IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Waseem Ahmad
28.10.2021

Outline

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

Executive Summary

- Summary of methodologies

Space X data related to launches is analysed using Juypter note book in IBM Skills Network Cloud as well on IBM Cloud DB2 and Watson Studio. Data was gathered and cleaned on past launches and its exploratory analysis was performed. Machine learning classification modelling of data was done using logistic regression, Support Vector Machines, Tree Classification and KNN algorithms with python library Sklearn.

- Summary of all results

Space X cost is almost half of other companies as it can reuse part of rocket and launch vehicle. Launch sites are located near equator and sea. A higher mass of payload increases success of launch (Falcon 9 and above series). Machine learning classification models are capable of predicting with 83% of accuracy for successful launch. EDA analysis also highlight a very high success probability for higher pay load from Launch Site CCAFS SLC 40. In nutshell, a higher Payload mass, CCAFS SLC site and use of data science machine learning tools to select features increase successful launch and can reduce costs.

Introduction

- Project background and context
- Space X or Space Exploration Technologies Corporation is owned by Elon Musk, who is also the owner of PayPal and Tesla. SpaceX places satellites in the orbit and also delivers cargo to ISS (International Space Station). The cost of SpaceX is considerably low compare to others as they have successfully reused rocket and launch system.

- Problems you want to find answers
- In this datasicence capstone project, we are analysing data of past launches of SpaceX to find the important factors in saving the first part of the launch rocket and identify the factors can be used to predict successful launch. We will be doing exploratory data analysis as well as develop machine learning models to predict outcome. We will also calculate the accuracy of our prediction.

Section 1

Methodology

Methodology

- Following is the summary of Methodolgy for all parts.
- Data is collected from SpaceX rest API and from wiki pages, it is wrangeld (cleaned) for further Exploratory, Visualization and Machine learning modelling analysis.
- Pandas, Numpy, Matplotlib, Seaborn, Folium, Plotly, Dash, Sklearn are the main python packages used in this project.
- IBM Wastson studio cloud, IBM DB2 cloud are used to perform analysis and Github repository is used to store note books for peer evaluation.
- Proess applied on data include, making scatter plots, bar charts, interactive maps and charts, quering data using sql as per various conditions and developing KNN, Tree classifier, SVM and logistic regression models.
- Based on findings and analysis, understanding how spaceX has achieved successful launches at a very low cost compared to its competitors and also being able to predict using features if the launch first part will be successfully recovered on or not.

Data Collection

- Data is collected from Space X Rest API which is api.spacexdata.com/v4
- There are many end points in this API like api.spacexdata.com/v4/capsules or api.spacexdata.com/v4/cores
- We will be using the end point ‘api.spacexdata.com/v4/launches/past’ to collect the data.
- Data about launches include rocket used, payload delivered, launch specifications and landing outcome. Our goal is to predict whether SpaceX will attempt to land a rocket or not.
- The github link for the data collection notebook is as below
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/7-jupyter-labs-spacex-data-collection-api.ipynb

Data Collection – SpaceX API

- Data collection with SpaceX REST calls using requests library
- We will use SpaceX rest API; https://api.spacexdata.com/v4/launches/past to get data using request library as
- Response = request.get(url) and we get a response which is JSON object. We can view JSON object using response.json() and can convert / normalise it as

  data=pd.json_normalise(response.json())

- The github link for data collection notebook is as below
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/7-jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

Web Scraping Process and Use of given functions to collect data and process a clean Dataframe

- Data is collected from wikipedia pages of SpaceX for Falcon 9 launches.
- Python BeautifulSoup package will be used to web scrape data in HTML tables. This data will be then parsed in Python Pandas dataframe.
- The specific data from target endpoints will be gathered with the given functions; getBoosterVersion(), getLaunchSite(), getPayloadData(), getCoreData()
- These functions will get the id from web scraped table and then get the data from SpaceX rest API as per specific ID. The data gathered by these functions will be stored in lists and used to create a dataset for further analysis.
- The github link for notebooks having Data collection and web scraping are as below
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/labs-jupyter-spacex-Data%20wrangling.ipynb
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/7-jupyter-labs-spacex-data-collection-api.ipynb

Data Wrangling

- Data cleaning of wrangling involves analysing and making sure data set has variables in the required format (numeric, factors, strings etc) as well as dealing with nulls.
- Once the dataframe was made after webscraping, we filtered the data to include on Falcon 9 launches.
- We also checked for the missing values. There were 5 missing values for PayLoadMass and 26 for Landing Pad.
- PayLoadMass mean value was calculated as

data_falcon9.PayloadMass.mean()

- This mean values is used to replace the missing values as

data_falcon9['PayloadMass'].fillna(value=data_falcon9['PayloadMass'].mean(), inplace=True)

- We also change the bad outomes to 0 and successful outcomes to 1 for Landing class variable.
- The github link for notebooks having Data collection and web scraping are as below
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/labs-jupyter-spacex-Data%20wrangling.ipynb
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/7-jupyter-labs-spacex-data-collection-api.ipynb

EDA with Data Visualization

- I have used pandas, numpy, matplotlib and seaborn libraries of python for EDA.
- 4scatter plot were prepared to analyse relation ship between PayLoadMass and Flight Number, Launch Site and Flight Number,Launch Site and Payload Mass, Flight No and Orbit, and PayLoad Mass and Orbit. It is found that higher payload mass increases chances of success landing of rocket. VAFB SLC 4E and CCAFS SLC 40 has higher success rates.
- A bar chart of Mean outcome vs orbit was also plotted that shows outcome of 1 for ES L1, SSO, HEO or GEO orbits.
- A line chart of success rate vs years shows that successful launhes increase consistently since 2013
- The github link of note book for EDA and Data Visualization is as
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- I connected my notebook on IBM Watson studio cloud to my datafile spacex.csv table on DB2 cloud, using service credentials of DB2.
- I assessed the the table from note book using SQL magic and performed 10 Tasks. The 10 SQL querries are 10 Tasks given in the note book. The first three sql querries are as and remainig can be checked on the notebook uploaded:
- %sql select DISTINCT Launch_Site from ABC
- %sql select * from ABC where Launch_Site like 'KSC%' Limit 5
- %sql select SUM(PAYLOAD_MASS__KG_) from ABC where Customer = 'NASA (CRS)‘
- The note book having all the EDA with SQL is present at the following github address
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/jupyter-labs-eda-sql-edx.ipynb

Build an Interactive Map with Folium

- We used Folium Library and its dependencies to viualise location data on world map.
- Markers are added with pop up labels to identify the launch locations.
- Various pops are added to markers to visually see success rates of each launch site.
- Distance from coast line, nearyby highway and city was also visualized on the map.
- The notebook having interactive maps built using follium is present at this github address
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Plotly / Dash is used to add interactivity to visual graphs and increases the understaning and finding insights for the user.
- These visuals objects contain sliders, dropdown list etc which help the reader interact to the graph and see in depth the information.

