IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ashutosh Singh Thakur
04/11/2023

OUTLINE

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

Executive Summary

Summary of Methodologies

The research attempts to identify the factors for a successful rocket landing. To make this determination, the following methodologies where used:

- **Collect data** using SpaceX REST API and web scraping techniques
- **Wrangle data** to create success/fail outcome variable
- **Explore data** with data visualization techniques, considering the following factors: payload, launch site, flight number and yearly trend
- **Analyze** the data with SQL, calculating the following statistics: total payload, payload range for successful launches, and total # of successful and failed outcomes
- **Explore** launch site success rates and proximity to geographical markers
- **Visualize** the launch sites with the most success and successful payload ranges
- **Build Models** to predict landing outcomes using logistic regression, support vector machine (SVM), decision tree and K -nearest neighbor (KNN)

Executive Summary

Summary of all results

Exploratory Data Analysis:

- Launch success has improved over time
- KSC LC-39A has the highest success rate among landing sites
- Orbits ES -L1, GEO, HEO, and SSO have a 100% success rate

Visualization/Analytics:

- Most launch sites are near the equator, and all are close to the coast

Predictive Analytics:

- All models performed similarly on the test set. The decision tree model slightly outperformed

Introduction

Background

SpaceX, a leader in the space industry, strives to make space travel affordable for everyone. Its accomplishments include sending spacecraft to the international space station, launching a satellite constellation that provides internet access and sending manned missions to space. SpaceX can do this because the rocket launches are relatively inexpensive ($62 million per launch) due to its novel reuse of the first stage of its Falcon 9 rocket. Other providers, which are not able to reuse the first stage, cost upwards of $165 million each. By determining if the first stage will land, we can determine the price of the launch. To do this, we can use public data and machine learning models to predict whether SpaceX – or a competing company – can reuse the first stage.

Explore

- How payload mass, launch site, number of flights, and orbits affect first-stage landing success
- Rate of successful landings over time
- Best predictive model for successful landing (binary classification)

Section 1

Methodology

METHODOLOGY

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scraping from Wikipedia.
- Perform data wrangling
  - One-hot encoding was applied to categorical features
- Perform exploratory data analysis (EDA) using 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

- **Collect** data using SpaceX REST API and web scraping techniques
- **Wrangle** data – by filtering the data, handling missing values and applying one hot encoding – to prepare the data for analysis and modelling
- **Explore** data via EDA with SQL and data visualization techniques
- **Visualize** the data using Folium and Plotly Dash
- **Build Models** to predict landing outcomes using classification models. Tune and evaluate models to find best model and parameters

Data Collection – SpaceX API

- **Request data** from SpaceX API (rocket launch data)
- **Decode response** using .json() and convert to a dataframe using .json_normalize()
- **Request information** about the launches from SpaceX API using custom functions
- **Create dictionary** from the data
- **Create dataframe** from the dictionary
- **Filter dataframe** to contain only Falcon 9 launches
- **Replace missing values** of Payload Mass with calculated .mean()
- **Export data** to csv file

1. Get request for rocket launch data using API

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

2. Use json_normalize method to convert json result to dataframe

In [12]:
```
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

In [30]:
```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```

The link to the notebook is
. https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-spacex-data-collection-api.ipynb

DATA COLLECTION - SCRAPING

- **Request data** (Falcon 9 launch data) from Wikipedia
- **Create BeautifulSoup object** from HTML response
- **Extract column names** from HTML table header
- **Collect data** from parsing HTML tables
- **Create dictionary** from the data
- **Create dataframe** from the dictionary
- **Export data** to csv file

1. Apply HTTP Get method to request the Falcon 9 rocket launch page

```
In [4]: static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

```
In [5]: # use requests.get() method with the provided static_url
        # assign the response to a object
        html_data = requests.get(static_url)
        html_data.status_code

Out[5]: 200
```

2. Create a BeautifulSoup object from the HTML response

```
In [6]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content
        soup = BeautifulSoup(html_data.text, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
In [7]: # Use soup.title attribute
        soup.title

Out[7]: <title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

3. Extract all column names from the HTML table header

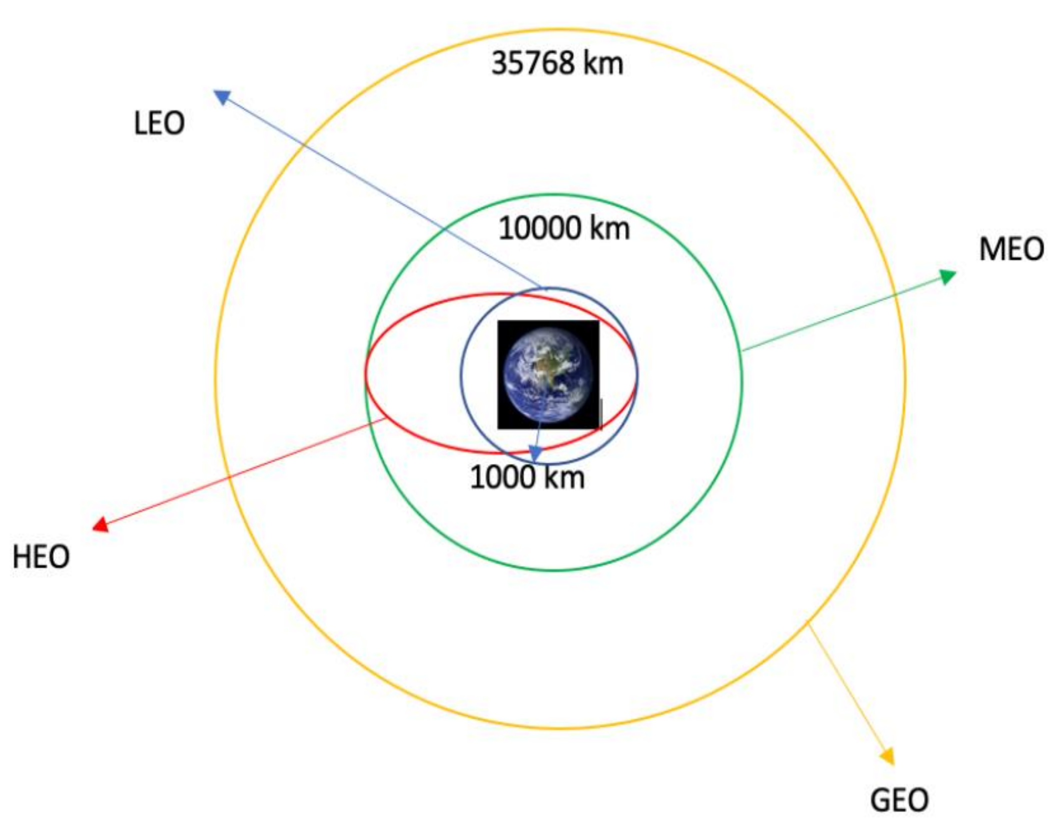
```
In [10]: column_names = []

         # Apply find_all() function with `th` element on first_launch_table
         # Iterate each th element and apply the provided extract_column_from_header() to get a column name
         # Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names

         element = soup.find_all('th')
         for row in range(len(element)):
             try:
                 name = extract_column_from_header(element[row])
                 if (name is not None and len(name) > 0):
                     column_names.append(name)
             except:
                 pass
```

