IBM Data Science Capstone project

Ryan Dbritto

10th August 2021

Outline

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

Executive Summary

Summary of methodologies

- Data Collection
- Data Wrangling
- EDA with Data Visualization
- EDA with SQL
- Building an interactive map with Folium
- Building a dashboard with Plotly Dash
- Predictive Analysis

Summary of all results

- Exploratory Data Analysis results
- Interactive analytics demo in screenshots
- Predictive Analysis Results

Introduction

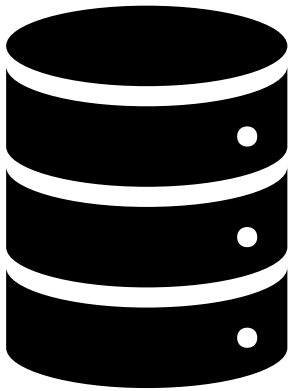
The final course of the Data Science Professional Certificate consist of a capstone project where in all the skills and relevant knowledge that one has gathered from this 9 intense courses has to be applied on a final capstone project.

The commercial space age is here, companies are making space travel affordable for everyone. Virgin Galactic is providing suborbital spaceflights. Rocket Lab is a small satellite provider. Blue Origin manufactures sub-orbital and orbital reusable rockets. Perhaps the most successful is SpaceX. SpaceX's accomplishments include: Sending spacecraft to the International Space Station.Starlink, a satellite internet constellation providing satellite Internet access. Sending manned missions to Space. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. Spaces X's Falcon 9 launch like regular rockets. Our goal is to use the data to predict whether SpaceX will attempt to land a rocket or not.

Methodology

Data collection

For this project I worked with the SpaceX Launch Data that was gathered from an API, specifically the SpaceX REST API. Another method I used for getting the Falcon 9 launch data is web scrapping related wiki pages.



Data collection – SpaceX API

The SpaceX API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.

- Use Space X REST API
- Filter dataframe for Falcon 9 data only / clean data
- Normalize data into flat data file such as .csv

Github URL of SpaceX notebook

1. Getting Response from API

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

2. Convert response to a .json file

```
data = pd.json_normalize(response.json())
```

3. Apply custom functions to clean the data

```
getPayloadData(data)
```

```
getBoosterVersion(data)
```

```
getLaunchSite(data)
```

```
getCoreData(data)
```

4. Construct the dataset using the data obtained

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

```
df_falcon = pd.DataFrame(launch_dict)
```

5. Filter the dataframe

```
data_falcon9 = df_falcon['BoosterVersion']!='Falcon 1'
```

Data collection – Web Scrapping

Another popular data source for obtaining Falcon 9 Launch data is web scraping related Wiki pages. I used Python BeautifulSoup package to web scrape some HTML tables that contain valuable Falcon 9 launch records.

- Getting HTML response from wiki page
- Extract data using BeautifulSoup
- Normalize data into flat data file such as .csv

Github URL of SpaceX notebook

1. Getting Response from HTML

```
response = requests.get(static_url)
```

2. Create BeautifulSoup Object

```
soup = BeautifulSoup(response.content, 'html5lib')
```

3. Finding tables

```
html_tables = soup.find_all('table')
```

4. Getting Column names

```
column_names = []


th=first_launch_table.find_all('th')
for i in range(len(th)):
    name=extract_column_from_header(th[i])
    if name is not None and len(name) > 0:
        column_names.append(name)
```

5. Creating a Dictionary

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

6. Appending data to keys

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
```

7. Convert dictionary to dataframe

