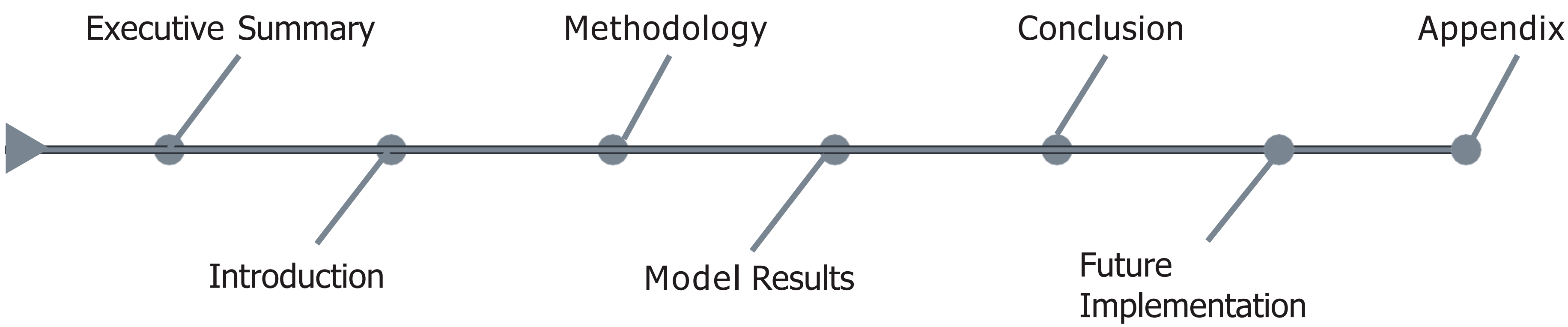
Rocket Analytics

Unveiling Key Performance Metrics for Optimal Boost Efficiency

Danish Nurhira Putra
IBM data scientist course

Project Outline



Executive Summary

1. Data preprocessing using Pandas functions, including handling null values and feature engineering.
2. Perform exploratory data analysis (EDA) using visualization and SQL queries.
3. Tree model present the best performance of predicting launch outcome.
4. By deploying the tree model, we can predict a 94% accuracy in determining whether the launch outcome will be successful or not.

Introduction

Over the past five decades, people have been trying to explore outer space, wondering what the experience might be like.

Advances in technology have increased the number and variety of rocket launches, but achieving a successful launch remains challenging.

I aim to collect and analyze data on rocket launches to better understand and improve the factors contributing to their success. Our project specifically focuses on SpaceX Falcon 9, which has had recent launches with abundant data available for analysis.

Methodology

Methodology

Executive Summary

- Data collection methodology
  1. Using get request on SpaceX API to store data into data frame.
  2. Loading SpaceX Falcon 9 data on Wikipedia and storing data into data frame.
- Perform data wrangling
  1. Remove the unuse column
  2. Build up data frame to store data
  3. Make sure data type is align with the analysis
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection - SpaceX API

To load data from the SpaceX API into a DataFrame, we use the requests function to retrieve the data and then perform data wrangling to clean and prepare it for analysis.

Get data from API using Request. → Create column list and stored data into list. → Store list into dictionary and transfer to data frame. → Filter data to only contain Falcon 9. → Checking missing value and replace by mean value.

URL

https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/SpaceX%20Data%20Collection.ipynb

Data Collection - Scraping

To load data from the Wikipedia webpage into a DataFrame, we use the requests and beautiful soup function to retrieve the target table.

Visit Falcon 9 and Falcon Heavy Launches Records from Wikipedia.

Make sure the table we need and use ger request to load html object.

Transfer html by Beautiful soup library.

Extract target table and store into launch dictionary.

Check the data type and transfer dictionary to data frame .

URL

https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/SpaceX%20Webscraping.ipynb

Data Wrangling

1. Calculate the number of launches on each site.
2. Calculate the number and occurrence of each orbit.
3. Calculate the number and occurrence of mission outcome of the orbits.
4. Create a landing outcome label from Outcome column.

URL

https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/SpaceX%20Webscraping.ipynb

EDA with Data Visualization - 1

1. Scatter plot display relationship between "FlightNumber" and "PayloadMass"
   We could see whether success of launch is not relevant to "PayloadMass".
2. Scatter plot display relationship between "FlightNumber" and "LaunchSite"
   We could see CCAFS SLC 40 launch site have relatively low success rate.
3. Bar plot display relationship between "Orbit" and "Class" as success rate.
   We could see some orbit have 100% success rate and one the them are 0%.
4. Scatter plot display relationship between "FlightNumber" and "Orbit"
   We could see GTO and ISS orbit have relatively low success rate.
5. Scatter plot display relationship between "Orbit" and "PayloadMass" .
   We could see different orbits are use in different PayloadMass.
6. Line chart display relationship between "Year" and "Success rate"
   Success rate in increasing through the time.

