SPACEX EDA - IBM APPLIED DATA SCIENCE CAPSTONE

BY VADHNA SAMEDY HUN

I. OUTLINE

. EXECUTIVE SUMMARY

. INTRODUCTION

. METHODOLOGY

. RESULTS

. CONCLUSION

. APPENDIX AND REFERENCES

EXECUTIVE SUMMARY

Summary of methodologies:

- Data collection
- Data wrangling
- EDA with data visualization
- EDA with SQL
- Building interactive map with FOLIUM
- Building a Dashboard with Plotly
- Predictive machine learning

Summary of all results:

- EDA results
- Interactive analysis
- Predictive analysis

INTRODUCTION

Project background and context

- SpaceX’s Falcon 9 rocket launches with a cost of 62 million dollars; reusability is needed.
-

Problems I want to find answers:

- This project is aimed to check whether the first stage of a SpaceX rocket will land successfully

METHODOLOGY

DATA collection:
- SpaceX REST API
- Web scraping (BeautifulSoup)

DATA wrangling:
- One-hot encoding (get dummies)

EDA:
- Using SQL and data visualization tools

Interactive visual analytics:
- Folium and Plotly

Predictive analysis:
- Machine learning methods

DATA COLLECTION

- SpaceX data is gathered from the SpaceX REST API

- Another method used is web scraping from Wikipedia using BeautifulSoup

First, we collect the data

Then, we normalize them to a DataFrame using pd.json_normalize

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/da
```

We should see that the request was successfull with the 200 status response code

```
response.status_code
```

```
200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

```
# Use json_normalize meethod to convert the json result into a dataframe
data = pd.json_normalize(response.json())
```

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

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

https://github.com/SamedyHUNX/applied-data-science-capstone/blob/main/SPACEX%20-%20DATA%20COLLECTION%20-%20VADHNA%20SAMEDY%20HUN.ipynb

DATA WRANGLING

Data wrangling is one of the most important part in data analysis. In this project, first, we had to check for missing values; then we replaced them with a appropriate method.

Data Wrangling

We can see below that some of the rows are missing values in our dataset.

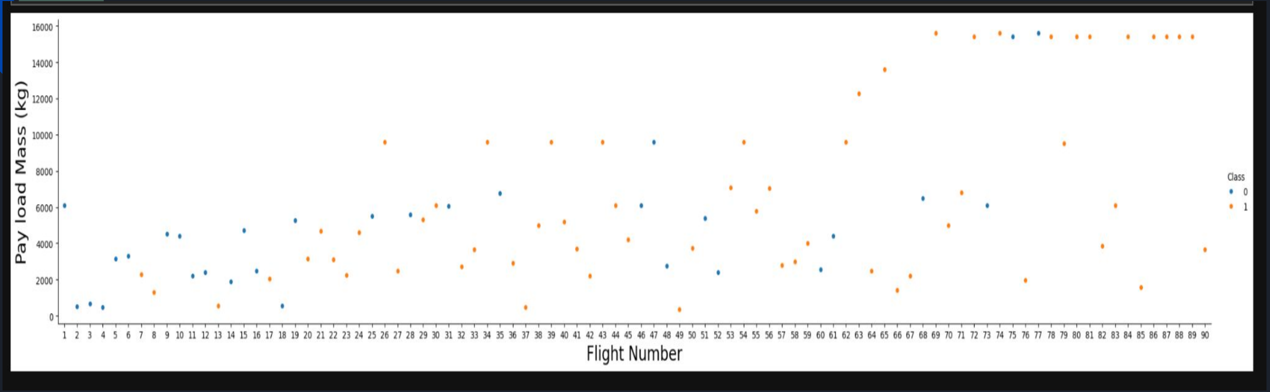
```
data_falcon9.isnull().sum()
```

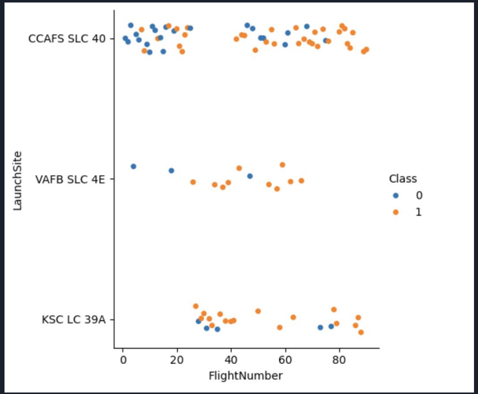
Task 3: Dealing with Missing Values

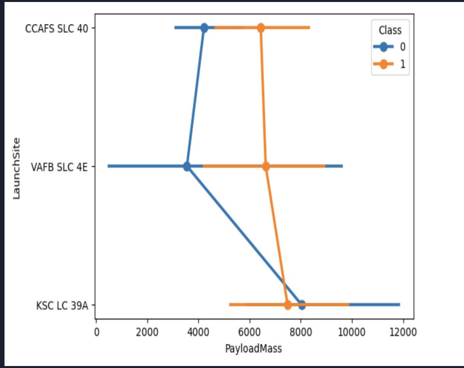
Calculate below the mean for the `PayloadMass` using the `.mean()`. Then use the mean and th
`np.nan` values in the data with the mean you calculated.

