***Final Presentation Capstone Project***

* Executive Summary

1. Summary of methodologies
2. Summary of all results

* Introduction

1. Project background and context
2. Problems you want to find answers

* ***Section 1***: Mehtodology

1. Describe how data was collected
2. Describe how data was processed

**Intro:** if determine the **first stage landing**, we can determine the **cost of a launch**.

**Objective:** Determine the ***price of each launch***, also determine if SpaceX will reuse the first stage.

You will train a **machine learning model** and use public information to predict if SpaceX will reuse **the first stage**.

**Problematic:**

* ***Data Collection (1st lab):***

Using an API (SpaceX REST API), to gather information about **rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.**

**This in order** to predict whether SpaceX will attempt to land a **rocket or not.**

* ***Process of collecting data:***

Get data from **an url**, then the response will be in the form of a **JSON,** specifically a list of **JSON** objects.

1. Then use, a **json-normalize** function to convert the **file into** a ***dataframe***. It will “normalize” the **structured json data into a flat table.**
2. Use ***Python BeautifulSoup package*** to web scrape some HTML tables that contain valuable **Falcon 9 launch records**.
3. Need to parse data **from those tables and convert** them into a ***Pandas*** data frame for ***further visualization and analysis.***

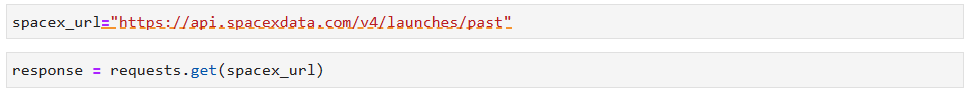
With those functions (**Booster, Launchpad, payload, and core**).

1. Calculate mean of the **PayloadMass** data, then replace null values in PayloadMass with the mean.

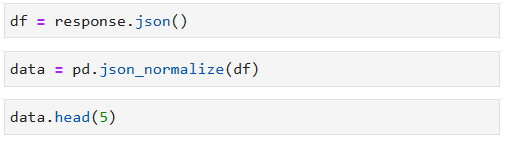
1. **Objective lab 1: Collect and make sure** the data is in the correct format from an API.
2. ***Data collection and filter of Columns (variables)***

(Add: imported libraries and functions used).

**Data** was requested from data of ***SpaceX API*** using a URL, and



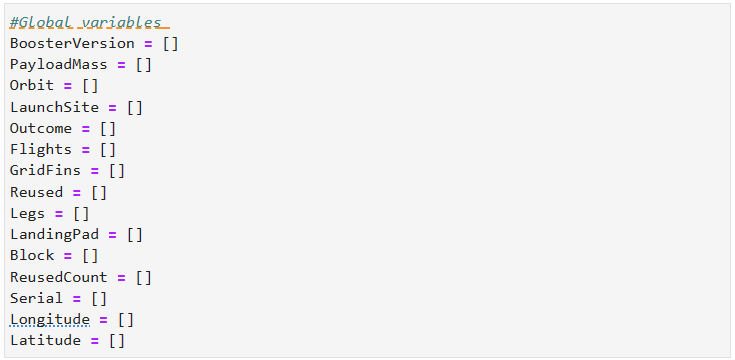
**Then we decode** the response content as a Json using.json() and turn it into a Pandas dataframe using.json\_normalize().

******

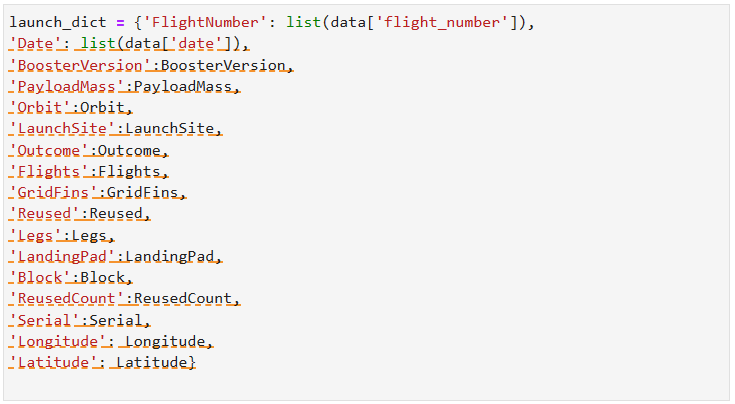
After **filtrating some columns** that will be used for this **project (rocket, payloads, launchpad, and cores)**, and doing the following tasks:

* Remove **rows with multiple cores,**
* Extract the single value in the list and replace the feature.
* Convert the date\_utc to a datetime datatype and then extract the date, leaving the time.
* Use date, to restrict dates for the launchs.

Then the **data requested from** the different **columns,** exposed before will be stored in the following **lists, that will be used** to create a new **dataframe.**



The columns then have been combined **into a dictionary (launch\_dict).**

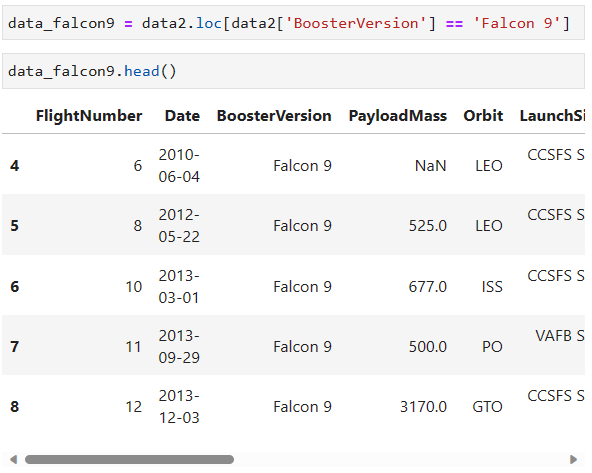


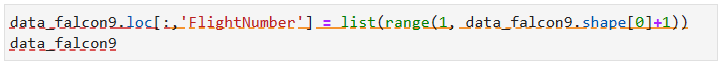
This dictionary, has been after used to create a new **Pandas data frame,** with 17 columns from which **16 are variables from the flight number** and **90 flights**.



1. ***Filter of data and inclusion of Falcon 9 launches***

Afterwards, we had filtered the data to a new data frame called **data\_falcon9. And secondly,** we had made the reset of the ***FlightNumber column.***





1. ***Data Wrangling (deal with missing values)***

For the **missing values columns,** such as **PayloadMass** we will **compute the mean and** replace **np.nan values** in the data with the mean that has been **calculated.**

Regarding the **values of Landing Pad,** we had leaved the missing values without any change.