4. Create a dataframe by parsing the launch HTML tables
5. Export data to csv

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-webscraping.ipynb

DATA WRANGLING

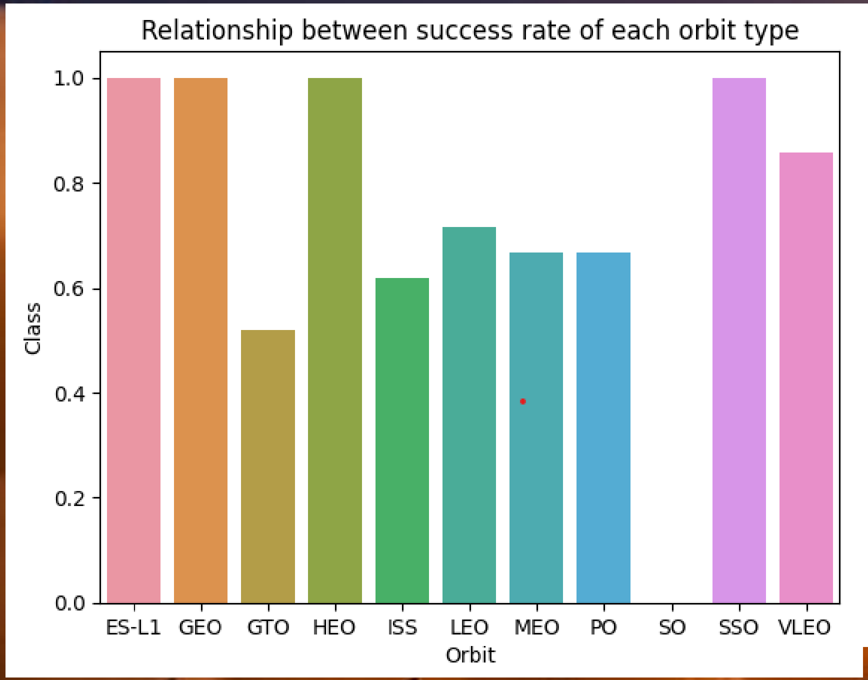
- **Perform EDA and determine data labels**
- **Calculate:**
  - • # of launches for each site
  - • # and occurrence of orbit
  - • # and occurrence of mission outcome per orbit type
- **Create binary landing outcome column (dependent variable)**
- **Export result to csv file**

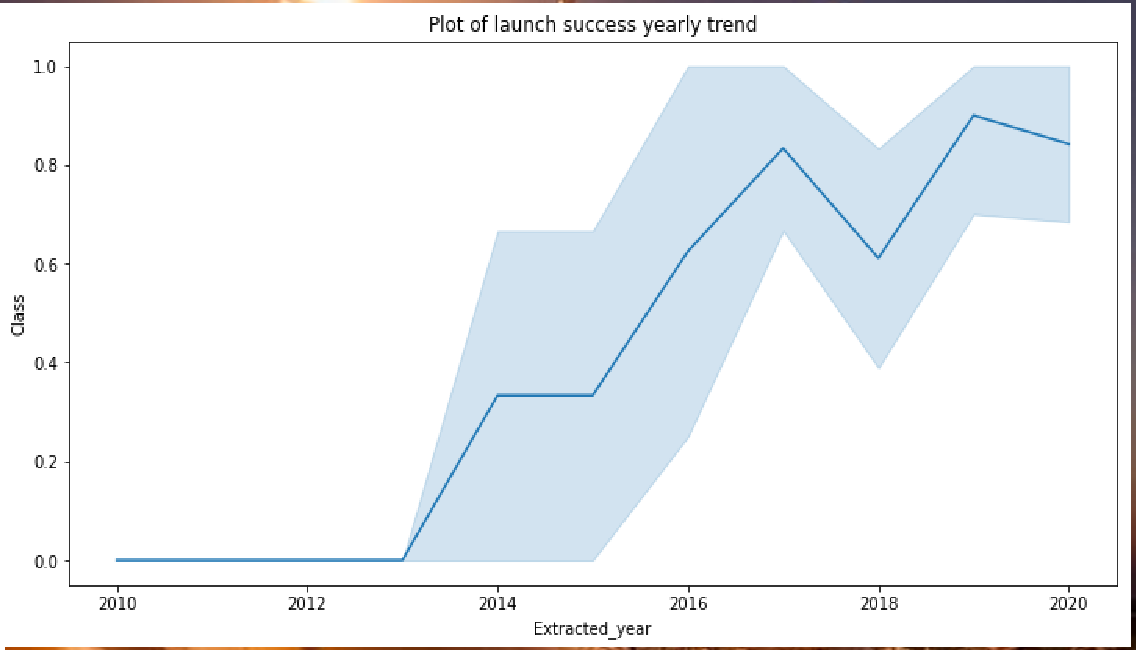


The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- We explored the data by visualizing the relationship between flight number and launch Site, payload and launch site, success rate of each orbit type, flight number and orbit type, the launch success yearly trend.





The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

EDA WITH SQL

Queries

Display:

- Names of unique launch sites
- 5 records where launch site begins with ‘CCA’
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1.

List:

- Date of first successful landing on ground pad
- Names of boosters which had success landing on drone ship and have payload mass greater than 4,000 but less than 6,000
- Total number of successful and failed missions
- Names of booster versions which have carried the max payload
- Failed landing outcomes on drone ship, their booster version and launch site for the months in the year 2015
- Count of landing outcomes between 2010-06-04 and 2017-03-20 (desc)

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

BUILD AN INTERACTIVE MAP WITH FOLIUM

Markers Indicating Launch Sites

- Added blue **circle** at **NASA Johnson Space Center's** coordinate with a **popup label** showing its name using its latitude and longitude coordinates
- Added red **circles** at **all launch sites coordinates** with a **popup label** showing its name using its name using its latitude and longitude coordinates

Colored Markers of Launch Outcomes

- Added **colored markers** of **successful** (green) and **unsuccessful** (red) **launches** at each launch site to show which launch sites have high success rates

Distances Between a Launch Site to Proximities

- Added **colored lines** to **show distance between** launch site **CCAFS SLC40** and its proximity to the **nearest coastline, railway, highway, and city**

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/lab_jupyter_launch_site_location.ipynb

BUILD A DASHBOARD WITH PLOTLY DASH

Dropdown List with Launch Sites

- Allow user to select all launch sites or a certain launch site

Pie Chart Showing Successful Launches

- Allow user to select payload mass range

Slider of Payload Mass Range

- Allow user to see successful and unsuccessful launches as a percent of the total

Scatter Chart Showing Payload Mass vs. Success Rate by Booster Version

- Allow user to see the correlation between Payload and Launch Success

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/Spacex_dash.py

PREDICTIVE ANALYSIS (CLASSIFICATION)

- We loaded the data using **numpy** and **pandas**, transformed the data, split our data into **training** and **testing**.
- We built different machine learning models and tune different hyperparameters using **GridSearchCV**.
- We used **accuracy** as the metric for our model, improved the model using **feature engineering** and **algorithm tuning**.
- We found the **best** performing classification model.

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

RESULTS

Exploratory Data Analysis

- Launch success has improved over time
- KSC LC-39A has the highest success rate among landing sites
- Orbits ES-L1, GEO, HEO and SSO have a 100% success rate

Visual Analytics

- Most launch sites are near the equator, and all are close to the coast
- Launch sites are far enough away from anything a failed launch can damage (city, highway, railway), while still close enough to bring people and material to support launch activities

