IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Sumanta Das>
<09-Sep.-2023>

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collecting & Data Wrangling.
  - EDA and interactive visual analytics
  - EDA with Data Visualization.
  - EDA with SQL.
  - Building a interactive map with Folium.
  - Building a Dashboard with Plotly Dash.
  - Predictive Analysis(Classification).
- Summary of all results
  - predictive analysis (classification)
  - interactive map with Folium.
  - EDA with Visualization and SQL.

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. 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.
- Problems you want to find answers
  - The Problem task answer is, we will predict if the Falcon 9 first stage will land successfully.

Section 1

Methodology

Methodology

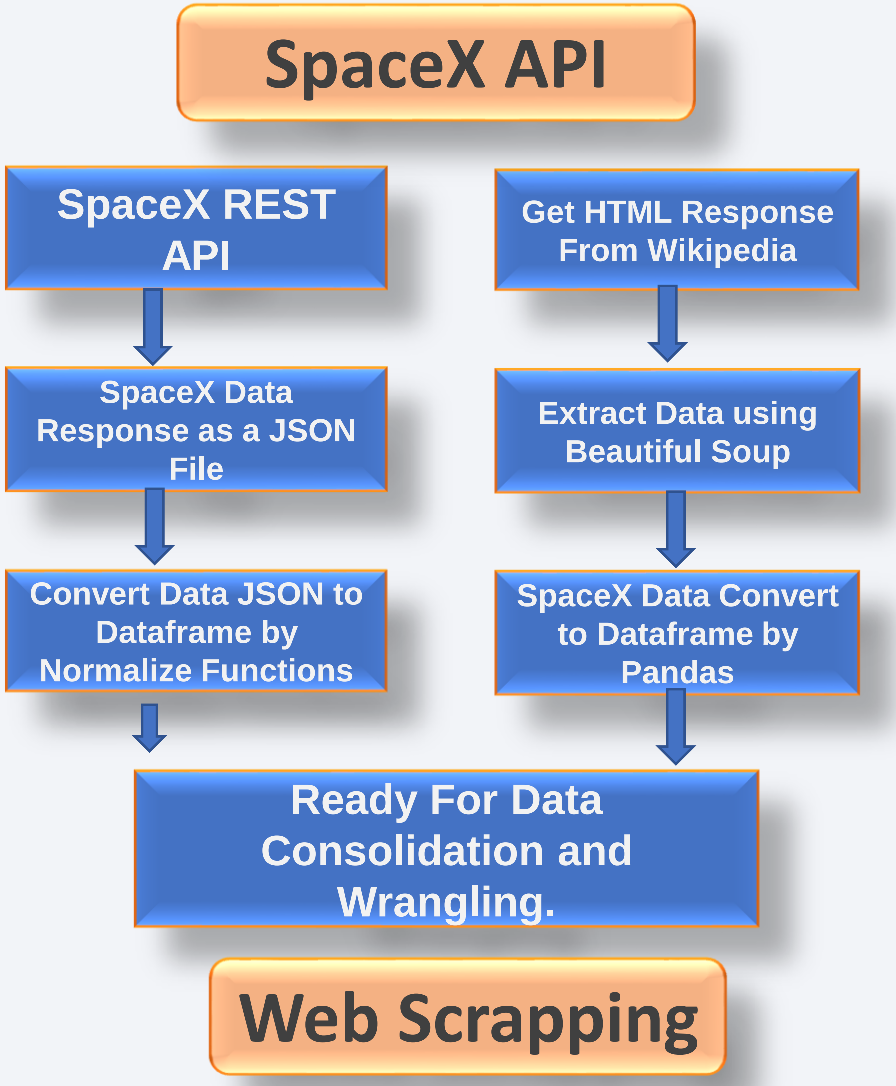
Executive Summary

- Data collection methodology:
  - Specifically the SpaceX REST API or URL, starts with api.spacexdata.com/v4/.
  - Another popular data source for obtaining Falcon 9 Launch data is web scraping related Wikipedia pages.
- Perform data wrangling
  - We want to transform this raw data into a clean dataset which provides meaningful data on the situation we are trying to address: Wrangling Data using an API, Sampling Data, and Dealing with Nulls. One hot encoding for ML 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,DT & SVM models have been built & evaluate for the best classifier.

Data Collection

Datasets was Collected by Following Methods.

- we will be collecting with SpaceX launch data that is gathered from an API, specifically the SpaceX REST API. The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.
- This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- Another popular data source for obtaining Falcon 9 Launch data is web scraping related Wiki pages.
- you will be using the Python BeautifulSoup package to web scrape some HTML tables that contain valuable Falcon 9 launch records.



Data Collection – SpaceX API

- Data collection with SpaceX REST calls using key phrases and flowcharts

- https://github.com/SumantaDasyuvi/testrepo/blob/main/jupyter-labs-spacex-data-collection-api%20(2).ipynb

1. Getting Data from SpaceX API .

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

2. Response Data as JSON file and Converting data to Pandas Dataframe by using .json_normalize()

- response = requests.get(spacex_url)
- data=pd.json_normalize(response.json())

3. Apply Custom Functions for Filter the DF & finding Missing Values in DF and Replace null values.

- getBoosterVersion(data)
- BoosterVersion[0:5]
- data_falcon9=df[df['BoosterVersion']!='Falcon 1']
- data_falcon9.isnull().sum()
- data_falcon9['PayloadMass'].replace(np.nan,payloadmassavg,inplace=True)

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

3.Creating new Dataframe and cleaning Data.

```
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)

Data Collection - Scraping

- From Wikipedia, Web scraping process using key phrases and flowcharts.

- https://github.com/SumantaDasyuvi/testrepo/blob/main/jupyter-labs-webscraping.ipynb

1.Getting Response from Wikipedia page.

```
static_url="https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
data=requests.get(static_url).text
```

2. Extract Data by using BeautifulSoup and extract all columns/variable Names.

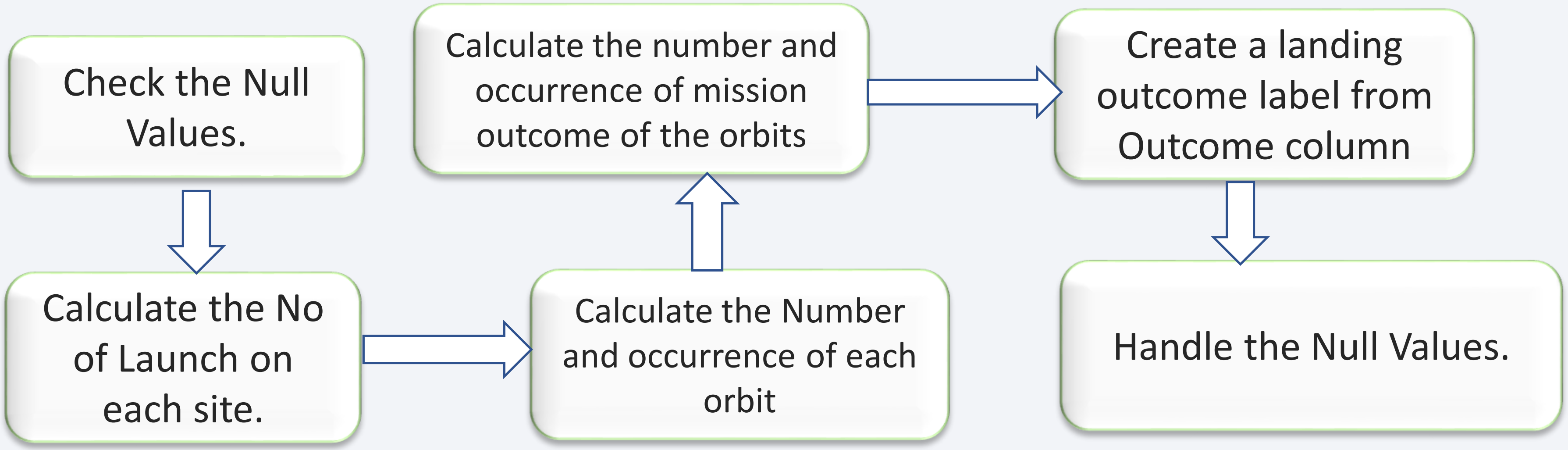
```
soup=BeautifulSoup(data,'html.parser')
html_tables=soup.find_all('table')
column_names = []
for row in first_launch_table.find_all('th'):
  name=extract_column_from_header(row)
  if name!=None and len(name)>0:
    column_names.append(name)
```

3. Data Convert Dict. to DataFrame by using Pandas.