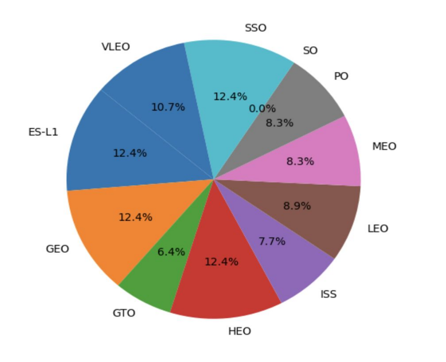
```
# Calculate the mean value of PayloadMass column
payloadmassmean = data_falcon9.PayloadMass.mean()

# Replace the np.nan values with its mean value
data_falcon9 = data_falcon9.replace(np.nan, payloadmassmean)
```

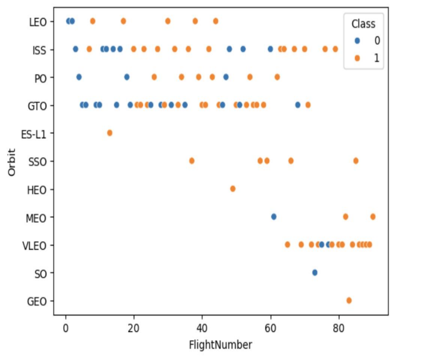
EDA AND VISUALIZATION

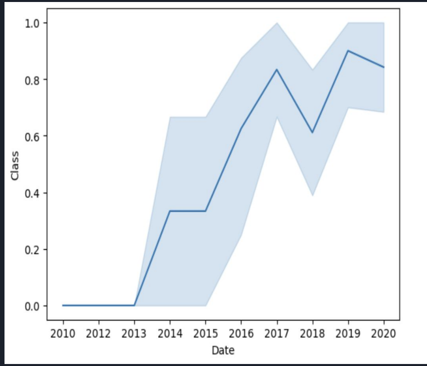












https://github.com/SamedyHUNX/applied-data-science-capstone/blob/main/SPCEX%20-%20DATA%20VISUALIZATION%20-%20VADHNA%20SAMEDY%20HUN.ipynb

EDA with SQL

SQL gives a better leverage to Python for data query. In this task, instead of using commands in Python to query data, SQL commands were used.

```
%load_ext sql
```

```
import csv, sqlite3

con = sqlite3.connect("my_data1.db")
cur = con.cursor()
```

```
!pip install -q pandas
```

```
%sql sqlite:///my_data1.db
```

```
import pandas as pd
df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/l
df.to_sql("SPACEXTBL", con, if_exists='replace', index=False,method="multi")
```

