IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

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

Executive Summary

Summary of methodologies

- Data collection by web scraping via API or webscraping.
- Data wrangling and data cleaning.
- Exploratory Data analysis using SQL and Python libraries.
- Data Analytics by Folium and Dash
- Predictive Analytics via Regression, Classification, Clustering.

Summary of all results

- We got to derive conclusive results from the Analytics tools.

Introduction

- This Applied data science Capstone project for Space X Falcon 9 rockets aims to predict the successful outcomes of the rocket launches for the first stage using classification models.
- The project aims to predict the reusability of the first stage of the rocket which would enable us to derive the cost of the launch.
- Space X is a organization which deals in sending rockets to outer space for space exploration purpose.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describes how data was collected using Space X API and web scraping from Wikipedia website
- Perform data wrangling
  - Describes how the data was processed by finding and replacing missing values and also converting columns with categorical values to numerical
- Perform exploratory data analysis (EDA) using Matplotlib visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- he flow chart in Figure 1 shows the data collection process using Python's library requests and Space X API.
- The final data frame is saved as a CSV file using df.to_csv()
- The URL below is the GIT repository containing the Jupyter notebook

https://github.com/tridib87/python-capstone-project

Data Collection – SpaceX API

- The flow chart in Figure 1 shows the data collection process using Python's library requests and Space X API.
- The final data frame is saved as a CSV file using df.to_csv()

- https://github.com/tridib87/python-capstone-project

Place your flowchart of SpaceX API calls here

Data Collection - Scraping

- The flow chart in Figure 2 shows the data collection process using Python's library requests and Beautiful Soup
- The final data frame is saved as a CSV file using data_falcon9.to_csv()
- The URL below is the GIT repository containing the Jupyter notebook

Place your flowchart of web scraping here

Data Wrangling

- The following stage involves conducting some Exploratory Data Analysis on the data
- Also, converting the outcomes/class into training labels with 1 meaning successful landed and 0 meaning failure to land.

- https://github.com/tridib87/python-capstone-project

EDA with Data Visualization

- In the exploratory data analysis using data visualization, we use scatter plots to visualize the relationship vis a vis Payload mass vs flight number, flight number vs launch site, and payload vs launch site.
- We also use a bar chart to visualize the success rate of each orbit type.
- Lastly we use a line plot to visualize the yearly trend of the number of successful launch.
- The github link is https://github.com/tridib87/python-capstone-project

EDA with SQL

Following findings were done by executing sql queries on python

- Display names of the booster versions that have carried the maximum payload mass
- Display month names, failure landing outcomes in drone ship, booster versions, and launch sites for the months in the year 2015.
- Rank the count of successful landing outcomes between the dates 04-06-2010 and 20-03-2017 in descending order.
- The git hub link is The github link is https://github.com/tridib87/python-capstone-project

Build an Interactive Map with Folium

- folium.Circle was used to add a highlighted circle area as an initial centre location
- folium.map.Markerwas used to create a marker at a specific launch location on the map
- MarkerCluster() was used to create cluster markers of successful and failed launches for a particular site
- MousePosition() was used to display the latitude and longitude coordinates of the mouse cursor's position on a map. Used to calculate the distance of the launch sites to the coasts.
- Folium.PolyLine() was used to create a series of connected line segments on the map to mark the distance of the launch sites to the coast, railways, highways, and major cities
- Git hub link is https://github.com/tridib87/python-capstone-project

Build a Dashboard with Plotly Dash

- The plots and graphs added to the dashboard include:
- Drop-down input containing all the launch sites.
- A rendered pie chart showing the success rates based on the launch site input selected.
- A range slider to select the payload mass.
- A scatter plot showing the correlations between the payload mass and the success for the launch sites selected.
- Git hub link is https://github.com/tridib87/python-capstone-project

Predictive Analysis (Classification)

- The data was loaded and split into features and target.
- The features columns were normalized and the target column was converted to a NumPy array
- Data was split into training and test set
- GridsearchCV was used on all classification algorithms which helped in determining the best parameters and best scores using the .best_params_ and .best_score_ respectively
- Accuracy of the test set was also calculated using .score() method.
- Confusion matrix was also derived accordingly.
- Github link is https://github.com/tridib87/python-capstone-project