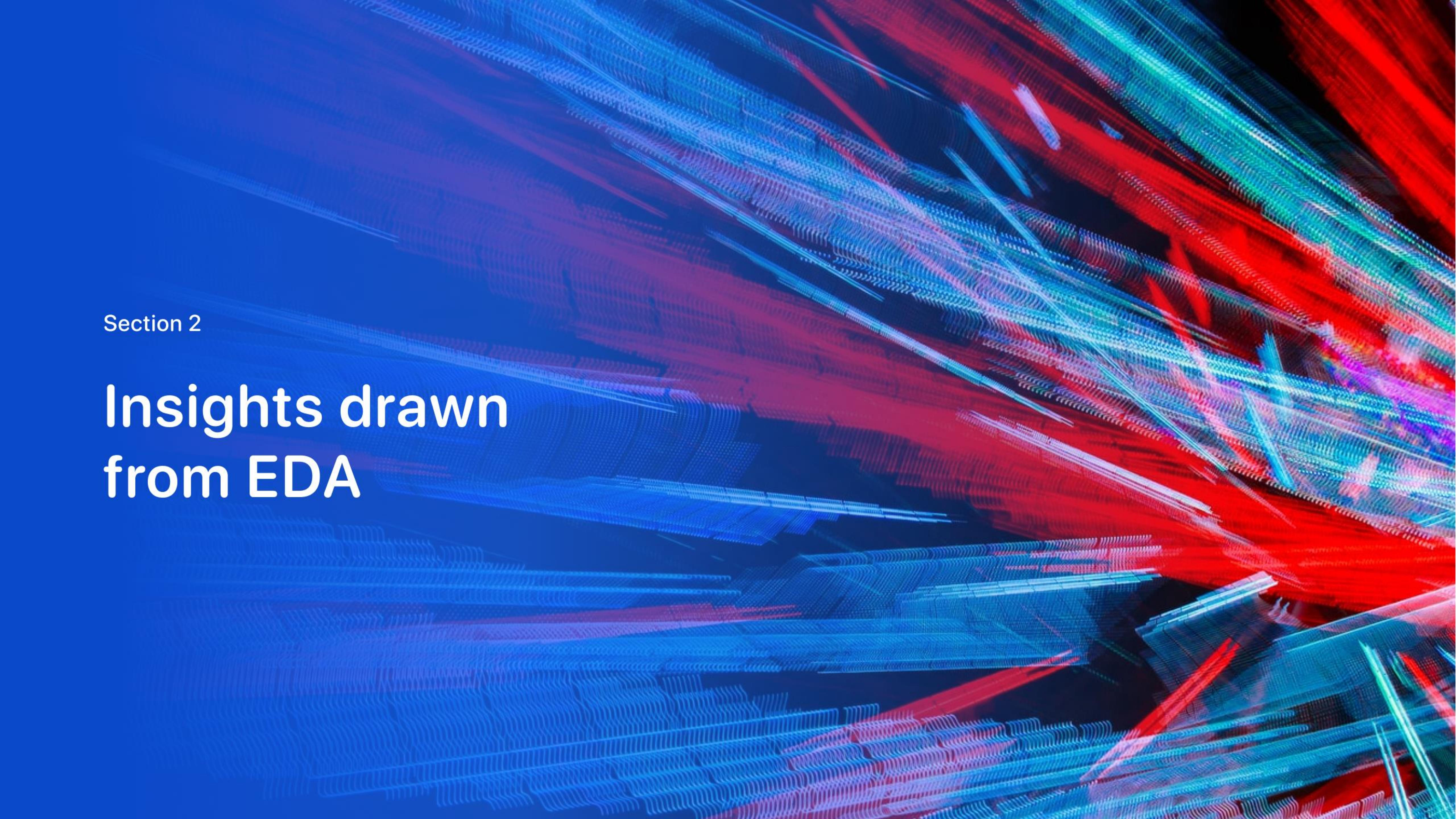
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive Analysis (Classification)

- In this part, I have used Sklearn library of the Python.
- Initally the features were normalized and then split into test and train sets.
- Four machine learning models were trained using training data set. These four machine learning models are Logistic Regression, SVM, Tree Classification and KNN (K Nearest Neighbours).
- The model parameters were turned using Gridsearh function from sklearn library and best tuning parameters are selected based on training data set results.
- The best tuned paramenters of each model was used to test the trest_data and then accuracy of prediction was found as well as confusion matrix was calculated.
- The accuracy of prediction for all of these models is found to be 83.33% on test data set. The training data accuacy was different and best for 87.5% for Tree Classifier. Hence tree classification is performig best in this case.
- The note book with predictive classification analysis is present at github and the address is as
- https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

- Based on Exploratory data analysis we found that success rates are improving since 2013. The launch site VAFB SLC 4E has highest success rates of 77%. Considering the orbits ES L1, SSO, GEO and LEO orbits have 100% success rates.
- It was also found that higher payload increase the success rates.
- It was found that Launch sites are near the equator and sea shore.
- Based on Machine Learning algorithms it is possible to predict with 83.33% accuracy of success for launch based on the features using Tree Classier as well as KNN, SVM and Logistic Regression.
- These machine learning tools and data analysis can be used by Space X competitors to target their lower cost and it can be used by Space X to reduce its cost further and increase its success rate.



Section 2

Insights drawn from EDA

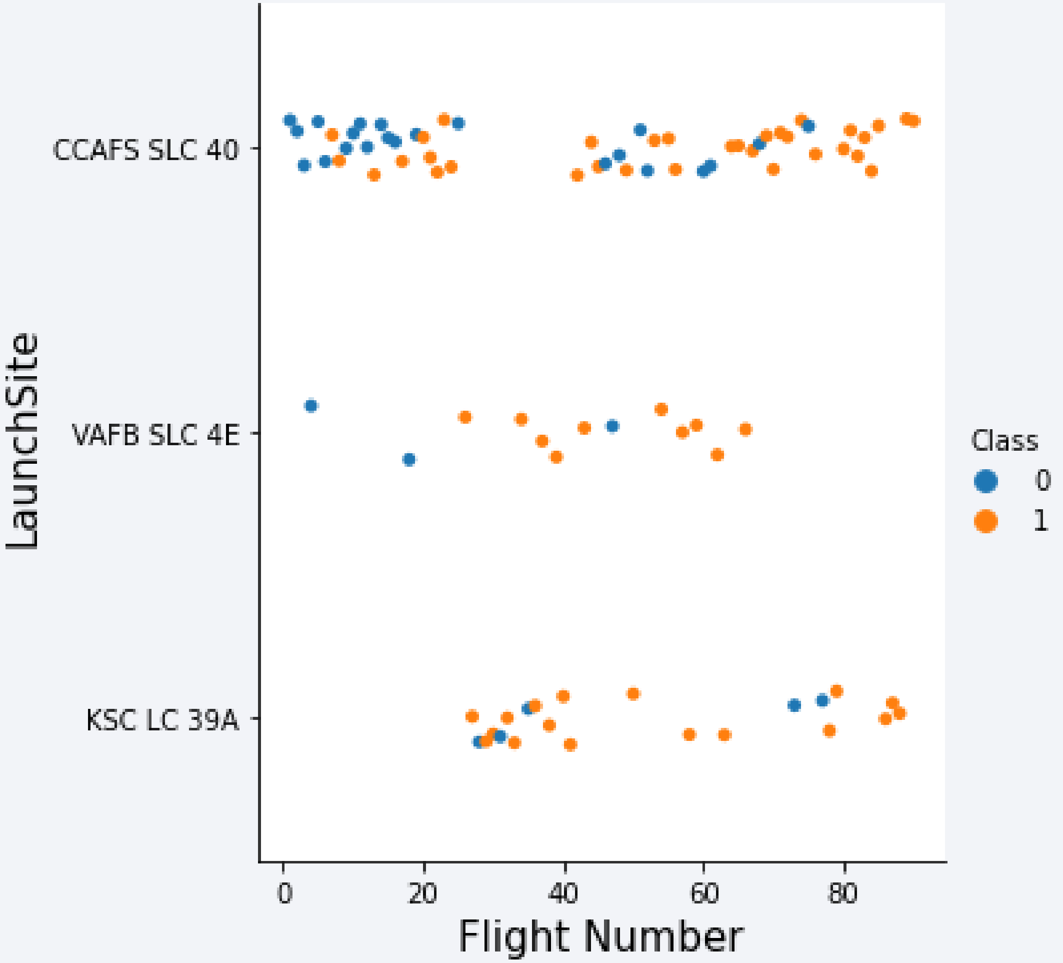
Flight Number vs. Launch Site

On the right is the Number vs Launch site Scatter plot.

