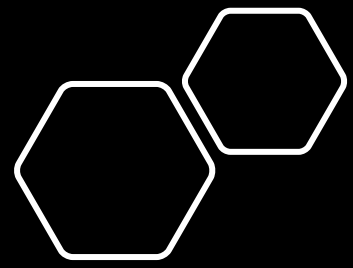


Data Science Capstone Project

Soham Rane

January 24, 2024

Outline

Executive Summary

Introduction

Methodology

Results

Conclusion

Appendix

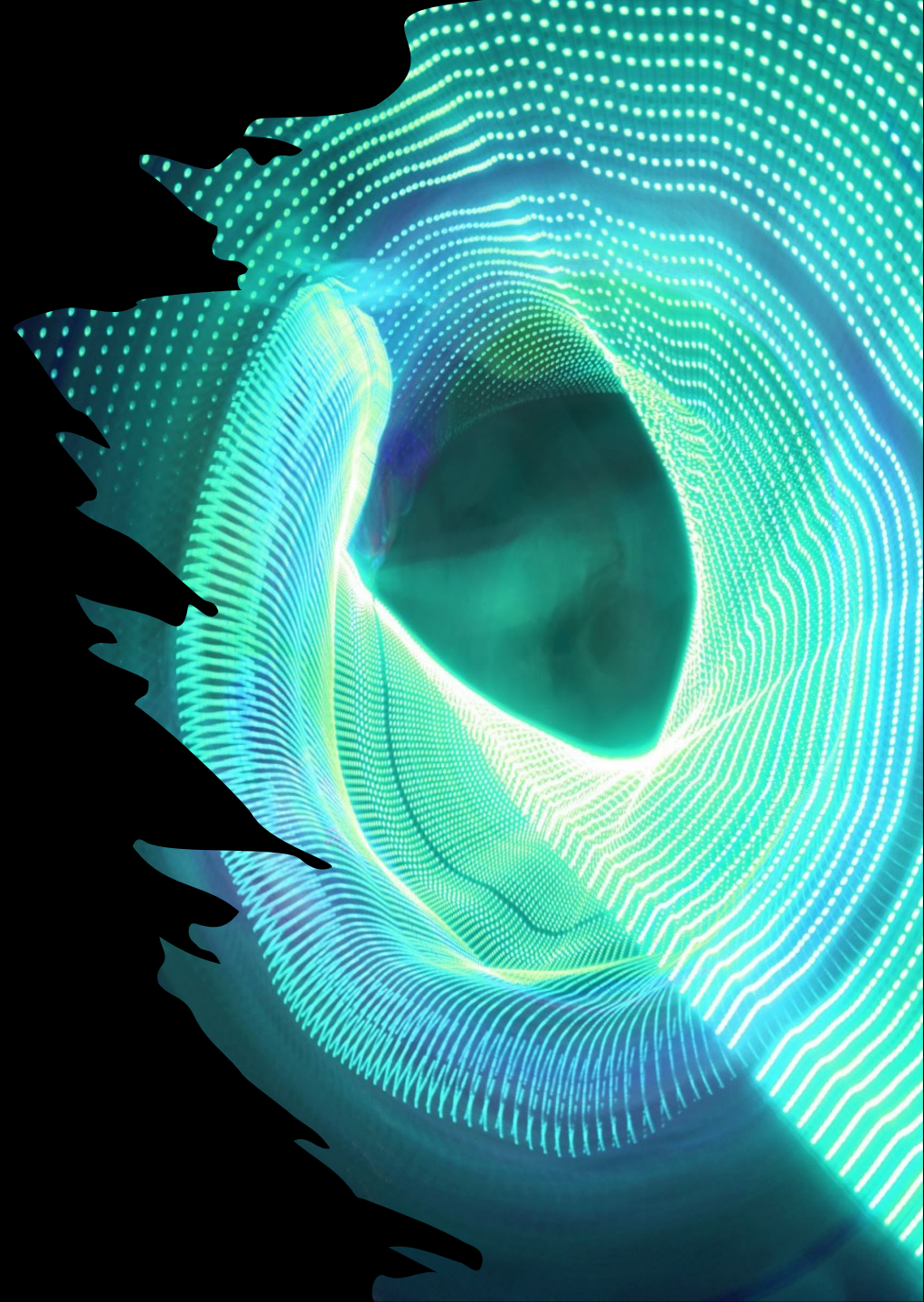
Executive Summary

- Data Collection
  - Use SpaceX Rest API
  - Web Scrapping from Wikipedia
- Data Wrangling
  - Filtering the data
  - Dealing with Null values
  - Using One Hot encoding to prepare the data
- Exploratory Data Analysis
  - EDA with SQL
  - EDA with Python Data Visualization
  - Interactive Visual Analytics and Dashboard
- Predictive analysis using Classification
  - Classification with multiple machine learning models
  - Models: Logistic Regression, KNN, SVM, Decision Tree

Introduction

- Project background and context’
  - Space X’s advertises its game-changing Falcon 9 rocket launces on its website with a cost of 62 million dollars, while other providers charge north of 165 million dollars each. SpaceX is able to advertise these launches at a fraction of the price as other competitors because the first stage is reusable. . Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. Spaces X’s Falcon 9 launch like regular rockets. In this project, we will take the role of a data scientist working for a new rocket company called Space Y that would like to compete with SpaceX founded by Billionaire industrialist Allon Musk. The final task is to determine the price of each launch and if SpaceX will reuse the first stage.

- Problems you want to find answers
  - What is the price of each launch of the Flacon 9?
  - What is the success rate of the Flacon 9 models for each of their sites?
  - What are the factors that affect the success rate of a successful landing?



Methodology

Methodology

Executive Summary

Data collection

Perform data wrangling

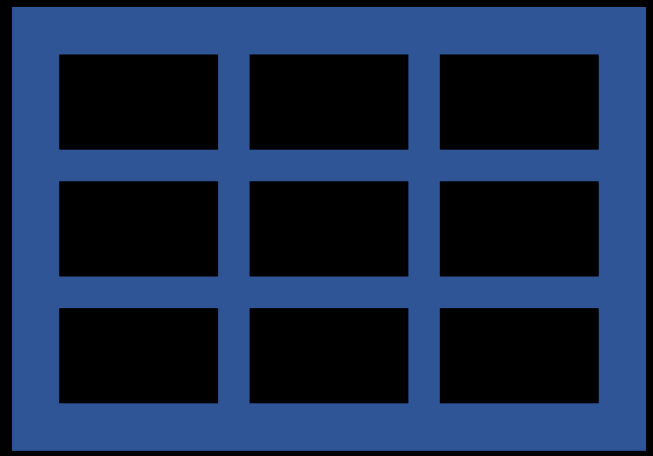
Exploratory Data Analysis (EDA) with Python Visualization & SQL