```
df=pd.DataFrame(launch_dict)
```

Data Wrangling

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship. We mainly convert those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.

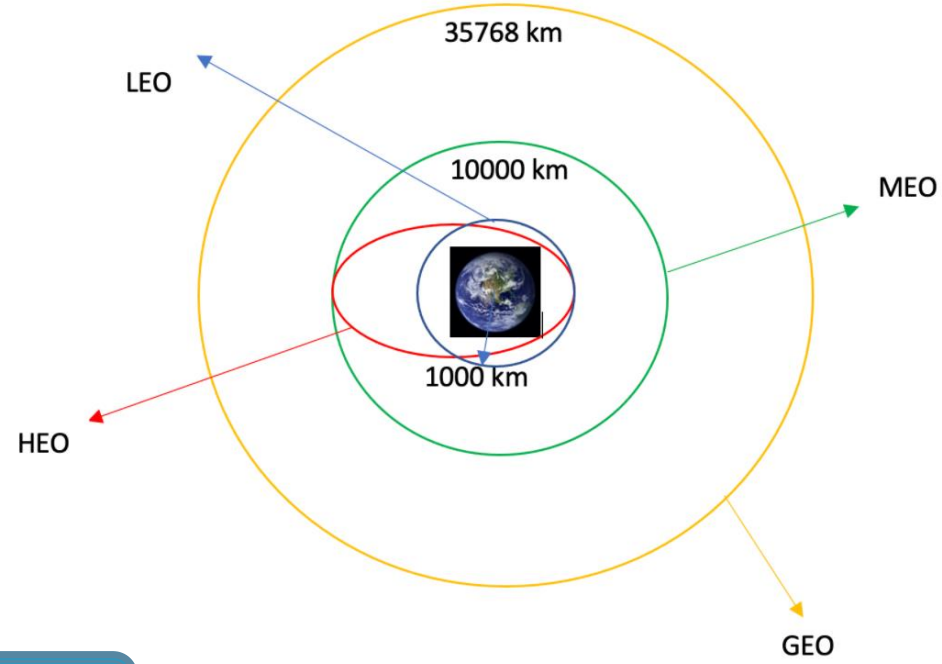


Diagram showing common orbit type SpaceX uses

Perform Exploratory Data Analysis (EDA) on dataset

Calculate the number of launches at each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per

Export dataset as .CSV

Create a landing outcome label from Outcome column

Work out success rate for every landing in dataset

Github URL of SpaceX notebook

EDA with data visualization

Plotted Scatter Graphs :

Flight Number VS Payload Mass

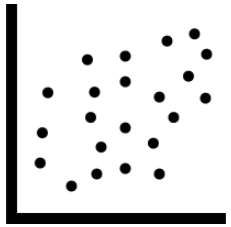
Flight Number VS Launch Site

Payload Mass VS Launch Site

Orbit VS Flight Number

Payload Mass VS Orbit Type

Orbit VS Payload Mass

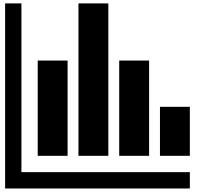


Scatter plots show how much one variable is affected by another. The relationship between two variables is called their correlation . Scatter plots usually consist of a large body of data.

Github URL of SpaceX notebook

Plotted Bar Graph :

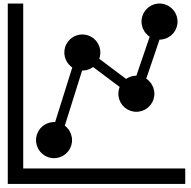
Mean VS Orbit



A bar diagram makes it easy to compare sets of data between different groups at a glance. The graph represents categories on one axis and a discrete value in the other. The goal is to show the relationship between the two axes. Bar charts can also show big changes in data over time.

Plotted Line Graph :

Success rate VS Year



Line graphs are useful in that they show data variables and trends very clearly and can help to make predictions about the results of data not yet recorded

EDA with SQL

- Display the names of unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the 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. Use a subquery
- List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch site for the months in year 2015
- Rank the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order.