Task 1

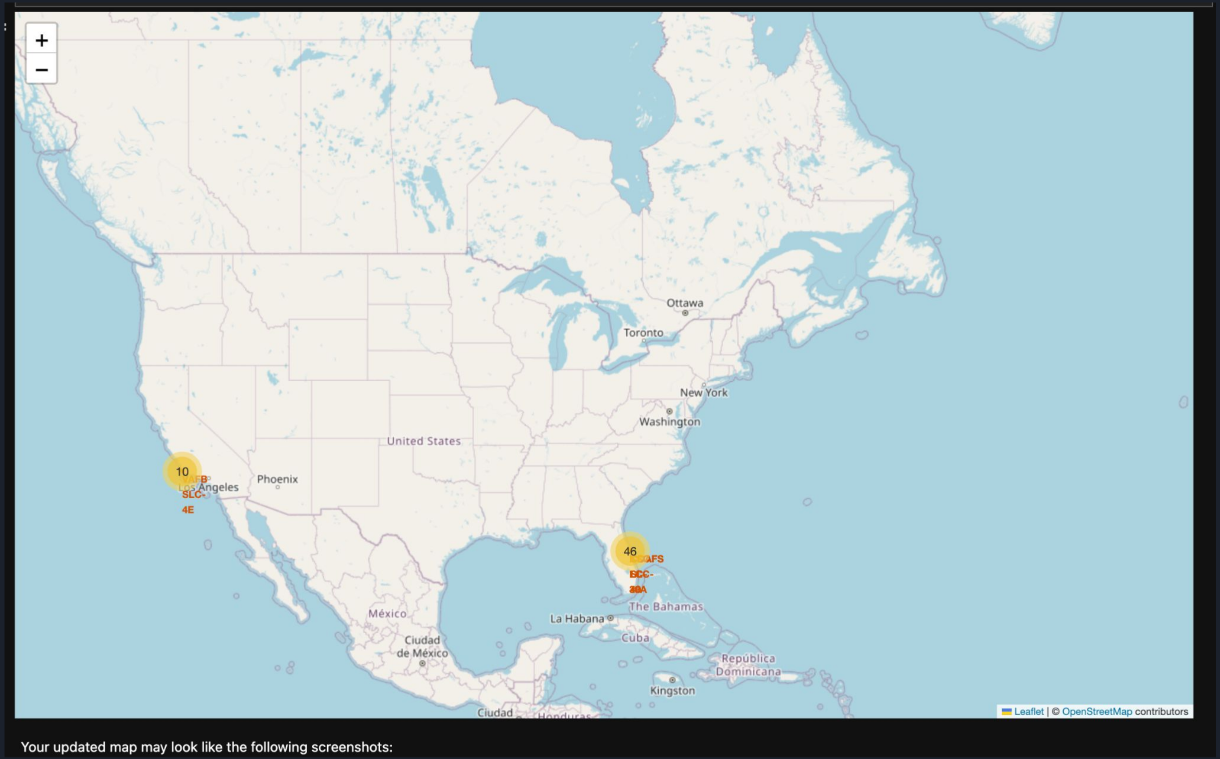
Display the names of the unique launch sites in the space mission

```
query = 'SELECT DISTINCT(Launch_site) FROM df'
cur.execute('SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL').fetchall()
```

```
[('CCAFS LC-40',), ('VAFB SLC-4E',), ('KSC LC-39A',), ('CCAFS SLC-40',)]
```

https://github.com/SamedyHUNX/applied-data-science-capstone/blob/main/SPACEX%20-%20CAPSTONE%20-%20SQL%20WITH%20PYTHON%20-%20VADHNA%20SAMEDY%20HUN.ipynb

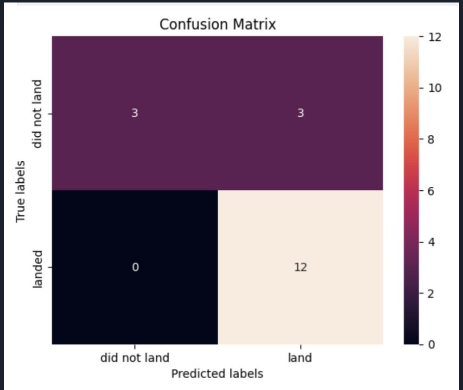
MAP with Folium

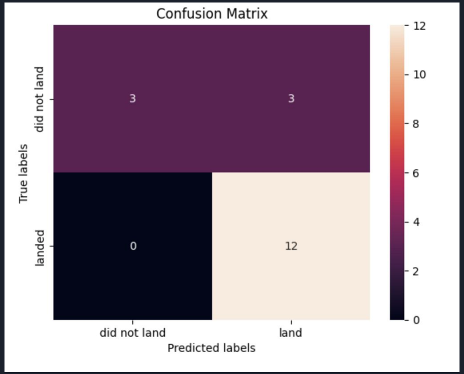


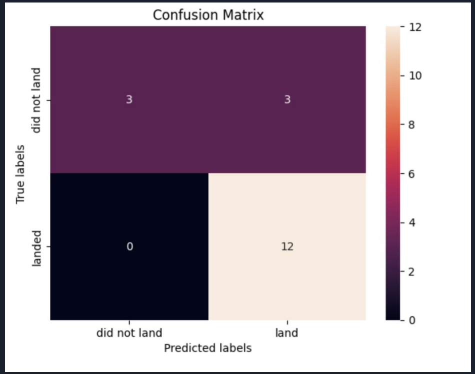
https://github.com/SamedyHUNX/applied-data-science-capstone/blob/main/SPACEX%20-%20FOLIUM%20-%20CAPSTONE%20PROJECT%20-%20VADHNA%20SAMEDY%20HUN.ipynb

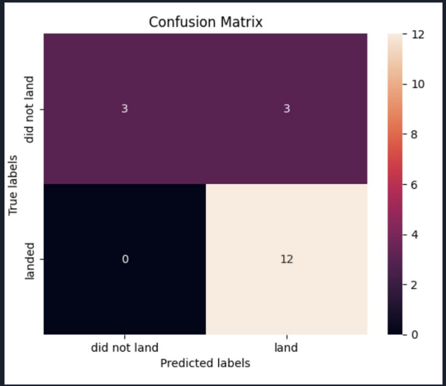
Predictive Analysis (Classification)

SVM, KNN, and Logistic Regression model similarly achieved the highest accuracy score of 83.33%.









