IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

KYRI CHANDRAKANTH
24-JULY-2022

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

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 (Classification)

- Summary of all results

  EDA results

  Interactive analytics

  Predictive analysis

Introduction

- Project background and context
  - SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.
- Problems you want to find answers
  - The project task is to predicting if the first stage of the SpaceX Falcon 9 rocket will land successfully

Section 1

Methodology

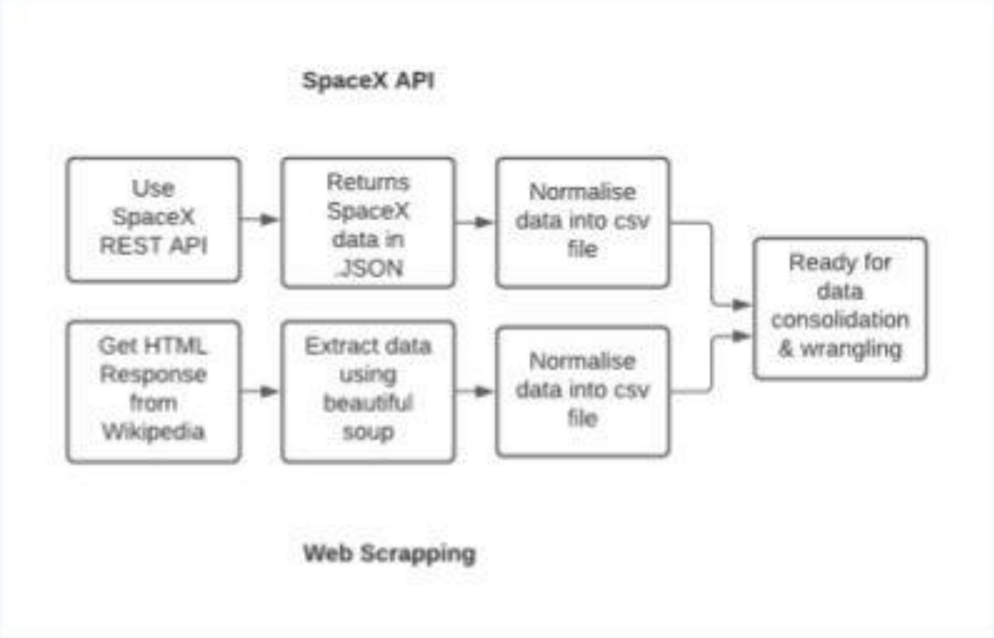
Methodology

Executive Summary

- Data collection methodology:
  - SpaceX Rest API
  - Web Scrapping from Wikipedia
- Perform data wrangling
  - One Hot Encoding data fields for Machine Learning and data cleaning of null values and irrelevant columns
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - LR, KNN, SVM, DT models have been built and evaluated for the best classifier

Data Collection

- The following datasets was collected:
  - SpaceX launch data that is gathered from the SpaceX REST API.
  - This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
  - The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.
  - Another popular data source for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup.



Data Collection – SpaceX API

- Data Collection with SpaceX REST calls

https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/Data%20Collection%20API.ipynb

To make the requested JSON results more consistent, we will use the following static response object for this project:

1. Getting response from API

In [60]:
```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
```

We should see that the request was successfull with the 200 status response code

In [61]:
```
response.status_code
```

Out[61]:
```
200
```

2. Converting response to a.json file

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

In [62]:
```
# Use json_normalize meethod to convert the json result into a dataframe
data = pd.json_normalize(response.json())
```

Finally lets construct our dataset using the data we have obtained. We we combine the columns into a dictionary.

3. Assign list to dictionary data frame

In [72]:
```
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}
```

4. Filter data frame and export to flat file (.csv)

[79]:
```
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping

- Web Scrapping from Wikipedia

https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

1 .Getting Response from HTML

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

2. Creating BeautifulSoup Object

```
soup = BeautifulSoup(page.text, 'html.parser')
```

3. Finding tables

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

4. Getting column names

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

5. Creation of dictionary

```
launch_dict= dict.fromkeys(column_names)

# Remove an 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']=[]
```

6. Appending data to keys (refer) to notebook block 12