Predictive Analytics

- Decision Tree model is the best predictive model for the dataset

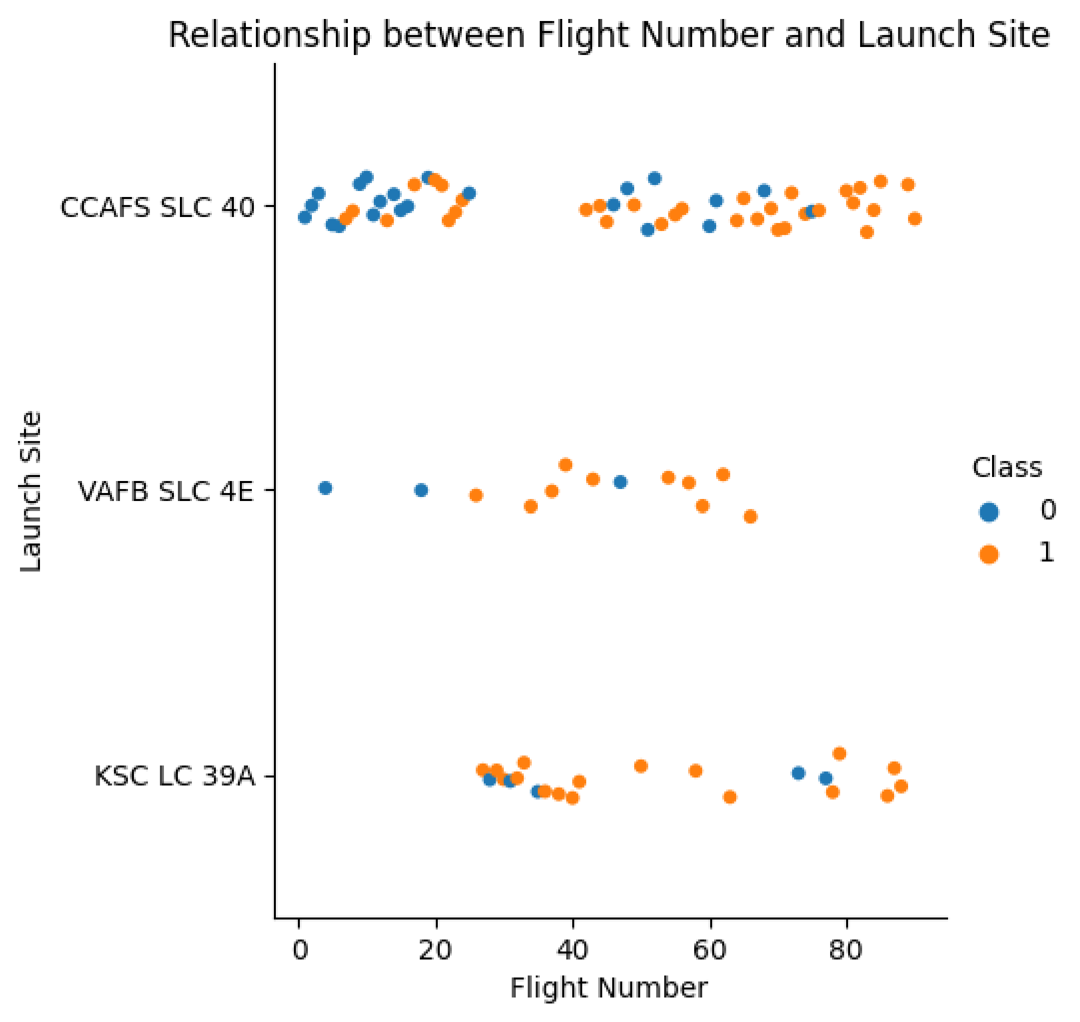
Section 2

Insights drawn from EDA

FLIGHT NUMBER VS. LAUNCH SITE

Exploratory Data Analysis

- **Earlier flights** had a **lower success rate** (**blue** = fail)
- **Later flights** had **a higher success rate** (**orange** = success)
- Around half of launches were from CCAFS SLC 40 launch site
- VAFB SLC 4E and KSC LC 39A have higher success rates
- We can infer that new launches have a higher success rate

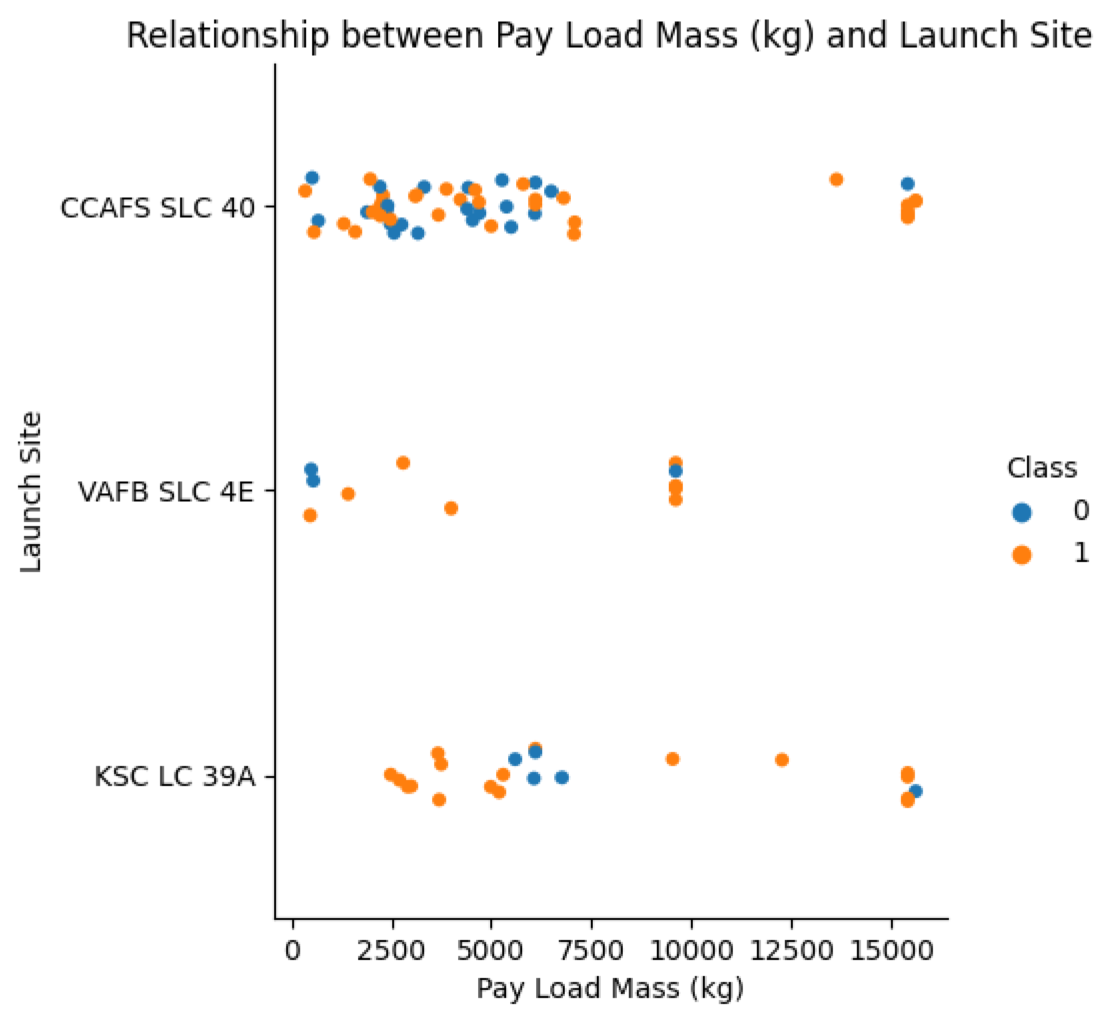


The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

PAYLOAD VS. LAUNCH SITE

Exploratory Data Analysis

- Typically, the higher the payload mass (kg), the higher the success rate
- Most launces with a payload greater than 7,000 kg were successful
- KSC LC 39A has a 100% success rate for launches less than 5,500 kg
- VAFB SKC 4E has not launched anything greater than ~10,000 kg