1. ***Exporting values as CSV***

Once these tasks done, we had **exported the document as a CSV.**



1. Perform exploratory data analysis(EDA) using visualization and SQL

* ***Data Wrangling EDA (Exploratory Data Analysis):***

**Objectives:** find some patterns in the data and determine what would be the label for training supervised models. This by analyzing the scenarios where, **booster did not land successfully (refer to figure X).**

Imported Libraries….

1. **Exploratory Data Analysis using Visualization**

We used the Space X **dataframe from last section for this** task:

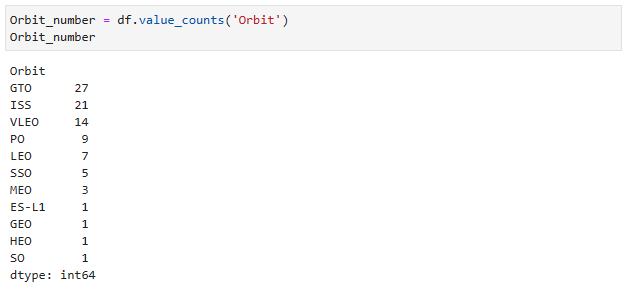
****

Afterwards, verifying the information of the data set processed before. We computed the number of **launch sites** by using the method **value\_counts(). This because** each launch aims to a **dedicated orbit.**

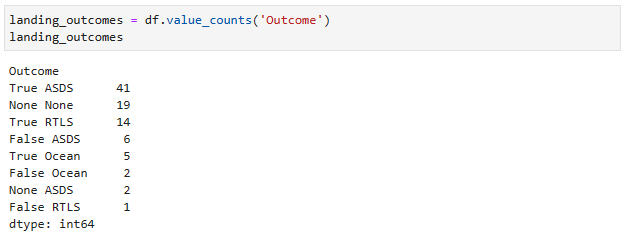


This data allowed us to count also the **number and occurrence of each orbit, and the number and occurrence of mission outcome of the orbits.**

***Number and occurrence of each orbit:***

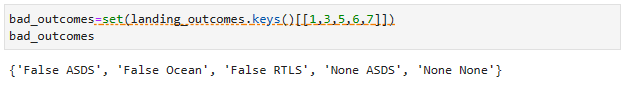


***Number and occurrence of mission outcome of the orbits:***



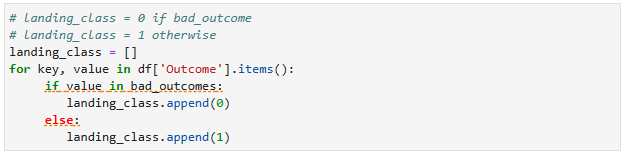
Where **good outcomes are (True Ocean, True RTLS, True ASDS)** and **bad outcomes are (False Ocean, False RTLS, False ASDS, None ASDS, None None).**

Therefore, we had created a set of **columns where the second stage did not land successfully.**

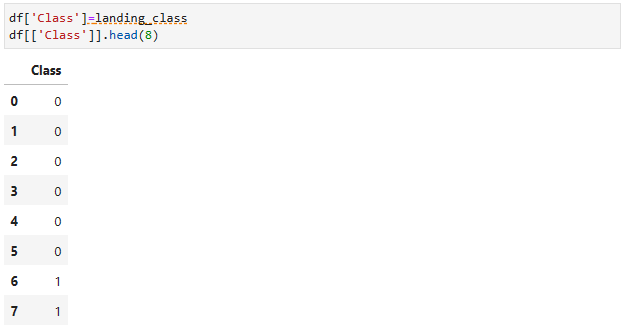


* Creation a **landing outcome label from Outcome column**

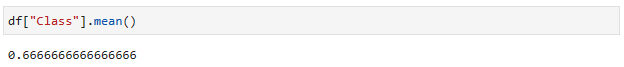
Thus, in order to simplify **bad outcomes,** we **created a list** where the element is **zero if the corresponding row is** in **bad\_outcome.** Otherwise, is **one.**



**In other terms, value of zero is** when the **first stage** did not land successfully, **one means** the **first stage landed Successfully.**



Afterwards, we define the **success rate using the following line:**



Finally, we had **exported the data frame** into a **CSV.**

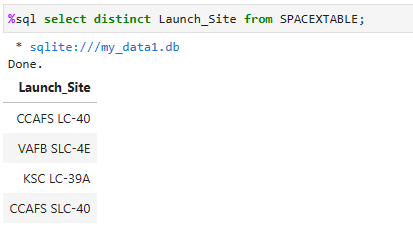


1. **Exploratory Data Analysis using SQL**

First, we required to **load the SQL extension and establish a connection** with the database.

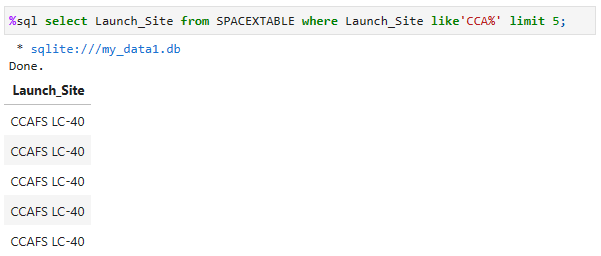


After, we had displayed the names of the **unique launch sites in the space mission:**

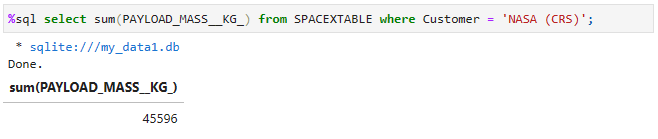


We had analyzed the following domains on the **data frame:**

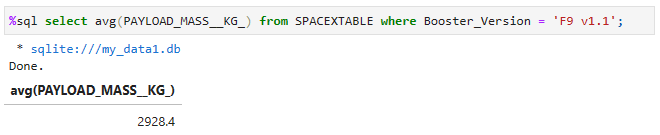
* **5 records where launch sites begin with the** string **‘CCA’,**

****

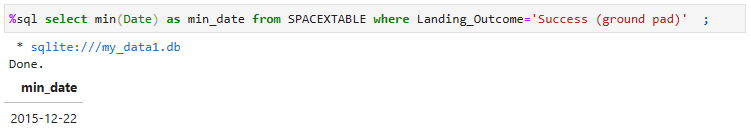
* **total payload mass carried by** boosters launched by **NASA (CRS),**

