IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Archana Kumari

12/03/2022

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 (Classification)
- Summary of all results
  - EDA results
  - Interactive analytics
  - Predictive analysis

Introduction

- Project background and context
  - The SpaceX Falcon 9 rocket with the Dragon cargo module lifted off from the Florida Space Launch Complex 40 at Cape Canaveral Air Force Station on May 4. The Commercial Resupply Services mission (CRS-17) launched over 5,550 pounds of NASA cargo into orbit and successfully delivered it to the International Space Station (ISS) on May 6.
  - We predicted if the Falcon 9 first stage will land successfully. 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. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.
- Common problems that needed solving
  - What influences if the rocket will land successfully?
  - The effect each relationship with certain rocket variables will impact in determining the success rate of a successful landing.
  - What conditions does SpaceX have to achieve to get the best results and ensure the best rocket success landing rate.
  - The project task is to predicting if the first stage of the SpaceX Falcon 9 rocket will land successfully.

Section 1

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API
  - Web scraping from Wikipedia
- Perform data wrangling(Transforming data for Machine Learning)
  - One Hot encoding data fields from Machine Learning and data cleaning of null values and irrelevant columns.
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Plotting: Scatter Graphs, Bar Graphs to show relationships between variables to show patterns of data.
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Logistic regression ,KNN,SVM,DT models have been built and evaluated for the best classifier.

Data Collection – SpaceX API

1. Collecting the data with API call

```python
# Use json_normalize meethod to convert the json re
data=pd.json_normalize(response.json())
```

2.Convert the JSON result into a dataframe

```python
# Use json_normalize meethod to convert the json re
data=pd.json_normalize(response.json())
```

3. Updating columns and rows (pe-processing)

```python
# Lets take a subset of our dataframe keeping only the features we want and the flight number, an
data = data[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]

# We will remove rows with multiple cores because those are falcon rockets with 2 extra rocket boo
data = data[data['cores'].map(len)==1]
data = data[data['payloads'].map(len)==1]

# Since payloads and cores are lists of size 1 we will also extract the single value in the list a
data['cores'] = data['cores'].map(lambda x : x[0])
data['payloads'] = data['payloads'].map(lambda x : x[0])

# We also want to convert the date_utc to a datetime datatype and then extracting the date leaving
data['date'] = pd.to_datetime(data['date_utc']).dt.date

# Using the date we will restrict the dates of the launches
data = data[data['date'] <= datetime.date(2020, 11, 13)]
```

4. **Filter the dataframe to only include `Falcon 9` launches**

```python
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = df[df.BoosterVersion == 'Falcon 9']
data_falcon9
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reuse
4	6	2010-06-04	Falcon 9	5919.165341	LEO	CCSFS SLC 40	None None	1	False	Fals
5	8	2012-05-22	Falcon 9	525.000000	LEO	CCSFS SLC 40	None None	1	False	Fals
6	10	2013-03-01	Falcon 9	677.000000	ISS	CCSFS SLC 40	None None	1	False	Fals

vert the data to csv file with name

'dataset_part_1.csv

```python
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

GitHub repo: https://github.com/riya0395/testproject/commit/5a5f610b0ffa92aac105ac51b24426bd2c483d56

Data Collection - Scraping

Web scraping from Wikipedia

GitHub repo: https://github.com/riya0395/testproject/blob/master/web%20scraping./aplied%20capstone.ipynb

1.Getting Response from HTML

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

2. Creating BeautifulSoup Object

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

3. Finding tables

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

4.Getting column names

```
column_names = []
for row in first_launch_table.find_all('th'):
    name = extract_column_from_header(row)
    if (name != None and len(name) > 0):
        column_names.append(name)
```

