IBM DATA SCIENCE PROJECT

SPACE_X FALCON 9

LOKESH KUMAR M

30-07-2023

IBM Developer

SKILLS NETWORK

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion

EXECUTIVE SUMMARY

- The SpaceX Falcon 9 Project represents a groundbreaking initiative that has revolutionized the space launch industry.
- With its reusable technology, high payload capacity, and impressive track record, the Falcon 9 continues to be a driving force in advancing space exploration and commercial space ventures.
- As SpaceX pushes the boundaries of rocketry, the Falcon 9 Project remains at the forefront of innovation, shaping the future of space travel.

INTRODUCTION

- Falcon 9’s ability to autonomously land on solid ground after completing its primary mission has not only reduced the cost of space travel but also heralded a new era of sustainable rocket technology.
- In this data science project, we embark on a journey to analyze the landing success of SpaceX Falcon 9 missions.
- By harnessing the power of data, we aim to gain insights into the factors that influence successful landings and identify patterns that contribute to the rocket's safe return to Earth's surface.

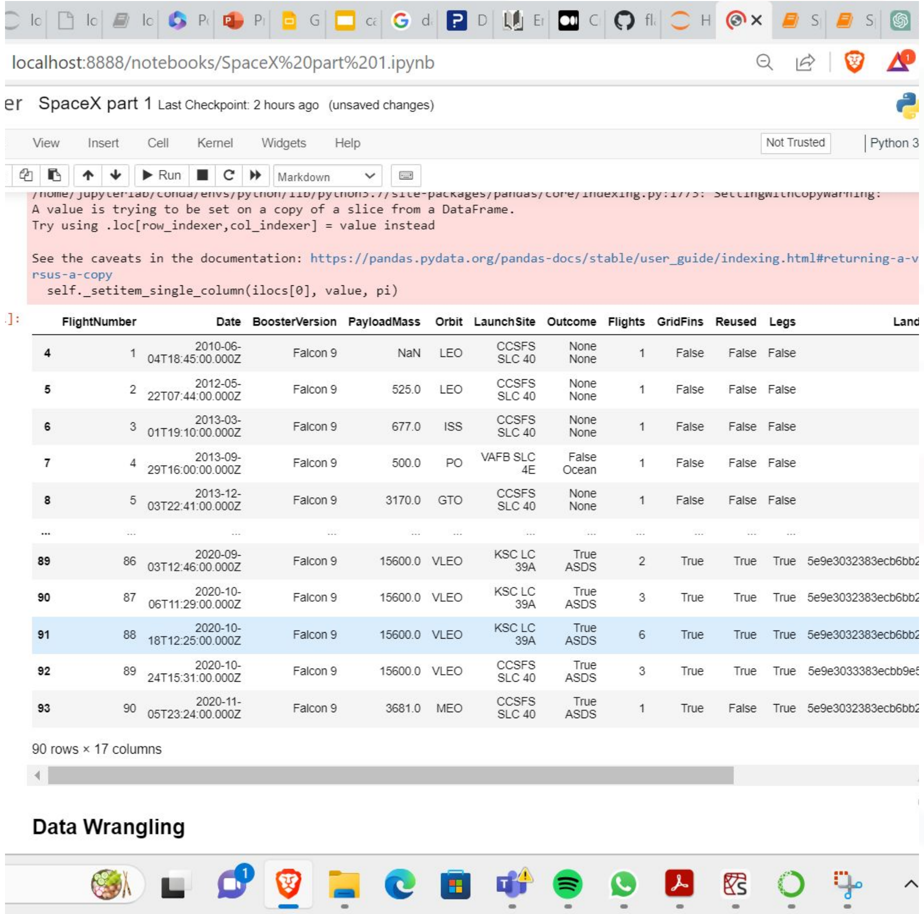
METHODOLOGY

- We will utilize historical data on Falcon 9 missions to perform a comprehensive analysis, exploring various parameters and metrics associated with each launch and landing attempt.

- Our data-driven approach involves collecting and preprocessing Falcon 9 mission data, including launch date, mission outcome, landing location, payload information, and other relevant attributes.

- Leveraging various data science techniques and machine learning algorithms, we will conduct an in-depth analysis to uncover trends and patterns related to landing success.

DATA COLLECTION



	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	Land
4	1	2010-06-04T18:45:00.000Z	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	
5	2	2012-05-22T07:44:00.000Z	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	
6	3	2013-03-01T19:10:00.000Z	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	
7	4	2013-09-29T16:00:00.000Z	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	
8	5	2013-12-03T22:41:00.000Z	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	
...	...	...	...	...	...	...	...	...	...	...	...	
89	86	2020-09-03T12:46:00.000Z	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6bb2
90	87	2020-10-06T11:29:00.000Z	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6bb2
91	88	2020-10-18T12:25:00.000Z	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6	True	True	True	5e9e3032383ecb6bb2
92	89	2020-10-24T15:31:00.000Z	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3	True	True	True	5e9e3033383ecbb9e5
93	90	2020-11-05T23:24:00.000Z	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1	True	False	True	5e9e3032383ecb6bb2

Data Collection from SpaceX API:

- SpaceX provides an API (Application Programming Interface) that allows you to access their data programmatically.

