IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mehrnoush Gh.Ghonchehnazi

Dec 2023

Outline

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

Executive Summary

➢ Summary of methodologies

- Collecting Data (Request from SpaceX API) by Webscraping
- Wrangling Data
- Exploratory Data Analysis Using Data Visualization and SQL
- Building an interactive map with Folium
- Building a Dashboard with Plotly Dash
- Classification Predictive Analysis

➢ Summary of all result

EDA Results

ML Results

Introduction

➢ Project background and context

SpaceX promotes that the Falcon 9 rocket can be launched for $62 million, much less than the $165 million estimated by other providers. The uniqueness of SpaceX is in reusing the rocket upon a successful first-stage landing.

➢ Problems you want to find answers

The project goal is to predict the successful landing of the first stage of the SpaceX Falcon 9 rocket. To achieve this, we plan to develop a machine learning model using observational data from Falcon 9 launches.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Request to the SpaceX's API Website; "https://api.spacexdata.com/v4"
  - Webscrping data : Extract a Falcon 9 launch records HTML table from Wikipedia
  - Perform data wrangling
  - Identify and calculate the percentage of the missing values in each attribute
  - Replacing missing numerical data with the mean value of parameter
  - Using One Hot Encoding to transform categorical variables to numerical ones.

Methodology

Executive Summary

- Perform exploratory data analysis (EDA) using visualization and SQL
  - Visualizing data through categorical, bar, and scatter plots to define relationships between parameters
  - Using SQL queries to determine the data analysis process
- Perform interactive visual analytics using Folium and Plotly Dash
  - Interactive Plotly Dashboard to visualize payload and success launch data
  - Using interactive Folium Maps to explore Launch Sites
- Perform predictive analysis using classification models
  - Standardizing parameters
  - Predictive multiple classification (Logistic, SVM, Decision Tree, KNN) model
  - Finding Best performance model

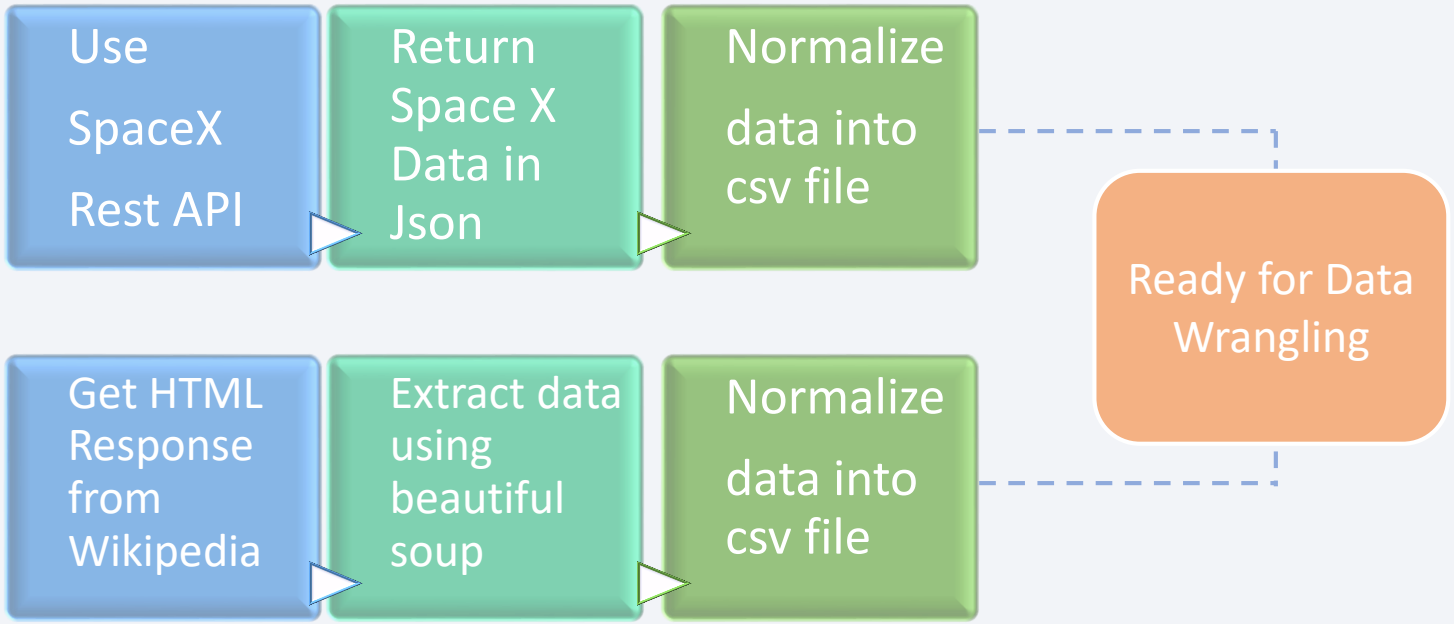
Data Collection

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.

- Using SpaceX REST API to gather data on rocket launches: https://api.spacexdata.com/v4

- Other data source for collecting Falcon 9 Launch data is web scraping Wikipedia using Beautiful Soup.



Data Collection – SpaceX API

1. Request Data From SpaceX API

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

response = requests.get(spacex_url)
```

2. Transform the Json File to a Pandas DataFrame

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

3. Extract required data and make lists

```
# Takes the dataset and uses the rocket column to call the API and append the data to the list
def getBoosterVersion(data):
    for x in data['rocket']:
       if x:
        response = requests.get("https://api.spacexdata.com/v4/rockets/"+str(x)).json()
        BoosterVersion.append(response['name'])
```

Data Collection – SpaceX API

4. transform lists to Datafram

```
# Create a data from launch_dict
df=pd.DataFrame(launch_dict)
```

5. Keep related Falcon V9 boosters and remove others

```
data_falcon9 =df[df['BoosterVersion']!='Falcon 1']
data_falcon9. info()
```

6. Convert Datafram into CSV file

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

Here's the link to the API Data Collection Notebook

Data Collection - Scraping

1.Getting response from HTML

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

```
page.status_code
```

```
200
```

Status code 200 indicates success

2.Ceating BeautifulSoup Object

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

3. Finding Tables

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

4. Getting column names

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

Here's the link to the Web Scraping notebook

5.Creating Dictionary

```
launch_dict= dict.fromkeys(column_names)

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

# Let's initial the launch_dict with each value to be an empty list
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'] = []
# Added some new columns
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"):
        #check to see if first table heading is as number corresponding to launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
```

7. Create a Dataframe from launch_dict

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

8. Export Dataframe to a CSV

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

Data Wrangling

Missing Values

1. Finding Missing values:

Identify and calculate the percentage of missing values in each attribute

```
df.isnull().sum()/df.shape[0]*100
```

2. Imputing Missing values:

Calculate Mean for Payload Mass and Replace nans for Payload Mass with Mean

```
mean_Mass=data_falcon9['PayloadMass'].mean()
mean_Mass

# Replace the np.nan values with its mean value
```

```
6123.547647058824
```

```
data_falcon9['PayloadMass']=data_falcon9['PayloadMass'].replace(np.nan, mean_Mass)
```

Data Wrangling

Performed Exploratory Data Analysis

1. **Calculating the number of launches on each site**

```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

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

2. **Calculate the number and occurrence of each orbit**

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
GTO        27
ISS        21
VLEO       14
PO          9
LEO         7
SSO         5
MEO         3
ES-L1       1
HEO         1
SO          1
GEO         1
Name: Orbit, dtype: int64
```

3. **Calculating the number and occurrence of mission outcome per orbit type**

```
# landing_outcomes = values on Outcome column
landing_outcomes=df['Outcome'].value_counts()
landing_outcomes
```

```
True ASDS       41
None None       19
True RTLS       14
False ASDS       6
True Ocean       5
False Ocean      2
None ASDS        2
False RTLS       1
Name: Outcome, dtype: int64
```

Data Wrangling

Performed Exploratory Data Analysis

4. Creating landing outcome label from Outcome column

Using the Outcome, created a list where the element was zero if the corresponding row in Outcome was in the set bad_outcome; otherwise, it was one. Then assigned it to the variable landing_class:

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
def  function(item):
    if item in bad_outcomes:
        return 0
    else:
        return 1
landing_class = df["Outcome"].apply(function)
landing_class
```

```
df['Class']=landing_class
df[['Class']].head(10)
```

5. The results were exported to a CSV file

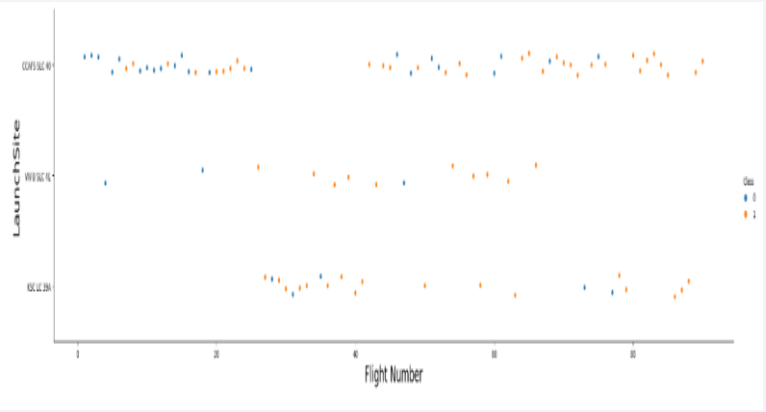
```
df.to_csv("dataset_part_2.csv", index=False)
```