5. Creation of dictionary

```
launch_dict= dict.fromkeys(column_names)
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

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

7. Converting dictionary to dataframe

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

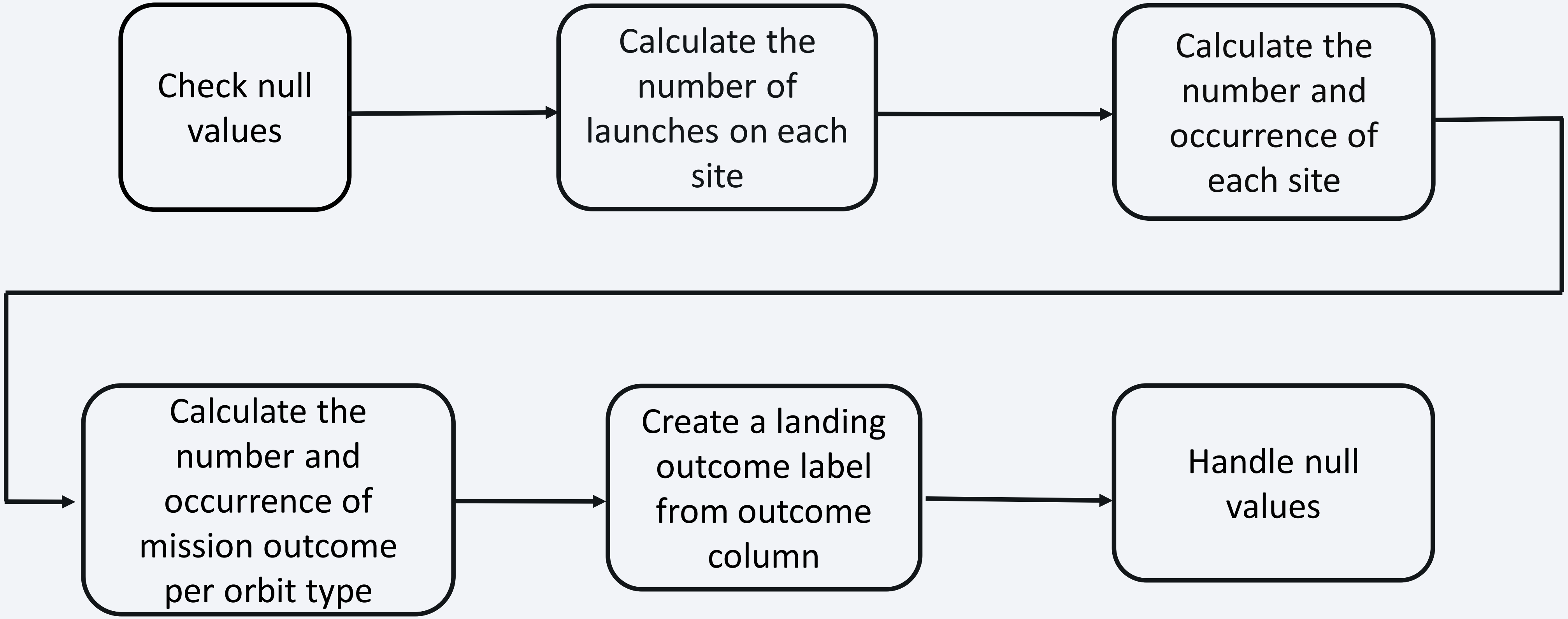
8. Dataframe to .CSV

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

Data Wrangling

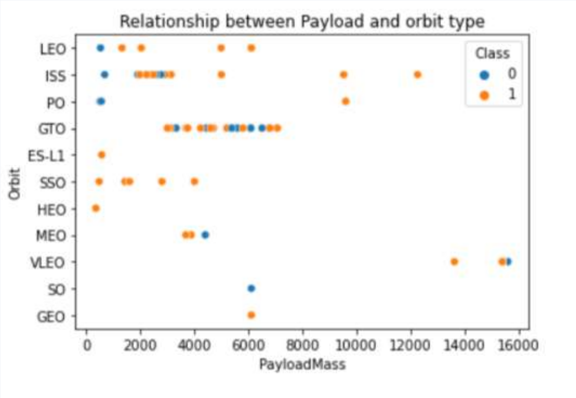
GitHub repo: https://github.com/riya0395/testproject/blob/master/EDA./aplied%20capstone.ipynb

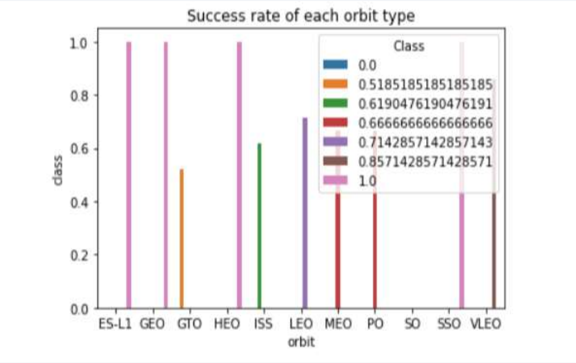
EDA Analysis

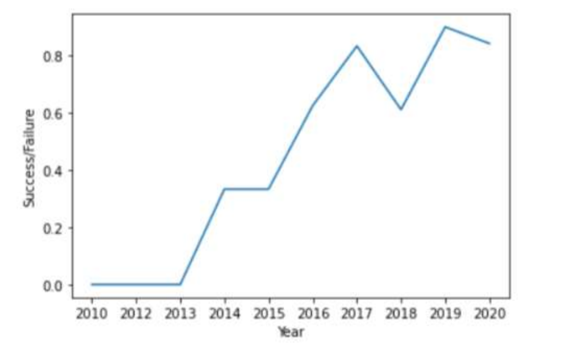


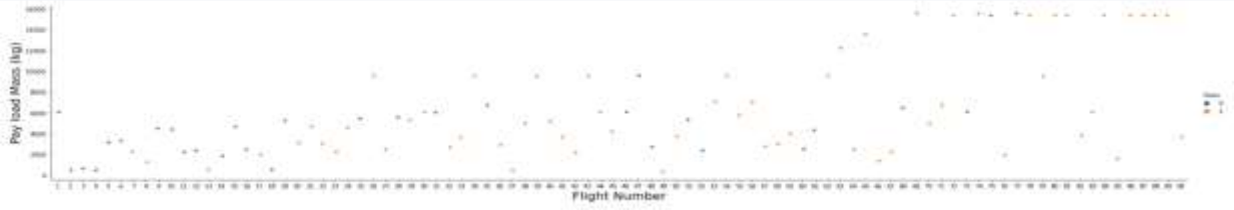
EDA with Data Visualization

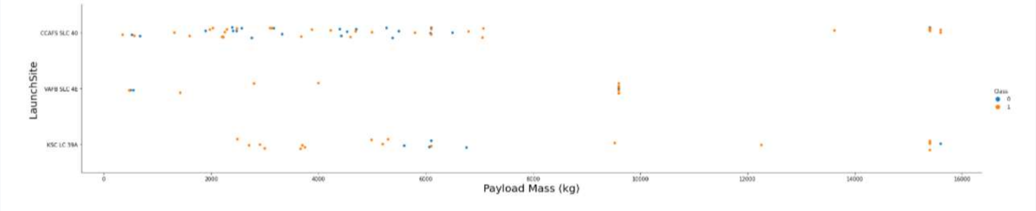
GitHub repo: https://github.com/riya0395/testproject/blob/capstone/aplied%20capstone.ipynb

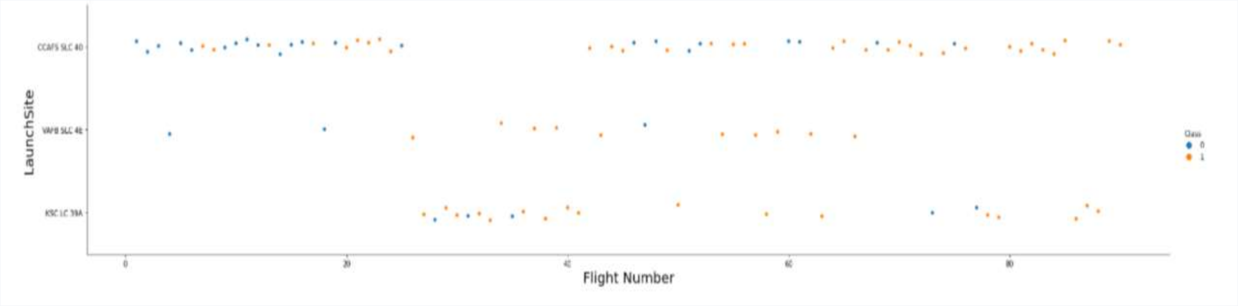


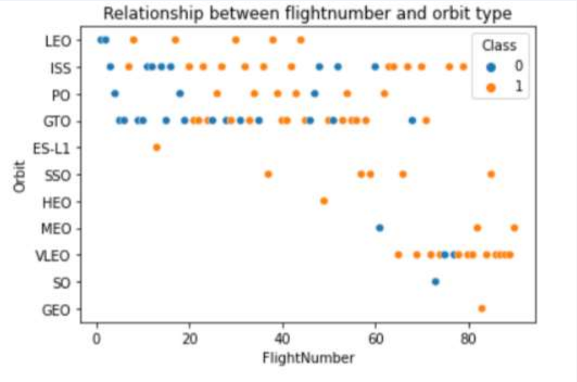










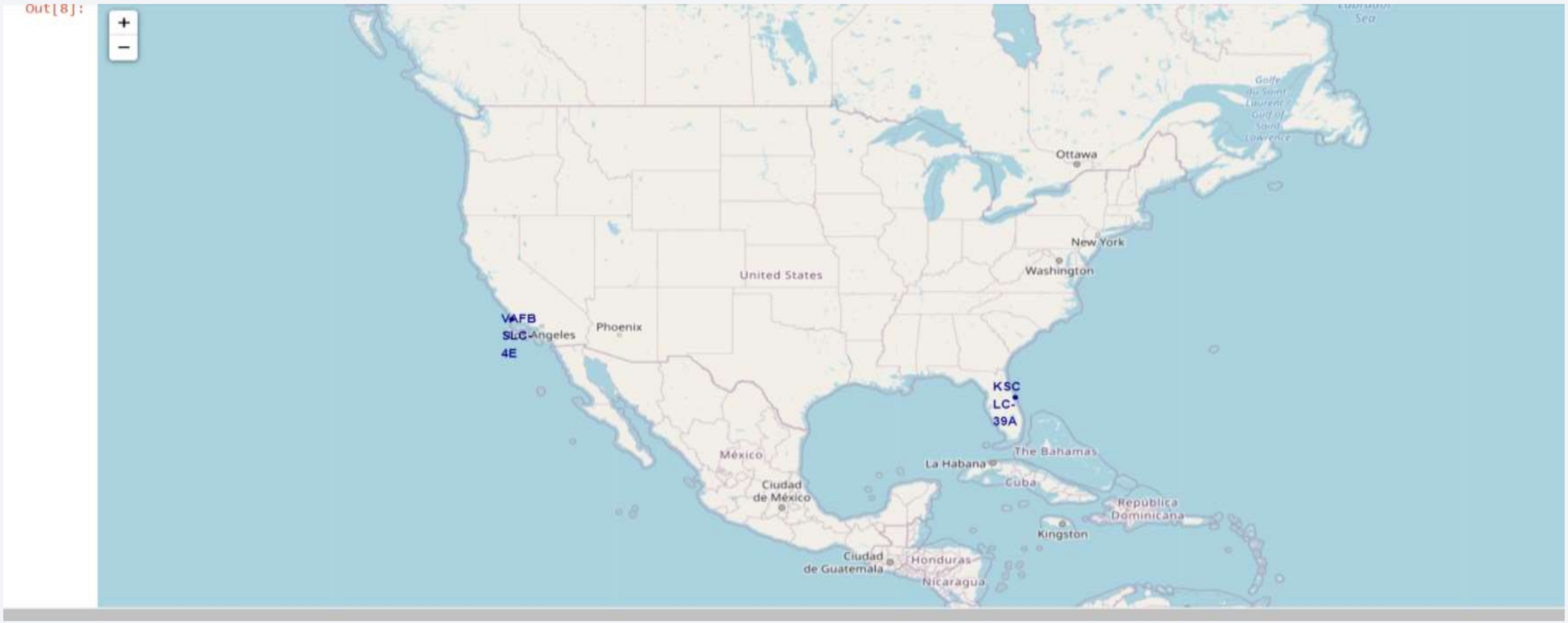


EDA with SQL

GitHub repo: https://github.com/riya0395/testproject/blob/master/EDA%20with%20SQL./aplied%20capstone.ipynb

- SQL queries performed include :
  - Displaying the names of the unique launch sites in the space mission
