



IBM Developer  
SKILLS NETWORK

# Winning Space Race with Data Science

<Name>

<Date>



# 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
- SpaceX advertise Falcon9 rocket launches on it's website, with cost of 62 Million dollars, other providers cost upwards of 165 million dollars each, much of savings is because SpaceX can reuse the first stage.
- Problems you want to find answers
- The project task is to predicated if the first statge of the spaceX falcon 9 rocket will land successfully



Section 1

# Methodology

# Methodology

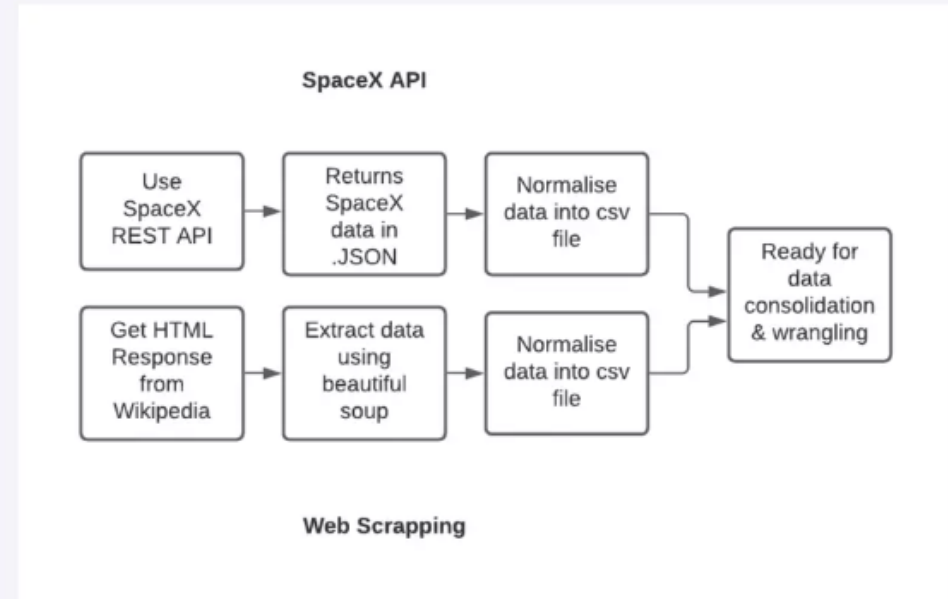
---

## Executive Summary

- Data collection methodology:
  - SpaceX rest API
  - Web scrapping form Wikipedia
- Perform data wrangling
  - One Hot encoding data files 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/](https://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/initiative1972/data-science/blob/master/10-IBM%20Data%20Science%20Capstone%20project.ipynb>

## 1. Getting Response from API

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url).json()
```

## 2. Converting Response to a .json file

```
response = requests.get(static_json_url).json()
data = pd.json_normalize(response)
```

## 3. Apply custom functions to clean data

```
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

```
getBoosterVersion(data)
```

## 4. Assign list to dictionary then 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}
```

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

## 5. Filter dataframe and export to flat file (.csv)