Data Collection from Wikipedia (Web Scraping):

- Web scraping is the process of extracting information from websites.
- To collect data from Wikipedia, you can use Python requests library to fetch the HTML content of relevant Wikipedia pages and then use a web scraping library like BeautifulSoup or lxml to parse the HTML and extract the necessary data.

DATA WRANGLING

```
In [5]: df.isnull().sum()/df.shape[0]*100

Out[5]: FlightNumber       0.000000
        Date               0.000000
        BoosterVersion     0.000000
        PayloadMass        0.000000
        Orbit              0.000000
        LaunchSite         0.000000
        Outcome            0.000000
        Flights            0.000000
        GridFins           0.000000
        Reused             0.000000
        Legs               0.000000
        LandingPad        28.888889
        Block              0.000000
        ReusedCount        0.000000
        Serial             0.000000
        Longitude          0.000000
        Latitude           0.000000
        dtype: float64
```

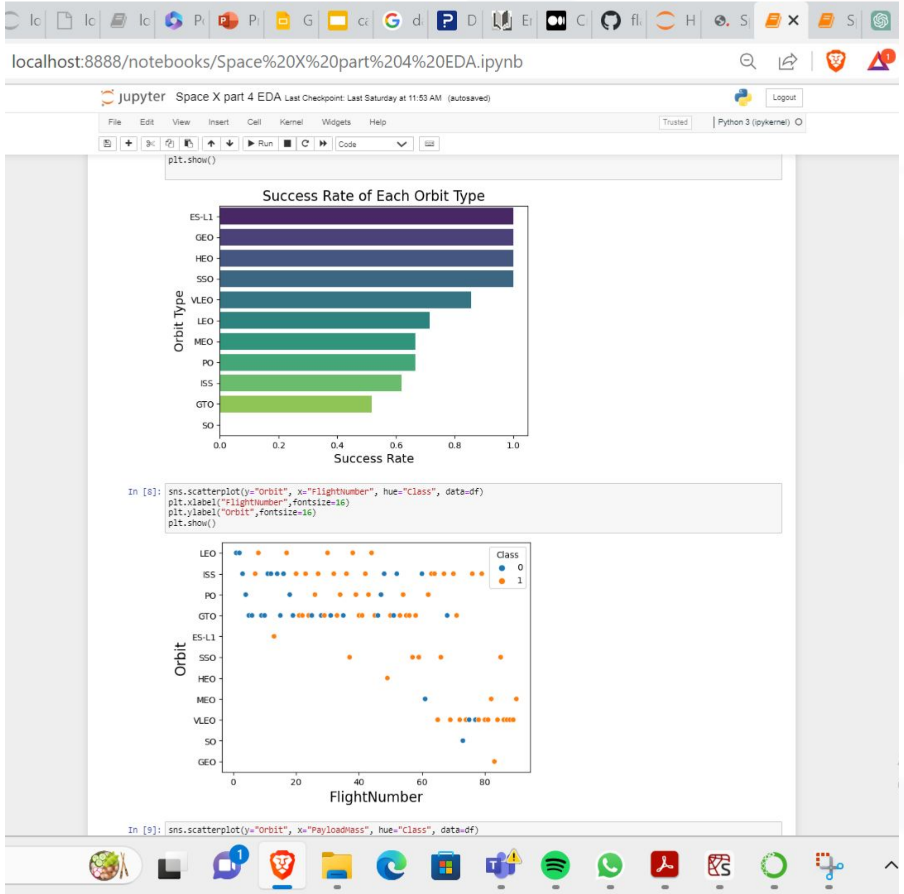
```
In [6]: df.dtypes

Out[6]: FlightNumber        int64
        Date               object
        BoosterVersion     object
        PayloadMass       float64
        Orbit              object
        LaunchSite         object
        Outcome            object
        Flights             int64
        GridFins             bool
        Reused               bool
        Legs                 bool
        LandingPad         object
        Block             float64
        ReusedCount         int64
        Serial             object
        Longitude         float64
        Latitude          float64
        dtype: object
```

Data Collection from SpaceX API:

- It involves cleaning, transforming, and organizing the collected data to make it suitable for analysis and modeling. Data wrangling helps ensure that the data is accurate, consistent, and in a usable format for further processing.
- Converting data types, creating new features from existing ones if they can provide additional insights or improve the performance of models. For example, we can derive features like the year from the launch date, calculate the payload-to-orbit ratio, etc and normalizing or scale numeric features to bring them to a similar scale, which can be helpful for certain algorithms etc.