****

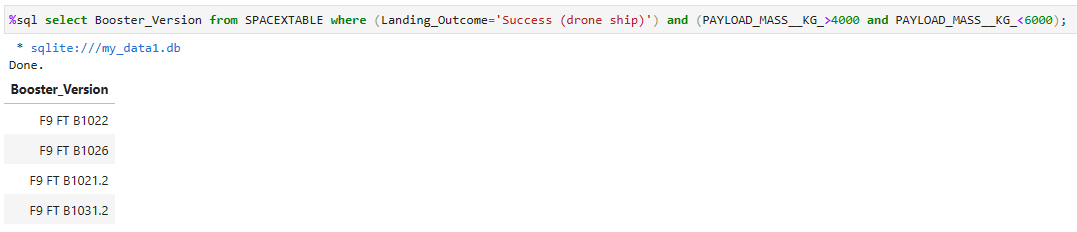
* display **average payload mass** carried **by booster** version F9 v1.1,

****

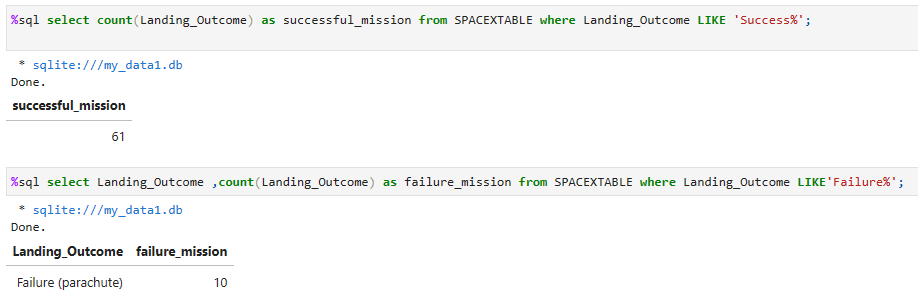
* List the **date when the first successful** landing outcome in ground pad was **achieved,**



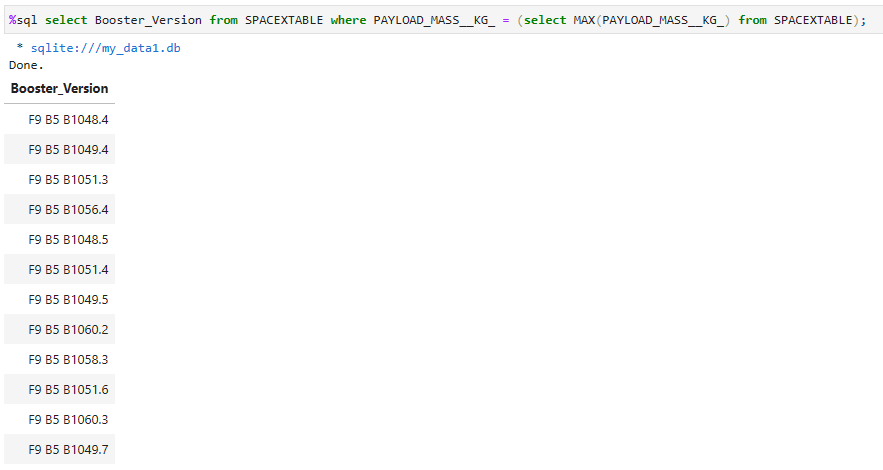
* **List** the **names of the boosters** which have **success** in drone ship and have **payload mass** greater than 4000 but less than 6000,

****

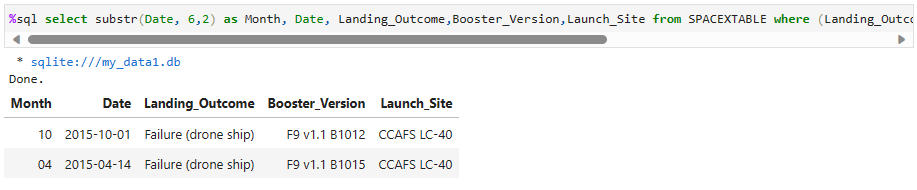
* **List the total number** of successful and failure **mission outcomes,**

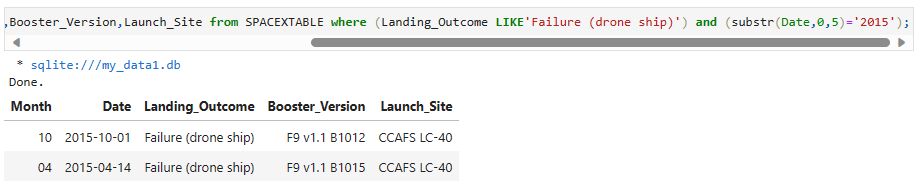


* **List the names** of the **booster\_versions** which have carried the **maximum payload mass (Subquery),**