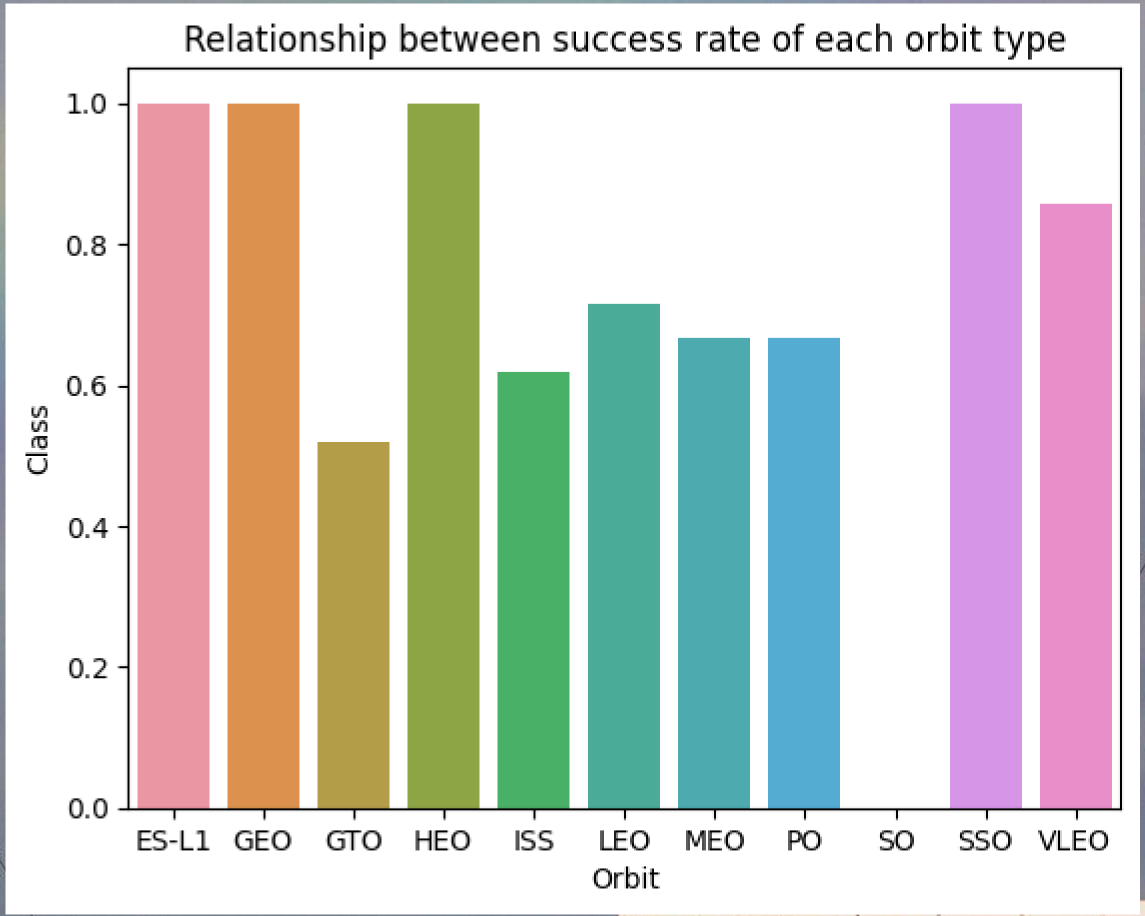


The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

SUCCESS RATE VS. ORBIT TYPE

Exploratory Data Analysis

- From the plot, we can see that ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- We can also see that LEO, ISS, MEO, and PO had higher success rate.
- We can also see that SO had no success rate.

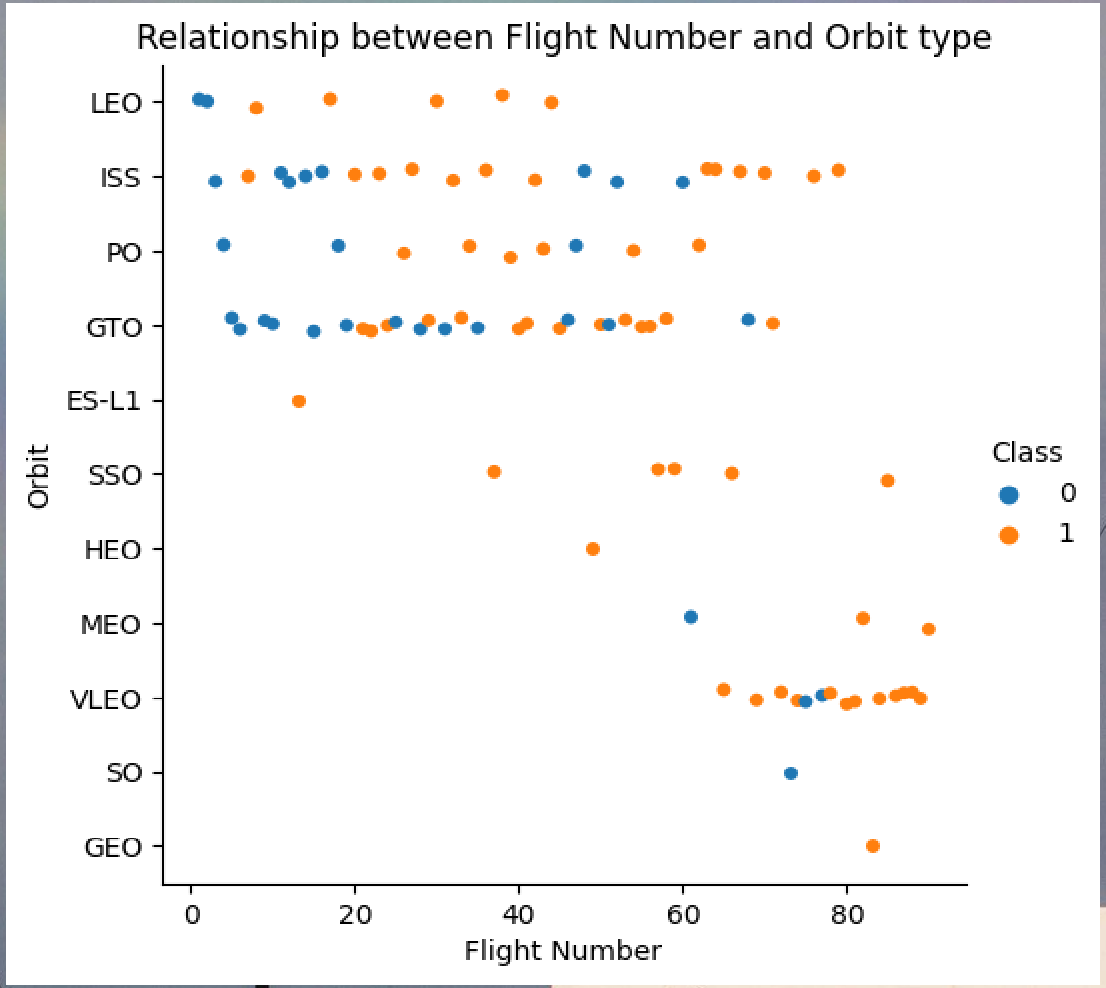


The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

FLIGHT NUMBER VS. ORBIT TYPE

Exploratory Data Analysis

The plot shows the Flight Number vs. Orbit type. We observe that in the LEO orbit, success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.