Perform interactive visual analytics using Folium and Plotly Dash

Perform predictive analysis using classification models

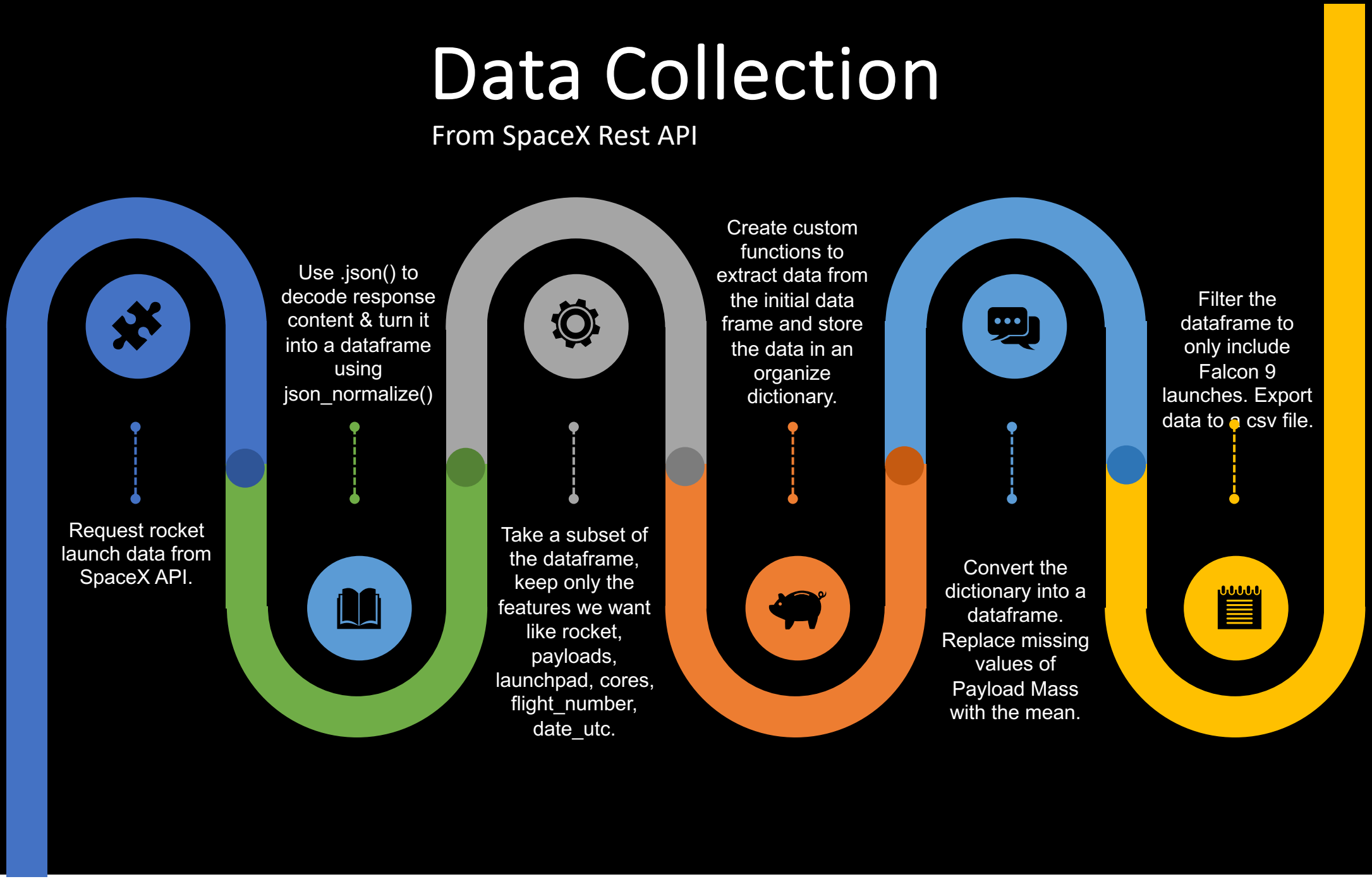
Data Collection

- Data collection from the SpaceX API using the requests library.
    - The response returned by the API is in the form of JSON objects so we use the json_normalize() function to convert the JSON data into a pandas data frame.
- Web scraping Flacon 9 Lunch Records using BeatufulSoup library
    - In order to extract the launch records which were stored in the HTML table, parsing through the HTML was necessary.
    - While parsing the important information that was needed was added to predefined lists.
    - Once finished through parsing all the tables, the lists were added as columns a pandas data frame.
- Dataframe was filtered to contain only Falcon 9 launches
- Missing values in Payload Mass were replaced by the mean of the Payload Mass column



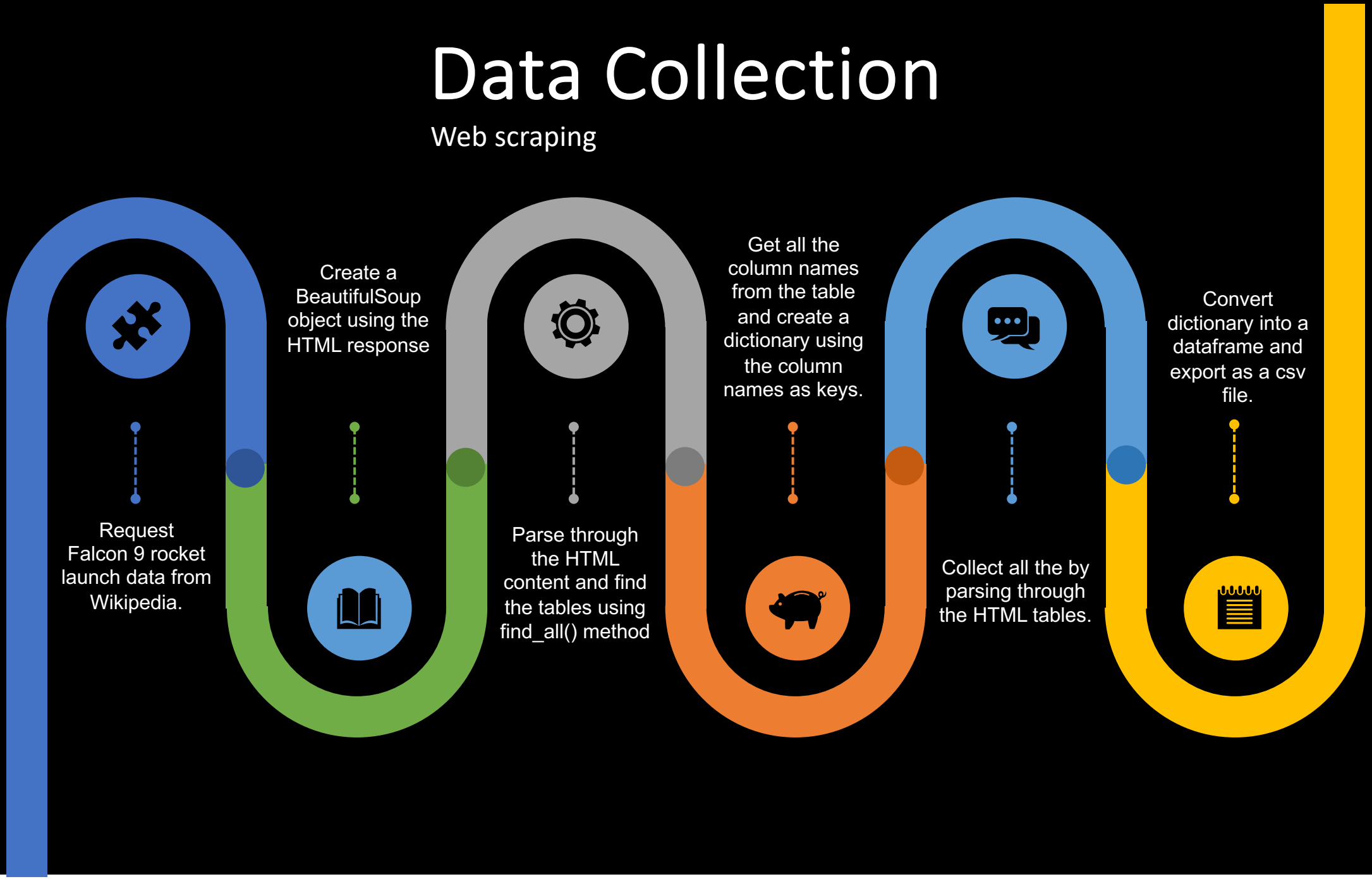
Data Collection

From SpaceX Rest API



Data Collection

Web scraping



Data Wrangling

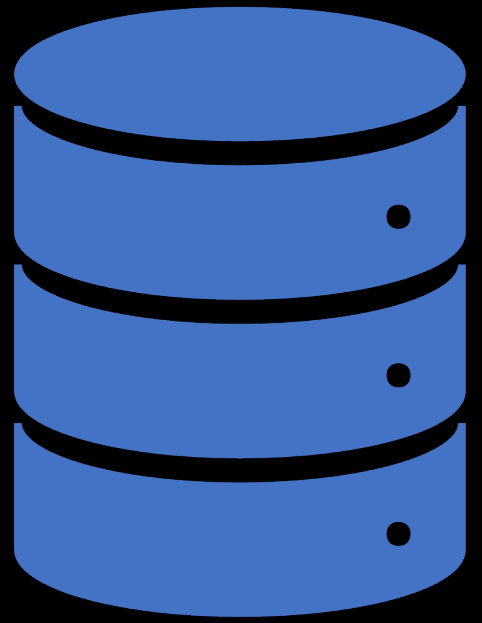
- Identify and calculate the % of missing values in each attribute.
- Identify which columns are numerical and categorical.
- Calculate:
  - # of launches of each site
  - # of occurrence of each orbit
  - # and occurrence of missing outcome of the orbits
- Landing Outcomes:
  - **True Ocean** means the mission outcome was successfully landed to a specific region of the ocean.
  - **False Ocean** means the mission outcome was unsuccessfully landed to a specific region of the ocean.
  - **True RTLS** means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad.
  - **True ASDS** means the mission outcome was successfully landed to a drone ship False ASDS means the mission outcome was unsuccessfully landed to a drone ship.
  - **None ASDS** and **None None** represent a failure to land.
- Create a binary landing outcome label based on the Outcome column to classify if landing was successful or unsuccessful
  - 1 if successful
  - 0 if unsuccessful
- Export data to a csv

EDA with Data Visualization

- Charts Plotted:
  - Flight Number vs. Payload Mass (scatter plot)
  - Flight Number vs. Launch Site (scatter plot)
  - Payload Mass vs Launch Site (scatter plot)
  - Orbit Type vs. Success Rate (bar chart)
  - Flight Number vs. Orbit Type (scatter plot)
  - Payload Mass vs Orbit Type (scatter plot)
  - Success Rate Yearly Trend (line chart)
- Scatter plots helps observe relationships between variables
- Bar charts is used to compare discrete categories and their values
- Line chart helps to visualize trends over time (time series)

EDA with SQL

- Find:
  - Unique launch sites
  - 5 records where launch sites being with ‘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 succesful landing outcome in ground pad was acheived.
  - 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. Use a subquery
  - 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.



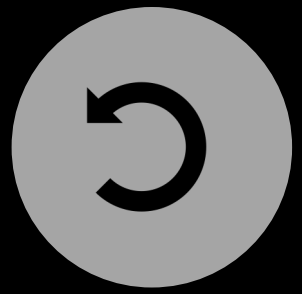
Build an Interactive Map with Folium

- Markers on all the Launch sites:
    - Blue Circle with popup label - NASA Johnson Space Center’s
    - Red Circles with popup label – All the launch sites in the database
    - Green marker – Successful launches
    - Red markers – Unsuccessful launches

- Added colored lines(distance) between CCAFS SLC-40 Launch Site to closest city, railway, highway, and coastline.

