IBM Data Science Capstone-SpaceX

KUMER SAURAV KESHRI

Table of Contents

Executive Summary

Methodologies

- ❑ Data Collection
- ❑ Data Wrangling
- ❑ EDA With Data Visualization
- ❑ EDA with SQL
- ❑ Interactive Maps with Folium
- ❑ Dashboard with Plotly Dash
- ❑ Predictive Modelling

Results

- ❑ Exploratory Data Analysis
- ❑ Interactive Visualization
- ❑ Predictive Modelling

Conclusion

Introduction

Background

The goal of this project is to predict if the first stage of SpaceX Falcon-9 will land successfully. SpaceX claims that Falcon-9 launch will be much cheaper than other rocket launches, as SpaceX can re-utilize the first stage.

Therefore, by predicting the success of the first launch of Falcon-9, the cost of rocket launch can be also estimated. This information can be helpful to other organizations planning to launch rockets themselves

Problems to be Analyzed

- Variables that affect rocket launching
- Relationships between different factors of rocket launching and their influence on the success of rocket launching
- The optimum value of the variables and factors that lead to the success of rocket launching

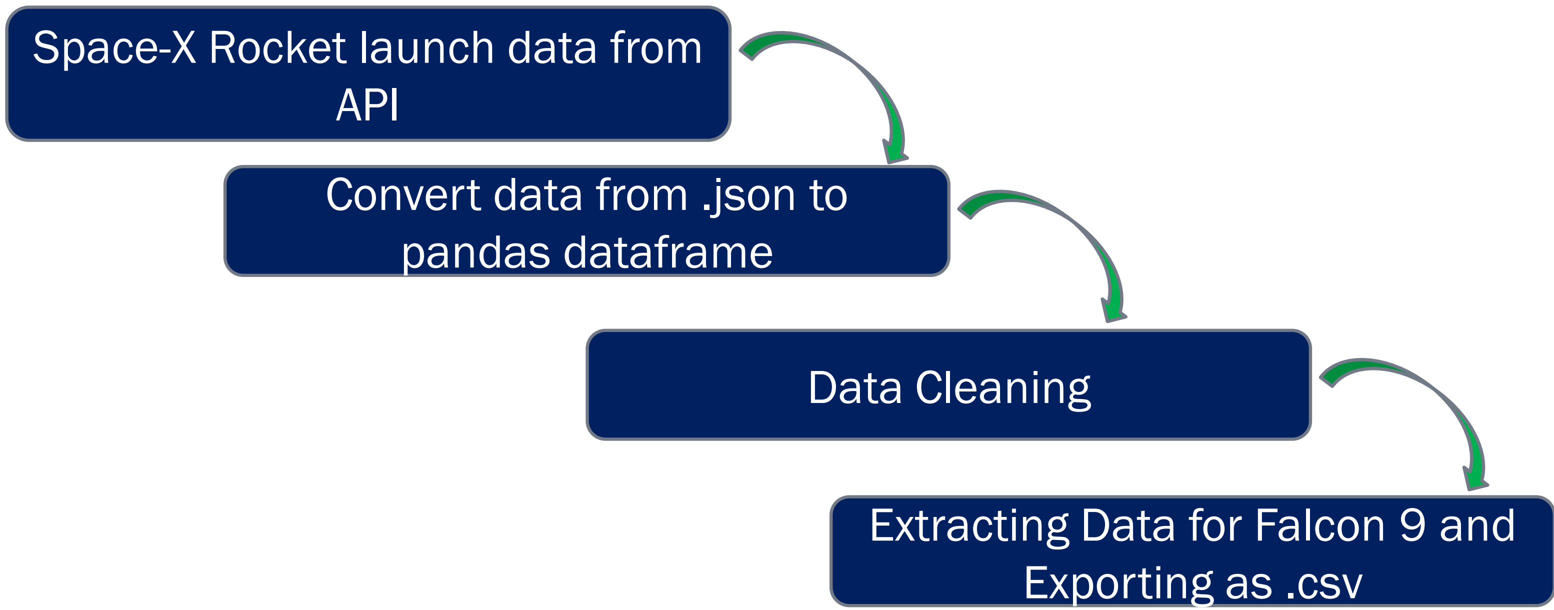
Methodology

Data Collection

GitHub url for Data Collection:

https://github.com/Kumer1991/Coursera_Capstone/blob/3861b0e858521eef66f82cccccf92ab367001afd/Data%20Collection.ipynb

Data Collection Flow Chart



Methodology

Data Collection

Summary. The data on SpaceX launches were obtained from the API and static response objects. The .json data was converted to pandas dataframe, cleaned with custom functions, dictionary was added to the dataframe. Data regarding Falcon 9 were extracted and exported after initial wrangling

- Collection of SpaceX rocket launch-data from API and conversion of the response to json file

```
#Collecting the data
spacex_url = "https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)

#conversion of the response to json file
data = pd.json_normalize(response.json())
```

- Using static response object for more consistent json results

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
response = requests.get(static_json_url)
```

Methodology

Data Collection

❑ Using functions to clean data

```
# Call getBoosterVersion
getBoosterVersion(data)
```

```
# Call getPayloadData
getPayloadData(data)
```

```
# Call getLaunchSite
getLaunchSite(data)
```

```
# Call getCoreData
getCoreData(data)
```

❑Assigning Dictionary to the Dataframe