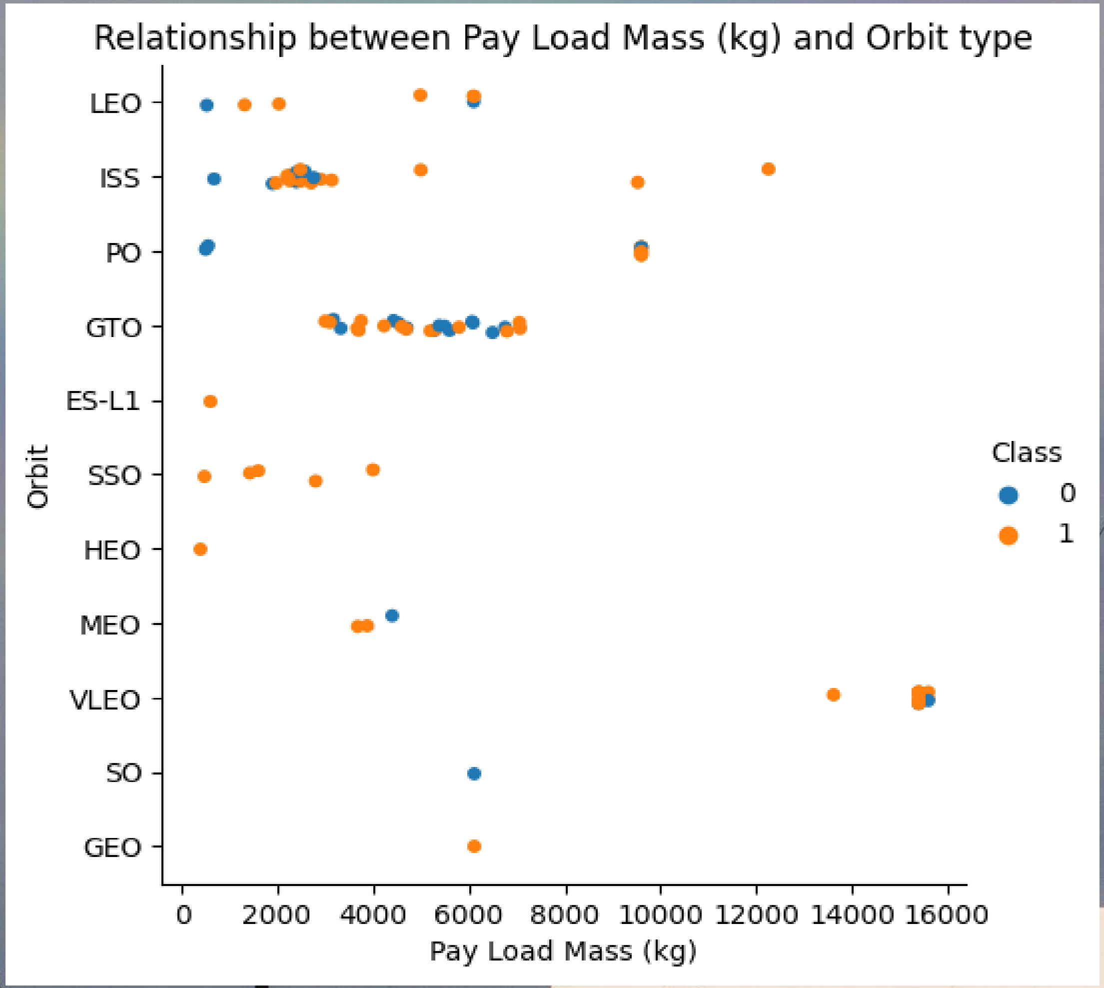


The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

PAYLOAD VS. ORBIT TYPE

Exploratory Data Analysis

- Heavy payloads are better with LEO, ISS and PO orbits
- The GTO orbit has mixed success with heavier payloads



The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

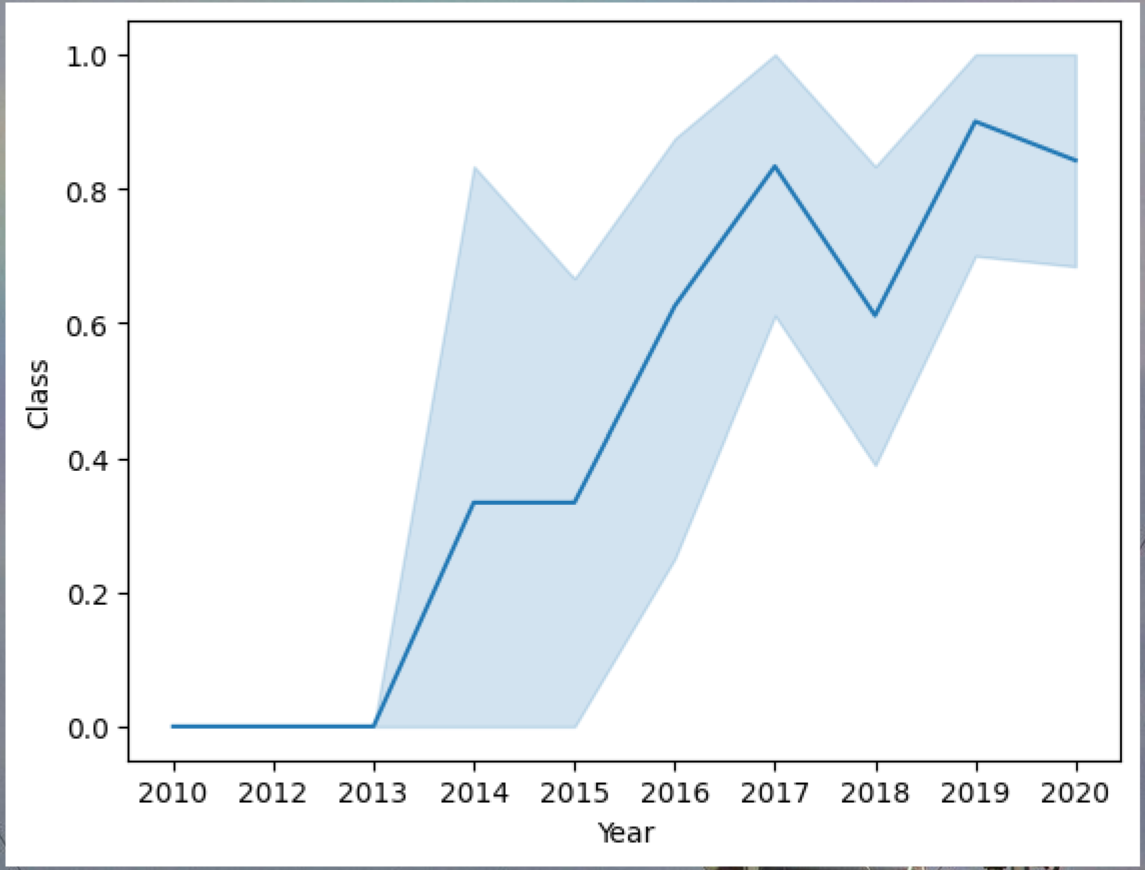
LAUNCH SUCCESS YEARLY TREND

Exploratory Data Analysis

- The success rate improved from 2013-2017 and 2018-2019

- The success rate decreased from 2017-2018 and from 2019-2020

- Overall, the success rate has improved since 2013



The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-dataviz.ipynb

ALL LAUNCH SITE NAMES

Exploratory Data Analysis

- We used the key word **DISTINCT** to show only unique launch sites from the SpaceX data.

Display the names of the unique launch sites in the space mission

```
%sql select distinct(Launch_Site) from SPACEXTABLE
```

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

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

The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

LAUNCH SITE NAMES BEGIN WITH 'CCA'

- We used the key word **LIMIT** to show only 5 launch sites from the SpaceX data begins with ‘CCA’.

Display 5 records where launch sites begin with the string '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
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

TOTAL PAYLOAD MASS

- We calculated the total payload carried by boosters from NASA (CRS) as **45596** using the *sum()* function.

Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql select sum(PAYLOAD_MASS__KG_) as 'Total Payload Mass Carried by Boosters Launched by NASA (CRS)'
from SPACEXTABLE where Customer == 'NASA (CRS)'
```

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

Total Payload Mass Carried by Boosters Launched by NASA (CRS)
45596

The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

AVERAGE PAYLOAD MASS BY F9 V1.1

- We calculated the average payload carried by booster version 'F9 v1.1' as **2928.4** using the *avg()* function.

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

```
%sql select avg(PAYLOAD_MASS__KG_) as 'Average Payload Mass carried by booster version F9 v1.1'
from SPACEXTABLE where Booster_Version == 'F9 v1.1'
```

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

Average Payload Mass carried by booster version F9 v1.1
2928.4

The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

FIRST SUCCESSFUL GROUND LANDING DATE

- We calculated First Successful Ground Landing Date as **22nd Dec. 2015** using the *min()* function.

```
%%sql select min(Date) as 'The Date when the First Succesful Landing Outcome in Ground pad was Acheived'
    from SPACEXTABLE where Landing_Outcome == 'Success (ground pad)'
```

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

The Date when the First Succesful Landing Outcome in Ground pad was Acheived
2015-12-22

The link to the notebook is https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Successful Drone Ship Landing with Payload between 4000 and 6000

- We used the **WHERE** clause to filter for boosters which have successfully landed on drone ship and applied the **AND** condition to determine successful landing with payload mass between 4000 and 6000

```
%sql select Booster_Version
from SPACEXTABLE
where Landing_Outcome LIKE '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

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

TOTAL NUMBER OF SUCCESSFUL AND FAILURE MISSION OUTCOMES

- We used the **CASE** clause and *sum() function* to calculate total successful and failure mission outcomes.

List the total number of successful and failure mission outcomes

```
%%sql
SELECT
    SUM(CASE WHEN Mission_Outcome = 'Success' THEN 1 ELSE 0 END) AS 'Successful Mission Outcomes',
    SUM(CASE WHEN Mission_Outcome = 'Failure' THEN 1 ELSE 0 END) AS 'Failure Mission Outcomes'
FROM SPACEXTABLE
WHERE Mission_Outcome IN ('Success', 'Failure')
```

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

Successful Mission Outcomes	Failure Mission Outcomes
98	0

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

BOOSTERS CARRIED MAXIMUM PAYLOAD

- We use a subquery to select maximum payload mass.
- We use WHERE clause to filter Booster Version by payload mass.