```
In [12]: extracted_row = 0
         #Extract each table
         for table_number,table in enumerate
             # get table row
              for rows in table.find_all("tr")
                   #check to see if first table
```

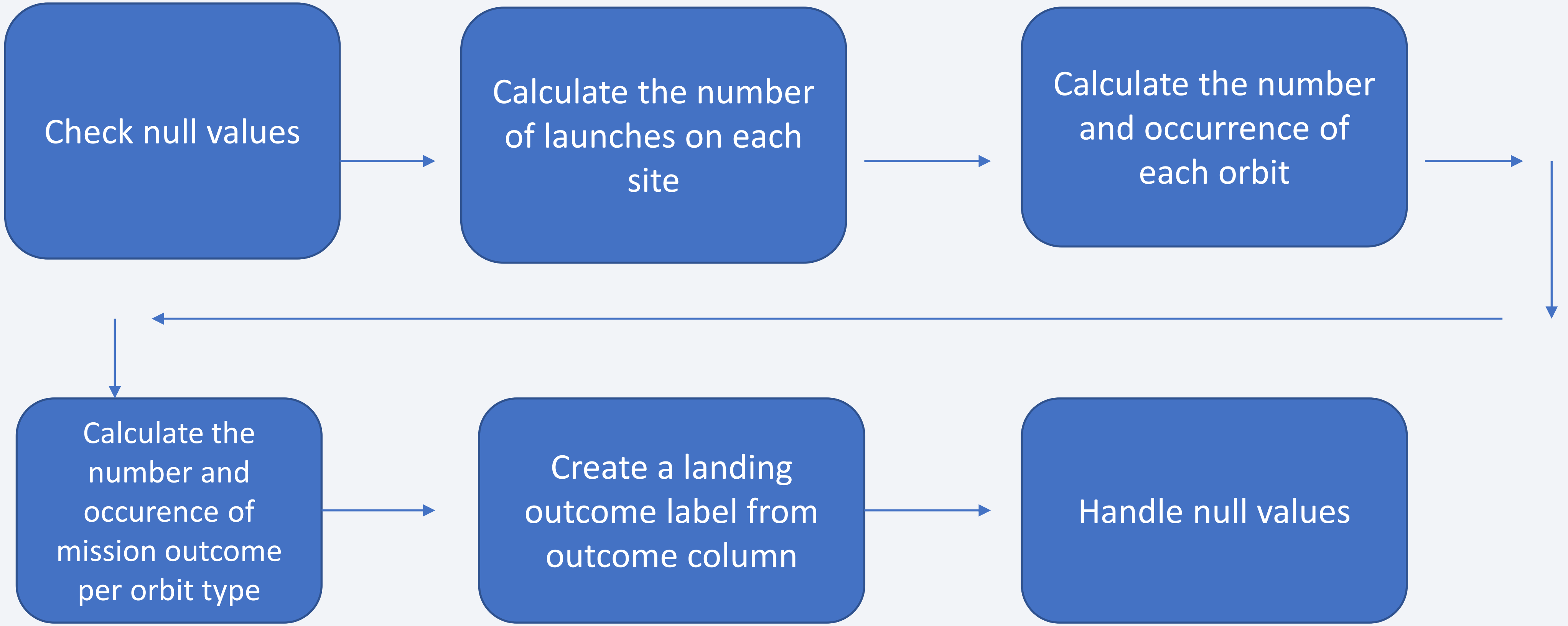
7. Converting dictionary to dataframe

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

8. Dataframe to .CSV

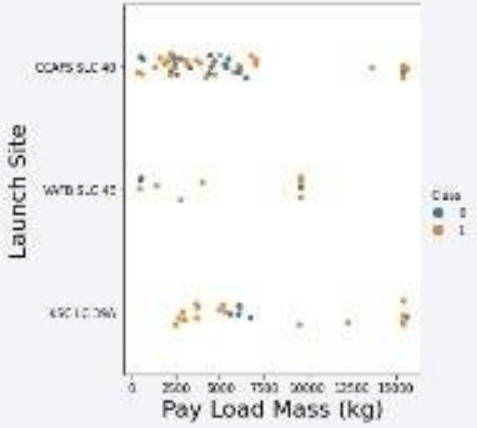
```
df.to_csv('spacex_web_scraped.csv', index=False)
```

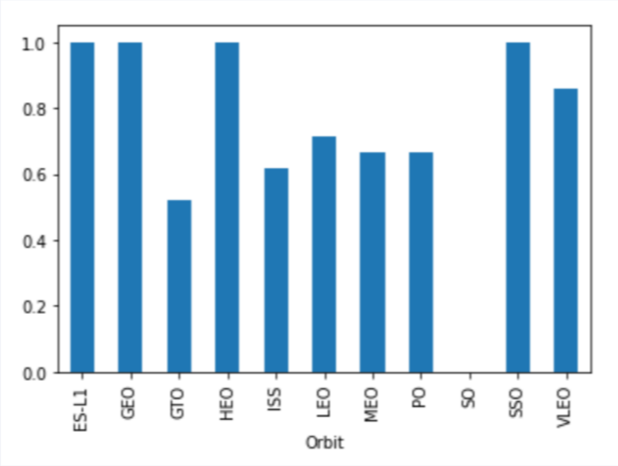
Data Wrangling

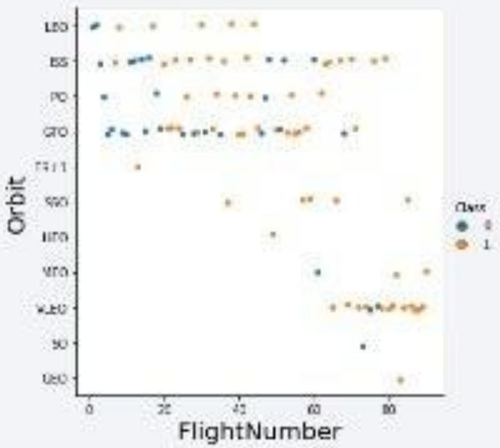


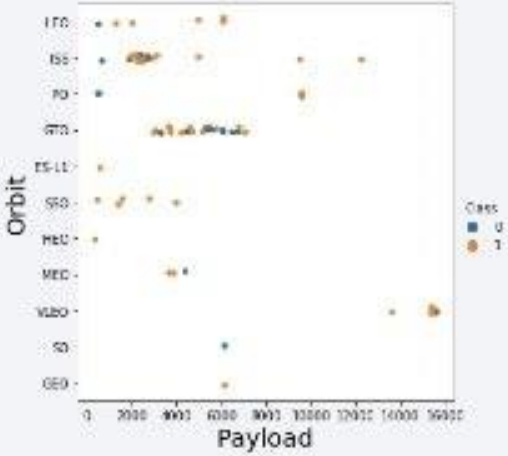
https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/EDI.ipynb

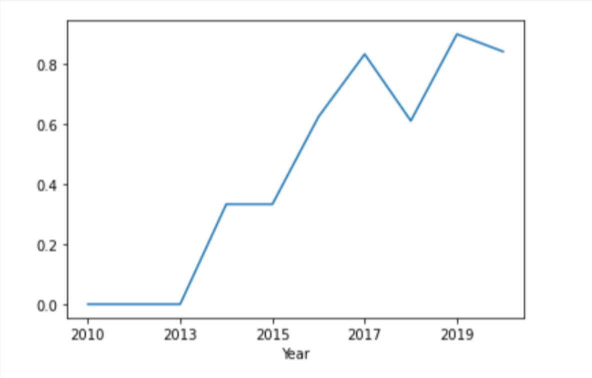
EDA with Data Visualisation











https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

- SQL queries performed include:
  - Displaying the names of the unique launch sites in the space mission
  - Displaying 5 records where launch sites begin with the string 'KSC'
  - Displaying 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 where the successful landing outcome in drone ship was achieved.
  - Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000

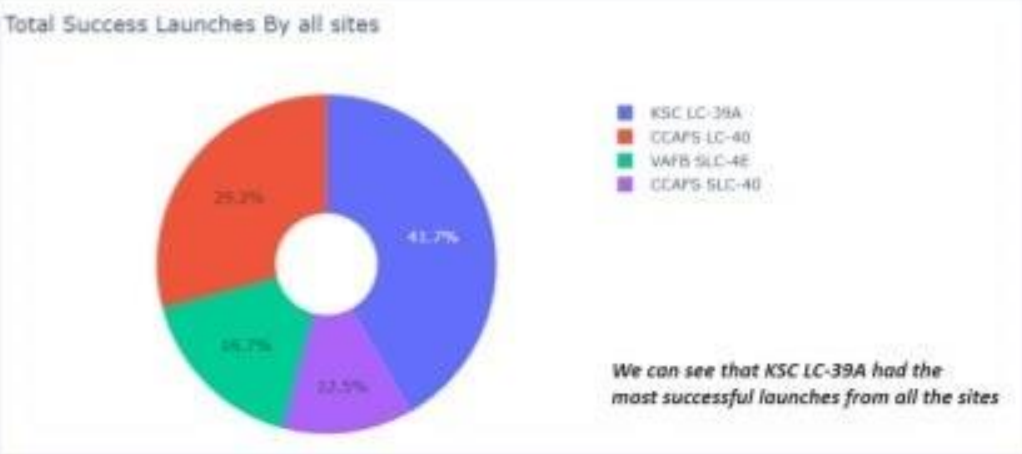
  but less than 6000
  - Listing the total number of successful and failure mission outcomes
  - Listing the names of the booster_versions which have carried the maximum payload mass.
  - Listing the records which will display the month names, successful landing_outcomes in ground pad ,booster
  - versions, launch_site for the months in year 2017
  - Ranking the count of successful landing_outcomes between the date 2010 06 04 and 2017 03 20 in descendingorder.

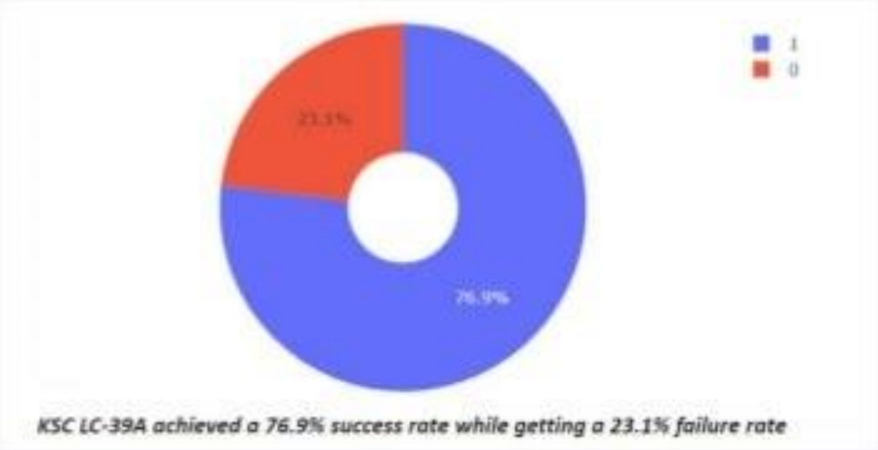
https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/EDA%20with%20SQL.ipynb

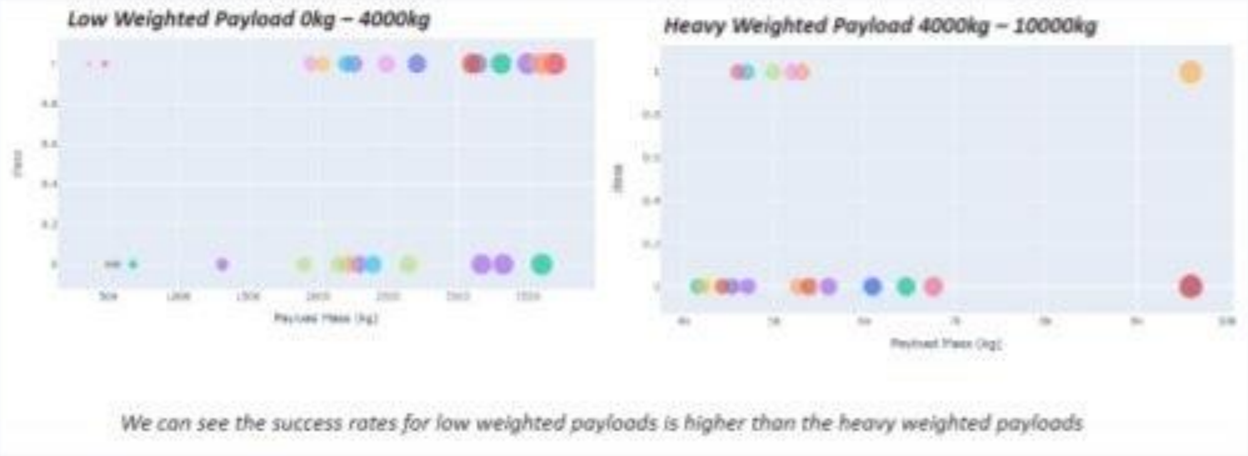
Build an Interactive Map with Folium

https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash



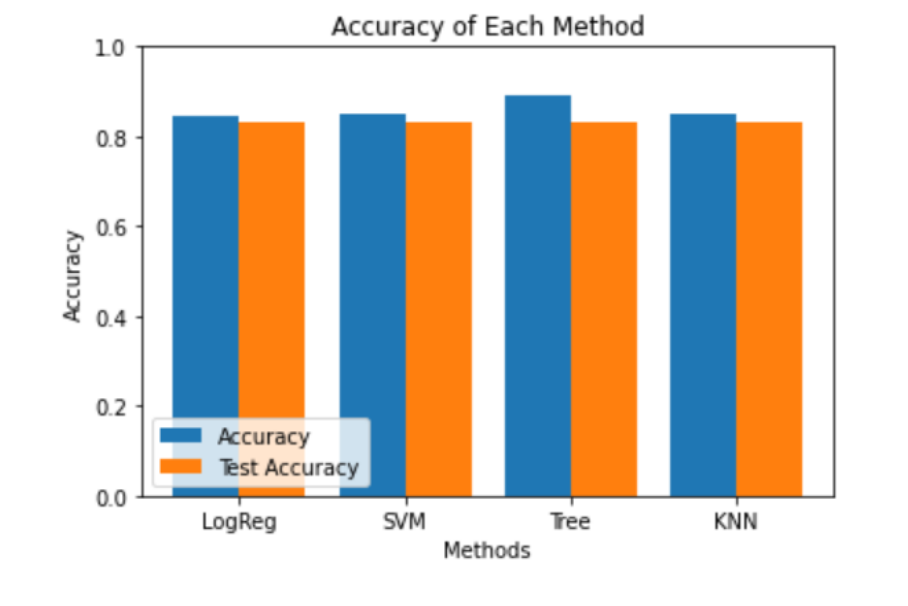


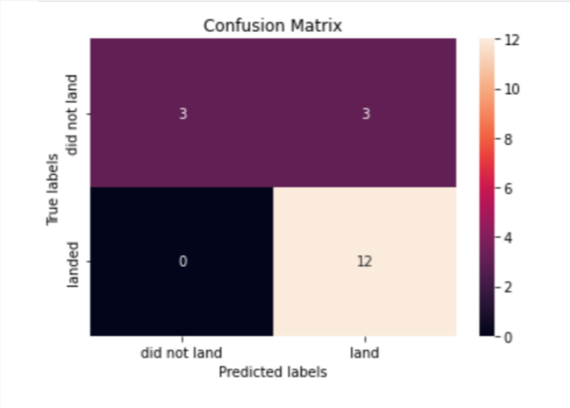


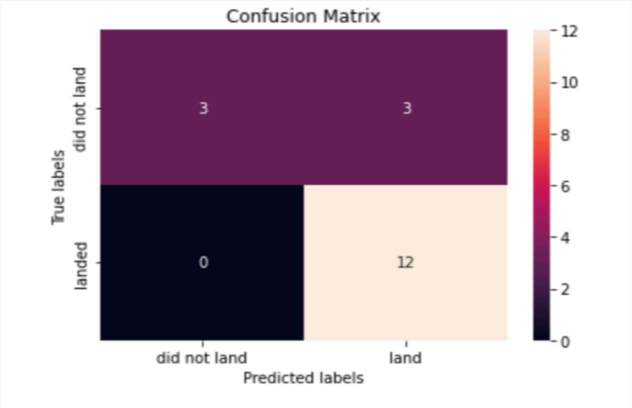
https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/Interactive%20Dashboard%20with%20Ploty%20Dash.ipynb

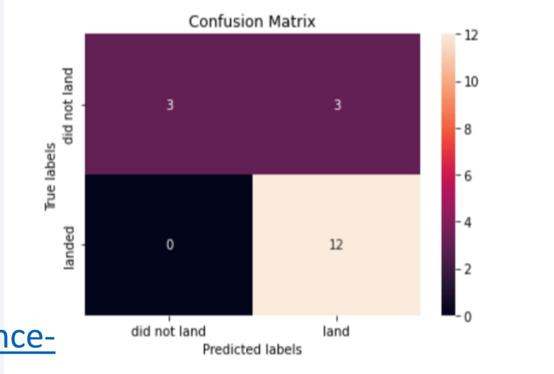
Predictive Analysis (Classification)