```
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']=[]
df= pd.DataFrame({ key:pd.Series(value) for key, value in launch_dict.items() })
df = pd.DataFrame.from_dict(launch_dict)
```

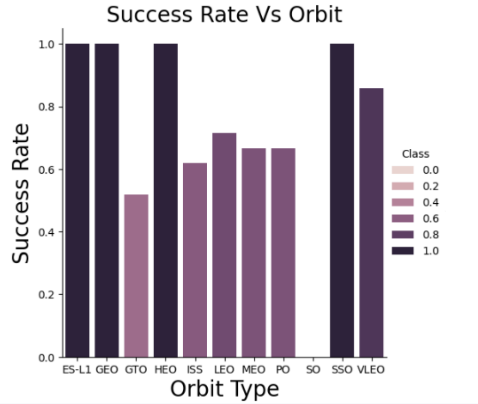
Data Wrangling

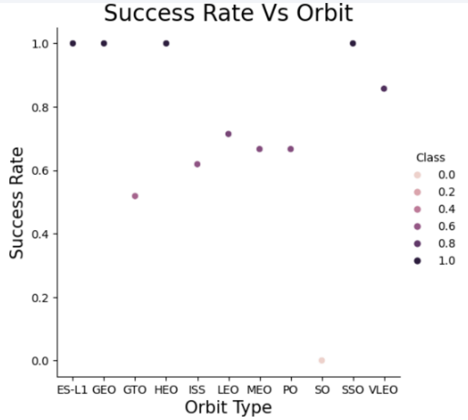
- Data Exploring by data analysis (find the null values and missing values and replace)

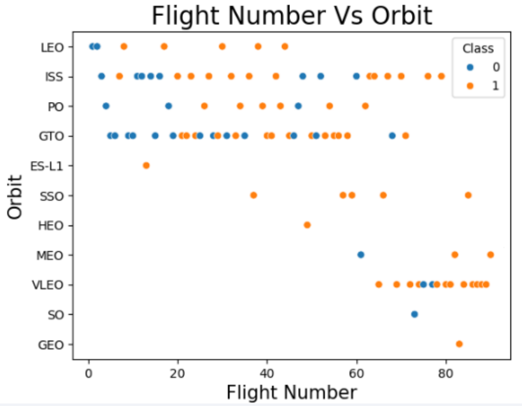


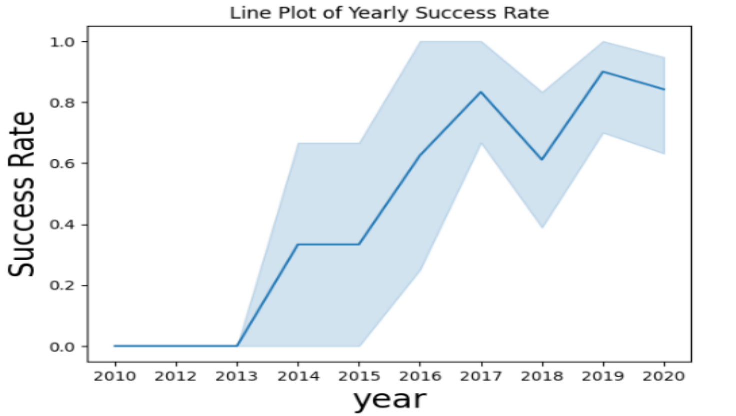
- https://github.com/SumantaDasyuvi/testrepo/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

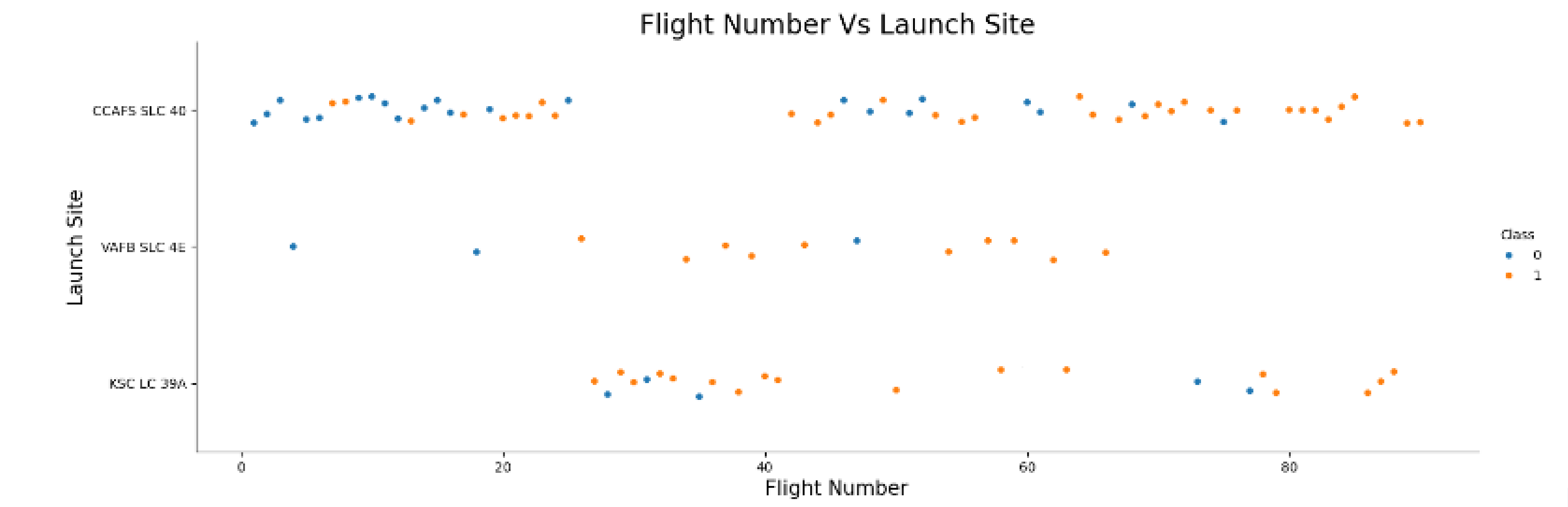
EDA with Data Visualization

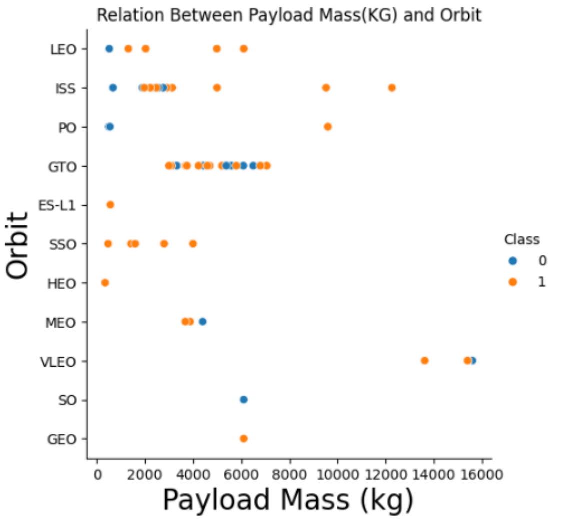










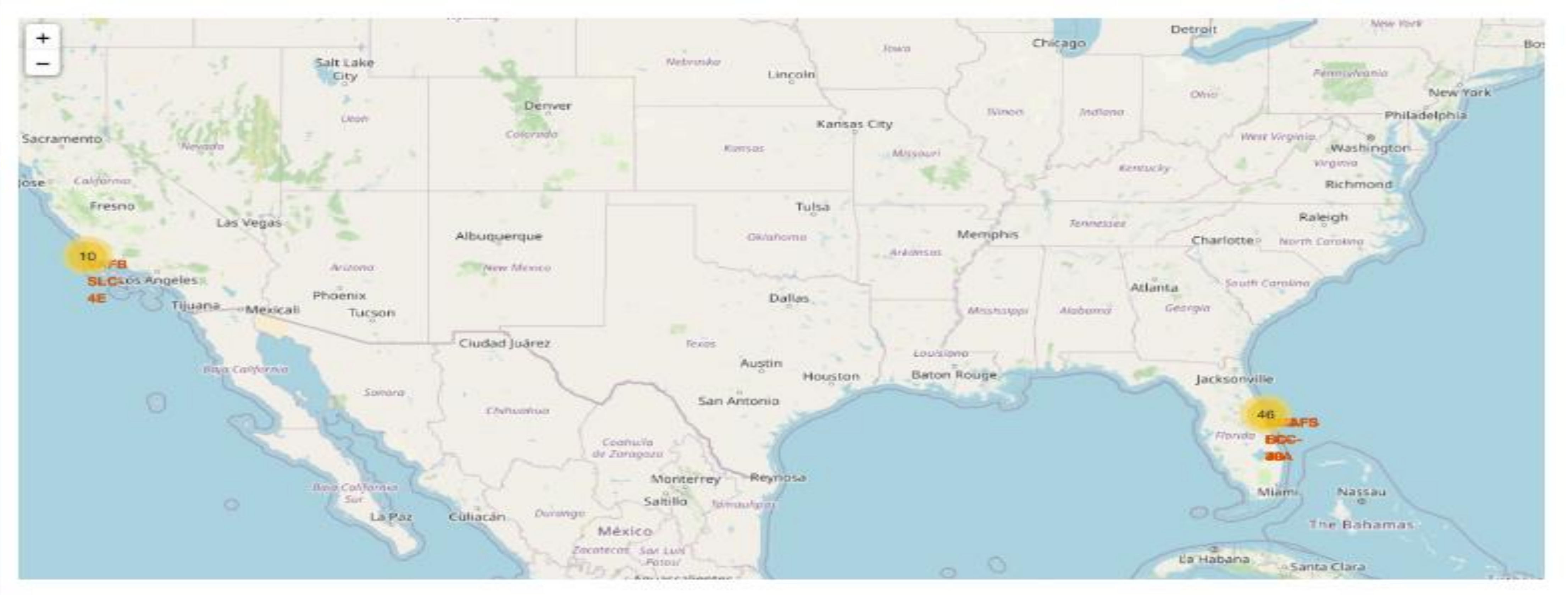


- https://github.com/SumantaDasyuvi/testrepo/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

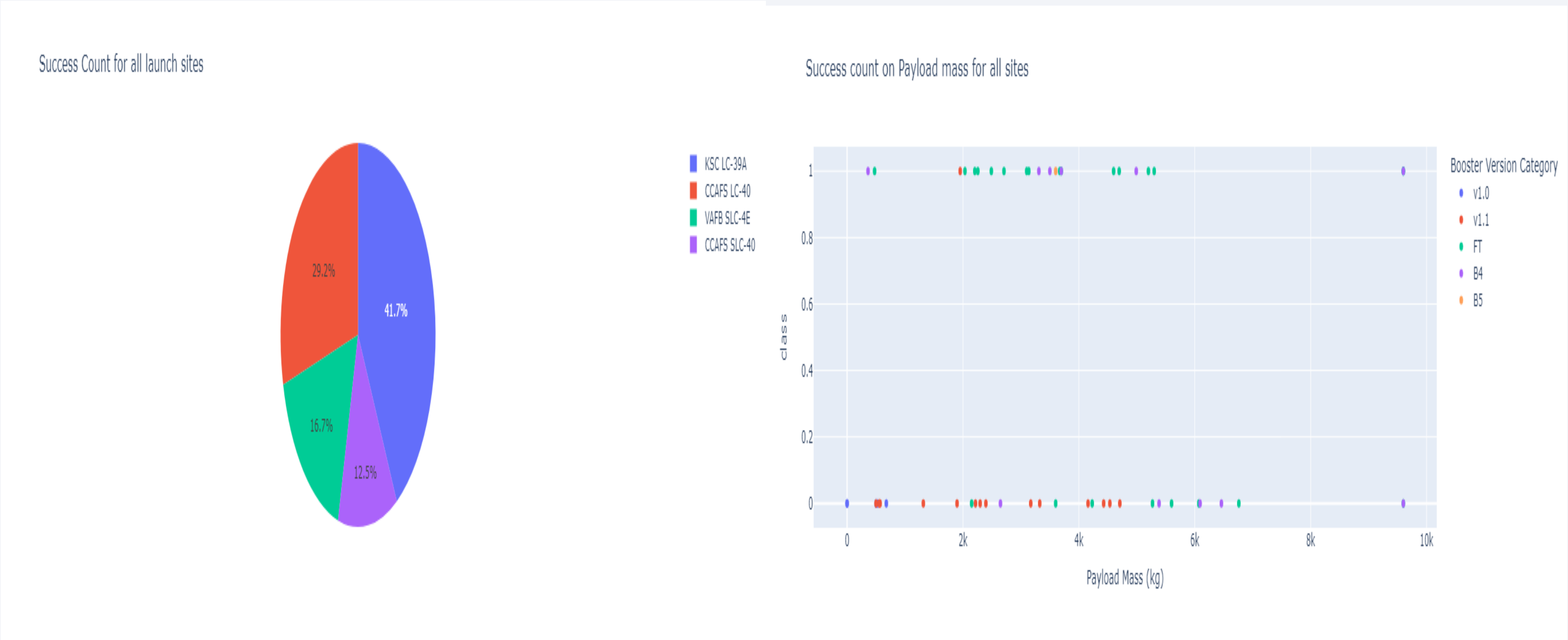
- Display the names of the unique launch sites in the space mission.
- Display 5 records where launch sites begin with the string 'CCA‘.
- Display the total payload mass carried by boosters launched by NASA (CRS).
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first succesful landing outcome in ground pad was acheived.