```
data_falcon9 = df.loc[df['BoosterVersion']!="Falcon 1"]
```

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



# Data Collection - Scrapping

- Web Scrapping from Wikipedia

<https://github.com/initiative1972/data-science/blob/master/10-IBM%20DS%20Capstone%20project-lab2-web%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 irrelevant 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

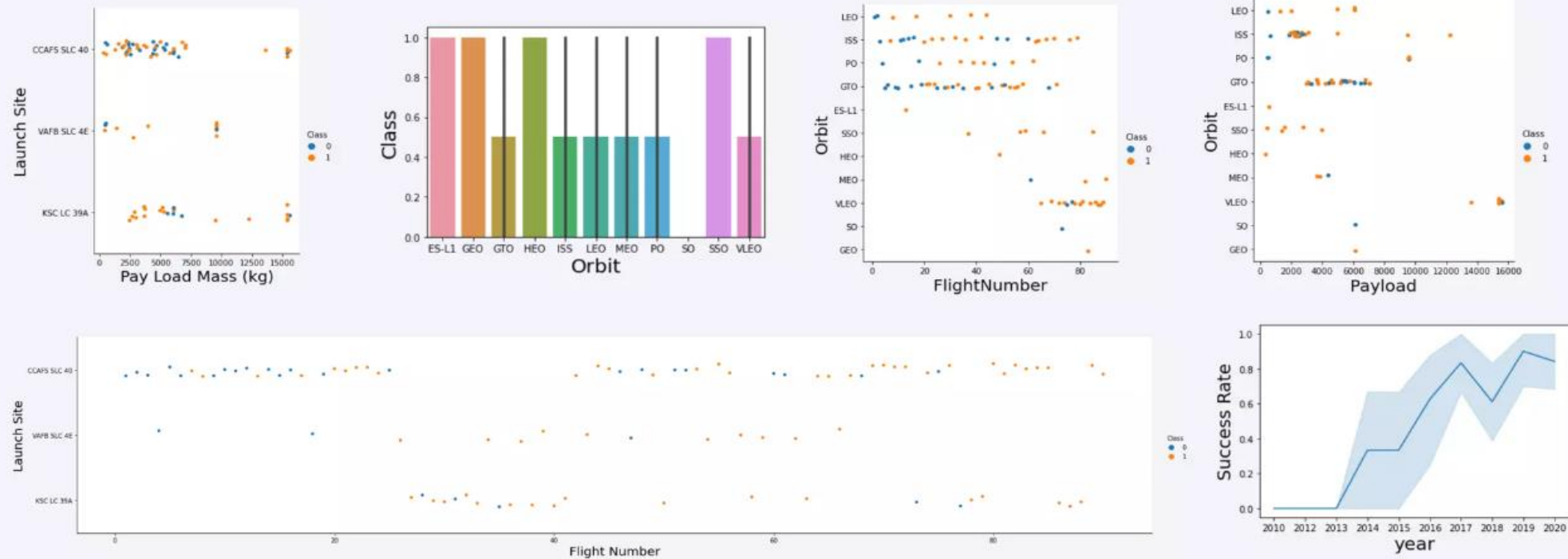
```
df.to_csv('spacex_web_scraped.csv', index=False)
```

# Data Wrangling

---

- Describe how data were processed
- You need to present your data wrangling process using key phrases and flowcharts
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose

# EDA with Data Visualization



# EDA with SQL

---

- Using bullet point format, summarize the SQL queries you performed
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose