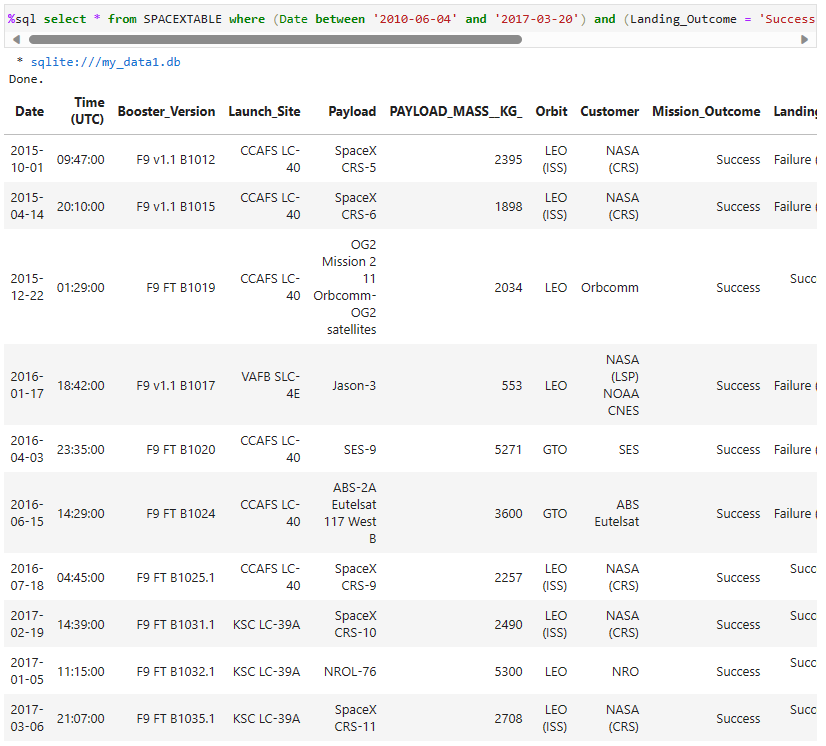


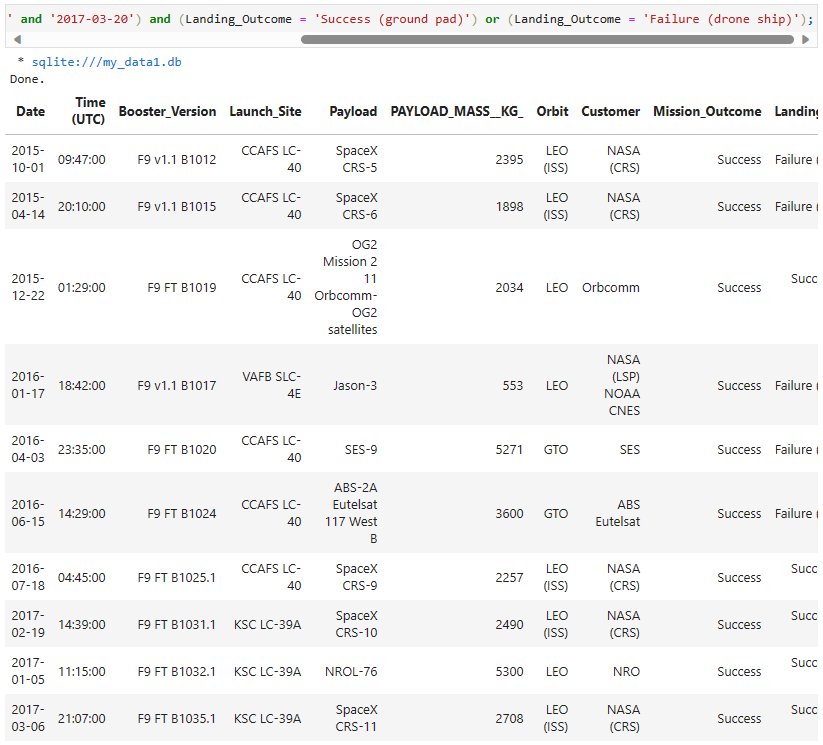
* List records that will display **the month names, failure landing\_outcomes in drone ship, booster versions, launch\_site** for the **month in year 2015,**

****

****

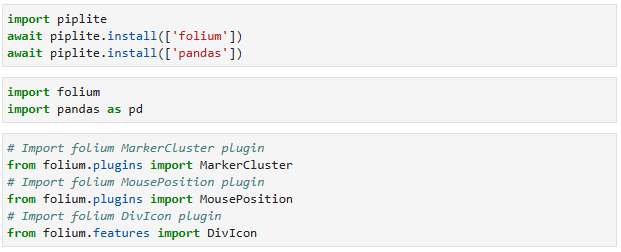
* **Rank the** count of landing **outcomes (such as Failure(**drone ship**)) or Success (**ground pad**) between** the date 2010-06-04 and 2017-03-20, **in descending order.**



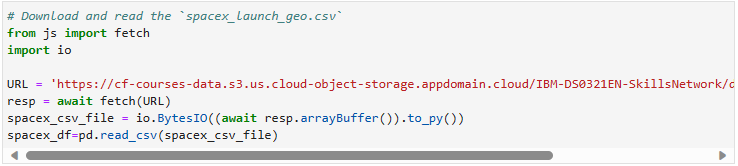


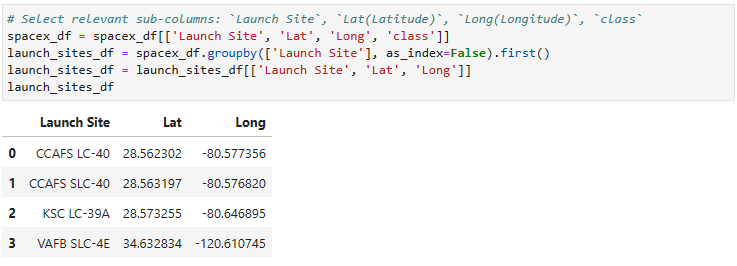
1. Perform interactive visual analytics using Folium and Plotly Dash

Import libraries.



Once, the required libraries are download, we **had added site´s location** on a map using **site´s latitude and longitude coordinates. Also,** we had created a new **name spacex\_launch\_geo.csv** augmented with latitude and longitude, added for each site.

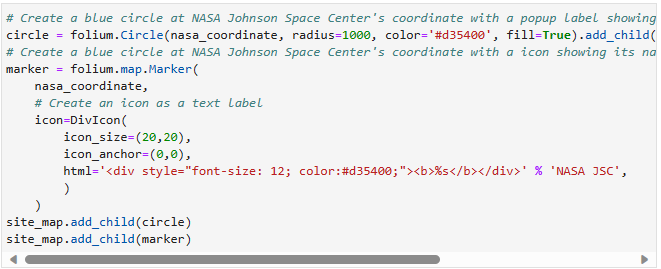


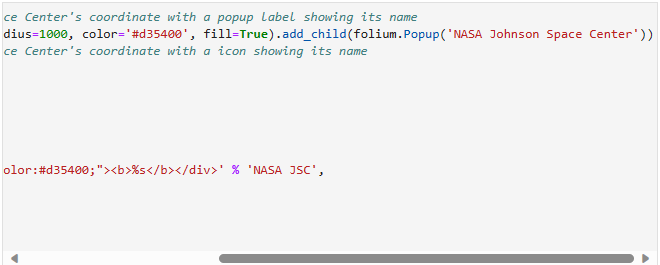


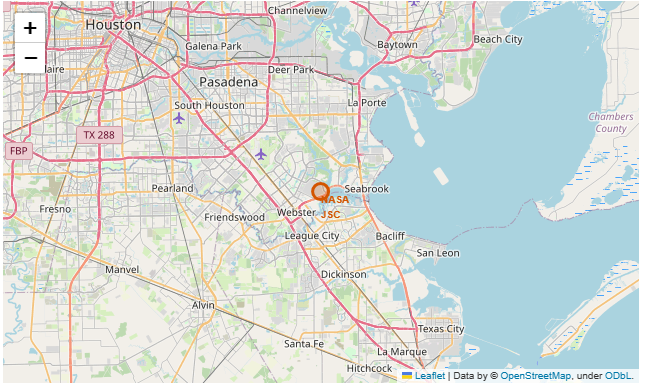
* Aterwards, we **pinned those locations** on a **map.**