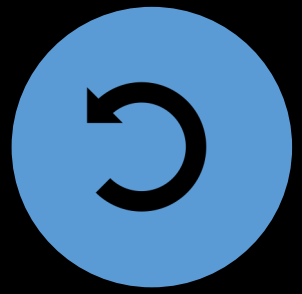
Build a Dashboard with Plotly Dash

Launch Site Drop-down Input Component



Callback function to render success-pie-chart based on selected site dropdown

Range Slider to Select Payload



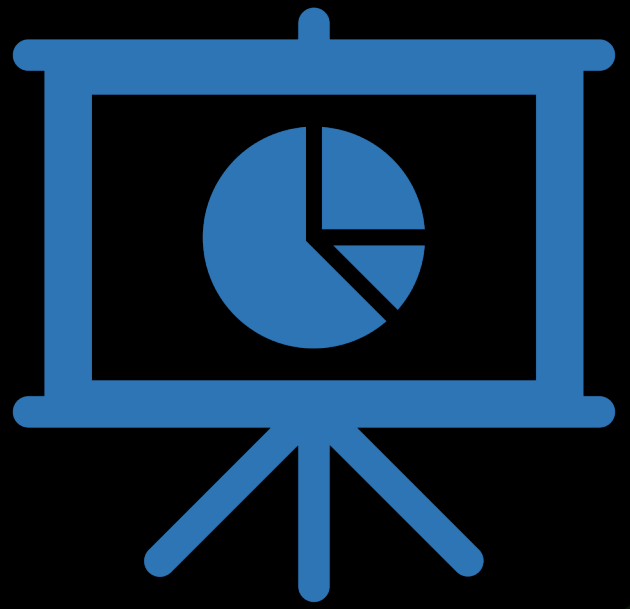
Callback function to render the success-payload-scatter-chart scatter plot

Predictive Analysis (Classification)

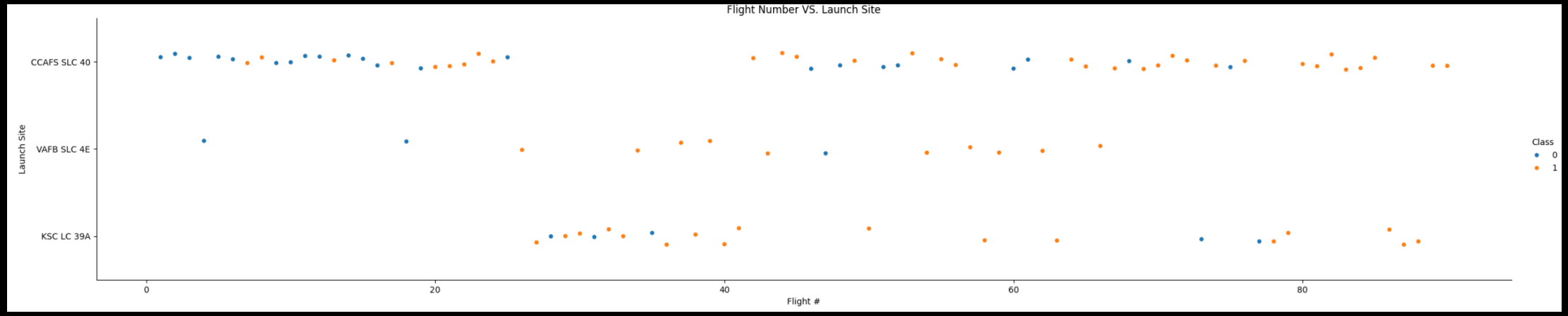
- Loaded the sklearn, pandas, numpy, and seaborn libraries
- Used the StandardScaler() from sklearn.preprocessing to standardize the feature variables.
- Converted the response variable (successful landing 1:T/ 0:F) to a numpy array
- Split dataset into 80% train set and 20% as the test set
- Used the following Machine Learning models:
  - Logistic Regression
  - SVM
  - Decision Tree
  - K nearest neighbours
- Used GridSearchCV to choose the best hyperparameters
- Found a model that performs the best

Results

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

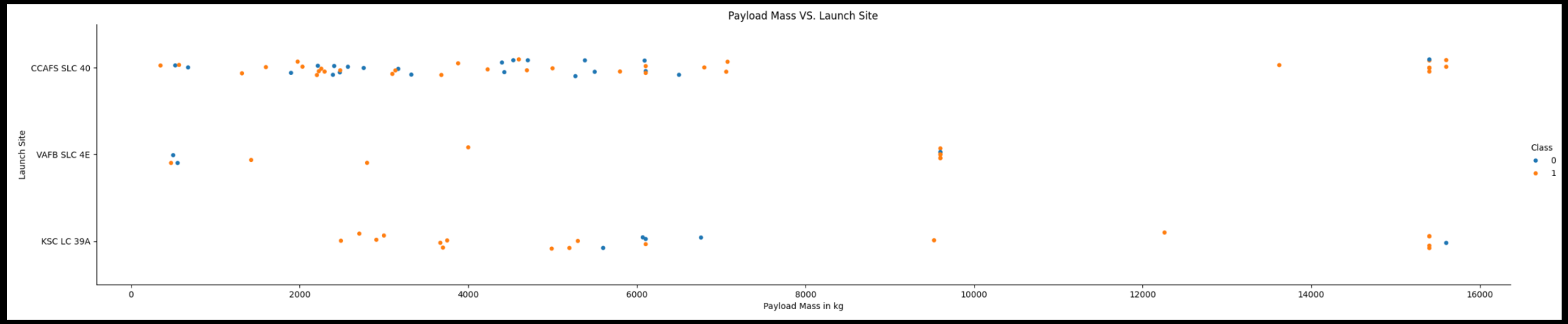


Insights drawn from EDA



Flight Number vs. Launch Site

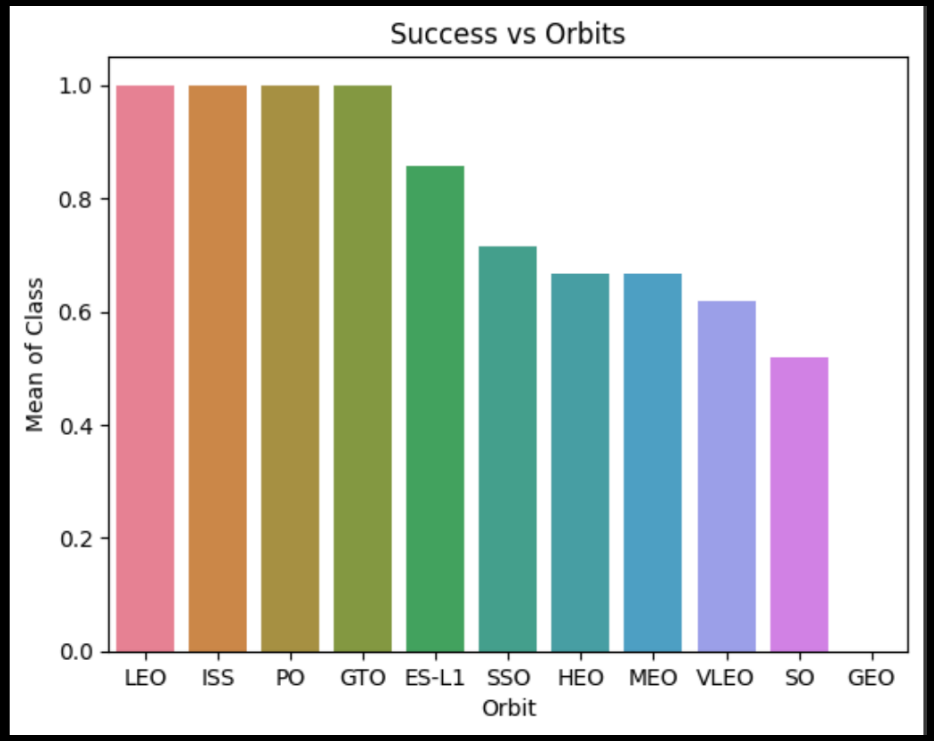
- Earlier flights(0 - 25) had considerably lower success rate
- Later flights(25 - 100) had a higher success rate
- CCAFS SLC-40 had ~50% success rate
- VAFB SLC 4E had ~77% success rate
- KSC LC 39A had ~77% success rate
- Therefore we can deduce that SpaceX learnt from its earlier launches as the later flights have a much higher success rate