  - Displaying 5 records where launch sites begins 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 successfully 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, successfully 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 2019-06-04 and 2017-03-20 in descending order.

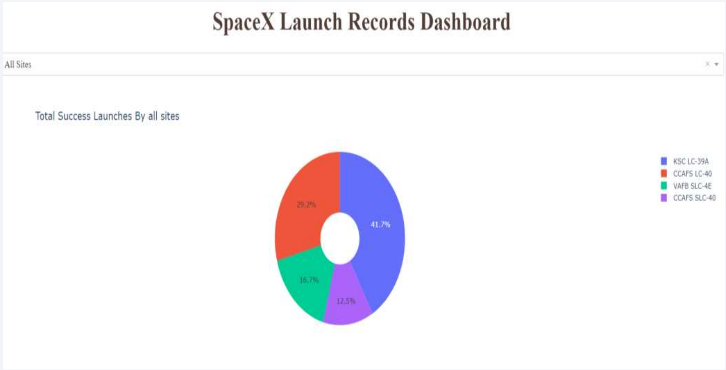
Build an Interactive Map with Folium

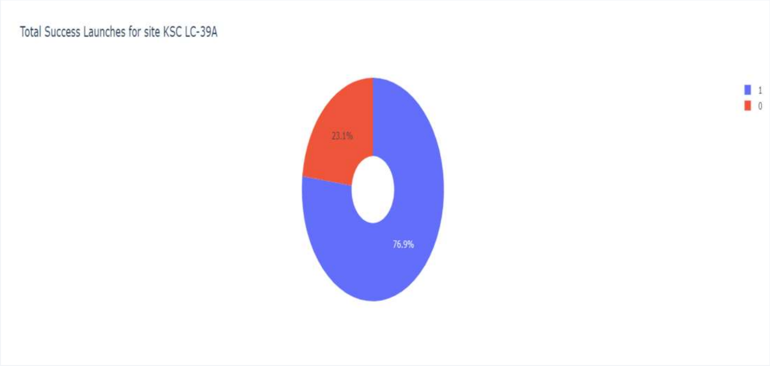


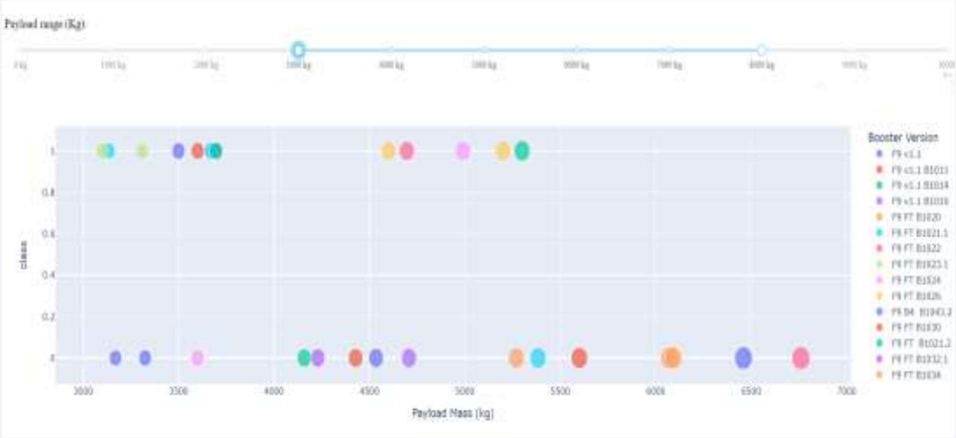
Map marker have been added to the map with aim to finding an optimal location for building a launch site

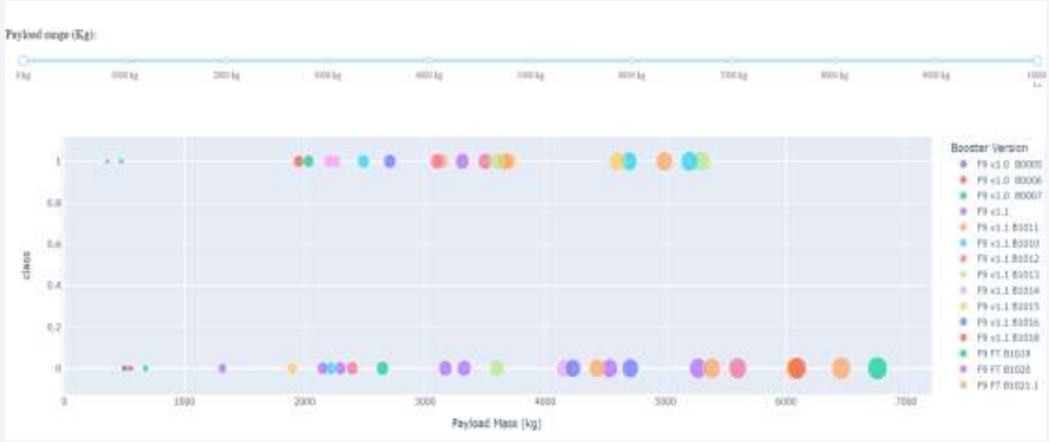
GitHub repo: https://github.com/riya0395/testproject/blob/master/Folium%20map./aplied%20capstone.ipynb

Build a Dashboard with Plotly Dash



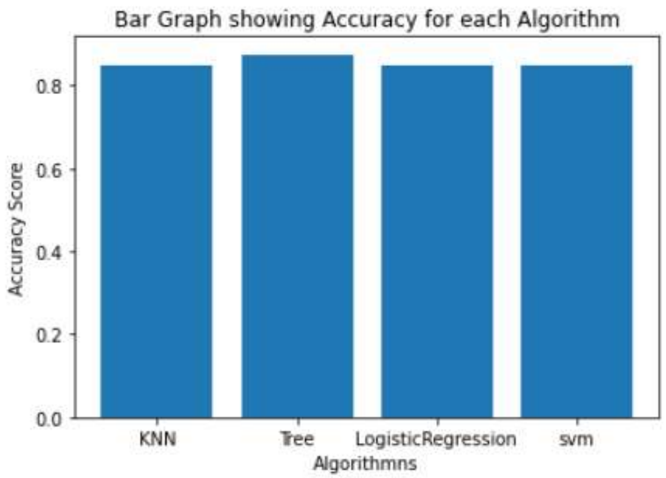


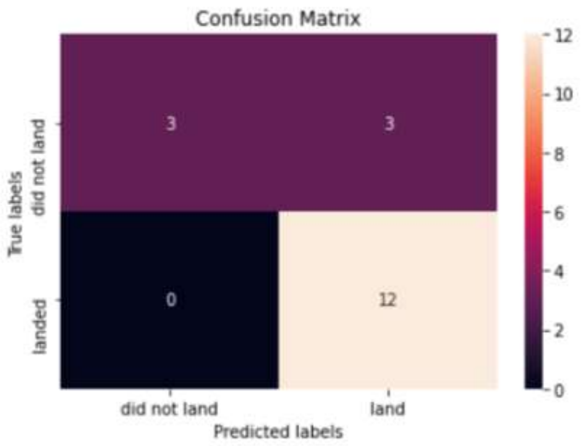




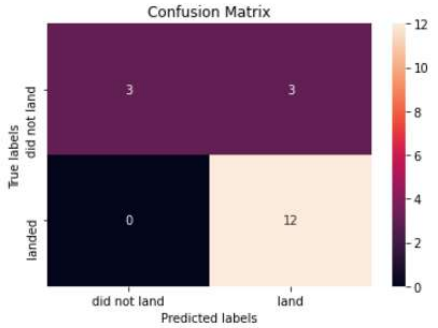
Predictive Analysis (Classification)

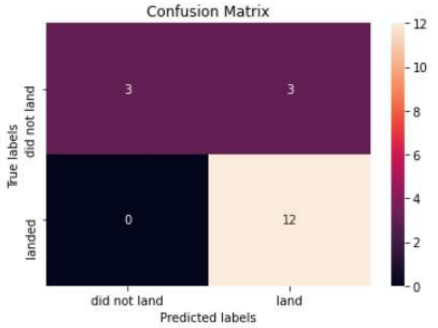
- The SVM, KNN, LR and Tree model achieved the highest accuracy at 83.3% , while the Tree perform the best with a score of 0.87

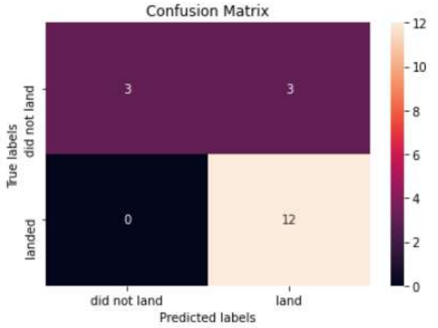




GitHub repo: https://github.com/riya0395/testproject/blob/master/aplied%20capstone.ipynb







Results

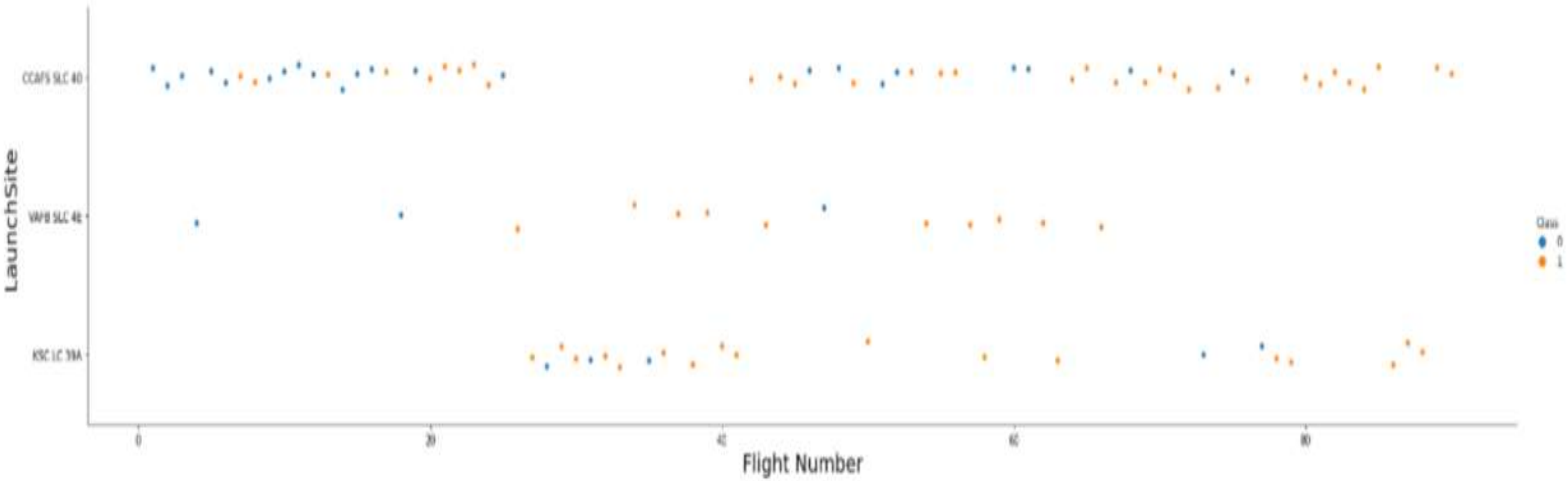