https://github.com/SamedyHUNX/applied-data-science-capstone/blob/main/SPACEX%20-%20PREDICTIVE%20ANAYLYSIS%20-%20VADHNA%20SAMEDY%20HUN.ipynb

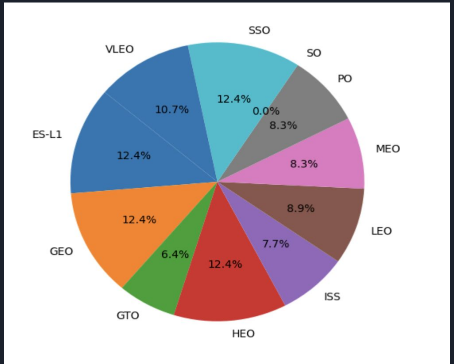
Results:

- KNN is the foremost model for forecasting outcome in this data.

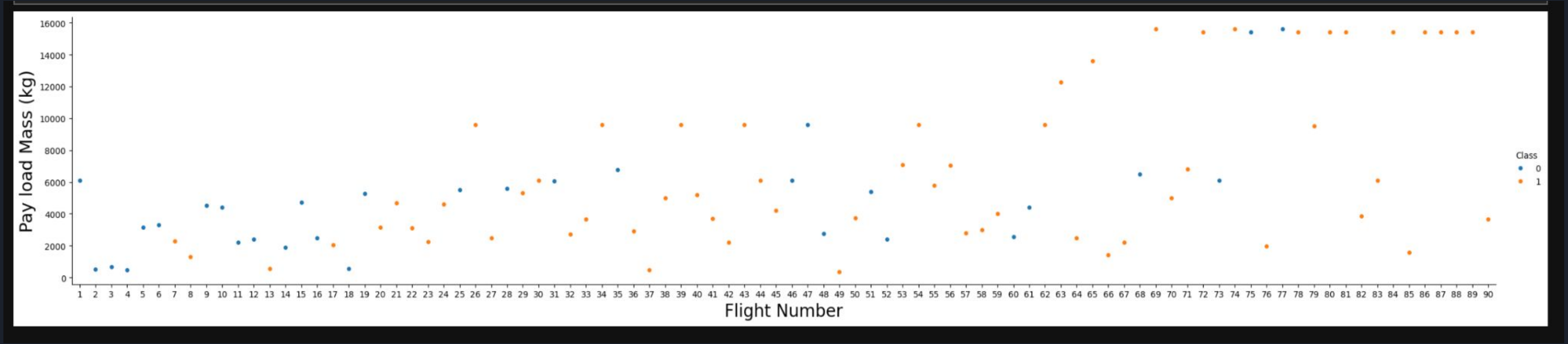
- Lighter payloads have a higher performance.

- The likelihood of SpaceX launches succeed increases with years.

- Launch Complex 39A at Kennedy Space Center has the highest successful launches.
-
- SSO, GEO, ES-L1, HEO orbit types have the highest success rates (all at 12.4%).

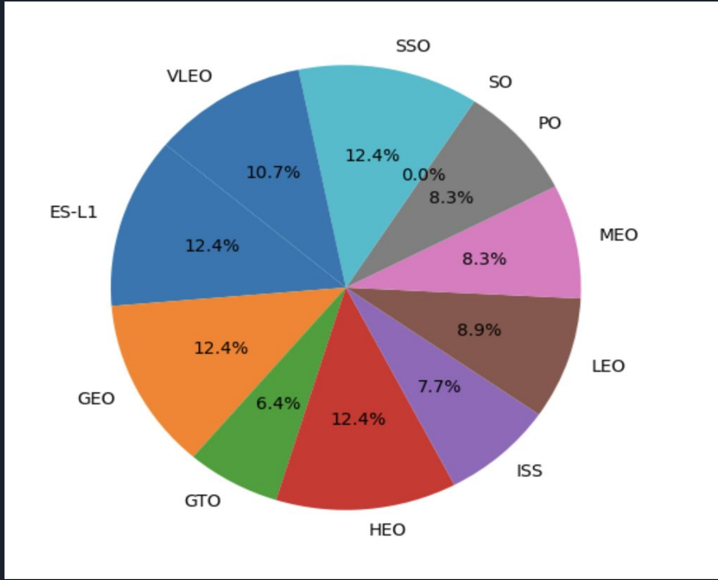


II. Insights drawn



Lower payloads mass have a higher succeed rates than higher ones, across all launch sites and models. Flight No. after 78 shows 100% success rates.