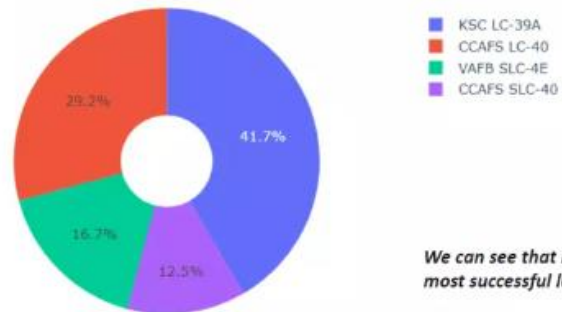


# Build an Interactive Map with Folium

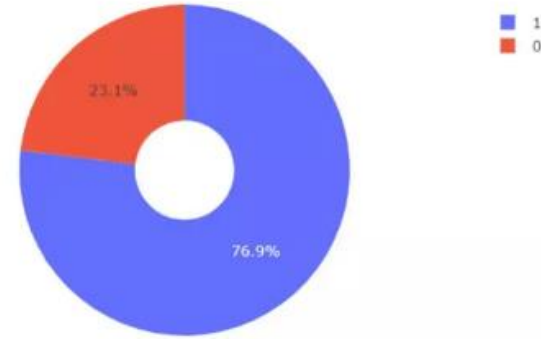


# Build a Dashboard with Plotly Dash

Total Success Launches By all sites

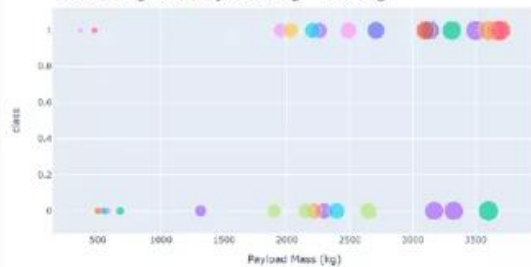


*We can see that KSC LC-39A had the most successful launches from all the sites*

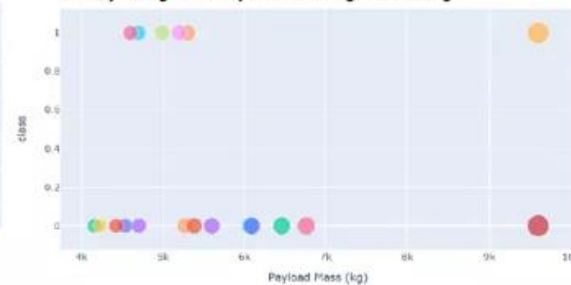


*KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate*

Low Weighted Payload 0kg – 4000kg



Heavy Weighted Payload 4000kg – 10000kg



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

# Predictive Analysis (Classification)

---

- Summarize how you built, evaluated, improved, and found the best performing classification model
- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

# Results

---

- 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.
- 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.



