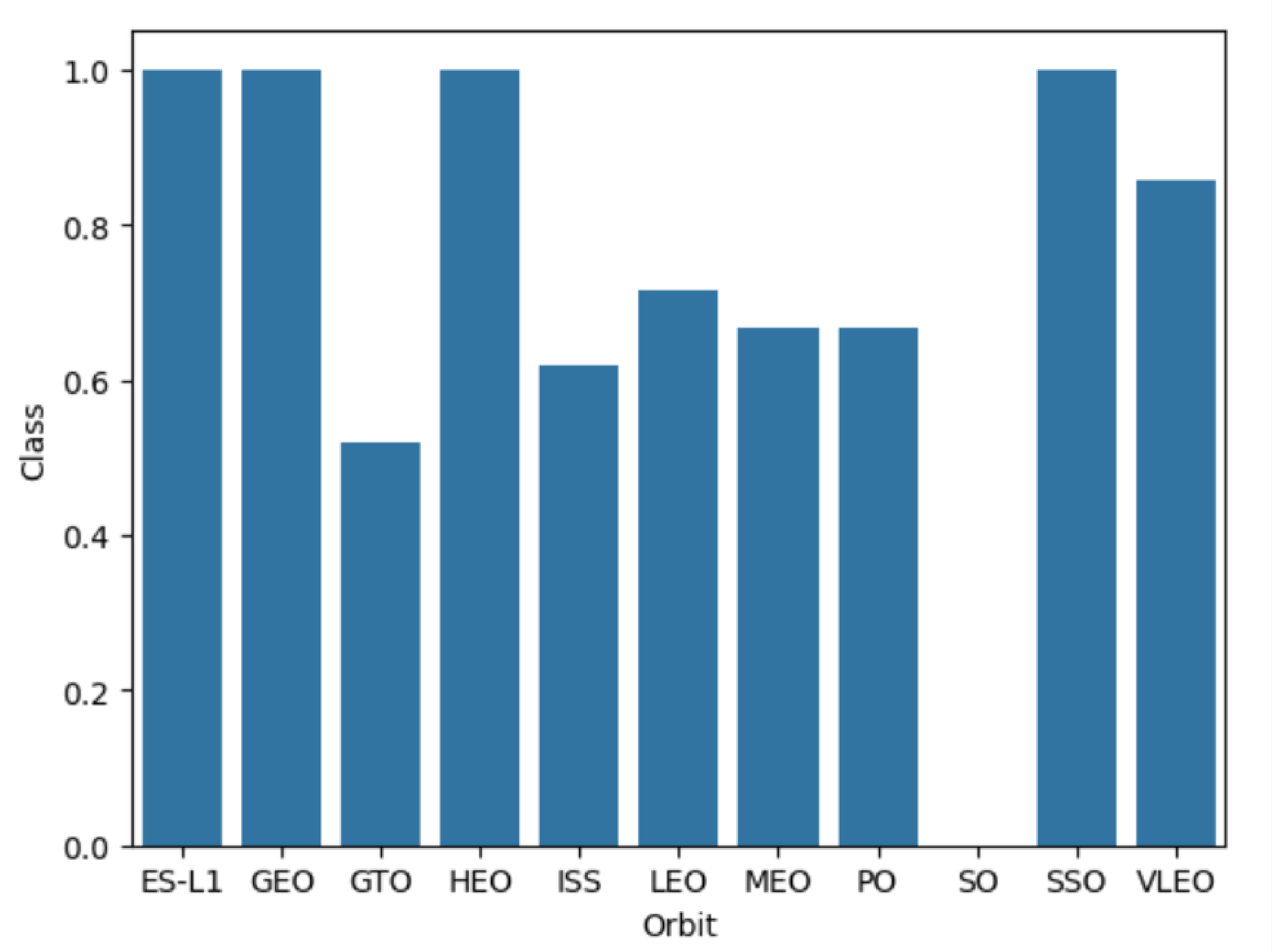
Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site
- Show the screenshot of the scatter plot with explanations
  - Payload > 8000 kg has higher success rate
  - Site VAFB SLC 4E doesn't have any Payload > 4000kg