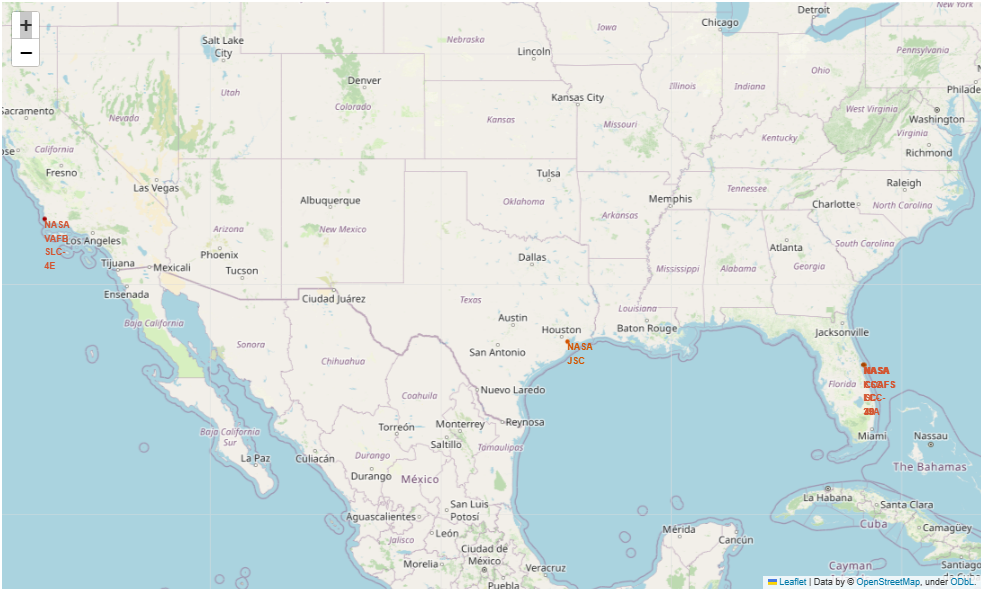


We had used, the **folium.Circle** and **folium.Marker** to add highlighted circle area with a text label **on a specific coordinate. This has** been used for each **launch site on the map.**



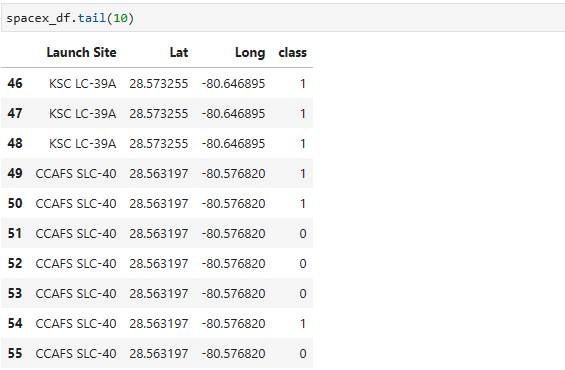




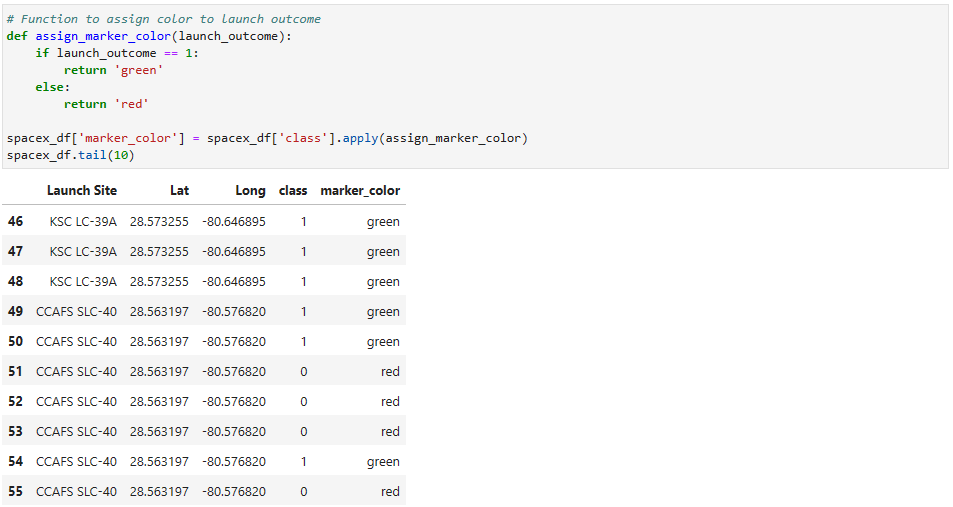


* After this is made, we **had marked the success/failed for each site on the map,**

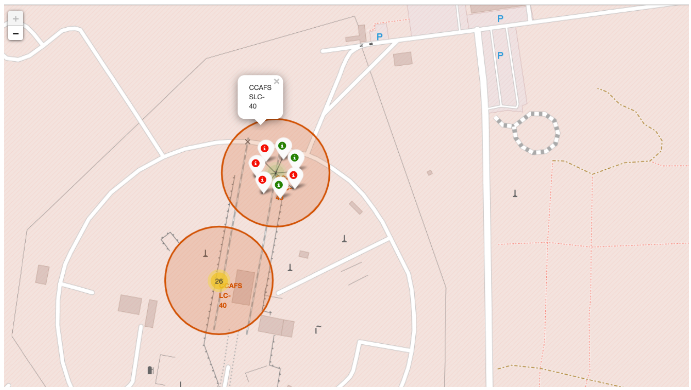
in order to have a view on the **success rate** regarding each site. For this purpose, we had **used the following** data set. Where a **successful launch (class=1)** and **failed launch (class=0). And** the use of a **MarkerCluster** object.