Results

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

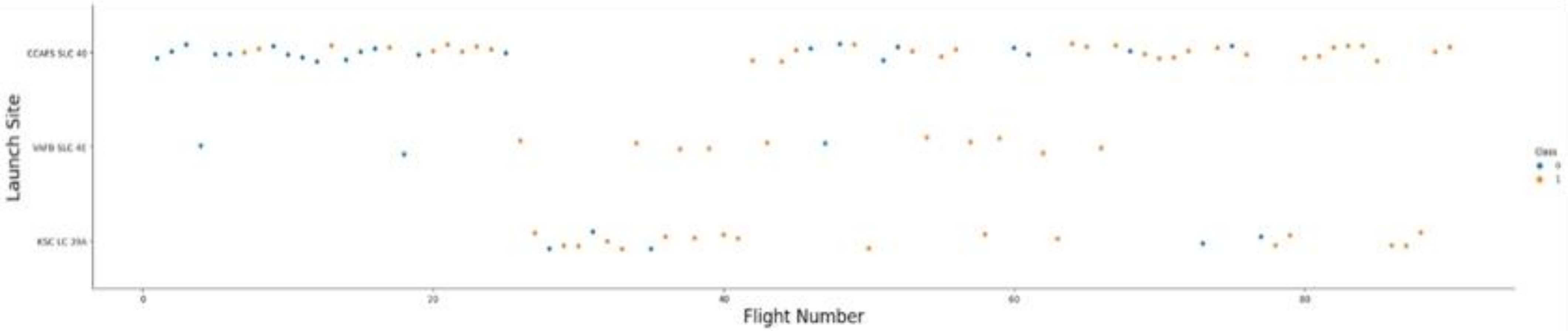
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

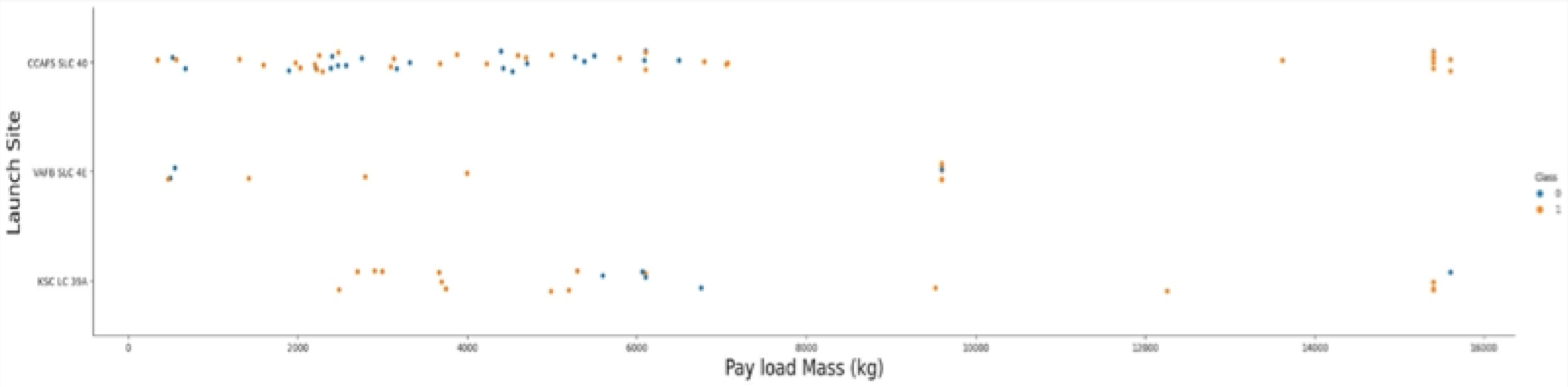
We can infer that the launch site CCAFS SLC 40 has launched the highest number of rockets compared to the other sites.

We can also find that the flights from launch sites VAFB SLC 4E and KSC LC 39A showed a higher success rate compared to the earlier flights.