Here's the link to the Data Wrangling notebook

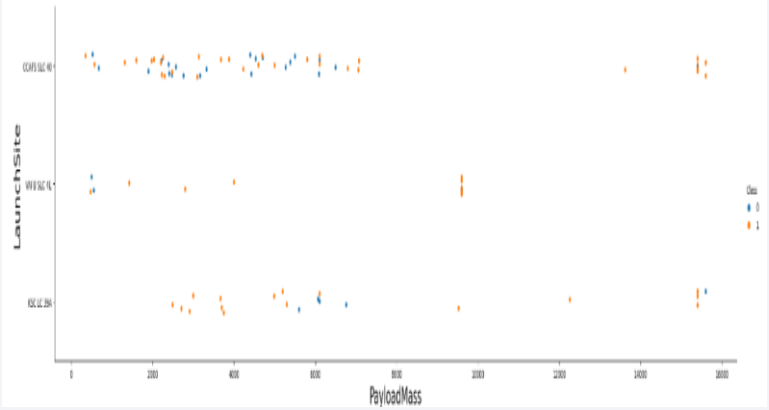
EDA with Data Visualization

Here's the link to the EDA data Visualization notebook

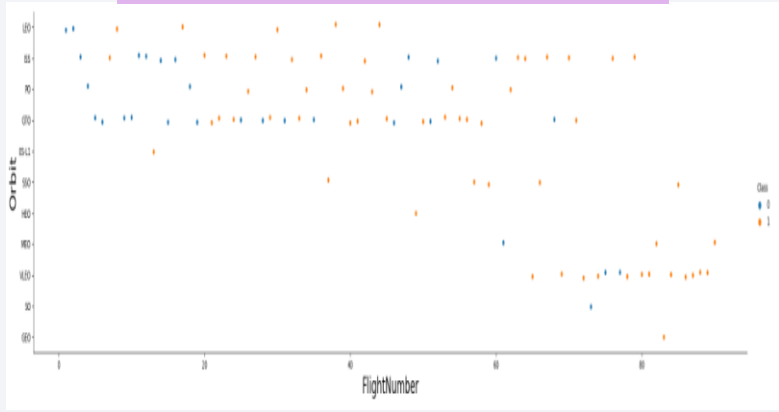
Flight Number Vs. Launch site



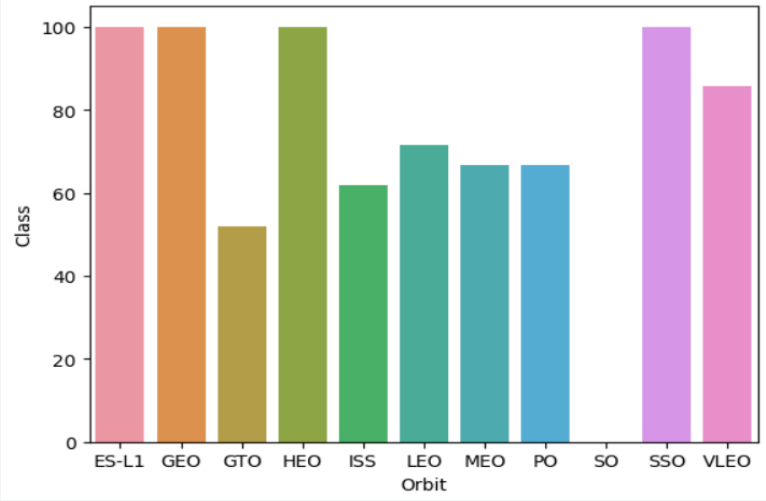
Payload Mass Vs. Launch site



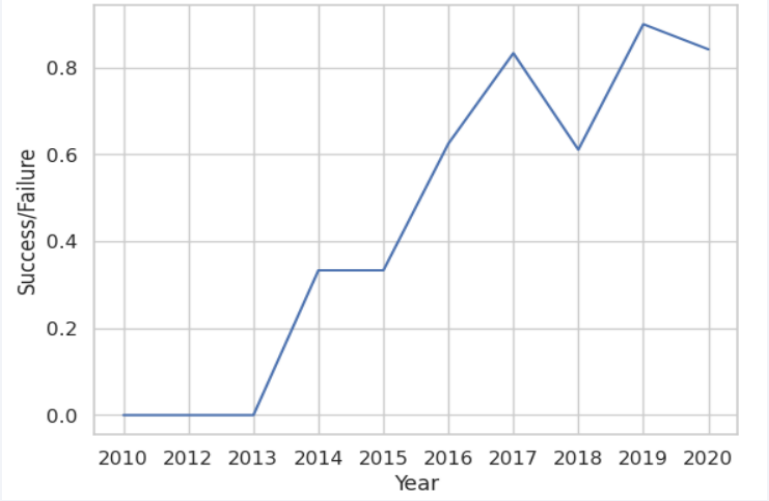
Flight Number Vs. Orbit



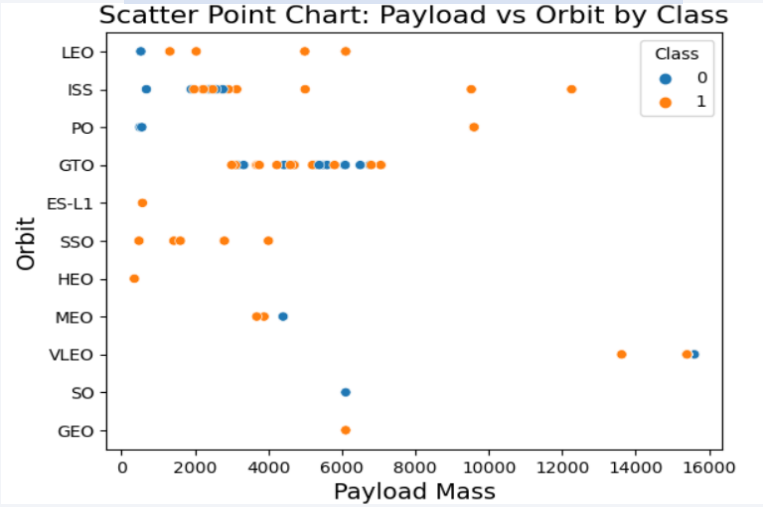
Success Rate by Orbit



Success Rate by Year



Payload Mass Vs. Orbit



ES-L1 , GEO , GTO and SSO have high success rate.

EDA with SQL

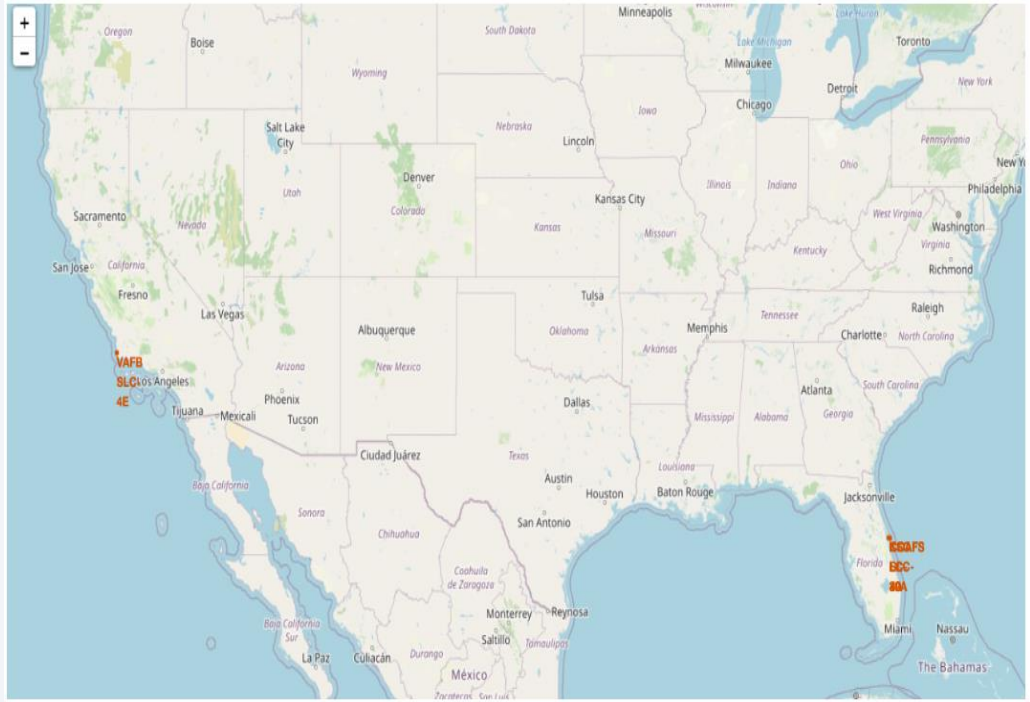
- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'KSC
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date when the first successful landing outcome in ground pad 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 months in year 2015
- Ranking 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.

Here's the link to the SQL notebook

Build an Interactive Map with Folium

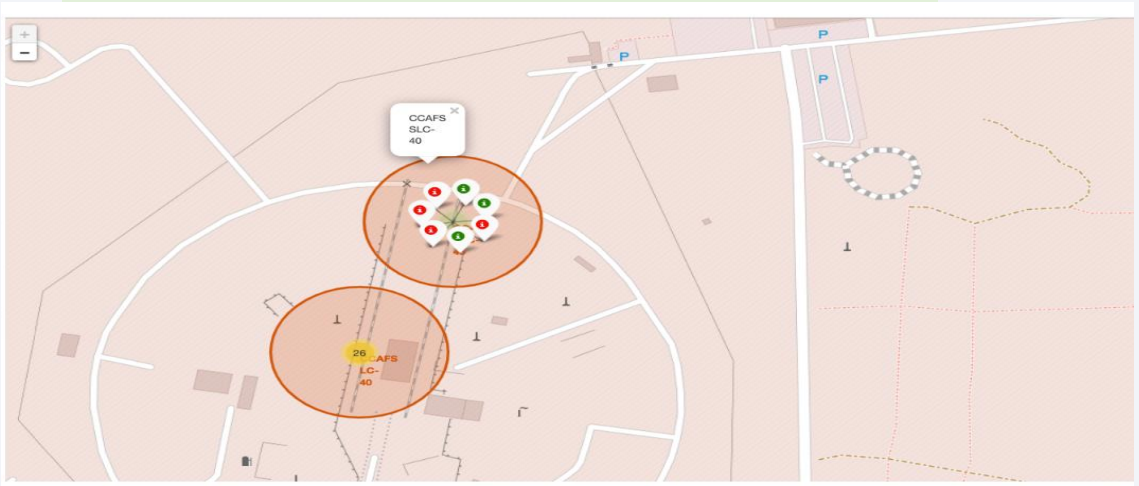
Marked all launch sites on a map

The generated map with marked launch sites should look similar to the following:



