IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

ARNAB CHATTERJEE

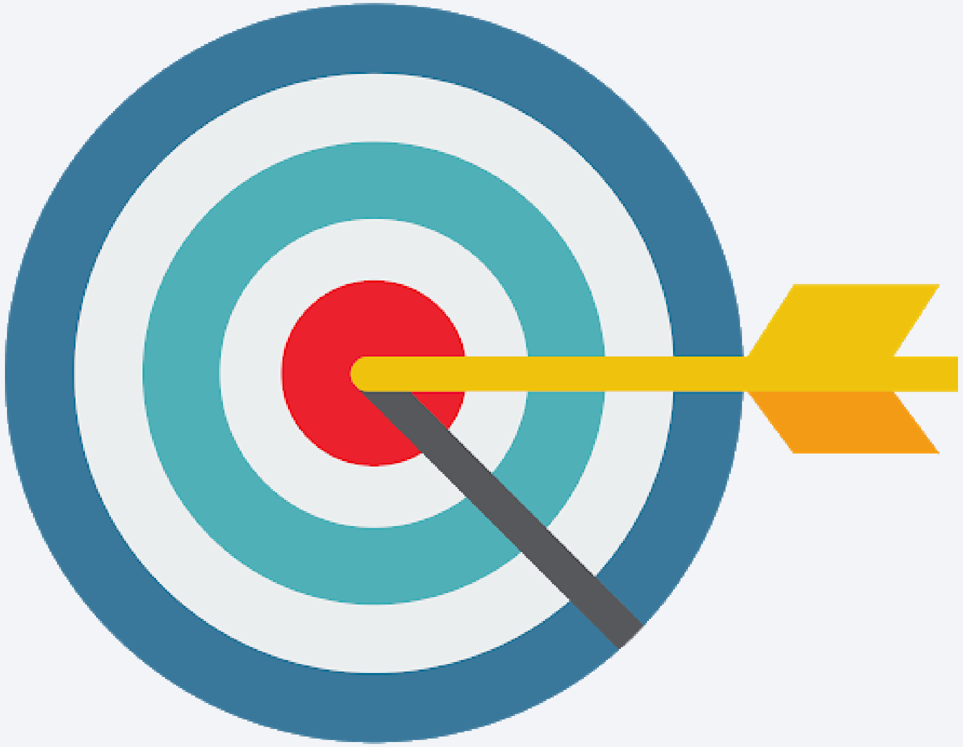
23 February 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - EDA with data visualization
  - EDA with SQL
  - Building an interactive map with Folium
  - Building a Dashboard with Plotly Dash
  - Predictive analysis (Classification)
- Summary of all results
  - Exploratory data analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results



Introduction

- Project background and context

  We predicted if the Falcon 9 first stage will land successfully. SpaceX 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 SpaceX 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 SpaceX for a rocket launch.

- Problems you want to find answers
  - ✓ What influences if the rocket will land successfully?
  - ✓ The effect each relationship with certain rocket variables will impact in determining the success rate of a successful landing.
  - ✓ What conditions does SpaceX have to achieve to get the best results and ensure the best rocket success landing rate.

Section 1

Methodology

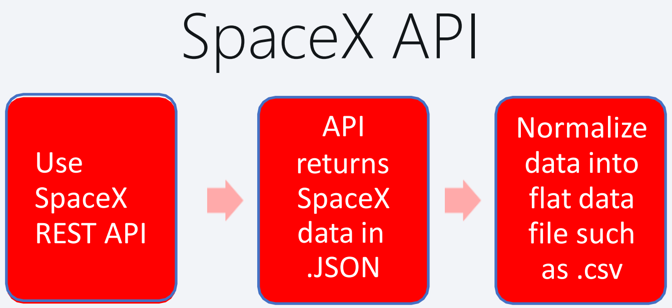
Methodology

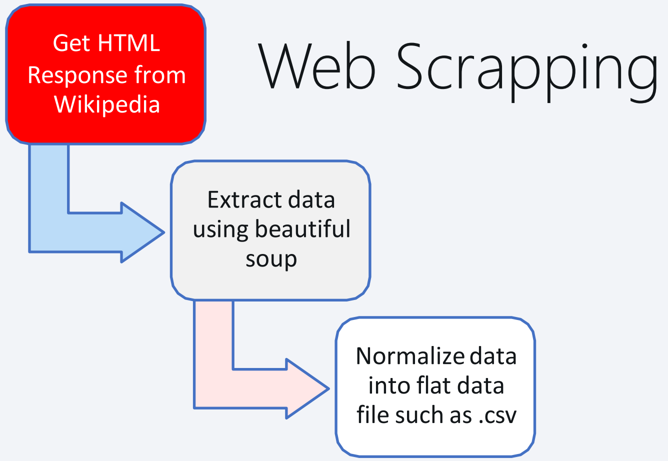
Executive Summary

- Data collection methodology:
  - SpaceX Rest API
  - (Web Scrapping) from Wikipedia
- Perform data wrangling
  - Exploratory Data Analysis
  - Determine Training Labels
- 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 a column for the class
  - Standardize the data
  - Split into training data and test data
  - Find best Hyperparameter for SVM, KNN, Classification Trees and Logistic Regression

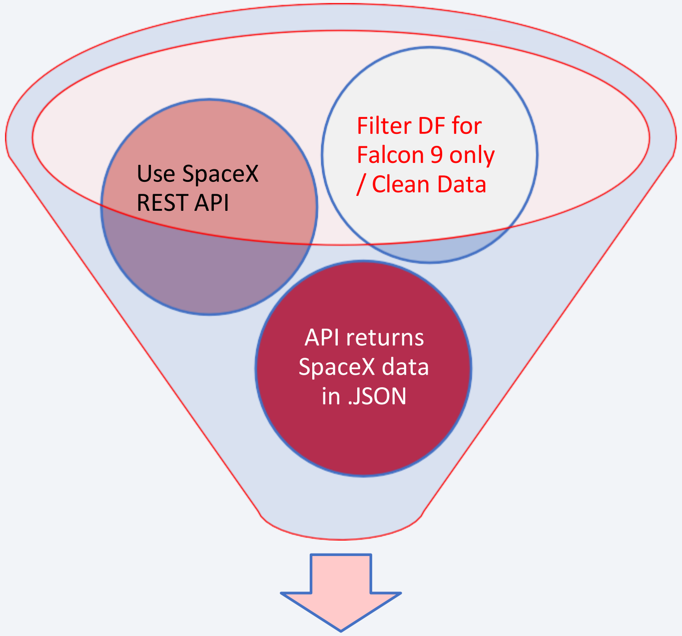
Data Collection

- The following datasets was collected by
  - We worked with SpaceX launch data that is gathered from the SpaceX REST API.
  - This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
  - Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not.
  - The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.
  - Another popular data source for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup.





Data Collection – SpaceX API



Normalize data into flat data file such as .csv

[GitHub URL to Notebook]

simplified flow chart

1 .Getting Response from API

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

2. Converting Response to a .json file

```
response = requests.get(static_json_url).json()
data = pd.json_normalize(response)
```

3. Apply custom functions to clean data

```
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

```
getBoosterVersion(data)
```

4. Assign list to dictionary then dataframe

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

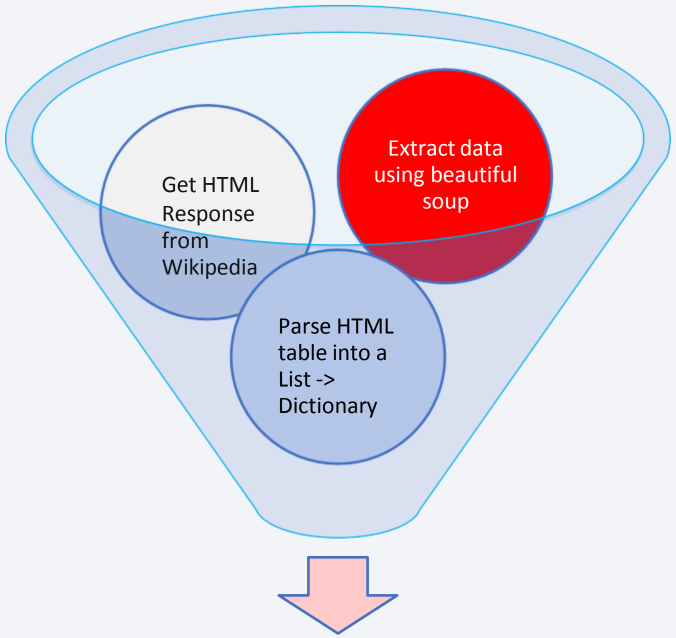
```
df = pd.DataFrame.from_dict(launch_dict)
```

5. Filter dataframe and export to flat file (.csv)