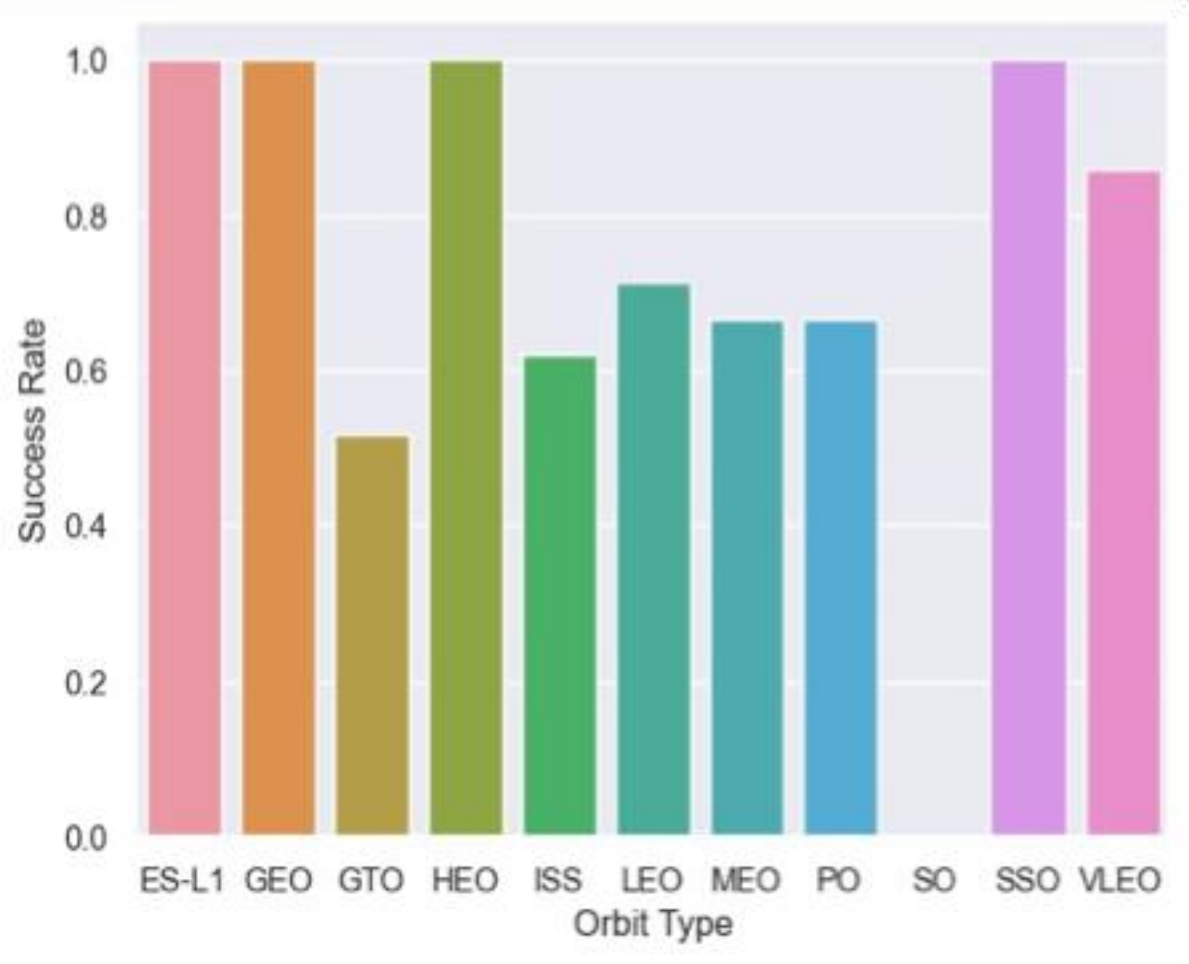
Payload vs. Launch Site

• We observe that VAFB-SLC 4E launch site has no rockets launched for heavy payload mass greater than 10000kg.
• We also observe that most of the rockets launched in all launch sites have a payload mass of less than 9000kg.
•Compared to VAFB-SLC 4E and KSC LC 39A, CCAFS SLC 40 has a higher success rate for with heavier payloads of 14000 kg and 16000 kg.



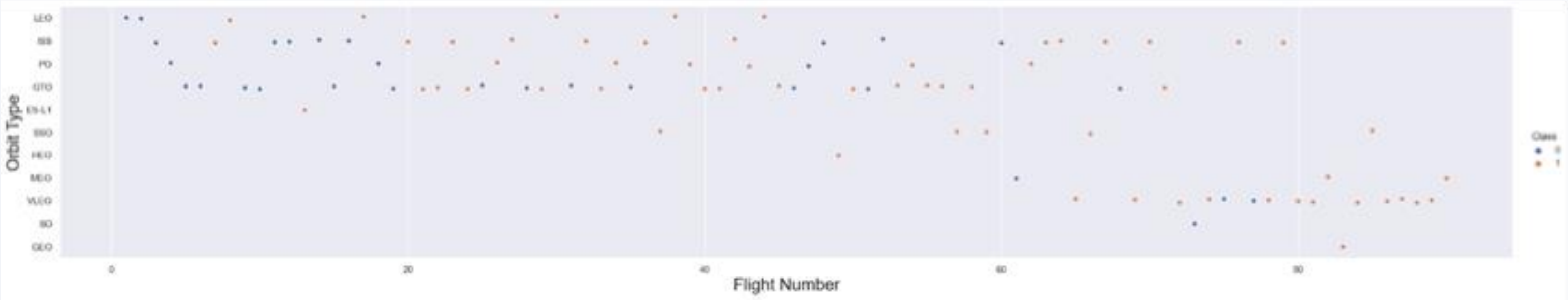
Success Rate vs. Orbit Type

- From the figure, it is observed that orbits VLEO, ES-L1, GEO, HEO, and SSO have the highest success rates compared to the other orbit types
- It is also observed that orbit SO has the least success rate



Flight Number vs. Orbit Type

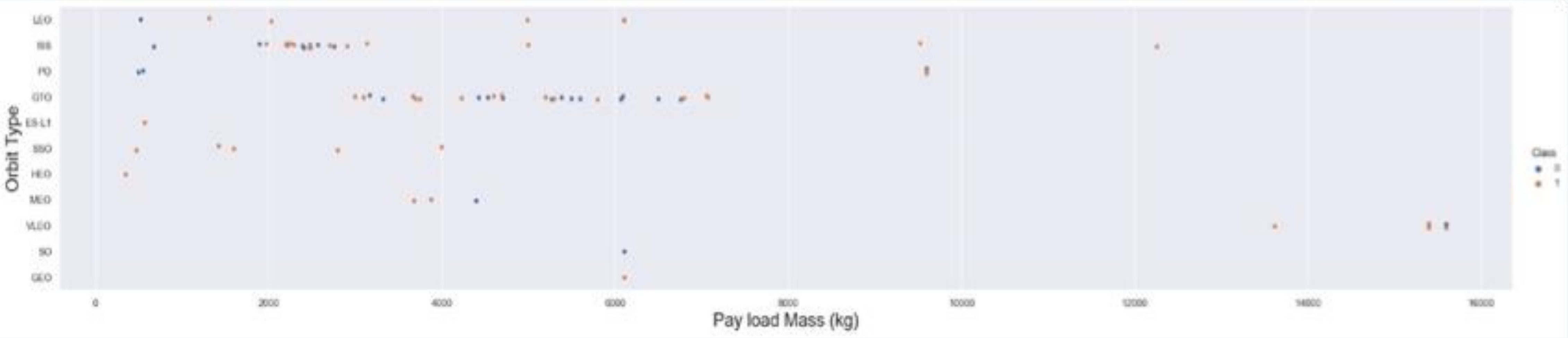
- From the figure we can observe that more rockets were launched in LES ISS PO GTO and VLEO