URL
https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/Data%20Exploration%20with%20SQLlite.ipynb

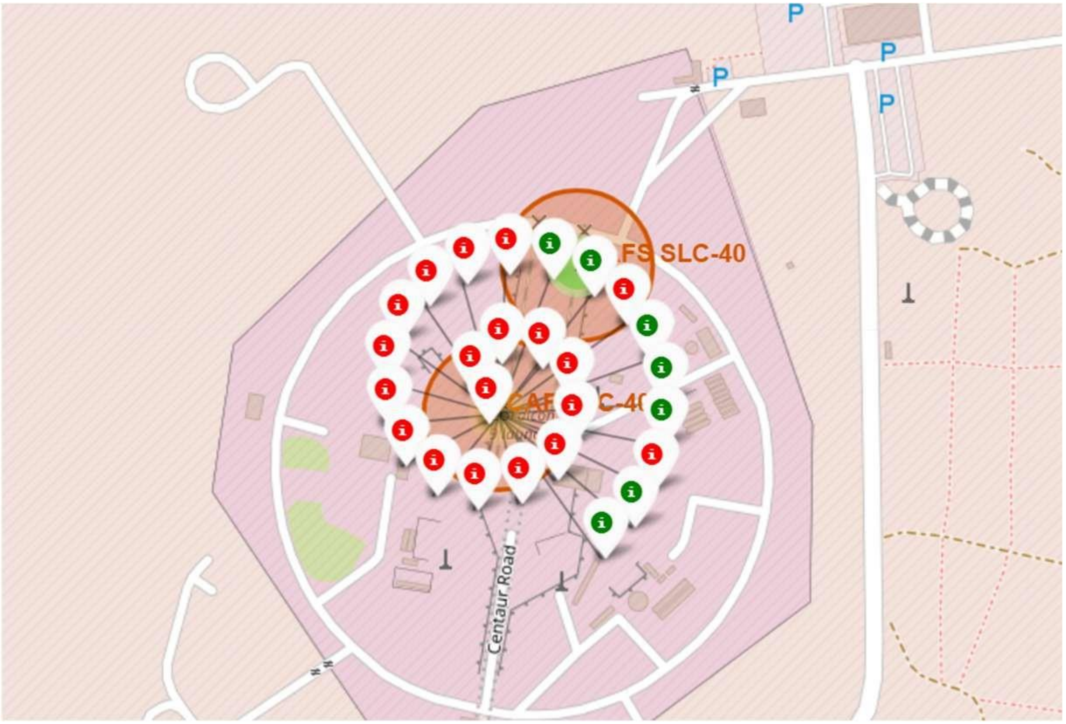
EDA with SQL

- Display the names of the unique launch sites in the space mission¶
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1¶
- List the date when the first succesful landing outcome in ground pad was acheived.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000¶
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery¶
- List the 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.

URL

https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/Exploring%20and%20Preparing%20Data%20Visualizations.ipynb

Build an Interactive Map with Folium



1. Add location
2. Add launch outcome
3. Add distance for important facility

URL

https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/Folium%20Site%20Locations.ipynb

Build a Dashboard with Plotly Dash

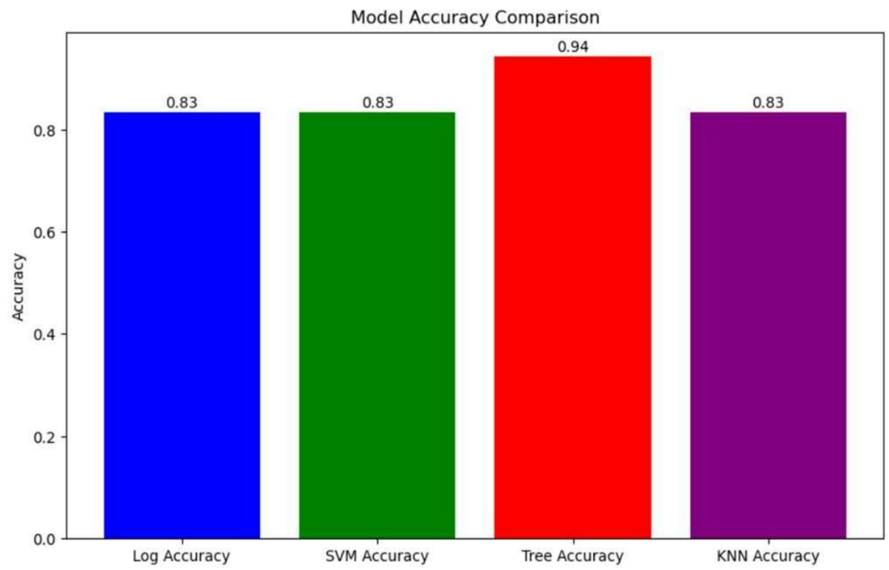
Build an interactive dashboard to visualize the proportion of successful launches at different launch sites using pie charts. Additionally, create a scatter chart to examine the relationship between payload mass and launch outcome.

URL

https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

By using grid search, we built several models, including Logistic Regression, SVM, Decision Tree, and KNN. We then compared the accuracy after selecting the best parameters for each model. The tree model demonstrated the best performance according to our accuracy metric.