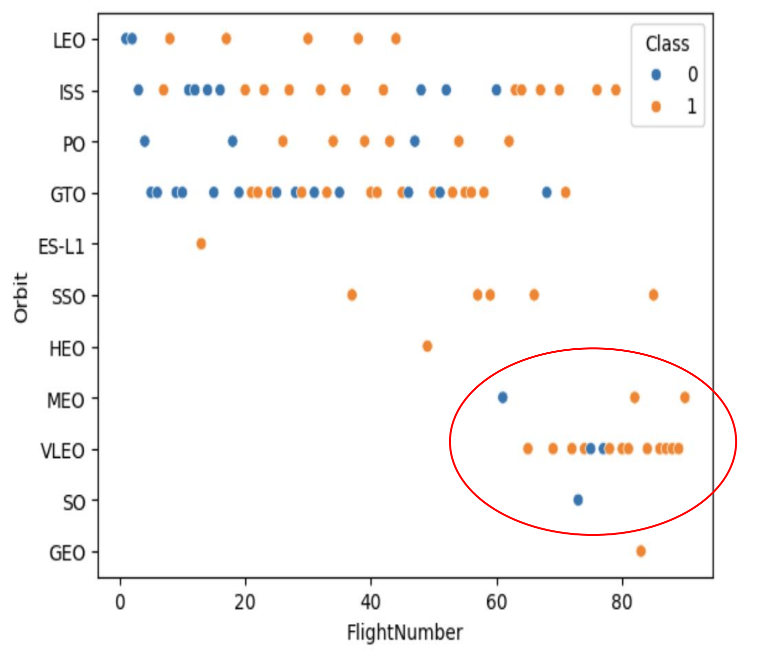
Success Rate vs. Orbit Type



- SSO, GEO, ES-L1, HEO orbit types have the highest success rates (all at 12.4%).

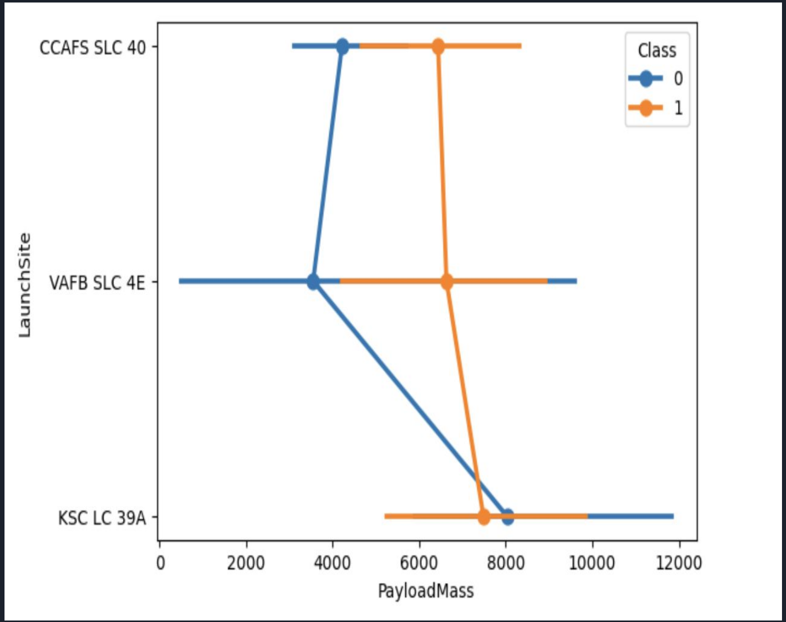
Flight Number vs. Orbit Type

Launches numbers and success rates are slowly shifted to VLEO orbit in the later years. (Circled)



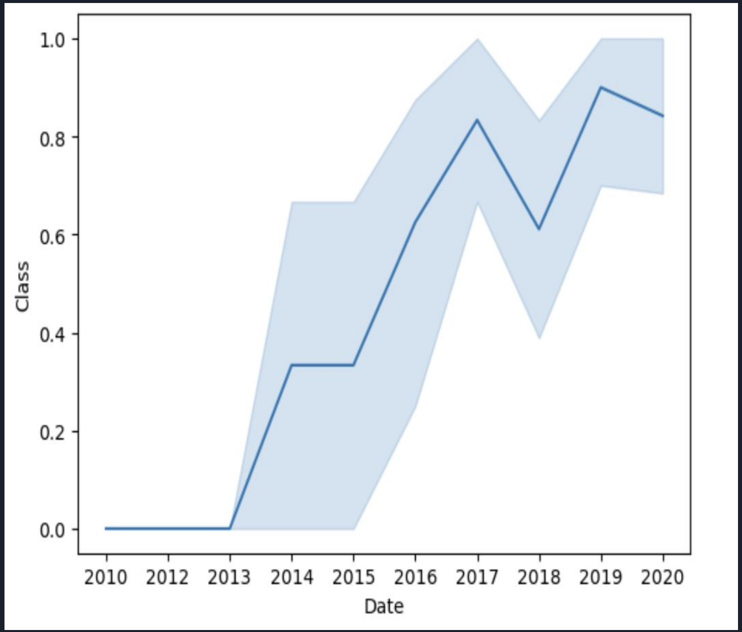
Payload mass vs. Launch sites

Optimal masses of a payload is between 6000 and 8500 kg.