It shows more flights are successful as we can see more orange points on the right upper corner.

The address of github repo having note book with these plots is here

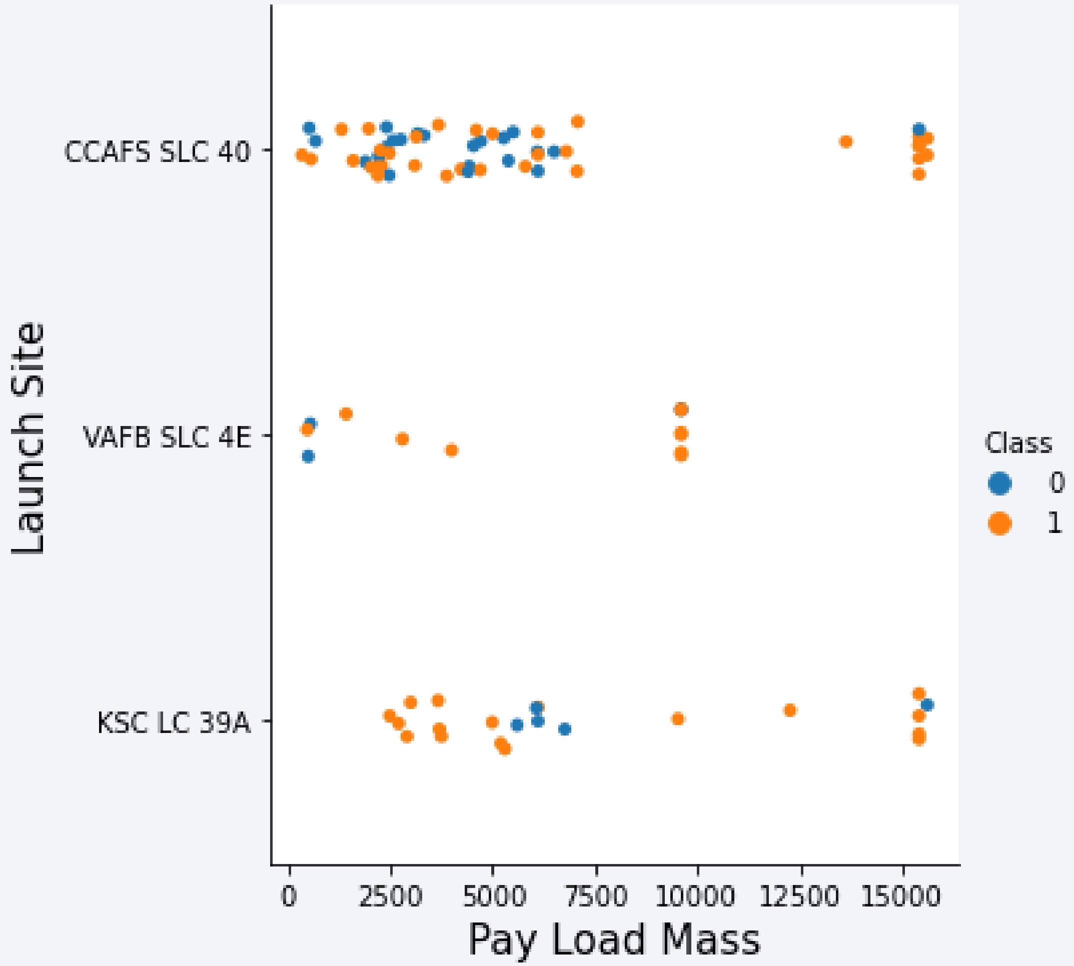
https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/jupyter-labs-eda-dataviz.ipynb



Payload vs. Launch Site

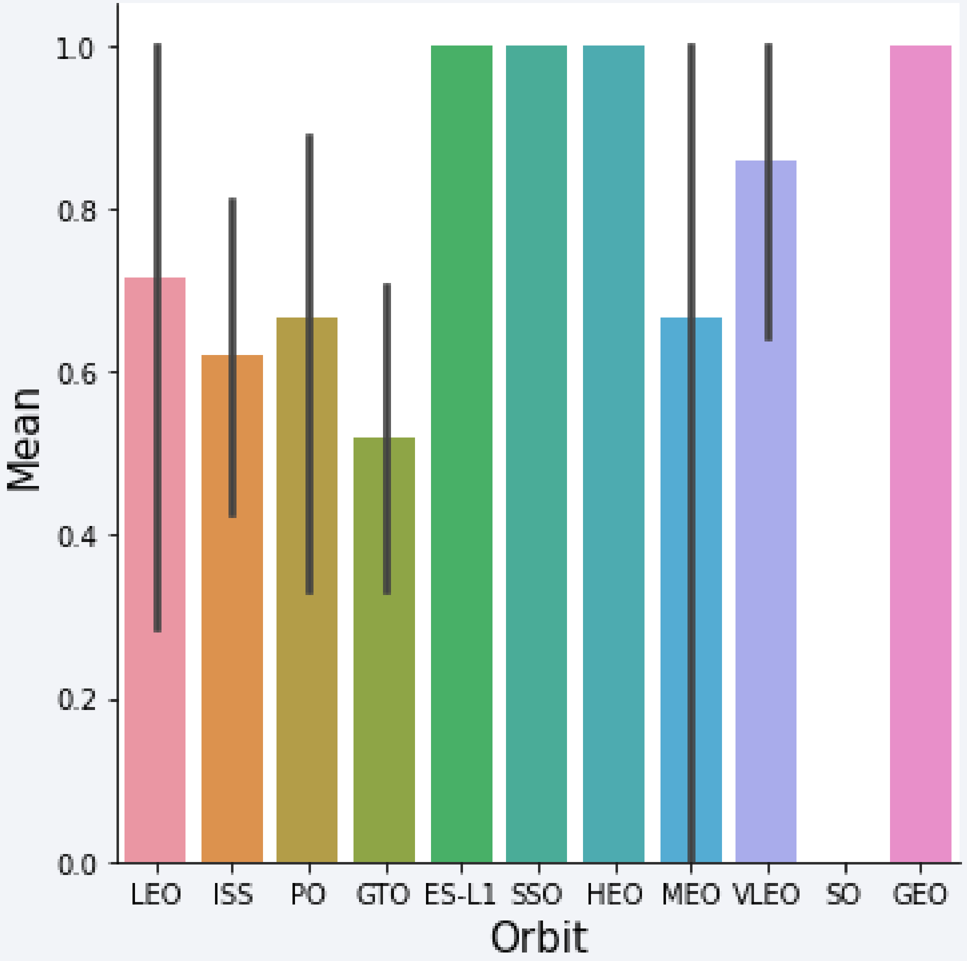
- Shown on right a scatter plot of Payload vs. Launch Site

- CCAFS SLC 40 has highest frequency of launches and hence highest frequency of high payload mass.
- VAFB has not launched payload mass above 10,000 Kg