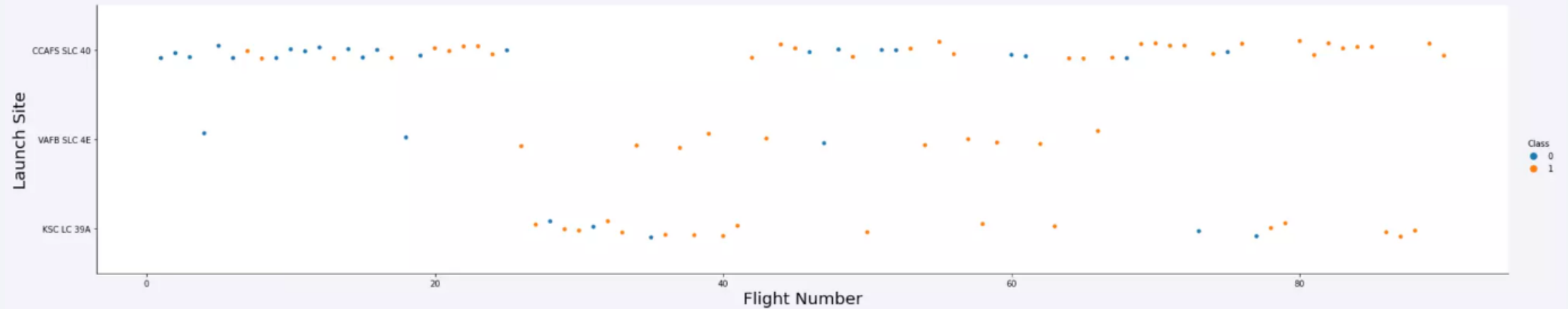
The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of red and cyan. A faint, light blue grid pattern is also visible, particularly in the lower half of the image. The overall effect is dynamic and technological.

Section 2

# Insights drawn from EDA

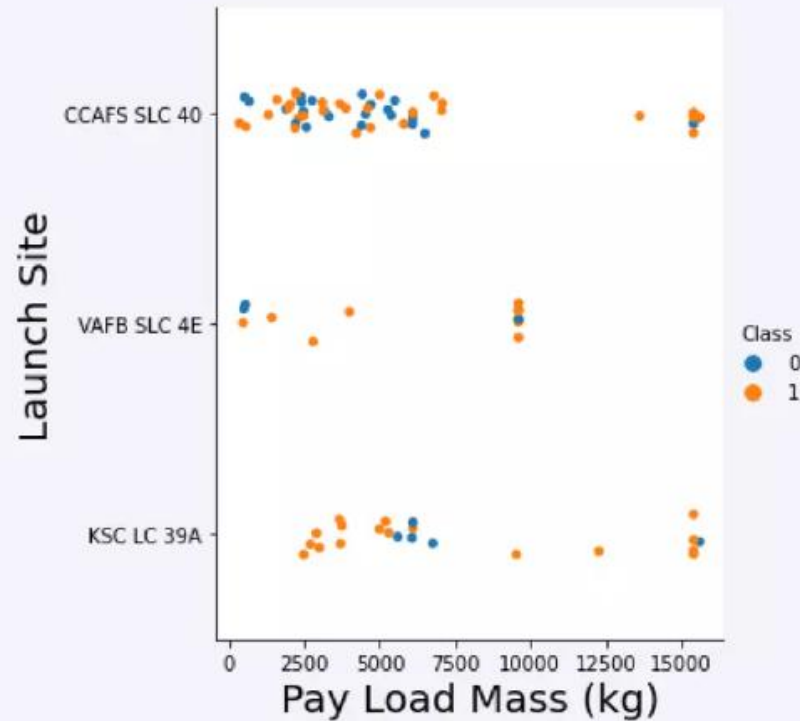


# Flight Number vs. Launch Site



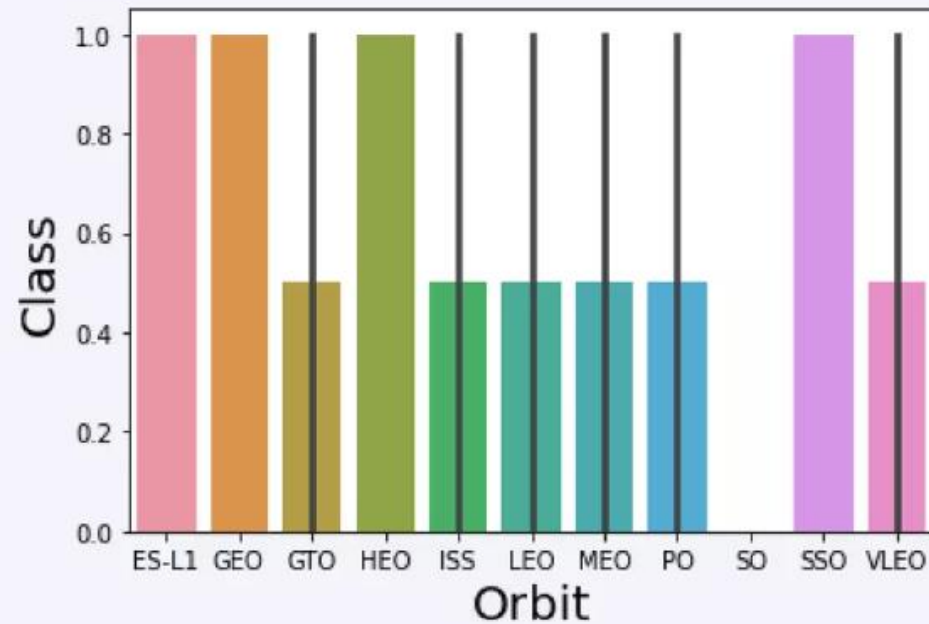
- Launches from the site of CCAFS SLC 40 are significantly higher than launches from other sites.

# 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.

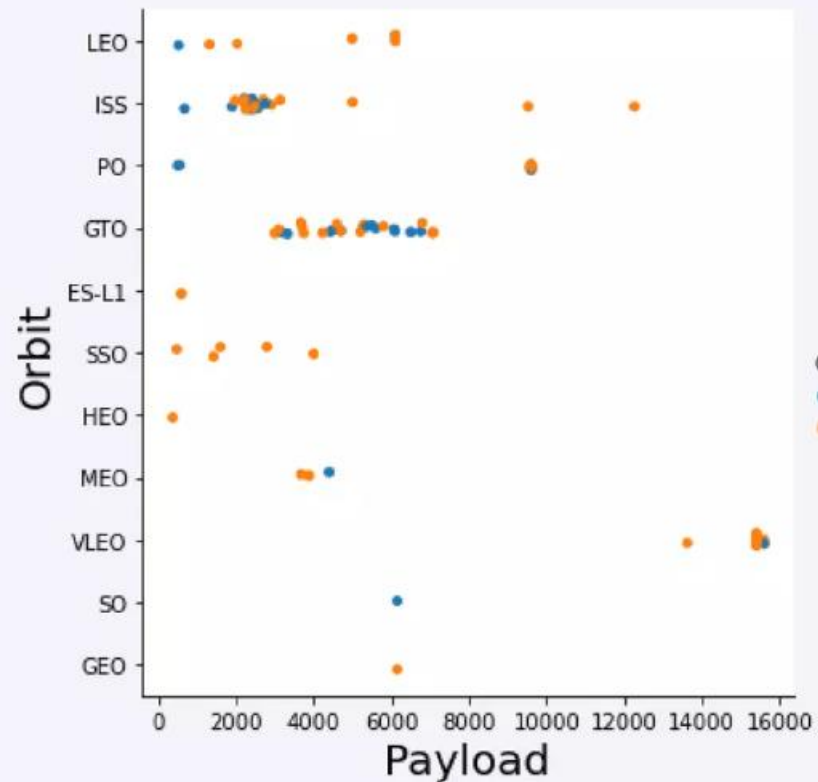


---



-

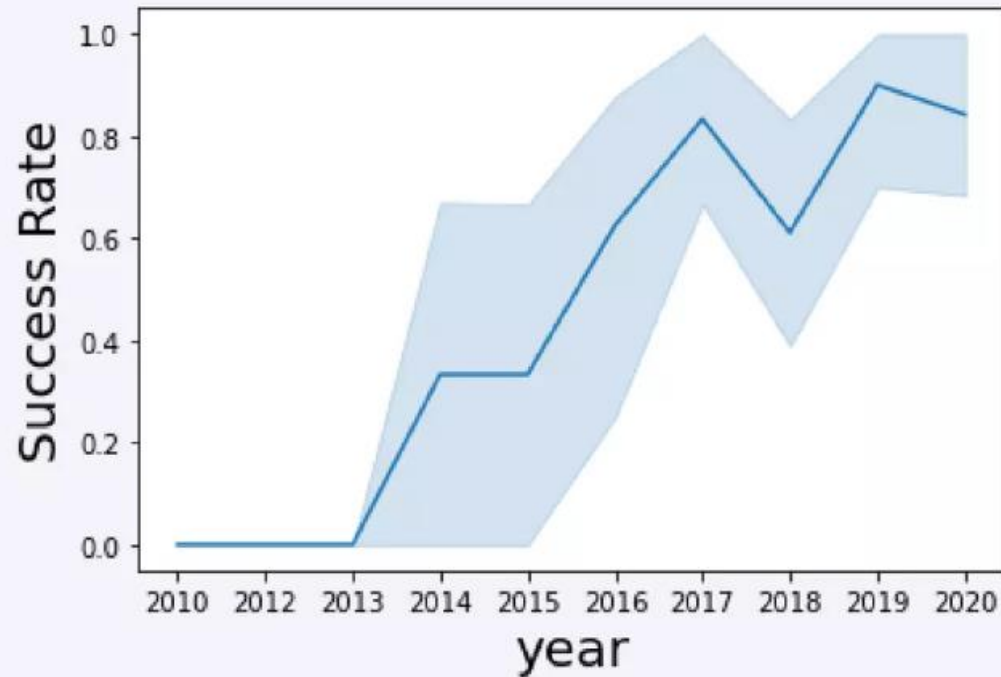
# Payload vs. Orbit Type



- 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 stabilised since 2019, potentially due to advance in technology and lessons learned.