Github URL of SpaceX notebook

Build an interactive map with Folium

- To visualize the Launch Data into an interactive map. I took the Latitude and Longitude Coordinates at each launch site and added a Circle Marker around each launch site with a label of the name of the launch site.
- We assigned the dataframe launch_outcomes(failures, successes) to classes 0 and 1 with Green and Red markers on the map in a MarkerCluster()
- Using Haversine's formula we calculated the distance from the Launch Site to various landmarks to find various trends about what is around the Launch Site to measure patterns. Lines are drawn on the map to measure distance to landmarks

Example of some trends in which the Launch Site is situated in.

•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

Github URL of SpaceX notebook

Build a Dashboard with Plotly Dash

The dashboard is built with Flask and Dash web framework.

Pie Chart showing the total launches by a certain site/all sites

- display relative proportions of multiple classes of data.
- size of the circle can be made proportional to the total quantity it represents.

Scatter Graph showing the relationship with Outcome and Payload Mass (Kg) for the different Booster

- Versions
- - It shows the relationship between two variables.
- - It is the best method to show you a non-linear pattern.
- - The range of data flow, i.e. maximum and minimum value, can be determined.
- - Observation and reading are straightforward

Predictive analysis (Classification)

BUILDING MODEL

- Load our dataset into NumPy and Pandas
- Transform Data
- Split our data into training and test data sets
- Check how many test samples we have
- Decide which type of machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit our datasets into the GridSearchCV objects and train our dataset.

EVALUATING MODEL

- Check accuracy for each model
- Get tuned hyperparameters for each type of algorithms
- Plot Confusion Matrix

IMPROVING MODEL