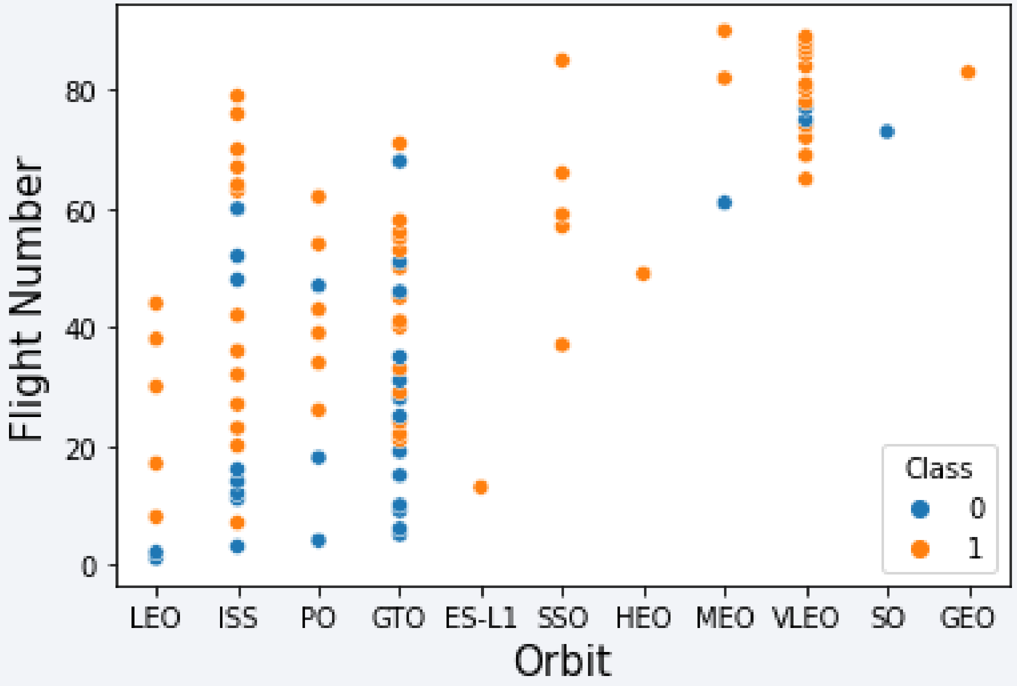
Success Rate vs. Orbit Type

- Shown on right is a bar chart for the success rate of each orbit type
- It shows that for ES L1, SSO, HEO and GEO has 100% success rate followed by VLEO.
- The lowest success rate is for GTO



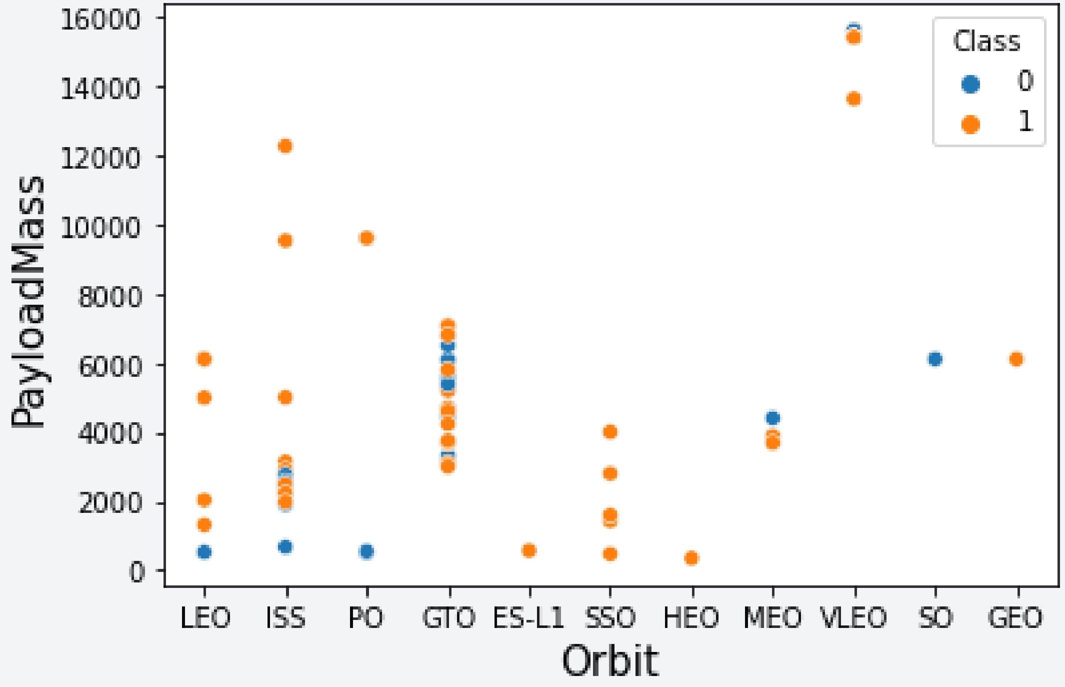
Flight Number vs. Orbit Type

- Shown or right a scatter point of Flight number vs. Orbit type
- GTO has highest proportion of unsuccessful first part landing as too many blue point.
- GEO, SSO, ES L1 and HEO has 100% success rate and high frequency of flights
- LEO has low frequency but quite higher success rate.
- Very few flight to ES L1 orbit,



Payload vs. Orbit Type

- Shown on right a scatter point of payload vs. orbit type

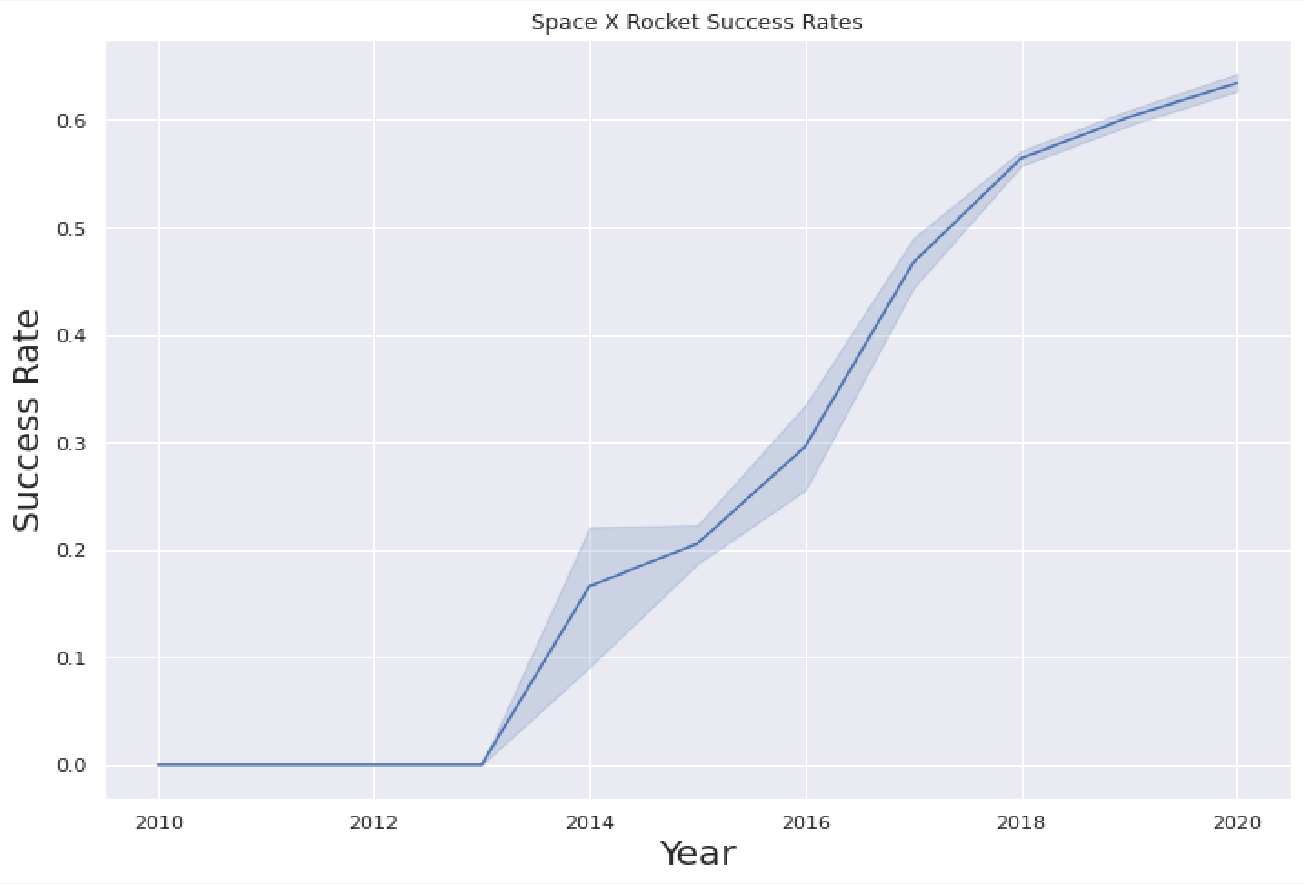


- Higher payloads for ISS, PO, VLEO orbits
- Low payloads are carried to LEO, PO, ES L1, SSO, HEO and MEO orbits

Launch Success Yearly Trend

- Shown on right is a line chart of yearly average success rate

- Success rate has considerably increased since 2013 and is having increasing trend continue.