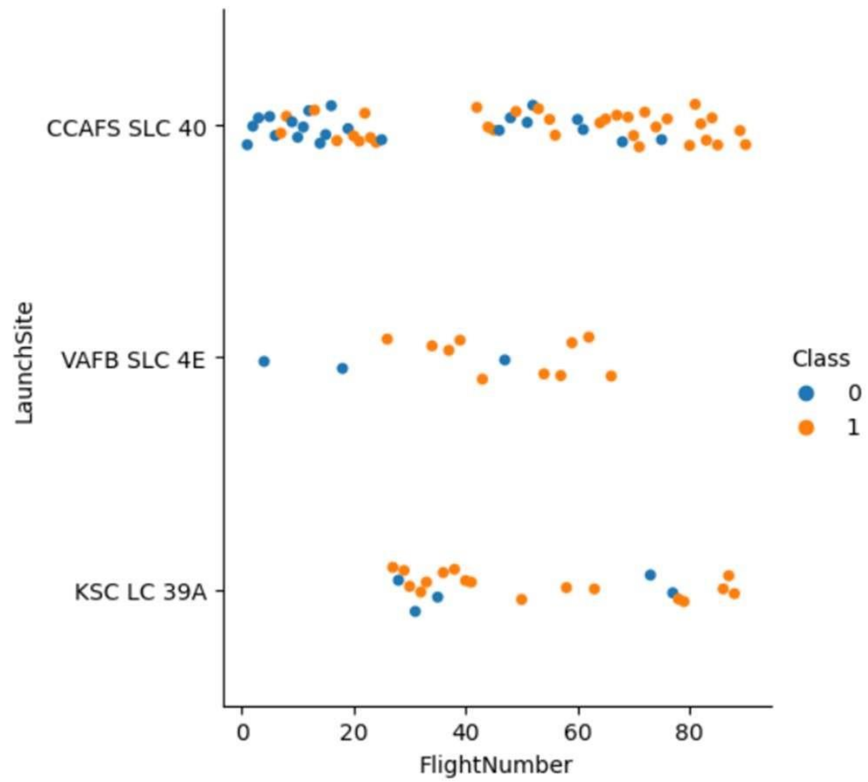
URL
https://github.com/DanishNurhira/Capstone-Project-Coursera/blob/main/SpaceX_Machine%20Learning%20Prediction.ipynb

Insight drawn form EDA

Flight Number vs. Launch Site

Scatter plot display relationship between “FlightNumber” and "LaunchSite"

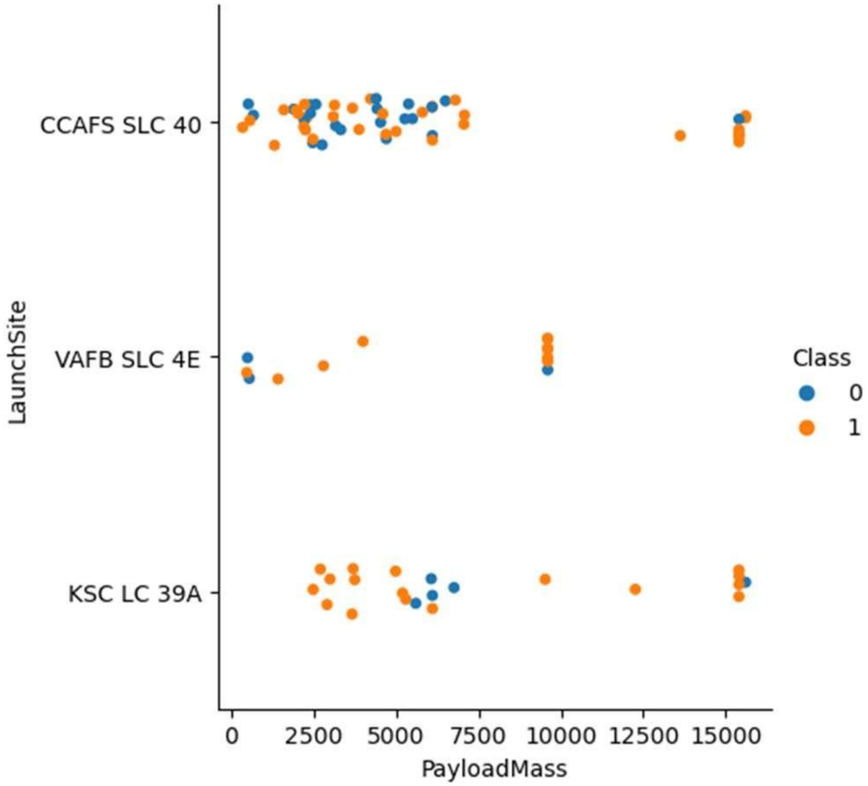
We could see CCAFS SLC 40 launch site have relatively low success rate.



Payload vs. Launch Site

Scatter plot display relationship between “Payload” and "LaunchSite"

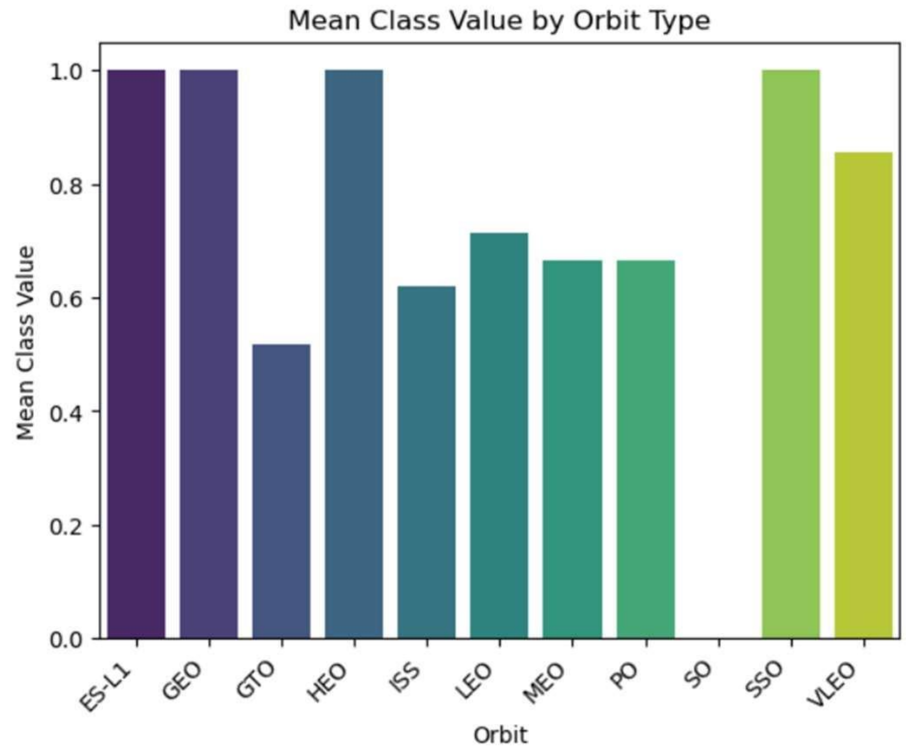
We can see that there is no strong relationship between payload mass and launch success for each launch site.