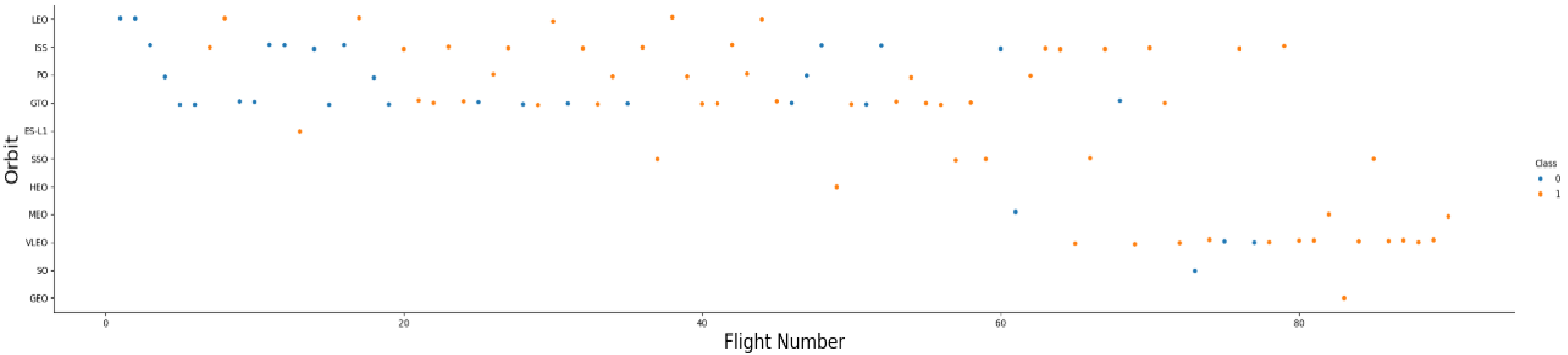
Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type
- Show the screenshot of the scatter plot with explanations
  - Orbit SO has not yet have any success launch



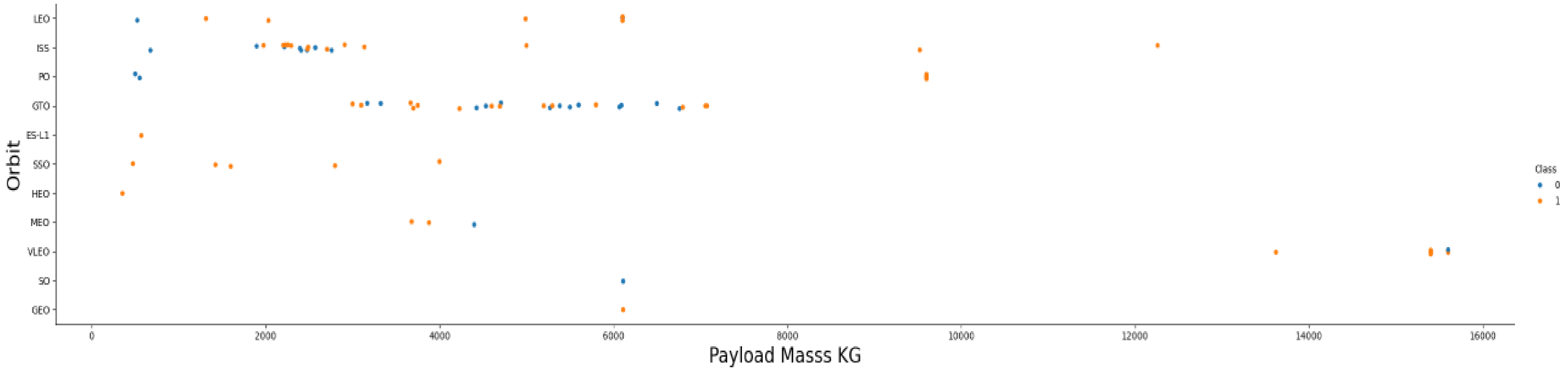
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations
  - Not seeing any correlation between Flight Number & Orbit