```
data_falcon9 = df.loc[df['BoosterVersion']!="Falcon 1"]
```

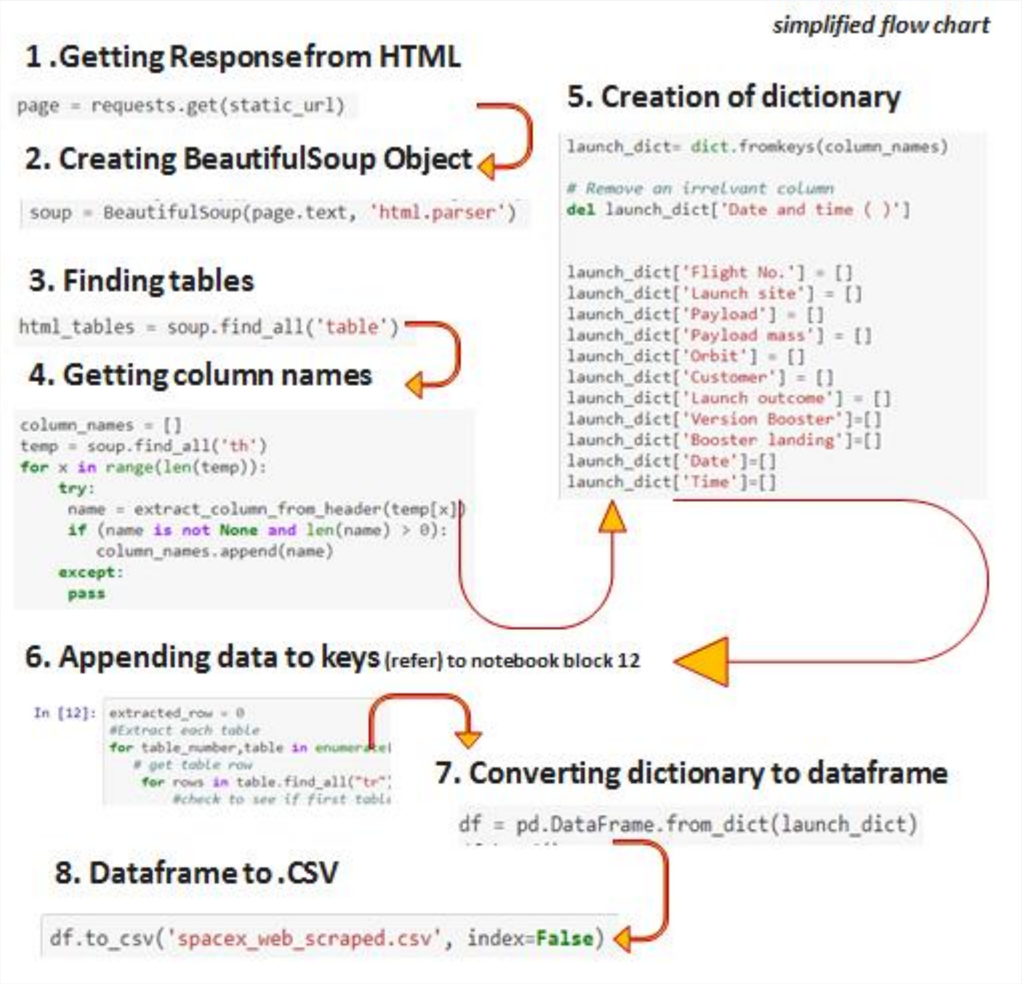
```
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping



Normalize data into flat data file such as .csv

[GitHub URL to Notebook]



Data Wrangling

Introduction

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while 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 on a drone ship False. ASDS means the mission outcome was unsuccessfully landed on a drone ship. We mainly convert those outcomes into Training Labels with 1 means the booster successfully
landed 0 means it was unsuccessful.

Process

Perform Exploratory Data Analysis EDA on Dataset

Calculate the number of launches at each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per orbit type

Export dataset as .CSV

Create a landing outcome label from Outcome column

Work out success rate for every landing in dataset

GitHub URL to Notebook

Each launch aims to an dedicated orbit, and here are some common orbit types:

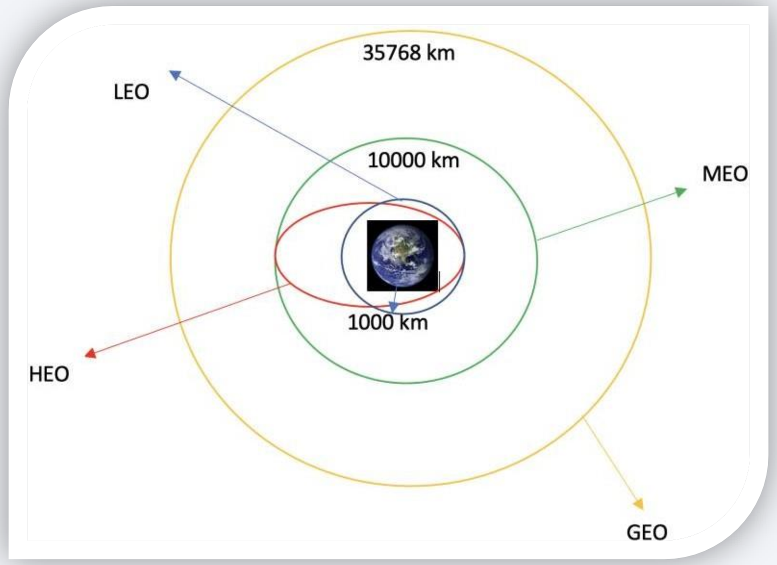


*Diagram showing common orbit types
SpaceX uses*

EDA with Data Visualization

Scatter Graphs being drawn:

- ✓ Flight Number VS. Payload Mass.
- ✓ Flight.Number VS. Launch Site.
- ✓ Payload VS.Launch Site.
- ✓ Orbit VS. Flight Number..
- ✓ Payload VS. Orbit Type.
- ✓ Orbit VS. Payload Mass.

Scatter plots show how much one variable is affected by another. The relationship between two variables is called their correlation . Scatter plots usually consist of a large body of data.

Bar Graph being drawn:

Mean VS. Orbit

A bar diagram makes it easy to compare sets of data between different groups at a glance. The graph represents categories on one axis and a discrete value in the other. The goal is to show the relationship between the two axes. Bar charts can also show big changes in data over time.

Line Graph being drawn:

Success Rate VS. Year

Line graphs are useful in that they show data variables and trends very clearly and can help to make predictions about the results of data not yet recorded.

[GitHub URL to Notebook]

EDA with SQL

Performed SQL queries to gather information about the dataset.

For example of some questions we were asked about the data we needed information about. Which we are using SQL queries to get the answers in the dataset :

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'KSC’
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date where the successful landing outcome in drone ship was achieved.
- Listing the names of the boosters which have success in ground pad 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.
- Listing the records which will display the month names, successful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017
- Ranking the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order.

GitHub URL to Notebook

Build an Interactive Map with Folium

To visualize the Launch Data into an Interactive Map

- ✓ **We took the Latitude and Longitude Coordinates** at each launch site and added a *Circle Marker around each launch site with a label of the name of the launch site.*

- ✓ **We assigned the dataframe launch_outcomes(failures, successes) to *classes 0 and 1*** with Green and Red markers on the map in a MarkerCluster()

- ✓ **Using Haversine's formula we calculated the distance** from the Launch Site to various landmarks to find various trends about what is around the Launch Site to measure patterns. **Lines** are drawn on the map to measure distance to landmarks

Example of some trends in which the Launch Site is situated in.
- Are launch sites in close proximity to railways? Yes
- Are launch sites in close proximity to highways? Yes
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

[GitHub URL to Notebook]

Build a Dashboard with Plotly Dash

The dashboard application contains two charts:

Graphs

- ✓ **Pie Chart** showing the total launches by a certain site/all sites.
- ✓ *Display relative proportions of multiple classes of data.*
- ✓ *Size of the circle can be made proportional to the total quantity it represents.*

Scatter Graph

- ✓ *Showing the relationship with Outcome and Payload Mass (Kg) for the different Booster Versions.*
- ✓ *It is the best method to show you a non-linear pattern.*
- ✓ *The range of data flow, i.e. maximum and minimum value, can be determined.*
- ✓ *Observation and reading are straightforward.*

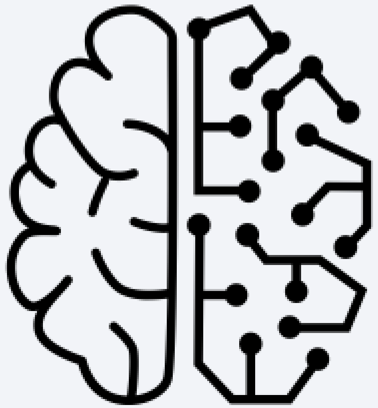
[GitHub URL to Notebook]

Predictive Analysis (Classification)

BUILDING MODEL

- ✓ Load our dataset into NumPy and Pandas.
- ✓ Transform Data.
- ✓ Split our data into training and test data sets.
- ✓ Check how many test samples we have.
- ✓ Decide which type of machine learning algorithms we want to use.
- ✓ Set our parameters and algorithms to GridSearchCV.
- ✓ Fit our datasets into the GridSearchCV objects and train our dataset.