Success Rate vs. Orbit Type

Bar plot display relationship between "Orbit" and "Class" as success rate.

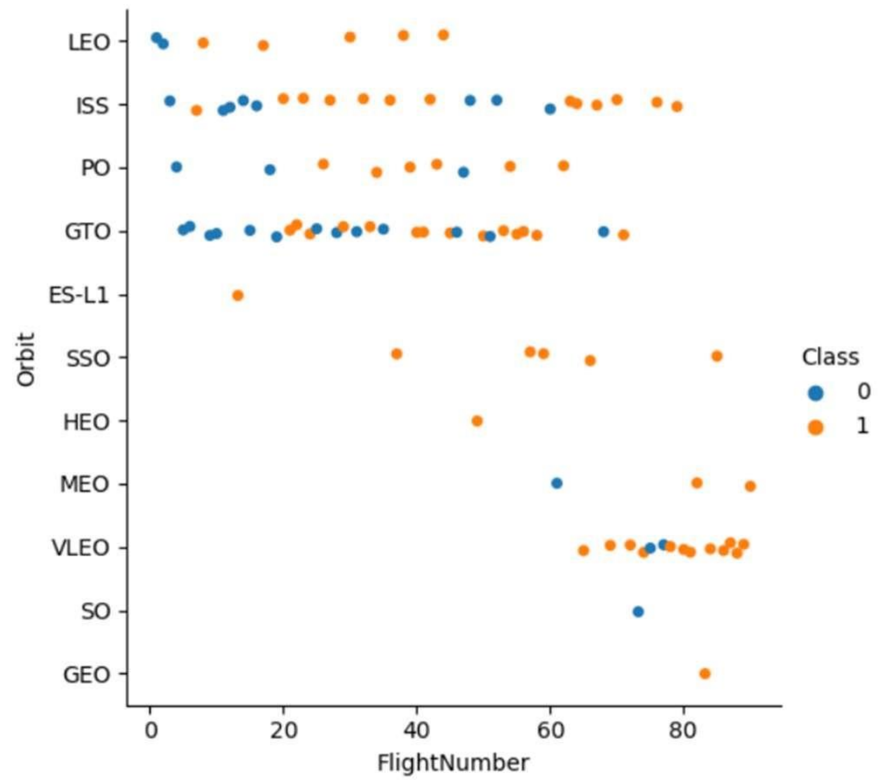
We could see some orbit have 100% success rate and one the them are 0%.



Flight Number vs. Orbit Type

Scatter plot display relationship between "FlightNumber" and "Orbit"

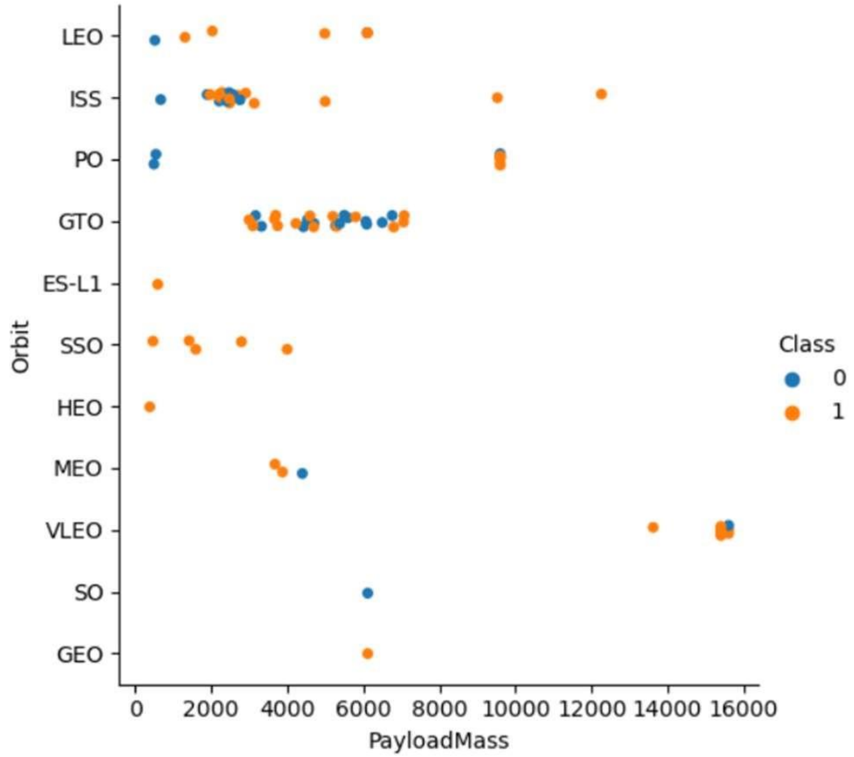
We could see GTO and ISS orbit have relatively low success rate.