- 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 landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.
- https://github.com/SumantaDasyuvi/testrepo/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium



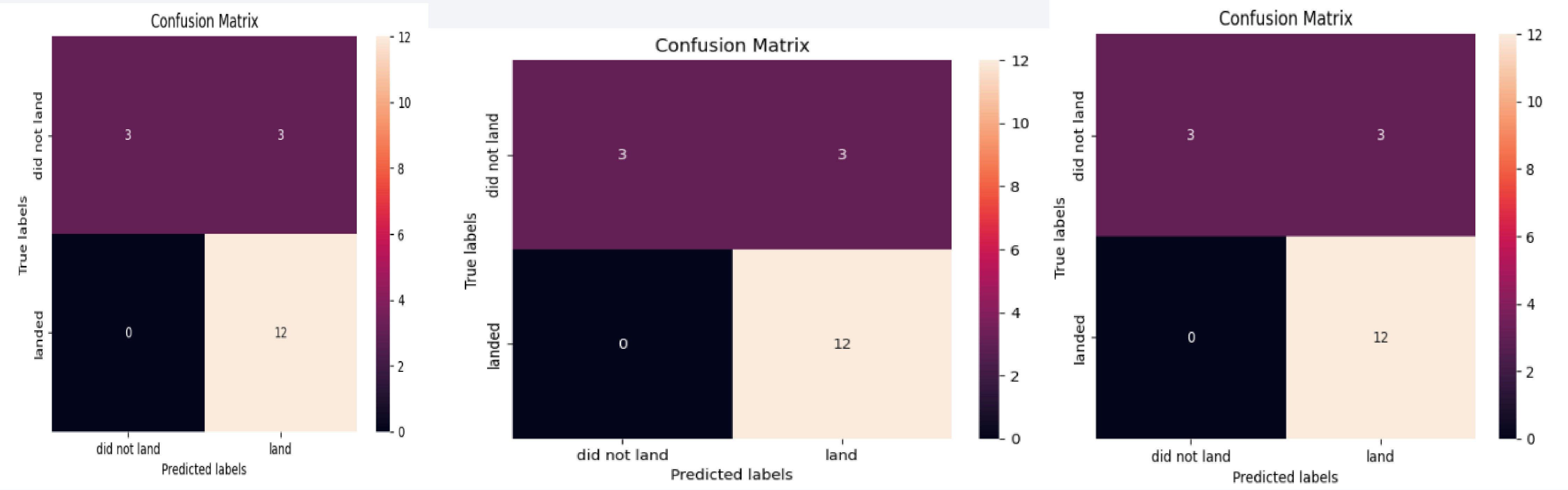
- Map Marker used for finding /aiming locations of launch site.
- https://github.com/SumantaDasyuvi/testrepo/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash



- KSC LC-39A is more success rate than others and Low weighted payloads is higher than more weighted payloads.
- https://github.com/SumantaDasyuvi/testrepo/blob/main/spacex_dash_app%20(1).py

Predictive Analysis (Classification)



- All Models are achieved highest accuracy at 83.33%, While SVM model preform better.
- Data Preprocessing → Convert data to Int() → Normalize the Data → The Train/Test Dataset → Data Modeling → Data Predicting → Data Evaluation → Confusion Matrix
- https://github.com/SumantaDasyuvi/testrepo/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

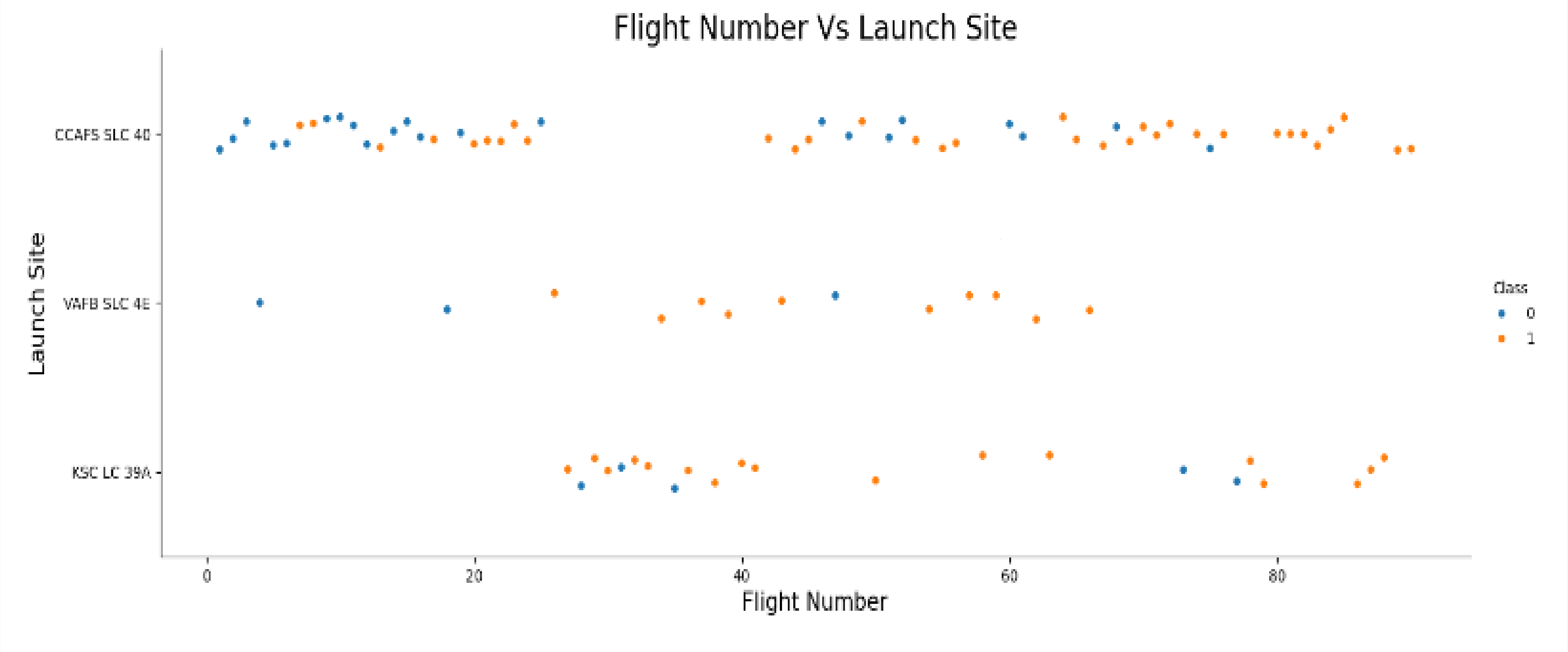
Results

- The SVM,KNN and Logistic Regression models are the best accuracy(83.33%) of the dataset.