GitHub URL to Notebook



EVALUATING MODEL

- ✓ Check accuracy for each model.
- ✓ Get tuned hyperparameters for each type of algorithms.
- ✓ Plot Confusion Matrix.

IMPROVING MODEL

- ✓ Feature Engineering.
- ✓ Algorithm Tuning.

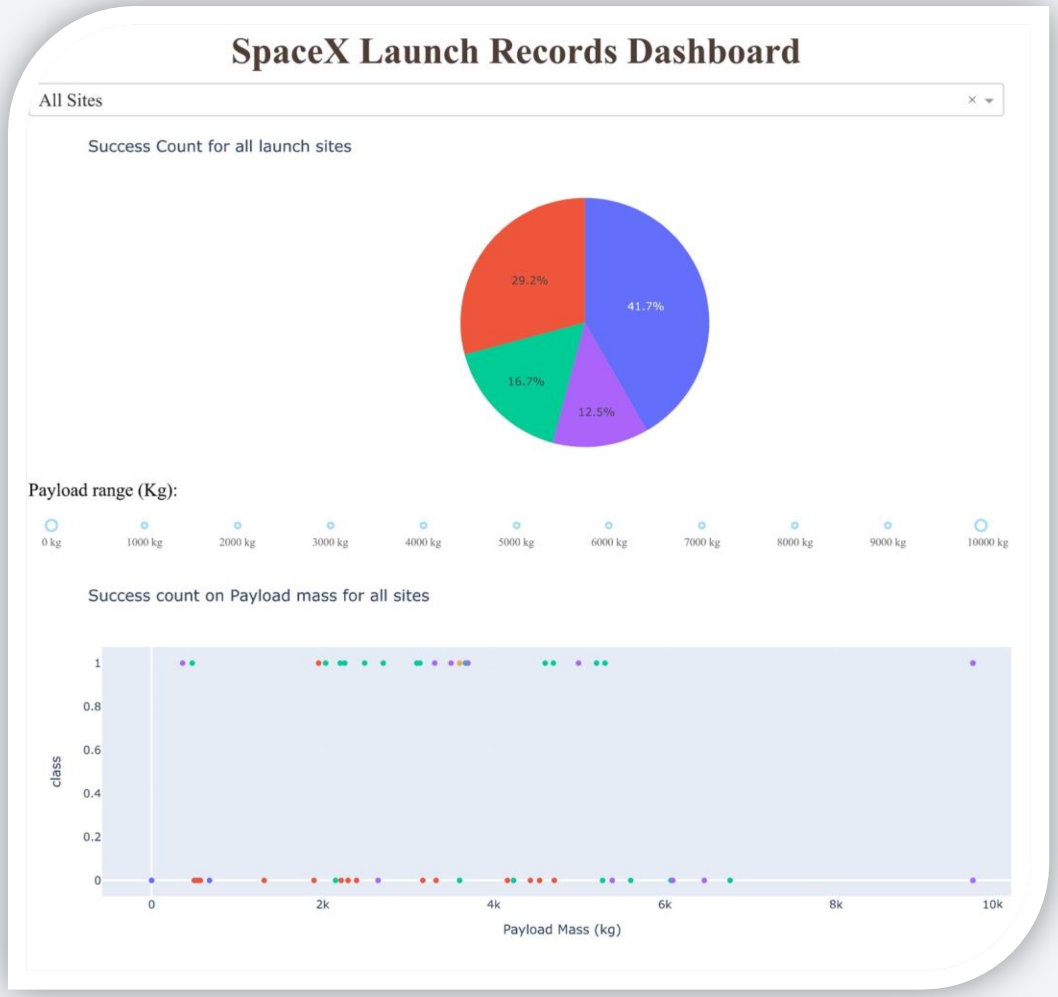
FINDING THE BEST PERFORMING CLASSIFICATION MODEL

- The model with the best accuracy score wins the best performing model.
- In the notebook there is a dictionary of algorithms with scores at the bottom of the notebook..

Results

- Exploratory data analysis results
  - ✓ The results of the exploratory data analysis reveal that different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.
- Predictive analysis results
  - ✓ The predictive analysis results showed that the Decision Tree algorithm was the best classification method with an accuracy of 94%.

Interactive analytics demo in screenshots



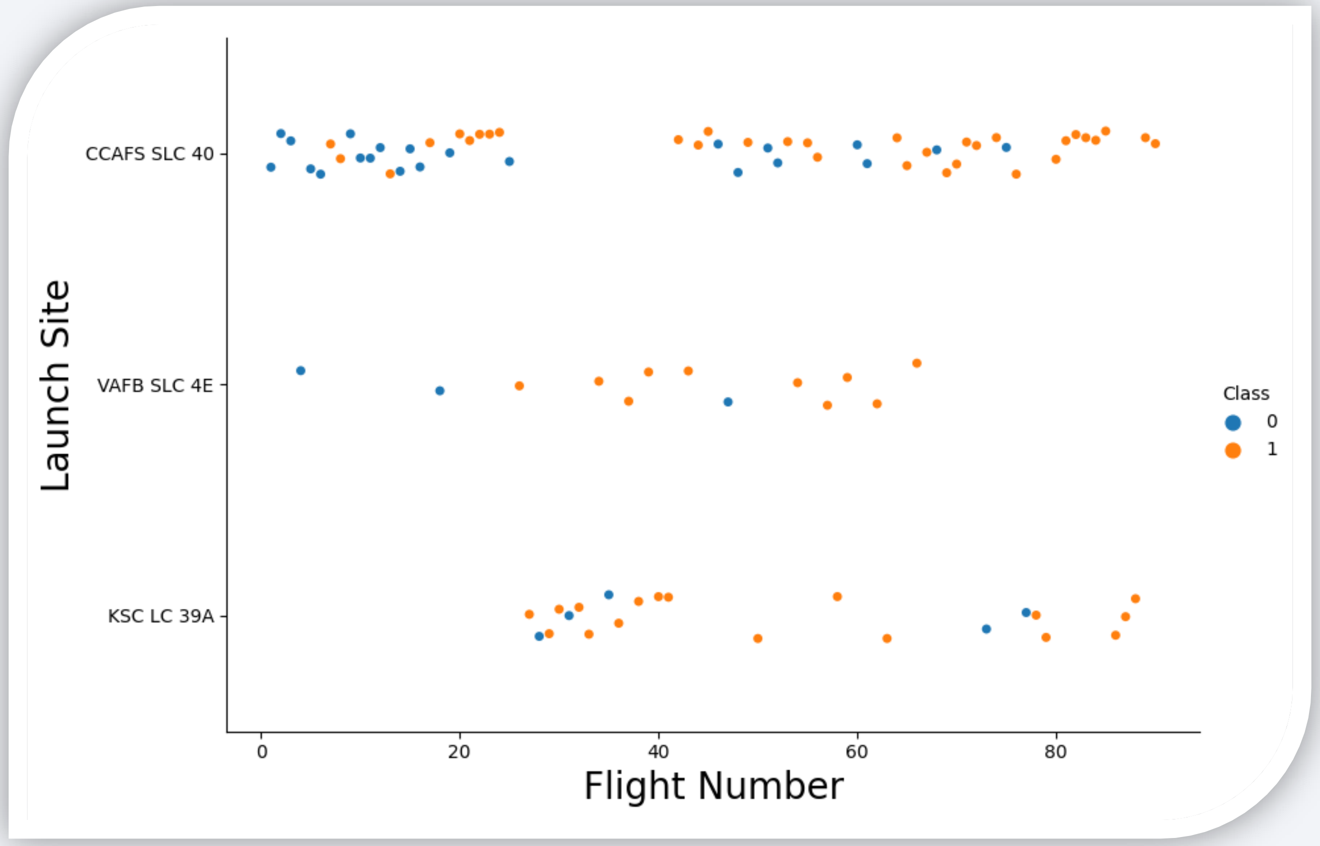
Section 2

Insights drawn from EDA

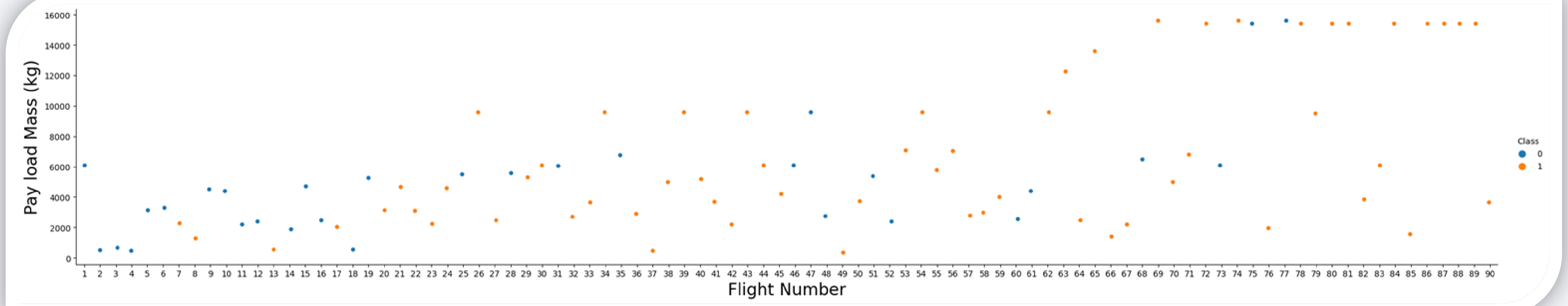
Flight Number vs. Launch Site

- ✓ Class 0 (blue) represents unsuccessful launch, and Class 1 (orange) represents successful launch.
- ✓ This figure shows that the success rate increased as the number of flights increased.
- ✓ As the success rate has increased considerably since the 20th flight, this point seems to be a big breakthrough.