Payload vs. Orbit Type

Scatter plot display relationship between "Orbit" and "PayloadMass" .

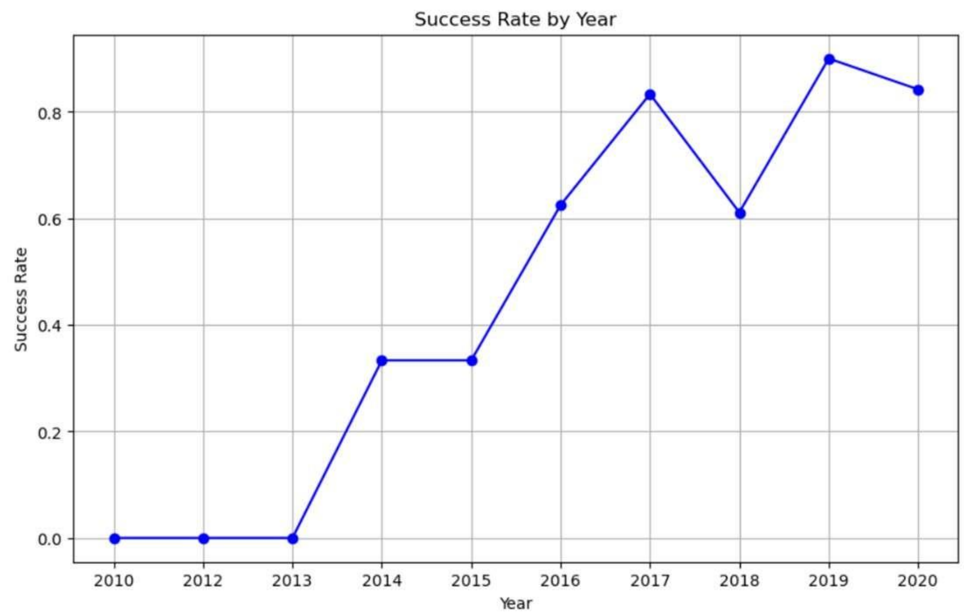
We could see different orbits are use in different PayloadMass.



Launch Success Yearly Trend

Line chart display relationship between "Year" and "Success rate"

Success rate in increasing through the time.



All Launch Site Names

Task 1

```
[9]: %sql SELECT DISTINCT Launch_Site FROM SPACEXTBL;

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

[9]:

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

The query displays the unique values in the 'launch_site' column.

Launch Site Names Begin with 'CCA'

The query displays the records for name begin with ‘CCA’ the 'launch_site' column.

```
%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

The query displays total payloadmass among all records.

```
[11]: %sql SELECT SUM(PAYLOAD_MASS__KG_) AS TotalPayloadMass FROM SPACEXTBL WHERE Customer LIKE '%NASA (CRS)%';

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

[11]:

TotalPayloadMass
48213

Average Payload Mass by F9 v1.1

The query displays average of payloadmass for records that booster version is F9 v1.1.

Task 4

```
[13]: %sql SELECT AVG(PAYLOAD_MASS__KG_) AS TotalPayloadMass FROM SPACEXTBL WHERE Booster_Version LIKE '%F9 v1.1%';

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

[13]:

TotalPayloadMass
2534.6666666666665

First Successful Ground Landing Date

The query displays the first date of successful ground landing.

Task 5

```
%sql SELECT Landing_Outcome, Date FROM SPACEXTBL WHERE Landing_Outcome LIKE 'Success (ground pad)' ORDER BY Date LIMIT 1
```

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

Landing_Outcome	Date
Success (ground pad)	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

The query displays average of payloadmass for records that booster version is F9 v1.1.

Task 6

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000;
```

```
 * 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

The query displays total number od success and fail outcome for launch.

Task 7

```
%sql SELECT Mission_Outcome, COUNT(*) AS TotalCount FROM SPACEXTBL GROUP BY Mission_Outcome;
```

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

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

2015 Launch Records

The query displays which month in 2015 have fail launch with drone ship.

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

Month_Name	Booster_Version	Launch_Site
January	F9 v1.1 B1012	CCAFS LC-40
April	F9 v1.1 B1015	CCAFS LC-40

```
%sql SELECT CASE substr(Date, 6, 2) WHEN '01' THEN 'January' WHEN '02' THEN 'February' WHEN '03' THEN 'March' WHEN '04' THEN 'April' WHEN '05' THEN 'May' WHEN '06' THEN 'June' WHEN '07' THEN 'July' WHEN '08' THEN 'August' WHEN '09' THEN 'September' WHEN '10' THEN 'October' WHEN '11' THEN 'November' WHEN '12' THEN 'December' ELSE 'Unknown' END AS Month_Name, Booster_Version, Launch_Site FROM SPACEXTBL WHERE Landing_Outcome = 'Failure (drone ship)' AND substr(Date, 1, 4) = '2015';
```

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

The query displays landing outcomes between June 4, 2010, and March 20, 2017.