EXPLORATORY DATA ANALYSIS

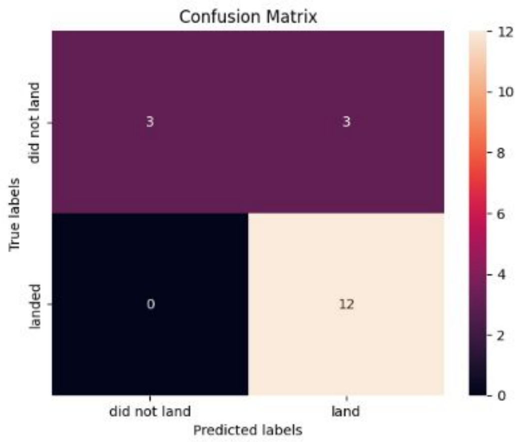


- Exploratory Data Analysis (EDA) is a critical step in the data analysis process. It involves examining and visualizing the data to gain insights, identify patterns, detect anomalies, and formulate hypotheses. EDA helps you understand the characteristics of the dataset and guide subsequent data processing, modeling, and decision-making.

- It includes summary statistics, data visualization, data distribution, correlation analysis, etc.

PREDICTIONS

In [36]:
```
yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



TASK 12

Find the method performs best:

In [37]:
```
# Assuming you have already performed the necessary steps for each algorithm and obtained the tuned models
# logreg_cv, svm_cv, tree_cv, and knn_cv.

# Calculate the accuracy on the test data for each tuned model
accuracy_logreg = logreg_cv.score(X_test, Y_test)
accuracy_svm = svm_cv.score(X_test, Y_test)
accuracy_tree = tree_cv.score(X_test, Y_test)
accuracy_knn = knn_cv.score(X_test, Y_test)

# Print the accuracy for each method
print(f"Accuracy for Logistic Regression: {accuracy_logreg:.2f}")
print(f"Accuracy for Support Vector Machine: {accuracy_svm:.2f}")
print(f"Accuracy for Decision Tree: {accuracy_tree:.2f}")
print(f"Accuracy for K Nearest Neighbors: {accuracy_knn:.2f}")
```