- Feature Engineering
- Algorithm Tuning

FINDING THE BEST PERFORMING CLASSIFICATION MODEL

- The model with the best accuracy score wins the best performing model
- In the notebook there is a dictionary of algorithms with scores at the bottom of the notebook.

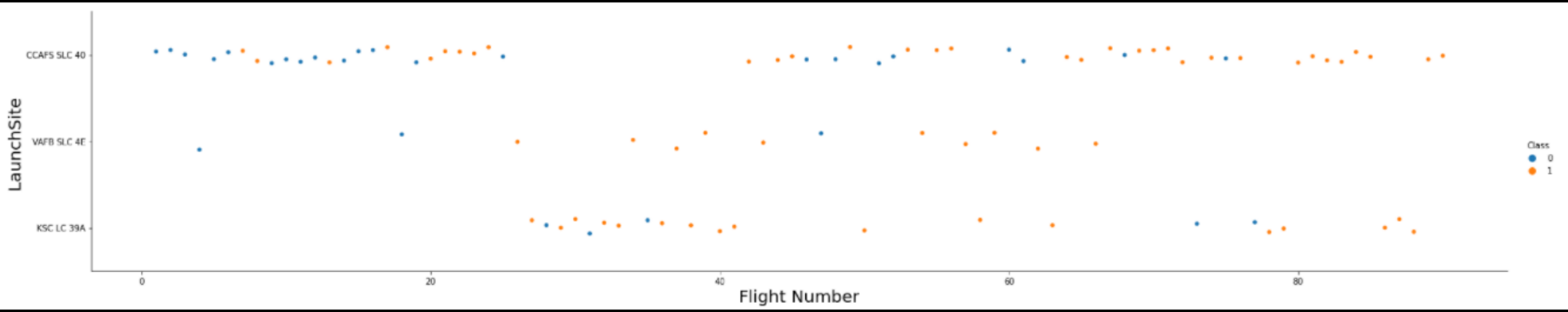
Github URL of SpaceX notebook

Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

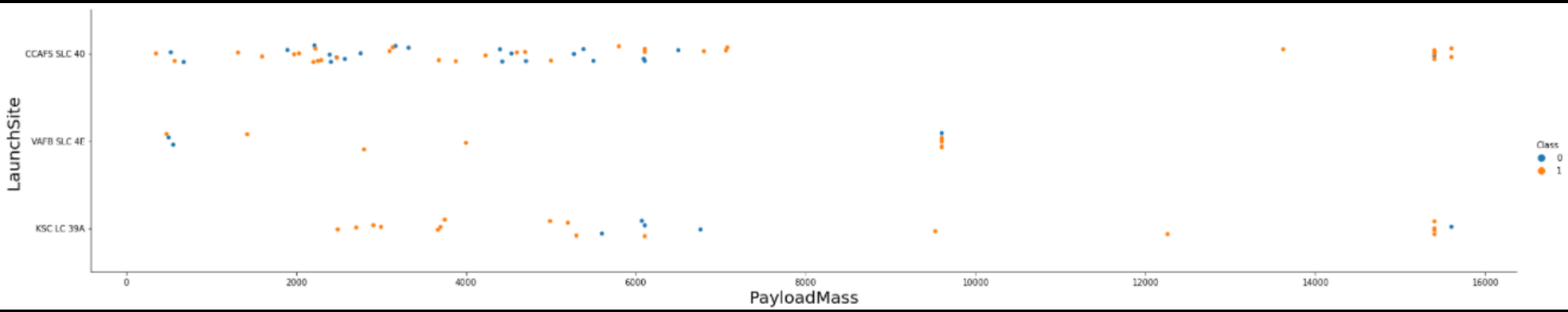
EDA with Visualization

FLIGHT NUMBER vs LAUNCH SITE



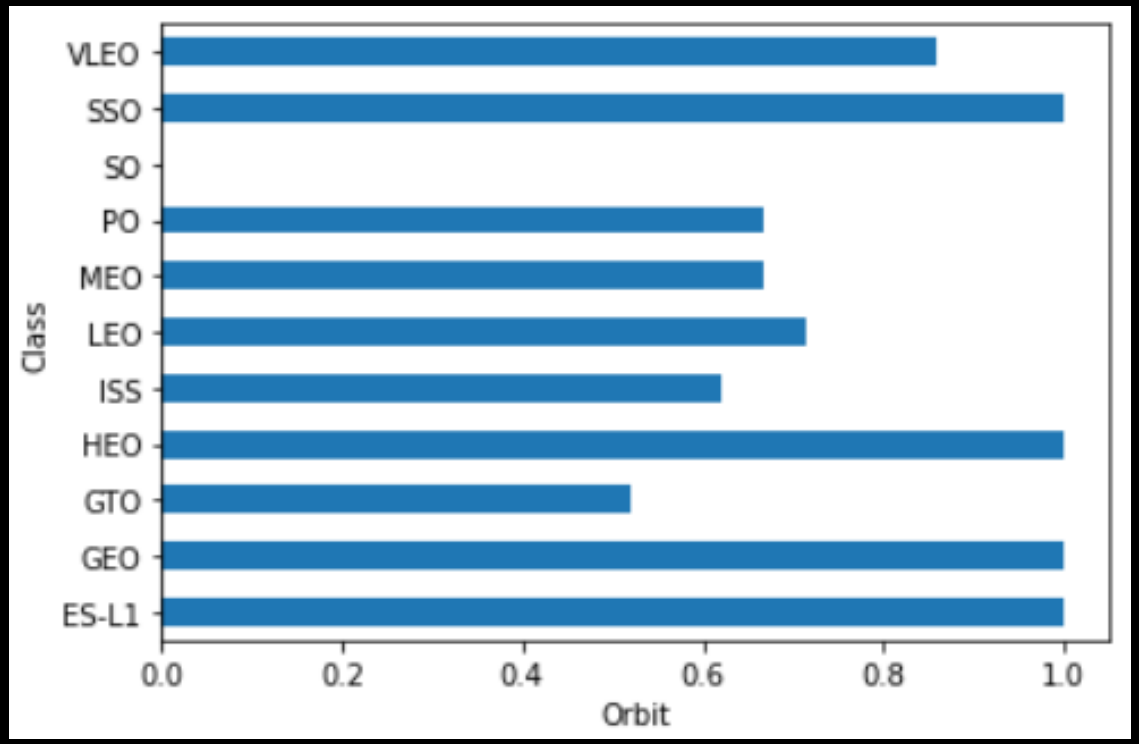
- We see that as the flight number increases, the first stage is more likely to land successfully.
- KSC LC – 39A & VAFB SLC - 4E have higher success rates compared to CCAFS LC – 40.

PAYLOAD MASS vs LAUNCH SITE



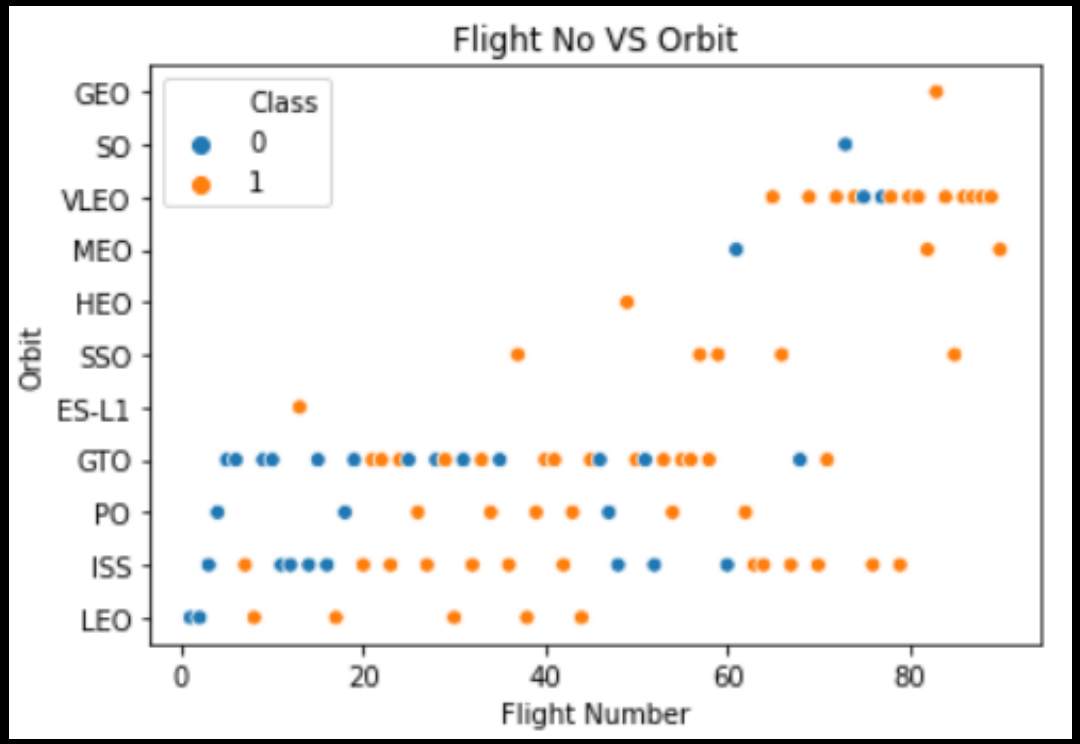
- For launch site CCAFS LC – 40, in the range of 12000 – 16000 load mass the success rate is much higher than that in the region of 0 - 8000.
- For KSC LC – 39A , the success rate is much higher in the region of less than 6000.
- For VAFB SLC - 4E, the success rate is much higher when the load is between 2000 - 6000.

SUCCESS RATE vs ORBIT



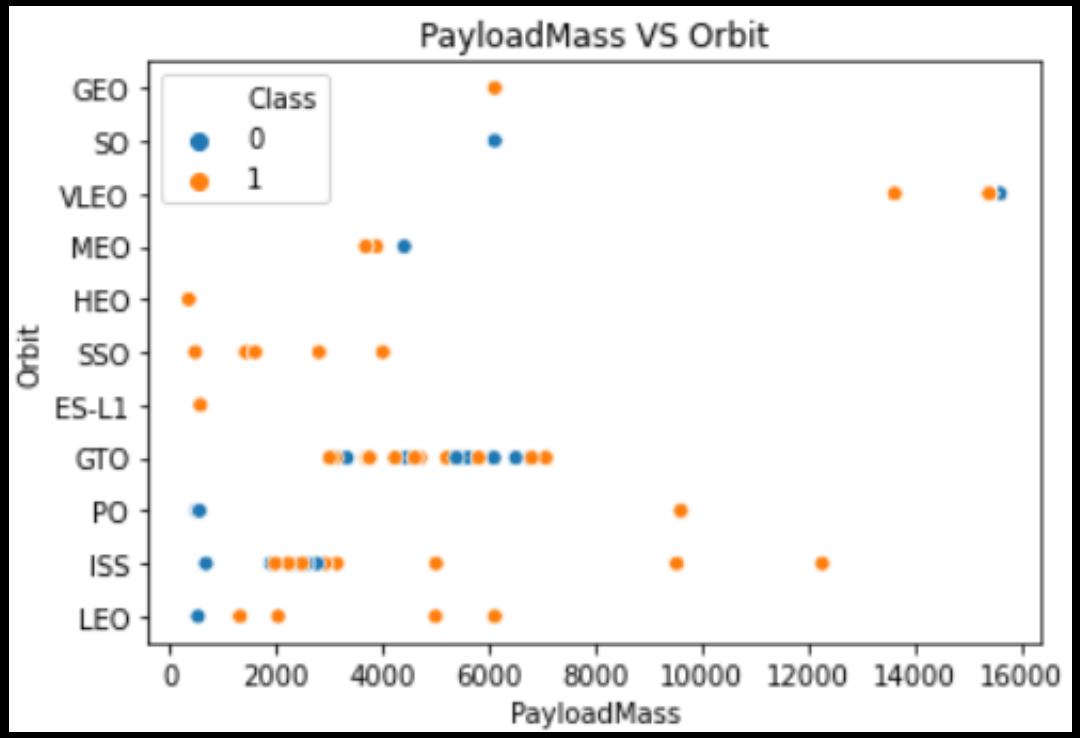
- There are higher chances for reaching out the orbit SSO, HEO, GEO, ES-L1.

FLIGHT NUMBER vs ORBIT



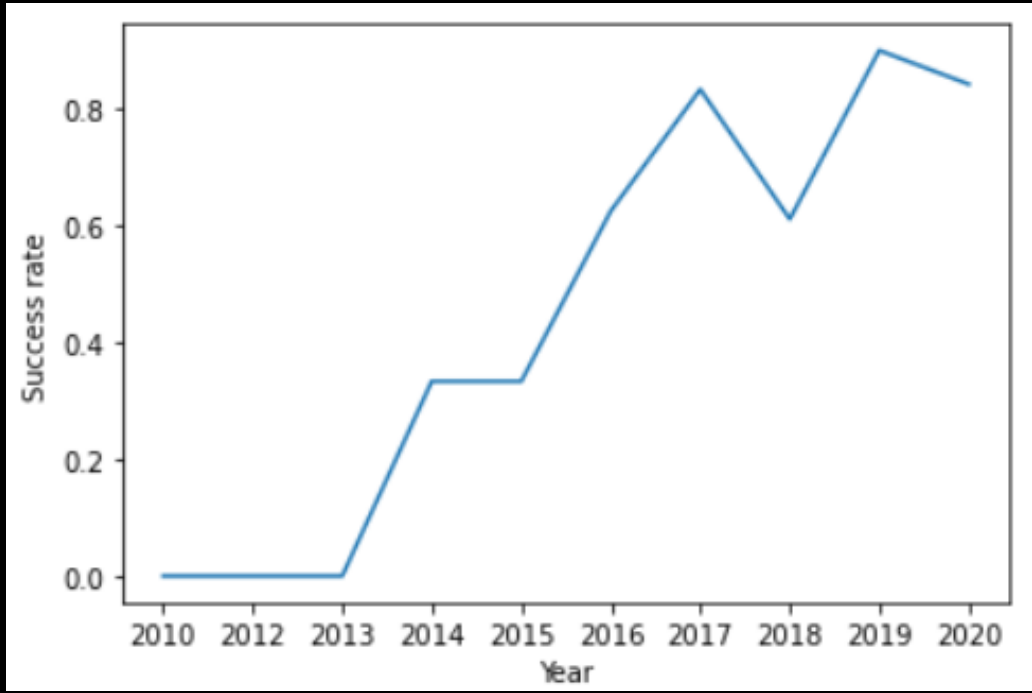
- In the LEO Orbit there is a strong relation between success and number of flights.
- On the other hand there is null relationship of success and number of flights in the GTO Orbit

PAYLOAD MASS vs ORBIT



- Heavy payload mass have a negative impact in GTO Orbits, whereas the impact is positive in LEO, ISS and Polar Orbits

SUCESSFUL LAUNCHES - YEARLY



- The success rate for the Falcon 9 launches has been inclining since 2013, although we see a decline in 2018.

EDA with SQL

All Launch Site names

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

In [23]:
```
%sql select distinct launch_site from SPACEXDATASET
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[23]:

launch_site
CCAFS LC-40
CCAFS SLC-40
CCAFSSLC-40
KSC LC-39A
VAFB SLC-4E

Launch Site names that begin with CCA

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

In [9]:
```
%sql select * from SPACEXDATASET WHERE launch_site like 'CCA%' LIMIT 5
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[9]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
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	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
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

In [25]:
```
%sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXDATASET where customer='NASA (CRS)'
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[25]:

total_payload_mass
45596

Total Payload Mass carried by booster version F9 v1.1

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

In [26]:
```
%sql select avg(payload_mass__kg_) as avg_payload_mass from SPACEXDATASET where booster_version='F9 v1.1'
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[26]:

avg_payload_mass
2928.400000

First Successful Landing Date

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

Hint:Use min function

In [27]:
```
%sql select min(DATE) from SPACEXDATASET where landing__outcome='Success (drone ship)'
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[27]:

1
2016-04-08

Successful Drone Ship Landing with Payload Mass between 4000 & 6000

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

In [28]:
```
%sql select distinct booster_version from SPACEXDATASET where landing__outcome='Success (ground pad)' and payload_mass__kg_ between 4000 and 6000
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[28]:

booster_version
F9 B4 B1040.1
F9 B4 B1043.1
F9 FT B1032.1

Total Number of Successful & Failure Mission

List the total number of successful and failure mission outcomes

In [29]:
```
%sql select mission_outcome,count(*) as count from SPACEXDATASET group by mission_outcome
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[29]:

mission_outcome	COUNT
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carrying Maximum Payload Mass

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

In [30]:
```
%sql select distinct booster_version from SPACEXDATASET where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXDATASET)
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[30]:

booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2017 Launch Records

In [31]:
```
%sql select monthname(DATE) as month_names,landing__outcome,booster_version,launch_site \
from SPACEXDATASET where landing__outcome='Success (ground pad)' AND year(DATE)=2017
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[31]:

month_names	landing__outcome	booster_version	launch_site
February	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
May	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
June	Success (ground pad)	F9 FT B1035.1	KSC LC-39A
August	Success (ground pad)	F9 B4 B1039.1	KSC LC-39A
September	Success (ground pad)	F9 B4 B1040.1	KSC LC-39A
December	Success (ground pad)	F9 FT B1035.2	CCAFS SLC-40

Success Count Between 04/06/2010 & 20/03/2017

Rank the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order.

In [32]:
```
%sql select landing__outcome, count(*) as count from SPACEXDATASET where landing__outcome \
like '%Success%' and DATE between '2010-06-04' and '2017-03-20' group by landing__outcome order by count(*) desc
```

```
 * ibm_db_sa://dkl65719:***@dashdb-txn-sbox-yp-lon02-13.services.eu-gb.bluemix.net:50000/BLUDB
Done.
```

Out[32]:

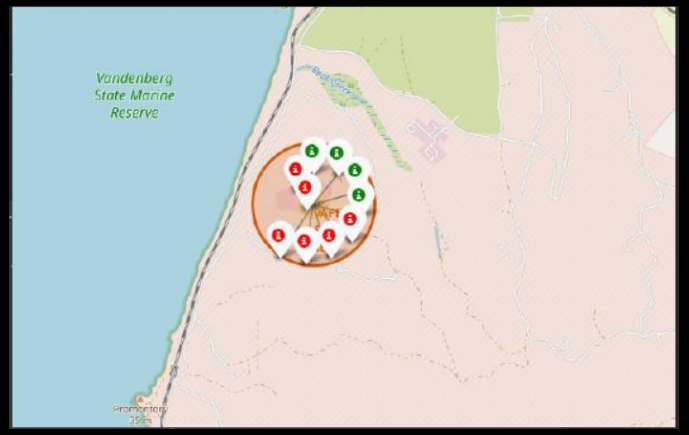
landing__outcome	COUNT
Success (drone ship)	5
Success (ground pad)	3

Interactive map with Folium

SpaceX Launch Sites

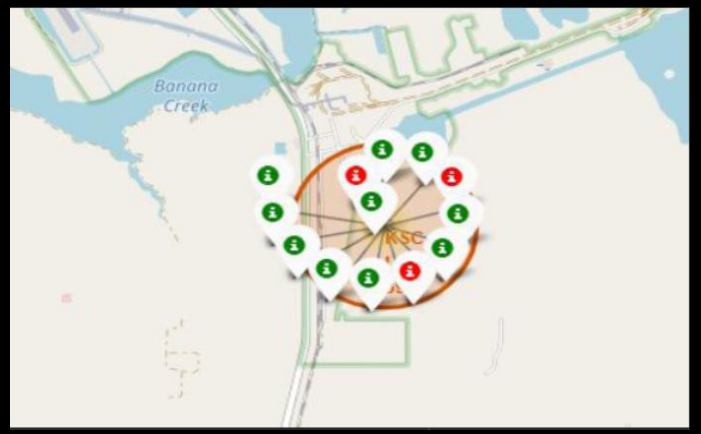
- All the SpaceX launch sites are located at the coast of USA
- VAFB SLC-4E site is located near the Lompoc Airport, California
- While, KSC LC 39 A , CCAFS SLC-40 and CCAFS LC-40 are at the coast of Titusville, Florida