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

Landing_Outcome	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

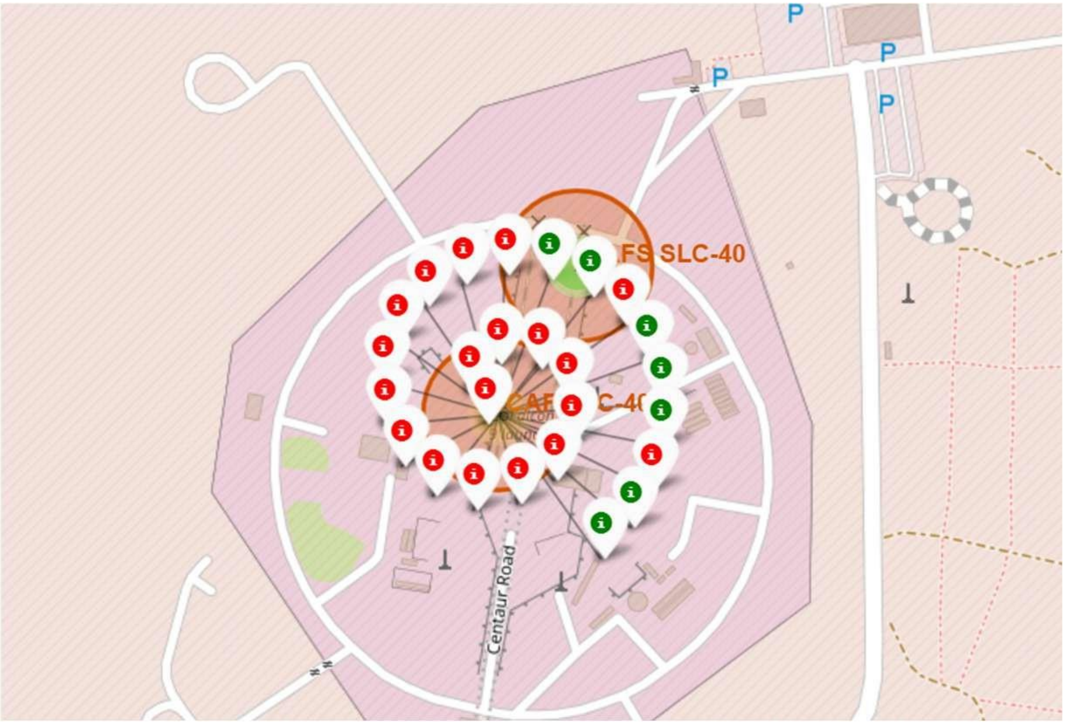
```
%sql SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTBL WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY Outcome_Count DESC;
```

Launch Sites Proximities Analysis

Folium Map –Launch Site

Display the location of each SpaceX launch site on Folium Map.

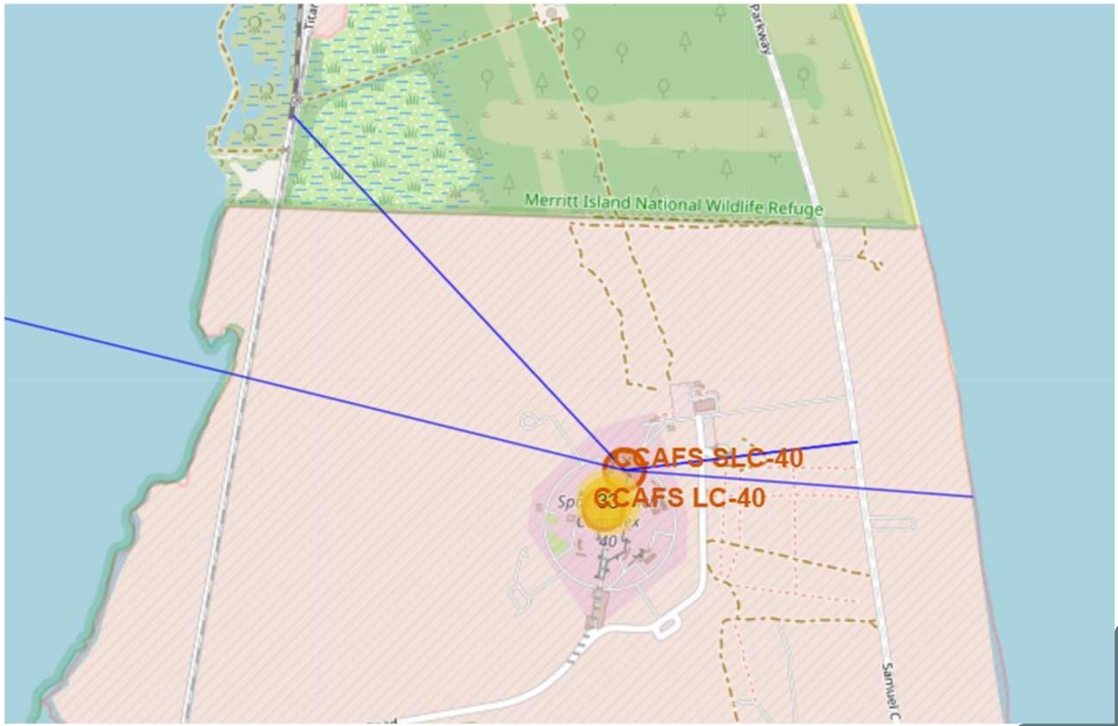
Folium Map –Launch Outcome



Build the map to show the location of the launch site and indicate successful launches in green and unsuccessful launches in red.