- Low weighted payloads are perform well then more weighted payloads.
- SpaceX Launch success rate is directly proportional time in years. Eventually it will perfect launch.
- KSC LC-39A site is most success than other sites.

Section 2

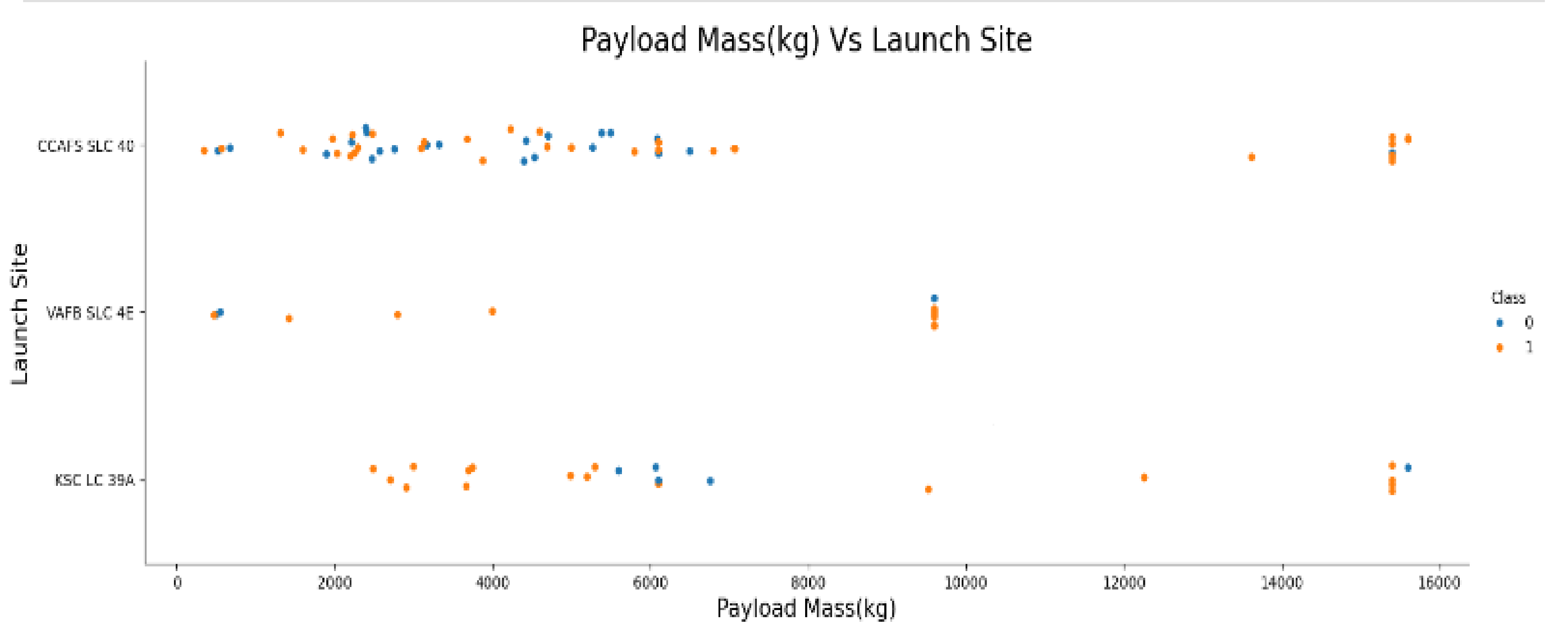
Insights drawn from EDA

Flight Number vs. Launch Site



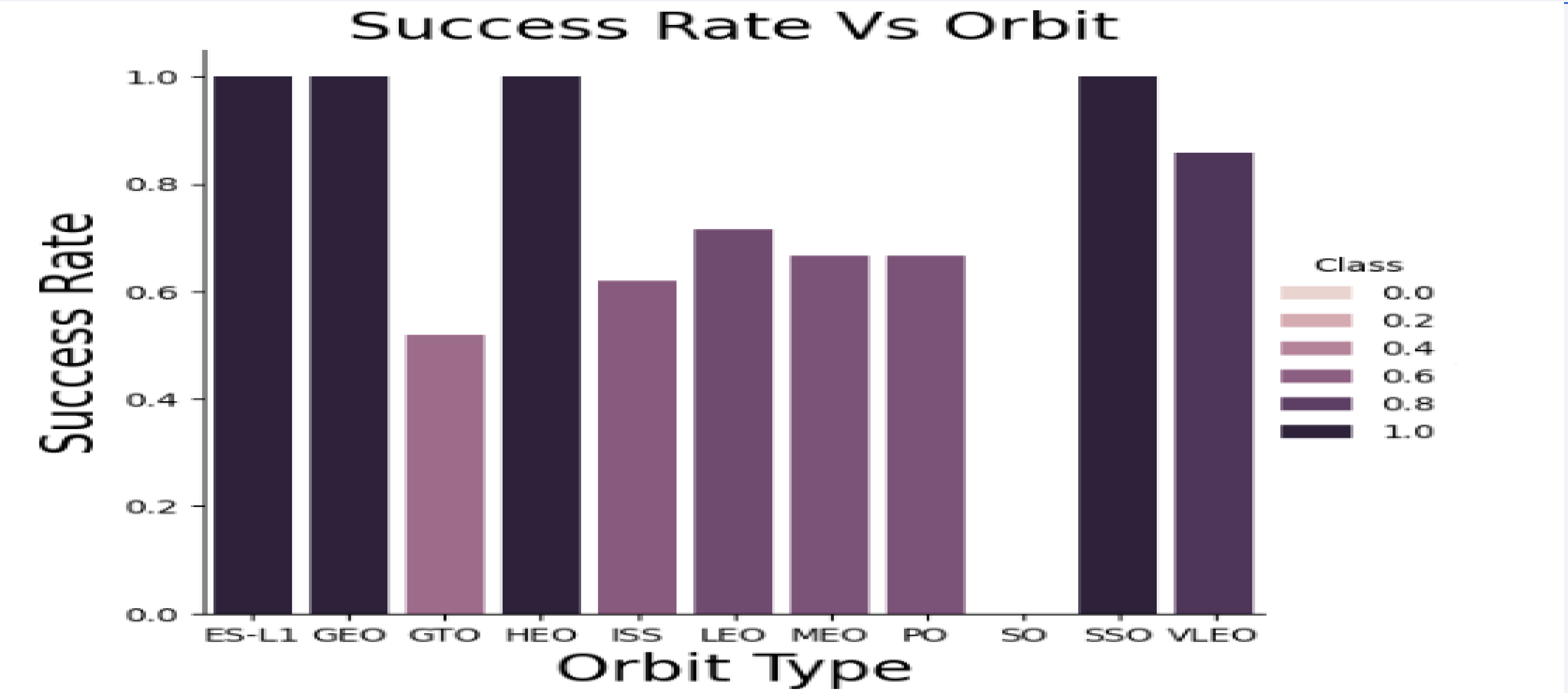
Launch Site –CCAFSSLC 40 use more than other launch sites and VAFB SLC 4E launch site is used for few launches.

Payload vs. Launch Site



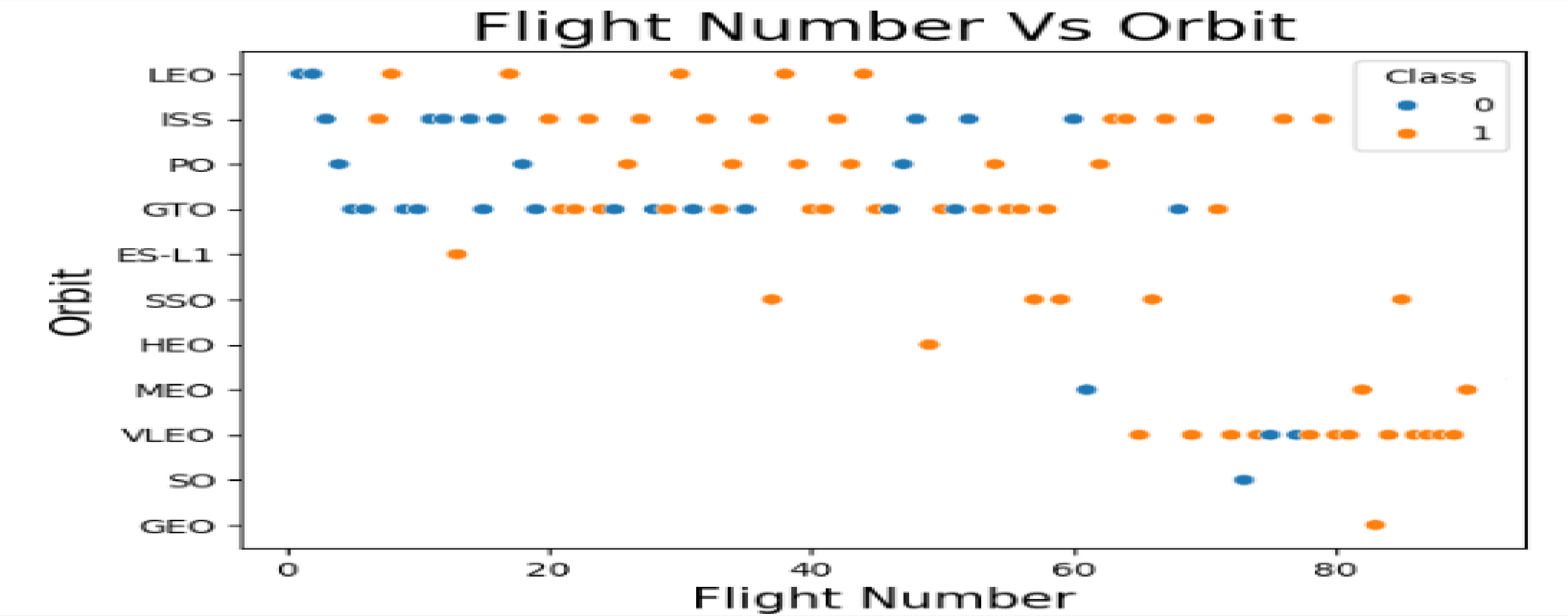
From CCAFS SLC 40, launches more payloads with low mass.

Success Rate vs. Orbit Type



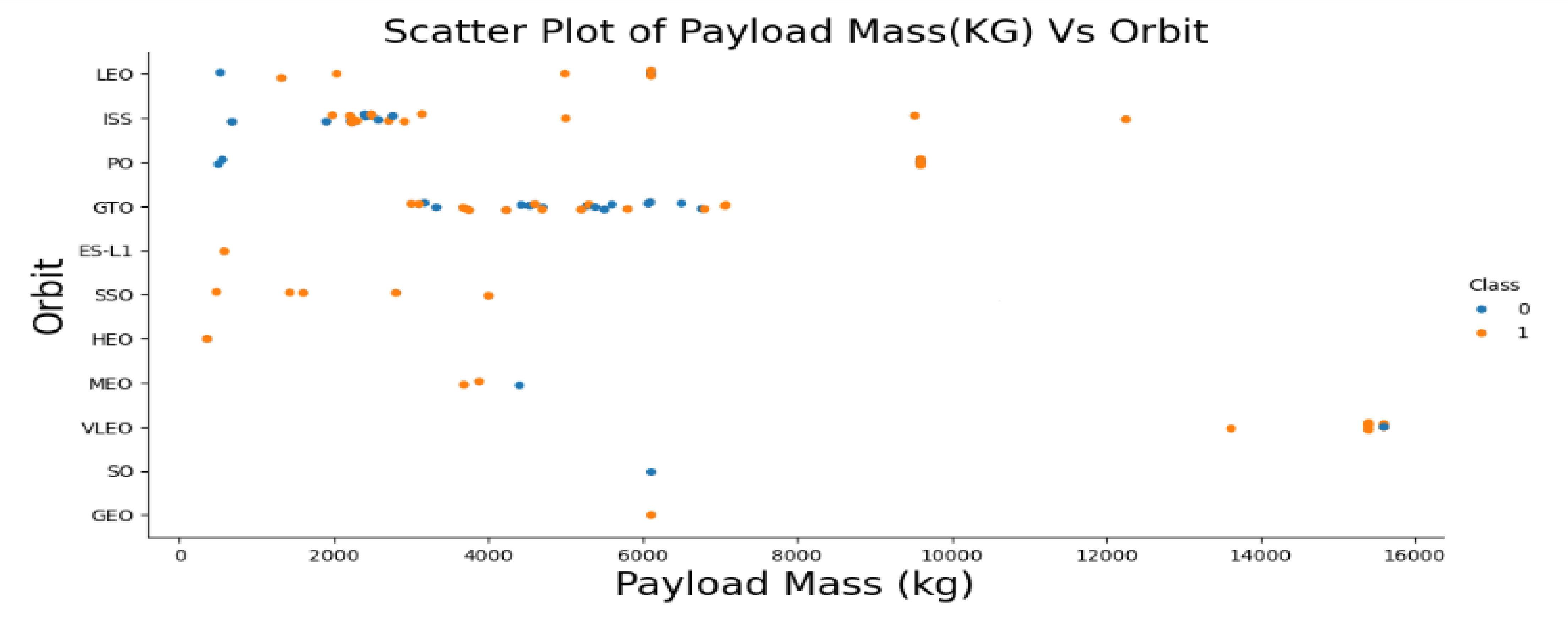
ESL1,GEO,HEO & SSO is highest success rate that all.

Flight Number vs. Orbit Type



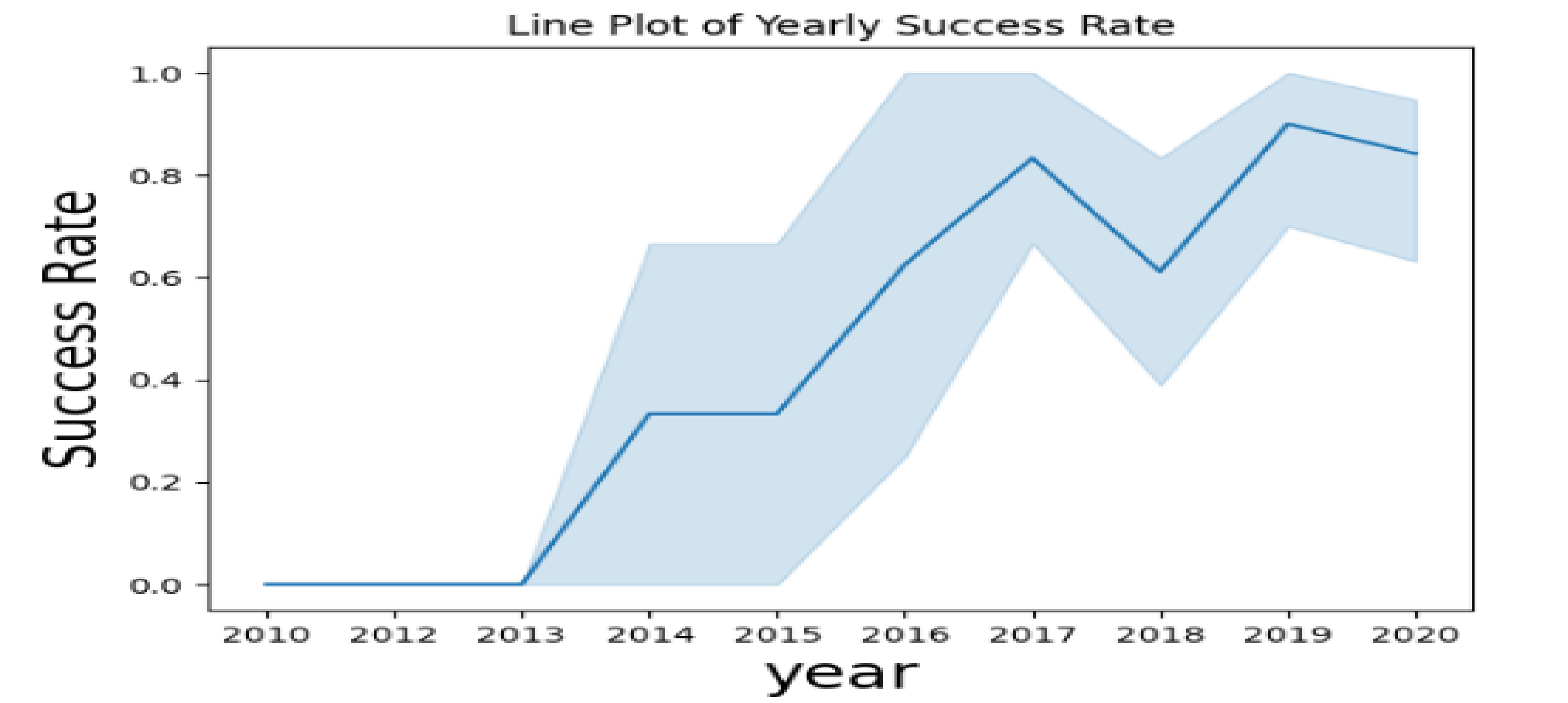
LEO shifted their launches to VLEO recent year and where ES-L1 launches only one.

Payload vs. Orbit Type



Maximum launches in ISS with 2000a and in GTO with range of mass 4000 to 8000.

Launch Success Yearly Trend



- More rate of success launch in 2019 and from 2013 success rate increase till 2019.

All Launch Site Names