Scatter Plot Flight no v/s Orbit Type

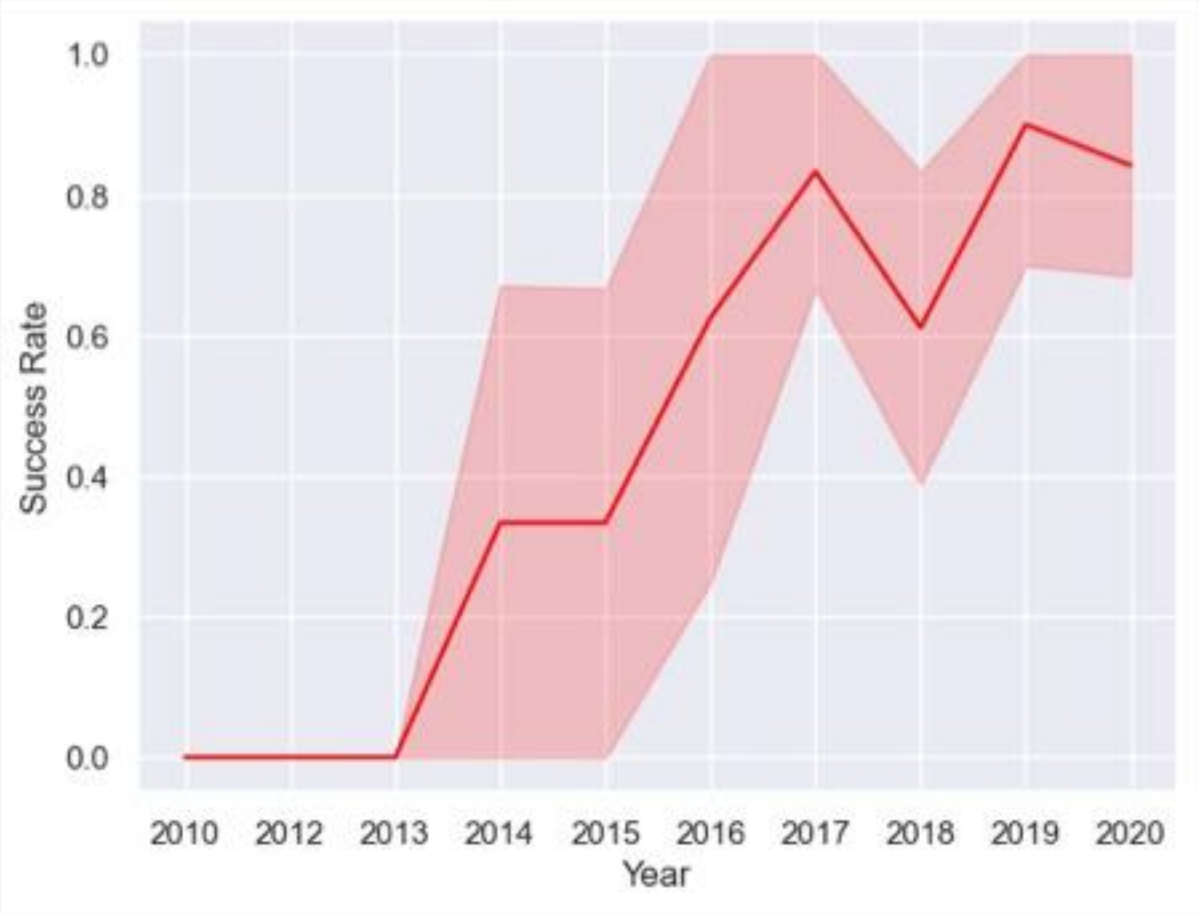
Payload vs. Orbit Type

- There is a higher success for rockets with heavy payloads launched in PO, LEO, and ISS.
- Rockets launched in SSO and MEO orbits on the other hand have a high success rate for lighter payloads.
- Rockets launched in GTO have both positive landing rates and negative landing rates regardless of the size of the payload



Launch Success Yearly Trend

- We observe in that the success rate since 2013 kept increasing till 2020



All Launch Site Names

- An SQL table called SPACEXTBL using the existing data frame.
- To find the Unique Launch Sites, the keyword DISTINCT was used on the column.

```
%%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
None

Launch Site Names Begin with 'CCA'

- Keyword LIKE `CCA%` was used to get launch site names beginning with `CCA`.
- LIMIT 5 keyword was used to display the first 5 records