Launch Success Yearly Trend

Launch Success trend is increasing sharply upwards since 2013. This is possibly due to advancement of technology and experience.



All Launch Site Names

Task 1

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

In [23]:
```
query = 'SELECT DISTINCT(Launch_site) FROM df'
cur.execute('SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL').fetchall()
```

Out[23]:
```
[('CCAFS LC-40',), ('VAFB SLC-4E',), ('KSC LC-39A',), ('CCAFS SLC-40',)]
```

- Display all launch sites using SQL commands: there are 4 unique launc sites.

Launch Site Names Begin with 'CCA'

Task 2

Display 5 records where launch sites begin with the string 'CCA'

In [26]:
```
cur.execute("SELECT * FROM SPACEXTBL WHERE SUBSTR(SUBSTR(Launch_Site, 1, INSTR(Launch_Site, ' ') - 1), 1, 3) = 'C
```

Out[26]:
```
[('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,
```

Total Payload Mass

Task 3

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

```
cur.execute("SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)'").fetchall()
```

```
[(45596,)]
```

Average Payload Masss by F9 v1.1

Task 4

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

```
import numpy as np
```

```
cur.execute("SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1'").fetchall()
```

```
[(2928.4,)]
```

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
cur.execute("SELECT MIN(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success'").fetchall()
```

```
[('2018-07-22',)]
```

Successful Drone Ship Landing with the Payload between 4000 and 6000

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
cur.execute("SELECT PAYLOAD FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success' AND PAYLOAD_MASS__KG_ > 4000 AND PAY
```

```
[('Merah Putih ',),
 ('Es hail 2',),
 ('Nusantara Satu, Beresheet Moon lander, S5',),
 ('RADARSAT Constellation, SpaceX CRS-18 ',),
 ('GPS III-03, ANASIS-II',),
 ('ANASIS-II, Starlink 9 v1.0',),
 ('GPS III-04 , Crew-1',)]
```

Success Rate (%)

Task 7

List the total number of successful and failure mission outcomes

```
cur.execute("SELECT (COUNT(CASE WHEN LANDING_OUTCOME = 'Success' THEN 1 END) * 100.0 / COUNT(*)) AS Success_Perce
```