```
%sql SELECT distinct(LAUNCH_SITE) FROM SPACEXTBL;
```

```
 * sqlite:///my_data1.db
Done.
```

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

Launch Site Names Begin with 'CCA'

```
%sql SELECT * \
    FROM SPACEXTBL \
    WHERE LAUNCH_SITE LIKE'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
Done.
```

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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)';
```

```
 * sqlite:///my_data1.db
Done.
```

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) \
FROM SPACEXTBL \
WHERE Booster_Version = 'F9 v1.1';
```

```
 * sqlite:///my_data1.db
Done.
```

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

```
%sql SELECT MIN(DATE) \
FROM SPACEXTBL \
WHERE LANDING_OUTCOME = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

MIN(DATE)
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_  BETWEEN 4000 AND 6000;
```

```
 * sqlite:///my_data1.db
Done.
```

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 MISSION_OUTCOME,COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER \
FROM SPACEXTBL \
GROUP BY MISSION_OUTCOME;
```

```
 * sqlite:///my_data1.db
Done.
```

Mission_Outcome	TOTAL_NUMBER
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

```
%sql SELECT BOOSTER_VERSION \
FROM SPACEXTBL \
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

```
 * sqlite:///my_data1.db
Done.
```

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 SUBSTR(DATE,6,2) AS MONTH,BOOSTER_VERSION,LAUNCH_SITE, LANDING_OUTCOME \
FROM SPACEXTBL \
WHERE LANDING_OUTCOME = 'Failure (drone ship)' AND SUBSTR(DATE,0,5)='2015';
```

```
 * sqlite:///my_data1.db
Done.
```

MONTH	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

```
%sql SELECT LANDING_OUTCOME,COUNT(*) AS COUNT_OUTCOMES \
FROM SPACEXTBL \
WHERE DATE BETWEEN '2010-06-04' and '2017-03-20' GROUP BY LANDING_OUTCOME ORDER BY COUNT_OUTCOMES DESC;
```

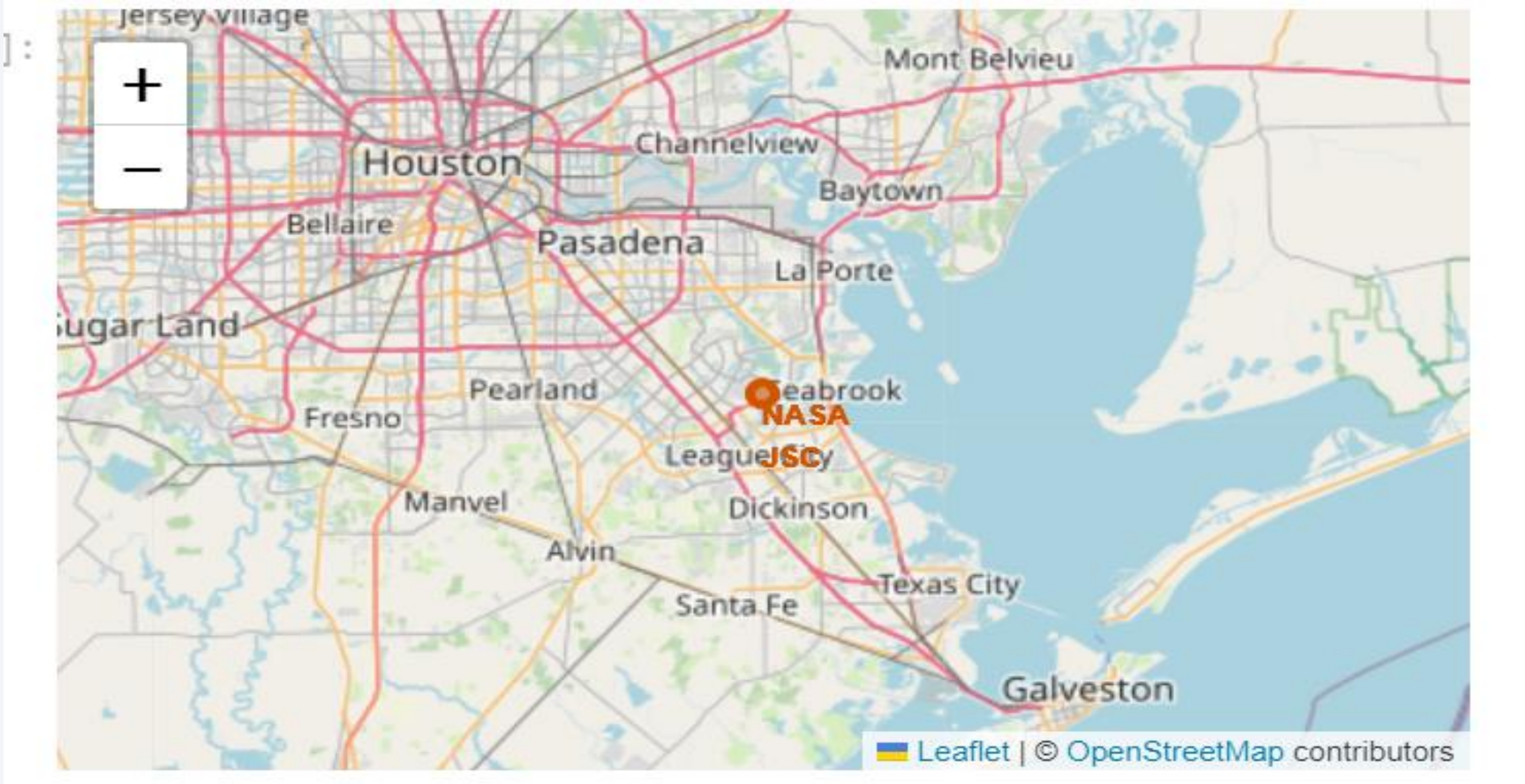
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	COUNT_OUTCOMES
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