```
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}
```

Methodology

Data Collection

- ❑ Extracting data for Falcon 9 and initial data wrangling

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

Falcon 9 Table

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitu
4	1	2010-06-04	Falcon 9	6123.547647	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.5773
5	2	2012-05-22	Falcon 9	525.000000	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.5773
6	3	2013-03-01	Falcon 9	677.000000	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.5773
7	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.6108
8	5	2013-12-03	Falcon 9	3170.000000	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-80.5773
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
89	86	2020-09-03	Falcon 9	15600.000000	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	7	B1060	-80.6039
90	87	2020-10-06	Falcon 9	15600.000000	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	7	B1058	-80.6039
91	88	2020-	Falcon 9	15600.000000	VLEO	KSC LC	True	6	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	9	B1051	-80.6039

- ❑ Export data as .csv file

```
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

Methodology

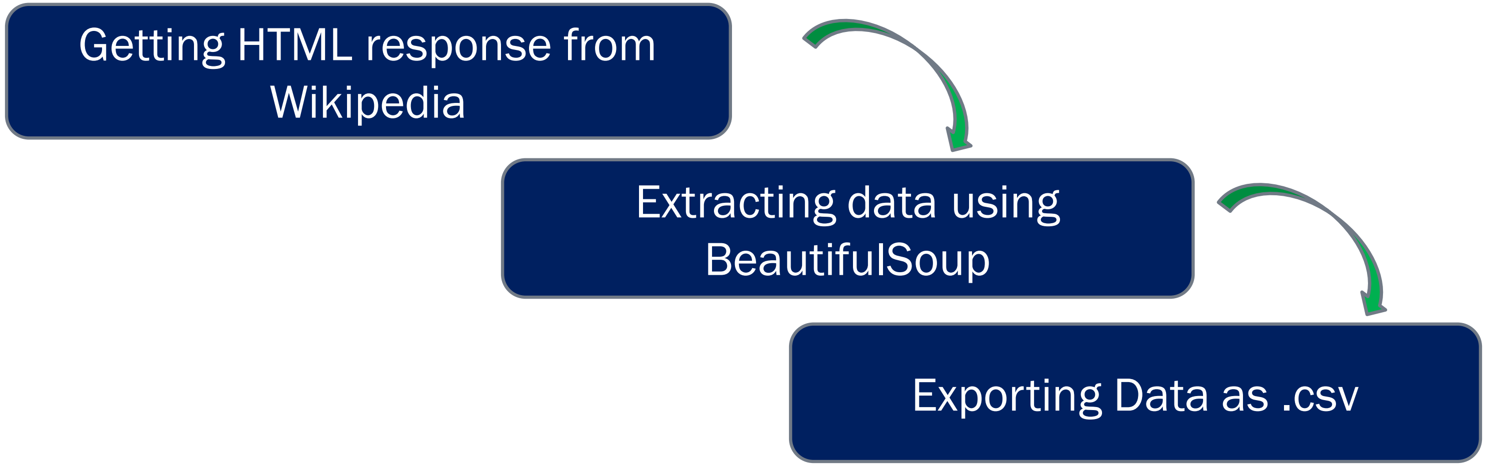
Web-Scrapping

Summary. Falcon-9 heavy launches record were web-scrapped from Wikipedia

GitHub link for Web-scrapping:

https://github.com/Kumer1991/Coursera_Capstone/blob/3861b0e858521eef66f82cccccf92ab367001afd/Web%20Scrapping.ipynb

Web Scrapping Flow Chart



Methodology

Web-Scrapping

Summary. Falcon-9 heavy launches record were web-scrapped from Wikipedia

- ❑ Necessary Packages are imported

```
!pip3 install beautifulsoup4
!pip3 install requests
```

- ❑ Getting HTML data

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

- ❑ BeautifulSoup object Creation

```
Soup = BeautifulSoup(response.text, 'html.parser')
```

- ❑ Finding HTML table

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

- ❑ Extracting Column names

```
column_names = []


th = Soup.find_all('th')

for x in range(len(th)):
    try:
     name = extract_column_from_header(th[x])
     if (name is not None and len(name) > 0):
        column_names.append(name)
    except:
     pass
```

Methodology

Web-Scrapping

❑ Creating Dictionary

```
launch_dict= dict.fromkeys(column_names)

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


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'] = []

launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

❑Converting Dictionary to dataframe and exporting as .csv

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

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

Methodology

Data-Wrangling

GitHub link for Data-Wrangling

https://github.com/Kumer1991/Coursera_Capstone/blob/3861b0e858521eef66f82cccccf92ab367001afd/Data%20Wrangling.ipynb

- ❑ Calculating the number of launches on each site

```
df["LaunchSite"].value_counts()
```

```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

- ❑ Calculating the number of occurrences of each dedicated orbits in the orbit column

```
df["Orbit"].value_counts("Orbit")
```

```
GTO      0.300000
ISS      0.233333
VLEO     0.155556
PO       0.100000
LEO      0.077778
SSO      0.055556
MEO      0.033333
GEO      0.011111
HEO      0.011111
ES-L1    0.011111
SO       0.011111
Name: Orbit, dtype: float64
```