```
Accuracy for Logistic Regression: 0.83
Accuracy for Support Vector Machine: 0.78
Accuracy for Decision Tree: 0.78
Accuracy for K Nearest Neighbors: 0.83
```

Hyperparameter Tuning:

- Fine-tune the model's hyperparameters to optimize its performance. This process involves selecting the best combination of hyperparameters to achieve the best results.

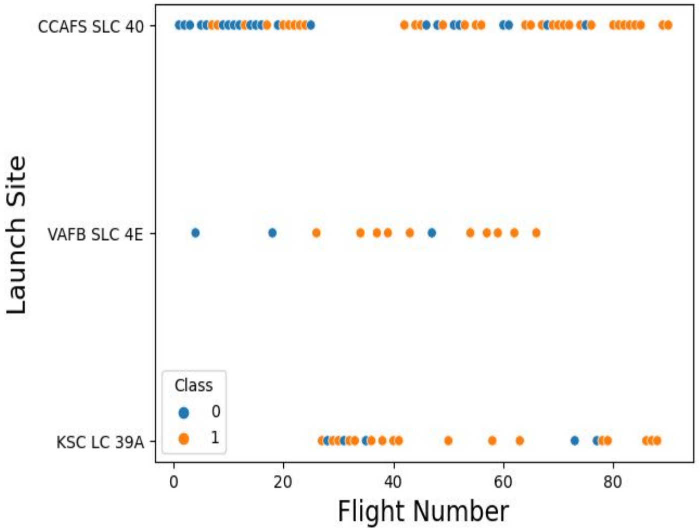
Model Deployment:

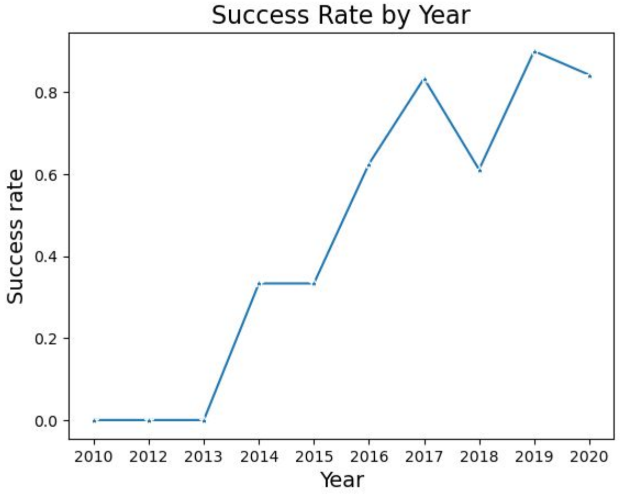
- Once the predictive model is trained and evaluated satisfactorily, deploy it to make predictions on new, real-world data.

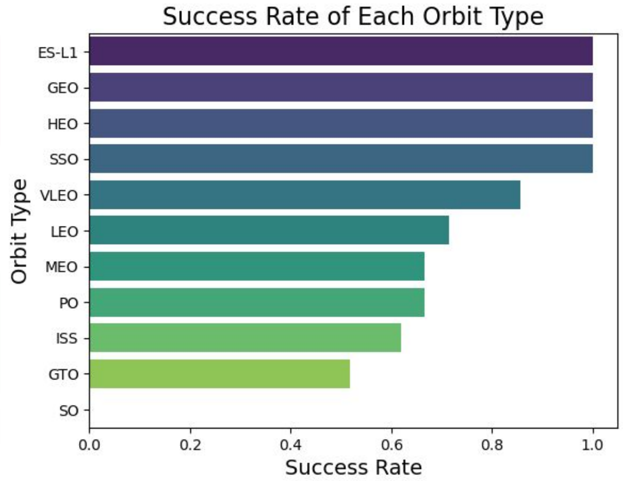
Monitoring and Maintenance:

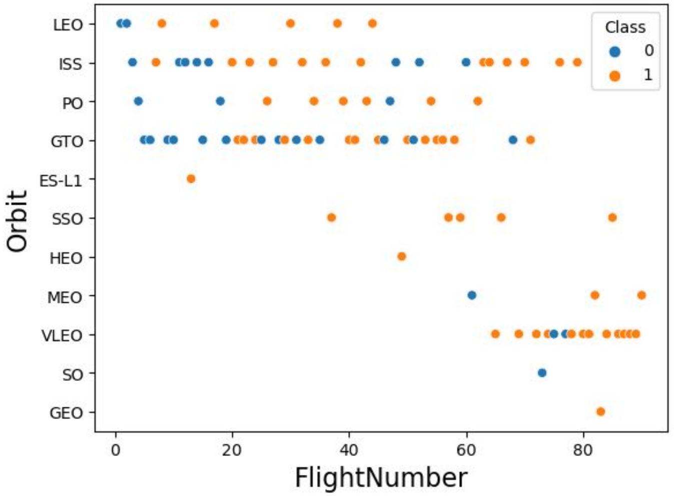
- Continuously monitor the model's performance and update it periodically to keep it relevant as new data becomes available.

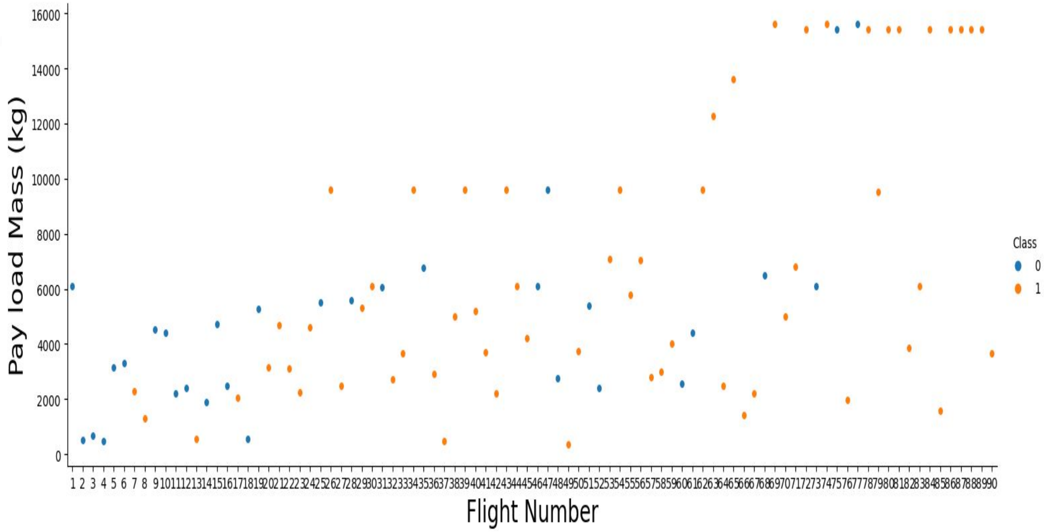
EDA - RELATIONSHIPS BETWEEN VARIABLES





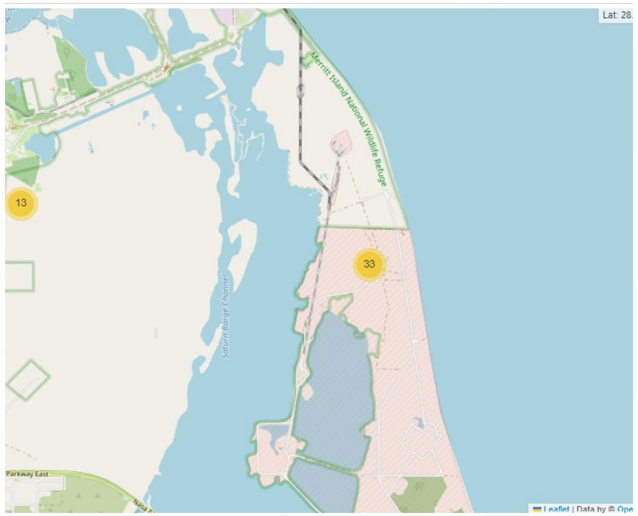


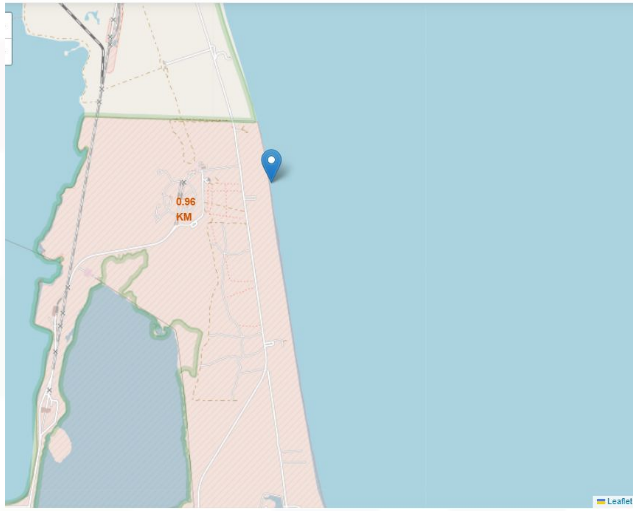


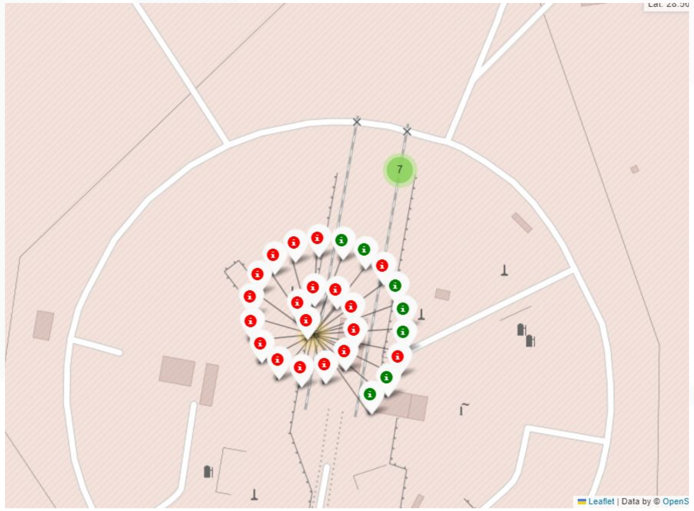


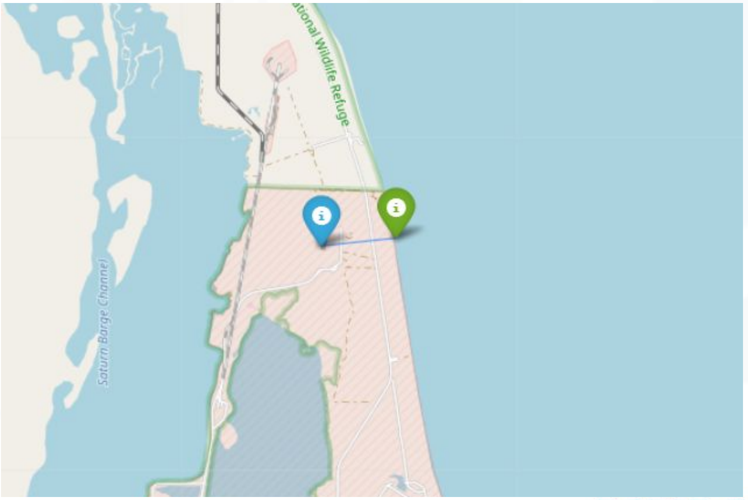
FOLIUM MAPS


Ottawa
Toronto
New York
Washington
United States
Phoenix
México