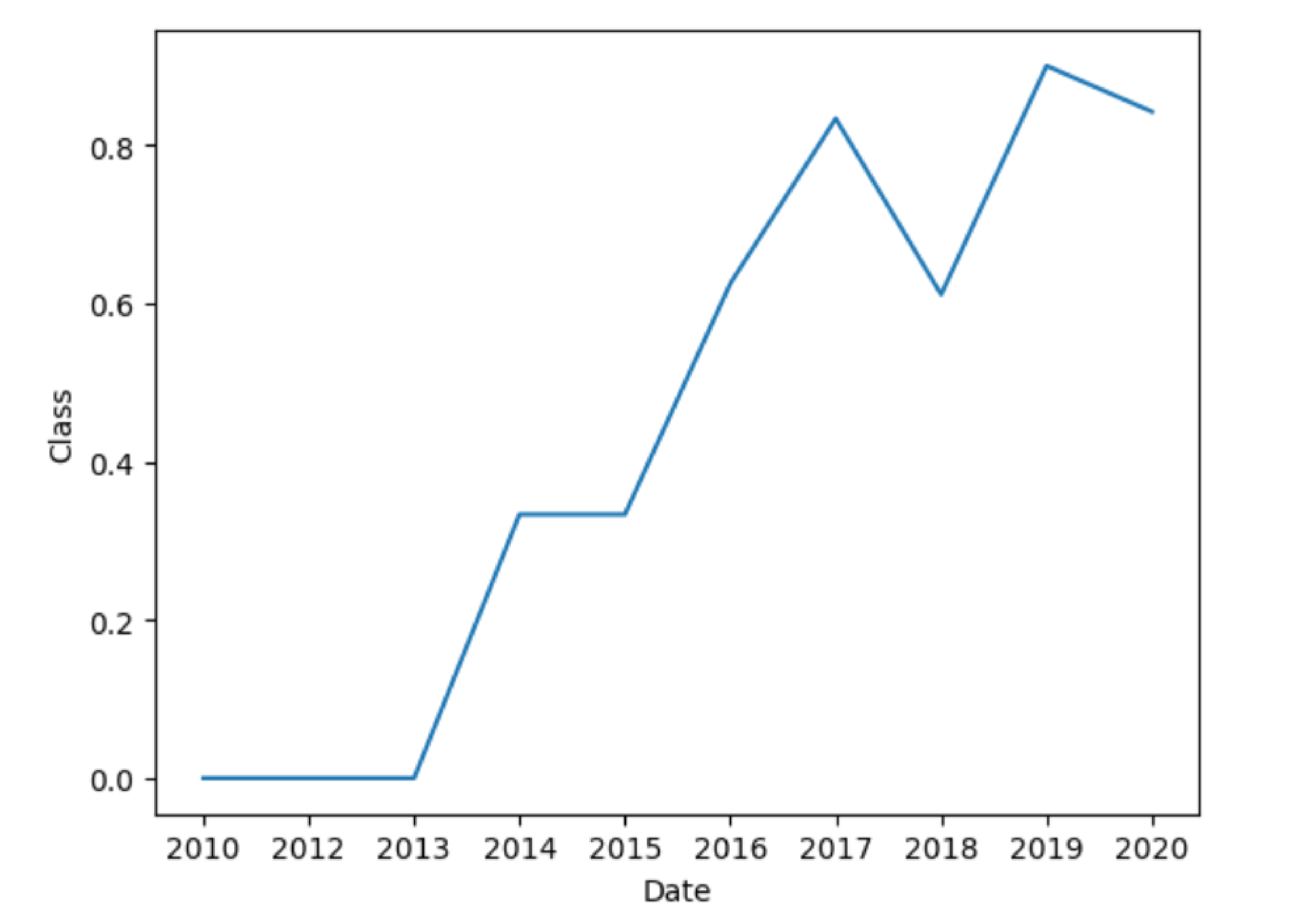
Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type
- Show the screenshot of the scatter plot with explanations
    - Most of orbit types always launch Payload < 10000 kg, while VLEO launches > 10000 kg



Launch Success Yearly Trend

- Show a line chart of yearly average success rate
- Show the screenshot of the scatter plot with explanations
  - Success rate has significantly improved since 2013



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here
  - We have 4 unique Launch Site

```
In [13]: %sql select distinct "Launch_Site" from SPACEXTABLE

 * sqlite:///my_data1.db
Done.
```