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

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

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

2015 Launch Records

- We find there are only two Launch records in year 2015 with Landing Outcome as Failure (drone ship).

Year	Month	Landing_Outcome	Booster_Version	Launch_Site
2015	October	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015	April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

RANK LANDING OUTCOMES BETWEEN 2010-06-04 AND 2017-03-20

- We use **GROUP BY** and **ORDER BY** clauses to display different landing outcomes ranking.
- We can see there are total 32 Landing Outcomes between 2010-06-04 and 2017-03-20
- We can see that there are 10 cases of Landing Outcomes as No attempt, and 1 case of Failure (parachute).

Landing_Outcome	count
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

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Section 3

Launch Sites Proximities Analysis

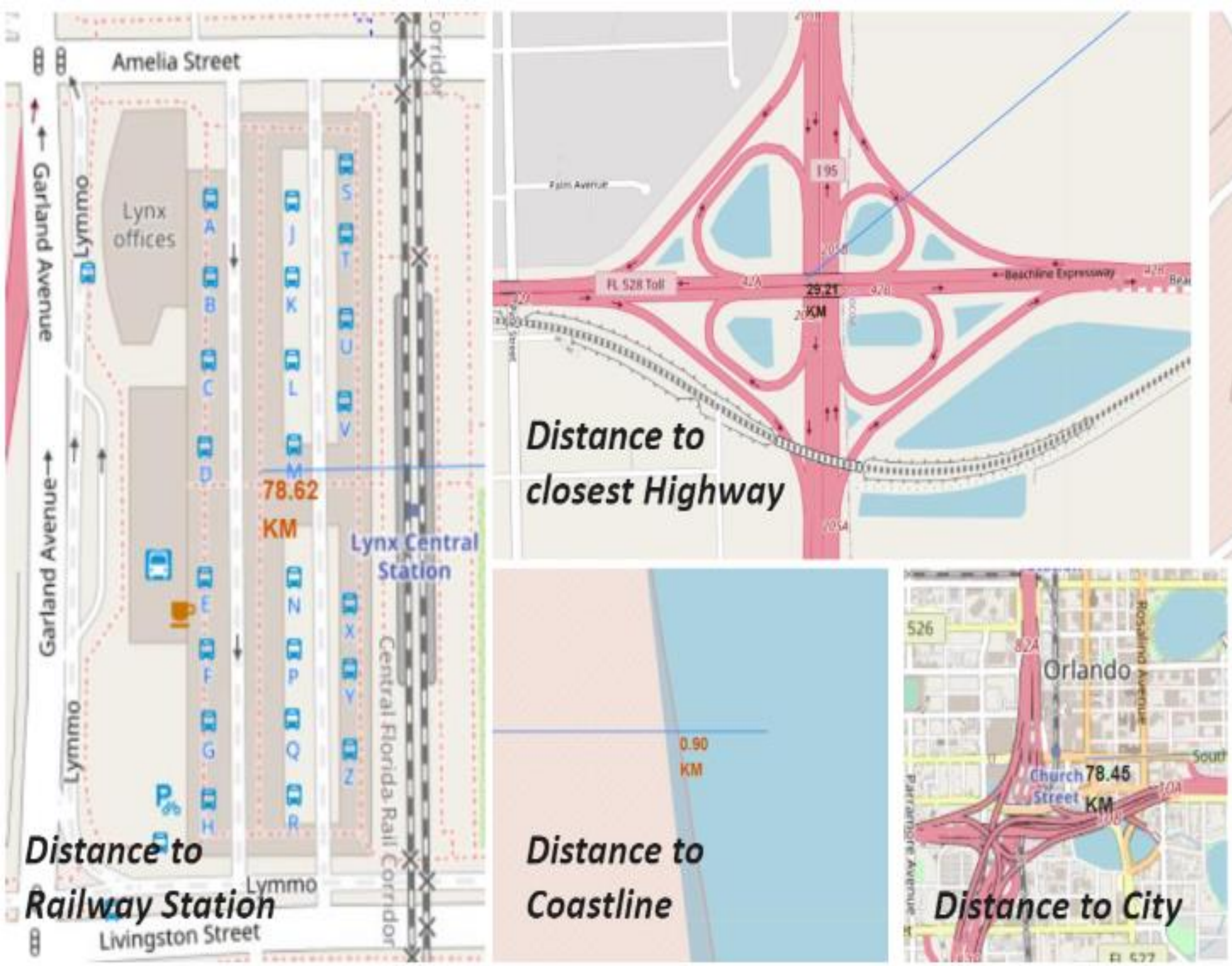
ALL LAUNCH SITES GLOBAL MAP MARKERS

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/lab_jupyter_launch_site_location.ipynb

MARKERS SHOWING LAUNCH SITES WITH COLOR LABELS

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/lab_jupyter_launch_site_location.ipynb

LAUNCH SITE DISTANCE TO LANDMARKS



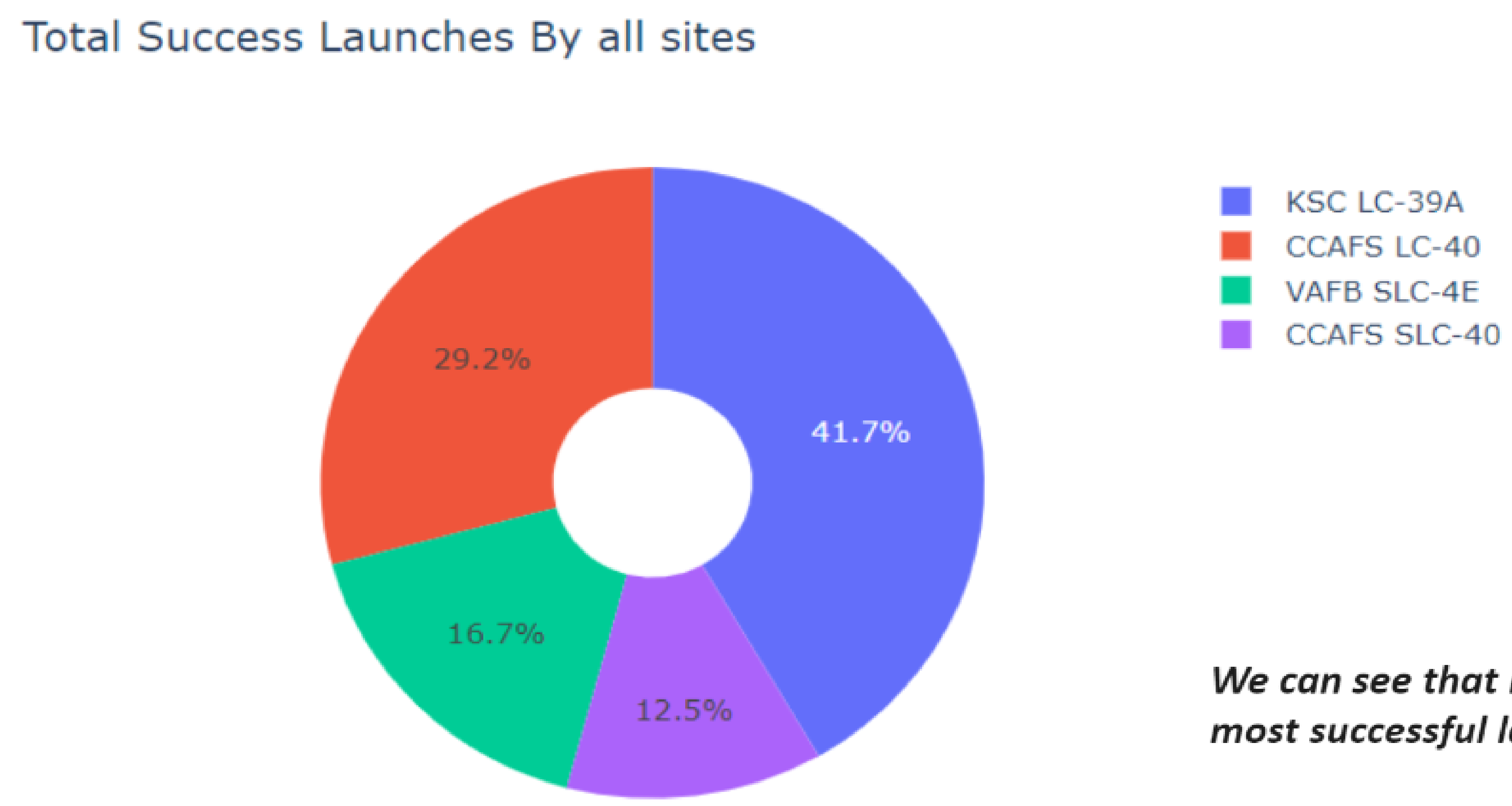
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