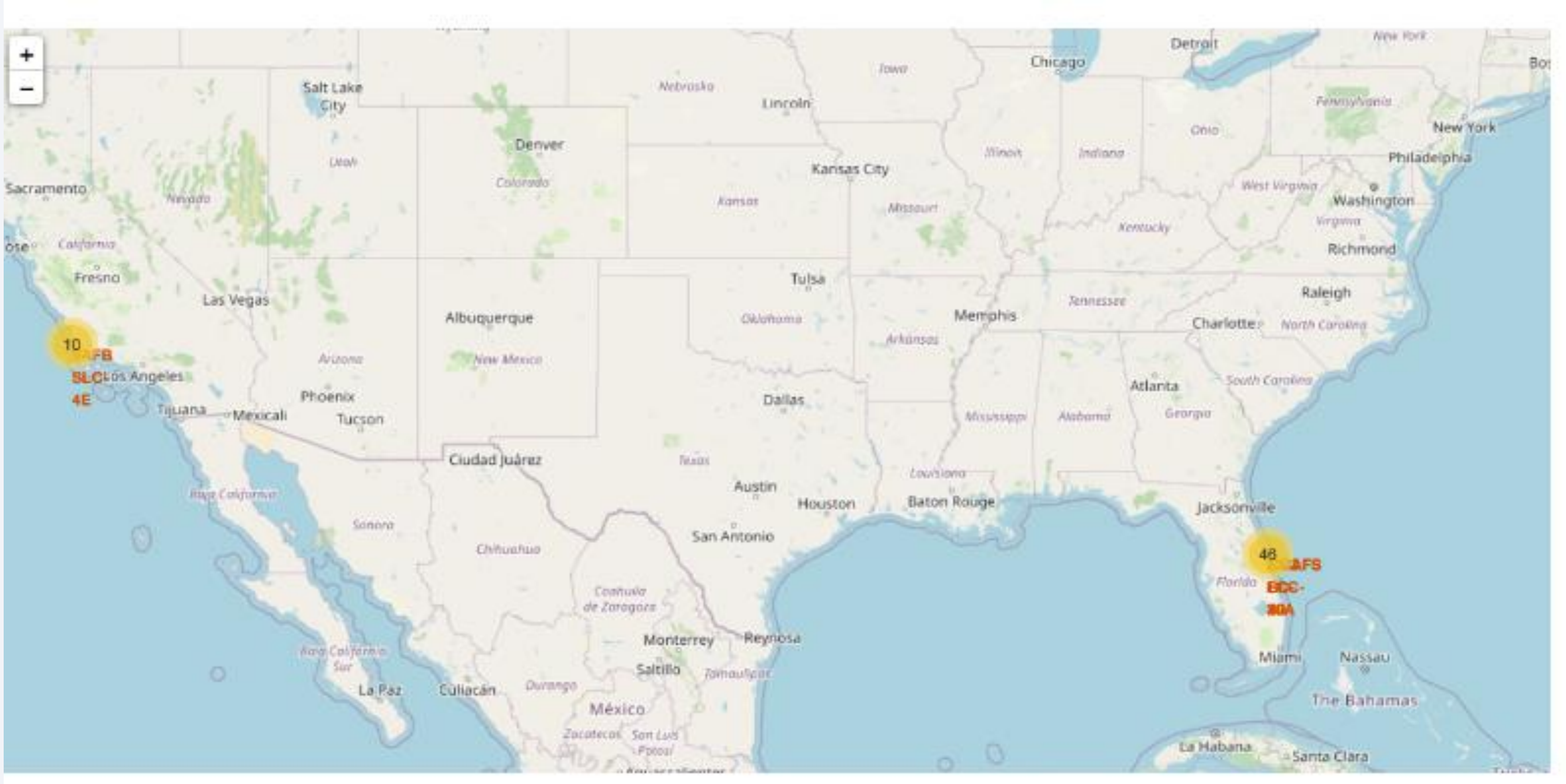
Section 3

Launch Sites Proximities Analysis

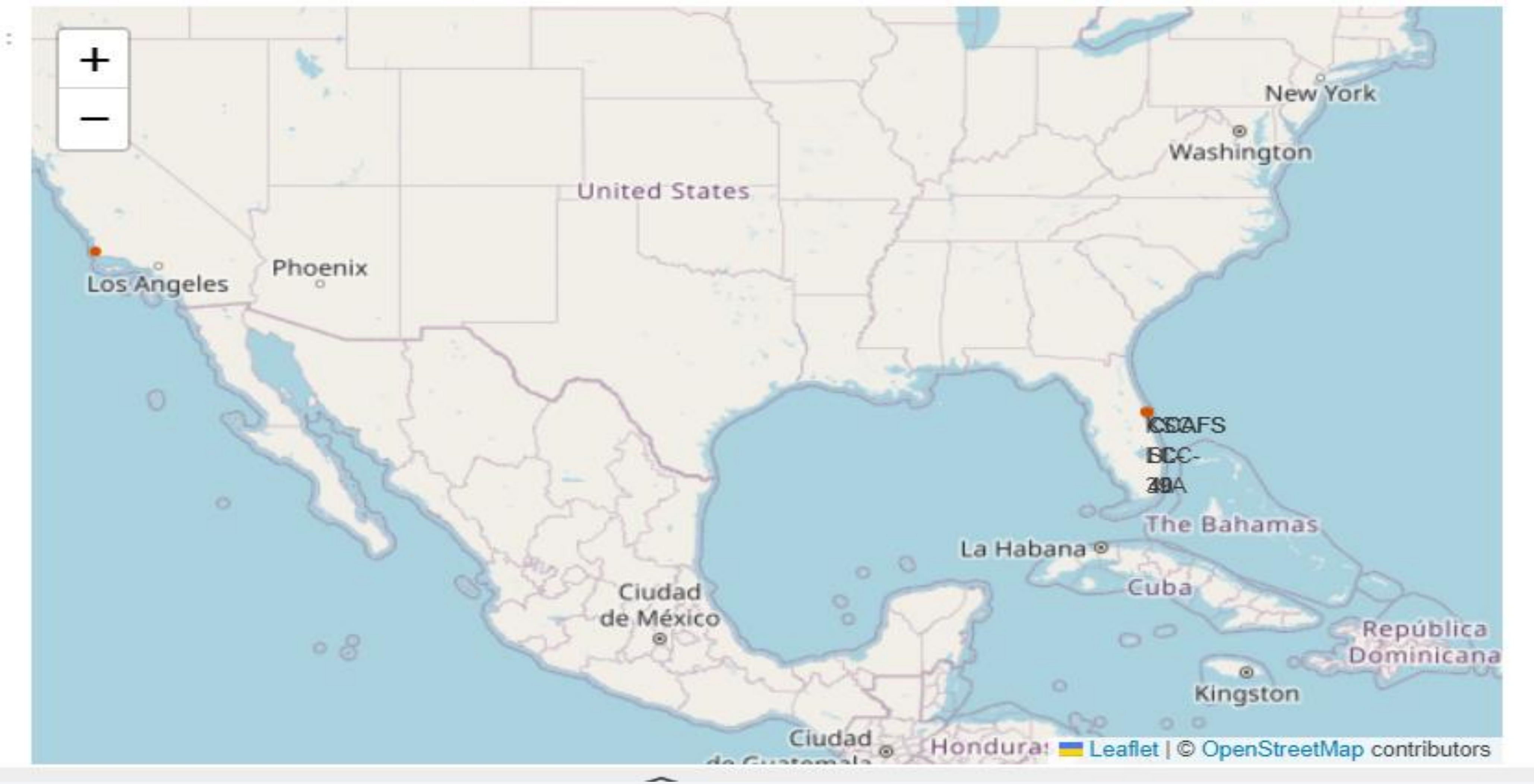
Mark all Launch Sites on Global Map



Mark the success/failed launches on the map



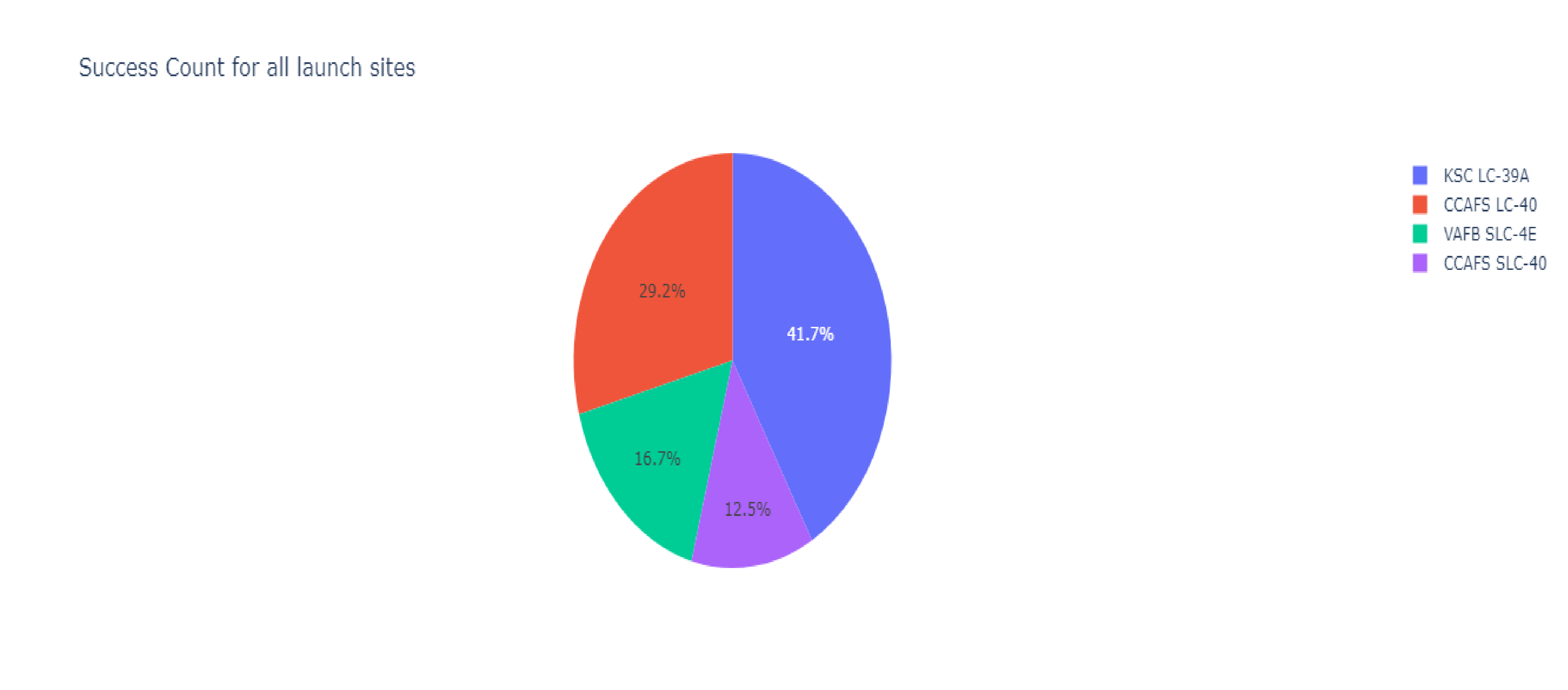
Calculate Distance Between a Launch Site to its Proximities.