Methodology

Exploratory Data Analysis with Data Visualization

GitHub link for Data Analysis with Data Visualization:

https://github.com/Kumer1991/Coursera_Capstone/blob/3861b0e858521eef66f82ccccccf92ab367001afd/Data%20Visualization.ipynb

The following graphs were drawn to analyze the data

- Scatter Graphs
  - Flight Number vs Payload Mass
  - Launch Site vs Flight Number
  - Launch Site vs Payload Mass
  - Orbit vs Flight Number
  - Orbit vs Payload Mass

  The scatter graphs can show a large amount of data and the co-relation between variables effectively

- Bar Diagram
  - Orbit vs Success Rate (Mean)

  The bar diagrams are effective in visualizing significant changes in the data

- Line Graph
  - Year vs Success Rate

  The line graphs are useful to identify trends in a dataset

Methodology

Exploratory Data Analysis by SQL

GitHub link for EDA with SQL

https://github.com/Kumer1991/Coursera_Capstone/blob/3861b0e858521eef66f82cccccf92ab367001afd/EDA%20with%20%20SQL.ipynb

SQL queries were performed to gather useful informations about the dataset, e.g.,

- **Unique rocket Launch Sites in the SpaceX mission**
- **Displaying 5 records where launch sites begin with the string 'CCA'**
- **Displaing the total payload mass carried by boosters launched by NASA (CRS)**
- **Displaying average payload mass carried by booster version F9 v1.1**
- **Listing the date when the first successful landing outcome in ground pad was achieved**
- **Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000**
- **Listing the total number of successful and failure mission outcomes**
- **Listing the names of the booster_versions which have carried the maximum payload mass**
- **Listing the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015**
- **Ranking the count of landing outcomes of failure or success between the date 2010-06-04 and 2017-03-20, in descending order**

Methodology

Interactive Map with Folium

GitHub link for Folium

https://github.com/Kumer1991/Coursera_Capstone/blob/3861b0e858521eef66f82cccccf92ab367001afd/Folium.ipynb

Tasks Performed in Folium

- **Visualizing Launch Sites on Interactive Map:** The latitude and longitude co-ordinates of the launch sites were taken and projected on the interactive map alongwith circular marker and site name along each launch site
- **Marking Failed and Successful Launches on Each Site:** The launch_outcomes of each site were marked with red (for failed launches) and green (for successful launches) on tha interactive map.
- **Calculation and Visualization of Distance between Launch Sites and Other Land Marks :** The distances between Launch Site VAFB SLC 4E and some landmarks like the nearest coastline, highway, railway and city were calculated using Haversine’s formula and were projected on the interactive map

Methodology

Predictive Analysis (Classification)

GitHub link for Predictive Analysis

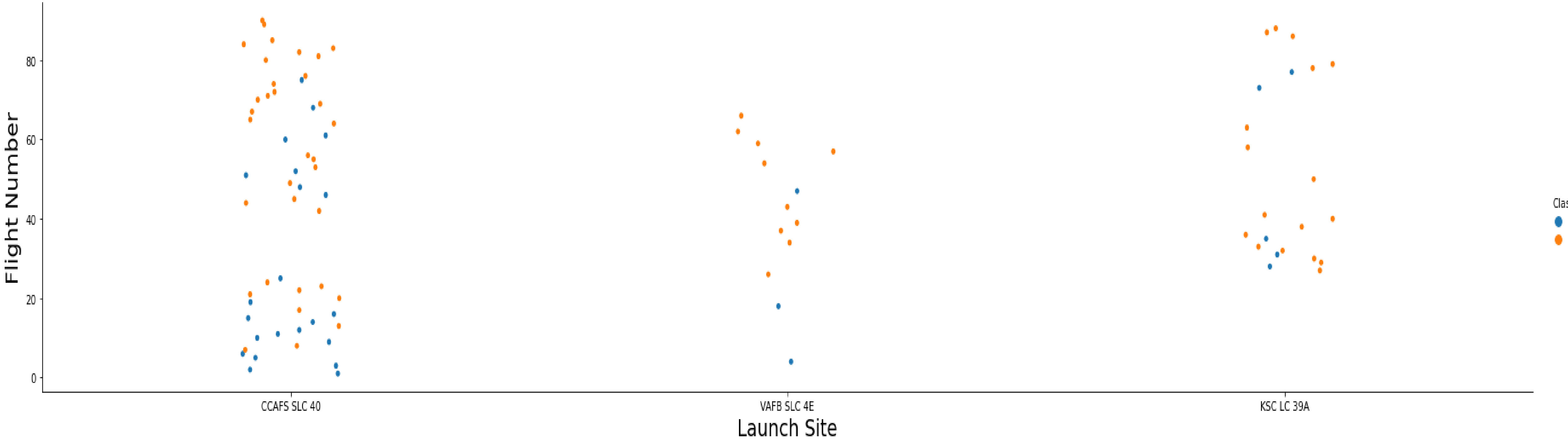
https://github.com/Kumer1991/Coursera_Capstone/blob/baaf67566b893d643c31e64d592a8bc24dbb9d56/Predictive%20Modelling.ipynb