- The SVM, KNN and LR models are the best in terms of prediction accuracy for this dataset.
- Low weighted payloads perform better than the heavier payloads.
- The success rate for SpaceX launches is directly proportional time in years they will eventually perfect the launches.
- KSC LC-39A had the most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

Section 2

Insights drawn from EDA

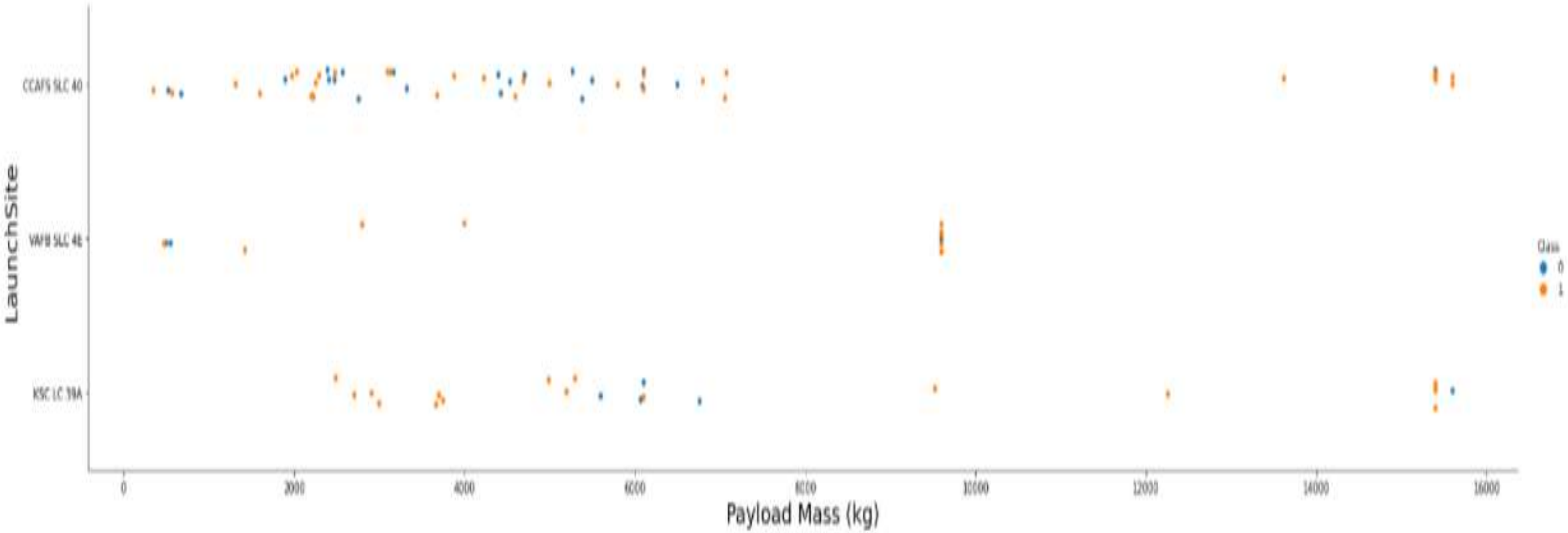
Flight Number vs. Launch Site

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



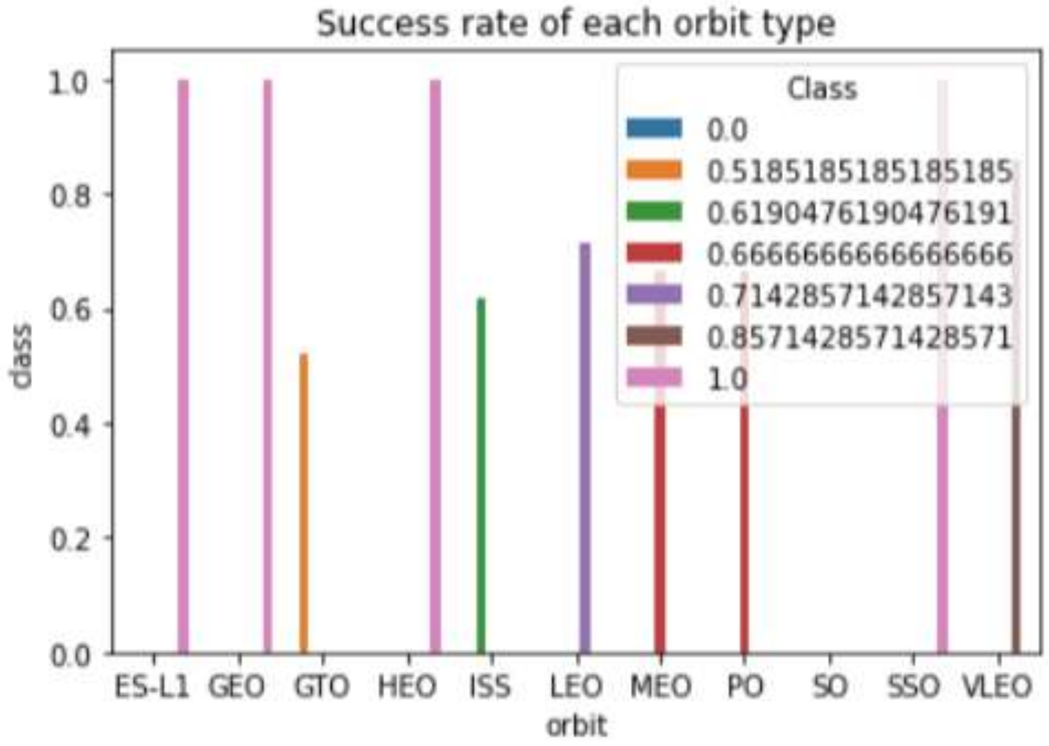
Payload vs. Launch Site

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



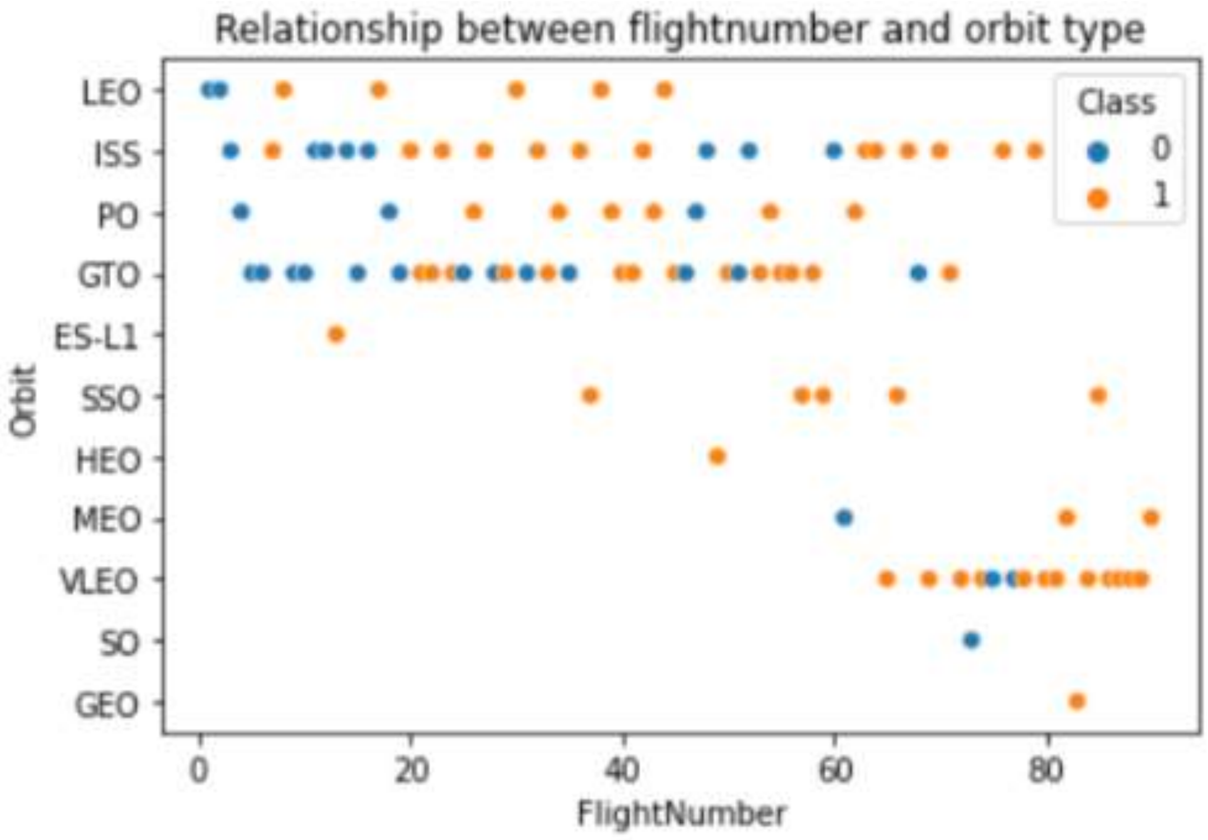
Success Rate vs. Orbit Type

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



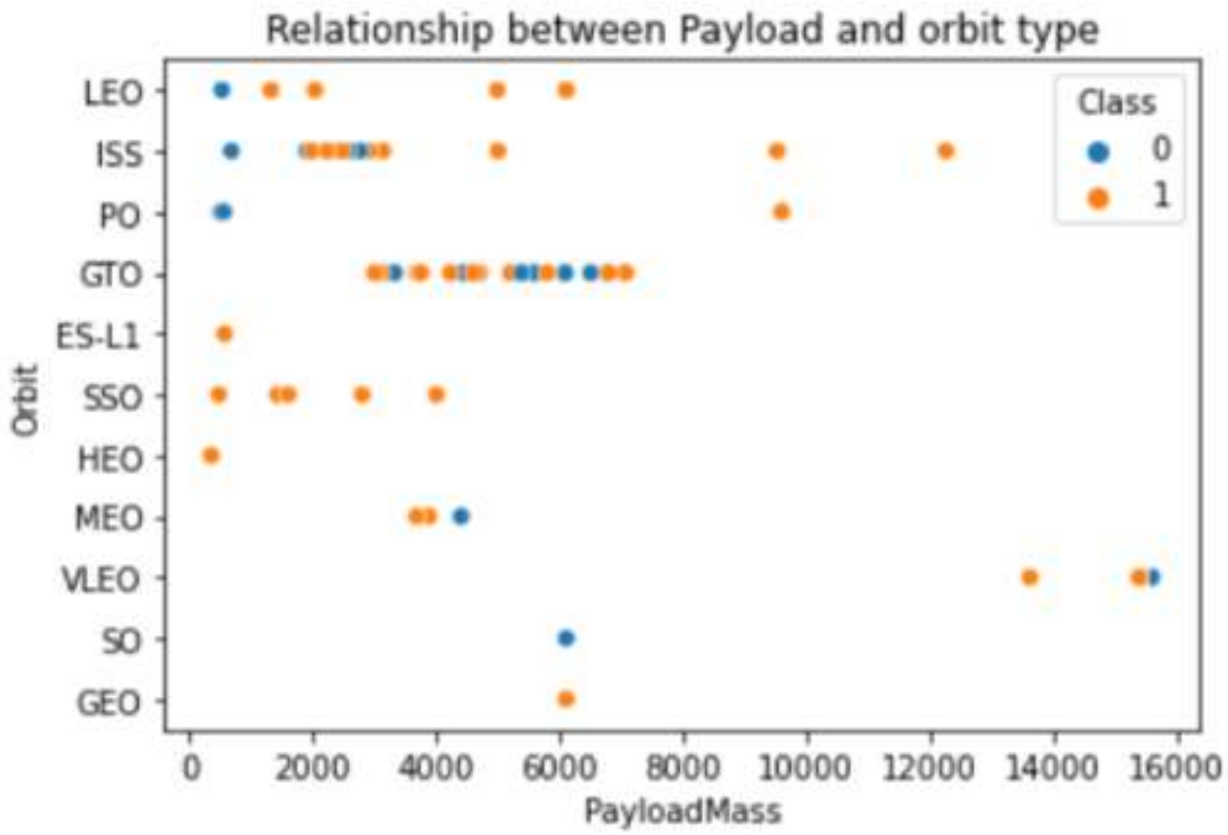
Flight Number vs. Orbit Type

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



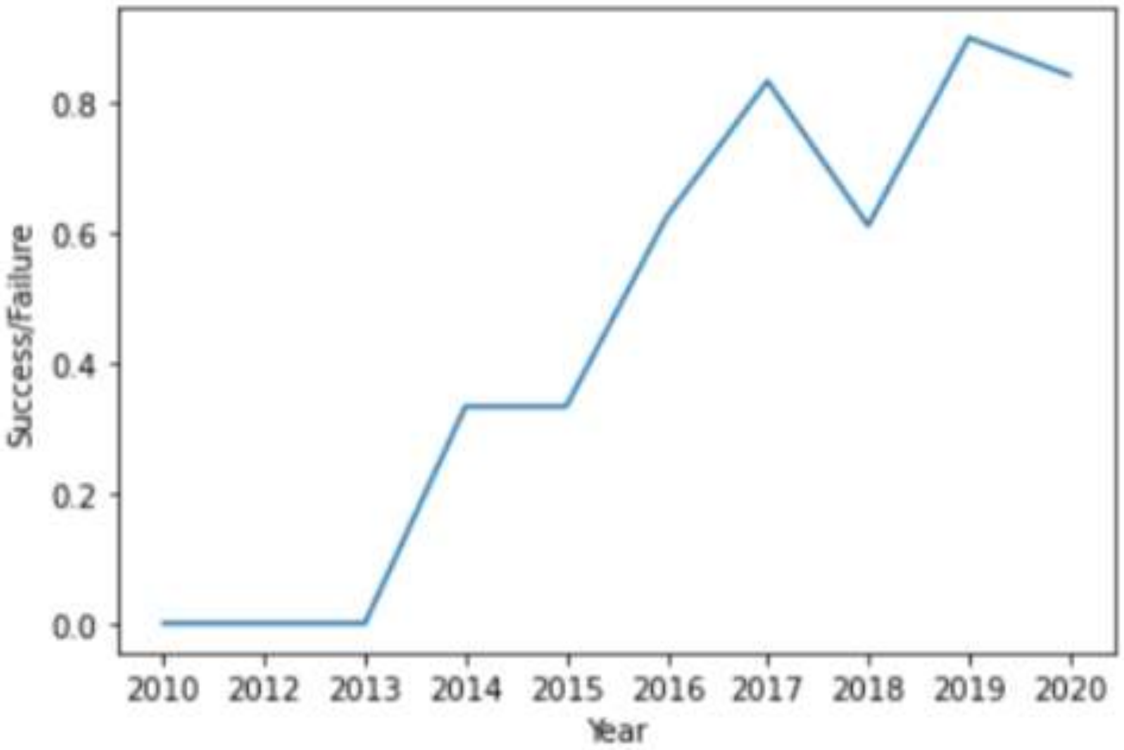
Payload vs. Orbit Type

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



Launch Success Yearly Trend

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



All Launch Site Names

- %%sql
- select unique(LAUNCH_SITE) from Spacex

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

Launch Site Names Begin with 'CCA'