```
%%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
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- SUM function was used to calculate the total payload mass of customers with the name ‘NASA (CRS)

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) AS Total_payload_NASA_CRS
FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)'
```

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

Total_payload_NASA_CRS
45596.0

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1 was calculated using the AVG function

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) AS Average_Payload_F9V1_1
FROM SPACEXTBL
WHERE Booster_Version LIKE 'F9 v1.1%'
```

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

Average_Payload_F9V1_1
2534.6666666666665

First Successful Ground Landing Date

- SQL query was run for the first successful landing on ground pad.
- The result shows that 22nd of December 2015 was the date for the first successful ground landing

```
%%sql

SELECT Date FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)'
ORDER BY Date DESC
LIMIT 1
```

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

Date
22/12/2015

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using keywords BETWEEN and AND, the names of boosters that have successfully landed on drone ship and had payload mass greater than 4000kg but less than 6000kg were displayed.
- The result shows 4 rockets.

```
%%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

Total Number of Successful and Failure Mission Outcomes

- The COUNT function was used to could the total number of successful missions and failed missions
- The results show that there were 100 successful missions and 1 failed mission.

```
%%sql
SELECT COUNT(Mission_Outcome)
AS Success_missions
FROM SPACEXTBL
WHERE Mission_Outcome LIKE '%Success%'
```

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

Success_missions
100

```
%%sql
SELECT COUNT(Mission_Outcome)
AS Failure_missions
FROM SPACEXTBL
WHERE Mission_Outcome LIKE '%Failure%'
```

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

Failure_missions
1

Boosters Carried Maximum Payload

- A sub-query with the MAX function was used to retrieve the boosters that carried the maximum payload.
- Results show that there are 12 in total

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

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

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600.0
F9 B5 B1049.4	15600.0
F9 B5 B1051.3	15600.0
F9 B5 B1056.4	15600.0
F9 B5 B1048.5	15600.0
F9 B5 B1051.4	15600.0
F9 B5 B1049.5	15600.0
F9 B5 B1060.2	15600.0
F9 B5 B1058.3	15600.0
F9 B5 B1051.6	15600.0
F9 B5 B1060.3	15600.0
F9 B5 B1049.7	15600.0

2015 Launch Records

- Substr() was used to extract the month and year from the Date column.
- The WHERE and AND keyword was used to get launch records of failed drone ship landings in 2015.
- The result shows that the failed landings occurred in the months of April (04) and Octobe

```
%%sql

SELECT substr(Date, 4, 2) as Month, Booster_Version, Landing_Outcome, Launch_Site
FROM SPACEXTBL
WHERE Landing_Outcome = 'Failure (drone ship)' AND substr(Date, 7, 4) = '2015'
```

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

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

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

- Keywords such as GROUP BY, ORDER BY, and DESC as well as functions like substr() and COUNT() were used to rank the count landing outcomes between 2010-06-04 and 2017-03-20, in descending order.
- The results show that there are high numbers for no attempt (10), success on drone ship (5) as well as ground (5)