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here
- Following is the query, ABC is the name of the table having data.
- %sql select DISTINCT Launch_Site from ABC
- The four sites are CCAFS LC-40, CCAFS SLC-KSC LC-39A40, VAFB SLC-4E
- ‘%sql’ is for sql magic, ‘select’ is DML statement, Launch Site is column name in the table ABC and Distinct is used to get unique values.

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here
- Here is the query, ABC is the name of the table having data.
- %sql select * from ABC where Launch_Site like ‘CCA%' Limit 5
- ‘%sql’ is for sql magic, ‘select’ is DML statement, ‘CCA%’ has used wild card character, and ‘Limit 5’ will give only 5 results

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Present your query result with a short explanation here
- Here is the query, ABC is the name of the table having data.
- %sql select SUM(PAYLOAD_MASS__KG_) from ABC where Customer Like 'NAS%‘
- The value found is 36679
- ‘%sql’ is for sql magic, ‘select’ is DML statement, SUM() adds the values in Column Payload Mass in the table ABC.

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here
- The query is here, ABC is the name of the table having data.
- %sql select AVG(PAYLOAD_MASS__KG_) from ABC where BOOSTER_VERSION = 'F9 v1.1‘
- The result is 3676
- ‘%sql’ is for sql magic, ‘select’ is DML statement, AVG() gives mean the values in Column Payload Mass in the table ABC when conditon BoosterVersion is F9 v1.1.

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here
- Here is the query, ABC is the table present in DB2 that has data
- %sql select MIN(DATE) from ABC where Landing__Outcome = 'Success (ground pad)‘
- The result of query is 2017-01-05
- ‘%sql’ is for sql magic, ‘select’ is DML statement, MIN(Date) gives first date value in Column Date in the table ABC when where Landing__Outcome = 'Success (ground pad)‘ is met.

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- Present your query result with a short explanation here
- Here is the query, ABC is the table present in DB2 that has data
- %sql select BOOSTER_VERSION from ABC where Landing__Outcome = 'Success (drone ship)'AND (payload_mass__kg_>4000 AND Payload_mass__Kg_<6000)
- The outcome is F9 FT B1022, F9 FT B1031.2
- ‘%sql’ is for sql magic, ‘select’ is DML statement, MIN(Date) gives Booster Versions for the condition

Total Number of Successful and Failure Mission Outcomes

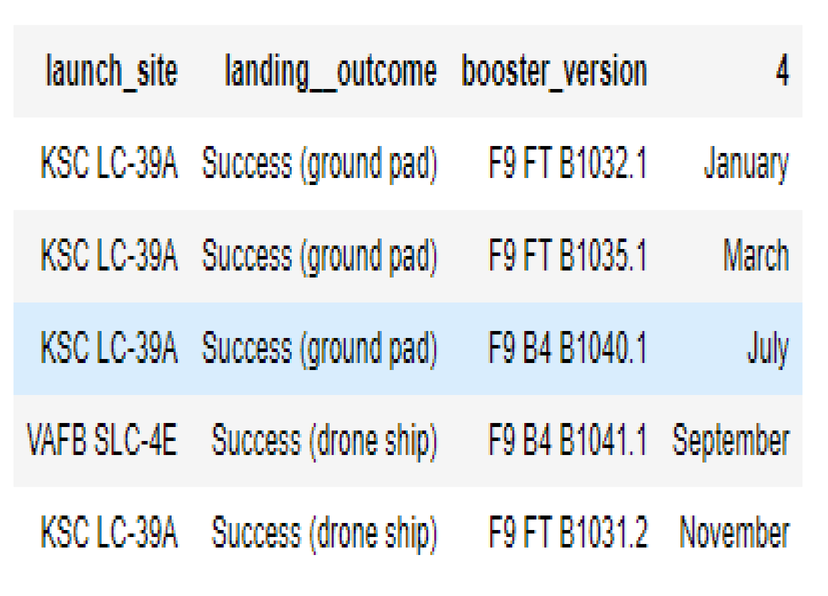
- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here
- Here is the query, ABC is the name of the table in DB2
- %sql Select Count(*) Landing__outcome from ABC
- ‘%sql’ is for sql magic, ‘select’ is DML statement, Count(*) gives total for colum Mission_Outcomes

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here
- Here is the sql query, ABC is the table name stored in DB2 having data
- %sql select Booster_version from ABC where Payload_mass__Kg_ = (select MAX(payload_mass__kg_) from ABC)
- This query has used subquery for the condtion of maximum pay load, using Select DML statement and MAX() function for column Payload__Mass_Kg

2015 Launch Records

- List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017
- Present your query result with a short explanation here
- Here is the query, ABC is the name of table with data
- %sql select Launch_site, Landing__Outcome, Booster_version, MONTHNAME(date) from ABC where (Landing__Outcome LIKE '%Success%') AND YEAR(Date) = '2017'
- The output is shown on the right.
- '%sql' is for sql magic, 'select' is DML statement for selecting required columns, string pattern is used with like statement and year () and Monthname() function are used to extract year and month name.



launch_site	landing__outcome	booster_version	4
KSC LC-39A	Success (ground pad)	F9 FT B1032.1	January
KSC LC-39A	Success (ground pad)	F9 FT B1035.1	March
KSC LC-39A	Success (ground pad)	F9 B4 B1040.1	July
VAFB SLC-4E	Success (drone ship)	F9 B4 B1041.1	September
KSC LC-39A	Success (drone ship)	F9 FT B1031.2	November

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

- Rank the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order
- Present your query result with a short explanation here
- Here is the query, ABC is the name of table containing data
- %sql SELECT COUNT(Landing__Outcome) FROM ABC WHERE (Landing__Outcome LIKE '%Success%') AND (Date >'2010-06-04') AND (Date < '2017-03-20')
- ‘%sql’ is for sql magic, ‘select’ is DML statement for selecting required columns values as the per the condtion and count() function will give the count of the required landing outcomes as per given dates.