- %%sql
- select LAUNCH_SITE FROM Spacex where (LAUNCH_SITE) like 'CCA%' limit 5

launch_site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Total Payload Mass

payloadmass
22007

%%sql

select sum(PAYLOAD_MASS__KG_) AS Payloadmass from Spacex where CUSTOMER = 'NASA (CRS)'

Average Payload Mass by F9 v1.1

- %%sql
- select AVG(PAYLOAD_MASS__KG_)
  AS Payloadmass from Spacex
  where booster_version = 'F9 v1.1'

payloadmass
3676

First Successful Ground Landing Date

- %%sql
- select DATE from Spacex WHERE LANDING__OUTCOME = 'Success (ground pad)' limit 1

DATE
2017-01-05

Successful Drone Ship Landing with Payload between 4000 and 6000

- %%sql
- select BOOSTER_VERSION FROM Spacex where PAYLOAD_MASS__KG_ between 4000 and 6000

booster_version
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1014
F9 FT B1020
F9 FT B1022
F9 FT B1032.1
F9 B4 B1040.1
F9 FT B1031.2
F9 B4 B1043.1
F9 B4 B1040.2
F9 B5 B1046.2
F9 B5 B1046.3
F9 B5 B1051.2
F9 B5B1062.1

Total Number of Successful and Failure Mission Outcomes

- %%sql
- SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS OUTCOME FROM Spacex group by MISSION_OUTCOME

mission_outcome	outcome
Success	44
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- %%sql