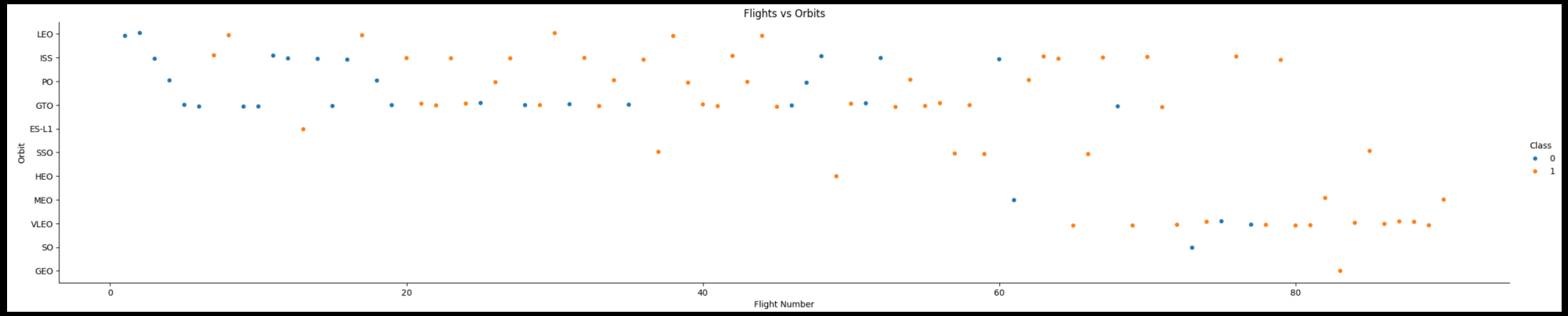
Payload vs. Launch Site

- Bigger the payload mass, the better the success rate
- At CCAFS SLC 40 there haven’t been launches of payload between 8,000kg to 13,000kg
- At VAFB SLC 4E there haven’t been launches of payload more then 10,000kg
- Most of the launches at the VAFB SLC 4E were successful except for the ones with very little payload
- Launches at KSC LC 39A were mostly successful except for a few around the 6,000kg payload mark

Success Rate vs. Orbit Type

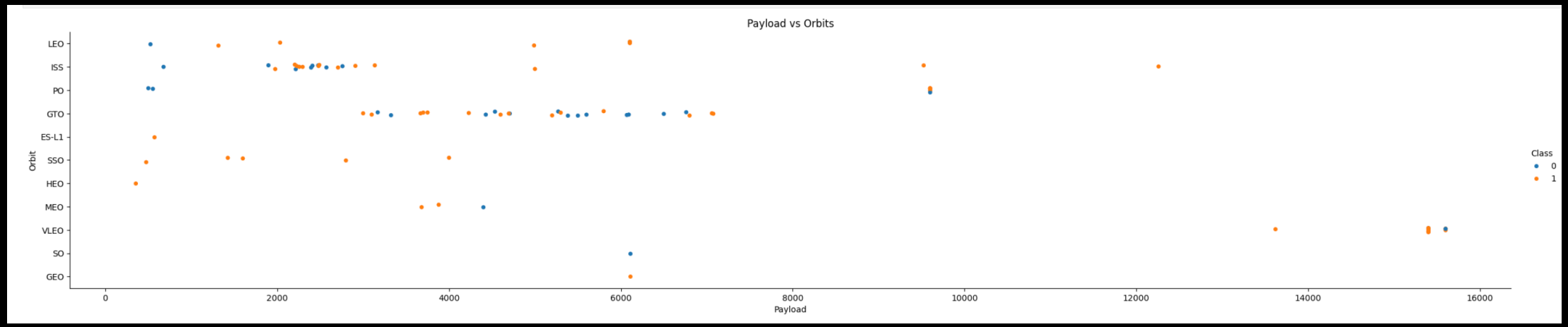


- 100% Success Rate – LEO, ISS, PO, GTO
- 50-90% Success Rate – ES-L1, SSO, HEO, MEO, VLEO, SO
- 0% Success Rate - GEO



Flight Number vs. Orbit Type

- As more flights are taken in the LEO, ISS, PO & VLEO orbit the success rate increases.
- Orbit GTO does not clearly indicate that there is a positive correlation between the success and # of flights.

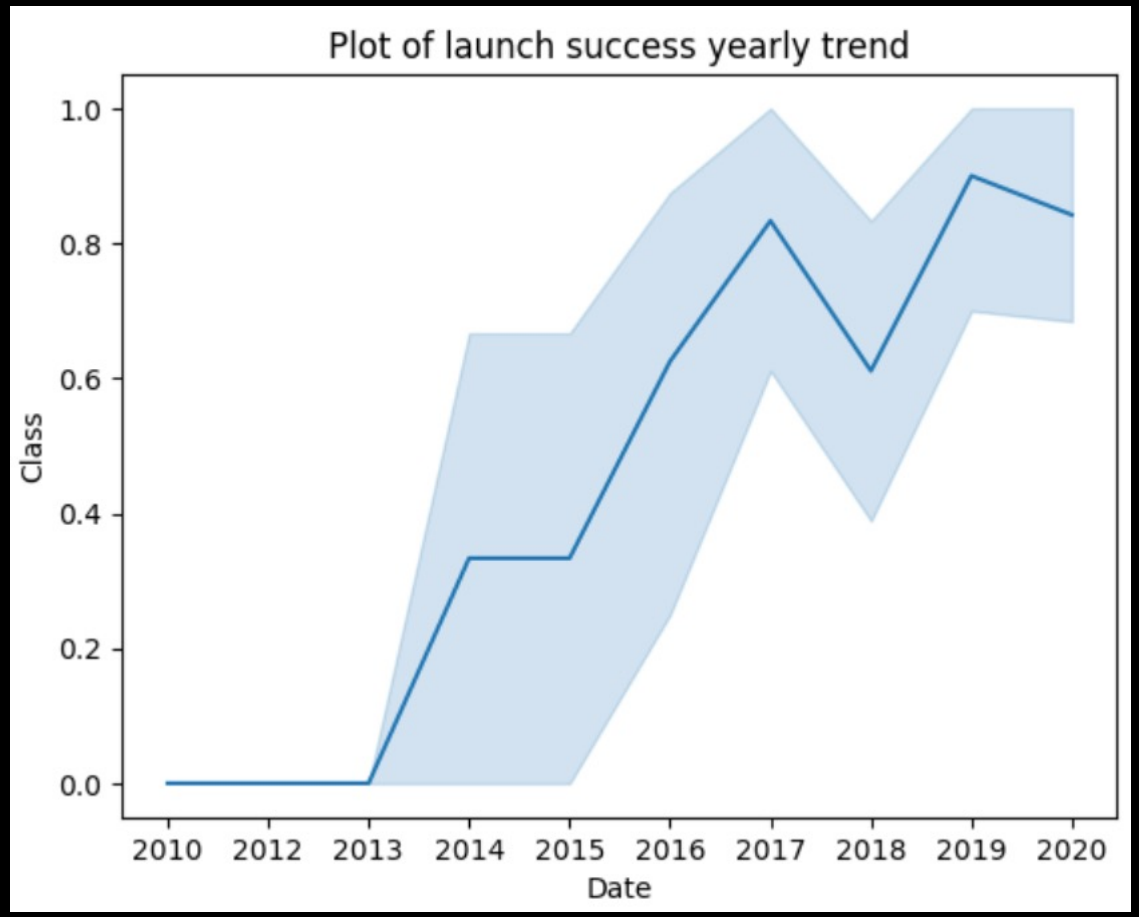


Payload vs. Orbit Type

- Falcon 9's with smaller payloads in SSO, HEO, MEO orbits overall mostly were successful.
- Falcon 9's with payloads of <2000 in orbit LEO, ISS, PO are mostly unsuccessful.

Launch Success Yearly Trend

- From years 2013 – 2020 Falcon9's success rate has increased
- From years 2010 – 2013 Faldon9's success has been minimally close to 0%



All Launch Site Names

- There are 4 Launch Sites:
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40
- SQL query:
  - SELECT DISTINCT Launch_Site FROM SPACEXTBL
  - DISTINCT only chooses the unique values from a column in a table

```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

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

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

```
%sql SELECT * FROM SPACEXTBL 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

Launch Site Names Begin with 'CCA'

- SQL Query:
  - SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE “%CCA%” LIMIT 5
  - The query shows 5 records whose Launch_Site begins with “CCA”

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS Total_Payload_MASS FROM SPACEXTBL WHERE Customer = "NASA (CRS)"
 * sqlite:///my_data1.db
Done.
```

Total_Payload_MASS
45596

Total Payload Mass

- Shows the total payload mass in kg of all the Falcon 9s in the table which was purchased by NASA (CRS)

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS "F9v1.1_AVG_PAYLOAD_MASS" FROM SPACEXTBL WHERE Booster_Version = "F9 v1.1"
 * sqlite:///my_data1.db
Done.
```

F9v1.1_AVG_PAYLOAD_MASS
2928.4

Average Payload Mass by F9 v1.1

- Use the WHERE clause to filter the data with the exact name of the Booster Version F9 v1.1.
- Average Payload Mass for Booster Version F9 v1.1 is 2928.4kg

```
%sql SELECT MIN(Date) as "first succesful landing" from SPACEXTBL WHERE Landing_Outcome = "Success (ground pad)"
 * sqlite:///my_data1.db
Done.
```

first succesful landing
2015-12-22

First Successful Ground Landing Date

- The first successful date was on December 22, 2015.
- Query:
  - Filtered the data by searching for only successful landings
  - Then used MIN(Date) to retrieve the smallest(earliest) date value

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = "Success (drone ship)" AND PAYLOAD_MASS__KG_ between 4000 AND 6000
```

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

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

Successful Drone Ship Landing with Payload between 4000 and 6000

- Boosters Versions:
  - F9 FT B1022
  - F9 FT B1026
  - F9 FT B1021.2
  - F9 FT B1031.2
- Query:
  - Filtered Booster versions WHERE Landing_Outcome as “Success (drone ship)” and payload mass is between 4000 and 6000

```
%sql SELECT Mission_Outcome ,COUNT(*) FROM SPACEXTBL WHERE Mission_Outcome LIKE "%S%" GROUP BY Mission_Outcome LIKE "%S%"
```

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

Mission_Outcome	COUNT(*)
Success	100