Section 4

Launch Sites Proximities Analysis

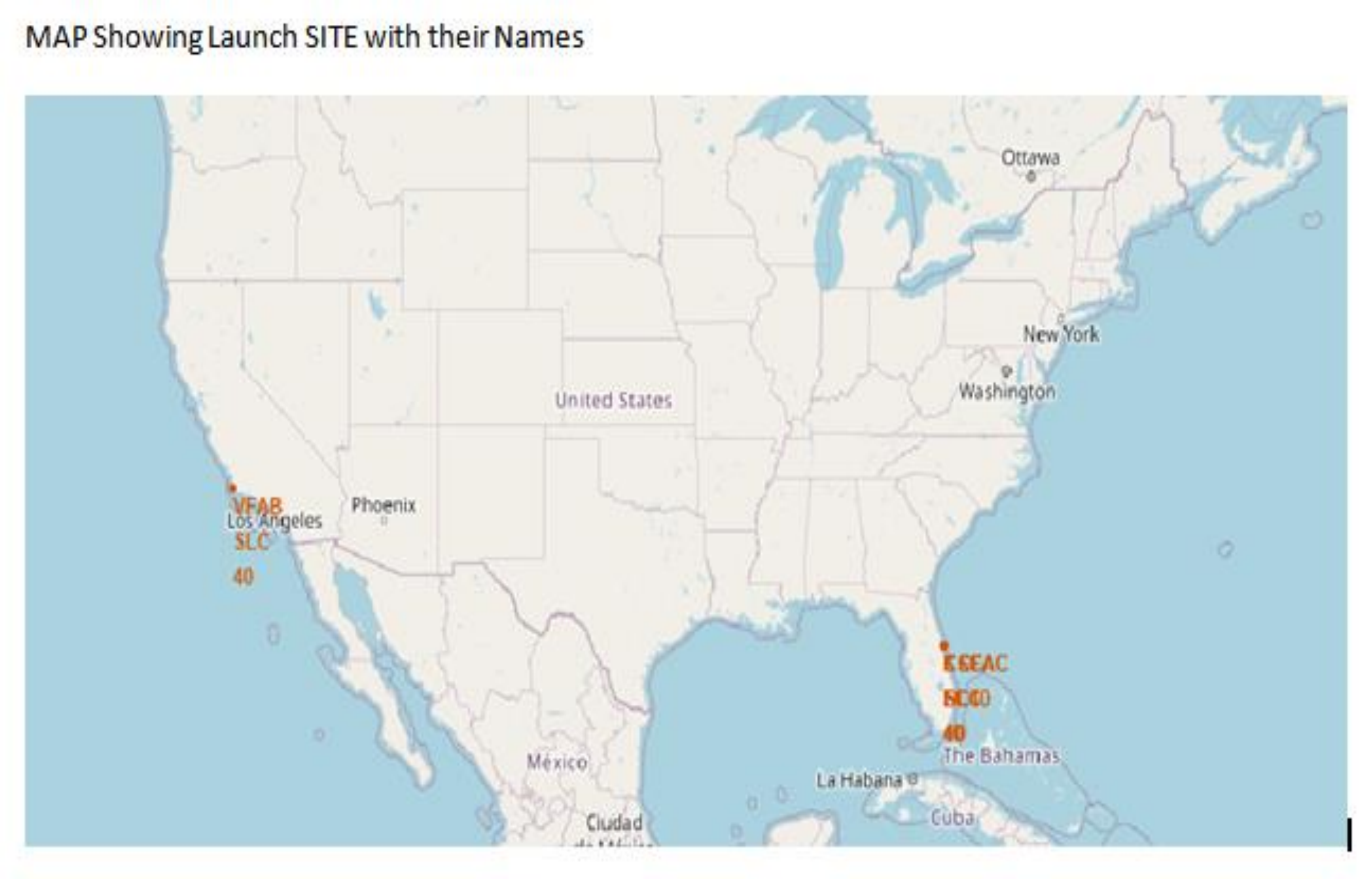
<Folium Map Screenshot 1>

The map shows the location of the launch SITE and their names are added as Marker on their location.

This help us visualise the place from where SpaceX is launching its Rockets.

The link of git hub repo with notebook is having code detailed maps is as

https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/lab_jupyter_launch_site_location.ipynb

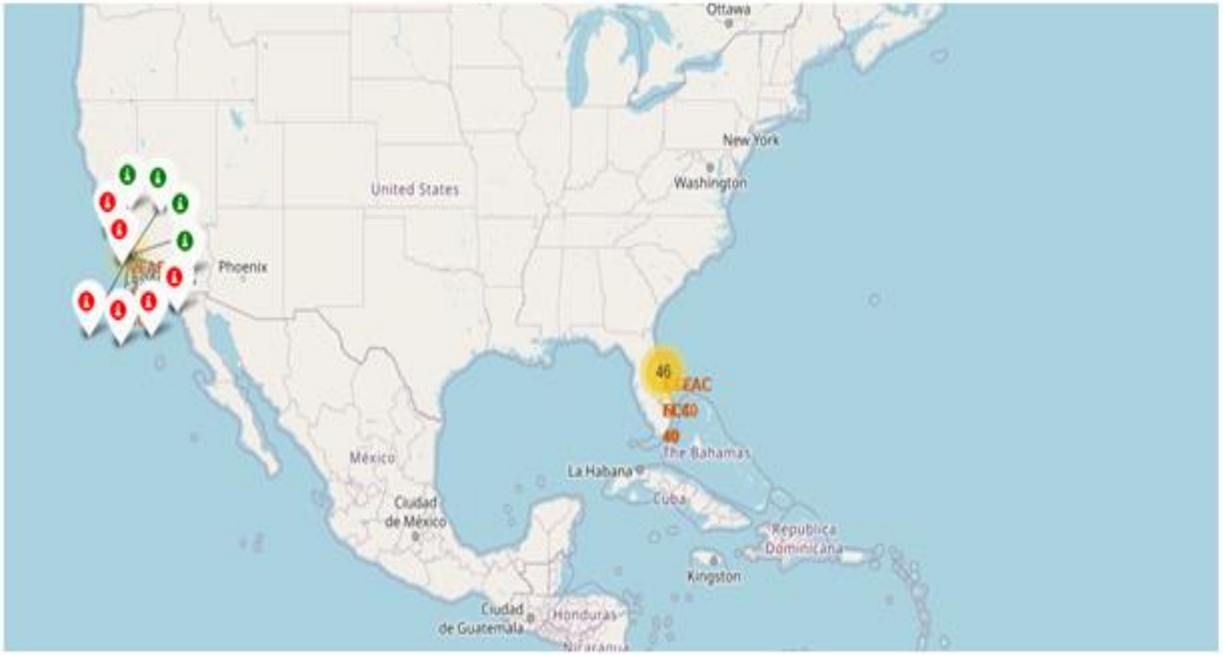


MAP Showing Launch SITE with their Names

<Folium Map Screenshot 2>

- This map gives details of launch sites, how many launches from each site as well as the success and failure launch informtion