# All Launch Site Names

---

- %sql select distinct(LAUNCH\_SITE) from SPACEXTBL

**launch\_site**

CCAFS LC-40

CCAFS SLC-40

KSC LC-39A

VAFB SLC-4E



# Launch Site Names Begin with 'CCA'

- %sql select \* from SPACEXTBL where LAUNCH\_SITE like 'CCA%' limit 5

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

---

- %sql select sum(PAYLOAD\_MASS\_\_KG\_) from SPACEXTBL where CUSTOMER = 'NASA (CRS)'

45596

# Average Payload Mass by F9 v1.1

---

- %sql select avg(PAYLOAD\_MASS\_\_KG\_) from SPACEXTBL where BOOSTER\_VERSION = 'F9 v1.1'

2928.400000

# First Successful Ground Landing Date

---

- %sql select min(DATE) from SPACEXTBL where Landing\_\_Outcome = 'Success (ground pad)'

2015-12-22

## Successful Drone Ship Landing with Payload between 4000 and 6000

---

- %sql select BOOSTER\_VERSION from SPACEXTBL where Landing\_\_Outcome = 'Success (drone ship)' and PAYLOAD\_MASS\_\_KG\_ > 4000 and PAYLOAD\_MASS\_\_KG\_ < 6000

**booster\_version**

F9 FT B1022

F9 FT B1026

F9 FT B1021.2

F9 FT B1031.2



# Total Number of Successful and Failure Mission Outcomes

---

- %sql select count(MISSION\_OUTCOME) from SPACEXTBL where MISSION\_OUTCOME = 'Success' or MISSION\_OUTCOME = 'Failure (in flight)'



100

# Boosters Carried Maximum Payload

---

- %sql select BOOSTER\_VERSION from SPACEXTBL where PAYLOAD\_MASS\_\_KG\_ = (select max(PAYLOAD\_MASS\_\_KG\_) from SPACEXTBL)

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

# 2015 Launch Records

- %sql select \* from SPACEXTBL where Landing\_\_Outcome like 'Success%' and (DATE between '2015-01-01' and '2015-12-31') order by date desc

time_utc_	booster_version	launch_site	payload	payload_mass_kg_	orbit	customer	mission_outcome	landing_outcome
14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (drone ship)