Ciudad de México
La Habana
Cuba
The Bahamas
Republica Dominicana
Leaflet | Data by © OpenStree


13
33
Merritt Island National Wildlife Refuge
Saturn Barge Channel
Parkway East
Leaflet | Data by © Ope


0.96 KM
Leaflet


7
Leaflet | Data by © OpenS


Saturn Barge Channel

RESULTS FROM THE EDA ANALYSIS

- Flights with payload mass above 8000 kg are more successful.
- Launch Site 'CCAFS SLC 40' has more failures.
- We can see that for any launch site, if payload mass is above 8000 kg is mostly successful.
- If the orbits are ES-LI, GEO, HEO and SSO, Success rate is high.
- Irrespective of the payload mass, Orbit SSO has 100% success rate.
- The success rate over years is shooting up.

RESULTS FROM SQL ANALYSIS

- There are several columns in the dataset such as Date Time (UTC), Booster_Version,Launch_Site, Payload, PAYLOAD_MASS__KG_, Orbit, Customer, Mission_Outcome and Landing_Outcome with respective meanings.
- There are 4 launch sites in the dataset like CCAFS LC-40, VAFB SLC 4E, KSC LC-39A and CCAF SLC-40.
- The average mass of the payloads is 2928 kgs.
- The first successful launch was happened on 22/10/2015.
- Boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000 are F9 FT B1022, F9 FT B1026,F9 FT B1021.2 and F9 FT B1031.2.
- 21 times falcon 9 has carried NASA(CSR) boosters and settled in different orbits.

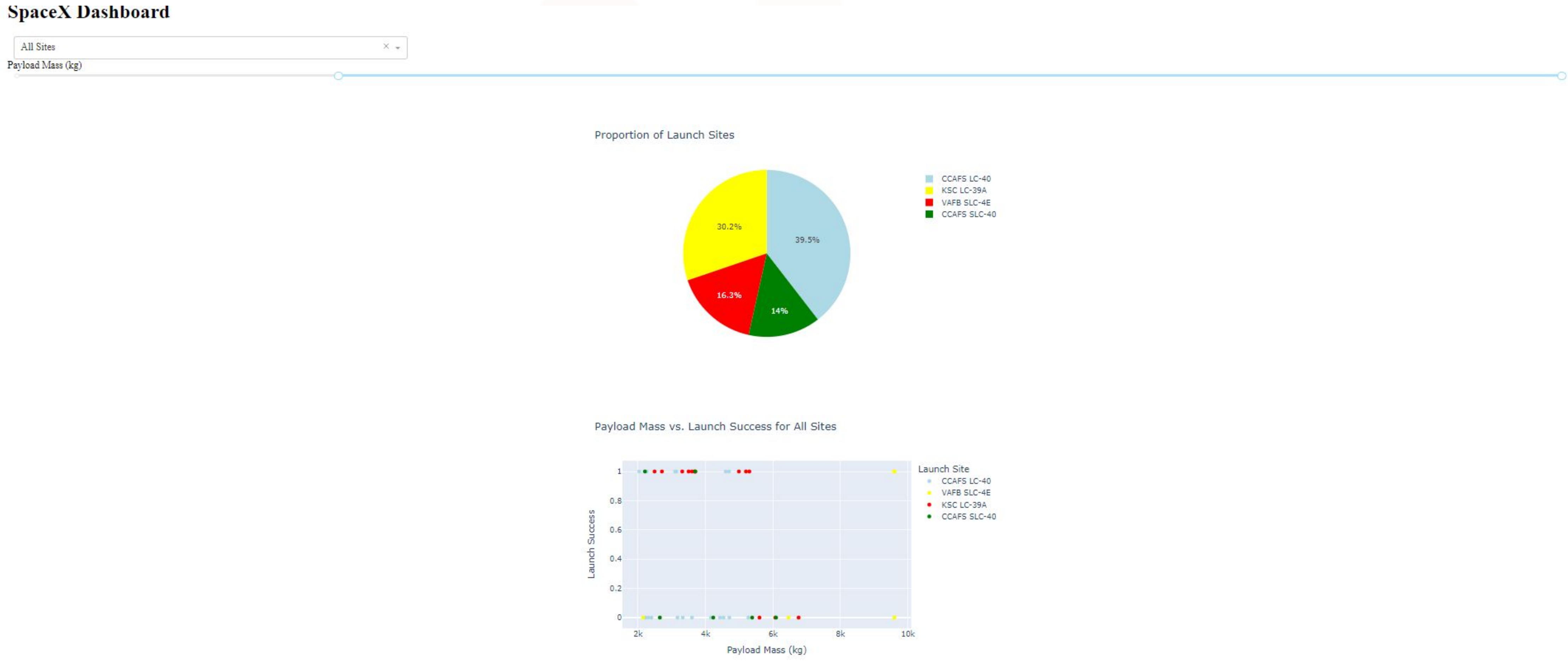
RESULTS FROM SQL ANALYSIS

- The success rate over years is shooting up.
- The total number of successful and failure mission outcomes are 98 and 1 respectively.
- The names of the booster_versions which have carried the maximum payload mass are 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 and F9 B5 B1049.7
- The count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 are 61 and 40

LINK DASHBOARD

http://127.0.0.1:8050/

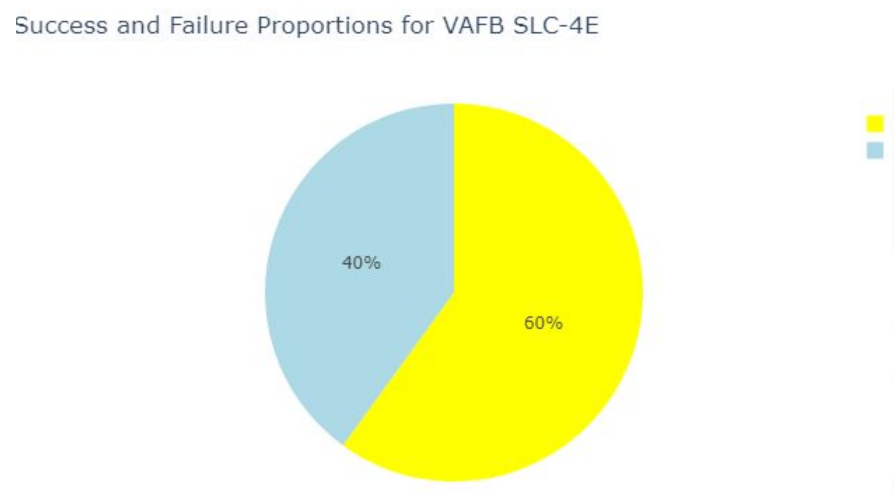
DASHBOARD VIEW

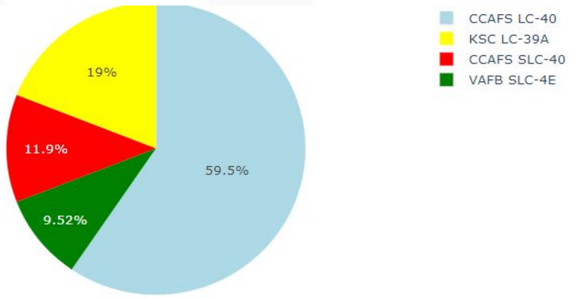


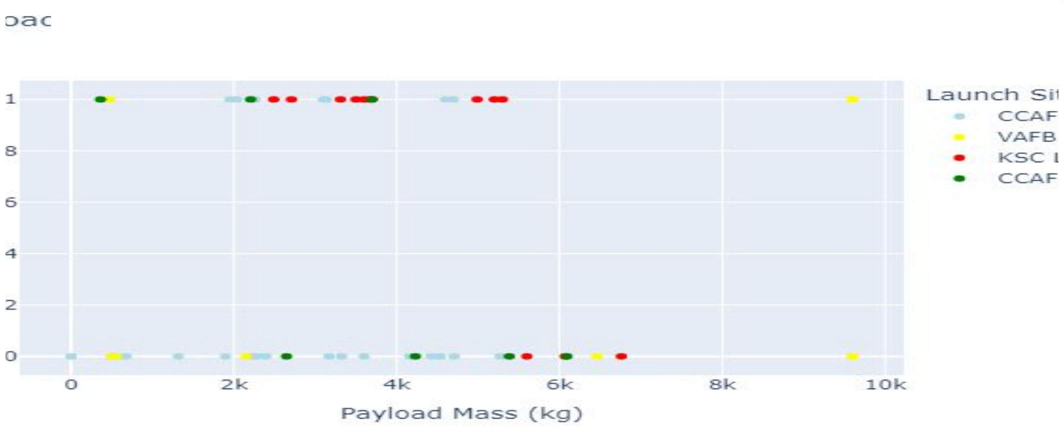
IBM Developer

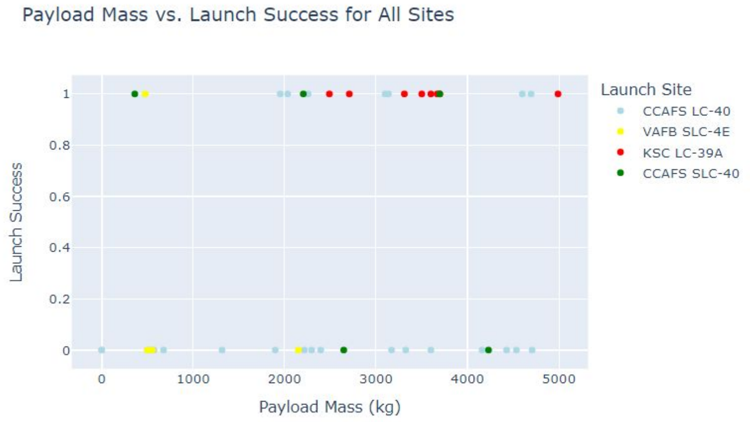
SKILLS NETWORK

SUCCESS PROPORTION FOR DIFFERENT PAYLOAD MASSES AND LAUNCH SITES



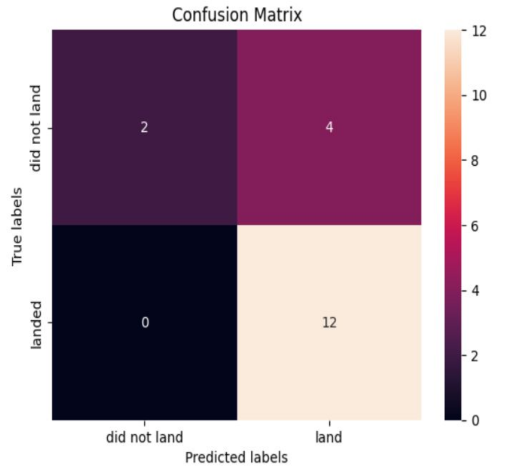




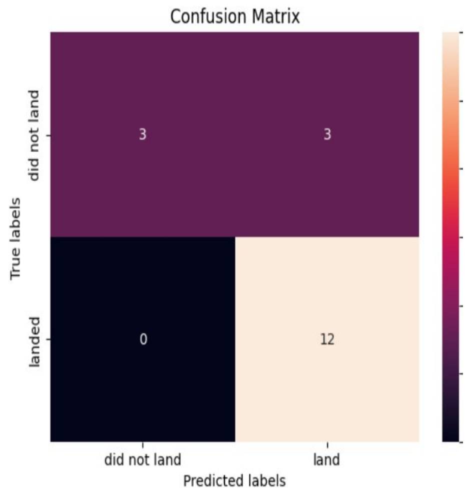


CONFUSION MATRIX FOR LR,SVM,TREE AND KNN

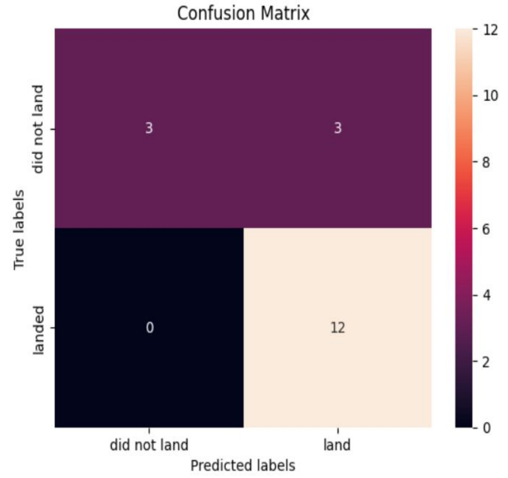
```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