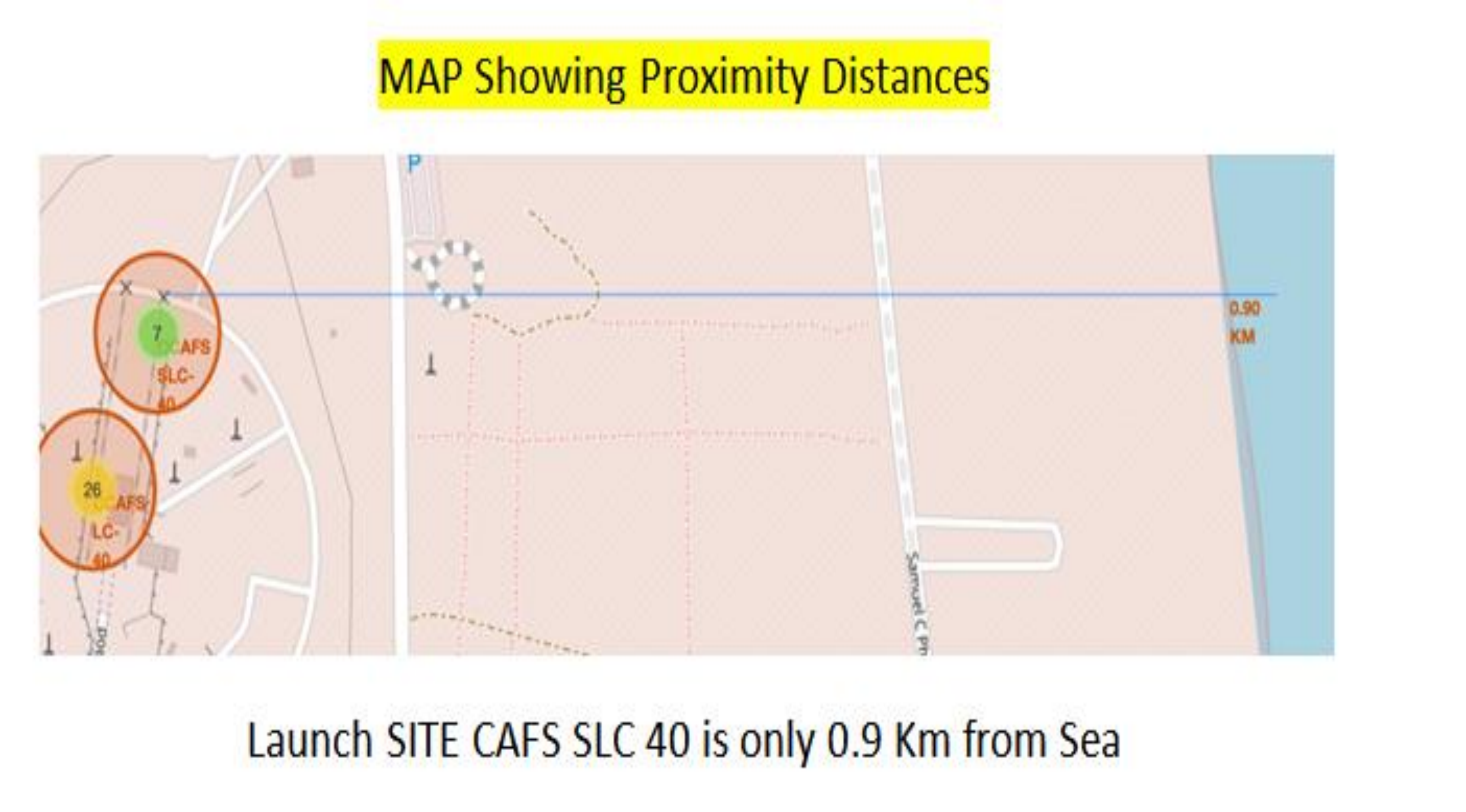
MAP Showing Launch sites and Number of Launches



Green colour indicate success and red failure

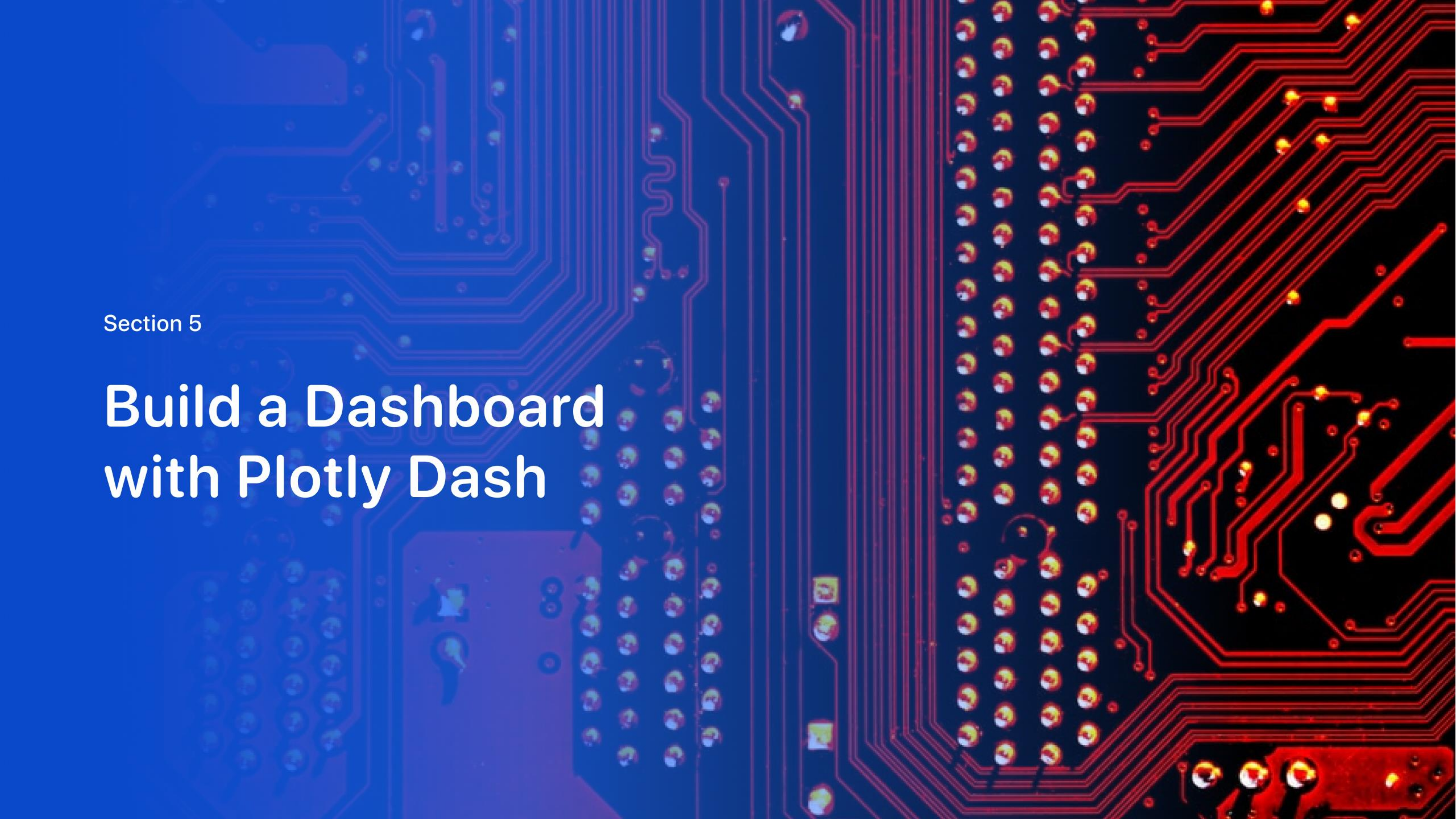
<Folium Map Screenshot 3>

- Launch Sites are usually close to sea shore line.
- In this details of proximity objects. In this case, sea shore which is only 0.9 km from the CAFS SLC 40



MAP Showing Proximity Distances

Launch SITE CAFS SLC 40 is only 0.9 Km from Sea



Section 5

Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>

- I have skipped this part.
- During the course of data visualization, I was able to do plotly but Dash was not working on my computer, even when using all the code given for Their environment. I tried on IBM skill networks labs also but there was error. Later, using various online videos from YouTube and reading some info on internet, I tried on Juypter note book of Anaconda, but it was still not working. I left it for the time being.
- While doing this project, I skipped Dash part.
- Later near the end, I tried again Dash using the instructions given in module 3 of capstone and to my surprise, it was working fine. Bust as I have a deadline my course is finishing on 31.10.2021 so I am skipping this for the moment and will do it later for my CPD. I regret this thing.

<Dashboard Screenshot 2>

- I have skipped this part.
- During the course of data visualization, I was able to do plotly but Dash was not working on my computer, even when using all the code given for Their environment. I tried on IBM skill networks labs also but there was error. Later, using various online videos from YouTube and reading some info on internet, I tried on Juypter note book of Anaconda, but it was still not working. I left it for the time being.
- While doing this project, I skipped Dash part.
- Later near the end, I tried again Dash using the instructions given in module 3 of capstone and to my surprise, it was working fine. Bust as I have a deadline my course is finishing on 31.10.2021 so I am skipping this for the moment and will do it later for my CPD. I regret this thing.