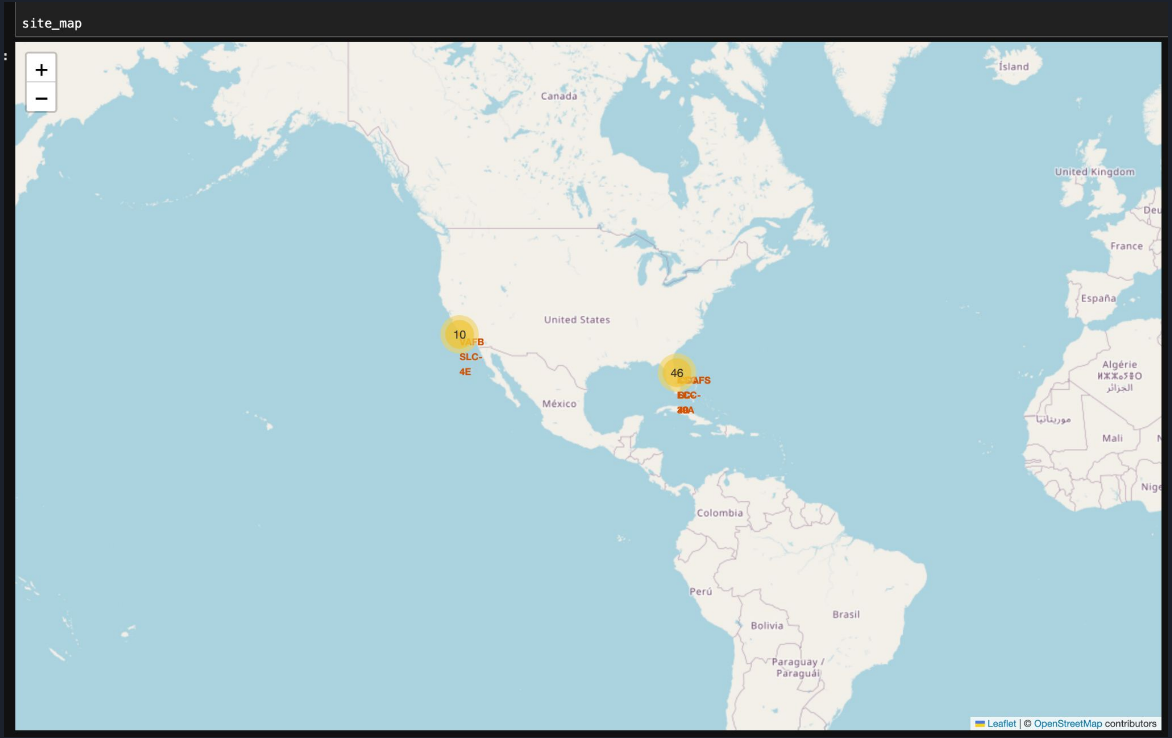
```
[(37.62376237623762,)]
```

Boosters Carried Maximum Payload Mass

```
cur.execute("SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM
```

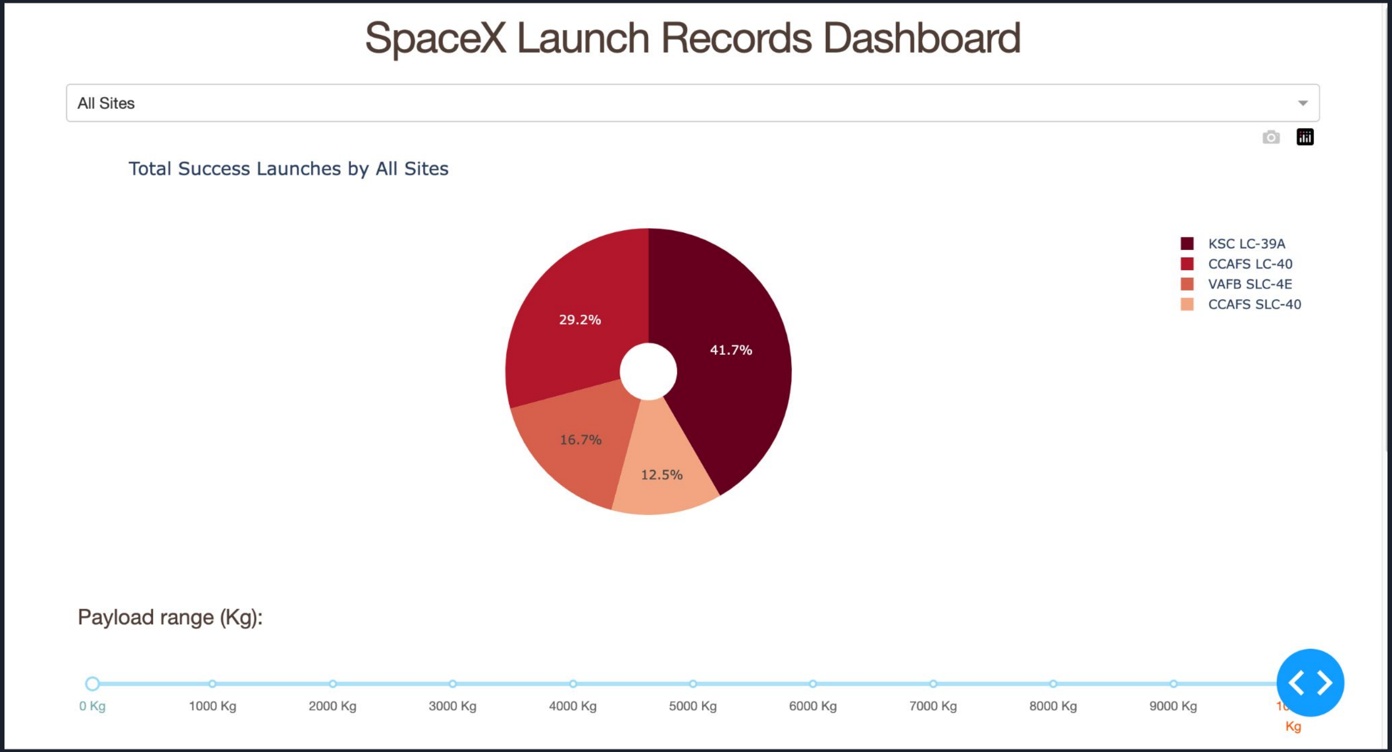
```
[('F9 B5 B1048.4',),
 ('F9 B5 B1049.4',),
 ('F9 B5 B1051.3',),
 ('F9 B5 B1056.4',),
 ('F9 B5 B1048.5',),
 ('F9 B5 B1051.4',),
 ('F9 B5 B1049.5',),
 ('F9 B5 B1060.2 ',),
 ('F9 B5 B1058.3 ',),
 ('F9 B5 B1051.6',),
 ('F9 B5 B1060.3',),
 ('F9 B5 B1049.7 ',)]
```

III. Launch Sites Approximity Analysis



Success/failed launches for each site

IV. Build a Dashboard with Plotly



Payload vs. Launch outcome

- Highest success rates occur when the payload is between 2000 and 4000kg.
- Booster FT (green) has the highest success rate.