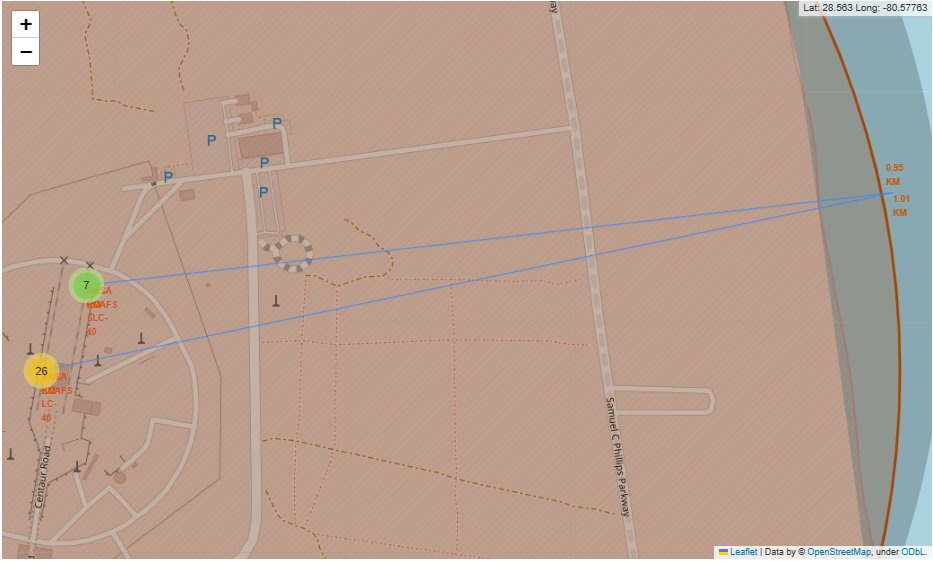


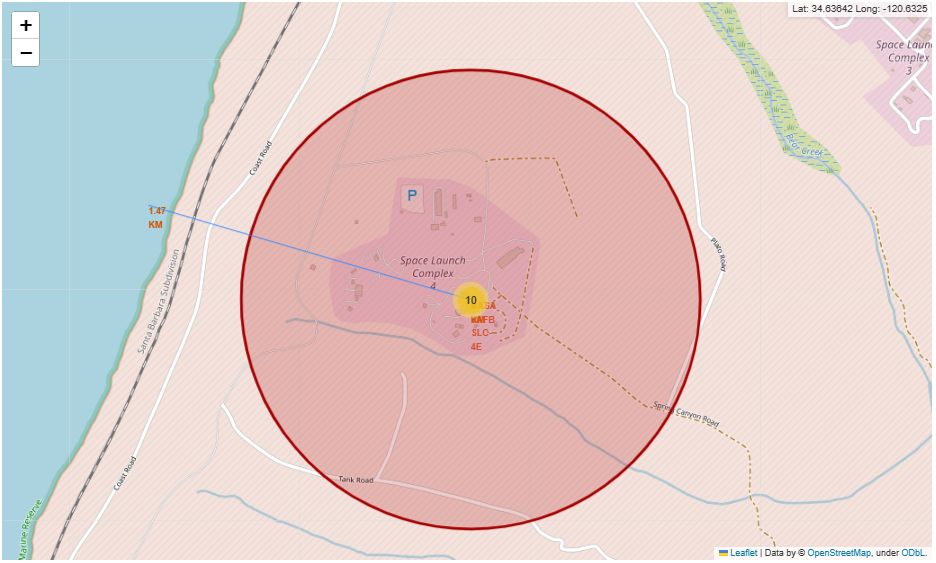
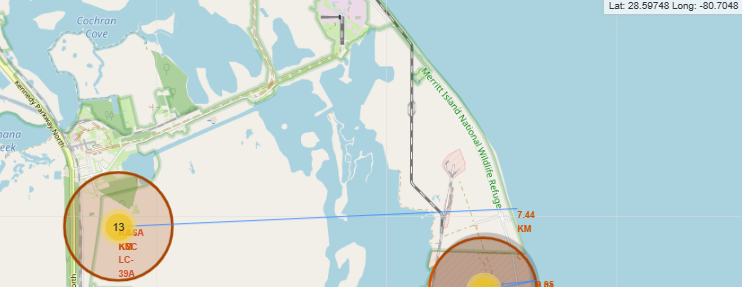
And for each **launch result** in **spacex\_df** data frame, we had added **folium.Marker** to **marker\_cluster. In order,** to identify ***high success rates launch sites***.



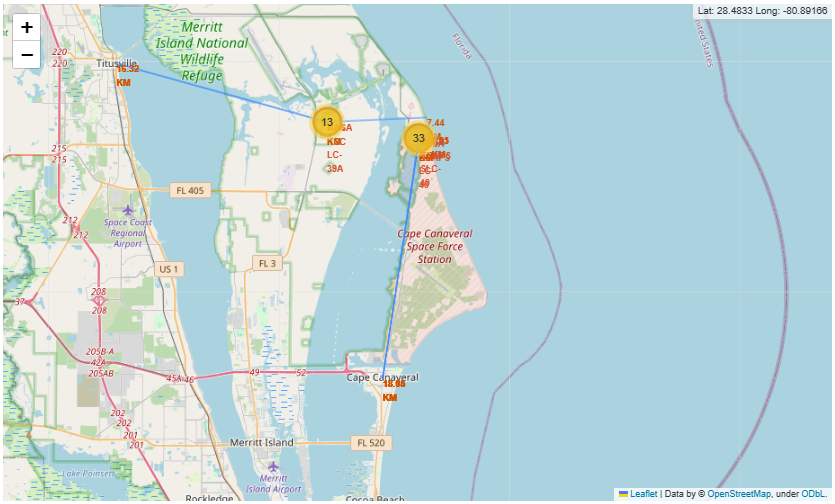
Following this addition, we had **drawn a PolyLine** between a **launch site** to the selected coastline point, closest city, railway and highway.