<Dashboard Screenshot 3>

- I have skipped this part.
- During the course of data visualization, I was able to do plotly but Dash was not working on my computer, even when using all the code given for Their environment. I tried on IBM skill networks labs also but there was error. Later, using various online videos from YouTube and reading some info on internet, I tried on Juypter note book of Anaconda, but it was still not working. I left it for the time being.
- While doing this project, I skipped Dash part.
- Later near the end, I tried again Dash using the instructions given in module 3 of capstone and to my surprise, it was working fine. Bust as I have a deadline my course is finishing on 31.10.2021 so I am skipping this for the moment and will do it later for my CPD. I regret this thing.

Section 6

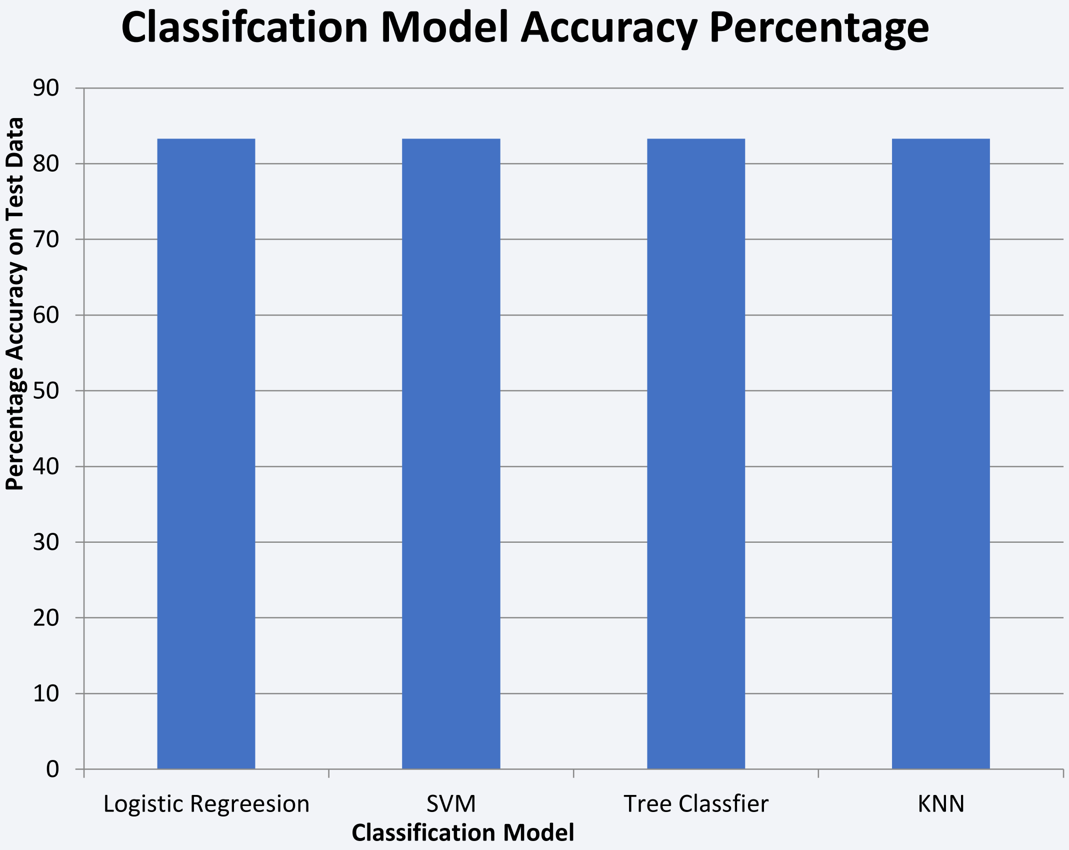
Predictive Analysis (Classification)

Classification Accuracy

All four classification model has shown 83.3% accuracy on the Test Data Set.

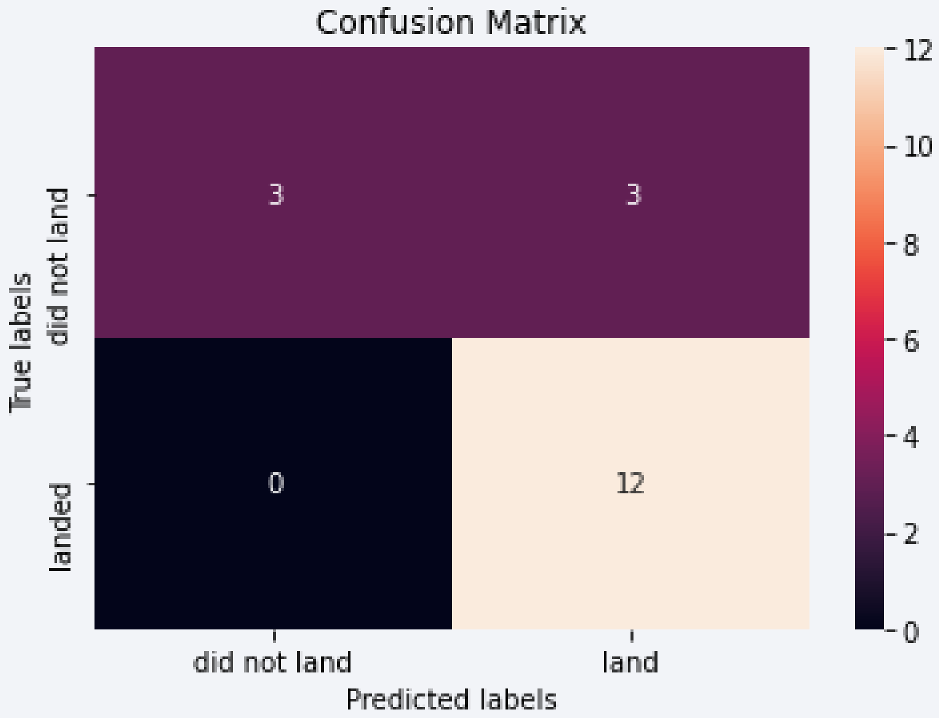
The notebook having these machine learning model development can be found here

https://github.com/dswaseem/IBM-Capstone-notebooks/blob/master/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



Confusion Matrix

- On Test dataset all models have 83.3% accuracy and their confusion matrix is also same. Based on both training accuracy and test data accuracy, tree classifier is the best performing model. It has training accuracy of 87.5% and test data accuracy of 83.3%.Its confusion matrix is attached. Cofusion Matrix gives information on True Positive (12), True Negative (3), False Positive (3) and False negative (0). These are shown also on the Confusion Matrix.



Conclusions

Space X is able to offer satellite launch services and deliver cargo to ISS at considerable low cost as it can use its rocket and parts in first part of the launch.

Satellite Launch stations are located near sea and equator, they have different rate of success.

A rocket launch with payload more than 10,000 Kg has high chances or being a successful launch and its rocket and launch parts of first stage to be safely landed and being reused. Since 2013, rate of success launch is continuously increasing.

It is possible to analyse the launch data with exploratory data analysis and see how launch success is related to rocket type, payload mass, landing type, orbit type and so on, to get insights of the data. On doing this, we find that ES L1, SSO, HEO and GEO orbits have almost 100% success rates and we can predict that future flights to thee orbits will be successful with high confidence.

Machine Learning models help us predict / classify a launch success of failure based on the features, Various machine learning classification models are available to use. When data analysed using Logistic Regression, Tree Classifier, SVM and KNN, a test data accuracy of 83.3% and up to 87.5% accuracy on training data was achieved and Tree classifier was found to be most appropriate because of its high accuracy on both training as well as test data.

Appendix

- The codes are on the notebook and are available on GitHub at following link.
- https://github.com/dswaseem/IBM-Capstone-notebooks

Thank you!