Predictive Analysis

- Dataset was loaded into pandas and NumPy
- The input (X) and output (Y) variables were selected and transformed by preprocessing
- The data was split into training and testing datasets
- Different machine learning algorithms (Logical Regression, K-Nearest Neighbor, Support vector machine, Decision Tree etc) were used to predict the data
- The accuracy of each model was measured and the best model was evaluated

Results
EDA by Data Visualization

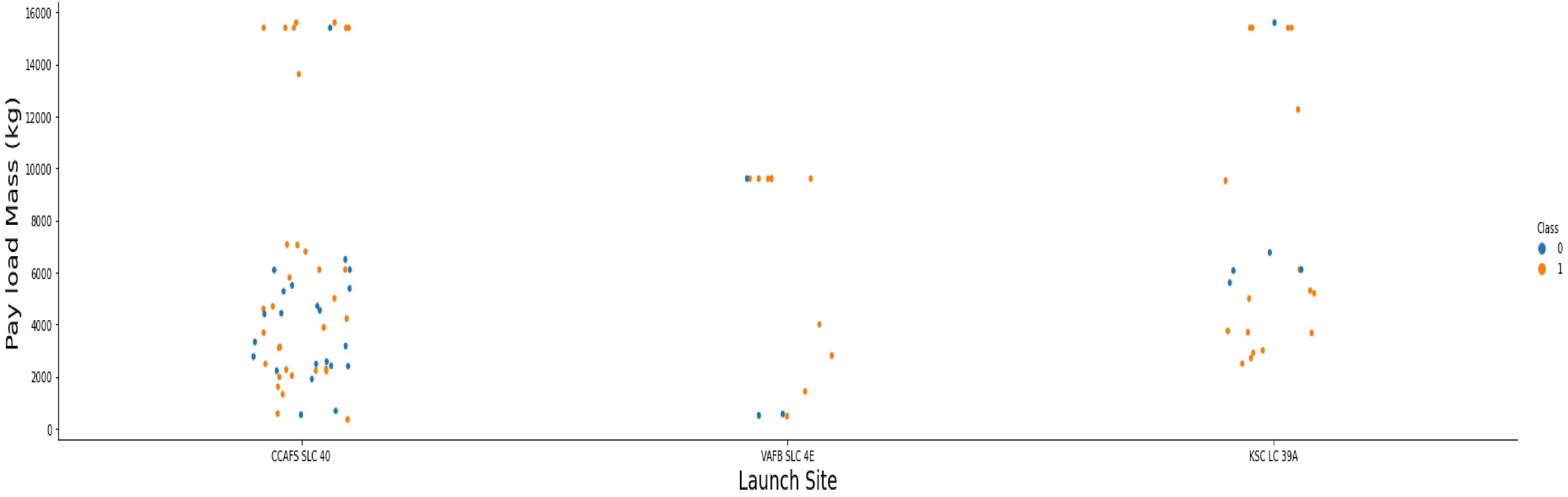
Flight Number vs Launch Site



Insight : The number of successful launches at a launch site increases with the total amount of flights at the launch site

Results
EDA by Data Visualization

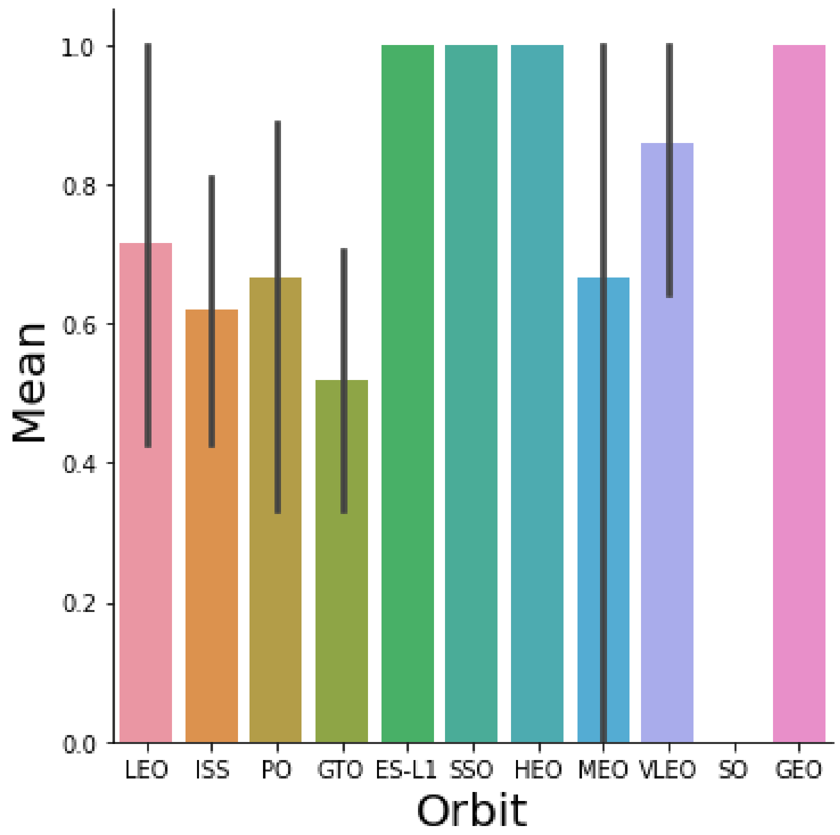
Payload Mass vs Launch Site



Insight: For CCAFS SLC 40 higher success is observed with higher payload mass. For VAB SLC 4E no flights were launched with payload mass higher than 10000kg. KSC LC 39A shows more number of launches and launch success at lower payload mass