[Here's the link to the Interactive Map notebook]

Marked the success/failed launches for each site on the map

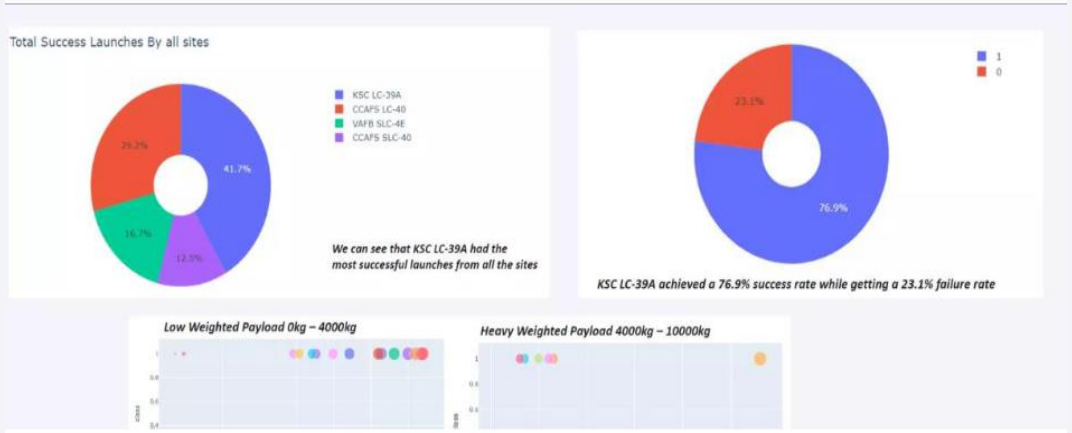


Calculated the distances between a launch site to its proximities

Your updated map with distance line should look like the following screenshot:

Build a Dashboard with Plotly Dash

- Built Plotly Dashboard to make an interactive web app to visualize launch data
- Includes Pie Charts to visualize launch landing success broken down by Launch Site
- If all sites are selected, we get the proportion successful launch landings each site accounts
- If we select an individual site, we see the proportion of all launches at that site which landed successfully
- Includes Scatter Plot of Payload Mass (kg) vs. landing Success Rating (0 for Failure, 1 for Success) color coded by booster version
- Selecting a single site removes points from other sites, Selecting All includes all data
- Plot Payload Mass Range can be selected by a slider for Min and Max

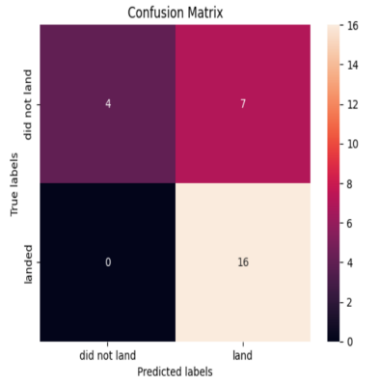


Dashboard with Plotly Dash

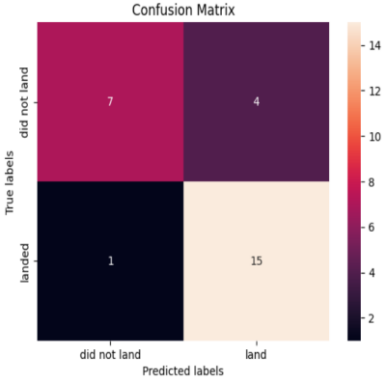
Here's the link to the Payload Dash notebook

Predictive Analysis (Classification)

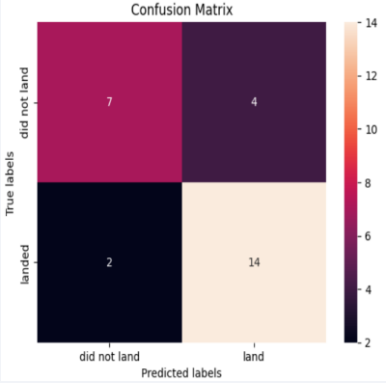
Logistic Regression



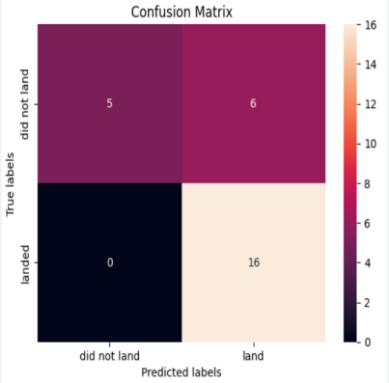
SVM



Decision Tree



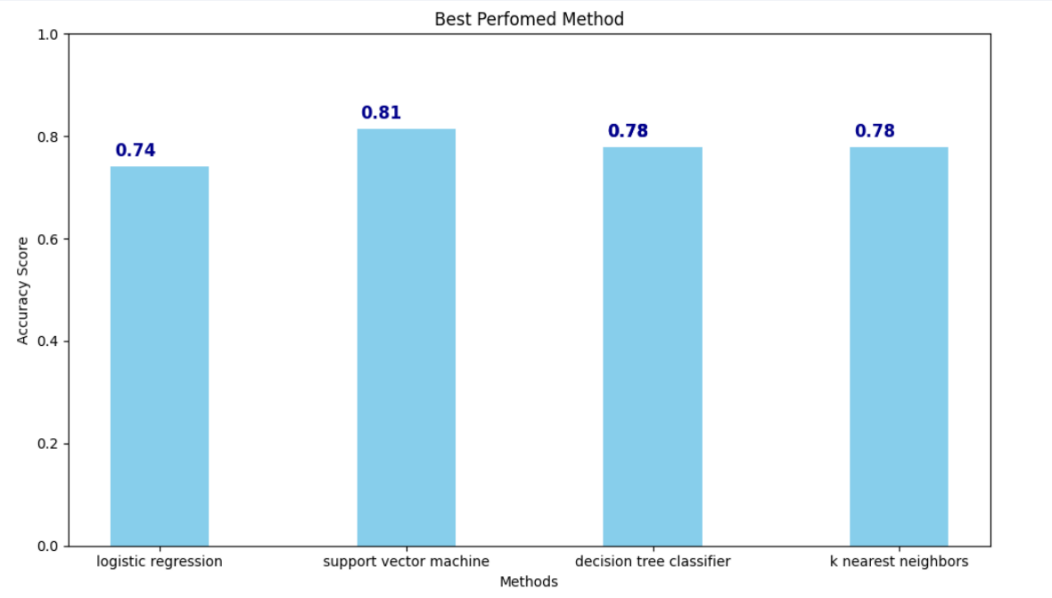
K-Nearest Neighbors



"The Support Vector Machine performed the highest accuracy, reaching 81%."

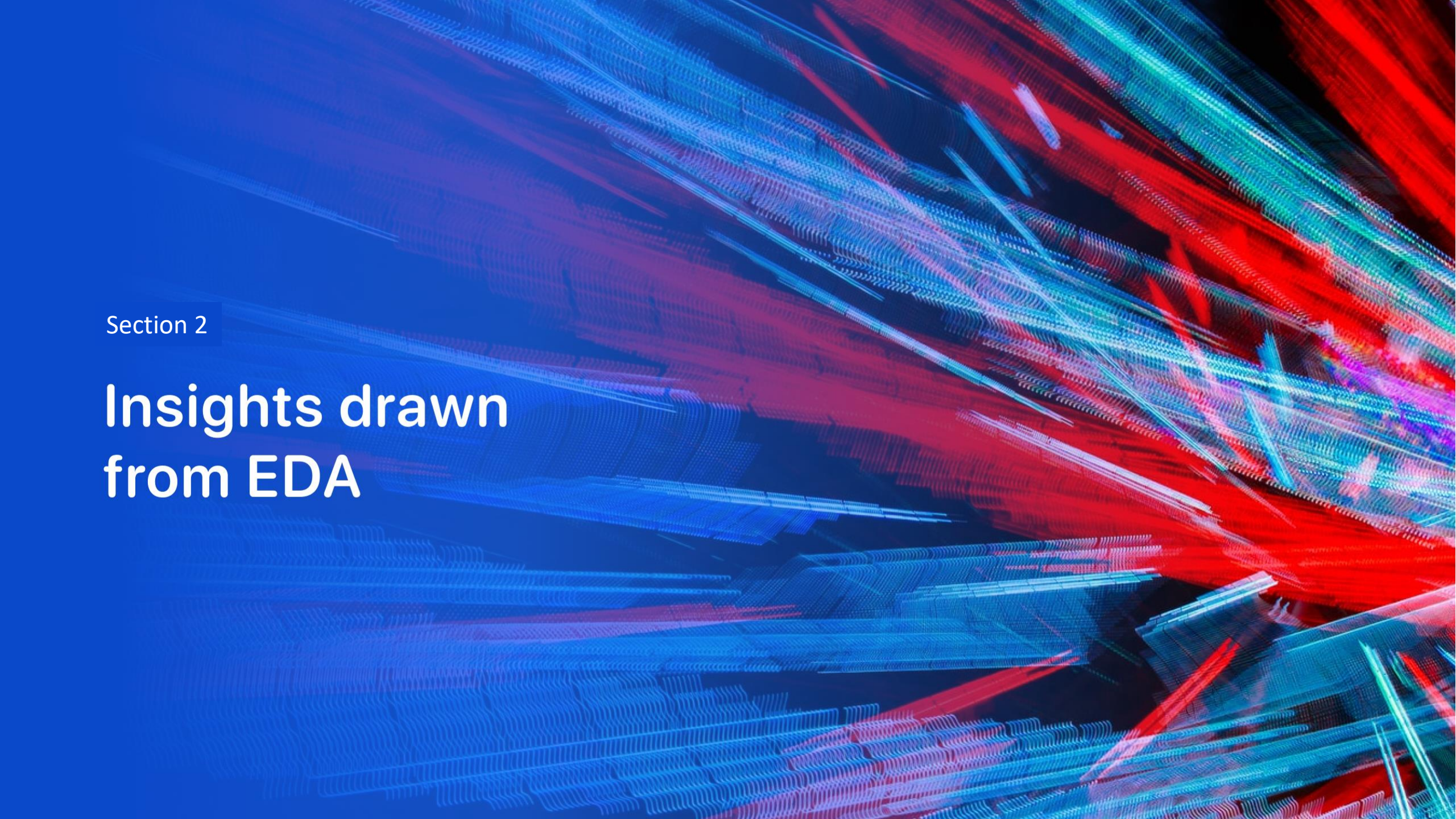
The Decision Tree and KNN predictive models achieved an accuracy of 78%, while the Logistic Regression model had the lowest accuracy score at 74%.