```
%sql SELECT Mission_Outcome ,COUNT(*) FROM SPACEXTBL WHERE Mission_Outcome LIKE "%F%" GROUP BY Mission_Outcome LIKE "%F%"
```

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

Mission_Outcome	COUNT(*)
Failure (in flight)	1

Total Number of Successful and Failure Mission Outcomes

- Used % wildcard to filter Mission Outcomes that have an S in them to final the total # of successful outcomes
  - Grouped the data by values that have an S in them
- Query was the same for Failure Outcomes except the S was replaced with F

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
 * 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

Boosters Carried Maximum Payload

- Used subquery in this SQL query
- Used “WHERE” to filter the data by Payload Mass
- Subquery: (SELECT MAX(PAYLOAD_MASS_KG) FROM SPACEXTBL) retrieves the maximum payload from the table
- Then we check if Payload Mass is equal to maximum payload mass to retrieve the Boosters that carried the maximum payload

```
%%sql SELECT substr(Date, 6,2) AS MONTH, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL
WHERE Date between"2015-01-01" AND "2015-12-31"  AND Landing_Outcome = "Failure (drone ship)"
```

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

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

2015 Launch Records

- Query:
  - Used substr(Date, 6,2) to only retrieve the months of the date
  - Used WHERE to retrieve data only for the year of 2015
  - Also retrieved rocket data that failed to land on drone ships