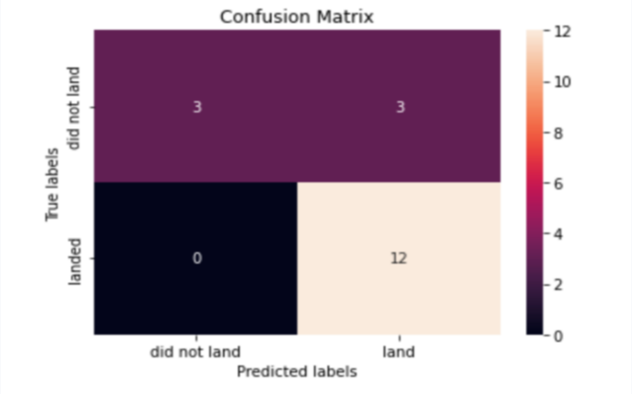
- The SVM, KNN, and Logistic Regression model achieved the highest accuracy at 83.33%, the SVM performs the best in terms of area under the curve











https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/Machine%20Learning%20Prediction.ipynb

Results

- SVM, KNN and Logistic Regression, Tree models are best in terms of prediction accuracy for this dataset
- Lower weighted payloads is performs better then the heavier payloads
- KSC LC 39A had the most successful launches from all
- Orbit GEO, HEO,SSO,ES L1 has the best success Rate

Section 2

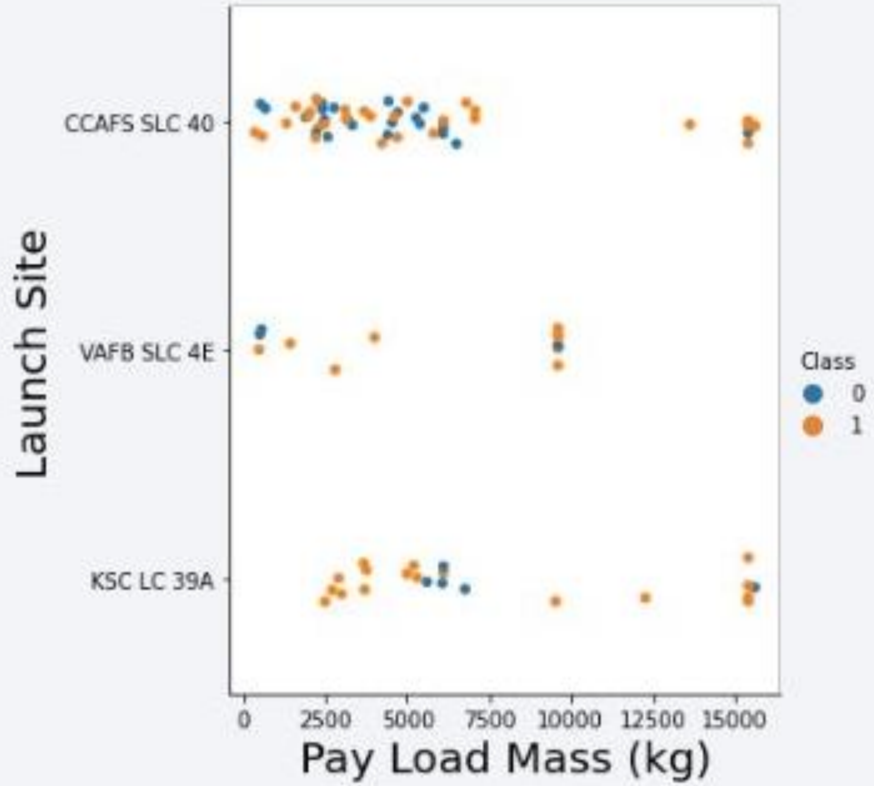
Insights drawn from EDA

Flight Number vs. Launch Site

- Launches from the site of CCAFS SLC 40 are significantly higher than launches form other sites.

https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/EDA%20with%20Data%20Visualization.ipynb

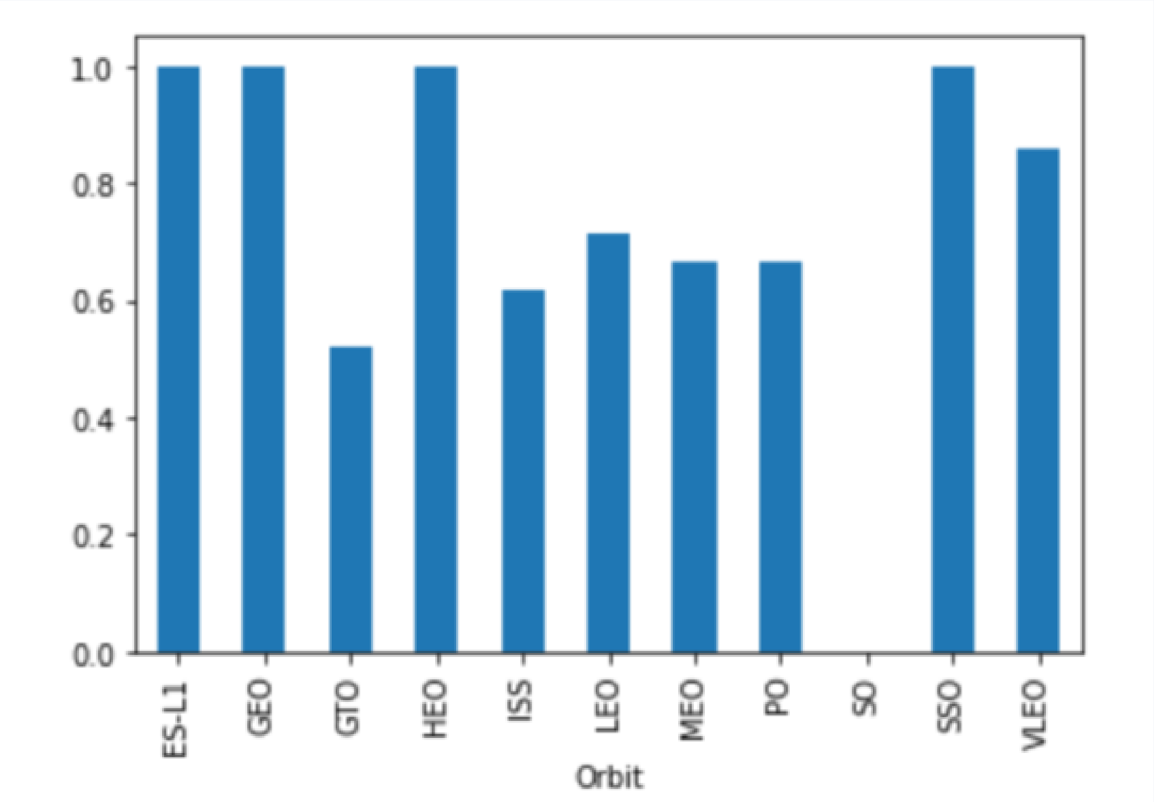
Payload vs. Launch Site



- The majority of IPay Loads with lower Mass have been launched from CCAFS SLC 40.

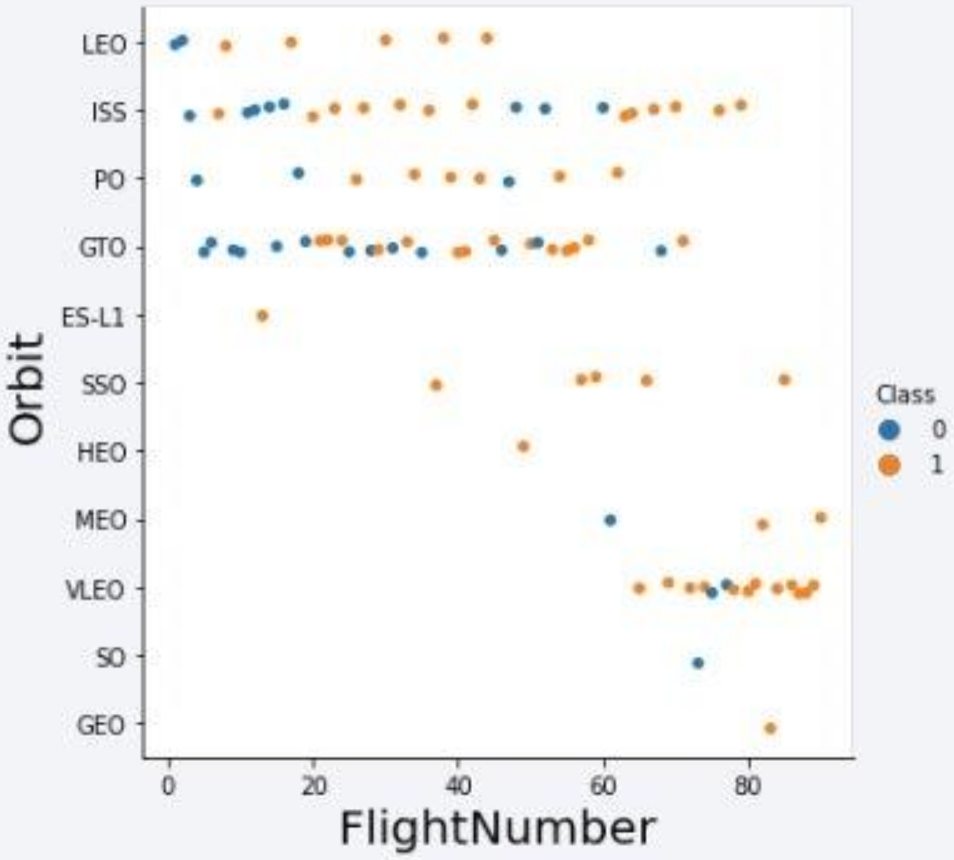