Here's the link to the Predictive Models notebook

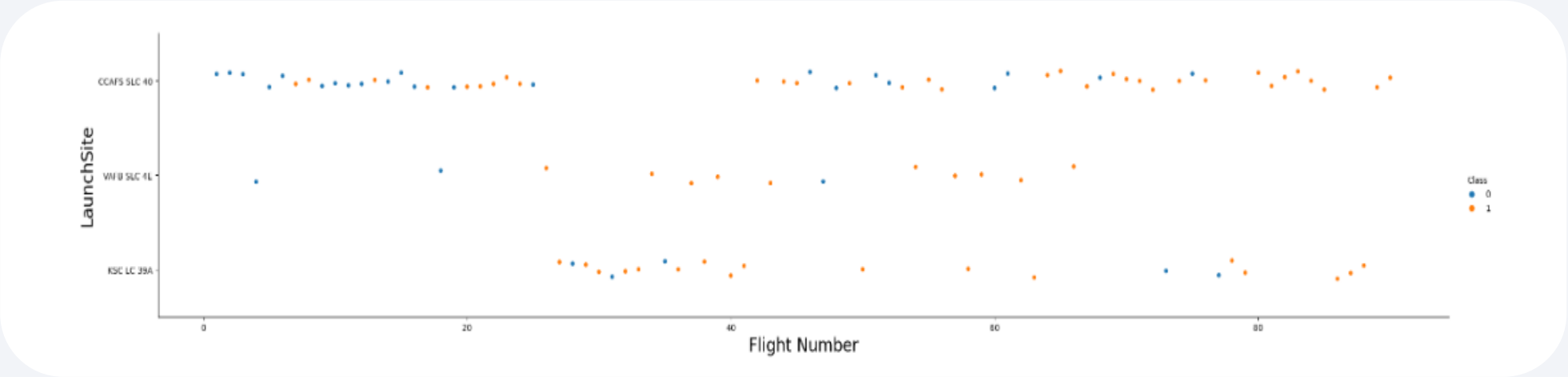


Results

- ✓ Support Vector Machine model are the best in terms of prediction accuracy for this dataset.
- ✓ Low weighted payloads perform better than the heavier payloads.
- ✓ A successful landing on a ground pad was achieved for the first time on December 22, 2015.
- ✓ 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

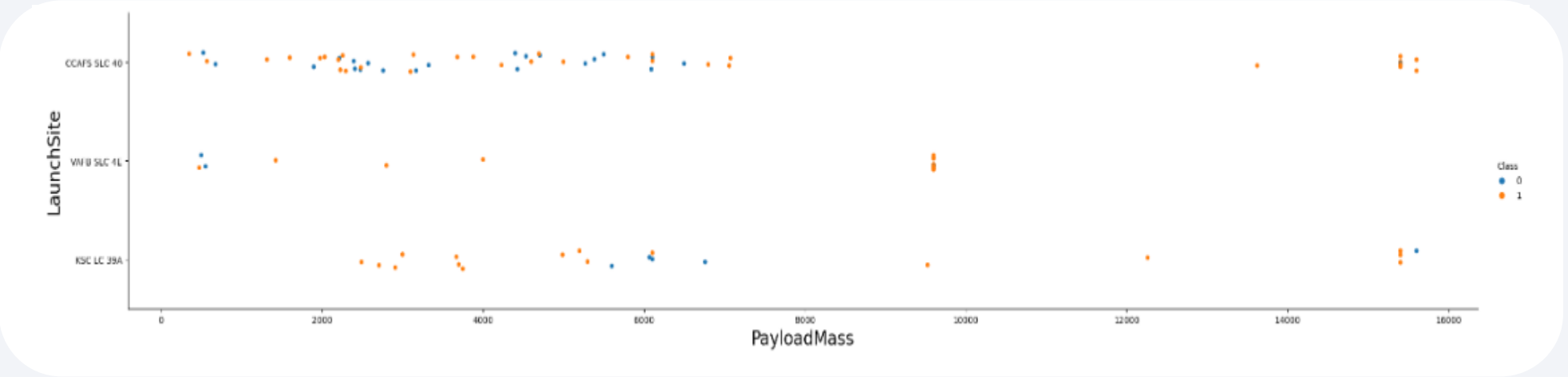
Flight Number vs. Launch Site



- **Launch Site_CCAFS SLC 40:** Flight numbers less than 20 flight numbers were more unsuccessful and Flight number between 40 and 80 were more successful. The figure above indicates that this launch site has seen the highest number of rocket launches in comparison to the other sites.
- **Launch Site_VAFB SLC 4E:** Flight numbers more than 20 landed more successfully.
- **Launch Site_ KSL LC 39A:** Flight numbers between 20 and 40 placed and mostly landed successfully.

We see tat different launch sites have different success rates.
CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.

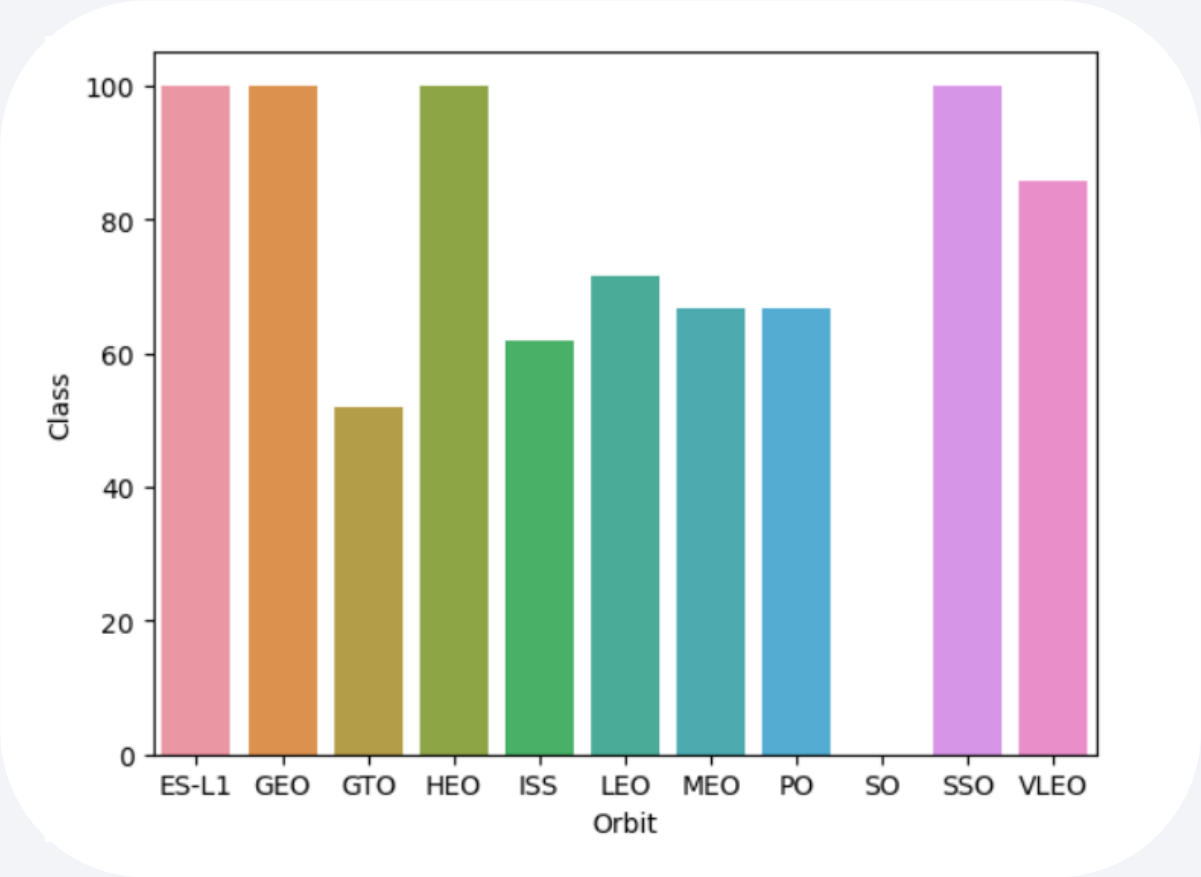
Payload vs. Launch Site



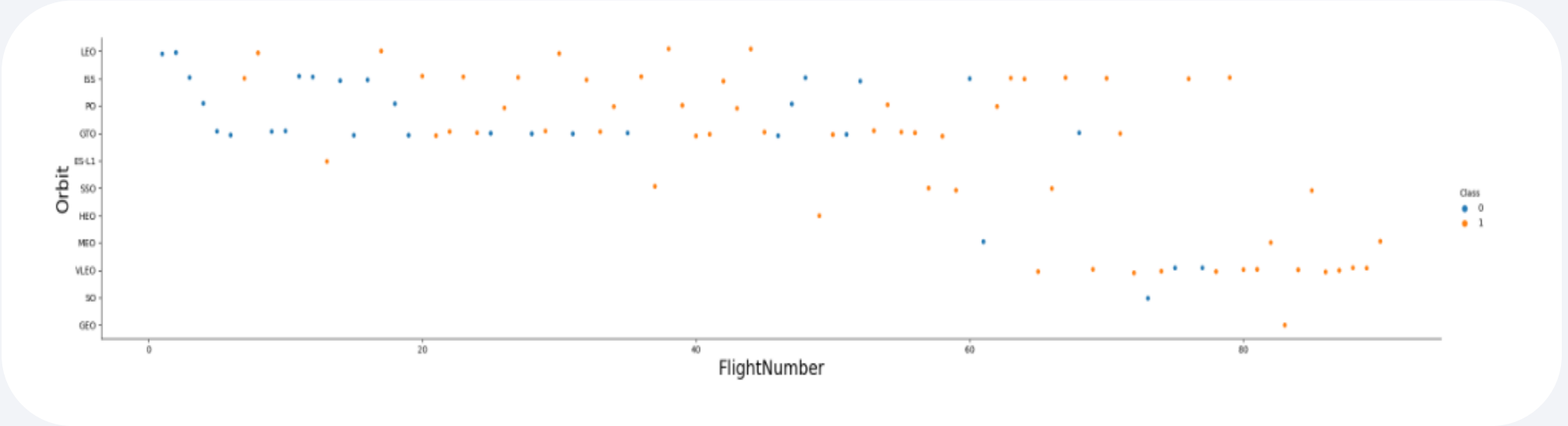
- **We can see that as payload mass increases for Site CCAFS SLC 40, the probability of a successful landing increases.**
- **According to the figure below, it is clear that at the VAFB-SLC 4E launch site there are no rockets launched for heavy payload mass(greater than 10000).**