We can deduce that, as the flight number increases in each of the 3 launch sites, so does the success rate. For instance, the success rate for the VAFB SLC 4E launch site is 100% after the Flight number 50. Both KSC LC 39A and CCAFS SLC 40 have a 100% success rates after 80th flight.



Payload vs. Launch Site

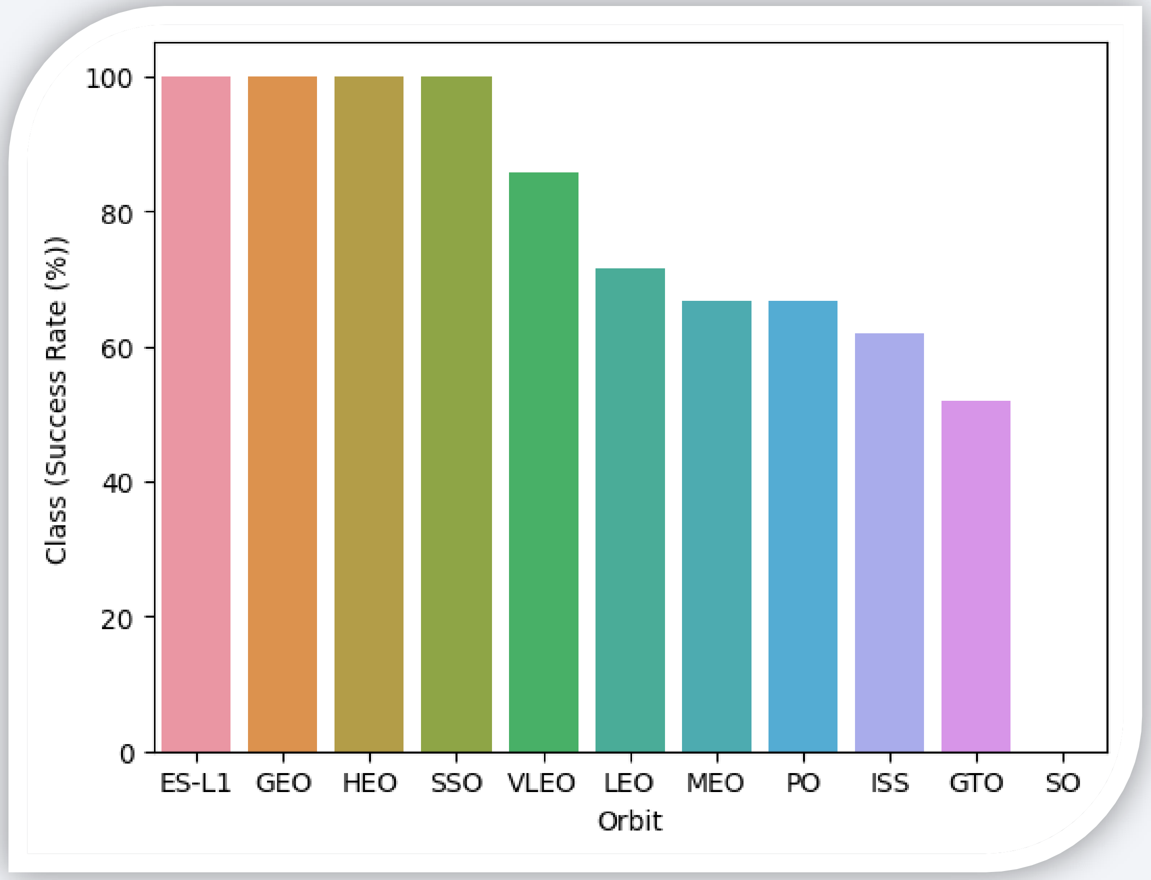


The flight number increases, the first stage is more likely to land successfully.

Success Rate vs. Orbit Type

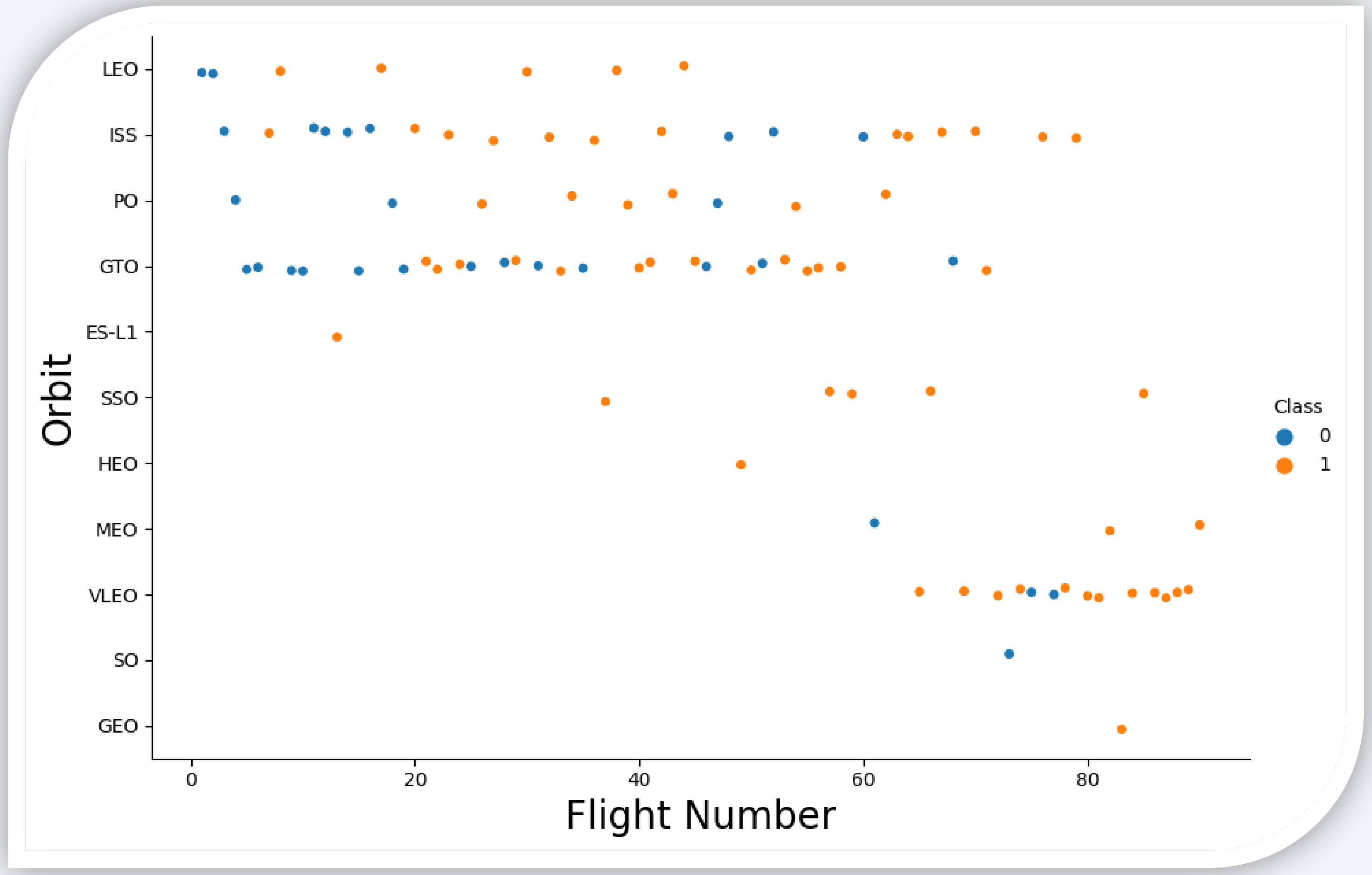
Analyze the plotted bar chart try to find which orbits have high success rate.

✓Orbits ES-L1, GEO, HEO & SSO have the highest success rates at 100%.