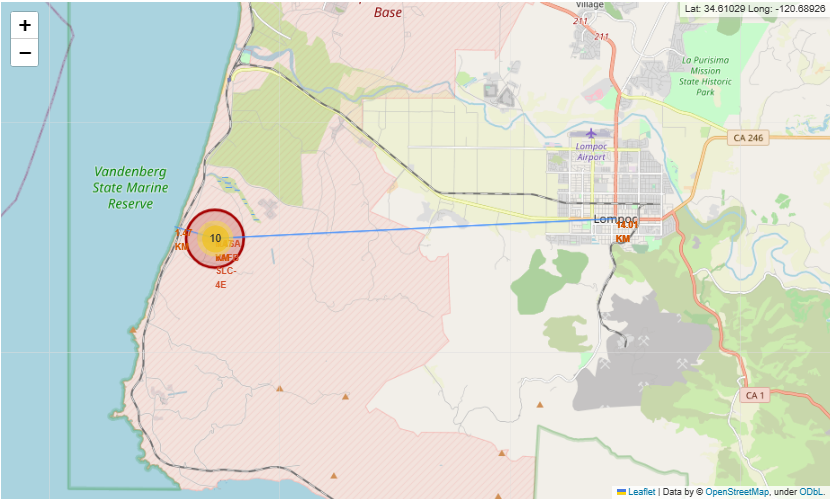
***Coastline point:***



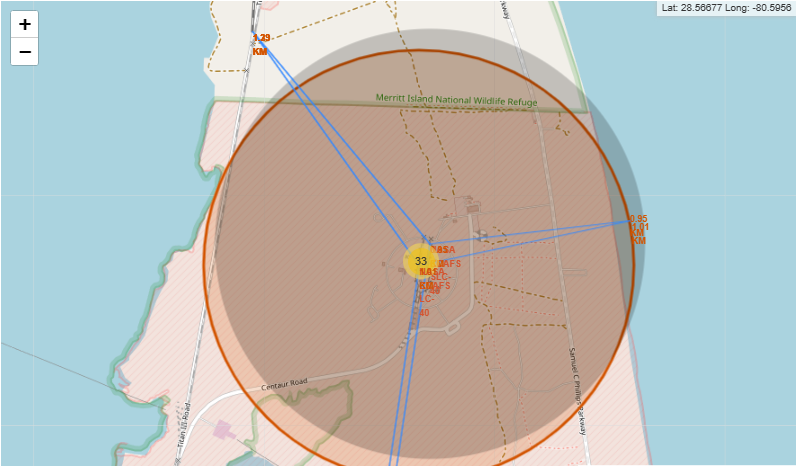


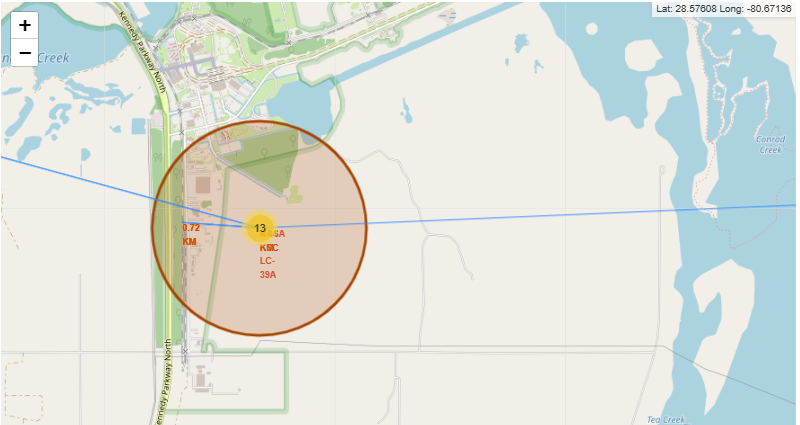
***Closest city:***

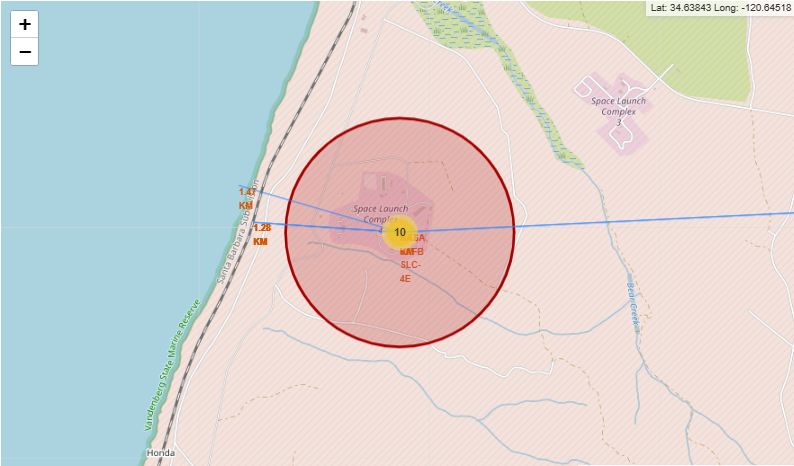




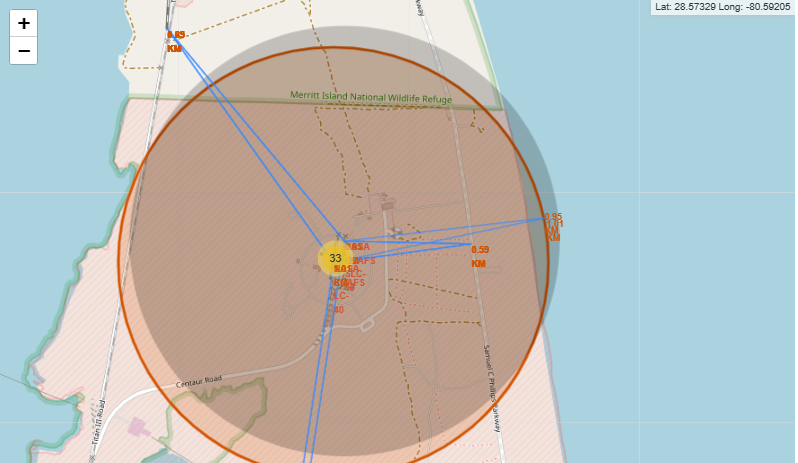
***Railway:***

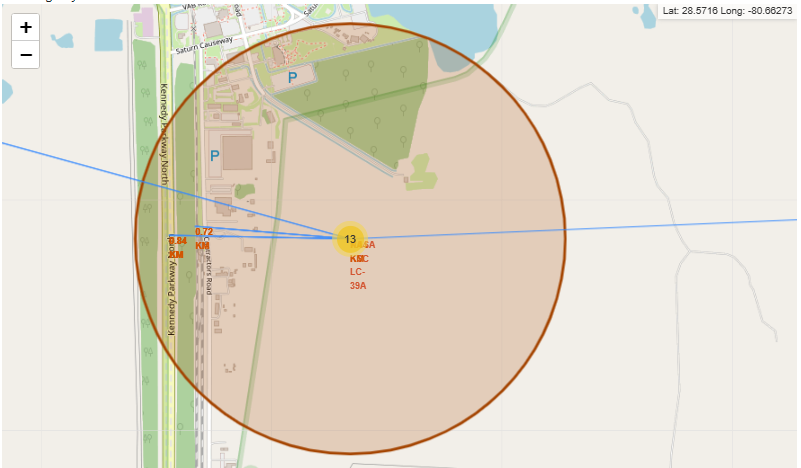






***Highway:***







* ***Build an Interactive Dashboard with Plotly Dash***

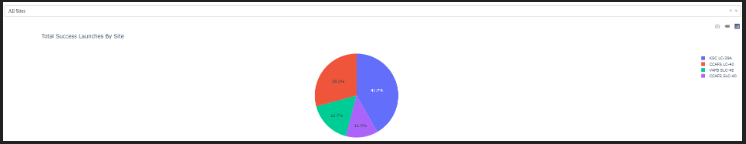
We will build a ***Plotly Dash application*** in order to perform interactive ***visual analytics on SpaceX*** launch data in real-time. This **application will contain** input components such as a ***dropdown list and a range slider*** to interact with a ***pie chart and a scatter point chart.***

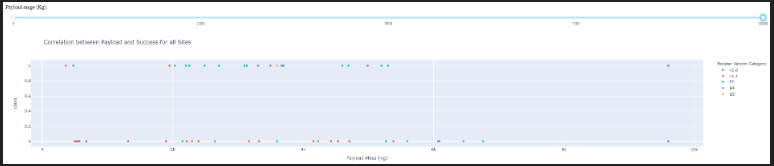
This ***application*** helped us to answer to the following questions:

1. Which site has the largest successful launches?
2. Which site has the highest launch success rate?
3. Which payload range(s) has the highest launch success rate?
4. Which payload range(s) has the lowest launch success rate?
5. Which F9 Booster version **(v1.0, v1.1, FT, B4, B5, etc.)** has the highest  
   launch success rate?