Success Rate vs. Orbit Type

- **ES-L1, GEO, HEO, and SSO orbits had a perfect success rate of 100%.**
- **VLEO did really well with over 80% success, and LEO also did great with over 70% success.**



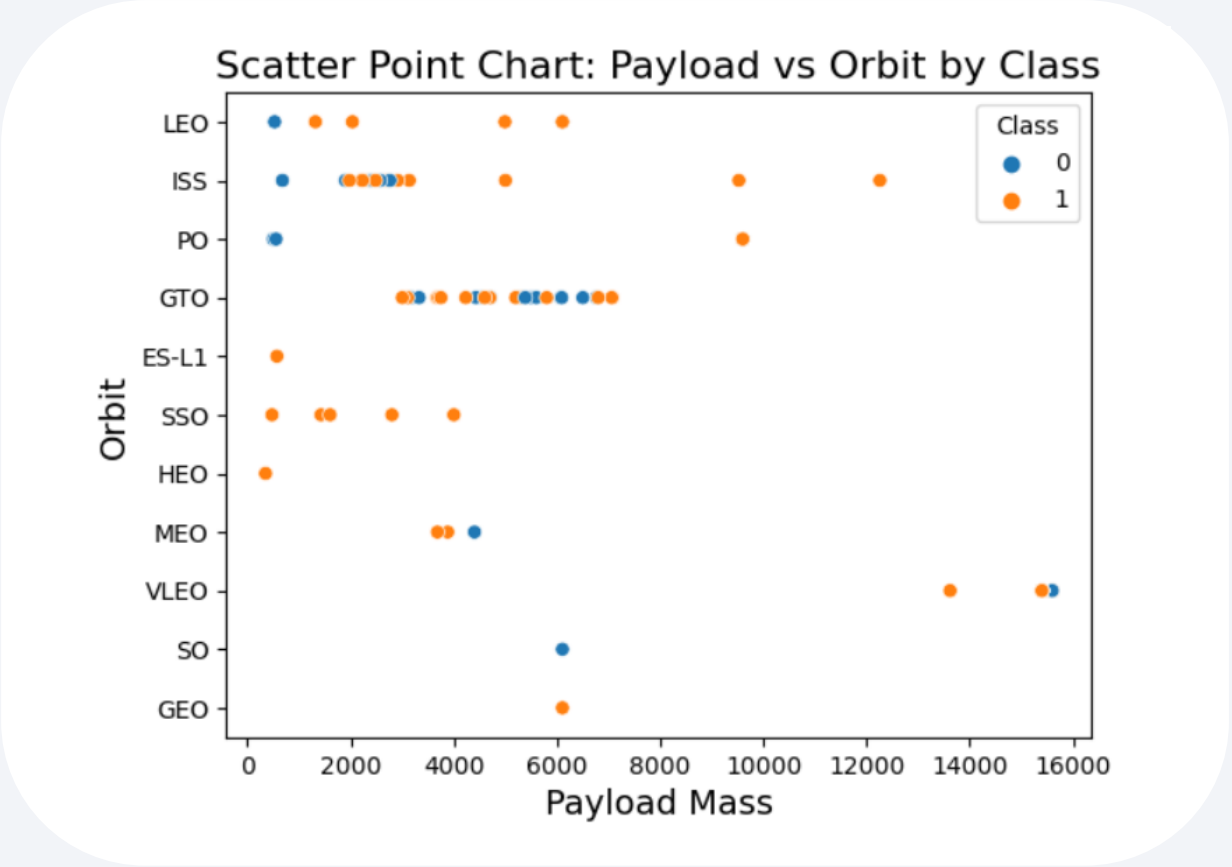
Flight Number vs. Orbit Type



- **SSO, ES-L1, GEO & HEO orbits have 100% success for all flight numbers.**
- **ISS, GTO, LEO, MEO, & VLEO orbits show that the higher flight number has higher chance of success.**

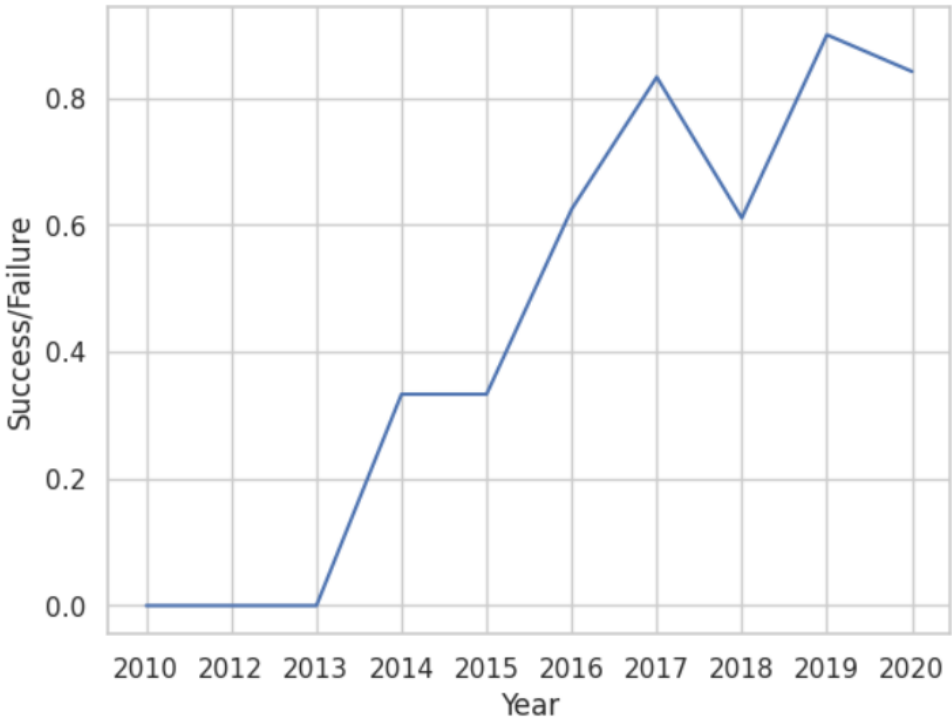
Payload vs. Orbit Type

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

- Success rate was zero up to 2013 and it was increased every year since 2013
- However the figure shows that success rate has declined between 2017-2018 & 2019-2020
- Maximum success rate was 0.9 and it happened on 2019



;

All Launch Site Names

Displaying the names of the unique launch sites in the space mission:

%sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL

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

Using "DISTINCT" keyword was applied to the corresponding column to display Launch Site Names

Launch Site Names Begin with 'CCA'

Displaying 5 records where launch sites begin with the string '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
2010-04-06	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)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	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

Total Payload Mass

Displaying the total payload mass carried by boosters launched by NASA (CRS):

%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)';

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)';
 * sqlite:///my_data1.db
Done.
```

SUM(PAYLOAD_MASS__KG_)
45596

The total payload mass for customers with the name 'NASA (CRS)' was computed using the SUM function.

Average Payload Mass by F9 v1.1

Calculating the average payload mass carried by booster version F9 v1.1:

%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION ='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

The average payload mass carried by booster version F9 v1.1 was calculated using the AVG function.

First Successful Ground Landing Date

Listing the date when the first successful landing outcome in ground pad was achieved:

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