SpaceX Launch Sites

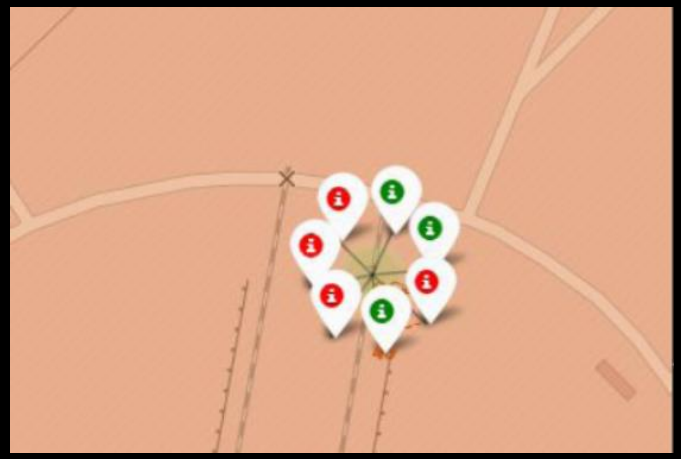


VAFB SLC-4E

- Green – Successful Mission
- Red – Failure Mission



KSC LC 39 A

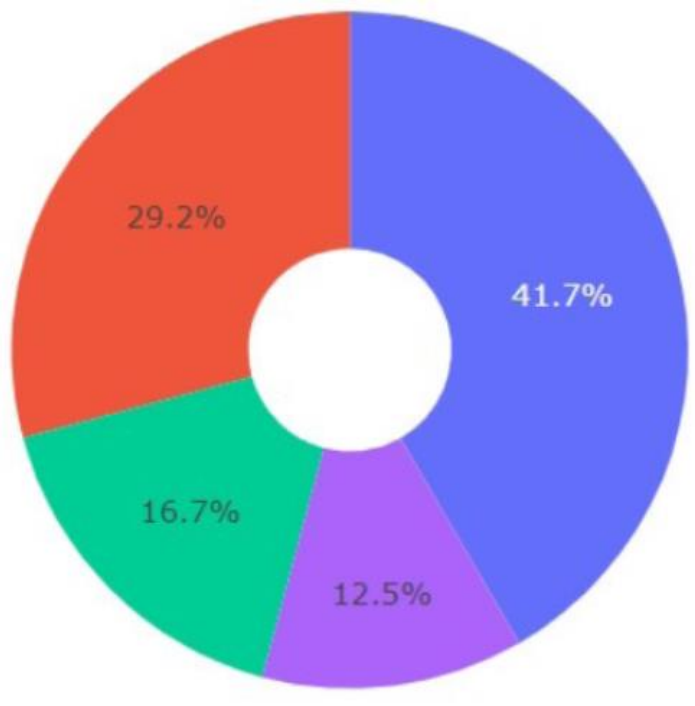


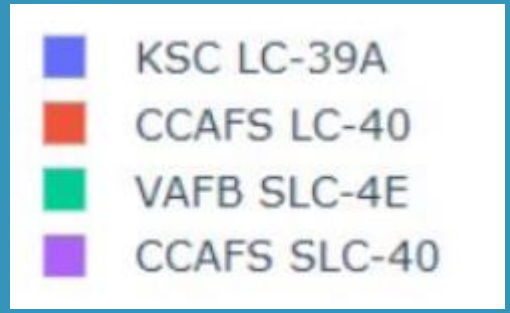
CCAFS SLC-40

CCAFS LC-40

Build a Dashboard with Plotly Dash

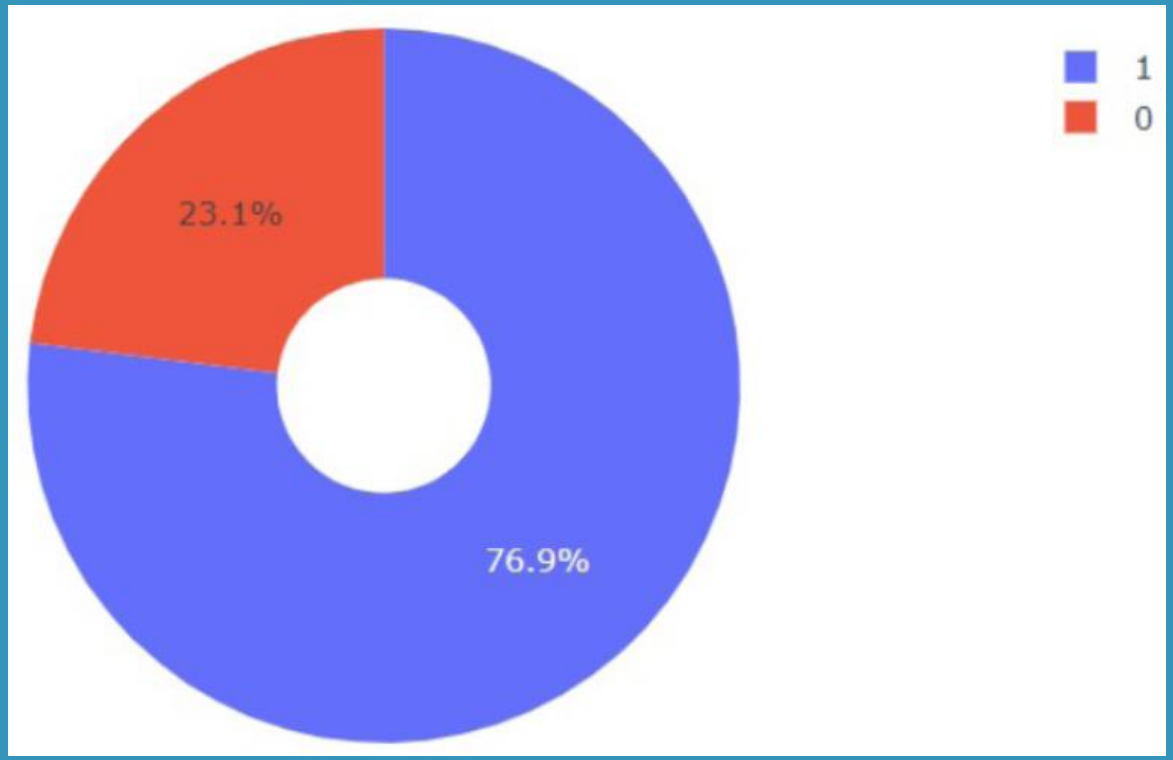
Dashboard – Total Success Launches by All Sites





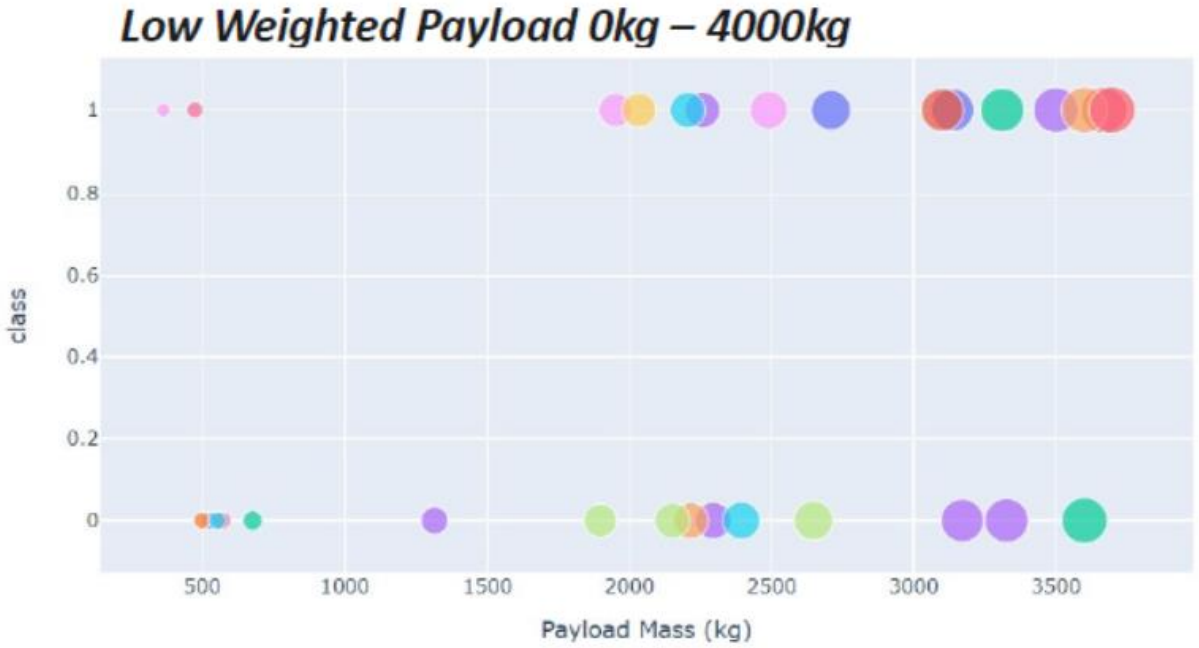
We can see that KSC LC – 39A had the most successful launches amongst all the other sites