```
%%sql
SELECT Landing_Outcome, COUNT(Landing_Outcome) AS Number
FROM SPACEXTBL
WHERE substr(Date,7)||substr(Date,4,2)||substr(Date,1,2)
BETWEEN '20100604' and '20170320'
GROUP BY Landing_Outcome
ORDER BY Number
DESC
```

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

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

Section 3

Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

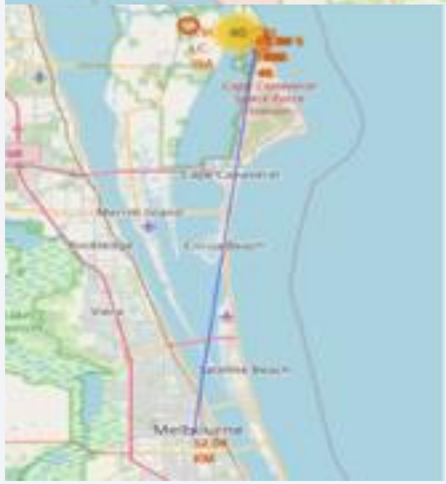
- • All launch sites as shown in the figure are located in coastal cities of the United States of America.

<Folium Map Screenshot 2>

- The figure shows the launch outcomes for various launch sites;
- Top left: VAFB SLC-4E
- Top right: KSC LC-39A
- Bottom left: CCAFS SLC-40
- Bottom Right: CCAFS LC-4

<Folium Map Screenshot 3>

- From the figure we can show that launch sites are located very close to the coast i.e. 0.95km from CCAFS SLC 40 and 1.52km from VAFB SLC 4E
- The same cannot be said for some railways and highways
- We can also infer that launch sites are located far from major cities, i.e. VAFB SLC 4E is 38.16km away from its closest city Santa Maria and CCAFS SLC 40 is 56.04km away from Melbourne

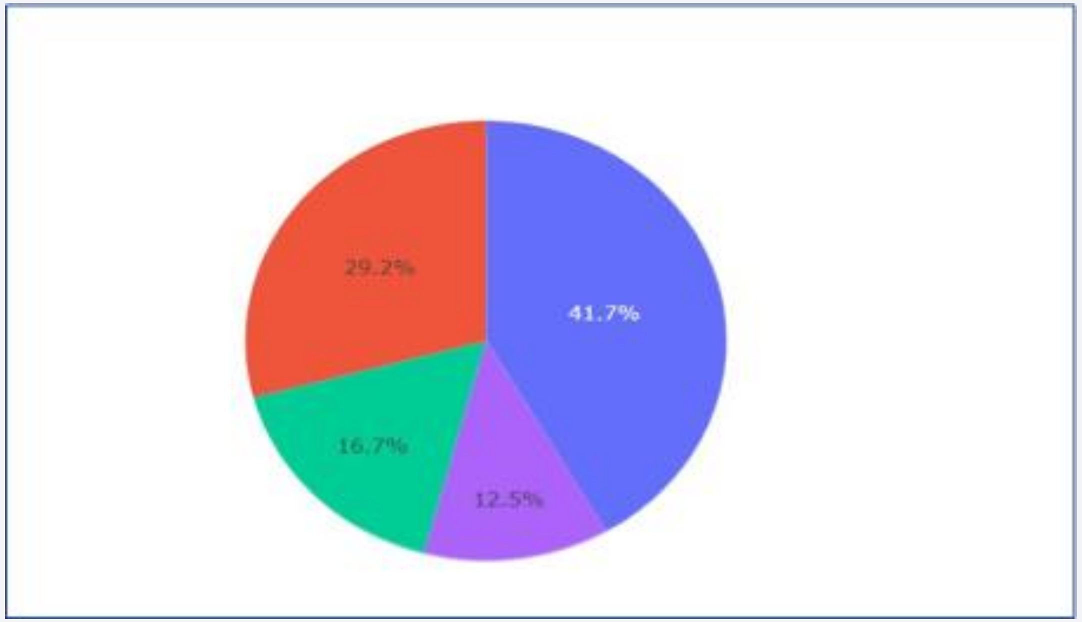


Section 4

Build a Dashboard with Plotly Dash

Pie Chart of Launch Sites v/s Success Rates

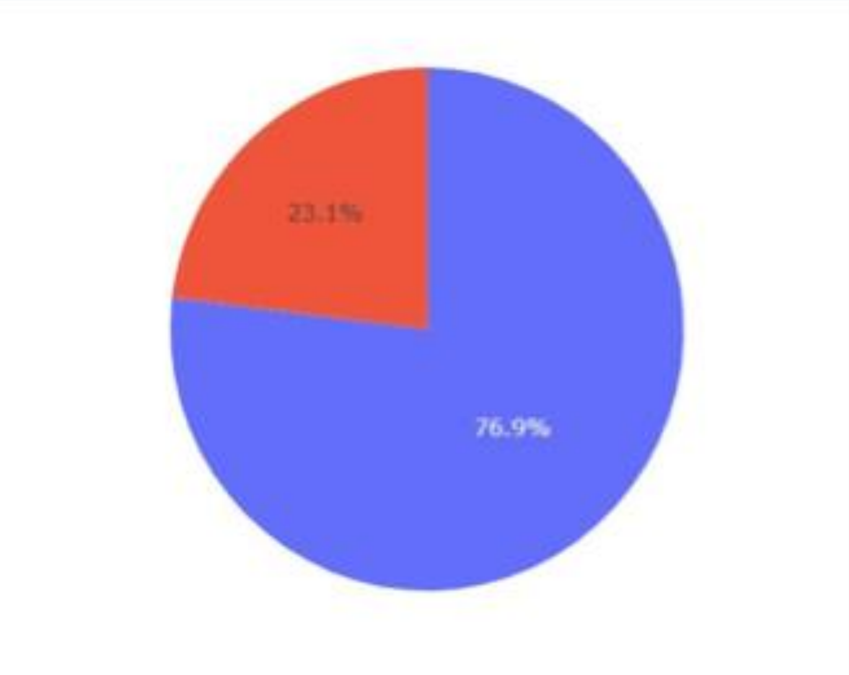