05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
05:34:00	F9 FT B1022	CCAFS LC-	JCSAT-14	4600	GTO	SKY Perfect JSAT	Success	Success (drone ship)

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

- %sql select \* from SPACEXTBL where Landing\_\_Outcome like 'Success%' and (DATE between '2010-06-04' and '2017-03-20') order by date desc

2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
2016-05-06	05:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)
2015-12-22	01:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The background is a deep blue gradient.

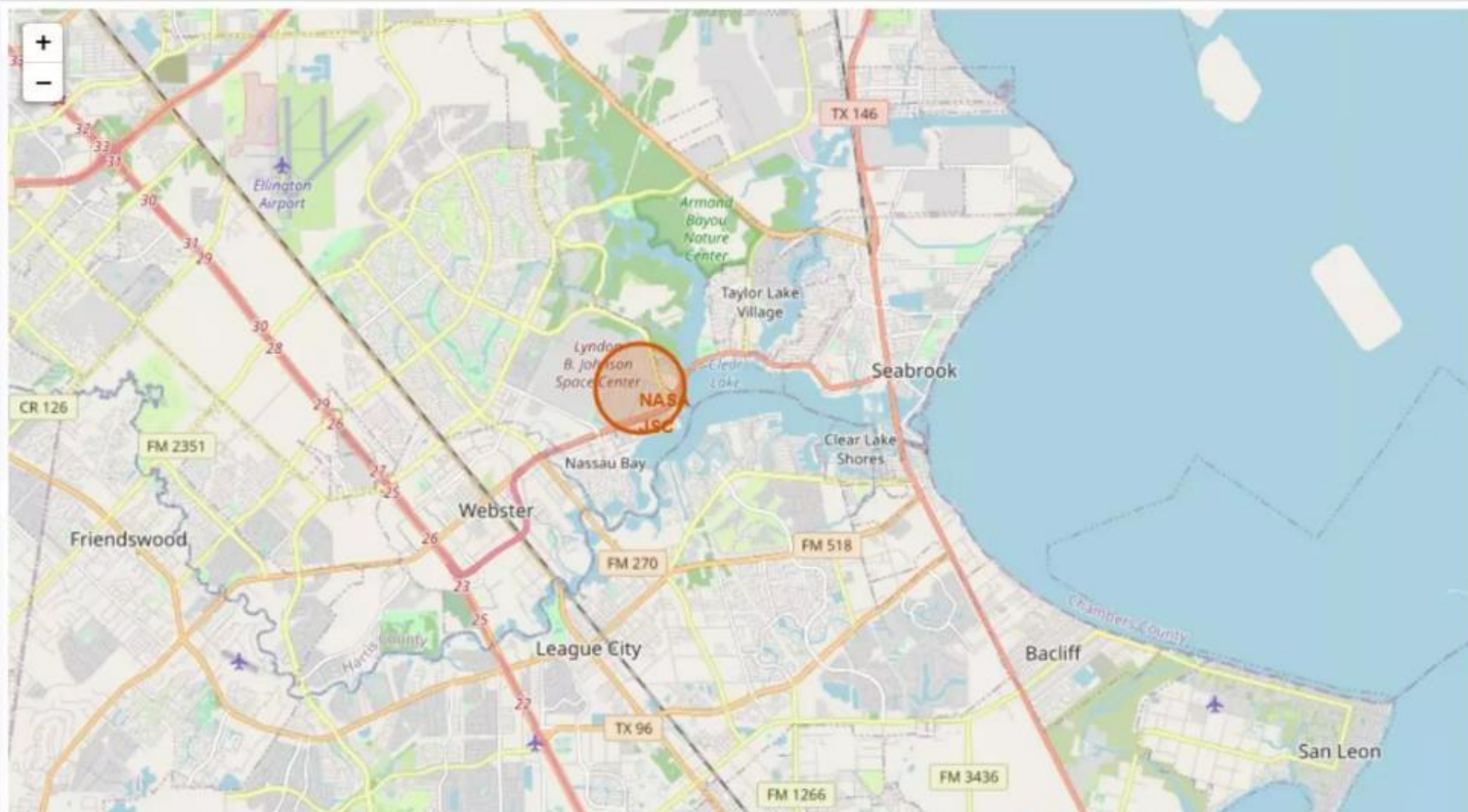
Section 3

# Launch Sites Proximities Analysis

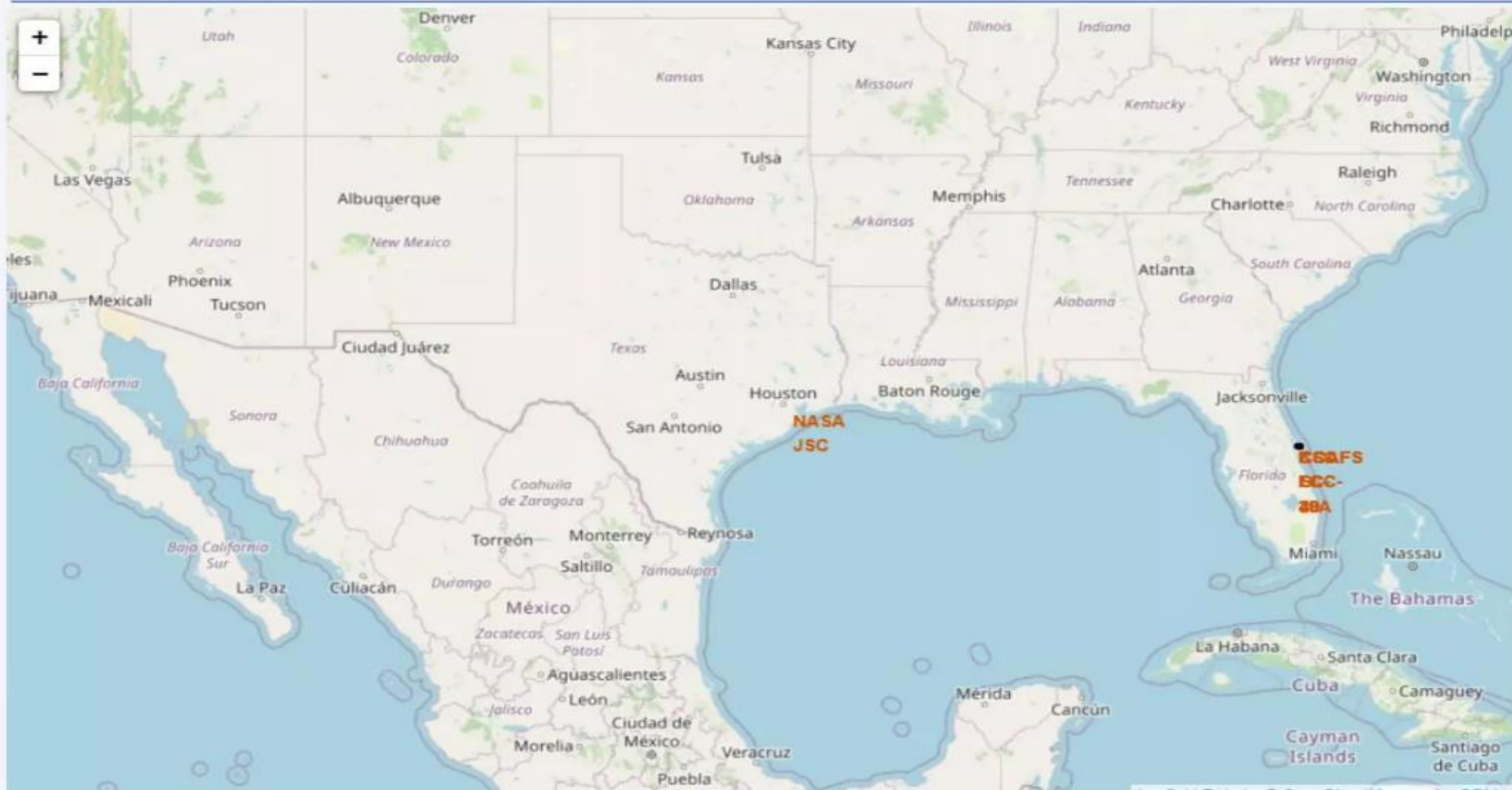


# <Folium Map Screenshot 1>

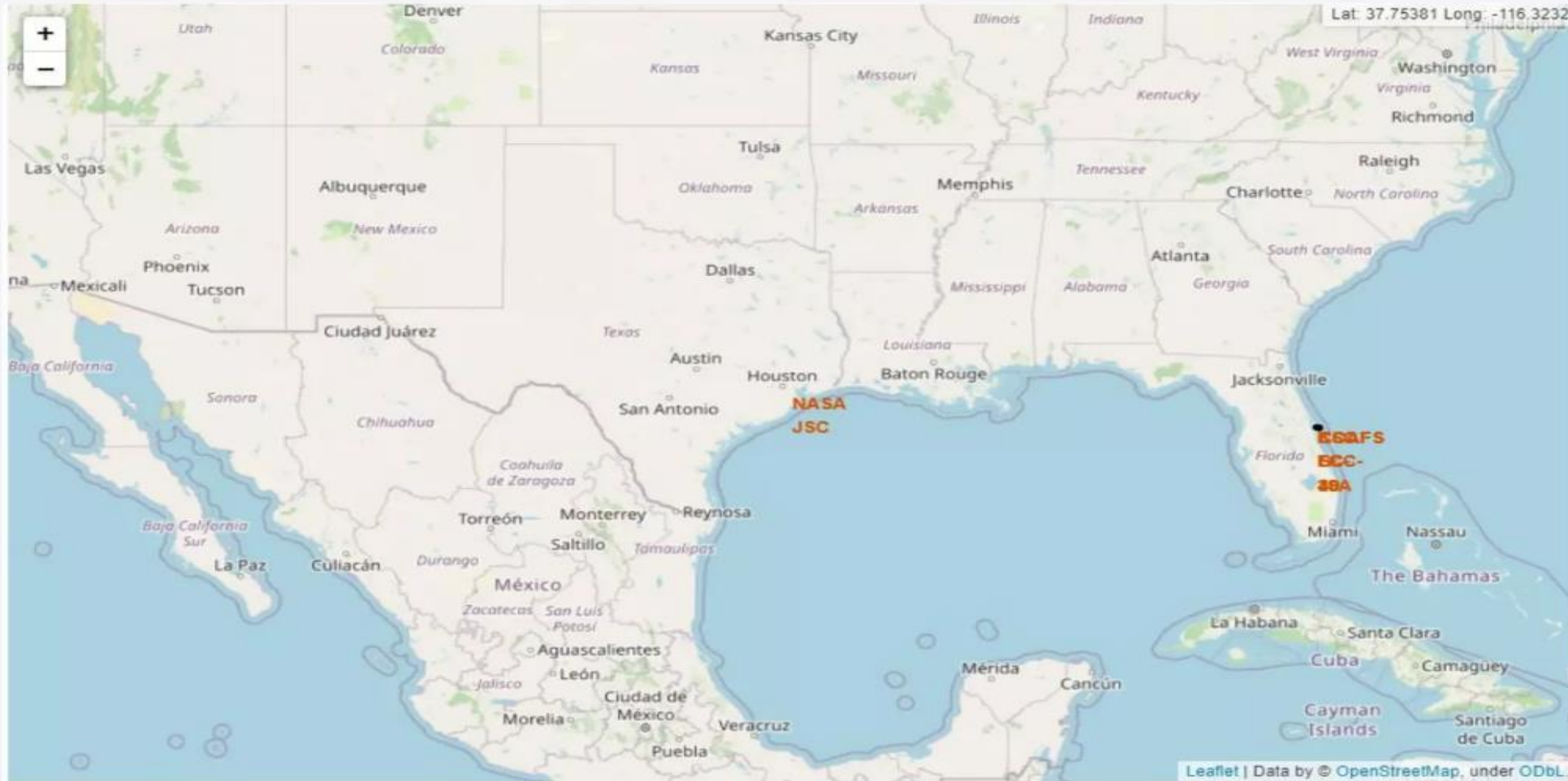
---



## <Folium Map Screenshot 2>



# <Folium Map Screenshot 3>







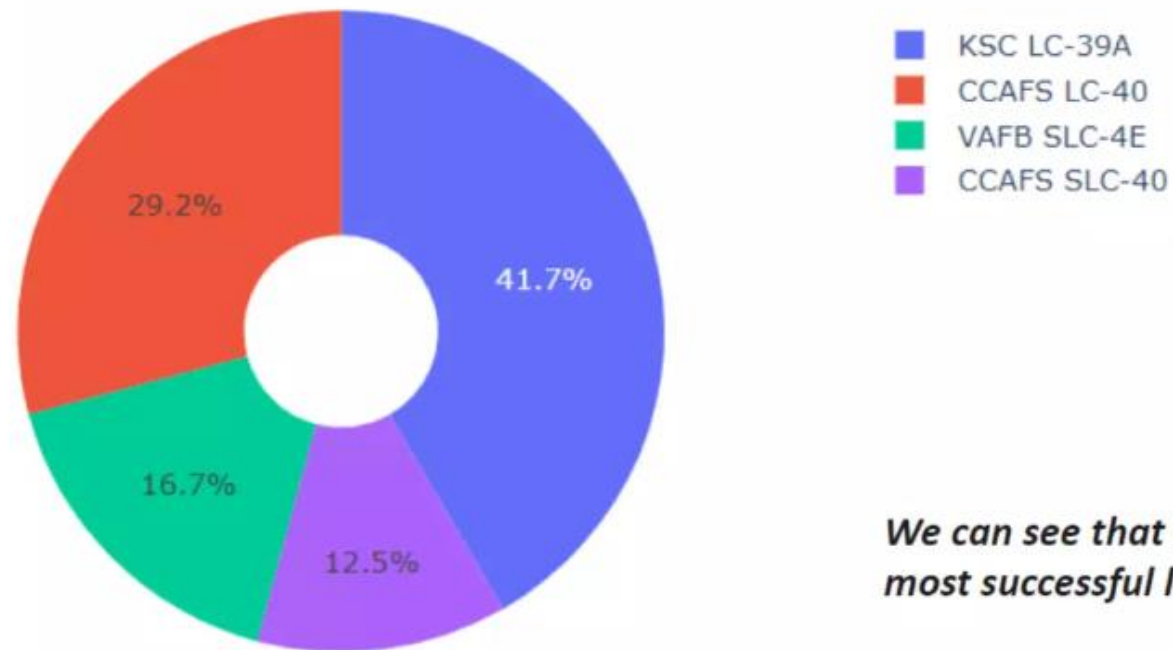