- SELECT BOOSTER_VERSION as boosterversion from Spacex where PAYLOAD_MASS__KG_ = (SELECT max(PAYLOAD_MASS__KG_) as avg_payloadmass FROM Spacex)

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

2015 Launch Records

- %%sql
- SELECT MONTH(DATE) ,LANDING__OUTCOME,BOOSTER_VERSION,LAUNCH_SITE FROM Spacex where EXTRACT(YEAR FROM DATE)=2015

1	landing__outcome	booster_version	launch_site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
11	Controlled (ocean)	F9 v1.1 B1013	CCAFS LC-40
2	No attempt	F9 v1.1 B1014	CCAFS LC-40

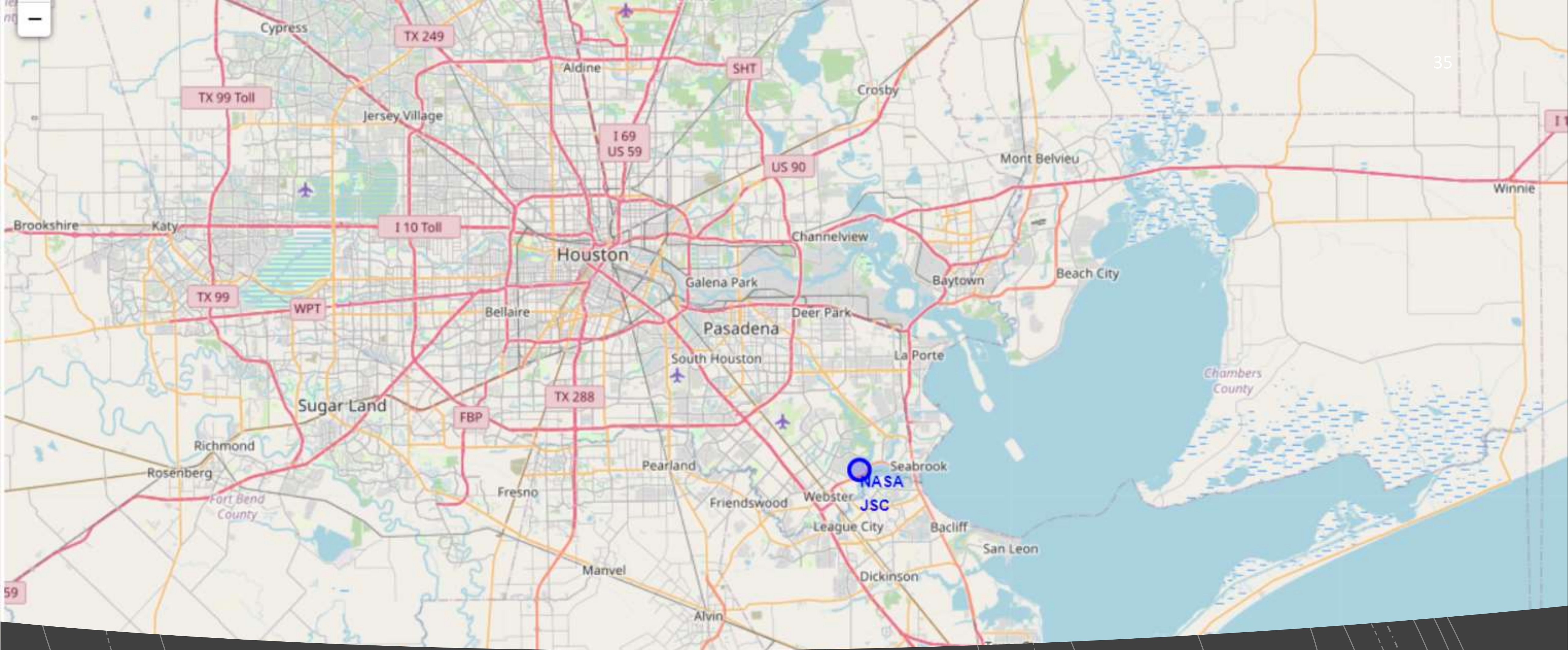
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- %%sql
- SELECT * FROM Spacex WHERE LANDING__OUTCOME='Success (ground pad)' or LANDING__OUTCOME='Failure (drone ship)' AND (DATE BETWEEN '2010-06-04' AND '2017-03-20') ORDER BY DATE DESC

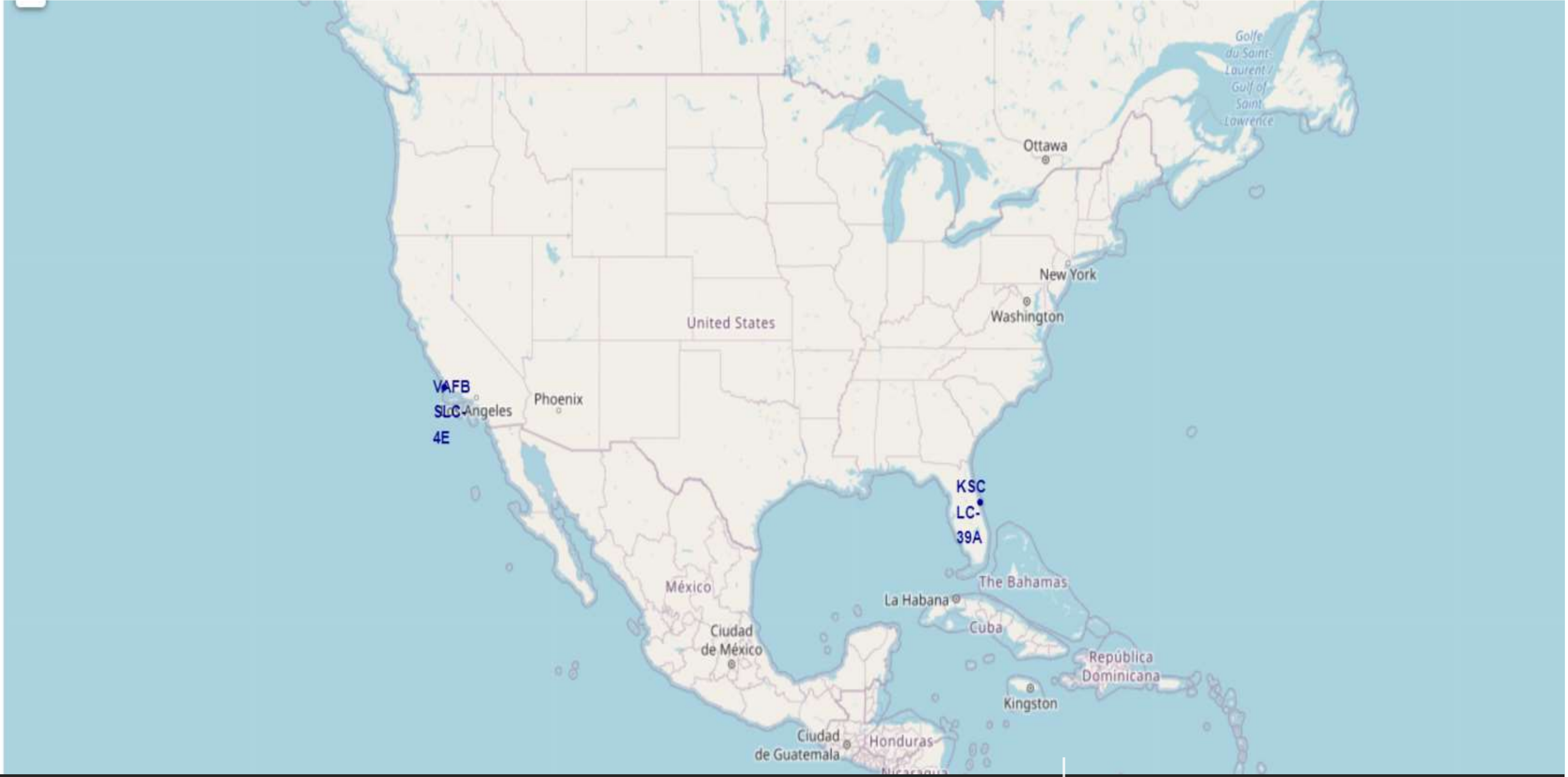
DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2018-08-01	01:00:00	F9 B4 B1043.1	CCAFS SLC-40	Zuma	5000	LEO	Northrop Grumman	Success (payload status unclear)	Success (ground pad)
2017-07-09	14:00:00	F9 B4 B1040.1	KSC LC-39A	Boeing X-37B OTV-5	4990	LEO	U.S. Air Force	Success	Success (ground pad)