Out[13]:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here
  - 5 records of launch site begin with CCA

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

```
 * sqlite:///my_data1.db
Done.
```

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
- Present your query result with a short explanation here
  - Total payload carried by boosters from NASA is 48213 kg

```
%%sql
    select sum(cast("PAYLOAD_MASS__KG_" as int)) total_payload_mass from SPACEXTABLE
    where Customer like "%NASA (CRS)%"
```

```
 * sqlite:///my_data1.db
Done.
```

total_payload_mass
48213

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here
  - Average payload mass carried by booster version F9 v1.1 is 2543 kg

Display average payload mass carried by booster version F9 v1.1

```
%%sql
    select AVG(cast("PAYLOAD_MASS__KG_" as int)) avg_payload_mass from SPACEXTABLE
    where "Booster_Version" like "%F9 v1.1%"
```

```
 * sqlite:///my_data1.db
Done.
```

avg_payload_mass
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here
    - 2025-12-22 is the first successful landing outcome on ground pad

```
%%sql
    select MIN(Date      ) from SPACEXTABLE
    where "Landing_Outcome" = "Success (ground pad)"
```

```
* sqlite:///my_data1.db
Done.
```

MIN(Date)
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
- Present your query result with a short explanation here
  - F9 BT is the boosters that successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

```
%%sql
    select distinct "Booster_Version"
    from SPACEXTABLE
    where cast("PAYLOAD_MASS__KG_" as int) > 4000 and cast("PAYLOAD_MASS__KG_" as int) < 6000
    and "Landing_Outcome" = "Success (drone ship)"
```

```
 * sqlite:///my_data1.db
Done.
```

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
- Present your query result with a short explanation here
    - 1 Failure & 99 Success

```
%%sql
    select "Mission_Outcome", count(*) nb
    from SPACEXTABLE
    group by 1
```

```
* sqlite:///my_data1.db
Done.
```

Mission_Outcome	nb
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
- Present your query result with a short explanation here
    - Booster Version F9 B5

```
%%sql
    select distinct "Booster_Version"
    from SPACEXTABLE
    where cast("PAYLOAD_MASS__KG_" as int) = (select max(cast("PAYLOAD_MASS__KG_" as int)) from SPACEXTABLE )
```

```
 * sqlite:///my_data1.db
Done.
```

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

- Present your query result with a short explanation here

```
%%sql
    select
        substr(Date,0,5) date_year,
         substr(Date, 6,2) date_month
    , count(*) nb
    from SPACEXTABLE
    where "Landing_Outcome" = "Failure (drone ship)"
        and substr(Date,0,5)='2015'
    group by 1,2
```

```
 * sqlite:///my_data1.db
Done.
```