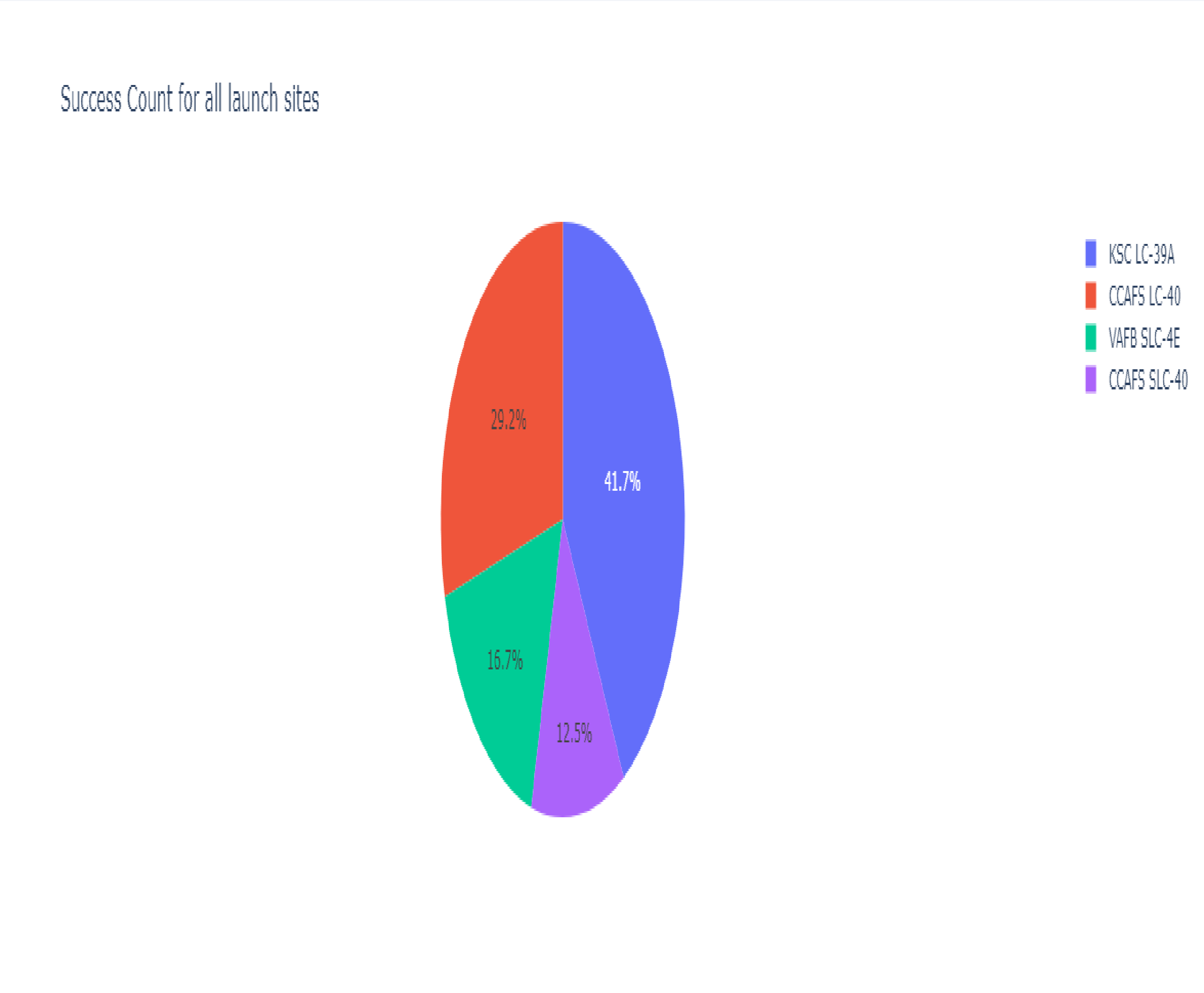
Section 4

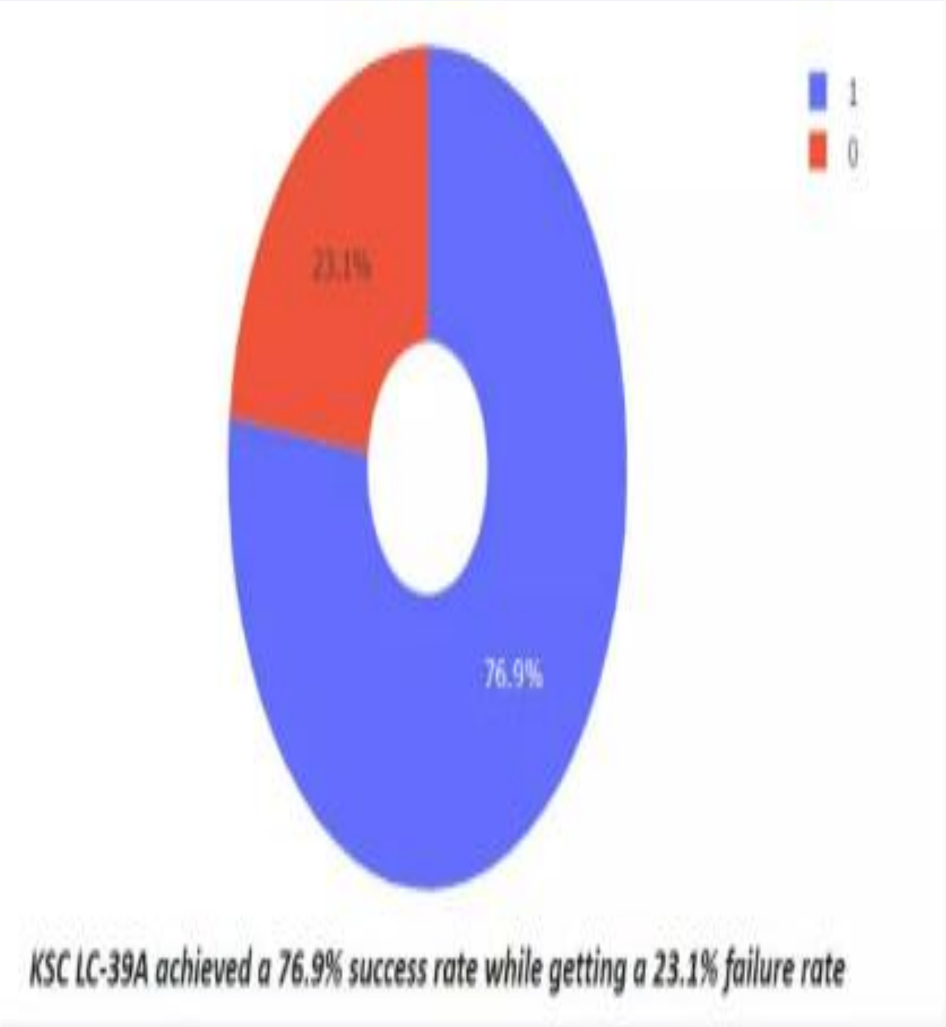
Build a Dashboard with Plotly Dash

Success count for all launch sites

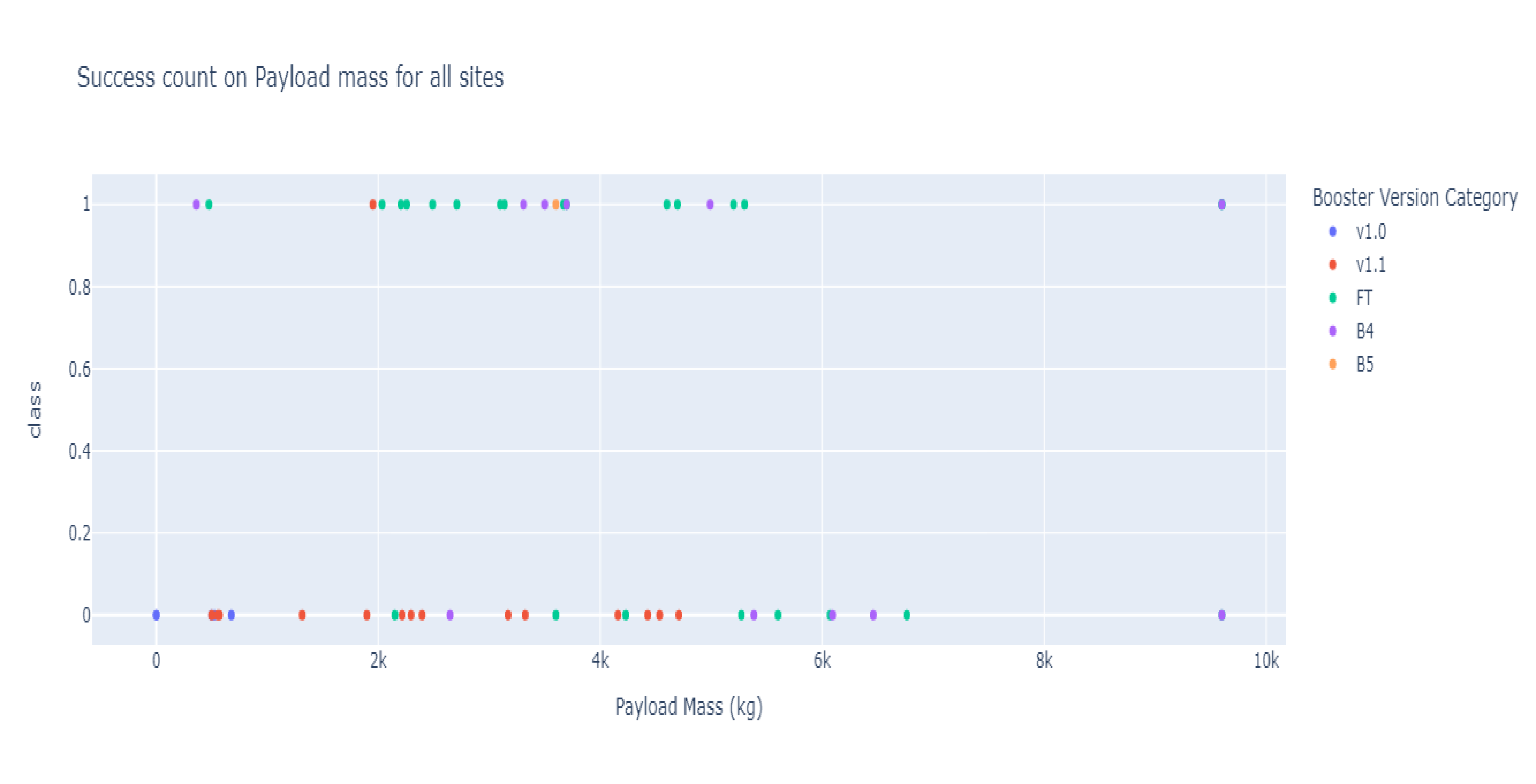


Highest Success Rate for the Launch Site.