Results
EDA by Data Visualization

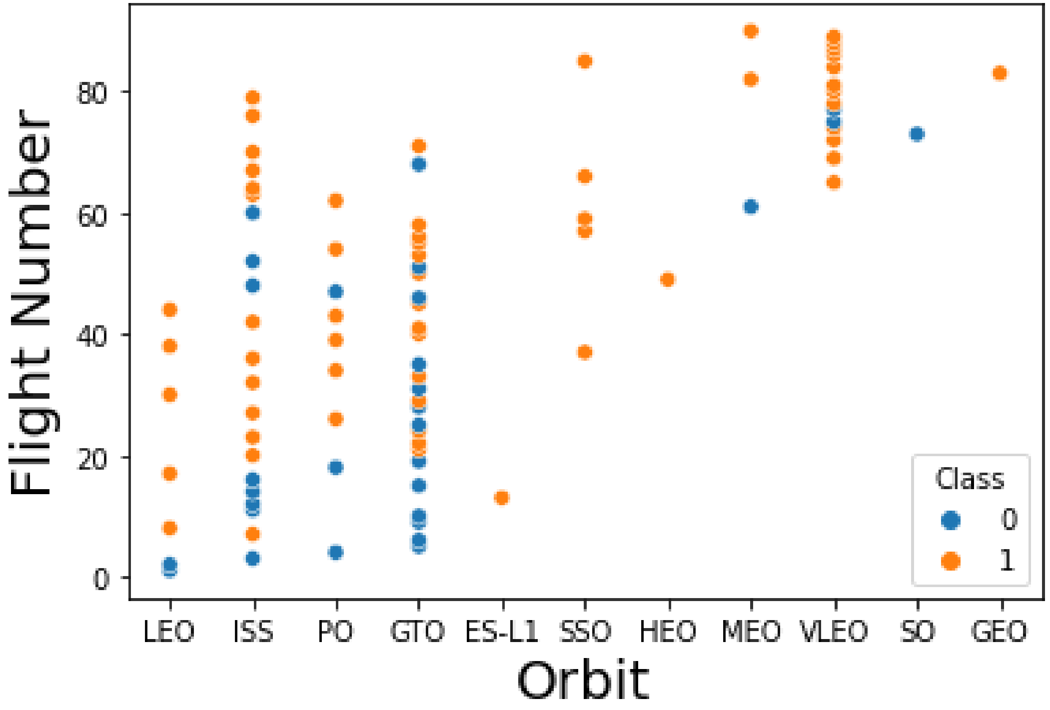
Mean (Success Rate) vs Orbit



Insights: The orbit GTO has the lowest success rate and the orbits ES-L1, SSO, HEO and GEO have the highest success rate

Results
EDA by Data Visualization

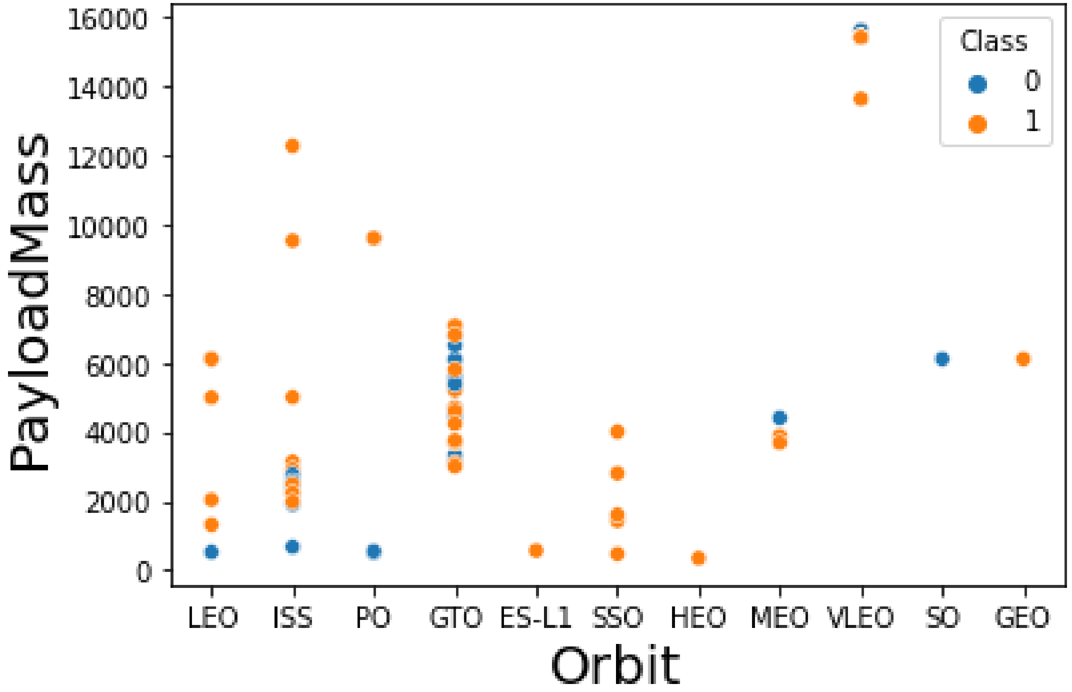
Flight Number vs Orbit



Insights: The SSO orbit has the most percentage of successful flight launches, on the other hand the GEO orbit has The least number of flight launches