date_year	date_month	nb
2015	01	1
2015	04	1

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

- Present your query result with a short explanation here
  - No Attempt has highest records in Landing Outcome

```
%%sql
    select
        "Landing_Outcome"
    , count(*) nb
    from SPACEXTABLE
    where Date between '2010-06-04' and '2017-03-20'
    group by 1
    order by count(*) desc
```

```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	nb
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

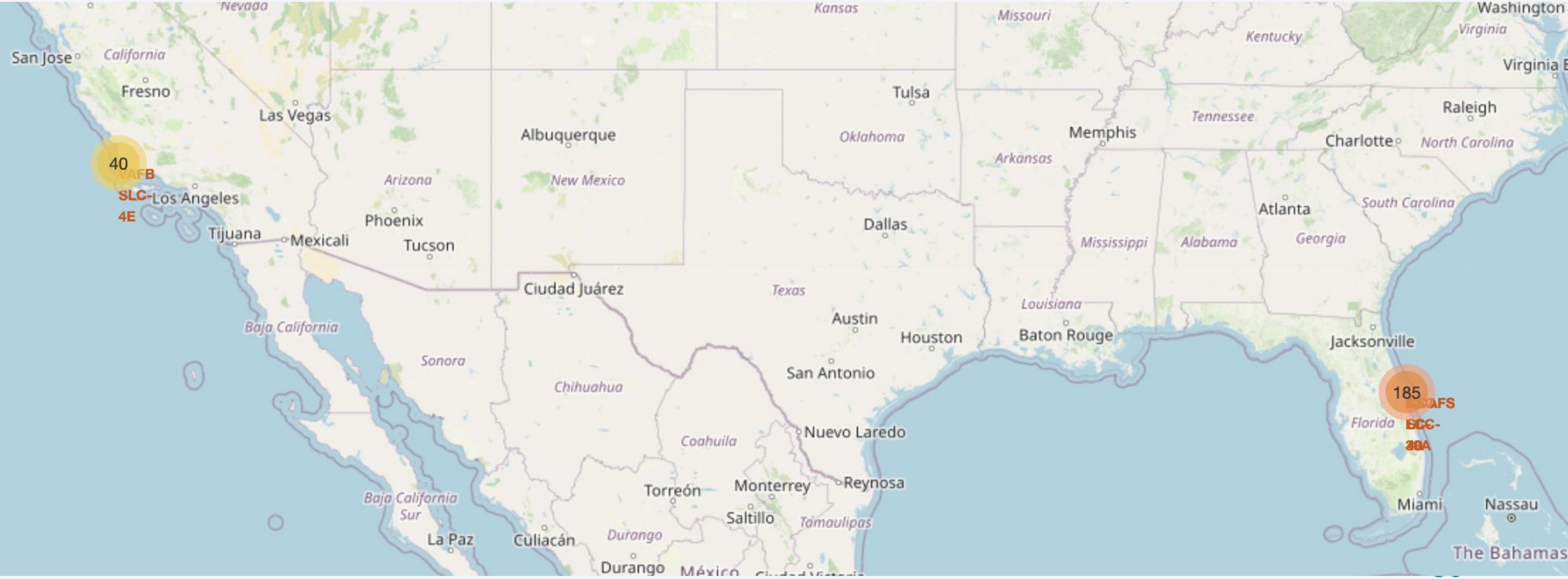
Section 3

Launch Sites Proximities Analysis

Launch Site location on Map