https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/EDA%20with%20Data%20Visualization.ipynb

Success Rate vs. Orbit Type



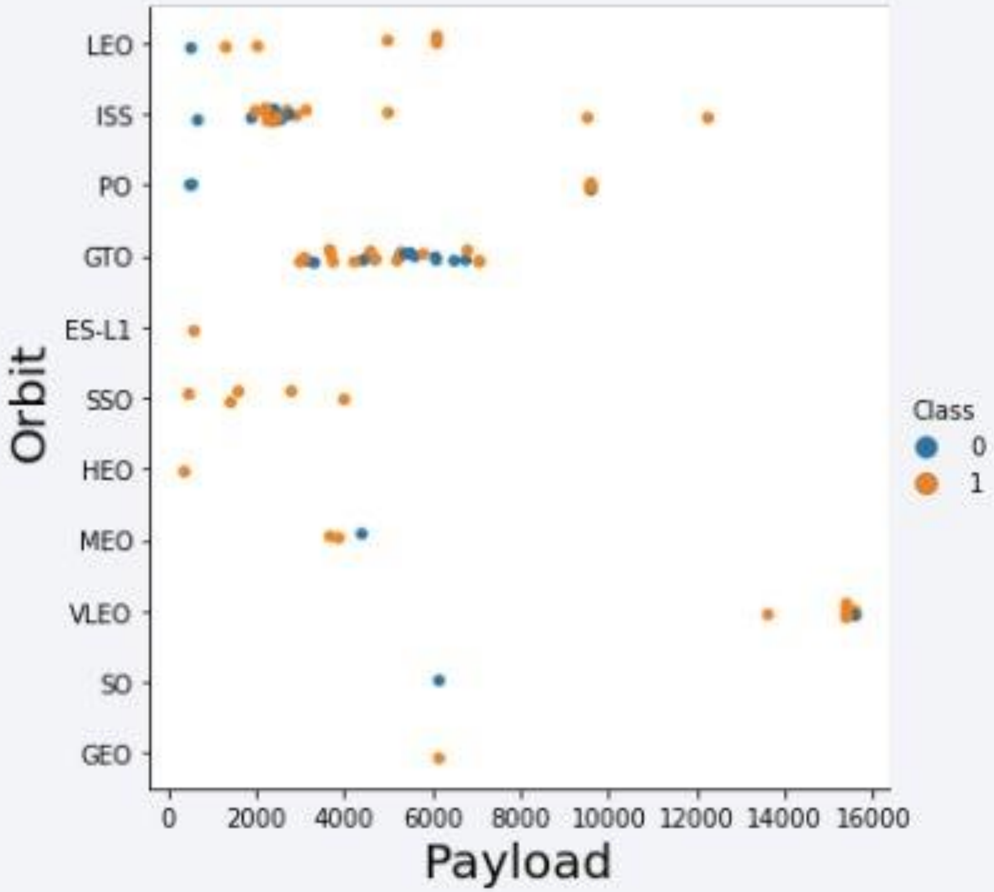
- The Orbit types of ES-L1, GEO, HEO, SSO are among the highest success rate.

Flight Number vs. Orbit Type



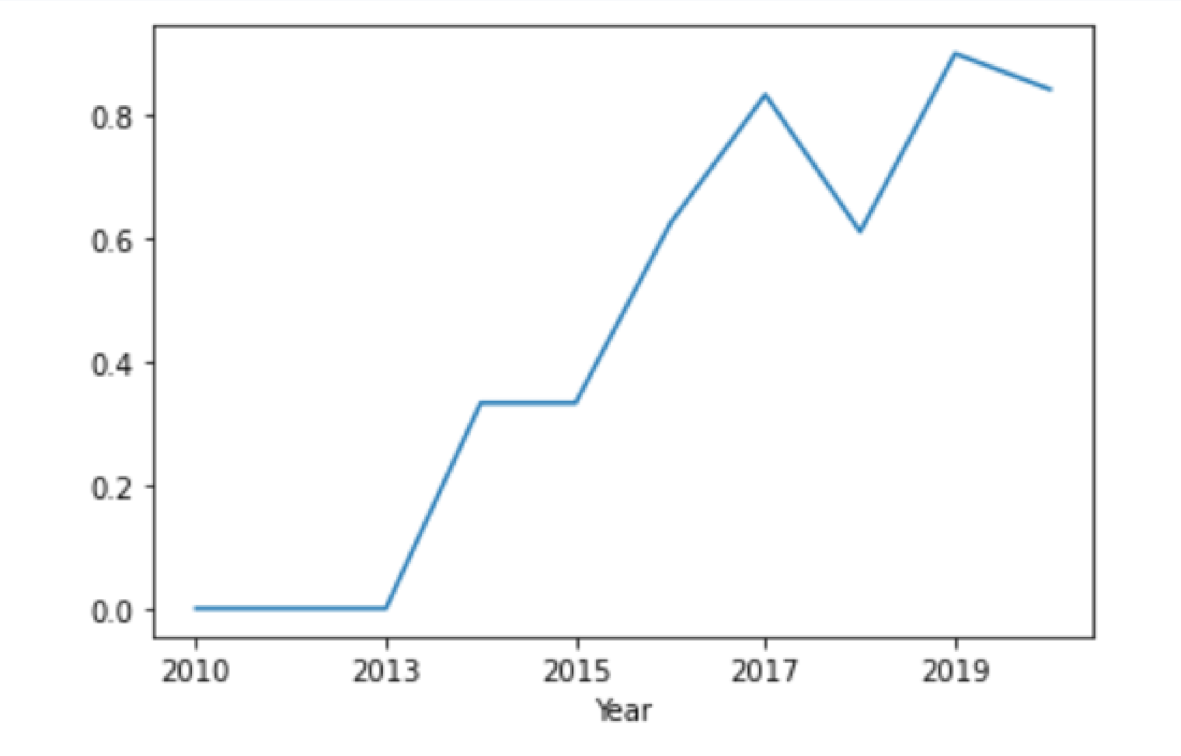
- A trend can be observed of shifting to VLEO launches in recent years.

Payload vs. Orbit Type



- There are strong correlation between ISS and Payload at the range around 2000, as well as between GTO and the range of 4000-8000.

Launch Success Yearly Trend



- Launch success rate has increased significantly since 2013 and has stablised since 2019, potentially due to advance in technology and lessons learned.

All Launch Site Names

- sql SELECT DISTINCT LAUNCH_SITE FROM SPACEX ORDER BY 1;

Out[37]:

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

https://github.com/kyrichandrakanth/IBM-coursera-data-science-capstone-project-on-SpaceX/blob/master/EDA%20with%20SQL.ipynb

Launch Site Names Begin with 'CCA'

- sql SELECT * FROM SPACEX WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;

Out[38]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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)