Success Count on payload mass for all launch sites



Section 5

Predictive Analysis (Classification)

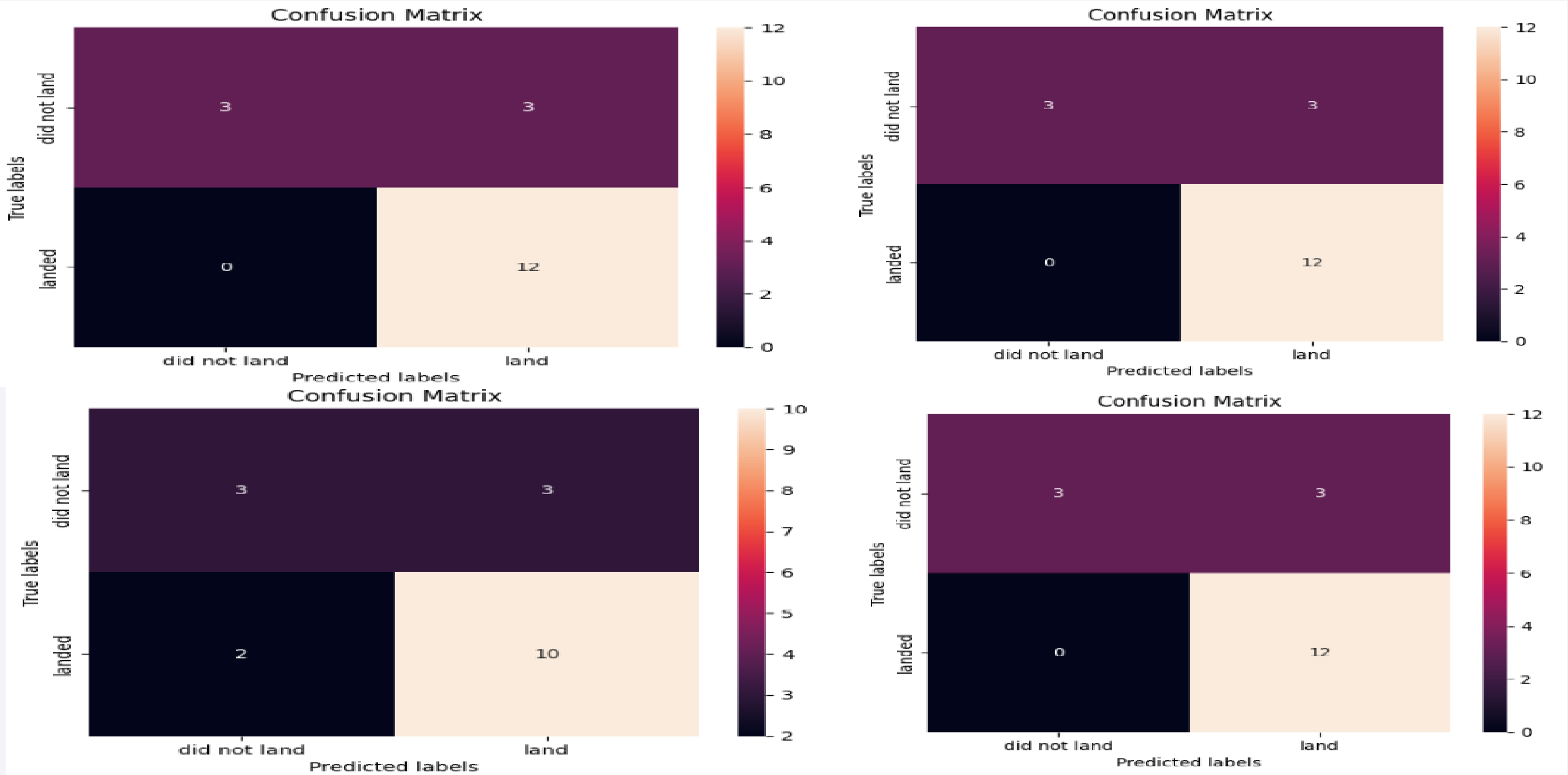
Classification Accuracy

Find the method performs best

```python
print('Accuracy of Logistic Regression Method:',logreg_cv.score(X_test, Y_test))
print('Accuracy of Support Vector Mechine Method:',svm_cv.score(X_test, Y_test))
print('Accuracy of Decision Tree Method:',tree_cv.score(X_test, Y_test))
print('Accuracy of K Nearsdt neighbors  Method:',knn_cv.score(X_test, Y_test))
```

```
Accuracy of Logistic Regression Method: 0.8333333333333334
Accuracy of Support Vector Mechine Method: 0.8333333333333334
Accuracy of Decision Tree Method: 0.7222222222222222
Accuracy of K Nearsdt neighbors  Method: 0.8333333333333334
```

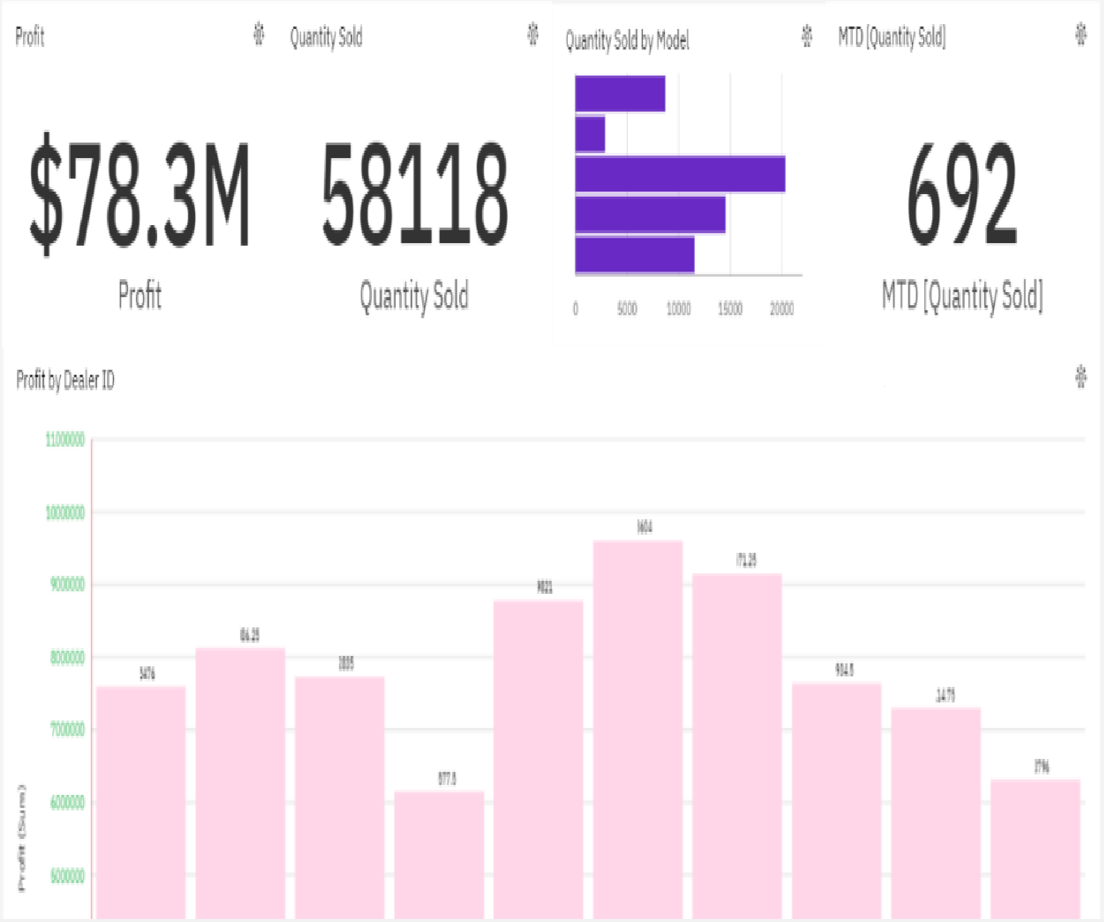
Confusion Matrix

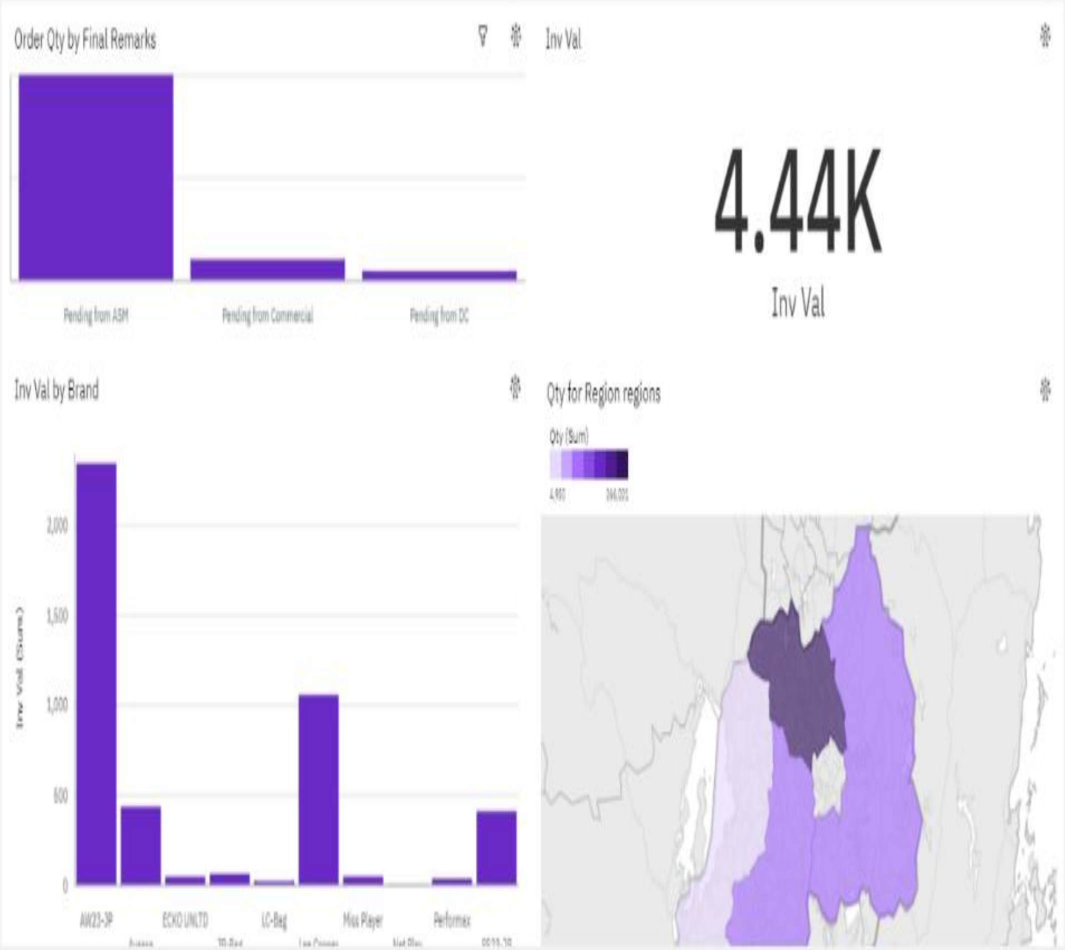


Conclusions

- The SVM,KNN and Logistic Regression models are the best accuracy(83.33%) of the dataset.
- Low weighted payloads are perform well then more weighted payloads.
- SpaceX Launch success rate is directly proportional time in years. Eventually it will perfect launch.
- KSC LC-39A site is most success than other sites.
- GEO,HEO,SSO,ES L1 has best Success Rate.

Appendix


Profit
Quantity Sold
Quantity Sold by Model
MTD [Quantity Sold]
$78.3M
Profit
58118
Quantity Sold
0
5000
10000
15000
20000
692
MTD [Quantity Sold]
Profit by Dealer ID
11000000
10000000
9000000
8000000
7000000
6000000
5000000
Profit (Sum)


Order Qty by Final Remarks
Pending from ASM
Pending from Commercial
Pending from DC
Inv Val
4.44K
Inv Val
Inv Val by Brand
2,000
1,500
1,000
500
0
Inv Val (Sum)
ECKO UNLTD
Performax
Qty for Region regions
Qty (Sum)

Thank you!