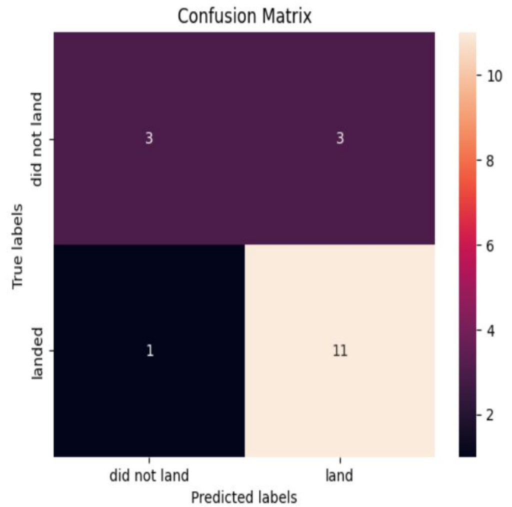

```
yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```




```
In [25]: yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```




```
yhat=svm_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



DECISION TREE	KNN	LOGISTIC REGRESSION	SVM

ACCURACY TEST SAYS KNN AND LR FIT BETTER

```
# Assuming you have already performed the necessary steps for each algorithm and obtained the tuned models:
# logreg_cv, svm_cv, tree_cv, and knn_cv.