Results
EDA by Data Visualization

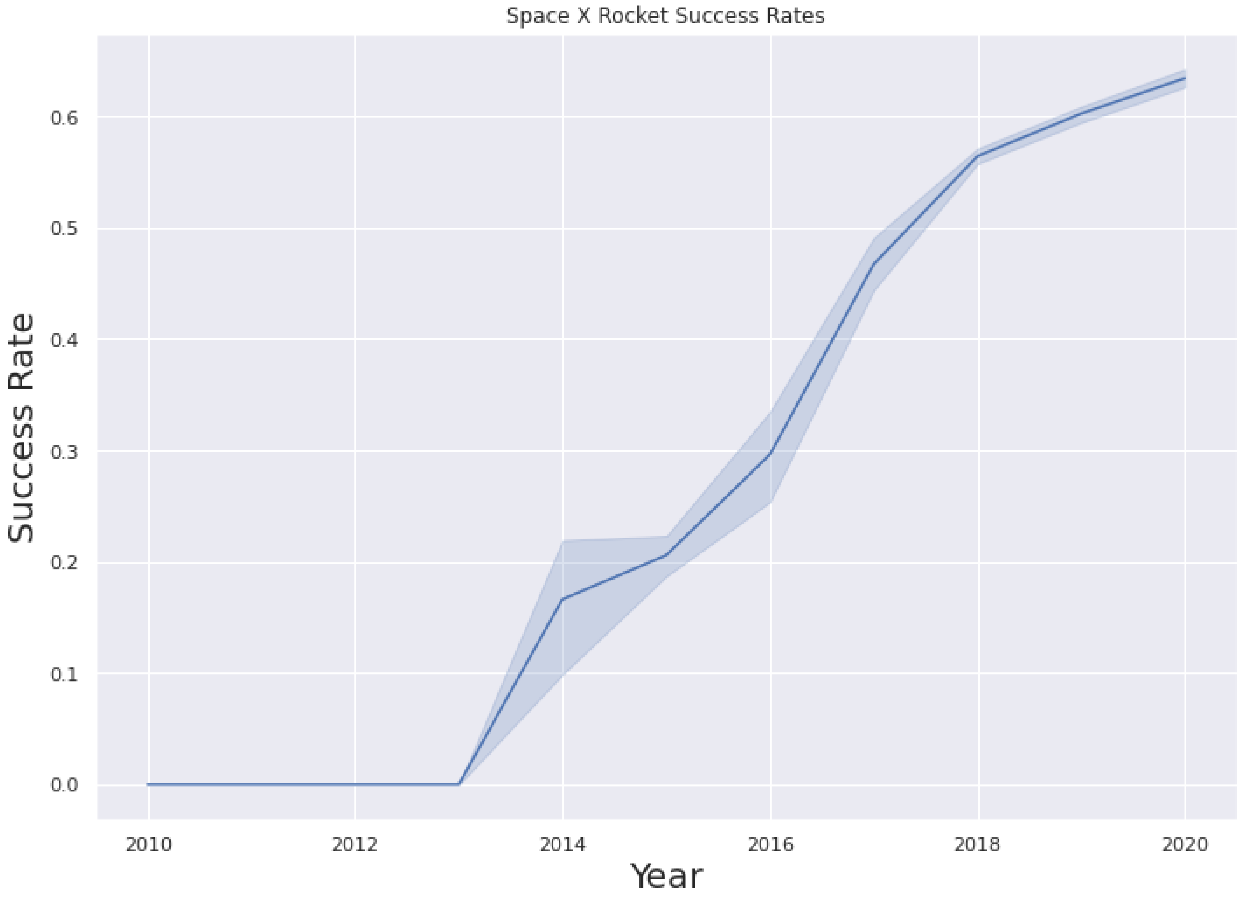
Payload Mass vs Orbit



Insights: The orbits like LEO show high success at medium payload mass and ISW shows high success at high payload mass

Results
EDA by Data Visualization

Success Rate vs Year



Insights: The success rate of rocket launches have continuously increased from 2013 to 2020

Results
EDA by SQL

Number of Unique Launch Sites:

```
In [90]: %sql SELECT UNIQUE(Launch_Site) from SPACEXDATASET

 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.
```

Out[90]:

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

Results
EDA by SQL

Query: Total payload mass where customer is NASA (CRS)

```
In [92]: %sql SELECT SUM(PAYLOAD_MASS__KG_) AS TotalPayLoadMass FROM SPACEXDATASET WHERE CUSTOMER LIKE 'NASA (CRS)'

 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.

Out[92]:
```

totalpayloadmass
91192

Results
EDA by SQL

Query : Average Payload mass where /booster Version is F9 v1.1

```
In [125]: %sql SELECT AVG(PAYLOAD_MASS__KG_) AvgPayLoadMass FROM SPACEXDATASET WHERE BOOSTER_VERSION = 'F9 v1.1'

 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.

Out[125]:
```

avgpayloadmass
2928

Results
EDA by SQL

Query: Date at which the first successful landing in ground pad was achieved

```
In [94]: %sql SELECT min(DATE) FIRSTLAND FROM SPACEXDATASET WHERE LANDING__OUTCOME = 'Success (ground pad)'

 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.

Out[94]:
```

firstland
2015-12-22

Results
EDA by SQL

Query: Name of boosters where the landing outcome was success in drone ship and the payload mass was between 4000 and 6000

```
In [95]: %sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE LANDING__OUTCOME = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000

 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.
```