2017-03-06	21:07:00	F9 FT B1035.1	KSC LC-39A	SpaceX CRS-11	2708	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-01-05	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2016-04-03	23:35:00	F9 FT B1020	CCAFS LC-40	SES-9	5271	GTO	SES	Success	Failure (drone ship)
2015-10-01	09:47:00	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)

Section 3

Launch Sites Proximities Analysis

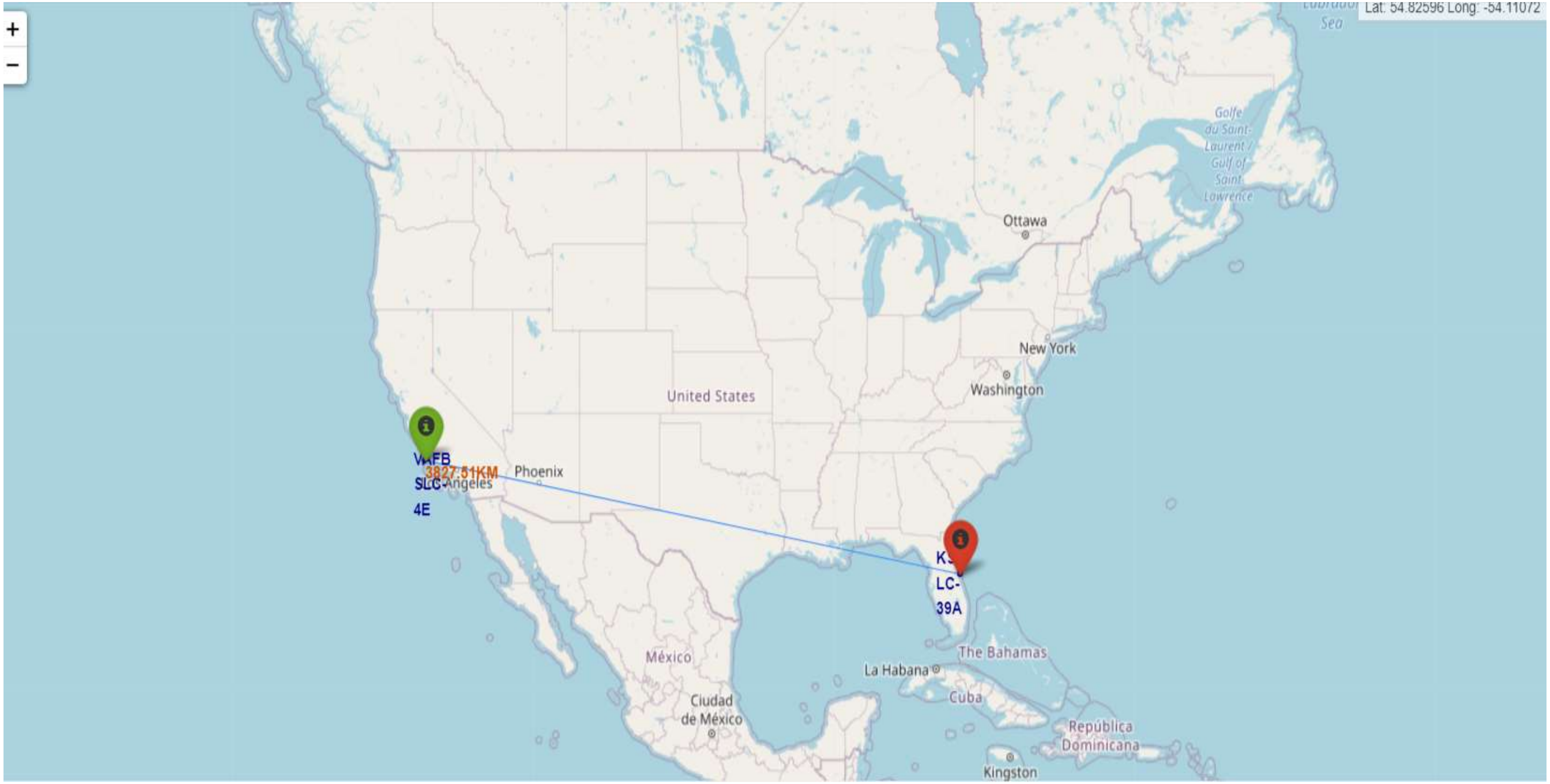


All launch sites marked on a map



Success / failed launches marked on the map

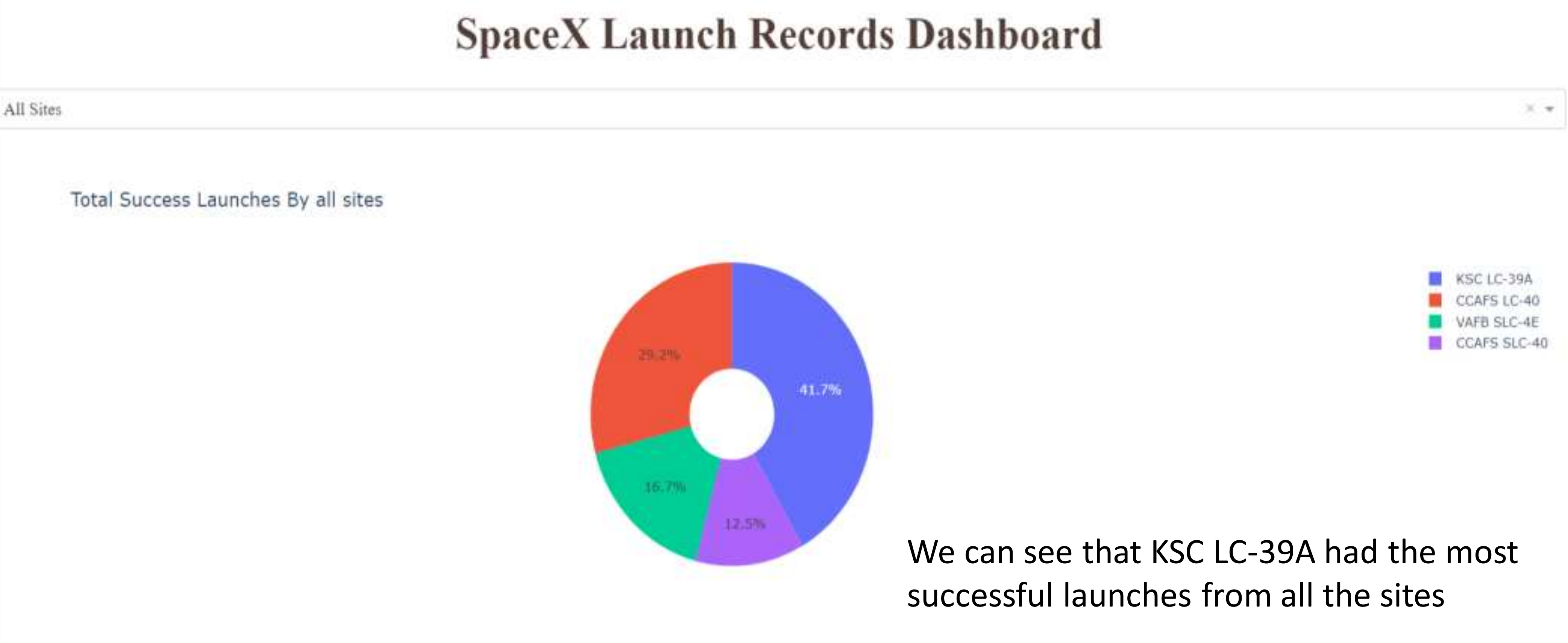
Distance between a launch site to its proximities



Section 4

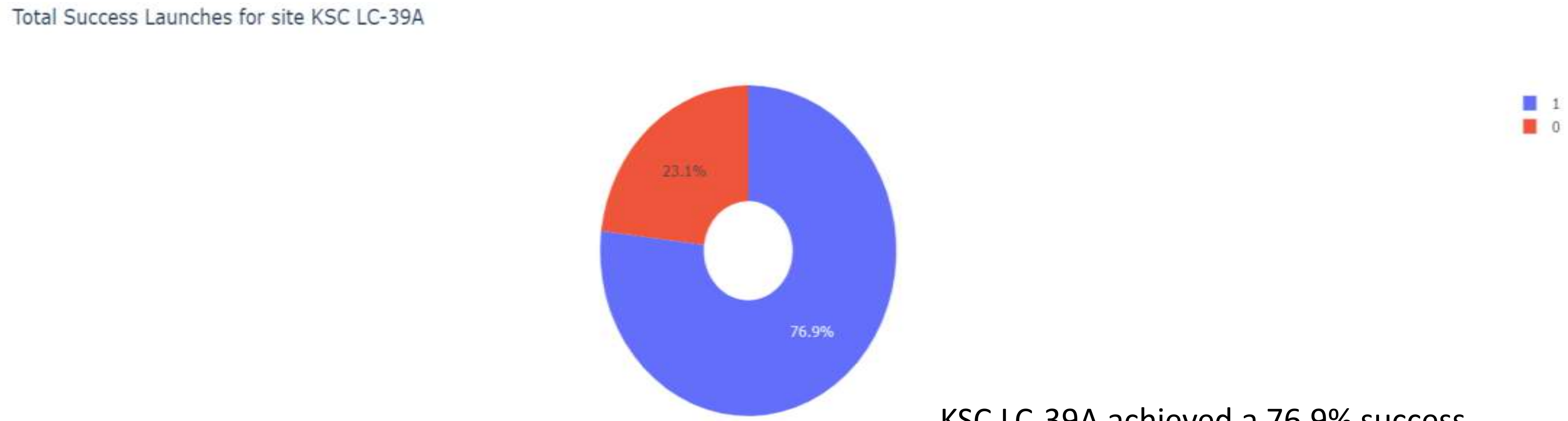
Build a Dashboard with Plotly Dash

Total success launches by all sites



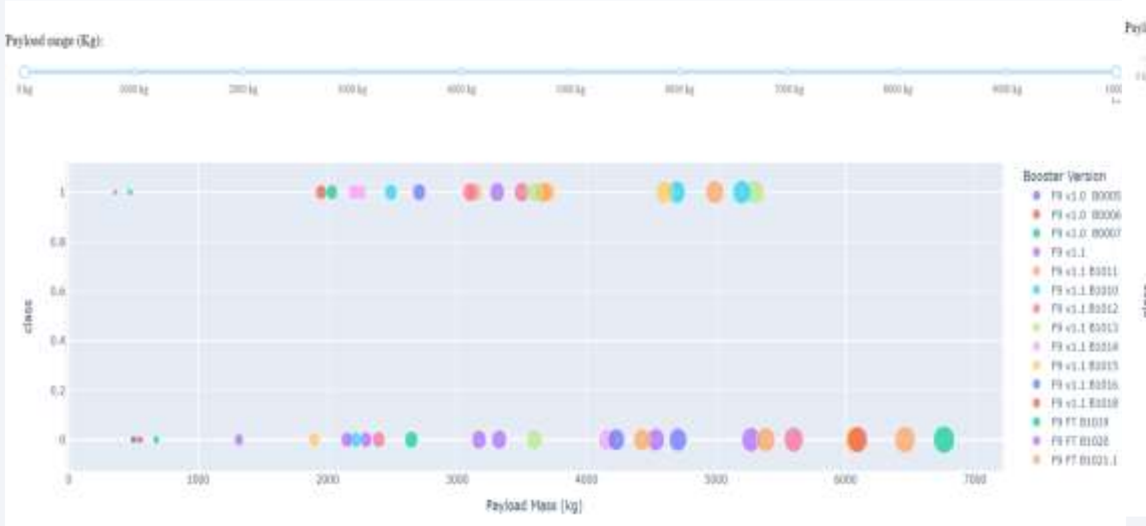
We can see that KSC LC-39A had the most successful launches from all the sites

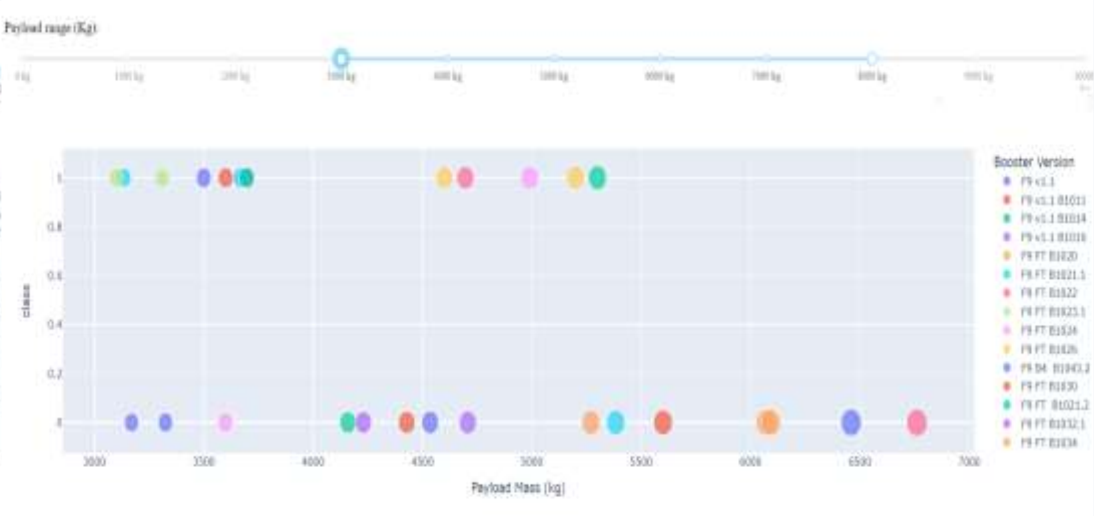
Success rate by site



KSC LC-39A achieved a 76.9% success rate while getting a23.1% failure rate

Payload vs launch outcome



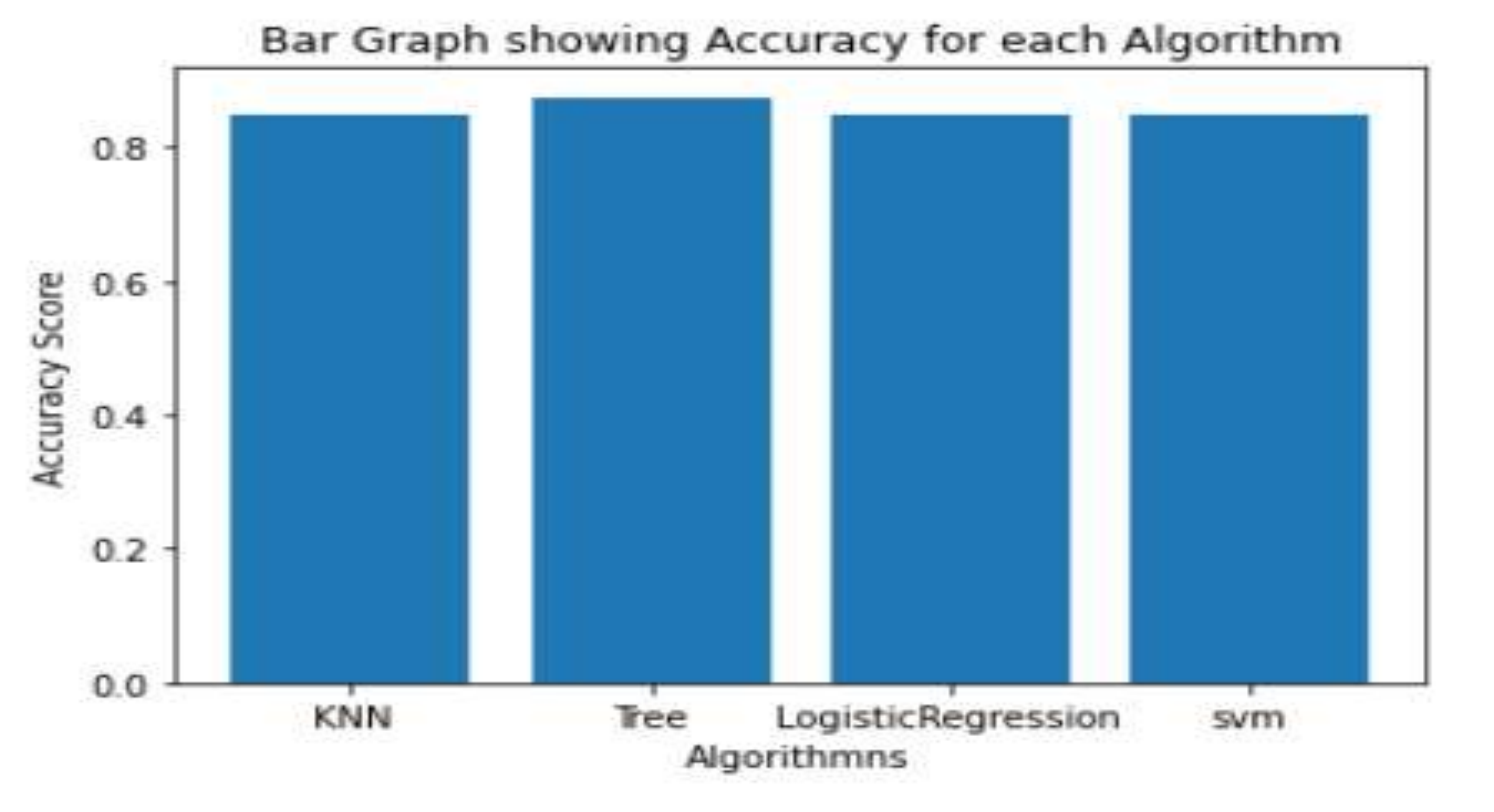


We can see the success rates for the low weighted payloads is higher than the heavy weighted payloads

Section 5

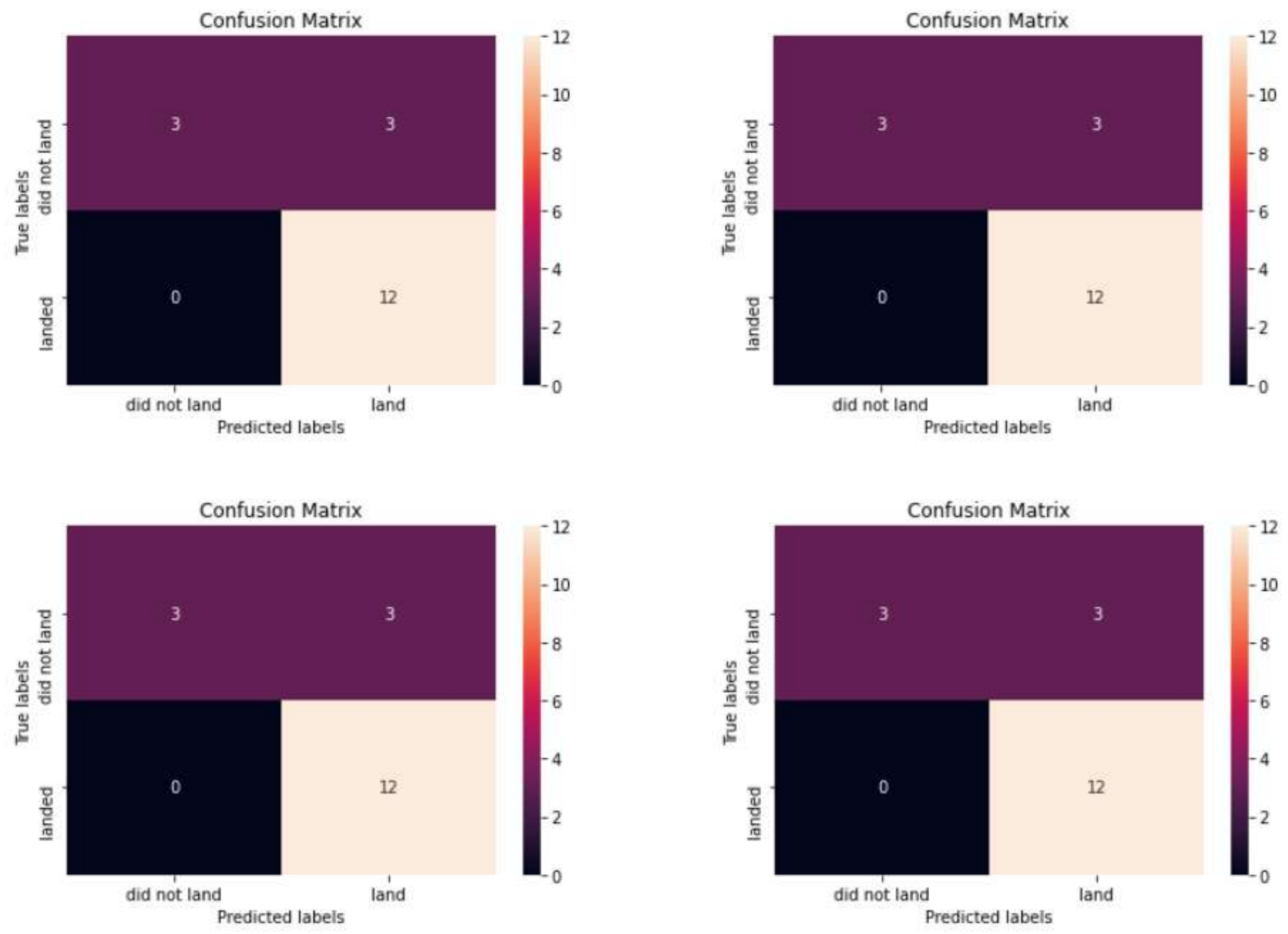
Predictive Analysis (Classification)

Classification Accuracy



Model accuracy on the Test Set

Confusion Matrix



Conclusions

- We have observed that:
- In the overall dataset, China's space agency has the most number of active rockets (satellites or payload etc) and in the ISRO, NASA, SpaceX dataset, ISRO has the most number of active space missions.
- It appears that the number of missions launched between 2000 and 2010 has gradually gone down while it appers that there is an upward spike in the number of missions launched in the last decade.
- Overall in the 1970s and after 2016 there is a high number of space missions launched.
- Of all the space missions launched by ISRO and SpaceX, none of them ever budgeted a $100 million dollar mark (all are under 100 million dollars).
- It appears that Russia launched a whopping 1600+ space missions overall. That is 8 times greater than NASA, 16 times greater than SpaceX and ISRO.
- That is it. It is fun to explore the space missions dataset and we have explored few things like who launched most number of missions, who spent least and most cost for each mission etc.
- I believe this is a fairly easy dataset to work on but I was intrigued with the "Space" word and hellbent to explore this dataset in the end.

Appendix

- All codes can be found on my GitHub

- GitHub repo: https://github.com/riya0395/testproject/tree/master

Thank you!