# Calculate the accuracy on the test data for each tuned model
accuracy_logreg = logreg_cv.score(X_test, Y_test)
accuracy_svm = svm_cv.score(X_test, Y_test)
accuracy_tree = tree_cv.score(X_test, Y_test)
accuracy_knn = knn_cv.score(X_test, Y_test)

# Print the accuracy for each method
print(f"Accuracy for Logistic Regression: {accuracy_logreg:.2f}")
print(f"Accuracy for Support Vector Machine: {accuracy_svm:.2f}")
print(f"Accuracy for Decision Tree: {accuracy_tree:.2f}")
print(f"Accuracy for K Nearest Neighbors: {accuracy_knn:.2f}")
```

```
Accuracy for Logistic Regression: 0.83
Accuracy for Support Vector Machine: 0.78
Accuracy for Decision Tree: 0.78
Accuracy for K Nearest Neighbors: 0.83
```

CONCLUSION

- THE LAUNCH SITES ARE FAR AWAY FROM CITIES AND CLOSER TO COASTLINE.
- HEAVY PAYLOADS ARE SUCCESSFUL THAN LOW MASS PAYLOADS
- LOGISTIC REGRESSION AND KNN MODELS ARE THE BET MODELS FOR THIS CLASSIFICATION.
- SPACE_X FALCON 9 PROJECT'S SUCCESS GRAPH TRENDS UPWARD OVER YEARS.