Section 4

# Build a Dashboard with Plotly Dash

# <Dashboard Screenshot 1>

---

Total Success Launches By all sites

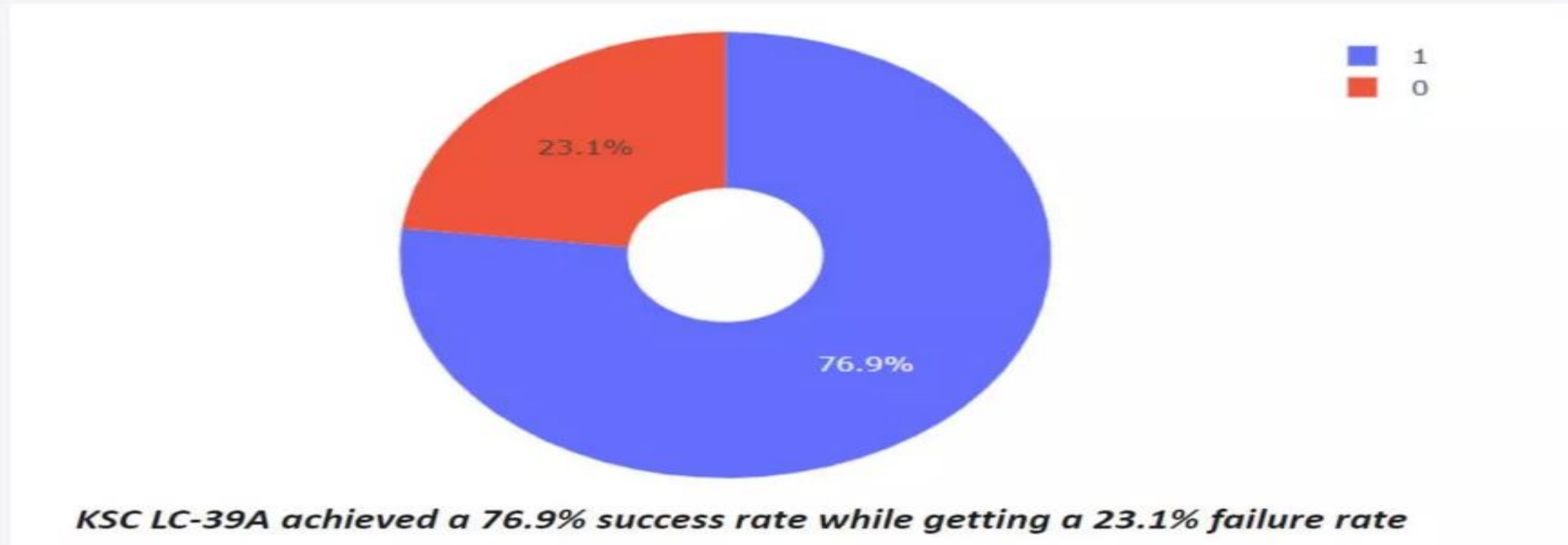


*We can see that KSC LC-39A had the most successful launches from all the sites*

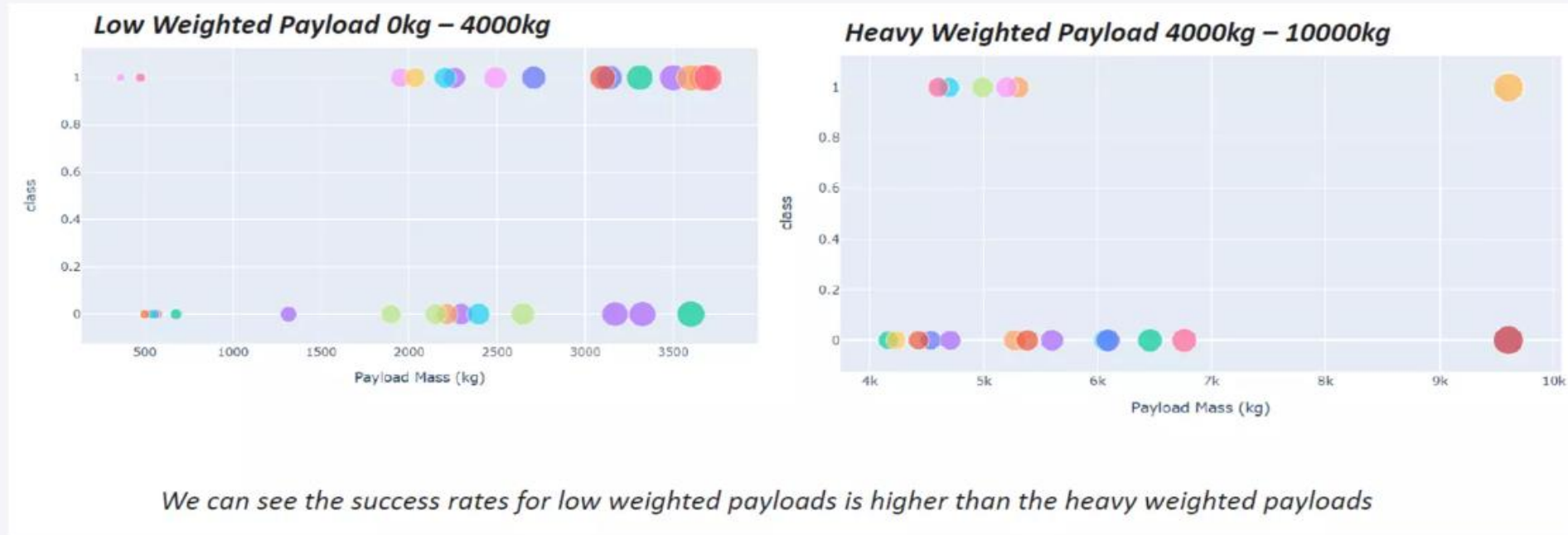


## <Dashboard Screenshot 2>

---



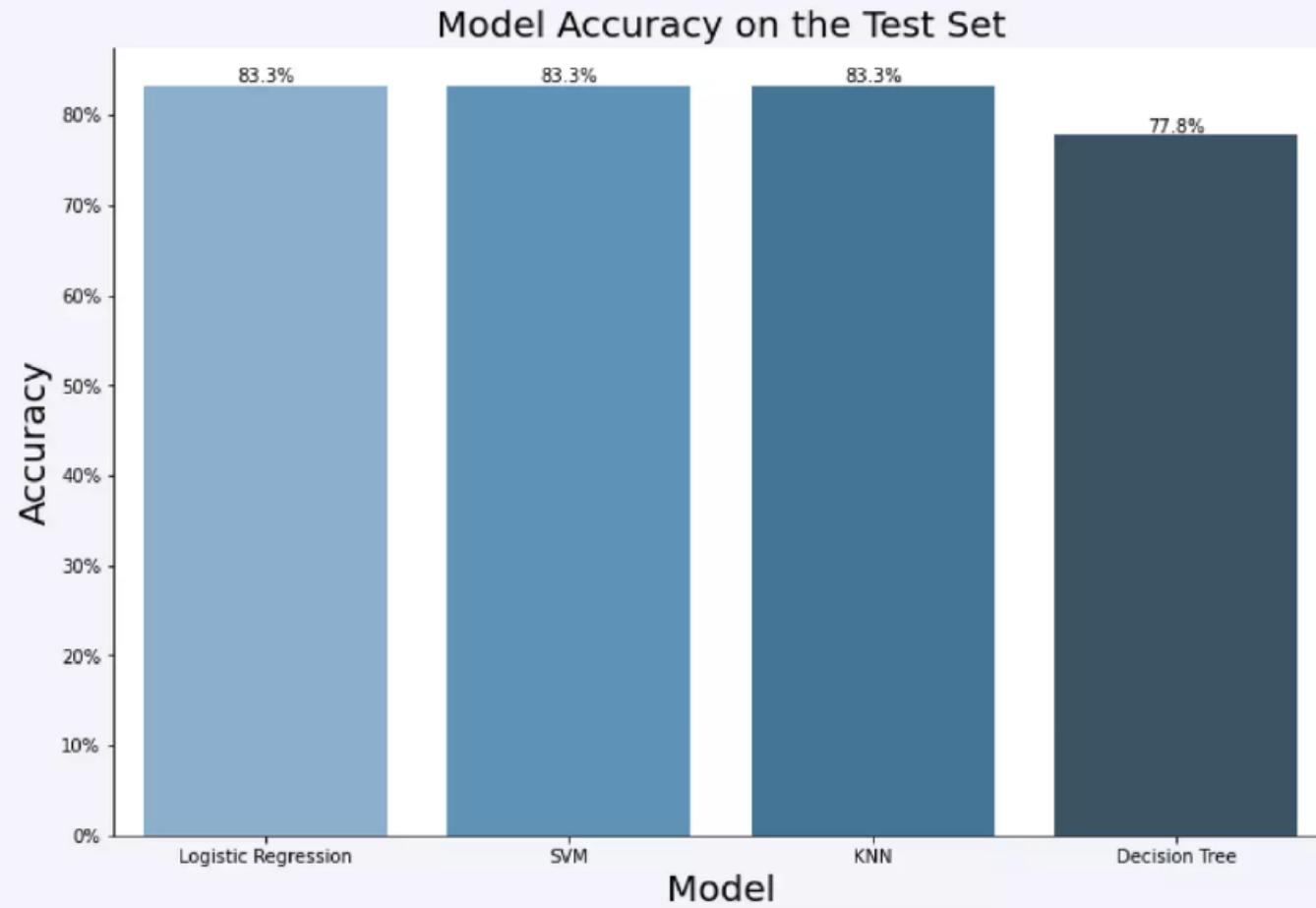
## <Dashboard Screenshot 3>



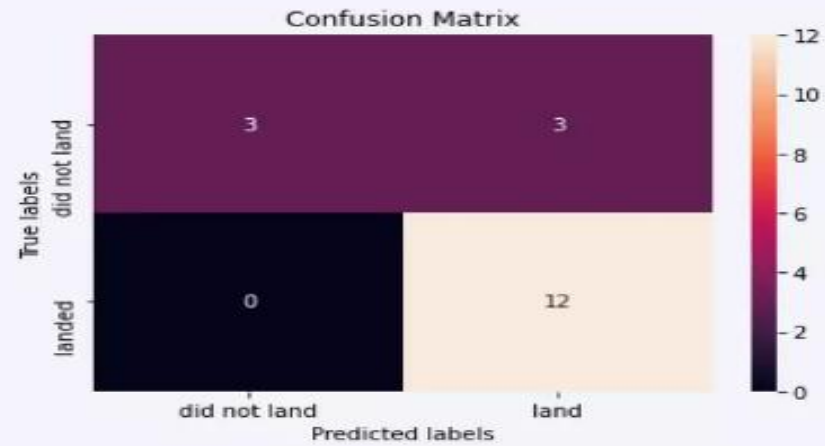
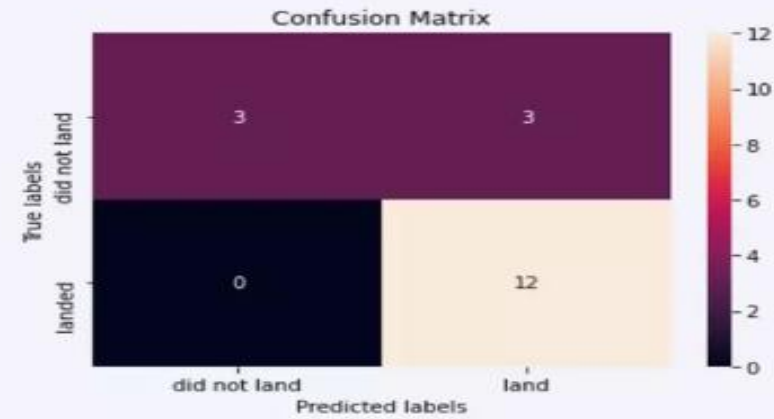
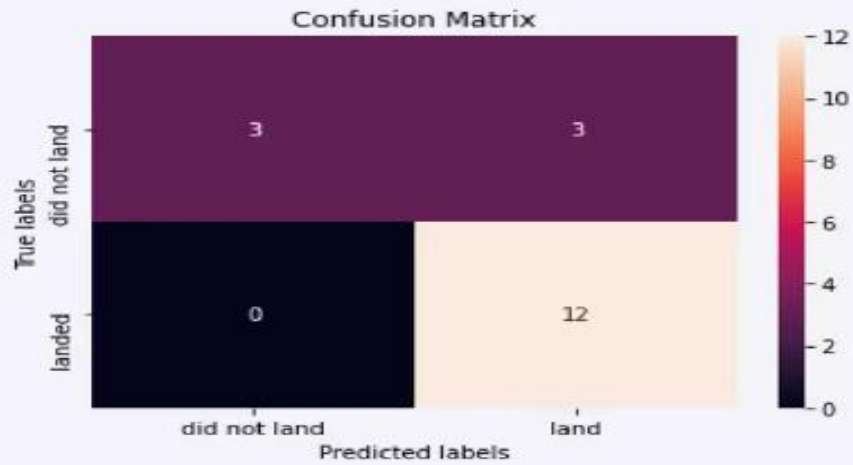
Section 5

# Predictive Analysis (Classification)

# Classification Accuracy



# Confusion Matrix





# Conclusions

---

- 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.
- 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.

# Appendix

---

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!