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-12	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES	Success	No attempt

Total Payload Mass

- sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SPACEX WHERE PAYLOAD LIKE '%CRS%’;

Out[40]:

total_payload
56479

Average Payload Mass by F9 v1.1

- sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD FROM SPACEX WHERE BOOSTER_VERSION = 'F9 v1.1’;

Out[41]:

avg_payload
3676

First Successful Ground Landing Date

- sql SELECT MIN(DATE) AS FIRST_SUCCESS_GP FROM SPACEX WHERE LANDING__OUTCOME = 'Success (ground pad)’;

Out[42]:

first_success_gp
2017-01-05

Successful Drone Ship Landing with Payload between 4000 and 6000

sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEX WHERE PAYLOAD_MASS__KG_
BETWEEN 4000 AND 6000 AND LANDING__OUTCOME = 'Success (drone ship)';

Out[44]:

booster_version
F9 FT B1031.2
F9 FT B1022

Total Number of Successful and Failure Mission Outcomes

- sql SELECT MISSION_OUTCOME, COUNT(*) AS QTY FROM SPACEX GROUP BY MISSION_OUTCOME ORDER BY MISSION_OUTCOME;

Out[46]:

mission_outcome	qty
Success	44
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEX WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEX) ORDER BY BOOSTER_VERSION;

Out[47]:

booster_version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1058.3
F9 B5 B1060.2

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEX WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND DATE_PART('YEAR', DATE) = 2015;

]:

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40

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

- sql SELECT LANDING__OUTCOME, COUNT(*) AS QTY FROM SPACEX WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING__OUTCOME ORDER BY QTY DESC;

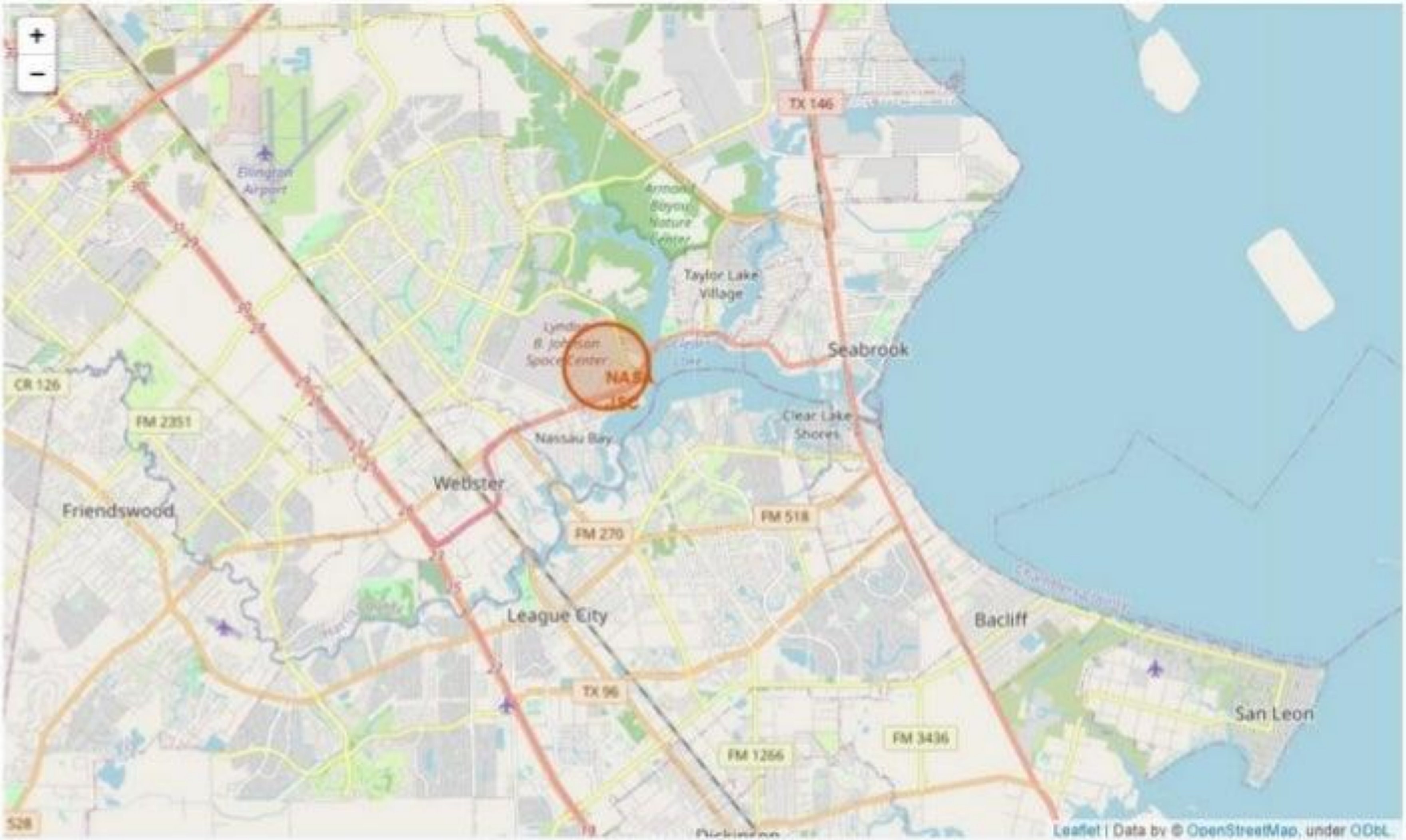
Out[50]:

landing_outcome	qty
No attempt	7
Failure (drone ship)	2
Success (drone ship)	2
Success (ground pad)	2
Controlled (ocean)	1
Failure (parachute)	1

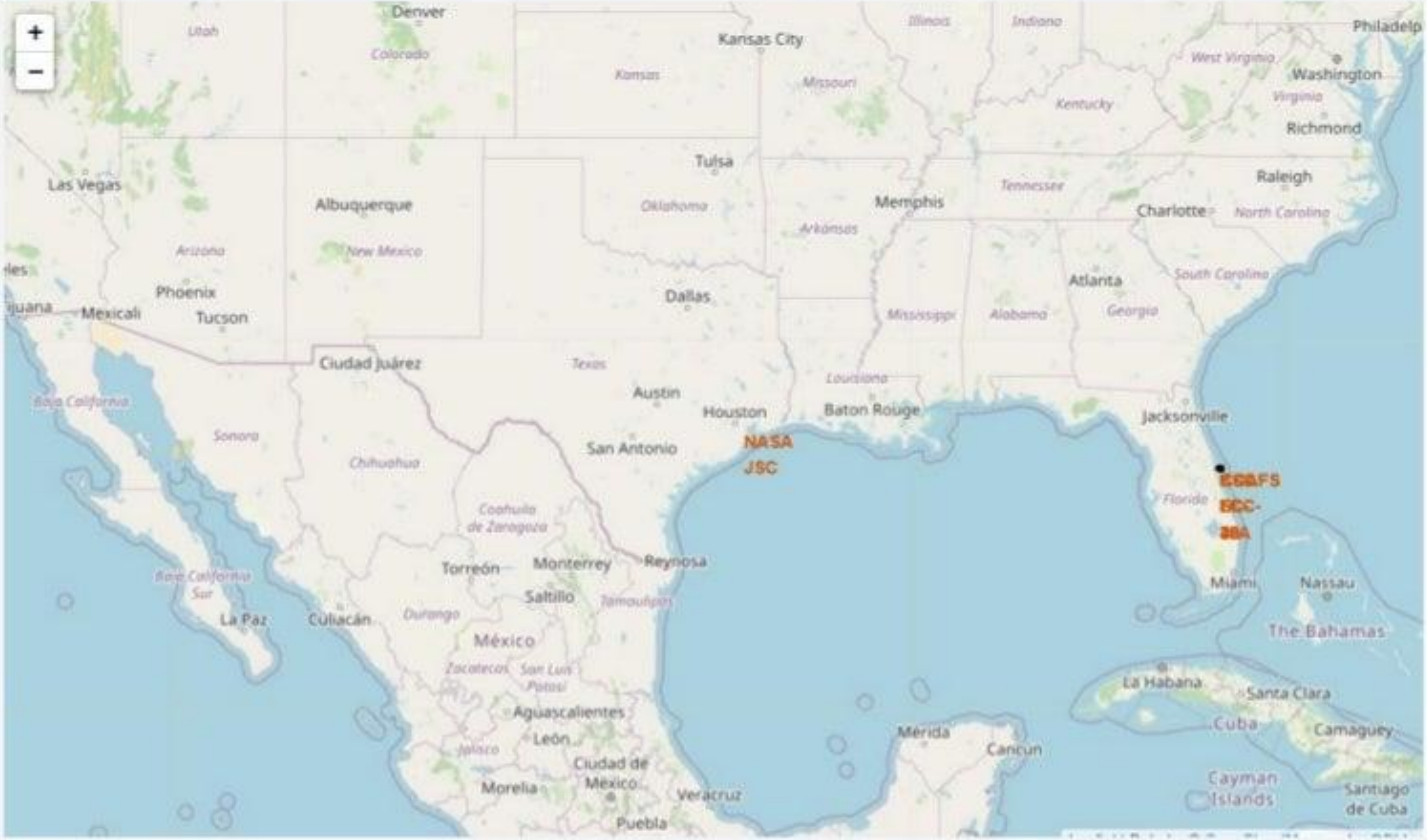
Section 3

Launch Sites Proximities Analysis

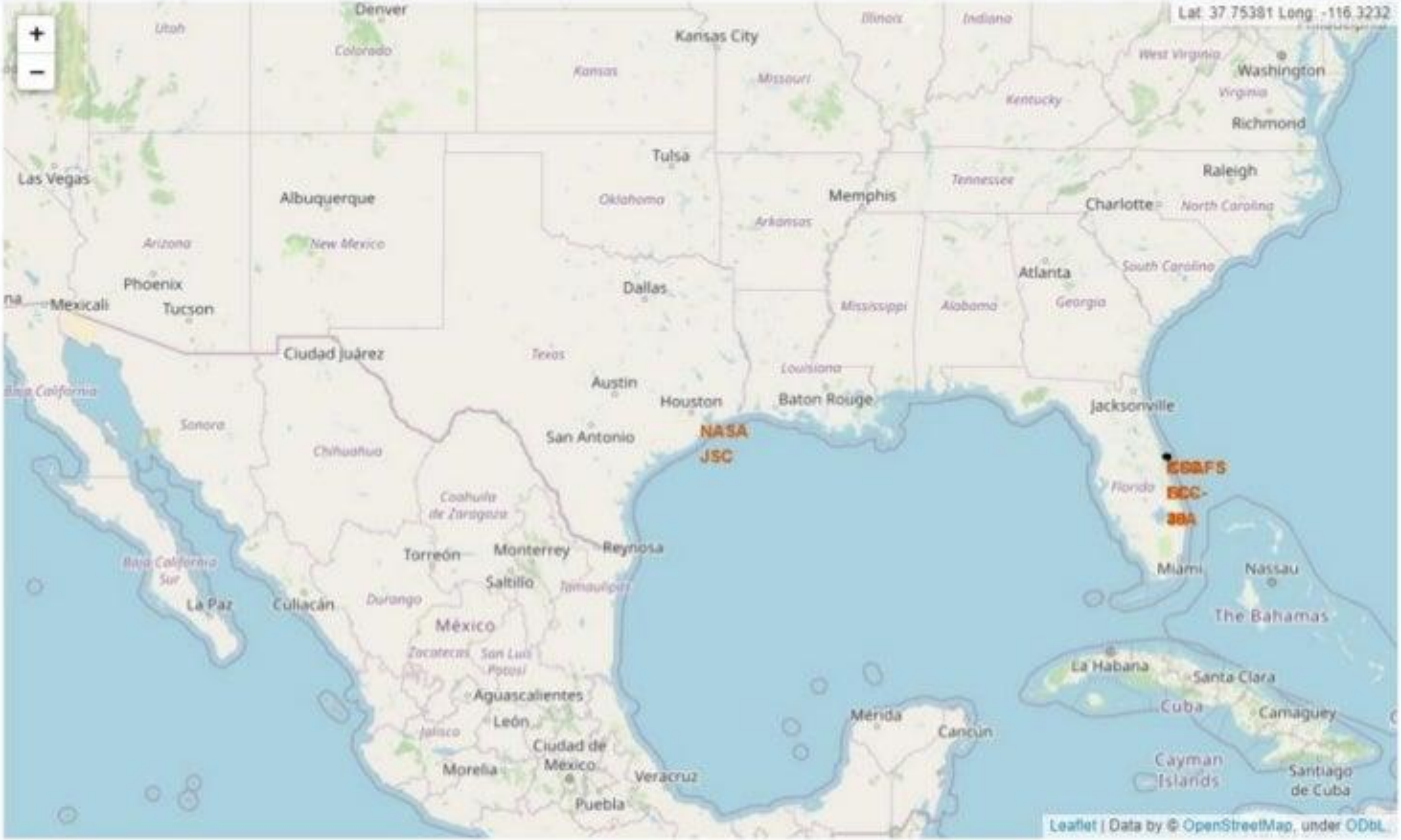
All launch sites marked on a map



Success/failed launches marked on the map



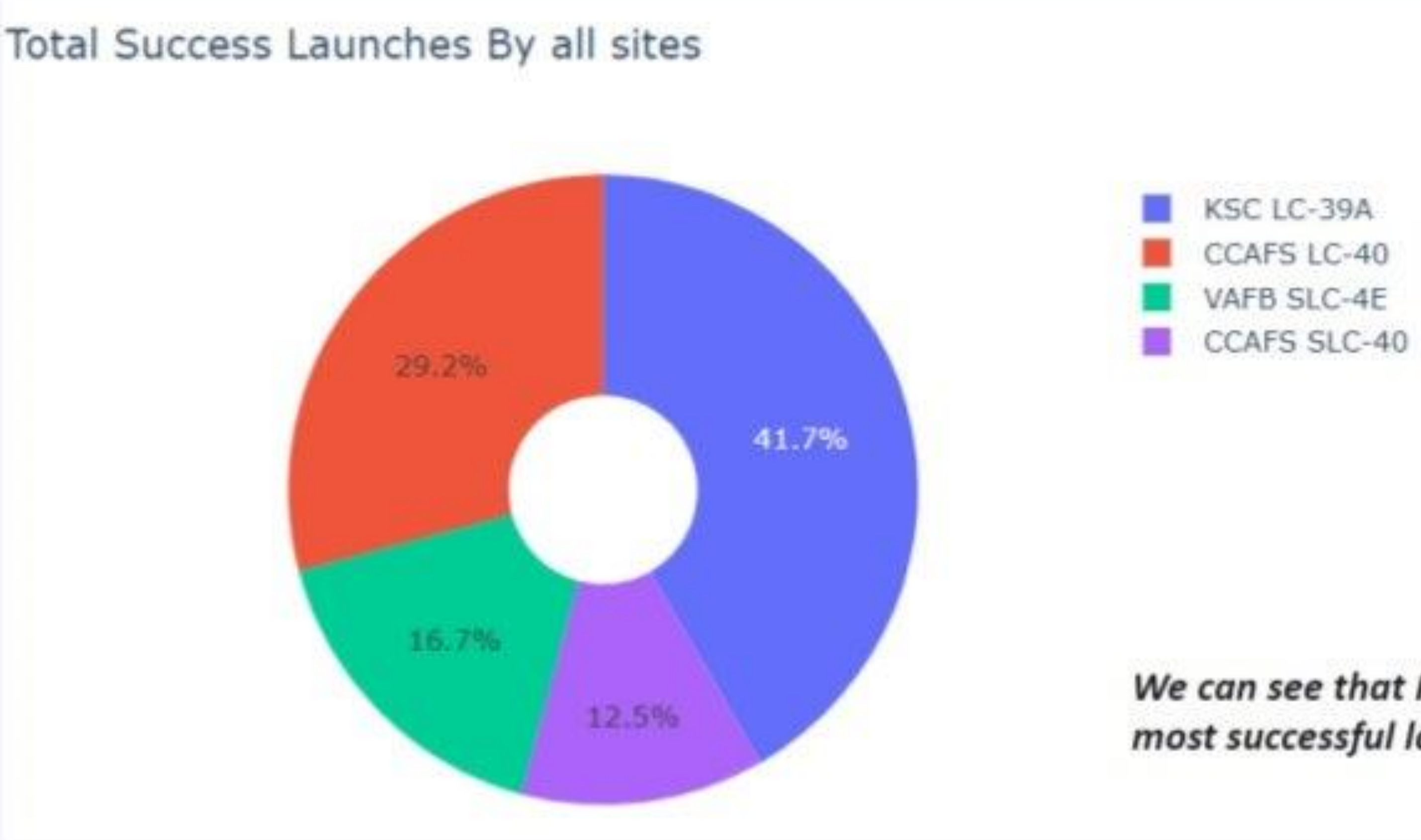
Distances between a launch site to its proximities