3 launch sites are on the West side & 1 on the East, all are near coastal line

Colored label launch outcome



<Folium Map Screenshot 3>

- Replace <Folium map screenshot 3> title with an appropriate title

- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

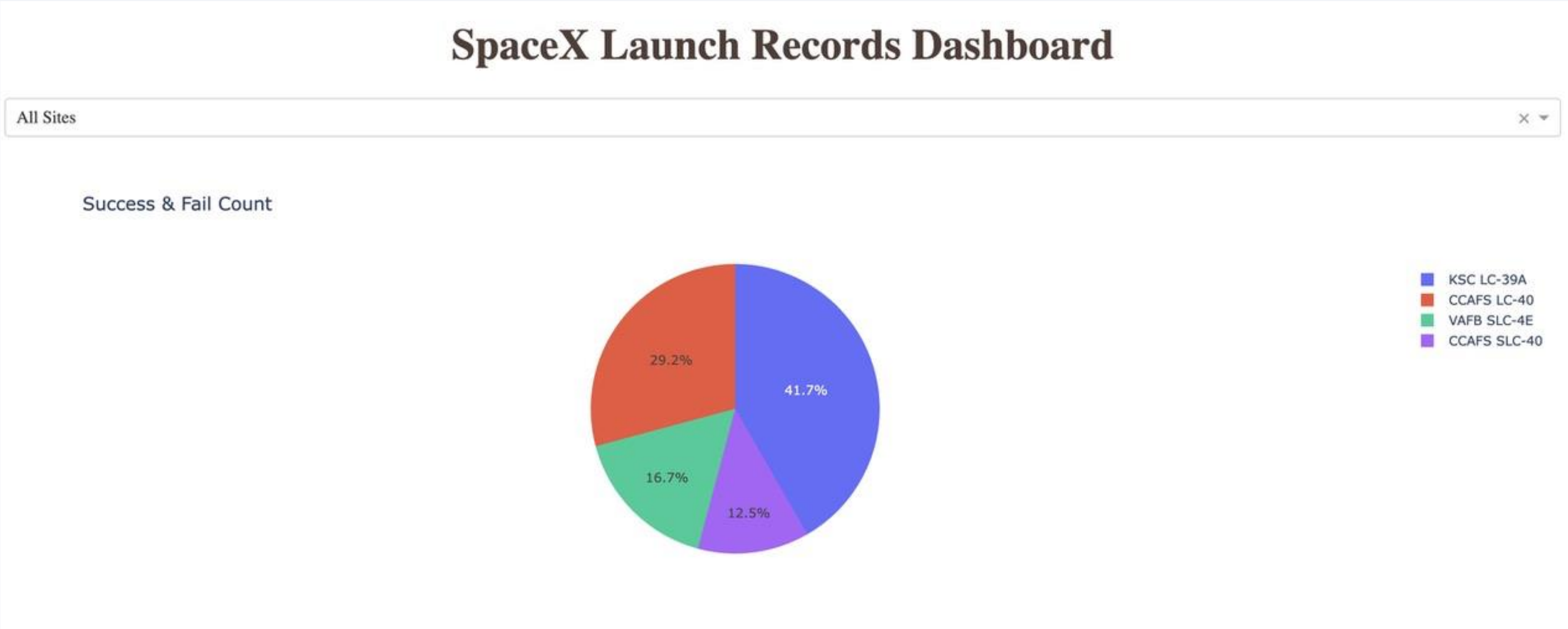
- Explain the important elements and findings on the screenshot

Section 4

Build a Dashboard with Plotly Dash

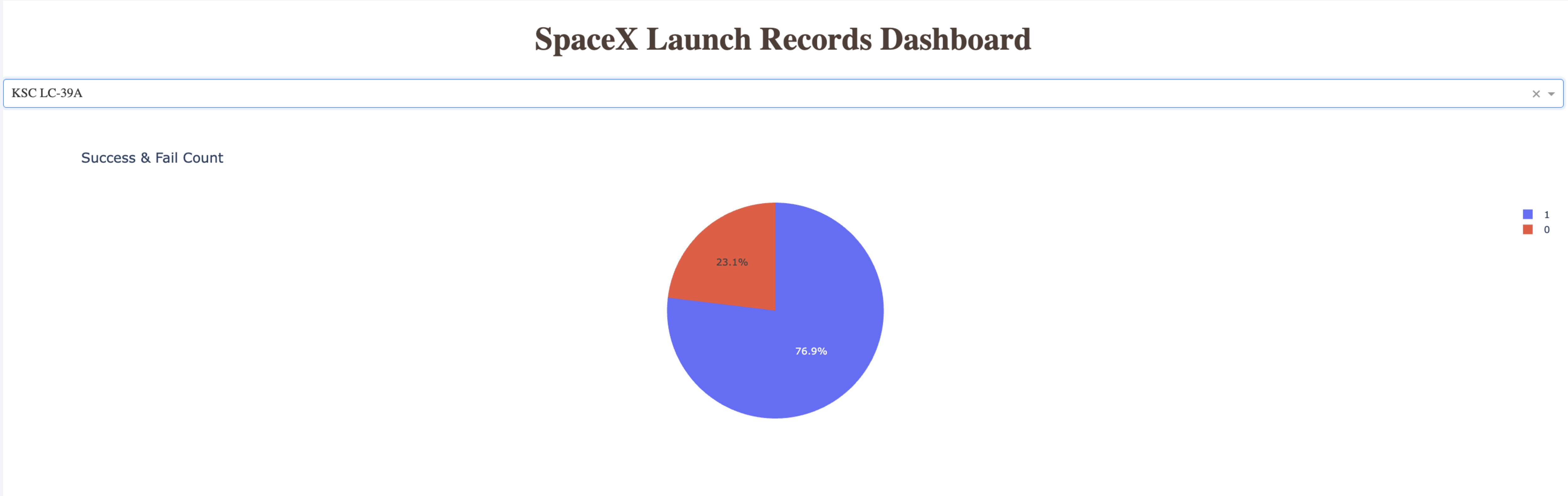
SpaceX Launch Record Dashboard

- Show the screenshot of launch success count for all sites, in a piechart
- Explain the important elements and findings on the screenshot