```
%sql SELECT Landing_Outcome, COUNT(*) FROM SPACEXTBL WHERE DATE(Date) between "2010-06-04" and "2017-03-20"GROUP BY Landing_Outcome ORDER BY COUNT(*) DESC
```

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

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

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

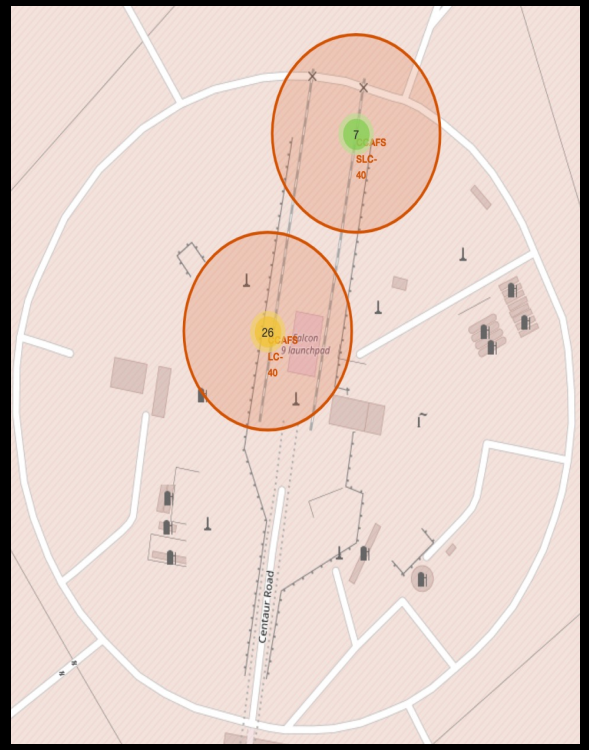
- Query:
  - Used WHERE to retrieve data between 2010-06-04 and 2017-03-20 dates
  - Grouped all the landing outcomes and ordered them descending using COUNT(*)
  - COUNT(*) gives the number of records that are retrieved in a query
  - USE CHAT GPT FOR QUERY EXPLANATIONS

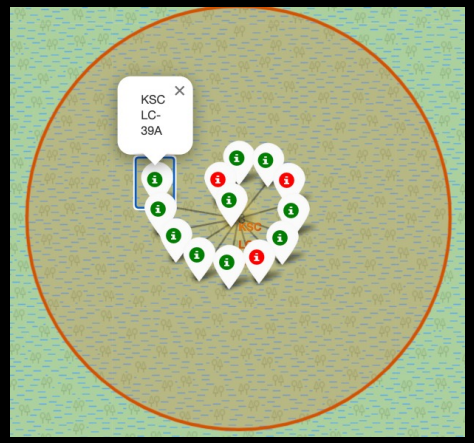
Launch Site Proximities Analysis

SpaceX launch sites

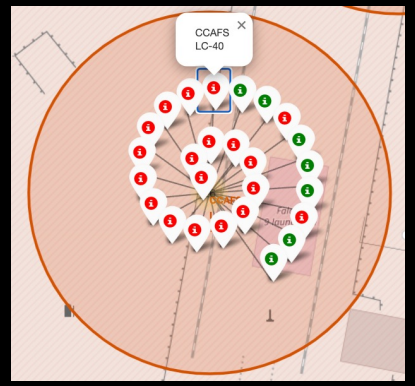
- SpaceX has launch sites in the USA only, in states: California and Florida
- Both of them of are near the ocean.
- 2 Launch Sites in Florida
- 1 Launch Site in California

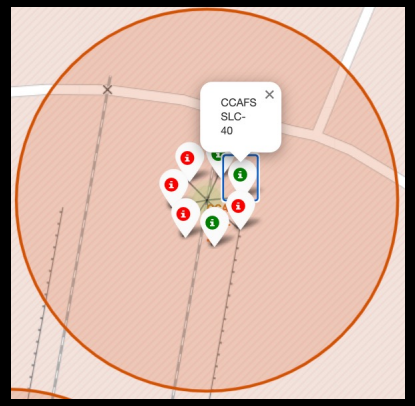
Launch Sites

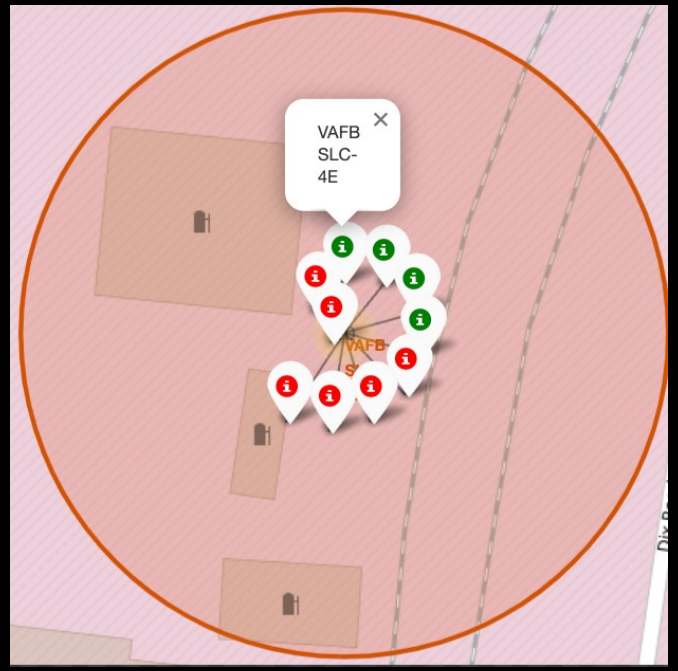




Florida Launch Sites





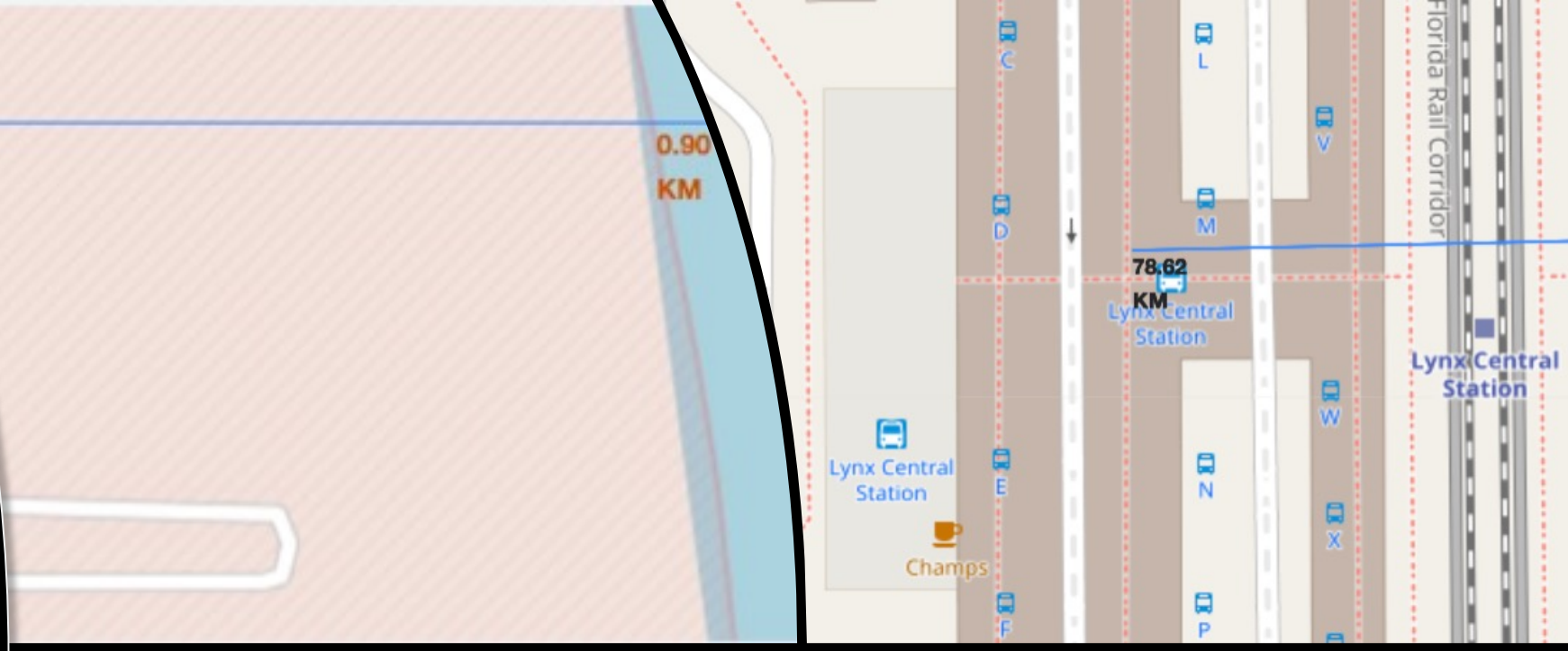


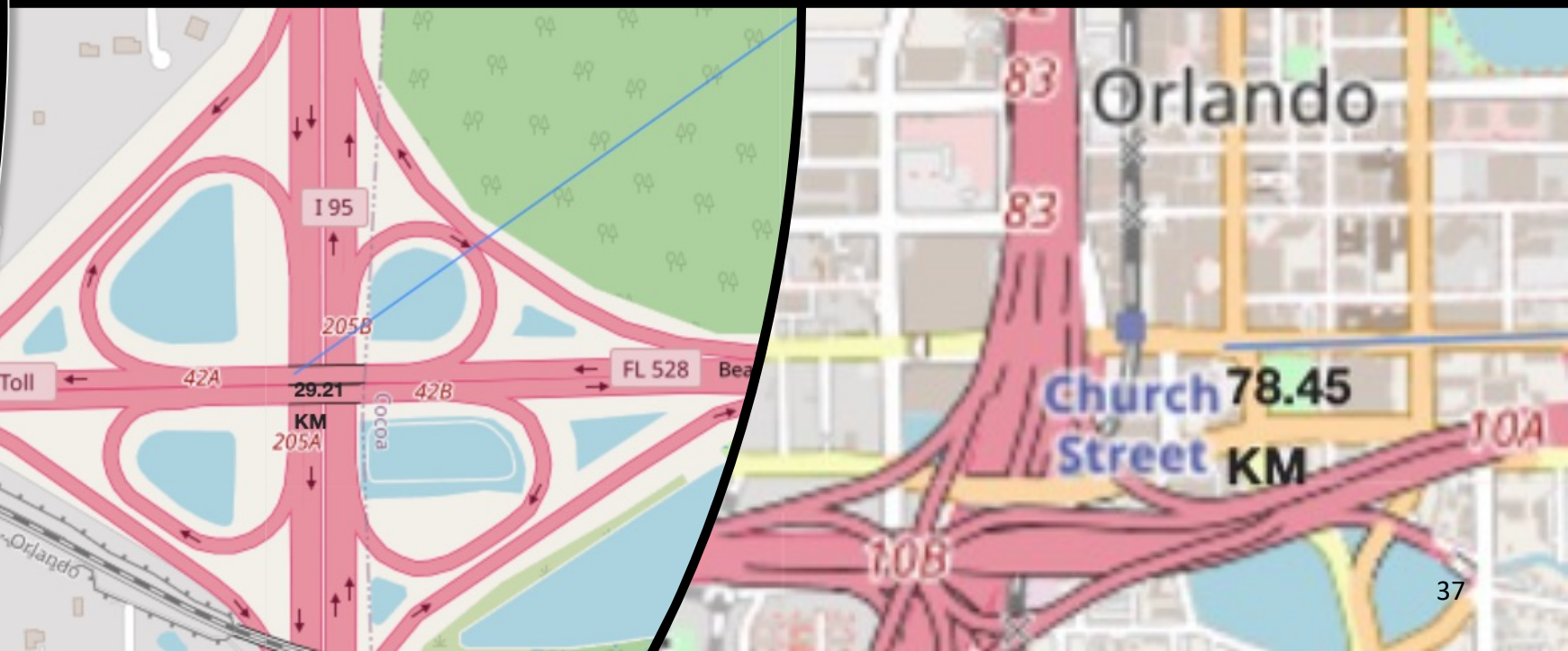
California Launch Sites

Green Markers show successful landings &
Red Markers show failed landings

Distances between a launch site to its proximities

- Proximities close to site:
  - Coastline: Yes
  - Railway: No
  - Highway: No
  - City: No

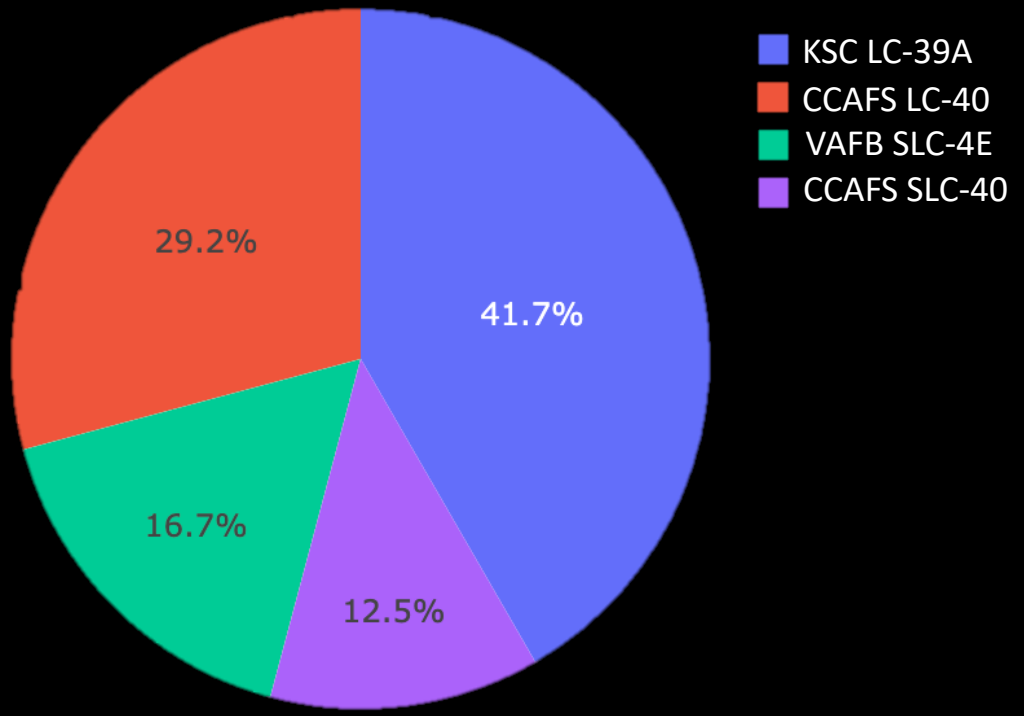




Build a Dashboard with Ploty Dash

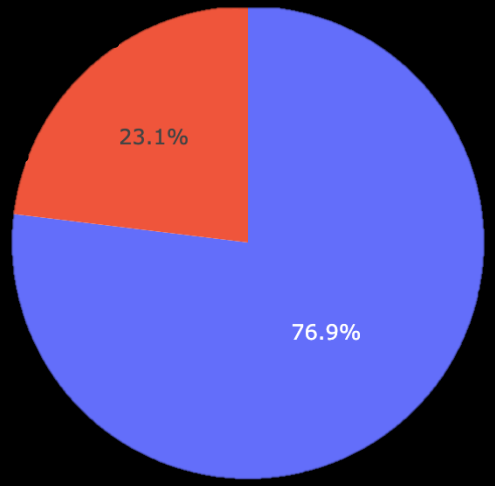
Pie Chart of SpaceX Launch Records

- Launches at KSC LC-39A were the most successful launches compared to all other sites