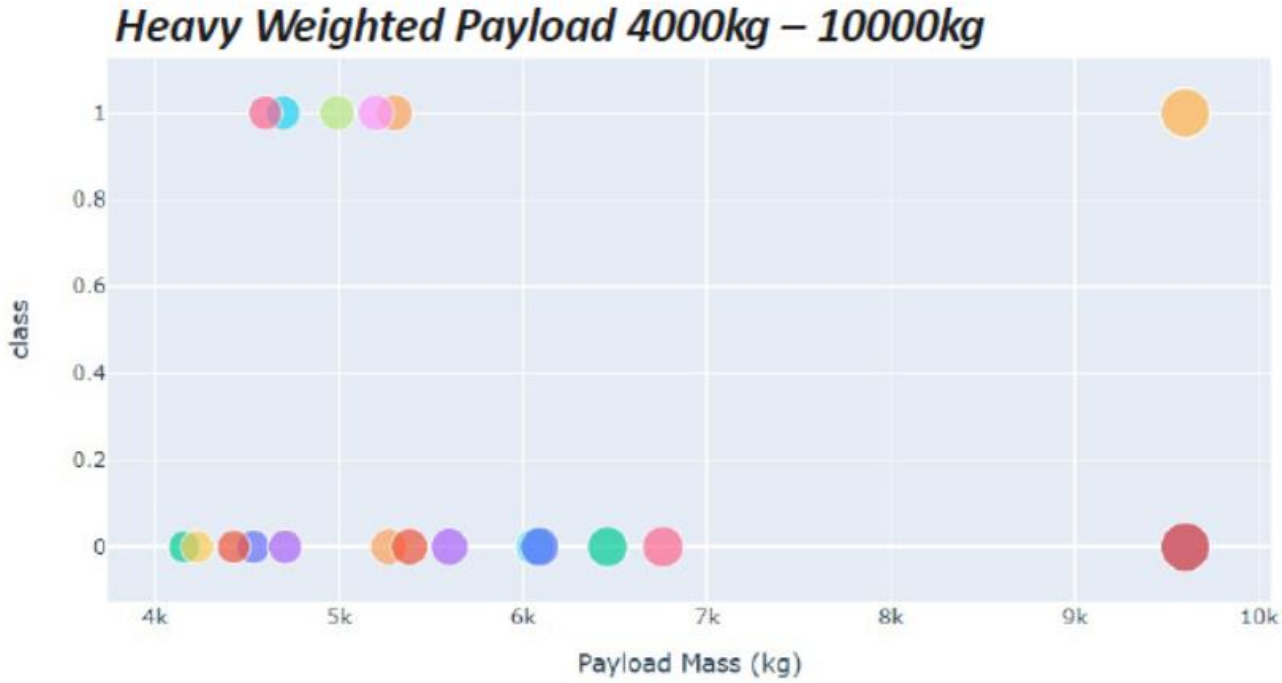
Dashboard – Launch Site With Highest Launch Success Ratio