Folium Map – Facility Nearby

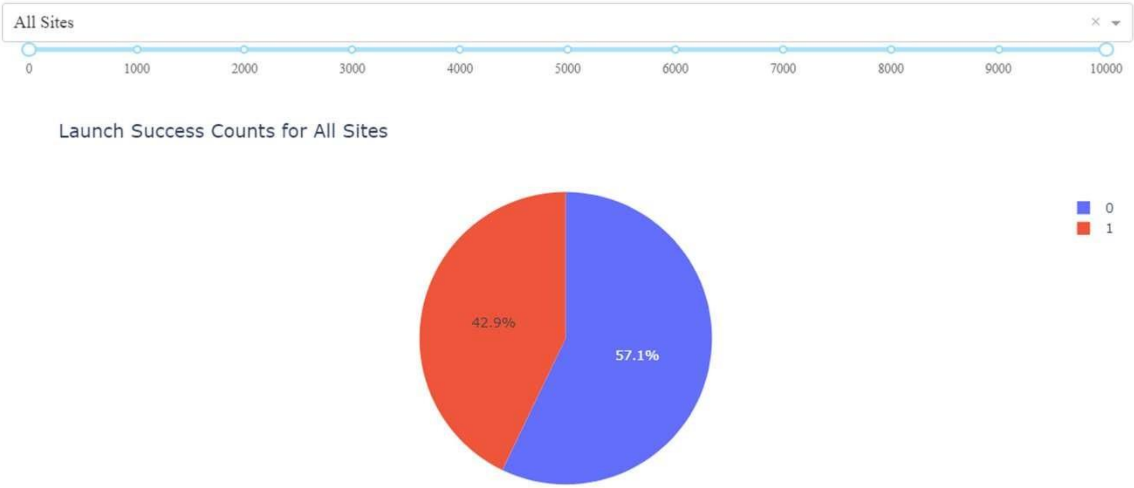
Blue line indicate the distance to important facility such as highway, railway and city.



Build a Dashboard with Plotly Dash

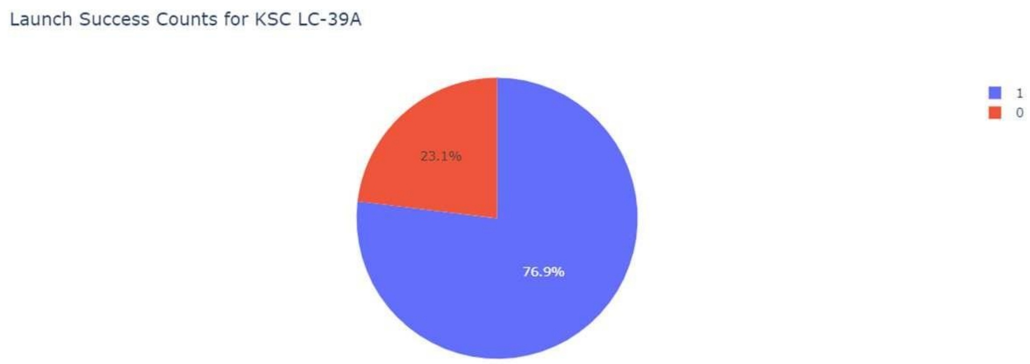
Dashboard –Pie Chart

Build an interactive dashboard to visualize the proportion of successful launches at different launch sites using pie charts.