Out[95]:

booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Results
EDA by SQL

Query : Total number of failure and success outcomes

```
%sql SELECT Count(Mission_Outcome) FAILURE from SPACEXDATASET where Mission_Outcome LIKE '%Failure%'

%sql SELECT Count(Mission_Outcome) SUCCESS from SPACEXDATASET where Mission_Outcome LIKE '%Success%'
```

```
 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.
 * ibm_db_sa://jkd82797:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/bludb
Done.
```

]:

success
200

Results
EDA by SQL

Query : The names of the booster versions which have carried the maximum payload mass

```
In [117]: %sql SELECT BOOSTER_VERSION, MAX(PAYLOAD_MASS__KG_) AS MaxiPayLoadMass FROM SPACEXDATASET GROUP BY BOOSTER_VERSION ORDER BY MaxiPayLoadMass
```

F9 v1.0 B0004	0
F9 B4 B1045.1	362
F9 FT B1038.1	475
F9 v1.0 B0006	500
F9 v1.1 B1003	500
F9 v1.0 B0005	525
F9 v1.1 B1017	553
F9 v1.1 B1013	570
F9 v1.0 B0007	677
F9 B5B1063.1	1192

Results
EDA by SQL

Query : 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

landing__outcome	COUNT
Failure (drone ship)	10
Success (drone ship)	10
Success (ground pad)	6
Failure (parachute)	2

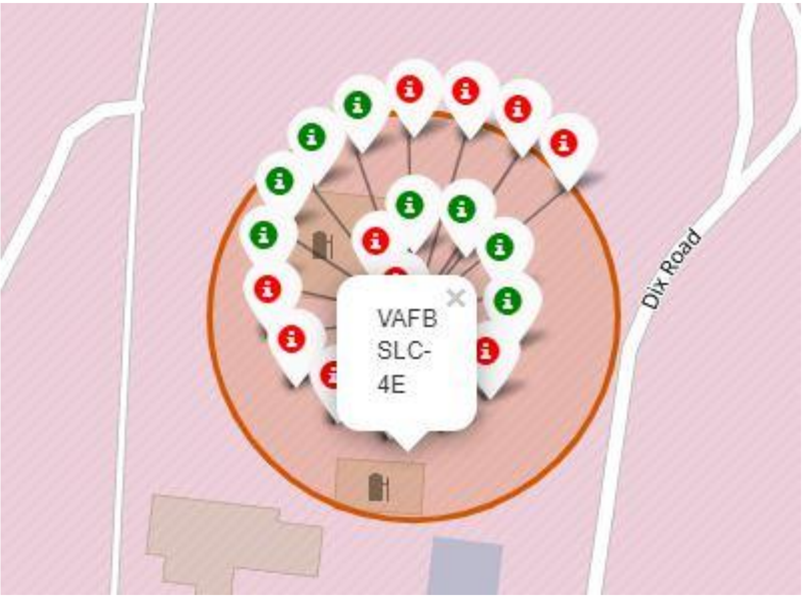
Results
Interactive Map with Folium

All Lauch Sites with Circular Markers

Insights: The launch sites of SpaceX are at the eastern and western coast sides of USA

Results
Interactive Map with Folium

Markers with Colors



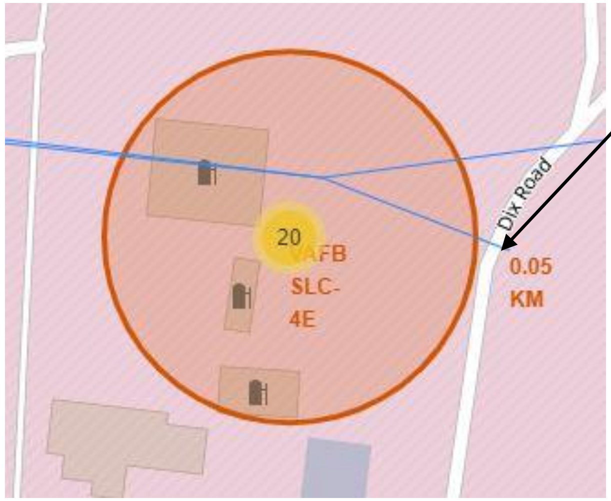
Insights: The red markers show failure and the green markers show the success of the launches