Section 4

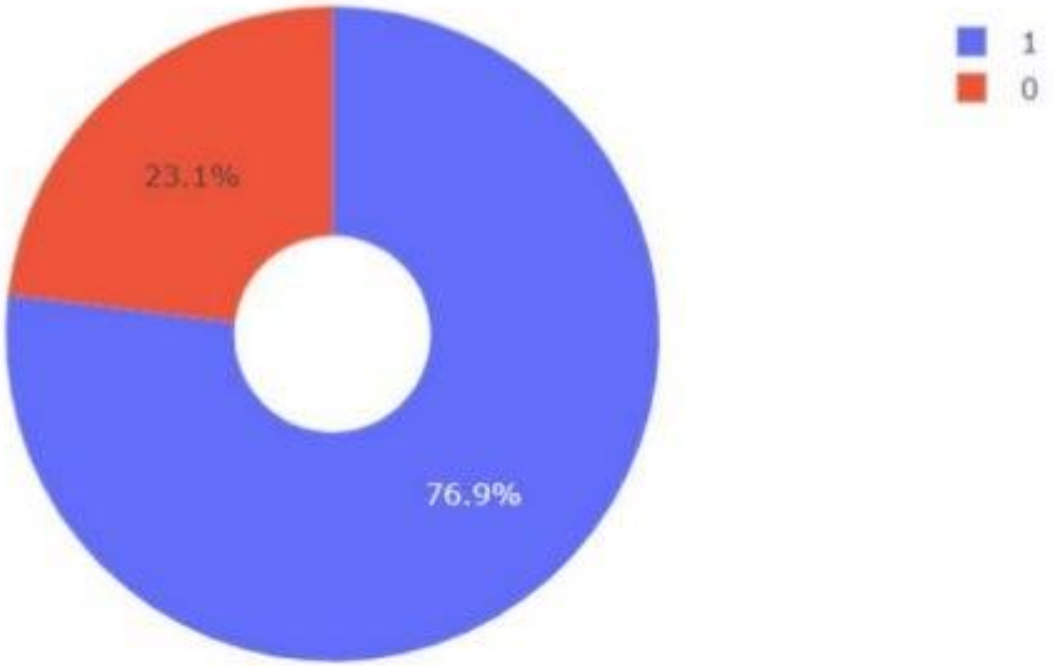
Build a Dashboard with Plotly Dash

Total success launches by all sites



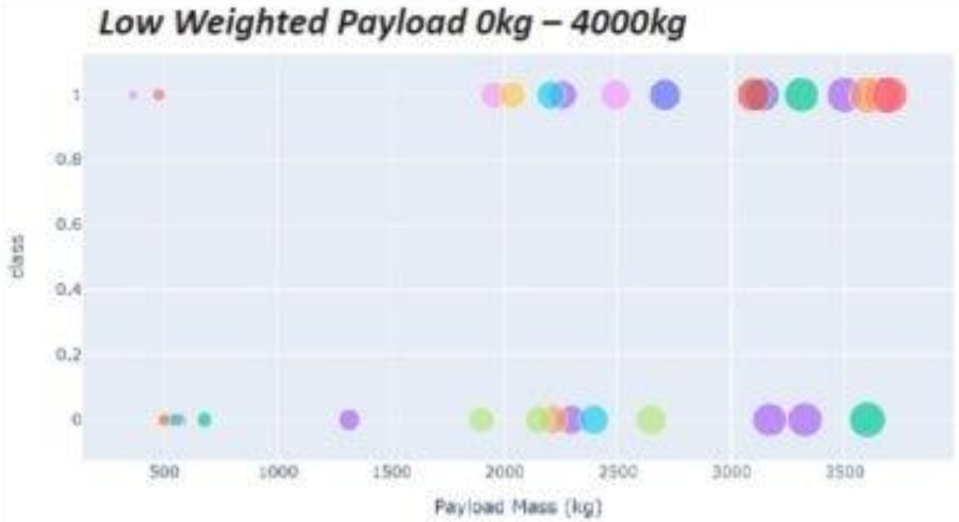
We can see that KSC LC-39A had the most successful launches from all the sites

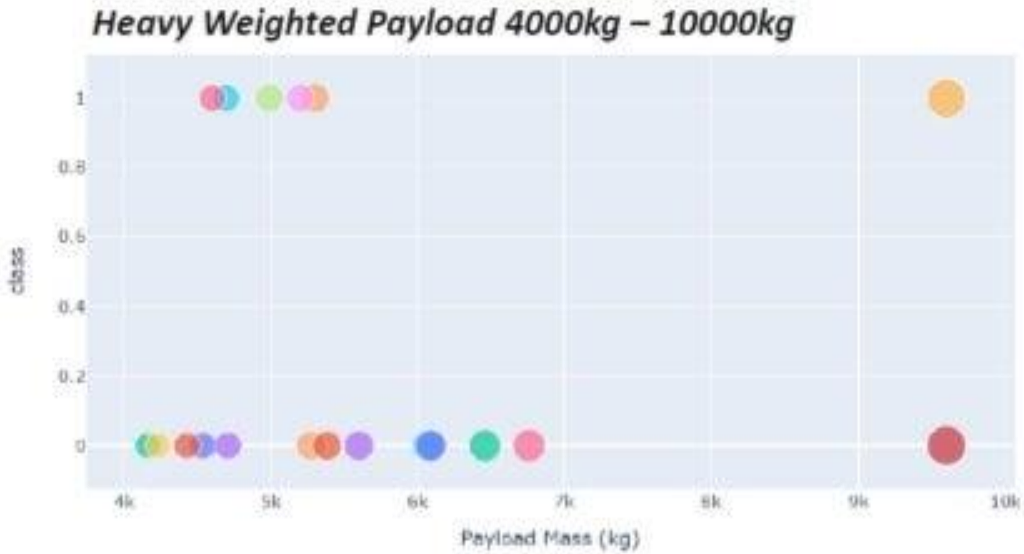
Success rate by site



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

Payload vs launch outcome



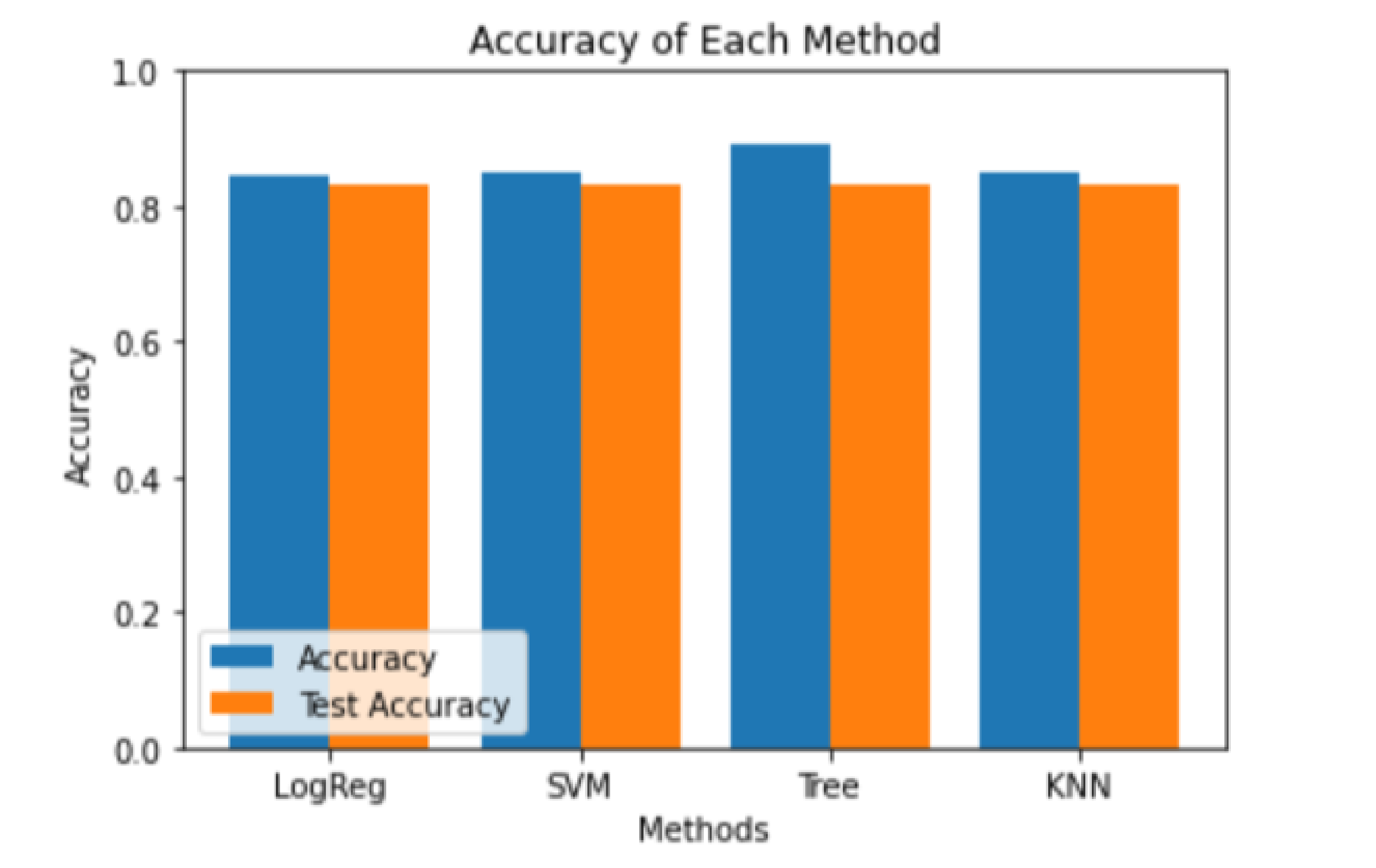


We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

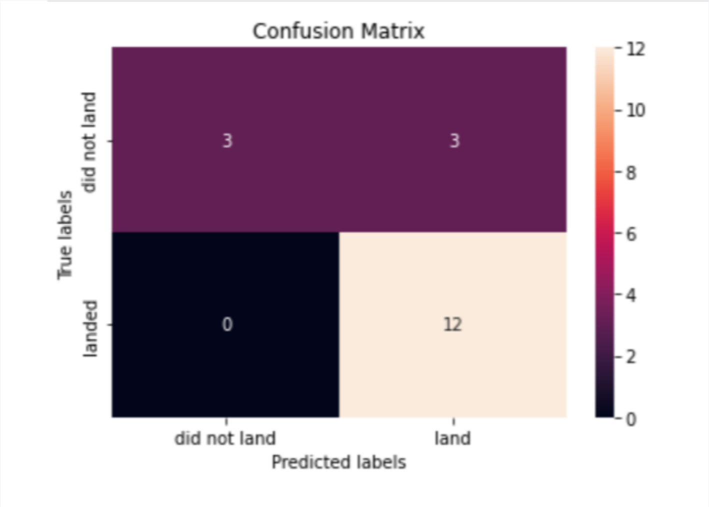
Section 5

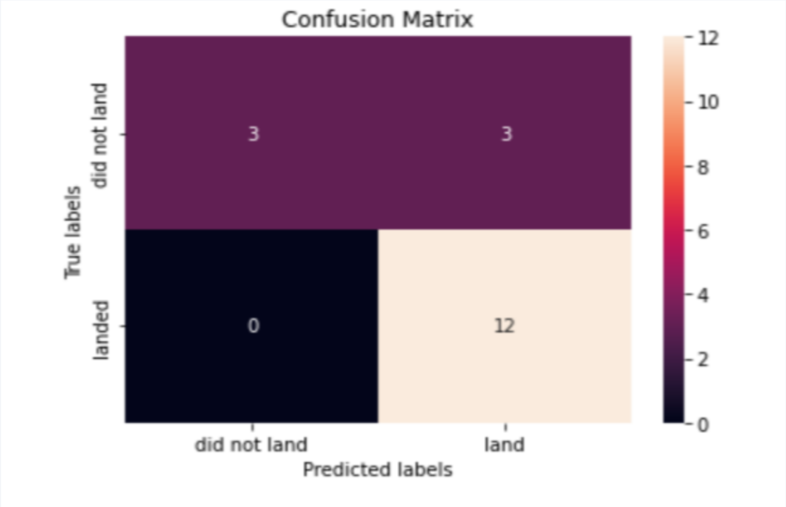
Predictive Analysis (Classification)

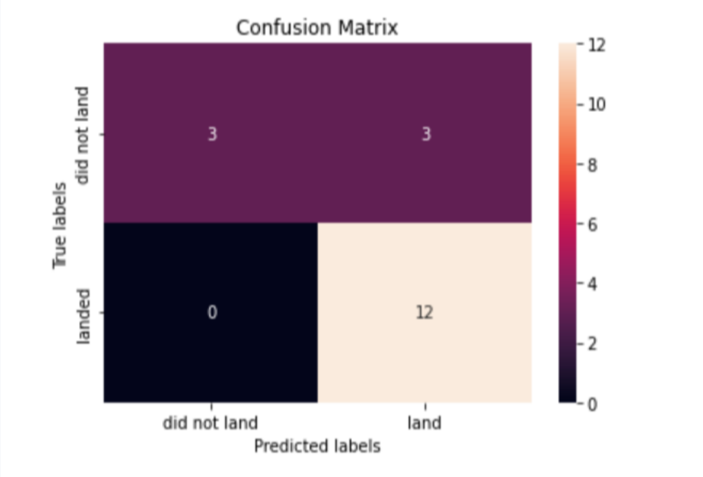
Classification Accuracy

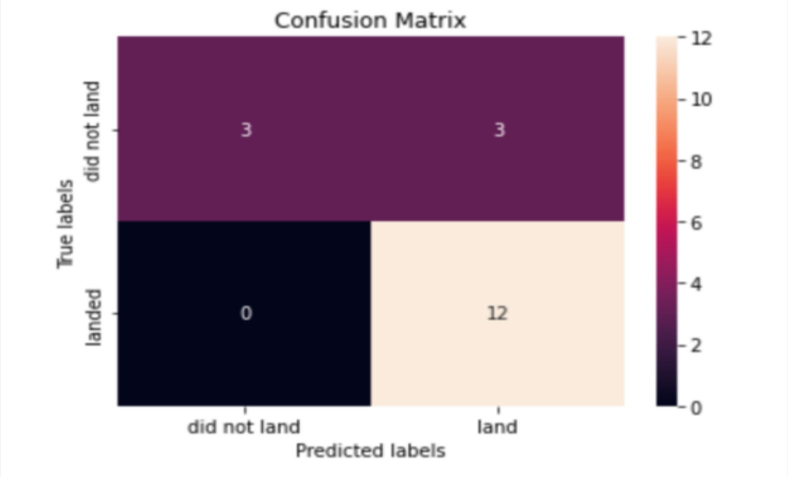


Confusion Matrix









Conclusions

- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches.
- KSC LC 39A had the most successful launches from all the sites.
- Orbit GEO,HEO,SSO,ES L1 has the best Success Rate.
- The SVM, KNN, and Logistic Regression models are the best in terms of prediction accuracy for this dataset.
- Low weighted payloads perform better than the heavier payloads.

Thank you!