```
%sql SELECT MIN(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME='Success (ground pad)';
 * sqlite:///my_data1.db
Done.
```

MIN(DATE)
2015-12-22

Using SQL query to identify the date of the first successful ground pad landing, which revealed that December 22, 2015, marked the initial successful ground landing.

Successful Drone Ship Landing with Payload between 4000 and 6000

Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000:

%sql SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ <6000

Using the keywords “BETWEEN” and “AND”, the names of boosters that had payload masses greater than 4000kg but less than 6000kg and had successfully landed on a drone ship were displayed, resulting in a total of 4 rockets being shown in the results.

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

Total Number of Successful and Failure Mission Outcomes

Calculate the total number of successful and failure mission outcomes:

%sql SELECT Mission_Outcome, COUNT(*) AS Total_Count FROM SPACEXTBL GROUP BY Mission_Outcome;

The total number of successful and failed missions was counted using the COUNT function, and the results indicate that there were 100 successful missions and 1 failed mission.

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

Boosters Carried Maximum Payload

Listing the names of the booster which have carried the maximum payload mass:

%sql SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);

The boosters that carried the maximum payload were retrieved using a sub-query with the MAX function, and a total of 12 were found in the results.

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

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

1. **%sql SELECT substr(Date,4,2) as month, DATE,BOOSTER_VERSION, LAUNCH_SITE,Landing_Outcome FROM SPACEXTBL where (Landing_Outcome = 'Failure (drone ship)' and substr(Date,7,4)='2015';**
2. **%sql SELECT substr(Date,6,2) AS Month, substr(Date,1,4) AS Year, DATE, BOOSTER_VERSION, LAUNCH_SITE, Landing_Outcome FROM SPACEXTBL where Landing_Outcome = 'Failure (drone ship)'and Year='2015';**

Month	Year	Date	Booster_Version	Launch_Site	Landing_Outcome
10	2015	2015-10-01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

Select Landing_Outcome,Booster Versions, Launch_Site, and Date as requested

Results show that landing outcome of Failure (Drone Ship) and launch date in the year 2015

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

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:

%sql SELECT Date, Landing_Outcome, COUNT(*) AS Count_Landing_Outcome FROM SPACEXTBL WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY Count_Landing_Outcome DESC;

Dates from the 2010-06-04 and 2017-03-20 are gathered and sorted descending in a list called `dates`.

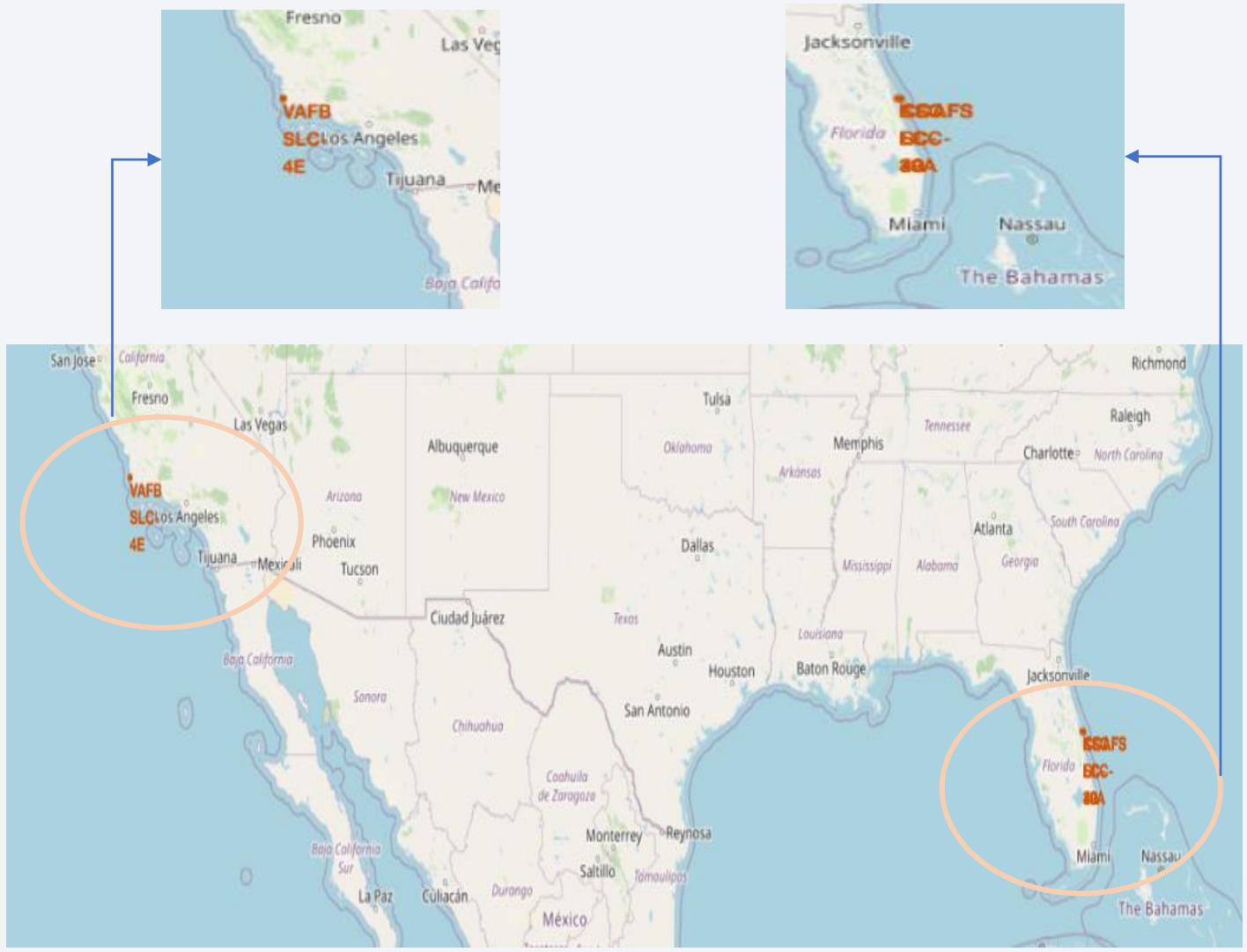
Date	Landing_Outcome	Count_Landing_Outcome
2012-05-22	No attempt	10
2015-12-22	Success (ground pad)	5
2016-08-04	Success (drone ship)	5
2015-10-01	Failure (drone ship)	5
2014-04-18	Controlled (ocean)	3
2013-09-29	Uncontrolled (ocean)	2
2015-06-28	Precluded (drone ship)	1
2010-08-12	Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

Launch Site Locations for Space X Falcon 9

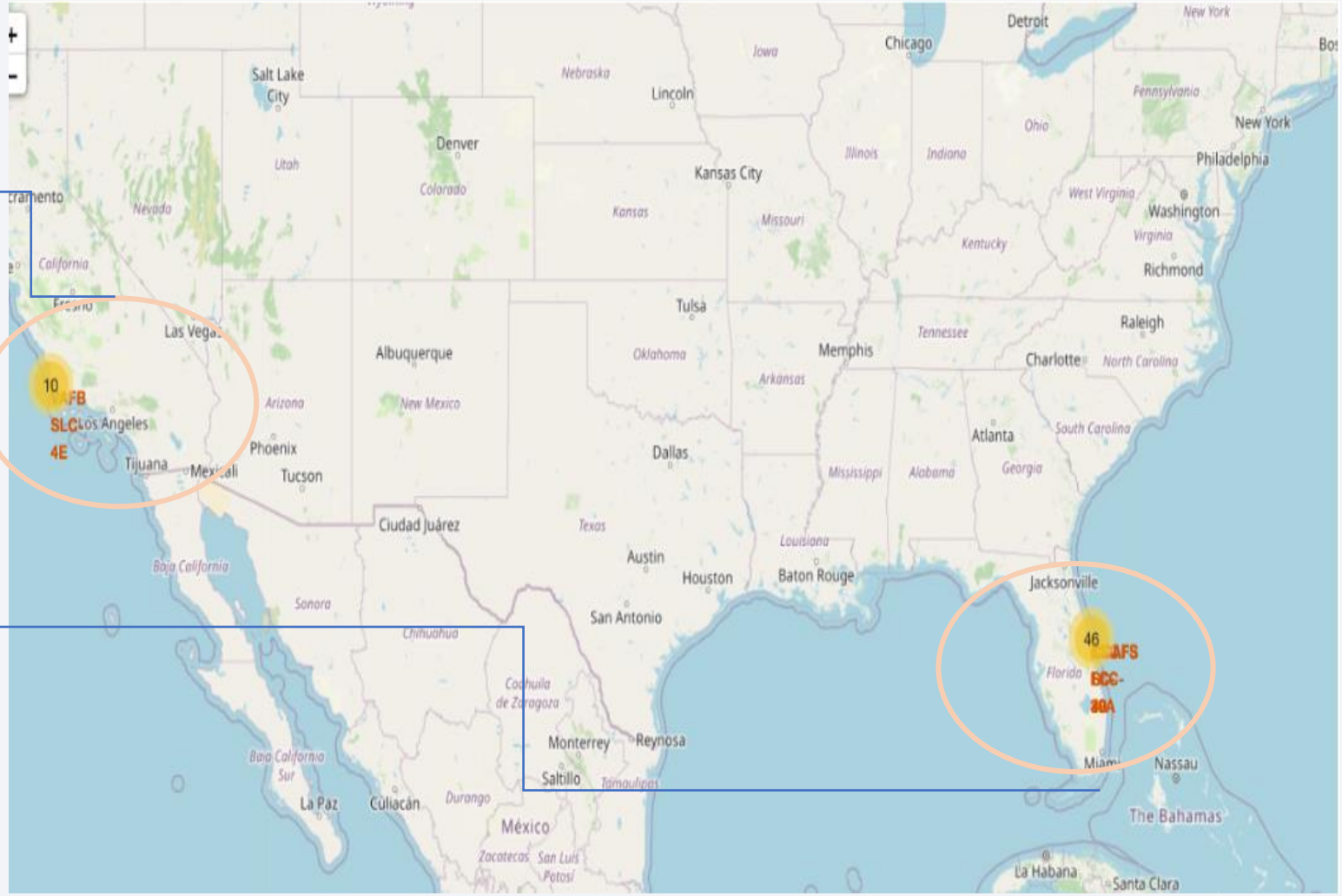