The link to the notebook is
https://github.com/Ashutosh-Singh-Thakur/Data-Science/blob/master/Course%2010%20Applied%20Data%20Science%20Capstone/lab_jupyter_launch_site_location.ipynb

Section 4

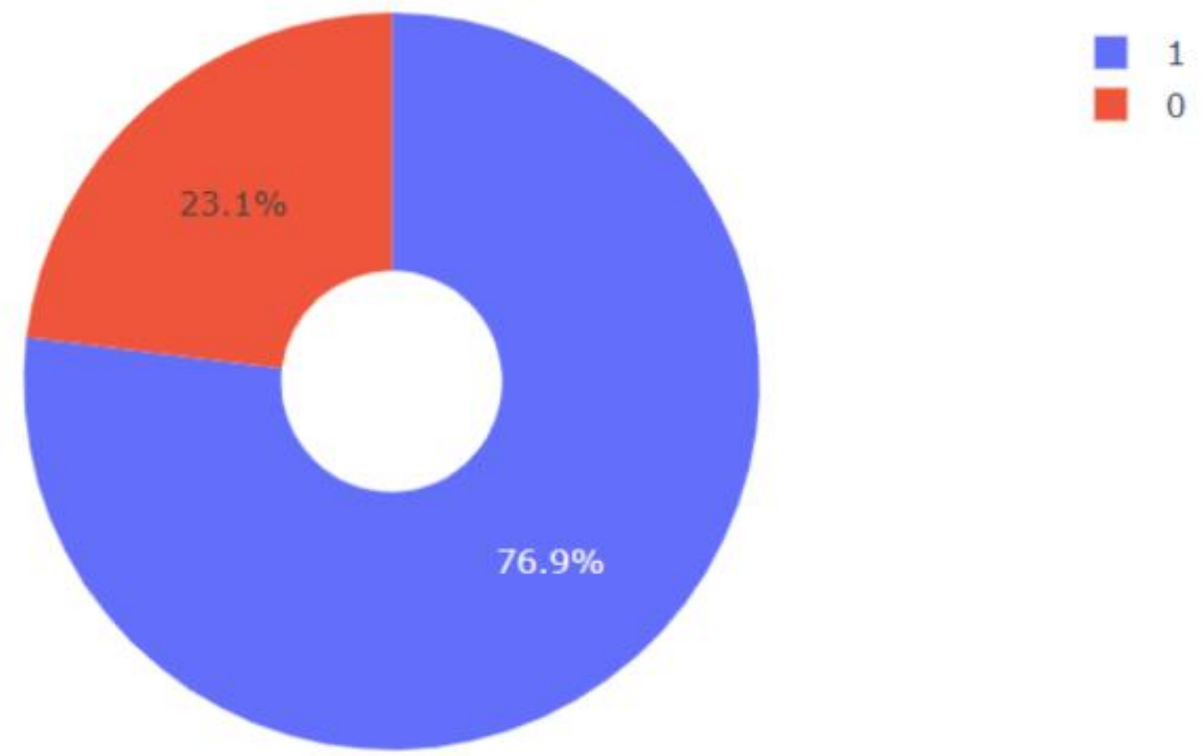
Build a Dashboard with Plotly Dash

Pie Chart Showing the Success Percentage Achieved by Each Launch Site



We can see that KSC LC-39A had the most successful launches from all the sites

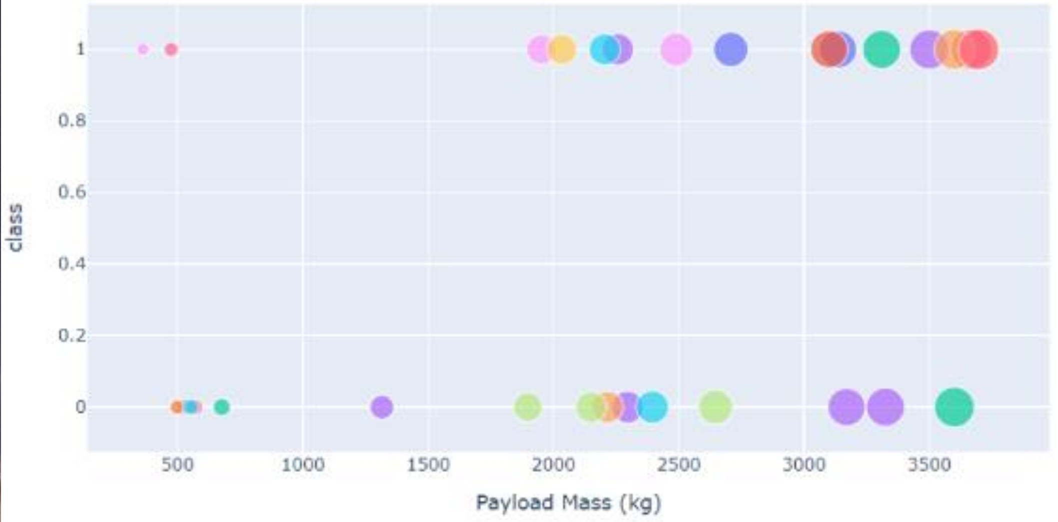
Pie chart showing the launch site with the highest launch success ratio



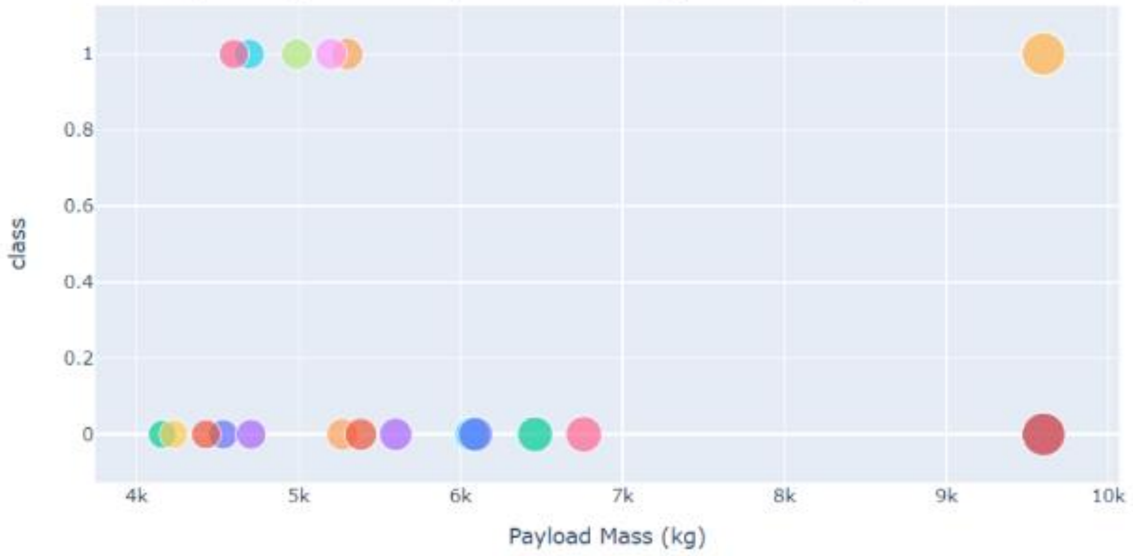
KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Scatter Plot of Payload vs Launch Outcome for All Sites, with Different Payload Selected in the Range Slider