V. Predictive Analysis (Classification)

Calculate the accuracy on the test data using the method `score` :

```
test_accuracy = logreg_cv.best_estimator_.score(X_test, Y_test)
print("Test set accuracy: ", test_accuracy)
```

```
Test set accuracy:  0.8333333333333334
```

Calculate the accuracy on the test data using the method `score` :

```
test_accuracy = svm_cv.best_estimator_.score(X_test, Y_test)
print("Test set accuracy: ", test_accuracy)
```

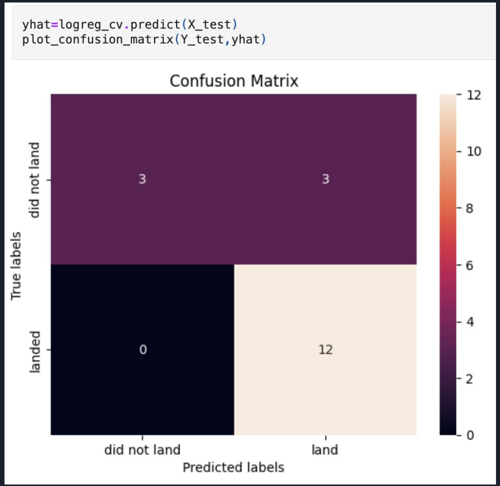
```
Test set accuracy:  0.8333333333333334
```

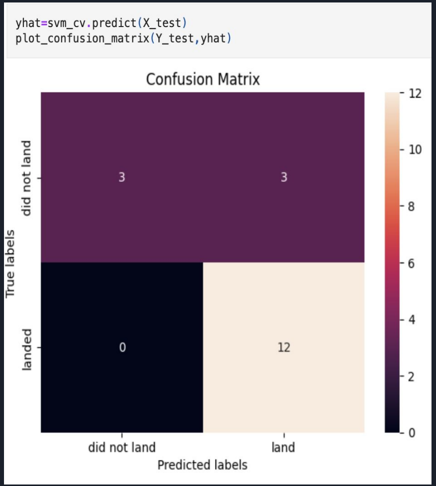
Calculate the accuracy of tree_cv on the test data using the method `score` :

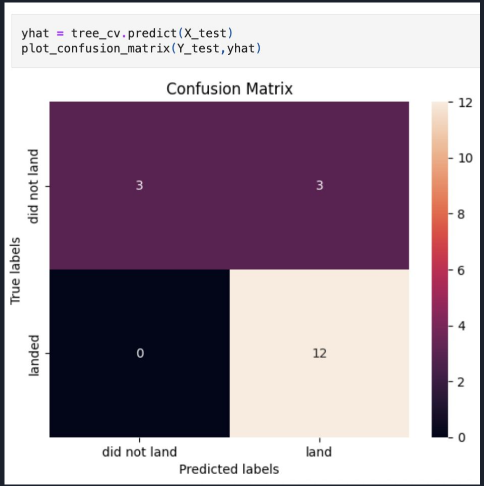
```
test_accuracy = tree_cv.best_estimator_.score(X_test, Y_test)
print("Test set accuracy: ", test_accuracy)
```

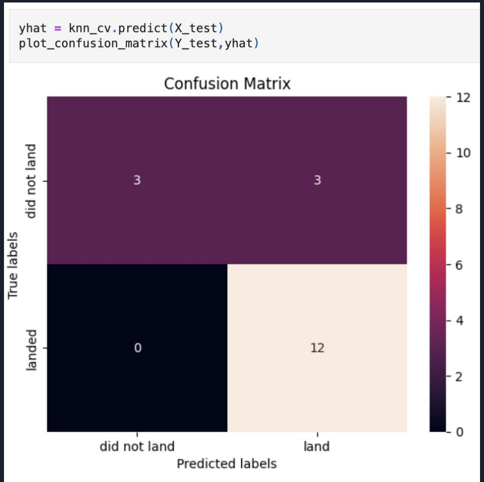
```
Test set accuracy:  0.8333333333333334
```

Confusion Matrix









- LR, SVM, KNN are good as their confusion matrix show predicted 12 successful landing correctly.

- They have the same accuracy off 83.33%.

Conclusions:

- LR, SVM, KNN can be used as model for forecasting (SpaceX).

- Lighter payloads have a higher success than heavier ones. But optimality between 2000 and 4000 should be opted for.

- The likelihood of success increases in respect with time, but can be certain that there will be increase of success rates in upcoming years.

- GEO, HEO, SSO, ES L1 orbit types exhibit the highest rates.

- Launch Complex 39A has the highest numbers of success launches.