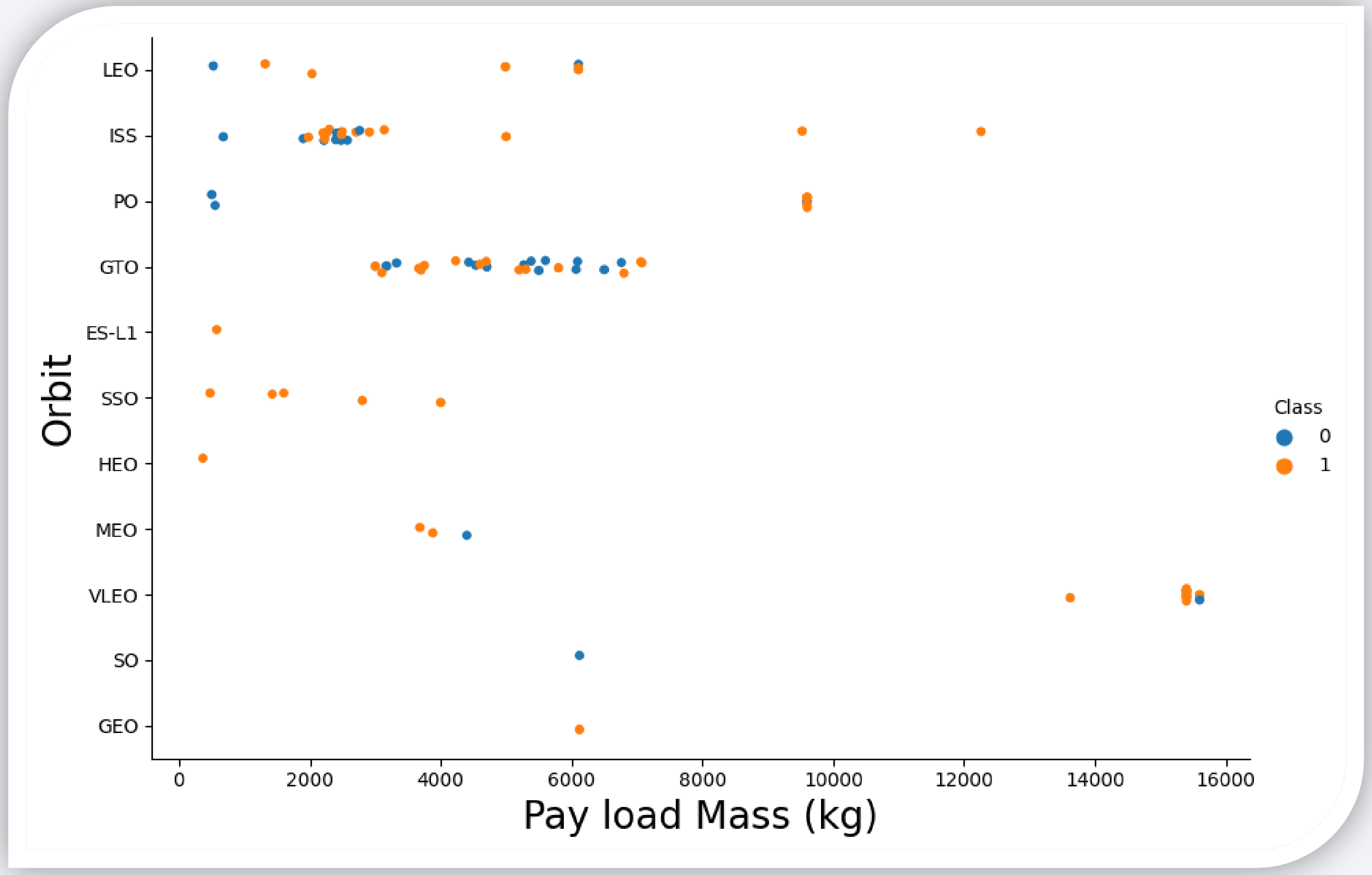
Flight Number vs. Orbit Type

In the LEO orbit the success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



Payload vs. Orbit Type

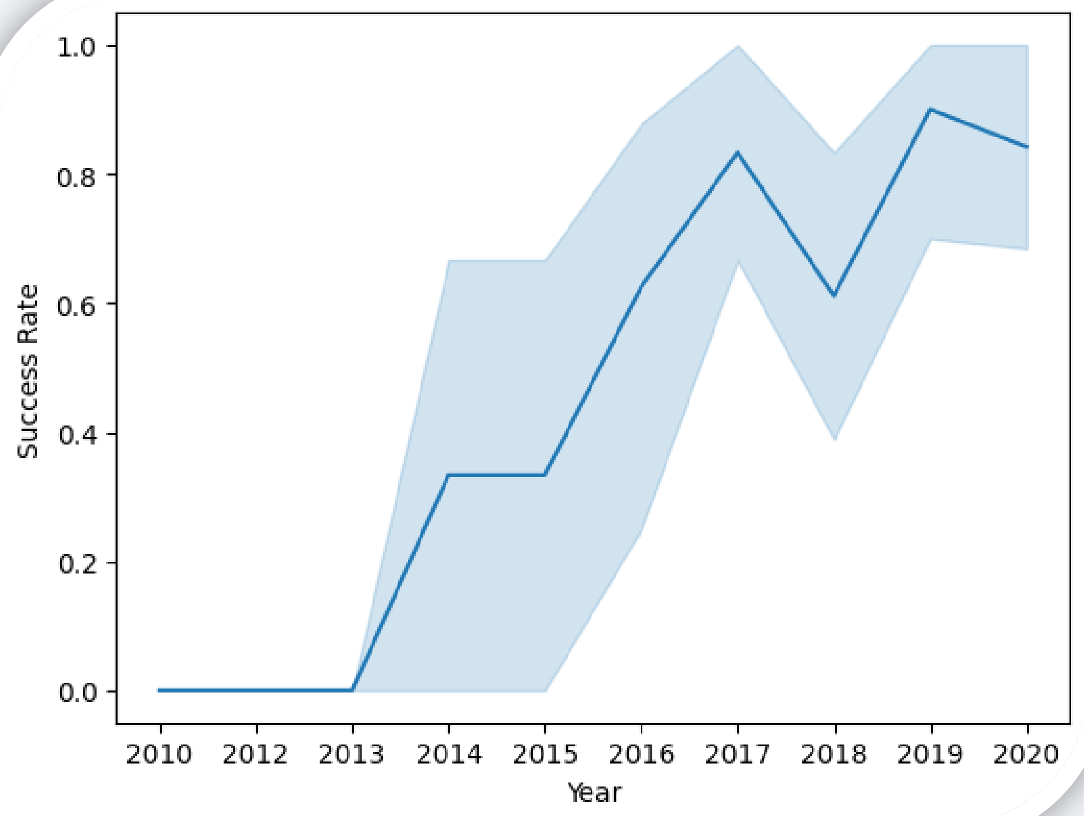
- ✓With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- ✓However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.



Launch Success Yearly Trend

Observation

✓The success rate since 2013 kept increasing till 2020.



All Launch Site Names

SQL QUARY

%sql SELECT DISTINCT LAUNCH_SITE as Launch_Sites FROM SPACEXTBL;

QUERY EXPLAINATION

Using the word ***DISTINCT*** in the query means that it will only show Unique values in the ***Launch_Site*** column from ***SPACEXTBL***

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

Launch Site Names Begin with 'CCA'

SQL QUARY

```
%sql SELECT * FROM SPACEXTBL
     WHERE Launch_Site LIKE 'CCA%'
     LIMIT 5;
```

QUERY EXPLAINATION

Using the word ***TOP 5*** in the query means that it will only show 5 records from ***SPACEXTB*** and ***LIKE*** keyword has a wild card with the words ***'CCA%'*** the percentage in the end suggests that the ***Launch_Site*** name must start with ***CCA***.

Date	Time (UTC)	Booster_ Version	Launch_Site	Payload	PAYLOAD_ MASS__KG_	Orbit	Customer	Mission_ Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

SQL QUARY

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) as Payload_Mass_Kg_TOTAL, Customer FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
```

Payload_Mass_Kg_TOTAL	Customer
45596	NASA (CRS)

QUERY EXPLAINATION

Using the function ***SUM*** summates the total in the column ***PAYLOAD_MASS_KG_*** *as* ***Payload_Mass_Kg_TOTAL***
Further adding column ***Customer*** into the output from ***SPACEXTBL***
The ***WHERE*** clause filters the dataset to only perform calculations on ***Customer*** = ***NASA (CRS)***

Average Payload Mass by F9 v1.1

SQL QUARY

%sql SELECT AVG(PAYLOAD_MASS__KG_) as Payload_Mass_Kg_TOTAL, Customer, Booster_Version FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.1%';

Payload_Mass_Kg_TOTAL	Customer	Booster_Version
2534.6666666666665	MDA	F9 v1.1 B1003

QUERY EXPLAINATION

Using the function ***AVG*** summates the average in the column ***PAYLOAD_MASS_KG_*** *as* ***Payload_Mass_Kg_TOTAL***
Further adding columns ***Customer, Booster_Version*** into the output from ***SPACEXTBL***
The ***WHERE*** clause filters the dataset to only perform calculations on ***Booster_Version LIKE 'F9 v1.1%'***

First Successful Ground Landing Date

SQL QUARY

%sql SELECT MIN(Date), [Landing _Outcome] as [Landing Outcome] FROM SPACEXTBL WHERE [Landing _Outcome] LIKE 'Success (ground pad)%';

MIN(Date)	Landing Outcome
01-05-2017	Success (ground pad)

QUERY EXPLAINATION

Using the function ***MIN*** works out the minimum date in the column ***Date***
Further adding column ***[Landing _Outcome] as [Landing Outcome]*** into the output from ***SPACEXTBL***
LIKE keyword has a wild card with the words 'Success (ground pad)%‘ the percentage in the end suggests that the ***Landing Outcome*** name must start with ***Success (ground pad)***.

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL QUARY