Results
Interactive Map with Folium

Launch Site Distance to landmarks using VAFB SLC 4E as reference

Highway

City



Coastline

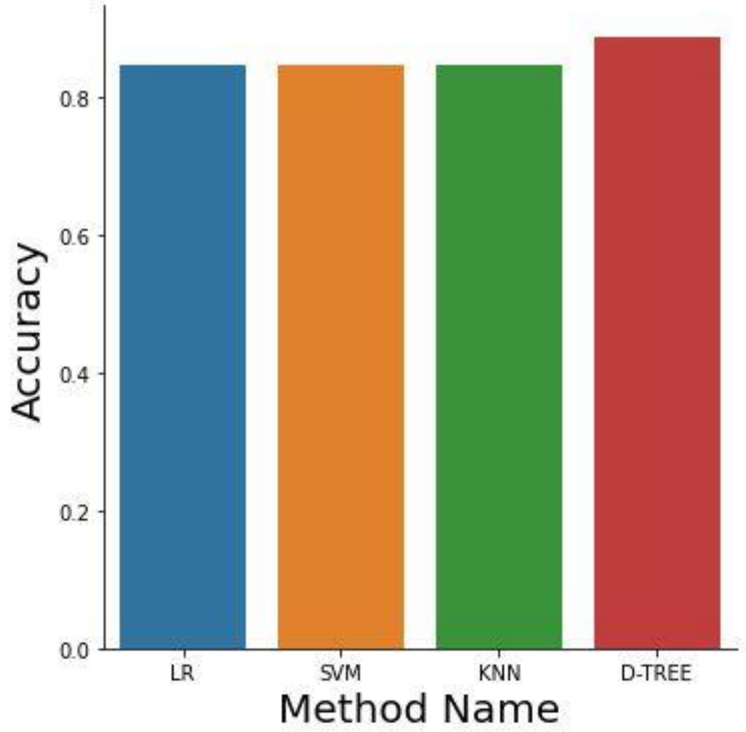
Railway

- •Are launch sites in close proximity to railways? Yes
- •Are launch sites in close proximity to highways? Yes
- •Are launch sites in close proximity to coastline? Yes
- •Do launch sites keep certain distance away from cities? No

Results
Predictive Analysis

Accuracy of Different Methods/Algorithms

	Method Name	Accuracy
0	LR	0.846429
1	SVM	0.848214
2	KNN	0.848214
3	D-TREE	0.889286

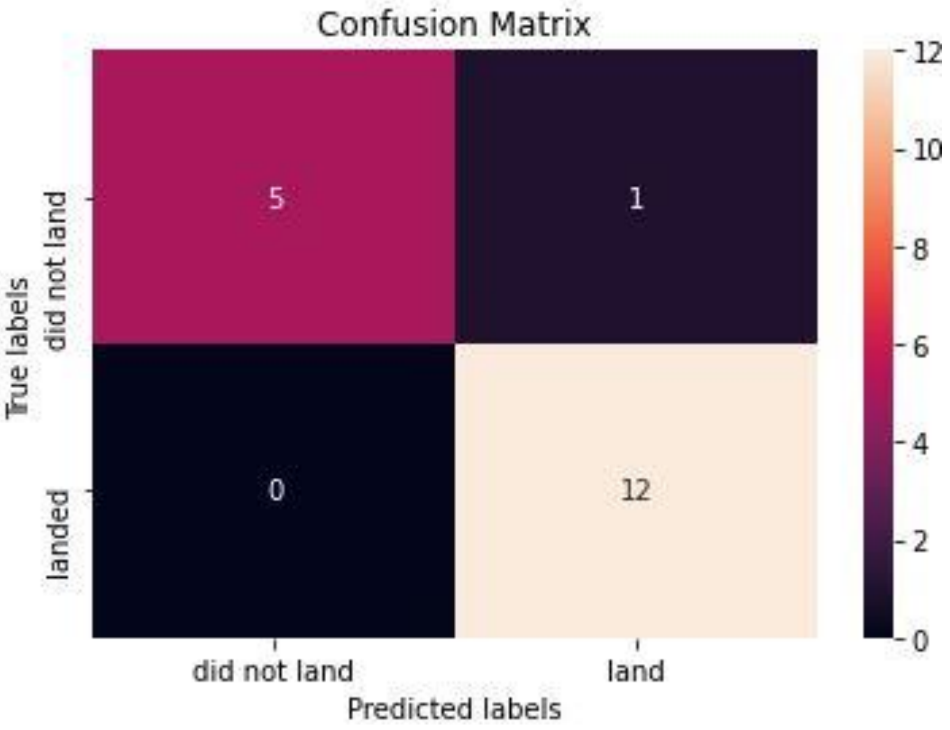


The Decision Tree/Tree method is the best method, the score/accuracy and best parameters are given below

```
Best Method is Decision Tree with a score of 0.8892857142857145
Best Params is : {'criterion': 'entropy', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 2, 'splitter': 'best'}
```

Results
Predictive Analysis

Confusion Matric for Tree Algorithm



Insights: It can be observed that the tree algorithm can distinguish the different classes (landed, did not land) in the data effectively. There is very few amount of false positive in the confusion matrix

Conclusion

- CCAFS SLC 40 and KSC LC 39 A has the most successful launches
- The Success rate of Space-X launches is increasing every year (from2013 to 2020)
- CCAFS SLC-40 has more successful launches at higher payload mass. But generally low to medium payload mass show better performance
- The orbits like ES-L1, SSO, HEO and GEO have the highest success rate
- All the launch sites are near coast-line, but are usually at significant distance from cities
- The Tree model can predict the landing outcomes with most accuracy

Thank You

KUMER SAURAV KESHRI