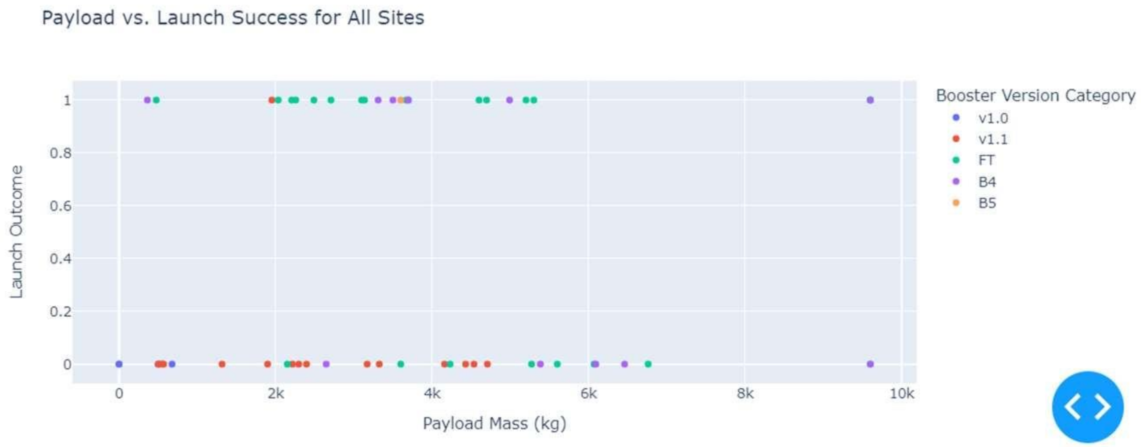
Dashboard –Highest Launch Success Ratio

KSC- LC-39A have the highest launch success ration as 76.9%



Dashboard –Scatter Chart

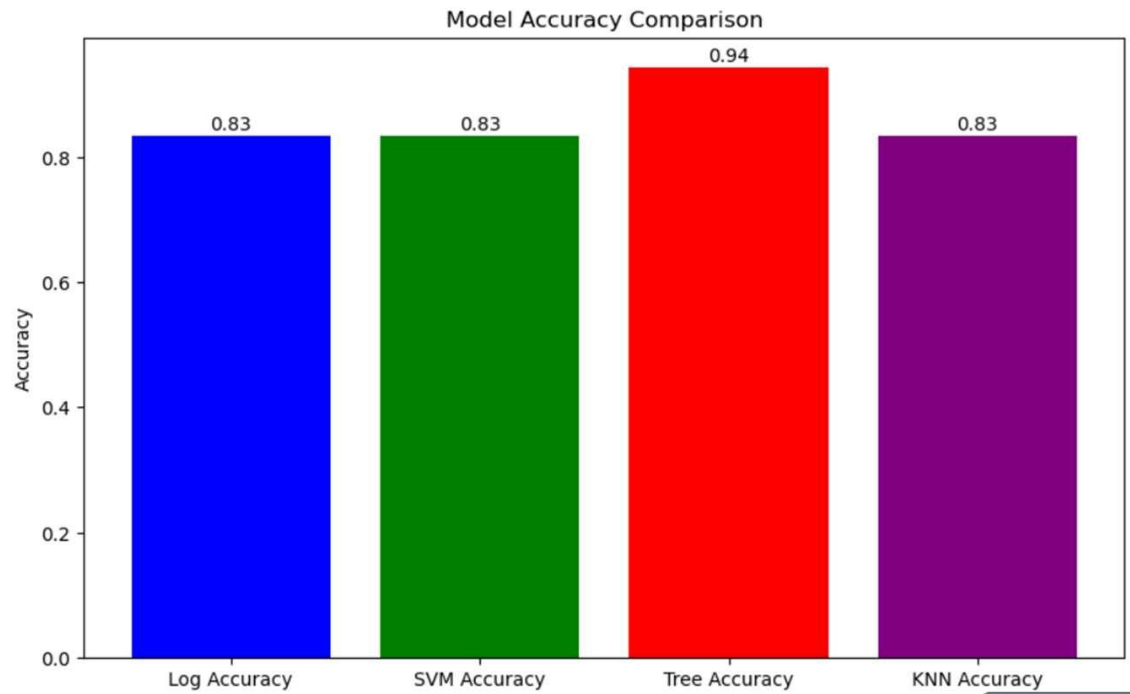
As versions 1.0 and 1.1 are early versions, we can see that the success rate is lower. As improved versions are released, the success rate increases.



Predictive Analysis (Classification)

Classification Accuracy

The most models have 83% accuracy, but the tree model has the highest accuracy. Thus, the tree model is considered the best model.



Confusion Matrix

The confusion matrix display the strong ability of classification.

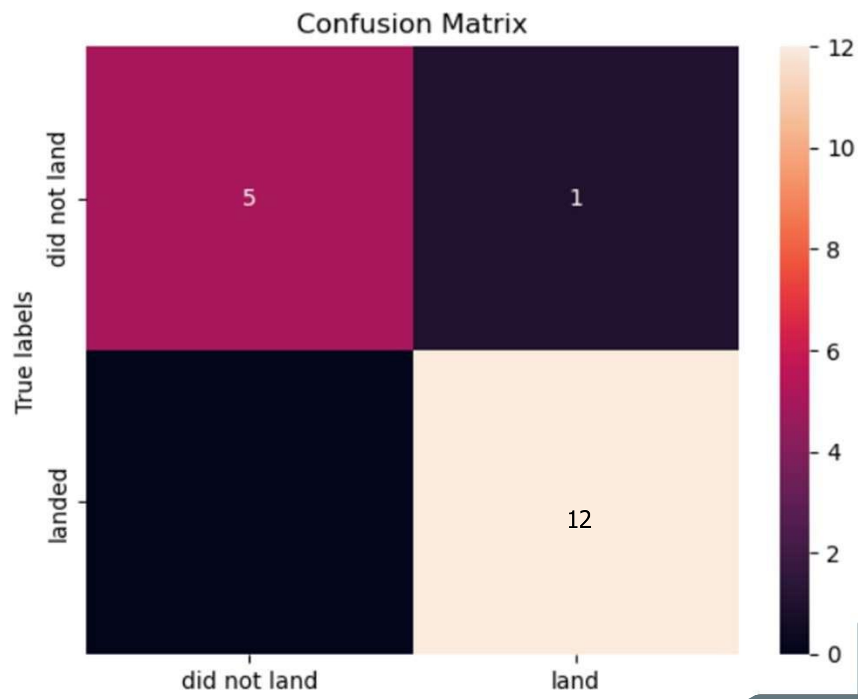
```
cm = confusion_matrix(y_test, yhat_tree)
cm
```

```
array([[ 5,  1],
       [ 0, 12]], dtype=int64)
```

```
[57]: test_accuracy_tree = tree_cv.score(X_test, y_test)
      test_accuracy_tree
```

```
[57]: 0.9444444444444444
```

```
[58]: yhat_tree = tree_cv.predict(X_test)
      plot_confusion_matrix(y_test, yhat_tree)
```



Conclusions

1. By deploying the tree model, we can achieve 94% accuracy in predicting whether the launch will be successful or not.
2. According to the tree model, it is a bit difficult to explain which variables contribute the most.
3. Based on the EDA, we can conclude that as time passes, technology improves, and the success rate increases over the years.
4. With more data, I believe we can increase accuracy and identify more accurate patterns in predicting launch outcomes.

Thank you!