```
%%sql SELECT Booster_Version
         FROM SPACEXTBL
         WHERE Mission_Outcome = 'Success'
        AND PAYLOAD_MASS__KG_ > 4000
        AND PAYLOAD_MASS__KG_ < 6000;
```

QUERY EXPLAINATION

SELECT only ***Booster_Version*** from ***SPACEXTBL***
The **WHERE** clause filters the dataset to ***Mission_Outcome*** = '***Success'***
The **AND** clause specifies additional filter conditions
Payload_MASS_KG_ > 4000 AND Payload_MASS_KG_ < 6000

Booster_Version
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1014
F9 v1.1 B1016
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1030
F9 FT B1021.2
F9 FT B1032.1
F9 B4 B1040.1
F9 FT B1031.2
F9 FT B1032.2
F9 B4 B1040.2
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5 B1048.3
F9 B5 B1051.2
F9 B5B1060.1
F9 B5 B1058.2
F9 B5B1062.1

Total Number of Successful and Failure Mission Outcomes

SQL QUARY

```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) as Total FROM SPACEXTBL GROUP BY Mission_Outcome;
```

Mission_Outcome	Total
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

QUERY EXPLAINATION

The ***COUNT()*** function is used to count the number of occurences of different ***Mission_Outcome*** with the help of the ***GROUPBY*** clause applied to the ***Mission_Outcome*** column.
A list of the ***Total*** number of successful and failure mission outcomes returned.

There have been 99 successful mission outcomes out of 101 missions

Boosters Carried Maximum Payload

SQL QUARY

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

QUERY EXPLAINATION

SELECT only ***Booster_Version*** from ***SPACEXTBL***
The **WHERE** clause filters the dataset to ***PAYLOAD_MASS__KG_*** with subqueries ***(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)*** used to produce the results.

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

SQL QUARY

```
%%sql SELECT substr(Date,7,4) as Year, substr(Date, 4, 2) as Month, Booster_Version, Launch_Site, Payload, PAYLOAD_MASS__KG_, Mission_Outcome, [Landing _Outcome]
        FROM SPACEXTBL WHERE substr(Date,7,4)='2015' AND [Landing _Outcome] = 'Failure (drone ship)';
```

Year	Month	Booster_ Version	Launch_Site	Payload	PAYLOAD_ MASS__KG_	Mission_ Outcome	Landing _Outcome
2015	01	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	Success	Failure (drone ship)
2015	04	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898	Success	Failure (drone ship)

QUERY EXPLAINATION

Used the ***subsrt()*** in the select statement to get the month and year from the date column where ***substr(Date,7,4)='2015'*** for year and ***Landing_Outcome*** was ‘Failure (drone) and return the records matching the filter.

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

SQL QUARY