- From Figure, it can be inferred that KSC LC-39A has the largest success rate with about 41.7% of the total success ratio with other sites.



Pie chart showing the Success rate of all Launch sites

Pie Chart of Launch Sites with highest success ratio

- It is also evident from Figure 25 that KSC LC-39A has the highest success ratio with about 76.9%, compared to the other sites;
- • 73.1% for CCAFS LC-40
  • 60% for VAFB SLC-4E
  • 57.1% for CCAFS SLC-40



Pie Chart showing the launch site with the highest success ratio

<Scatter Plot Payload vs Launch Outcome>

- From the figures below, Booster version FT has the highest success rate with its payload mass of about between 700kg to 5,500kg.
- It is also shown that rockets with payload mass above 5,500kg have a lower success rate, which means the heavier the payload, the slimmer the chance of a successful outcome.

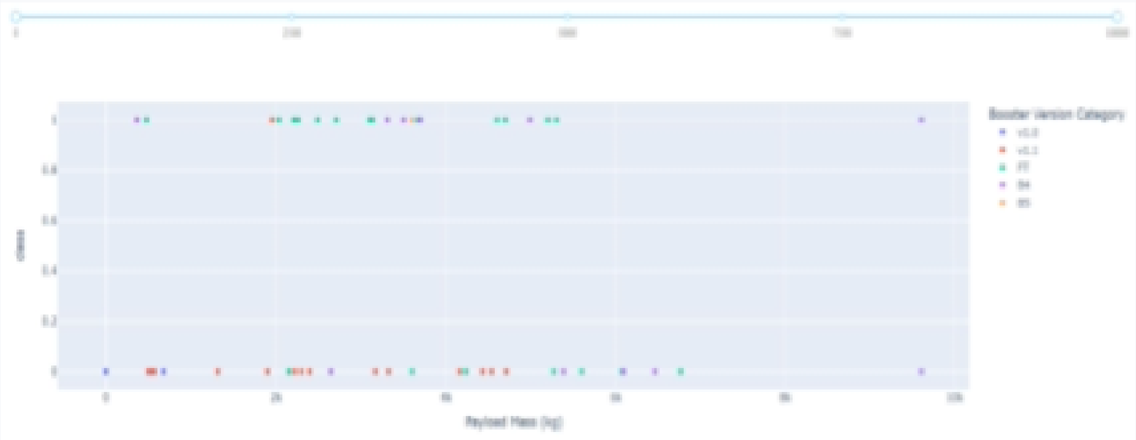
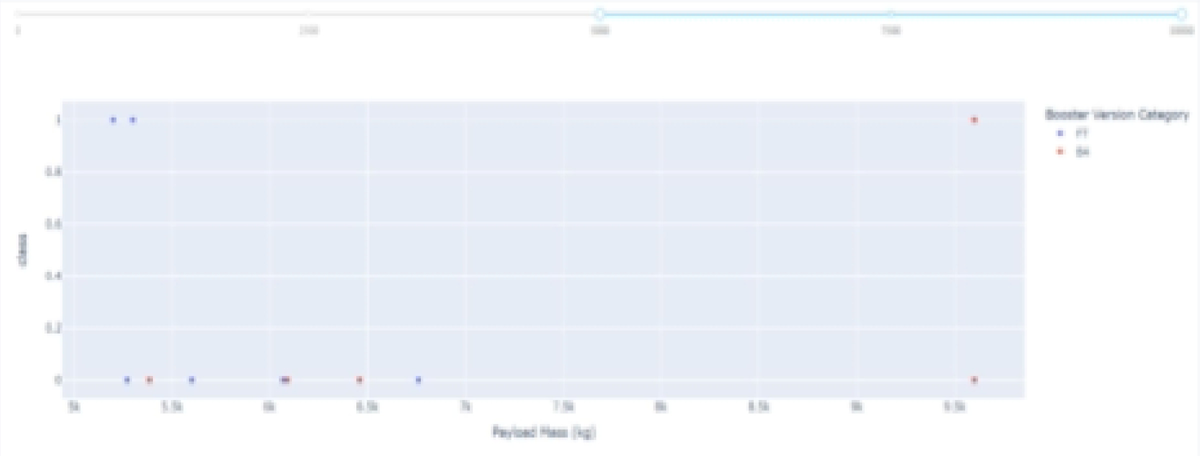


Figure showing outcome booster versions with varying payload mass from launch sites



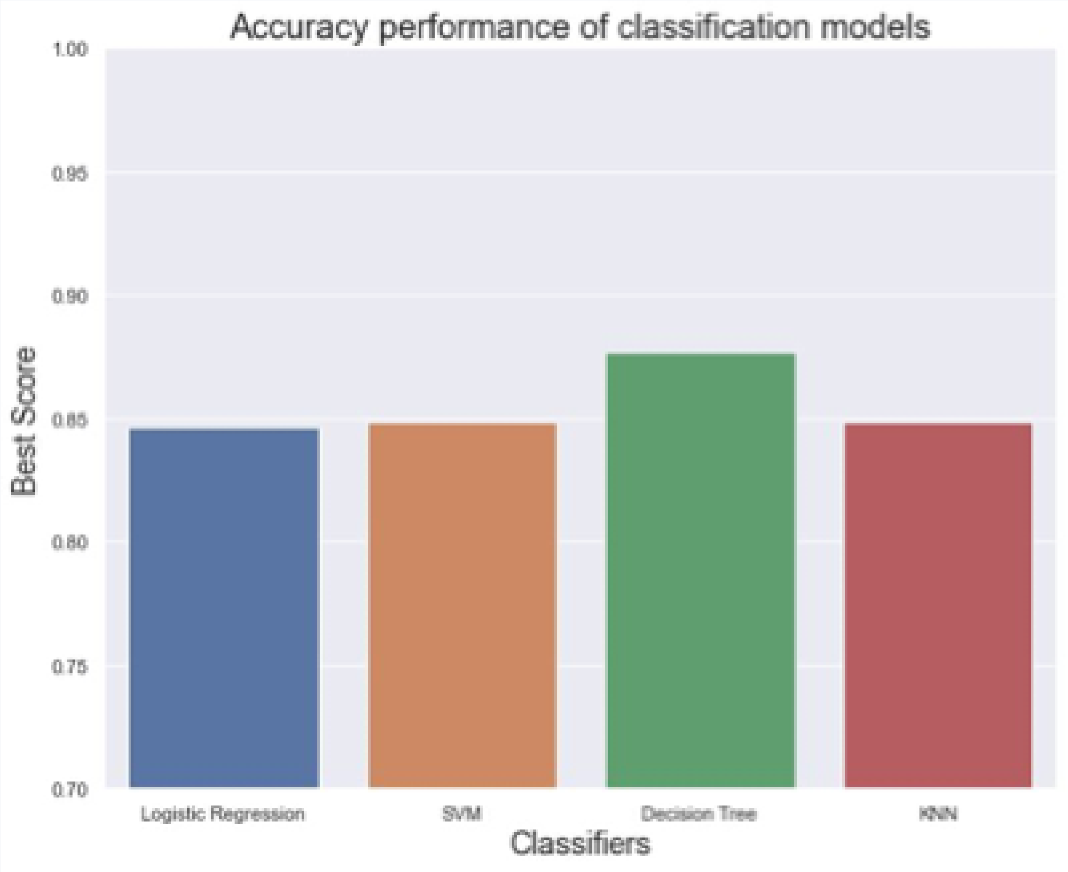
Figures showing outcome of booster versions with payload mass greater than 5500 kg

Section 5

Predictive Analysis (Classification)

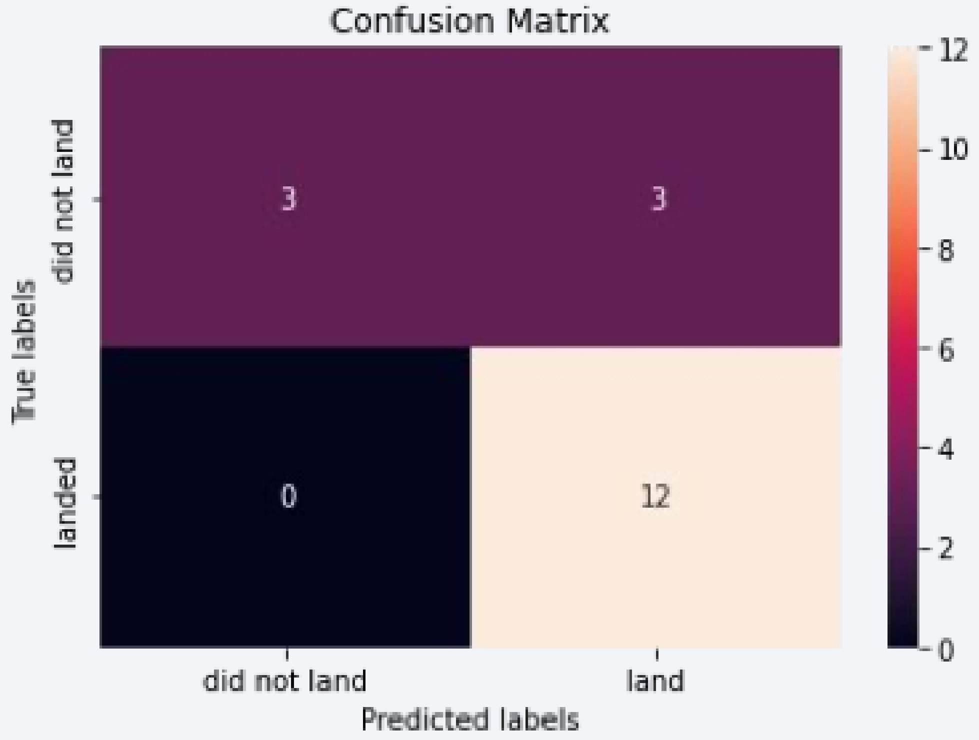
Classification Accuracy

- From the bar chart in Figure 28, Decision tree classifier performed the best with an accuracy score of approximately 0.875 or ~ 87%



Confusion Matrix

- After the dataset was spilt into training and test set, we ended up with only 18 test samples.
- From the test set, the decision tree classifier was able to correctly predict for 12 observations that the rocket landed (12 True positives) and also 3 observations when the rocket did not landed (3 True Negatives).
- The classifier also had 0 false negatives.
- However, it had 3 false positives as it predicted wrongly for 3 observations that the that the rockets will land although they didn't



Confusion matrix of decision tree

Conclusions

- Orbit type should be considered because rockets launched to certain orbits (VLEO, ES-L1, GEO, HEO, and SSO) had higher success rates compared to others.
- We can conclude that Launch sites are located in coastal cities for easy retrieval/recovery and are located far from busy areas like major highways and cities to minimize casualties in the event of a failure.
- In recent years the no. of successful launches have increased.
- Decision tree classifiers performed the best with approximately 87% making them a good model for landing outcome prediction

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

Thank you!