- Launches at CCAFS LC-40 were the second most successful launches compared to all other sites

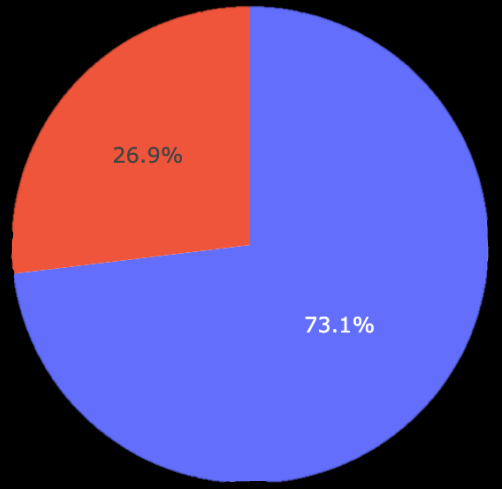


Pie Chart: Success of Individual Launch Sites

KSC LC-39A

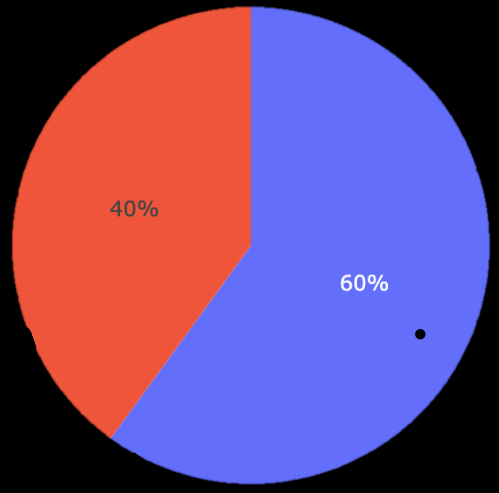


CCAFS LC-40

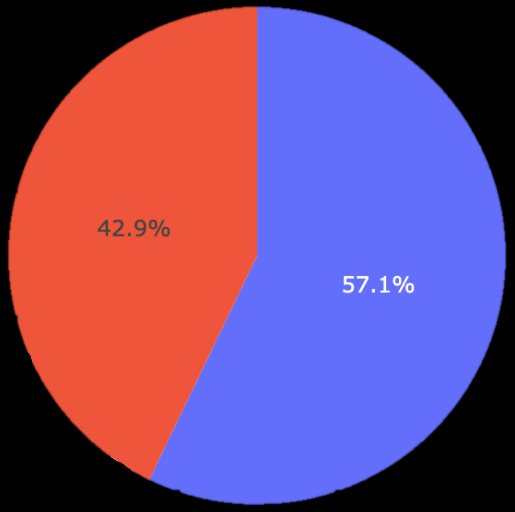


Success
Failure

VAFB SLC-4E

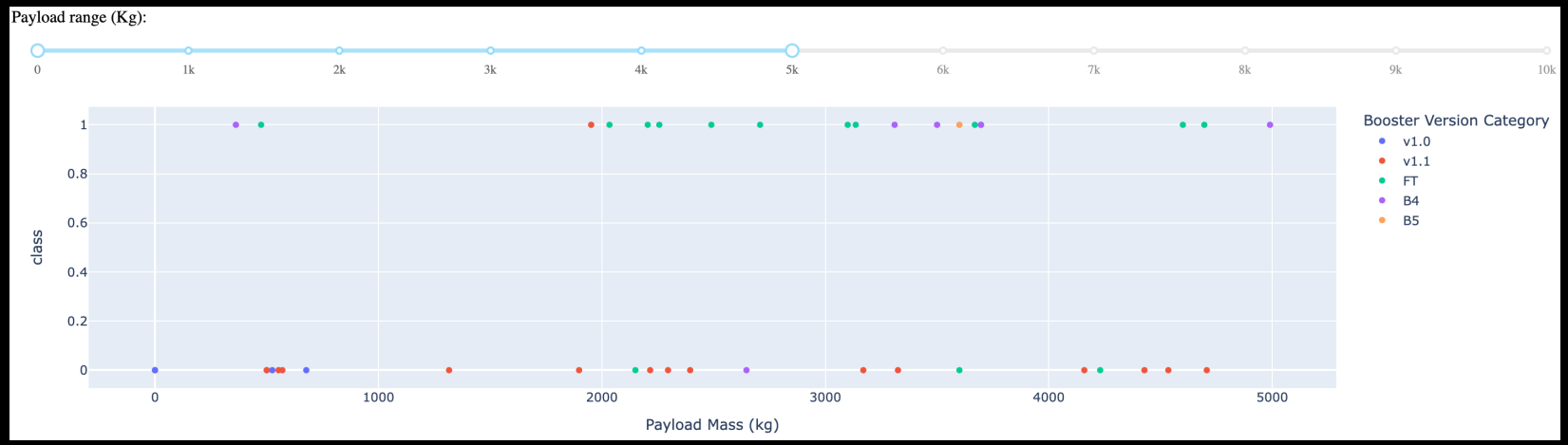


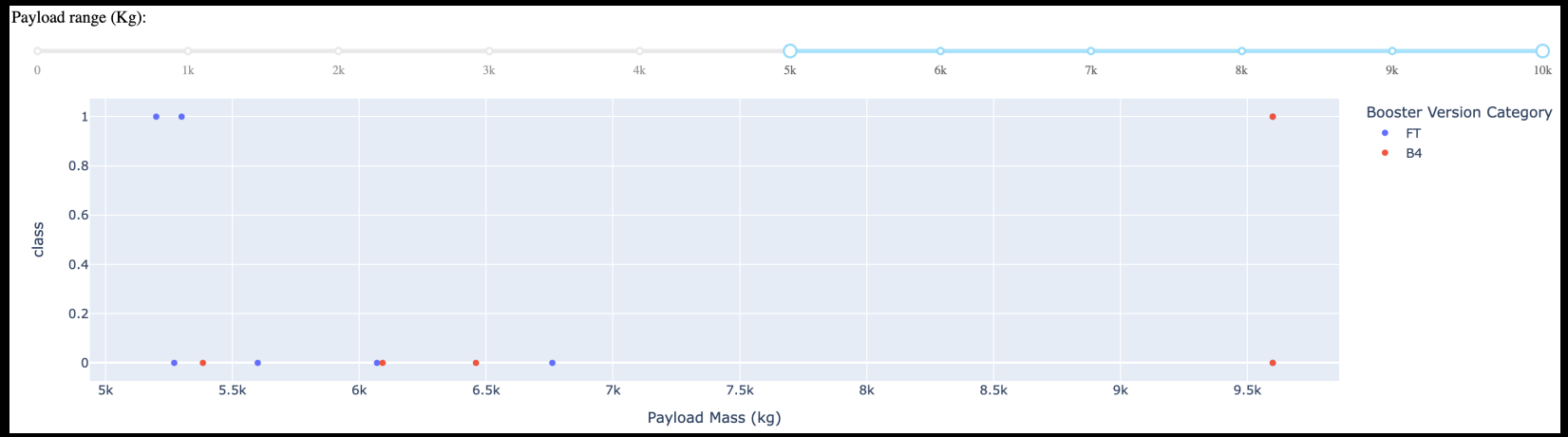
CCAFS SLC-40



Florida Launch Sites

- There have been more Falcon 9 rocket launches with payloads 0kg - 5,000kg than 5,000kg -10,000kg



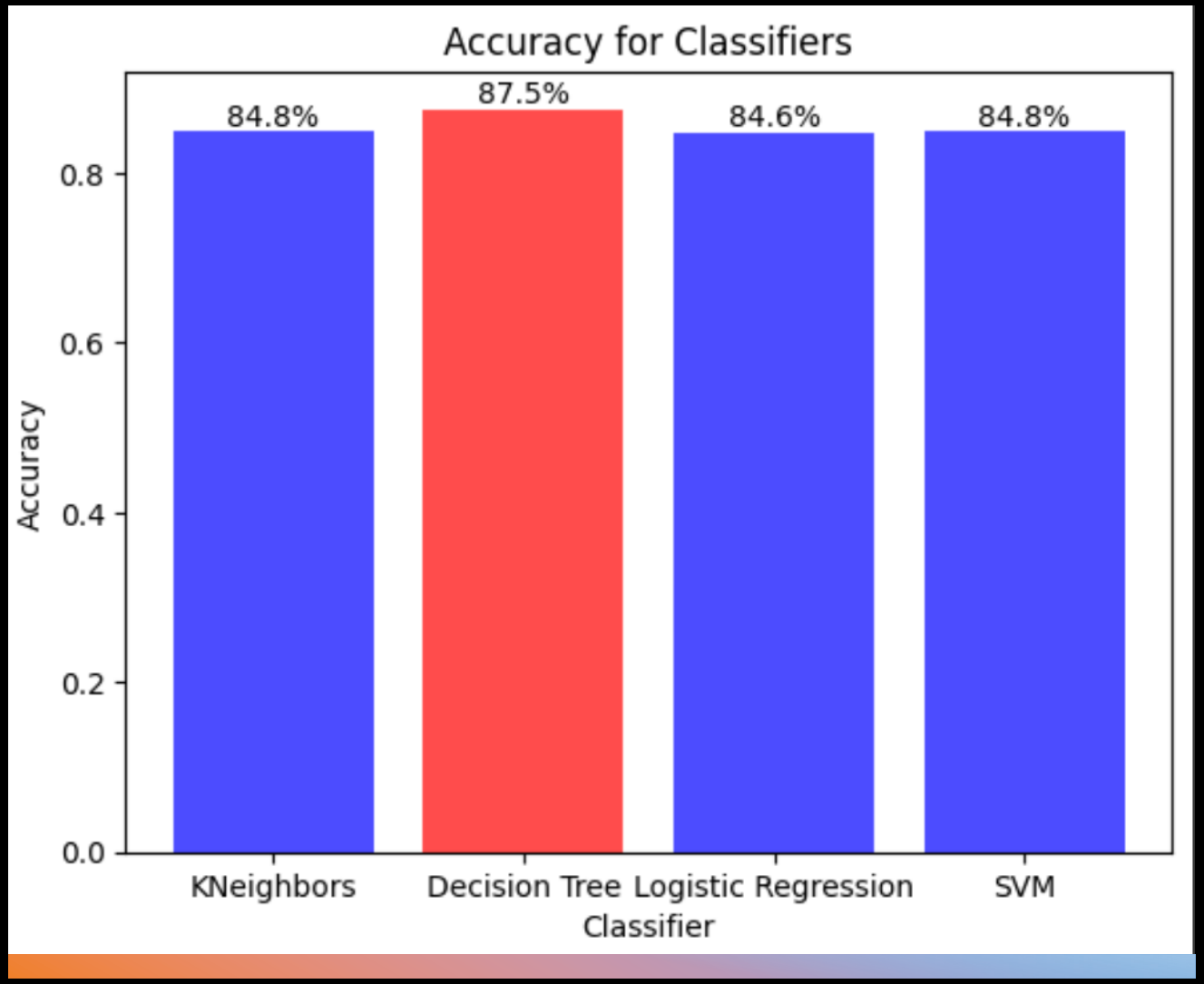


Predictive Analysis (Classification)

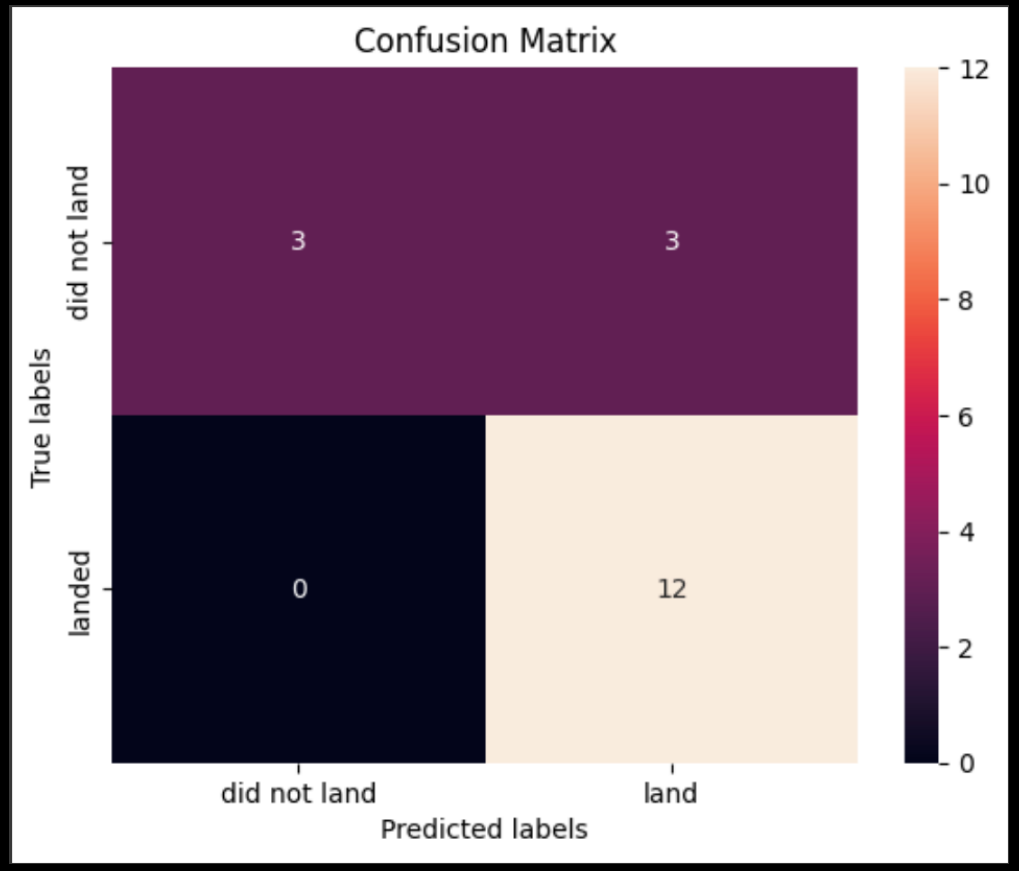
Classification Accuracy

- As we can see all of the classifiers have accuracy between 84% - 88%

- Decision Tree is the most accurate in classifying if the Falcon 9 rocket will successfully land



Confusion Matrix

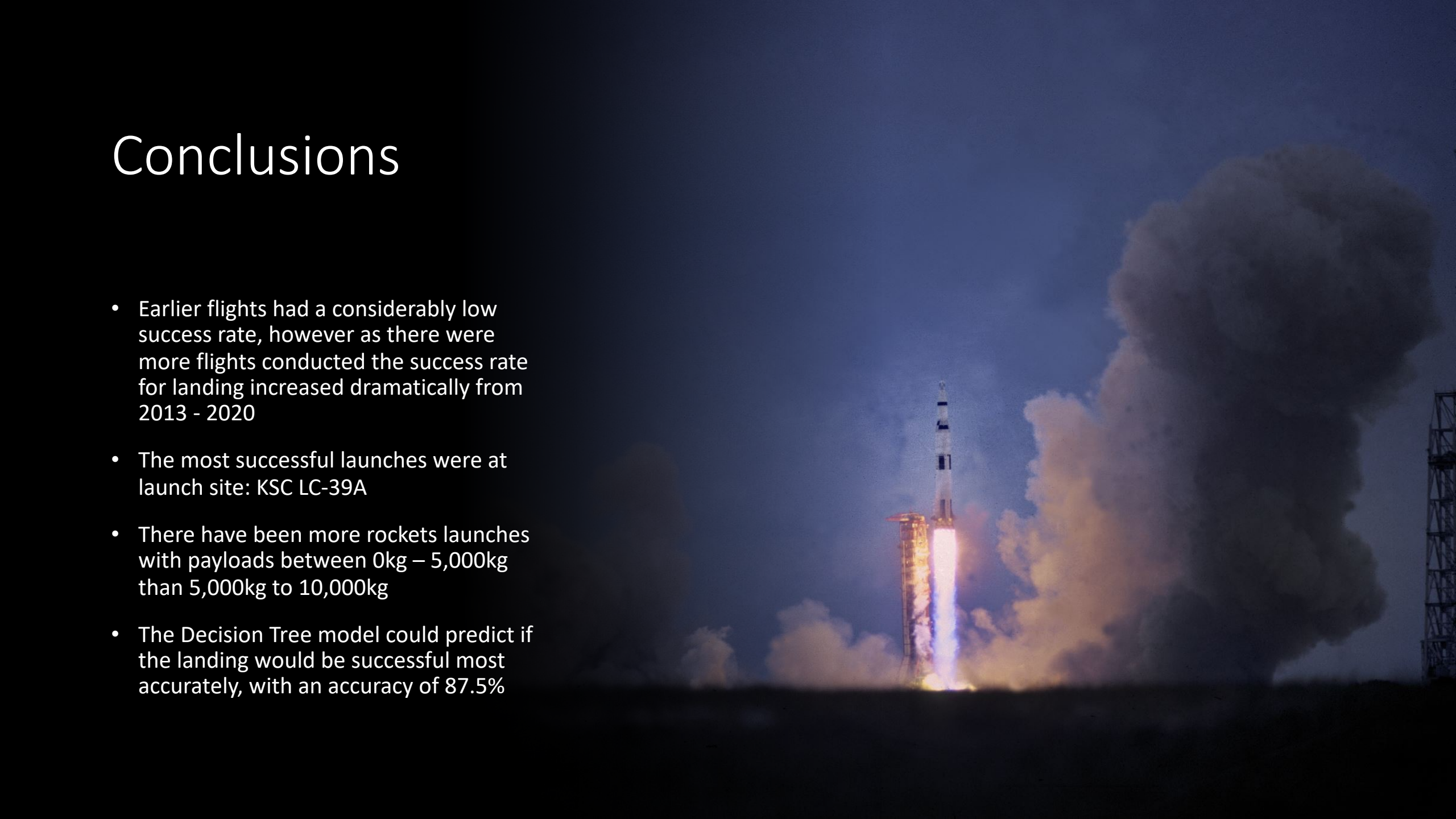


- The confusion matrix shows:
  - True Positives (12):
    - Actual & Predicted Label – “land”
  - True Negatives (3):
    - Actual & Predicted Label – “did not land”
  - False Positives (3):
    - Actual Label: “did not land”
    - Predicted Label: “land”
  - False Negatives (0):
    - Actual Label: “land”
    - Predicted Label: “did not land”

- Inferring from the confusion matrix the model can accurately predict rockets that did not land very well
- The only problem is the False Positives when the model is predicting the rocket has landed successfully however in reality the rocket failed to land

Conclusions

- Earlier flights had a considerably low success rate, however as there were more flights conducted the success rate for landing increased dramatically from 2013 - 2020
- The most successful launches were at launch site: KSC LC-39A
- There have been more rockets launches with payloads between 0kg – 5,000kg than 5,000kg to 10,000kg
- The Decision Tree model could predict if the landing would be successful most accurately, with an accuracy of 87.5%



Appendix

Code:
Data Collection via SpaceX REST API: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/data-collection-api.ipynb
Data Collection with Web scraping: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/webscraping.ipynb
Data Wrangling: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Data%20wrangling.ipynb
EDA with SQL: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/sql-coursera_sqllite.ipynb
EDA with Python Visualization: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/eda_dataviz.ipynb
EDA with Folium: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/dataviz_folium.ipynb
Plotty Dashboard: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/spacex_dash_app.py
Machine Learning Prediction: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Machine_Learning_Prediction.ipynb

Datasets:
Data Wrangling: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Data_Wrangling.csv
EDA with SQL: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/SQL.csv
EDA with Python Visualization: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Python_Data_Viz.csv
EDA with Folium: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Folium.csv
Plotty Dashboard: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Plotty_Dash.csv
Machine Learning Prediction: https://github.com/cosmo71/IBM_DataScience_Capstone/blob/main/Machine_Learning.csv

Thank You!