```
%%sql SELECT [Landing _Outcome],
    COUNT([Landing _Outcome])
    as [Successful Landing Outcomes]
    FROM SPACEXTBL
    WHERE [Landing _Outcome] LIKE 'Success%'
    AND (Date BETWEEN '04-06-2010' AND '20-03-2017')
    GROUP BY [Landing _Outcome]
    ORDER BY Date DESC;
```

Landing _Outcome	Successful Landing Outcomes
Success (ground pad)	6
Success (drone ship)	8
Success	20

QUERY EXPLAINATION

The ***COUNT()*** function is used to count the number of occurences of different ***Landing _Outcome.*** The WHERE clause filters the dataset to ***Landing _Outcome LIKE 'Success%'*** with the help of the ***GROUPBY*** clause applied to the ***Landing _Outcome*** column in descending order with ***ORDER BY Date DESC***

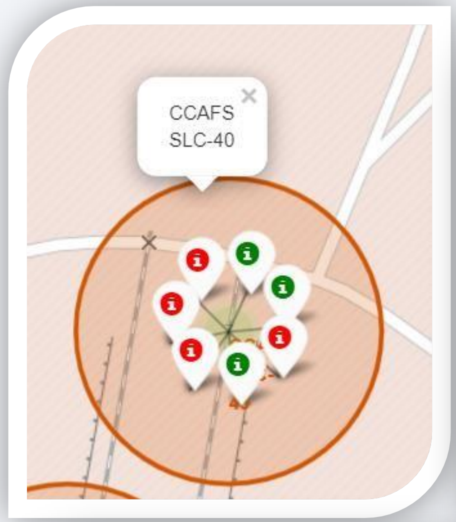
Section 3

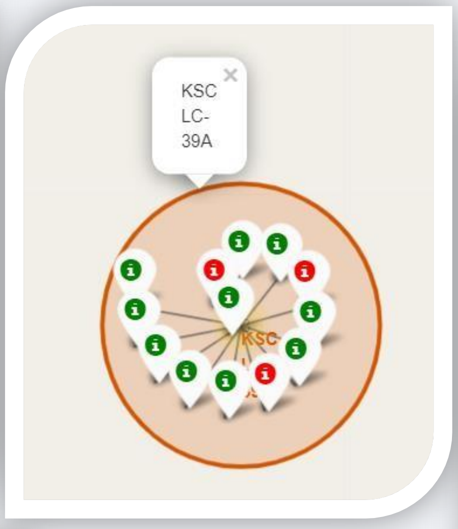
Launch Sites Proximities Analysis

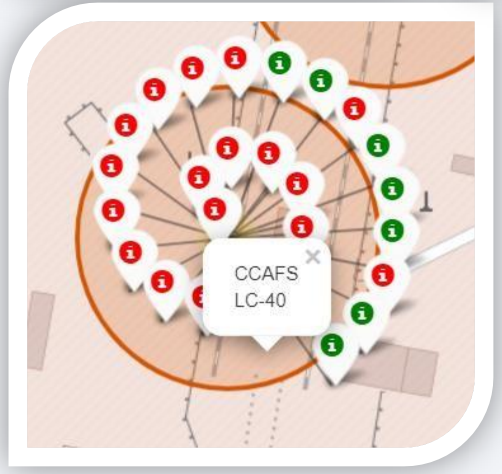
All Launch Sites' Location Markers On A Global Map

We can see that the SpaceX launch sites are in the United States of America coasts. Florida and California.

The Color-labeled Launch Outcomes On The Map





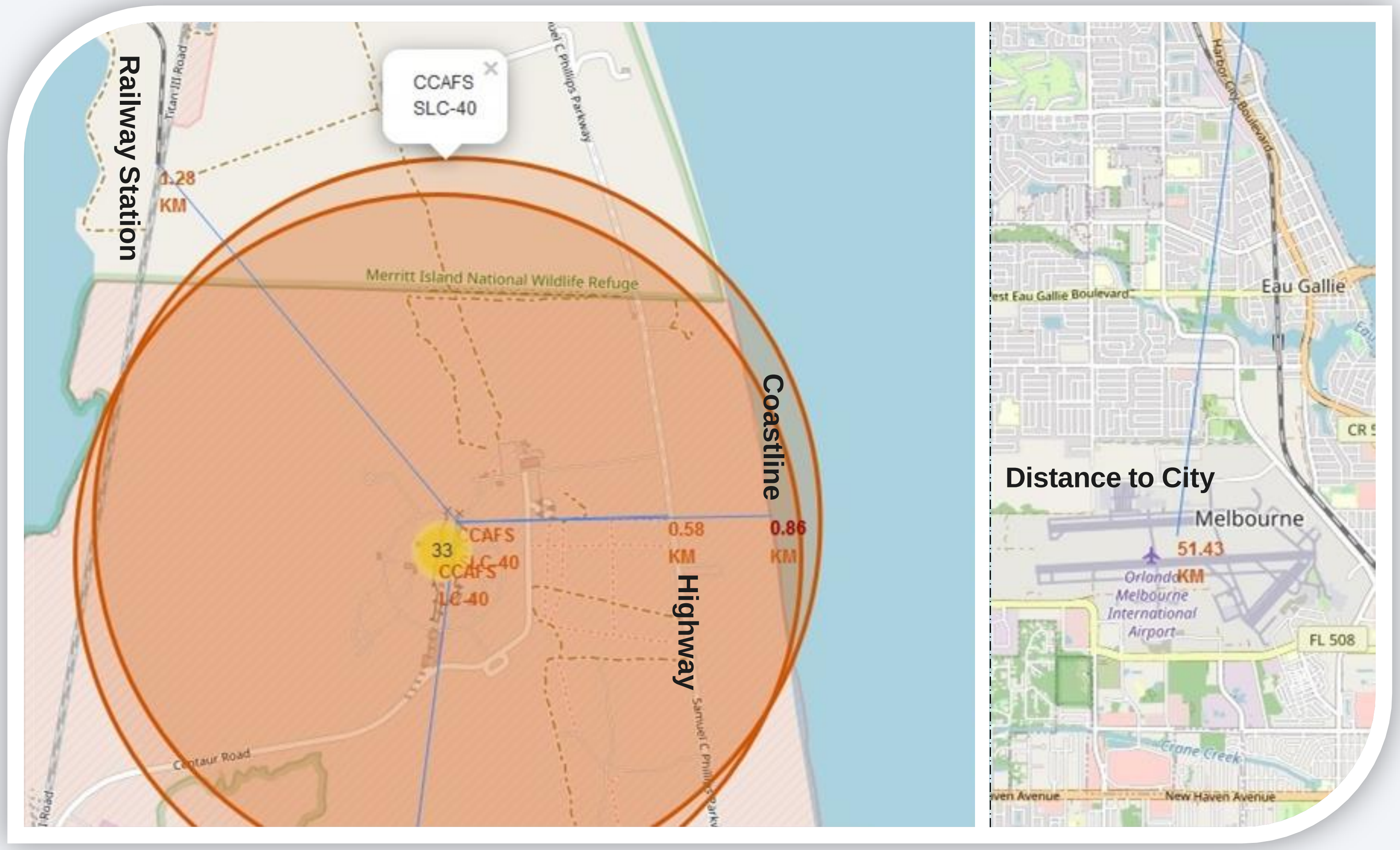


Florida Launch Sites

Green Marker shows successful Launches and Red Marker shows Failures

California Launch Site

Launch Site To Its Proximities of Selected Places



Explore to generate the Folium Map and show the launch site CCAFS-SLC-40 to its proximities such as railway, highway, coastline, city with distance calculated and displayed.

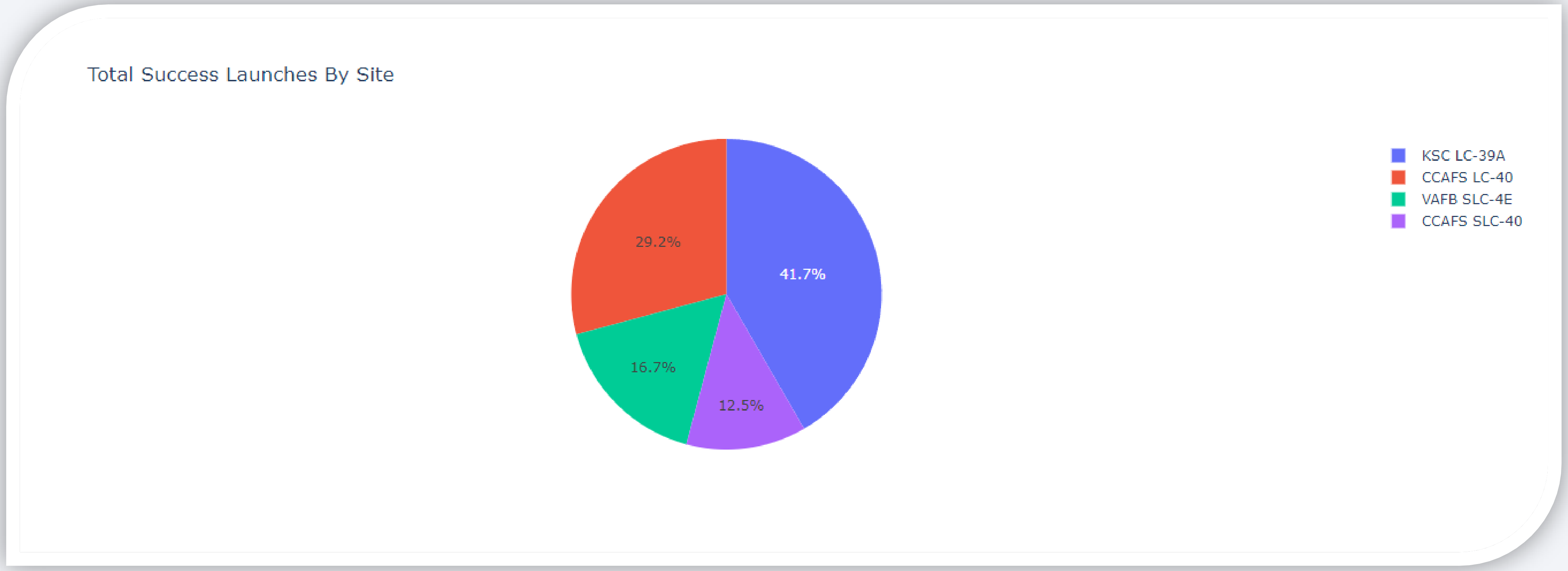
<u>Distance lines to the proximities:</u>

- ✓ Are launch sites in close proximity to railways? –Yes
- ✓ Are launch sites in close proximity to highways? –Yes
- ✓ Are launch sites in close proximity to coastline? –Yes
- ✓ Do launch sites keep certain distance away from cities? -Yes

Section 4

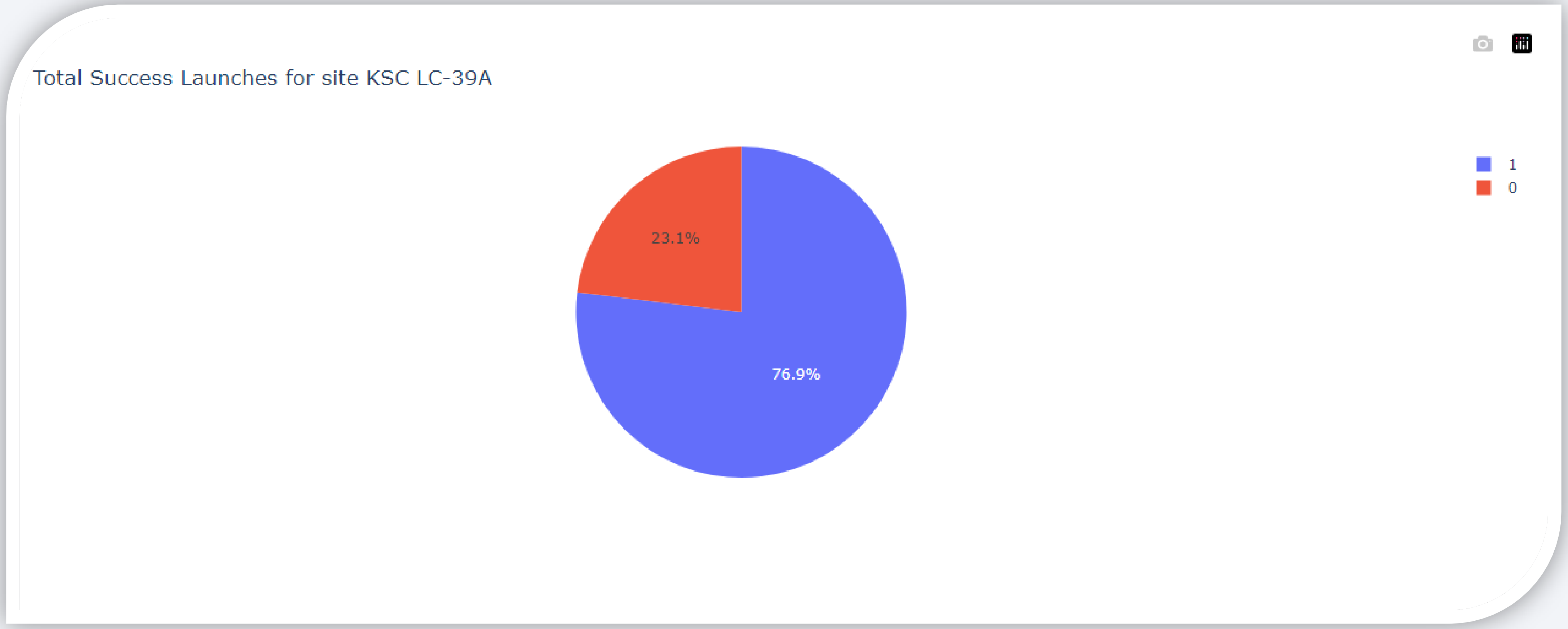
Build a Dashboard with Plotly Dash

Pie-Chart For Launch Success Count For All Sites



Launch site KSC LC-39A has the highest launch success rate at 41.7 % followed by CCAFS LC-40 at 29.2%, VAFB SLC-4E at 16.7% and lastly launch site CCAFS SLC-40 with a success rate of 12.5%.