  - KSC-LC-39A contributes the most launches, with 41.7% total Success & Fail count



Launch Site with highest Success Rate

- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot
  - KCS LC-39A is the launch site with highest success launch rate, 76.9%



Payload Range & Booster Version success rate

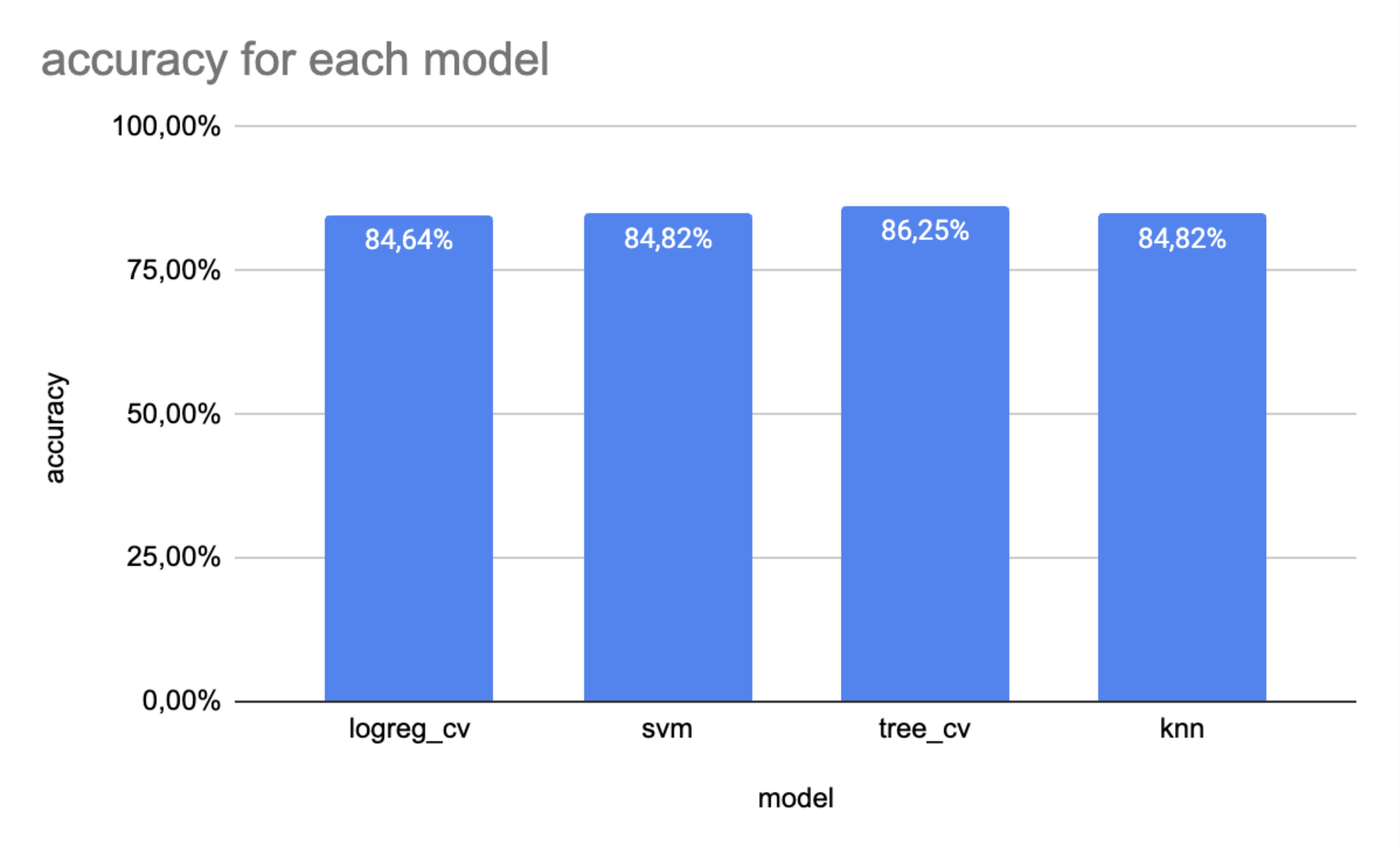
- Payload < 6000 kg seems to have higher success rate

Section 5

Predictive Analysis (Classification)

Classification Accuracy

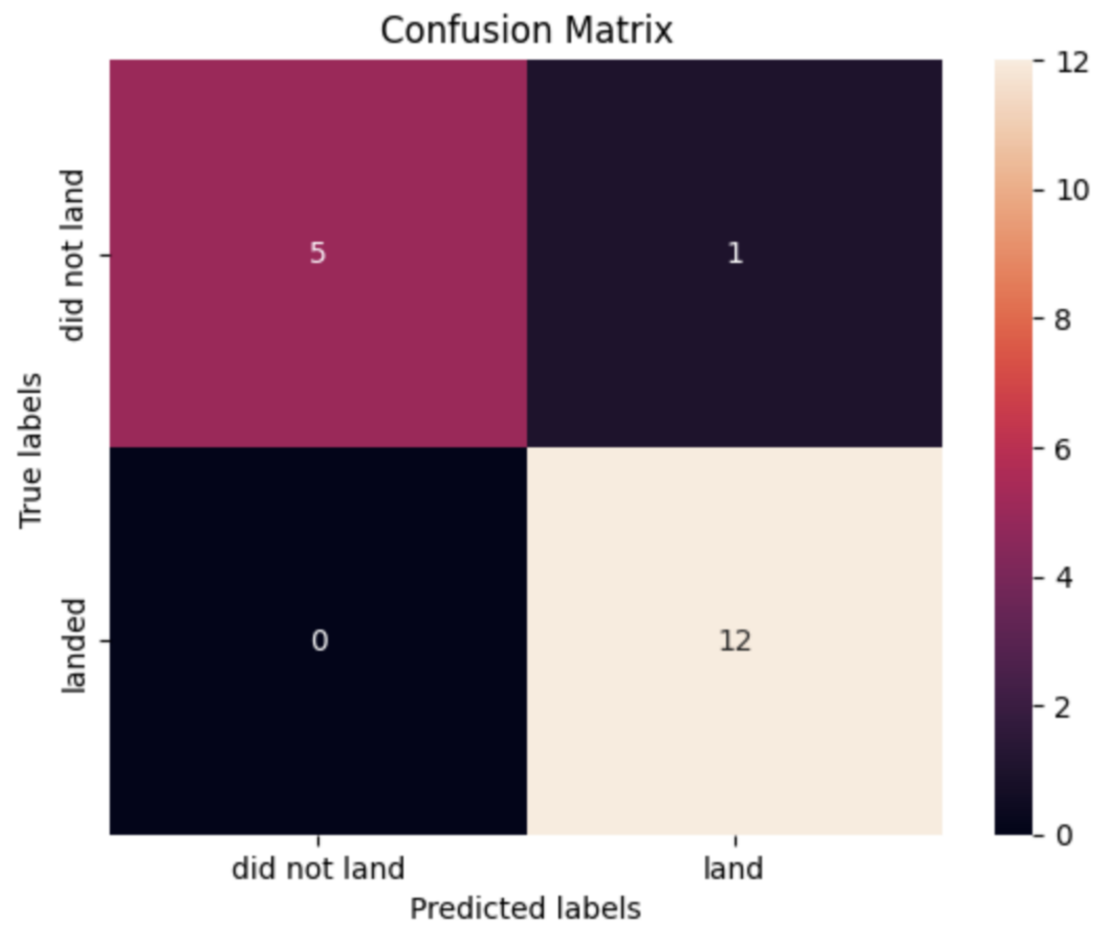
- Visualize the built model accuracy for all built classification models, in a bar chart
- Find which model has the highest classification accuracy
  - Decision Tree is the model with highest accuracy



Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation
  - True positive –12 (True label is landed, Predicted label is also landed)
  - False positive – 1 (True label is did not land, Predicted Landed)

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



Conclusions

- Using FlightNumber, PayloadMass, Orbit Type, Launch Site as feature engineering
- Testing through different models, we see that Decision Tree gives the best results (accuracy 86.25%)

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
  - GitHub URL for all assignments: https://github.com/thaonguyen2601/coursera-final-assignment

Thank you!