After installing the ***Python Packages required,*** we had started to create the **dashboard application and dataset.**







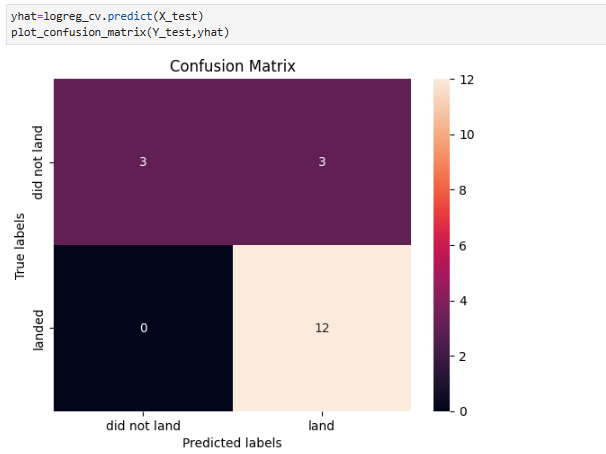
1. Perform predictive analysis using classification models.

(p117)

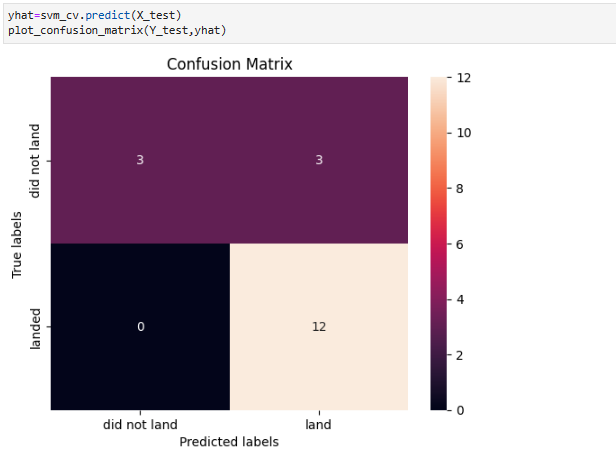
This ***machine learning pipeline,*** will predict if the first stage will land given the data on which we were working before.

***Requirements***: import Libraries and Define Auxiliary Functions, and define the **function** to plot the confusion matrix. This function will be called, ***plot\_confusion\_matrix.***

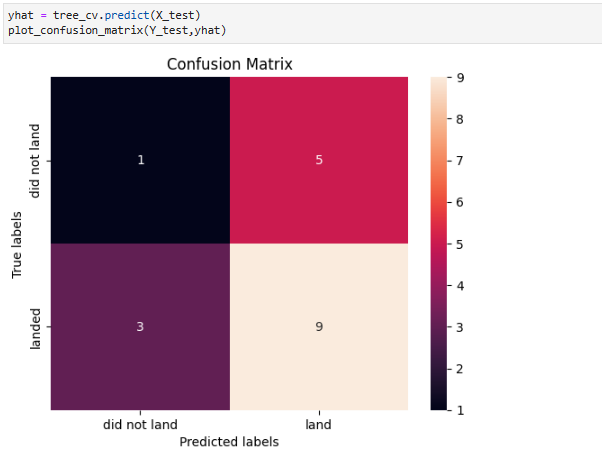
After **loading the data**, we used the following **classification models**: ***logistic regression,***



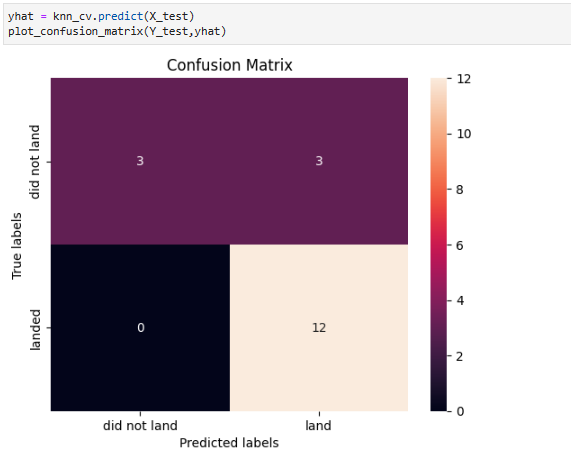
***Support vector machine,***

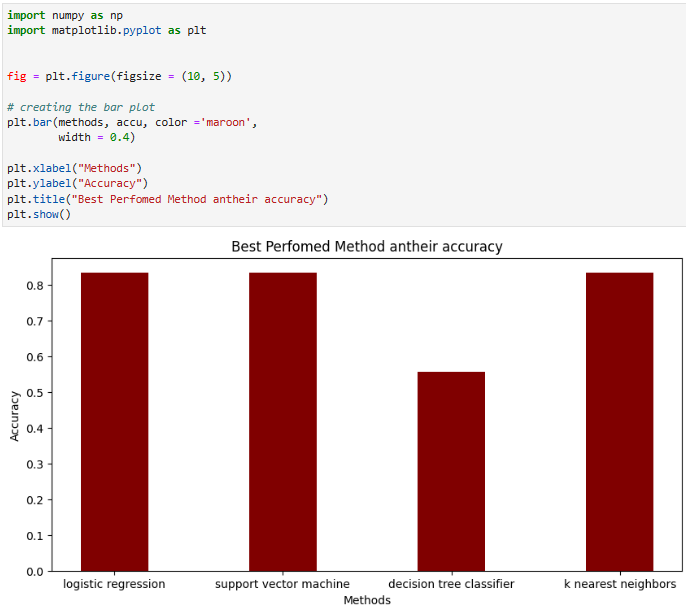


***decision tree classifier,***



***k nearest neighbors***.





1. Results

* ***Section 2:*** Insights drawn from EDA

1. Flight Number vs Launch Site
2. Payload vs Launch Site
3. Success Rate vs Orbit Type
4. Flight Number vs Orbit Type
5. Payload vs Orbit Type
6. Launch Success Yearly Trend
7. All Launch Site Names
8. Launch Site Names Begin with ‘CCA’
9. Total Payload Mass
10. Average Payload Mass by F9 v1.1
11. First Successful Ground Landing Date
12. Successful Drone Ship Landing with Payload between 4000 and 6000
13. Total Number of Successful and Failure Mission Outcomes
14. Boosters Carried Maximum Payload
15. 2015 Launch Records
16. Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

* ***Section 3:*** Launch Sites Proximities Analysis

1. Folium Map Screenshot 1
2. Folium Map Screenshot 2
3. Folium Map Screenshot 3

* ***Section 4:*** Build a Dashboard with Plotly Dash

1. Dashboard Screenshot 1
2. Dashboard Screenshot 2
3. Dashboard Screenshot 3

* ***Section 5:*** Predictive Analysis (Classification)

1. Classification Accuracy
2. Confusion Matrix
3. Conclusions
4. Appendix