Low Weighted Payload 0kg – 4000kg



Heavy Weighted Payload 4000kg – 10000kg



We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

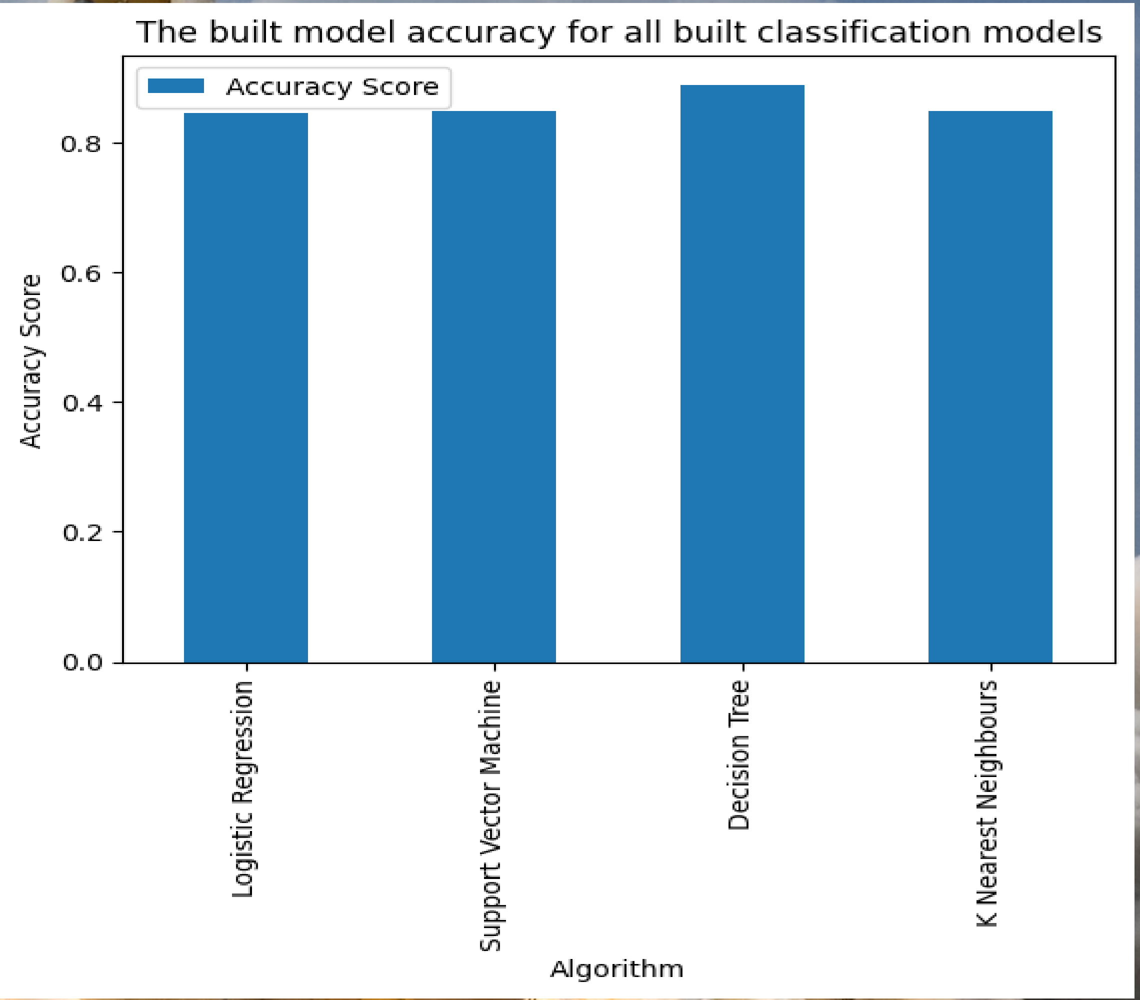
Section 5

Predictive Analysis (Classification)

CLASSIFICATION ACCURACY

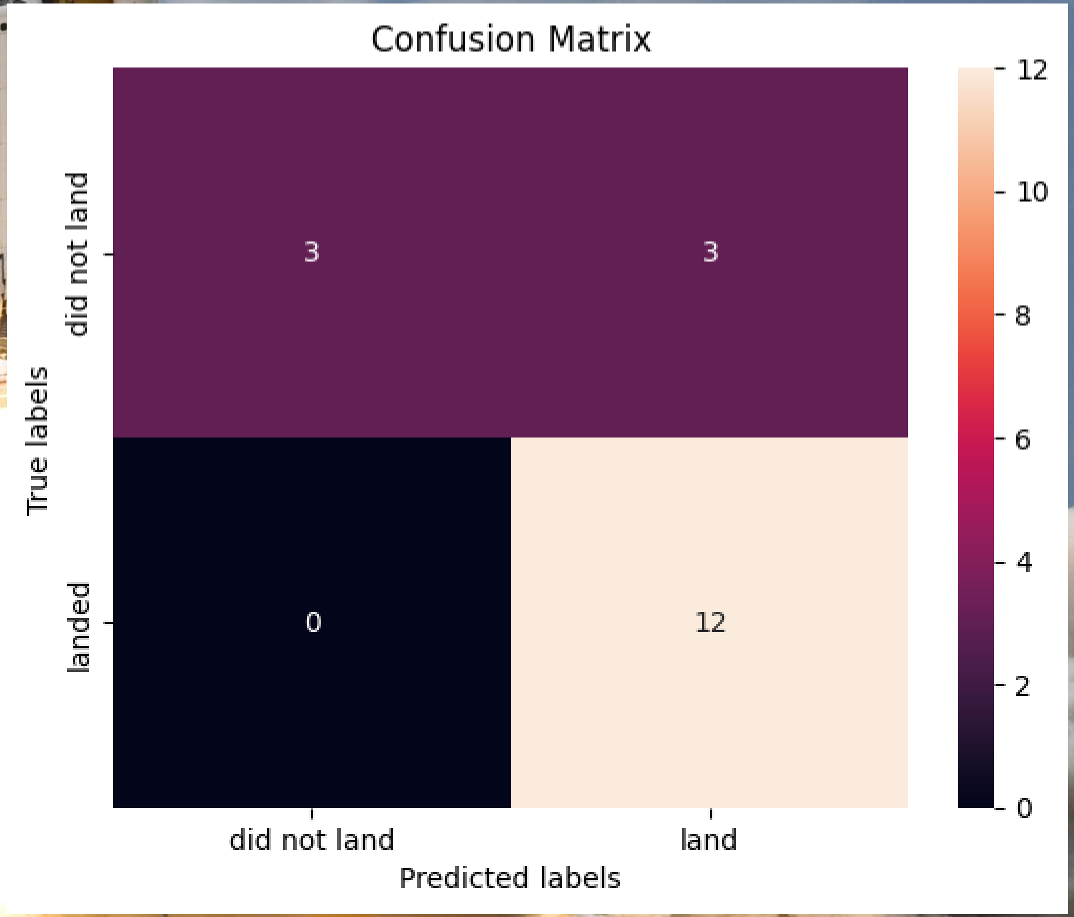
	Accuracy Score
Logistic Regression	0.846429
Support Vector Machine	0.848214
Decision Tree	0.889286
K Nearest Neighbours	0.848214

All the models performed at about the same level and had the same scores and accuracy. This is likely due to the small dataset. The **Decision Tree** model slightly outperformed the rest when looking at .best_score_



Confusion Matrix

- The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.



CONCLUSIONS

We can conclude that:

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate started to increase in 2013 till 2020.
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- KSC LC-39A had the most successful launches of any sites.
- The Decision tree classifier is the best machine learning algorithm for this task.

APPENDIX

```
import requests

import io

URL1 = 'https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_2.csv'

resp1 = requests.get(URL1)

if resp1.status_code == 200:

    text1 = io.BytesIO(resp1.content)

    data = pd.read_csv(text1)

else:

    print("Loading Error!")
```

Thank you!