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

Dashboard – Payload vs Launch Outcome Scatterplot



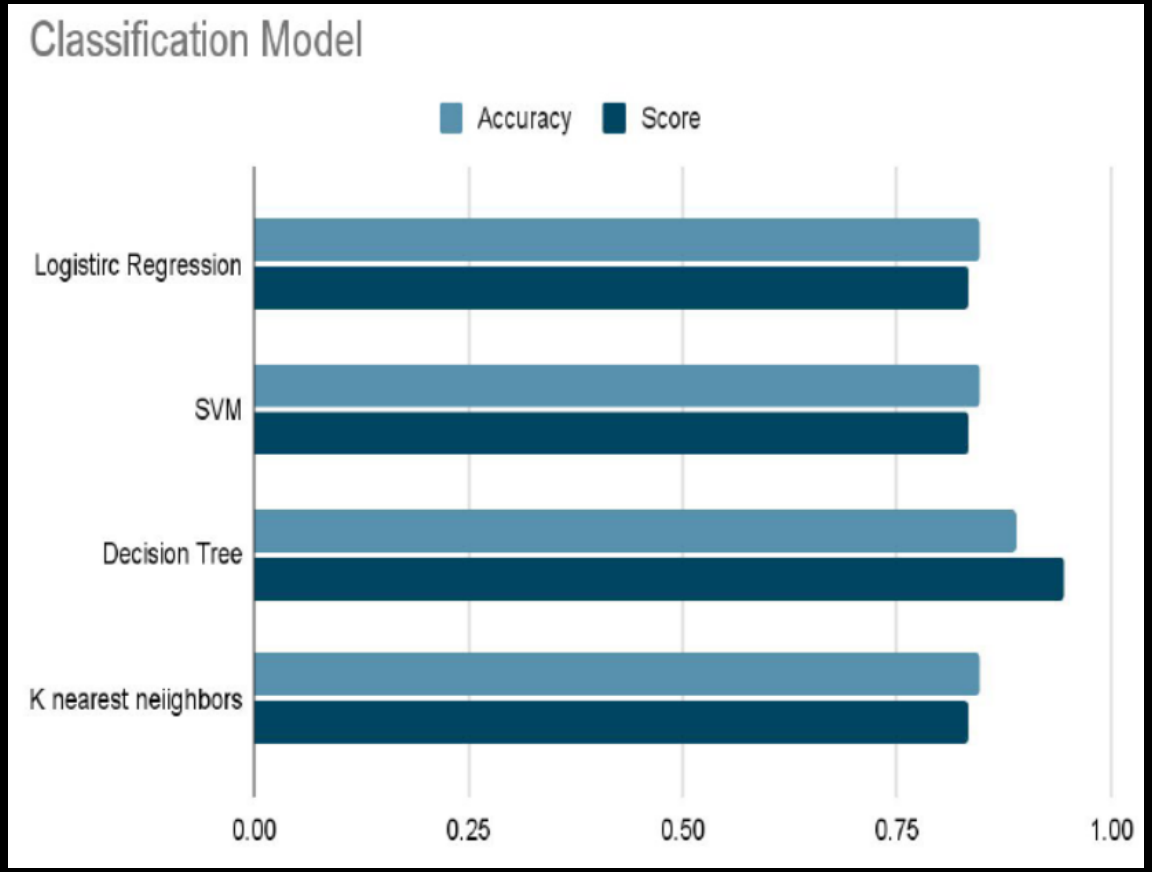


It is evident that the success rates for low weighted payloads is higher than the heavy weighted payloads

Predictive Analysis (Classification)

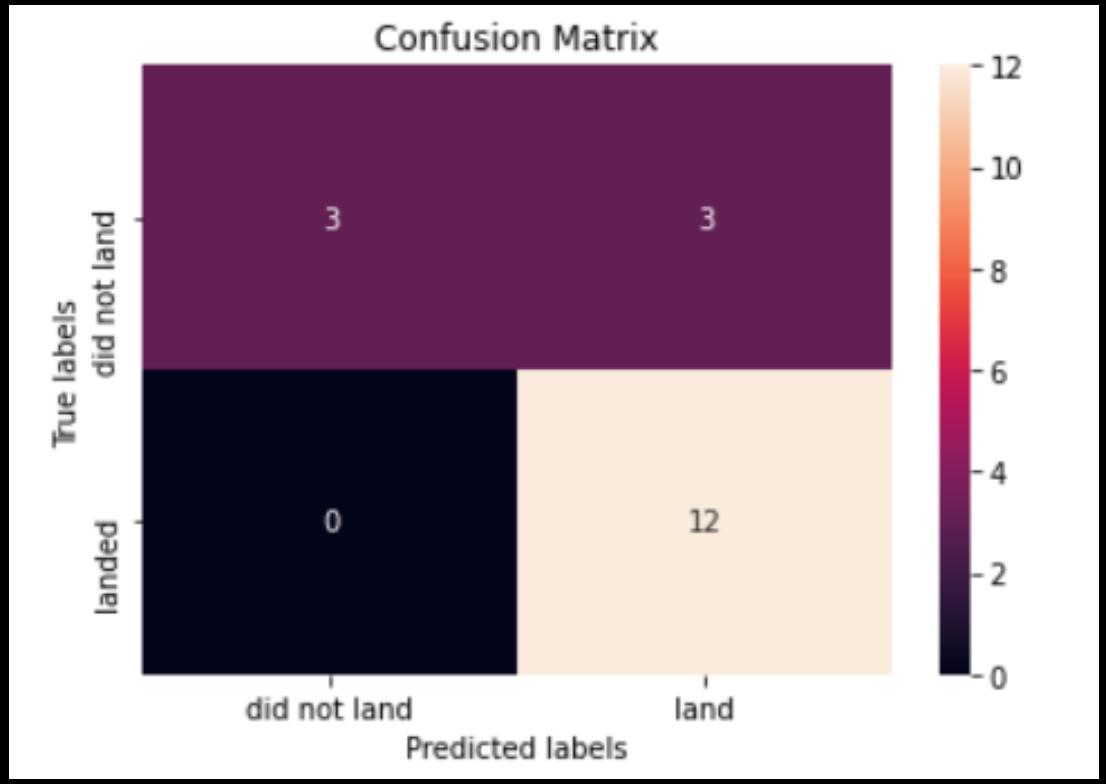
Classification Accuracy

- As you can see our accuracy is extremely close but we do have a winner its down to decimal places!
- The Best Algorithm for SpaceX model is Decision Tree
- After selecting the best hyperparameters for the decision tree classifier using the validation data, we achieved 83.33% accuracy on the test data.



Confusion Matrix

- Examining the confusion matrix, we see that Decision Tree can distinguish between the different classes. We see that the major problem is false positives.
- Confusion Matrix illustrates that it has accurately predict 15 out of 18 launches

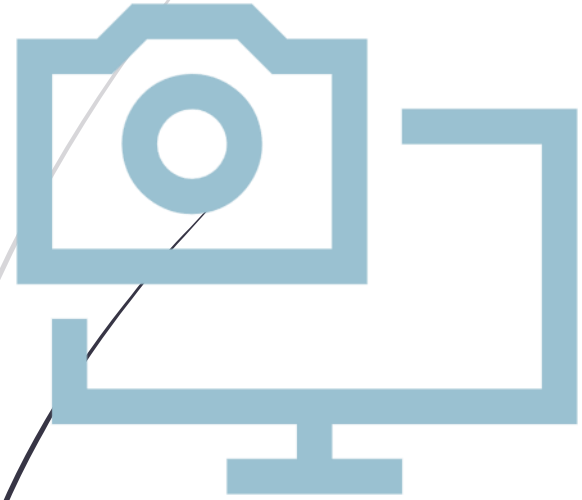


Confusion Matrix For Decision Tree Classifier

CONCLUSION

- The Tree Classifier Algorithm is the best for Machine Learning for this dataset
- Low weighted payloads perform better than the heavier payloads
- We can see that KSC LC-39A had the most successful launches from all the sites
- Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate
- The Success Rate has been significantly increasing since 2013

Appendix



Assets Used

- Python
- SQL
- Plotly Dash
- Folium
- Pandas
- Numpy
- Matplotlib
- Seaborn
- IBM Watson
- IBM DB2
- Scikit Learn
- Line Chart
- Wikipedia : https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- SpaceX REST API : https://api.spacexdata.com/v4/launches/past

THANK YOU