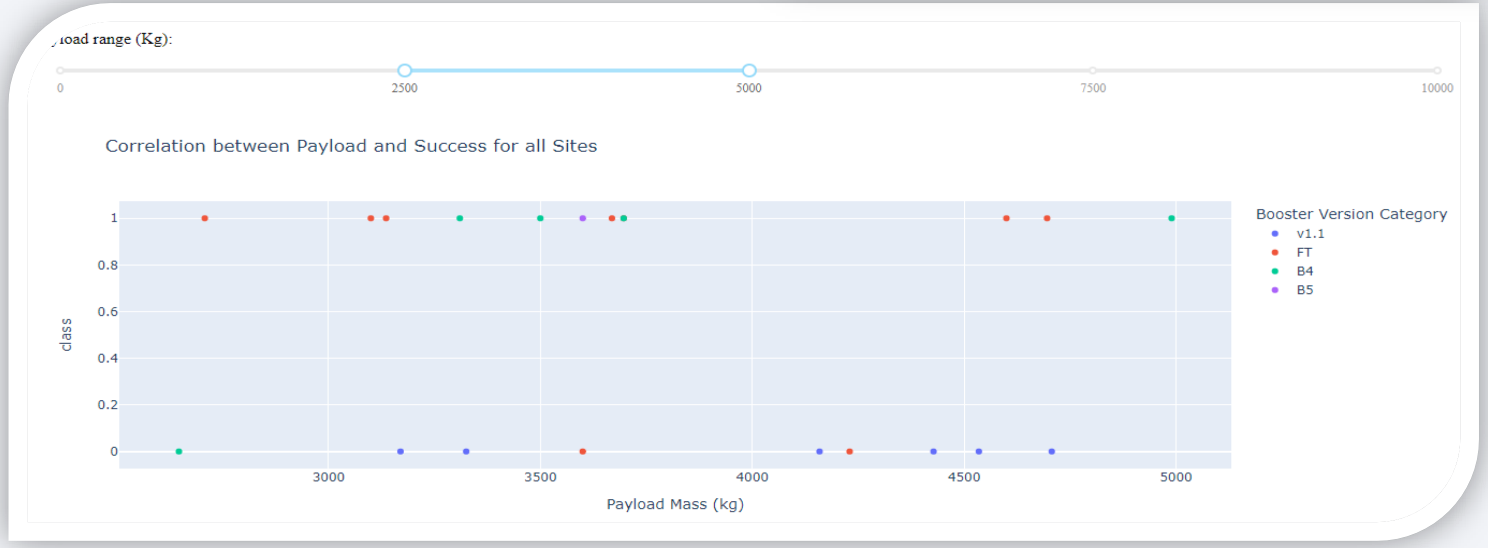
Pie-chart For The Launch Site With Highest Launch Success Ratio



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate.

Payloads vs Launch Outcome

The launch success rate for payloads 0-2500 kg is slightly lower than that of payloads 2500-5000 kg. There is in fact not much difference between the two.



The booster version that has the largest success rate, in both weight ranges is the ***v1.1***.

Section 5

Predictive Analysis (Classification)

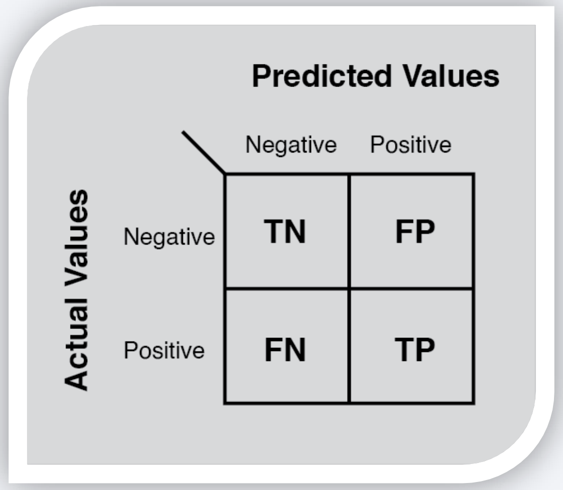
Classification Accuracy

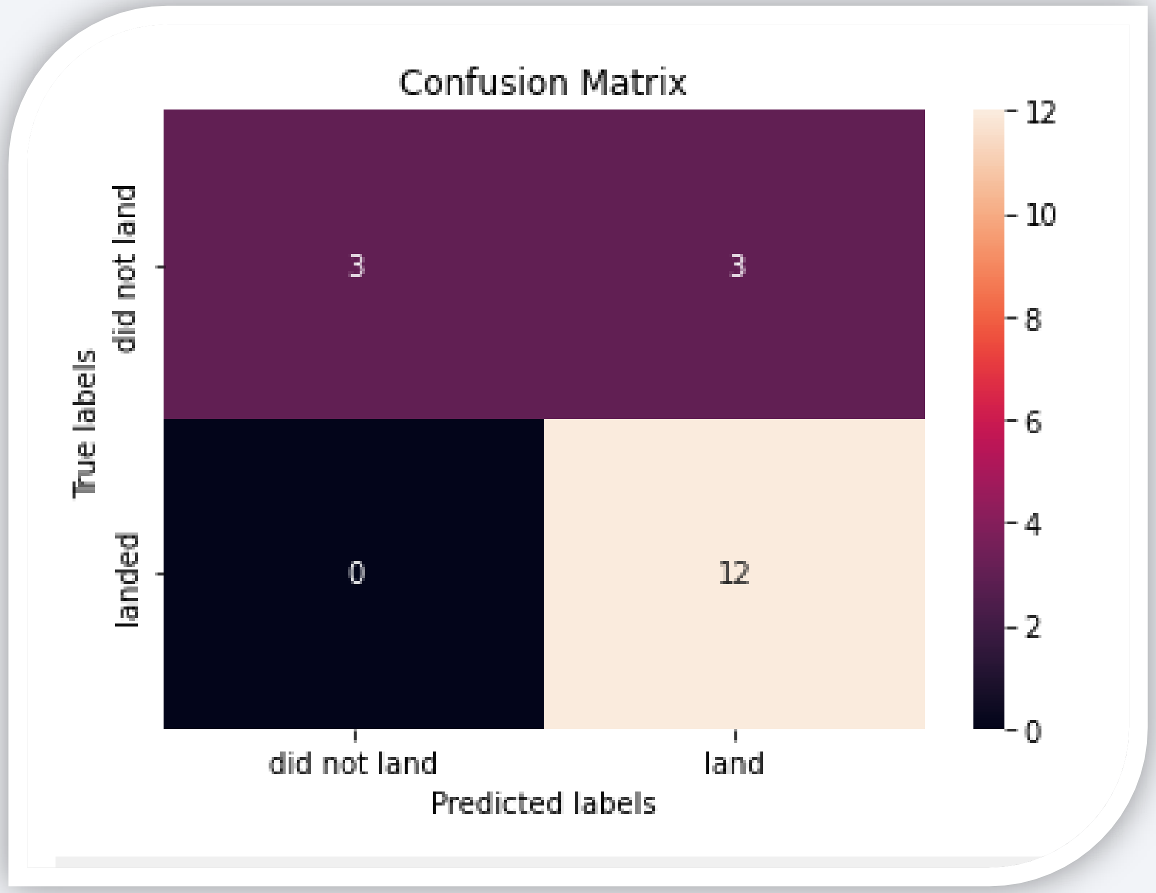
	Method	Test Data Accuracy
0	Logistic Regression	0.833333
1	Support Vector Machine	0.833333
2	Decision Tree classifier	0.833333
3	K-Nearest Neighbors	0.833333

All the methods perform equally on the test data: i.e. They all have the same accuracy of 0.833333 on the test Data.

Confusion Matrix

✓ All the 4 classification model had the same confusion matrixes and were able equally distinguish between the different classes. The major problem is false positives for all the models.





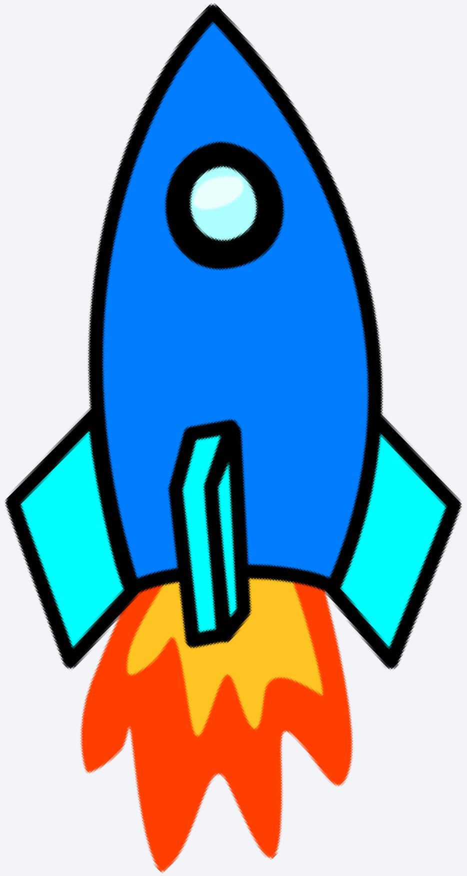
Conclusions

- ✓ The analysis showed that there is a positive correlation between number of flights and success rate as the success rate has improved over the years.
- ✓ There are certain orbits like SSO, HEO, GEO, and ES-L1 where launches were the most successful.
- ✓ Success rate can be linked to payload mass as the lighter payloads generally proved to be more successful than the heavier payloads.
- ✓ The launch sites are strategically located near highways and railways for transportation, but also far away from cities for safety.
- ✓ With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing (unsuccessful mission) are both there here.
- ✓ And finally the success rate since 2013 kept increasing till 2020.

Appendix

GitHub Repository:

https://github.com/Arnab-11/IBM-Data-Science-Capstone-Project-Space-X



Thank you!