This map illustrate that all launch sites are Lactated on the west coast in California and the east coast in Florida.



Launch Outcomes for Space X Falcon

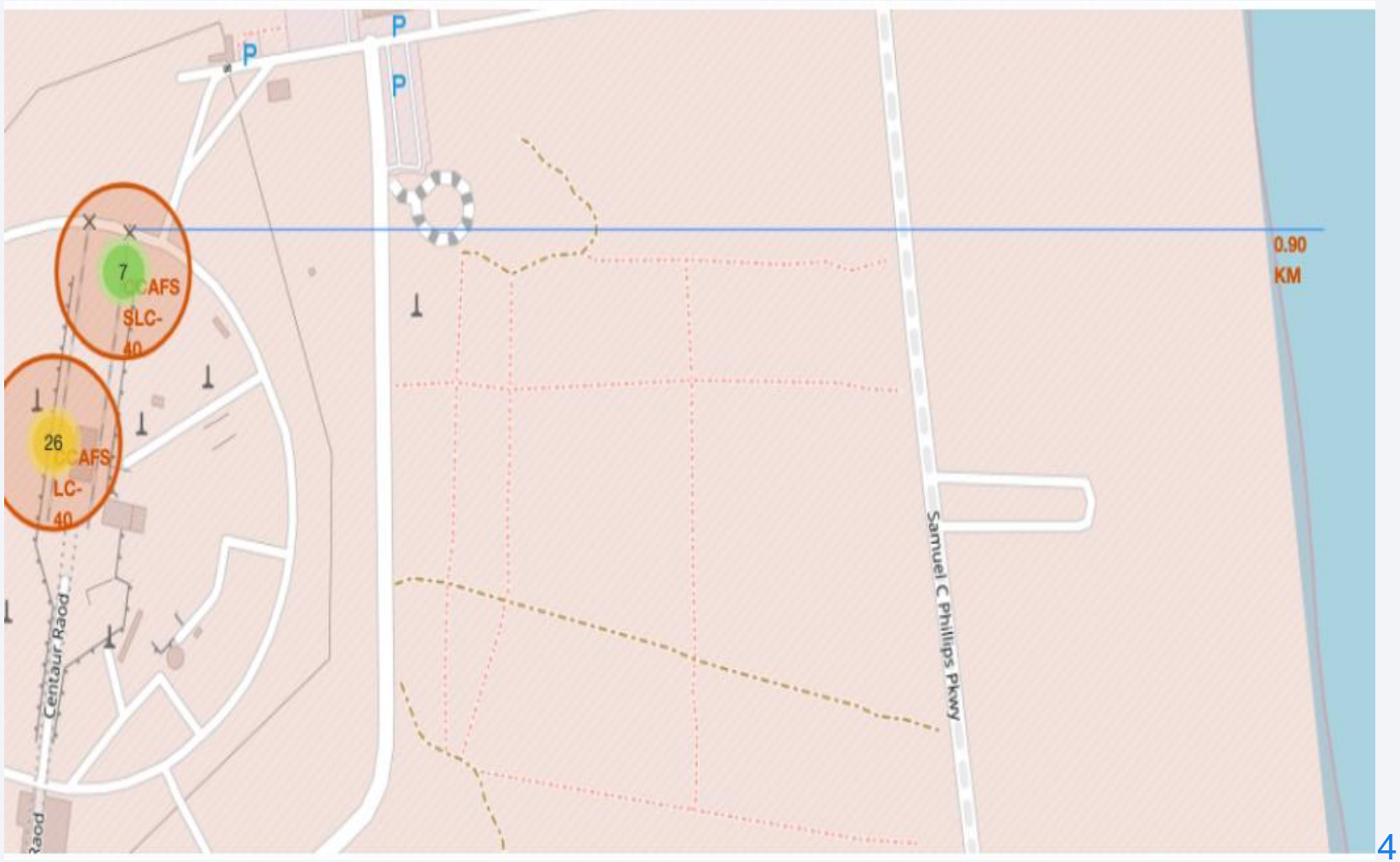
SpaceX VAFB SLC-4E Launch Site has a single Launch Site Stationed of California
Our Cluster Marker indicates 10 Falcon 9 launches have taken place at this site

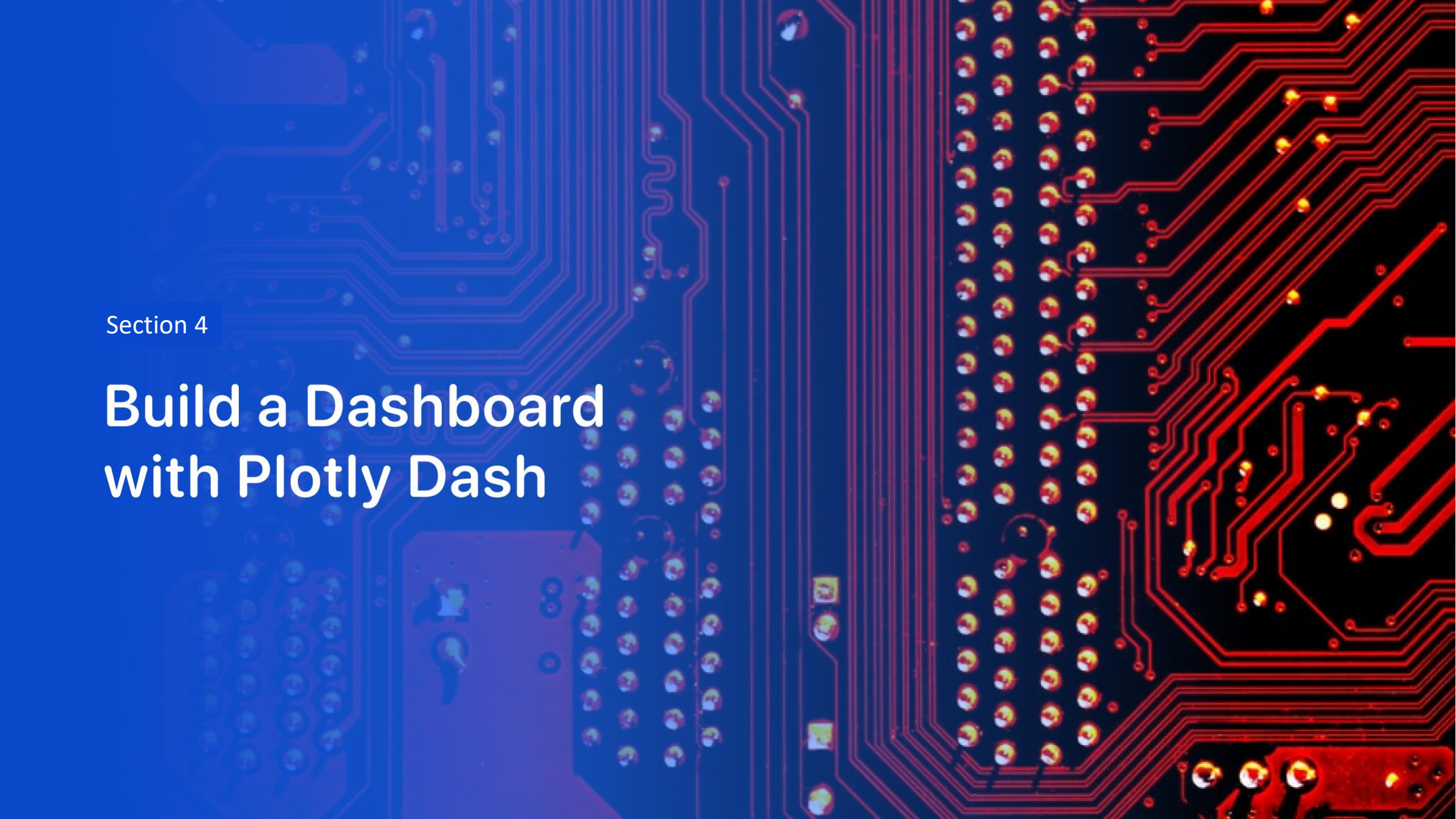
SpaceX Launch Sites KSC LC-39A , CCAFS LC40 and CCAFS SLC-40 are located in Florida and 46 lunches have taken place.



Launch site distance from coastline

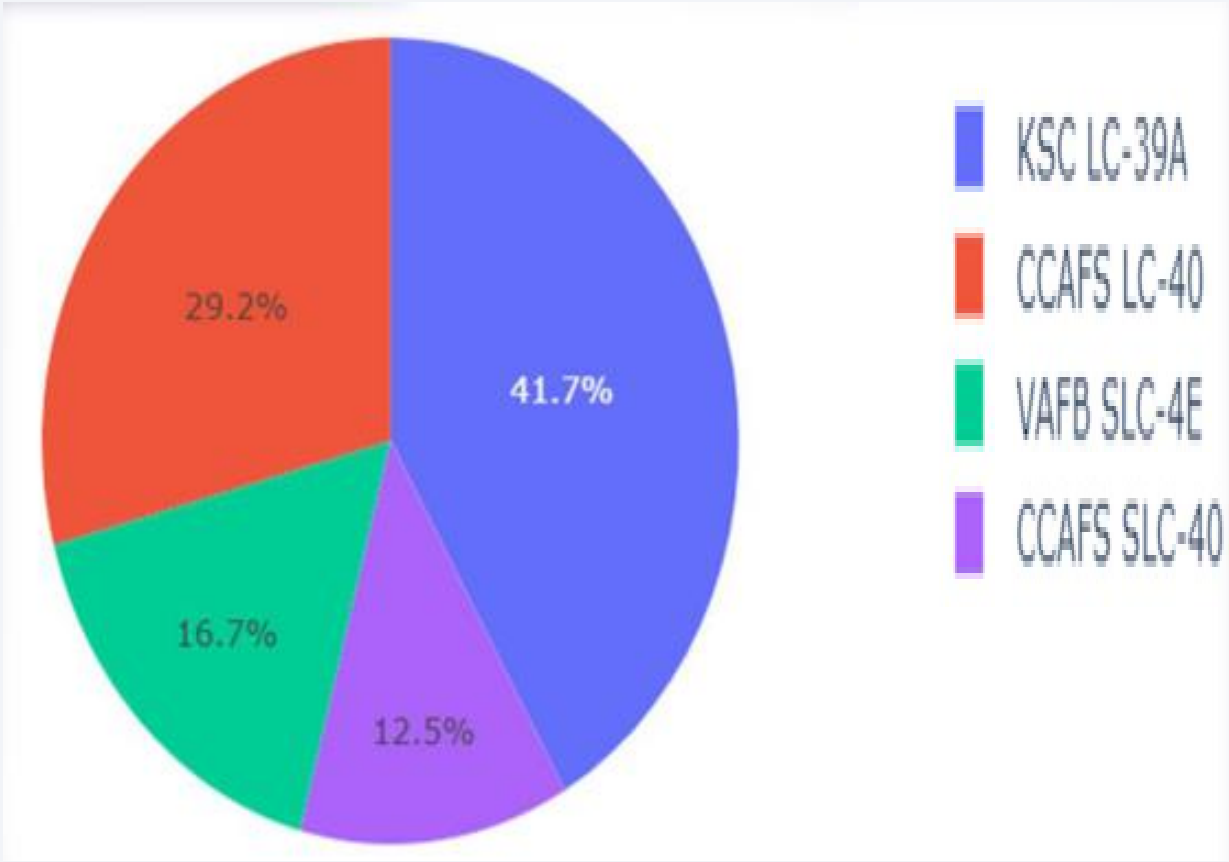
We can see that the nearest coastline to the Cape Canaveral sites is within 1 kilometer and exactly measured to 0.90km for the LC-40 site



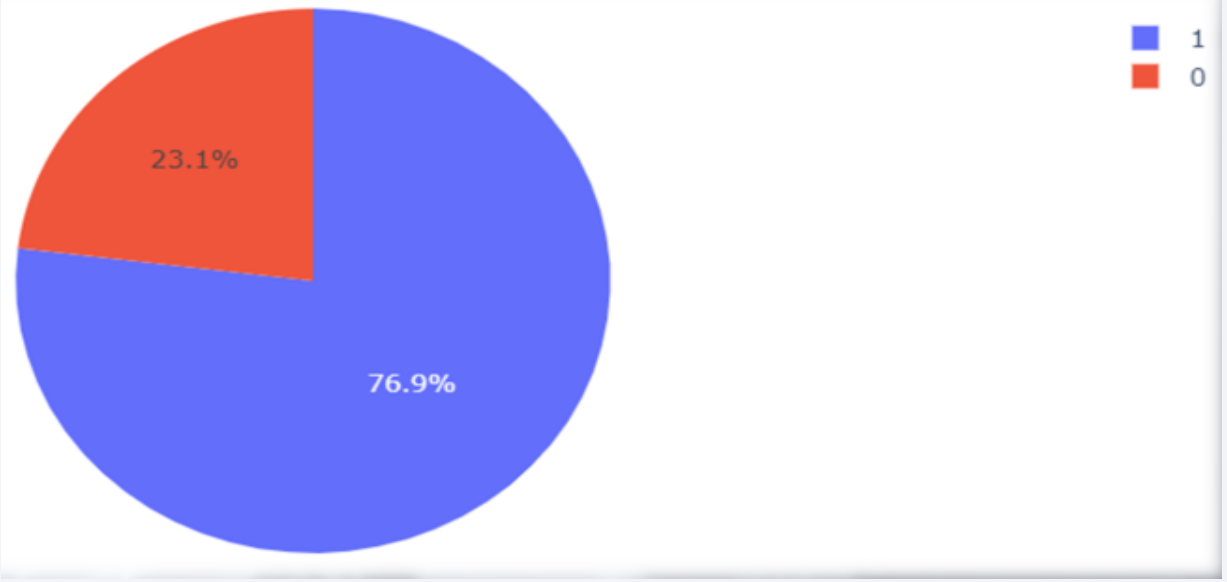

Section 4
Build a Dashboard with Plotly Dash

Pie Chart of Success Count for all Sites

KSC LC-39A Launch site accounts for the largest percentage of the total number of successful landings at 41.7%



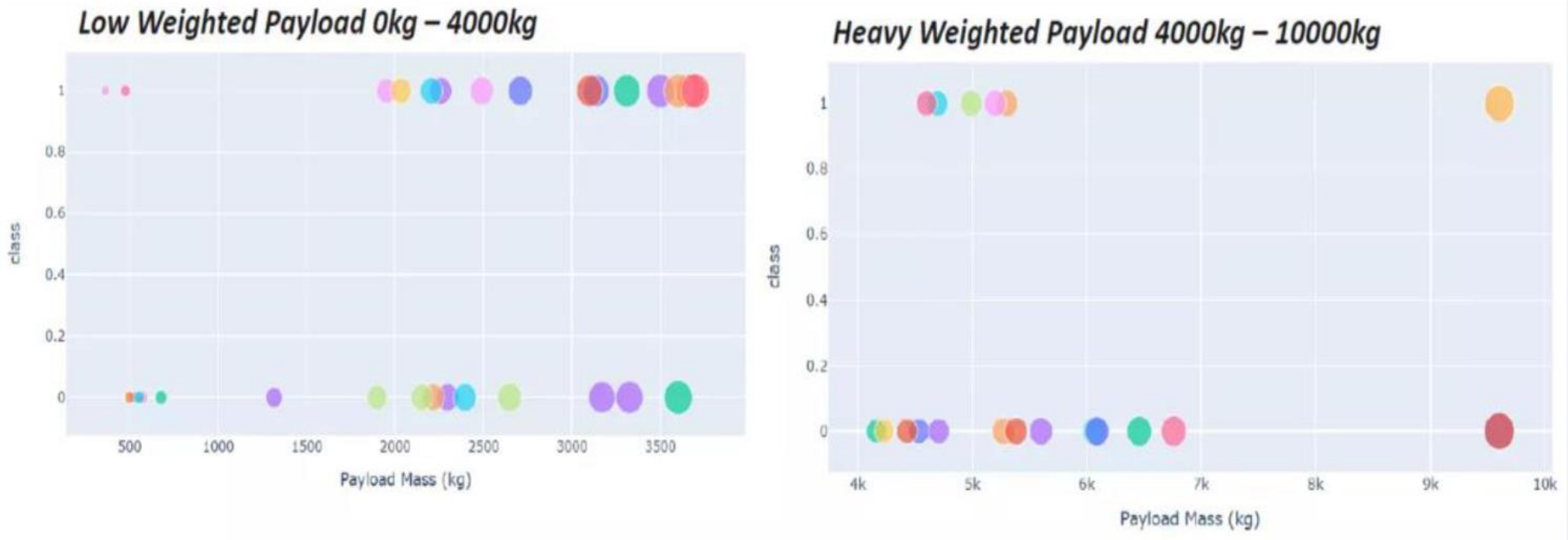
Launches for site KSC LC-39A



The KSC LC-39A Launch Site also has the highest probability of success per launch:

- 76.9% of all launches at the KSC LC-39A Site Land Successfully
- 23.1%of all launches at the KSC LC-39A Site Fail to Land

Payload vs Launch outcome for all sites



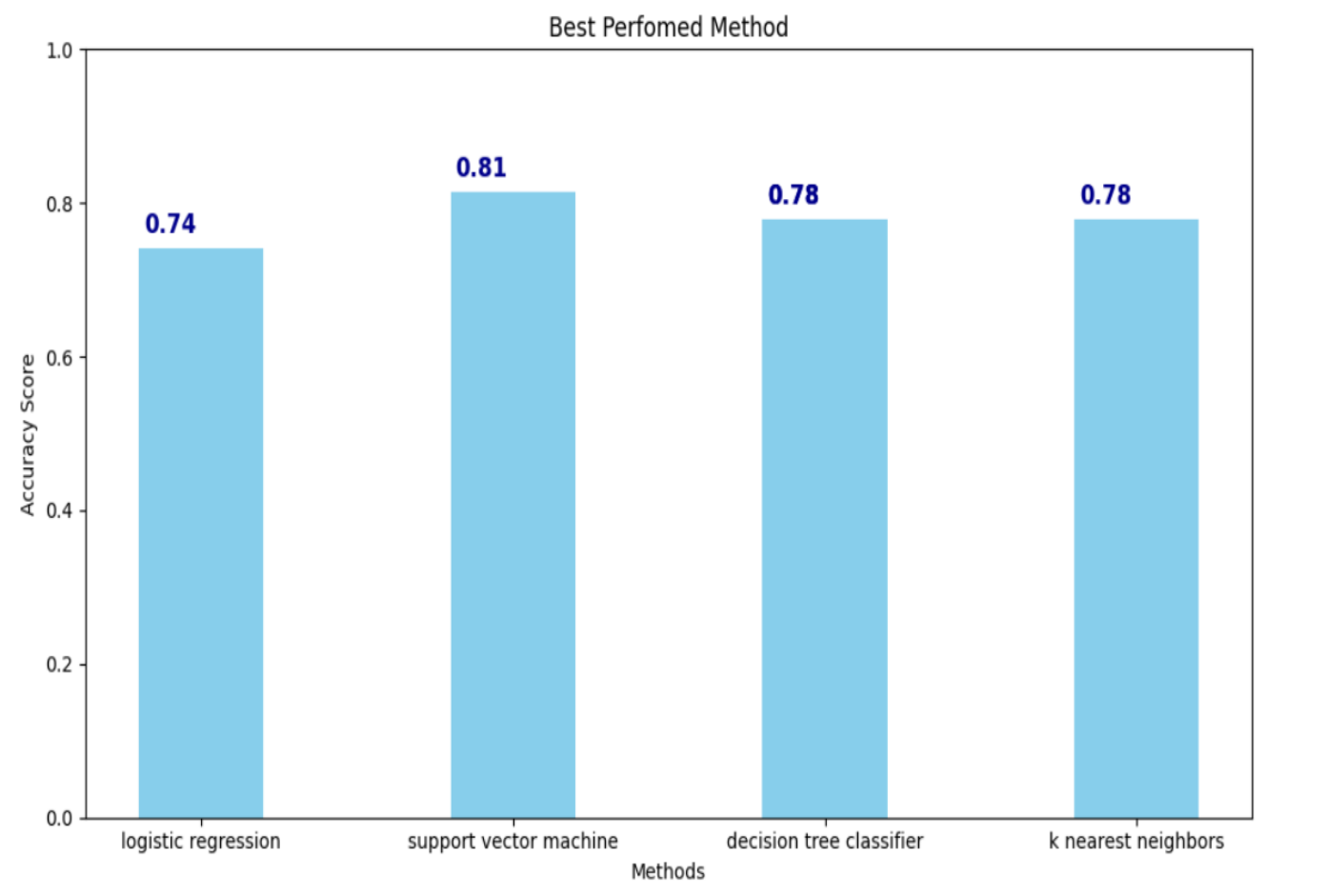
We can see the success rates for low weighted is higher than the heavy weighted Payload

Section 5

Predictive Analysis (Classification)

Classification Accuracy

Support Vector Machine model performed the best, achieving the highest accuracy score of 81%.

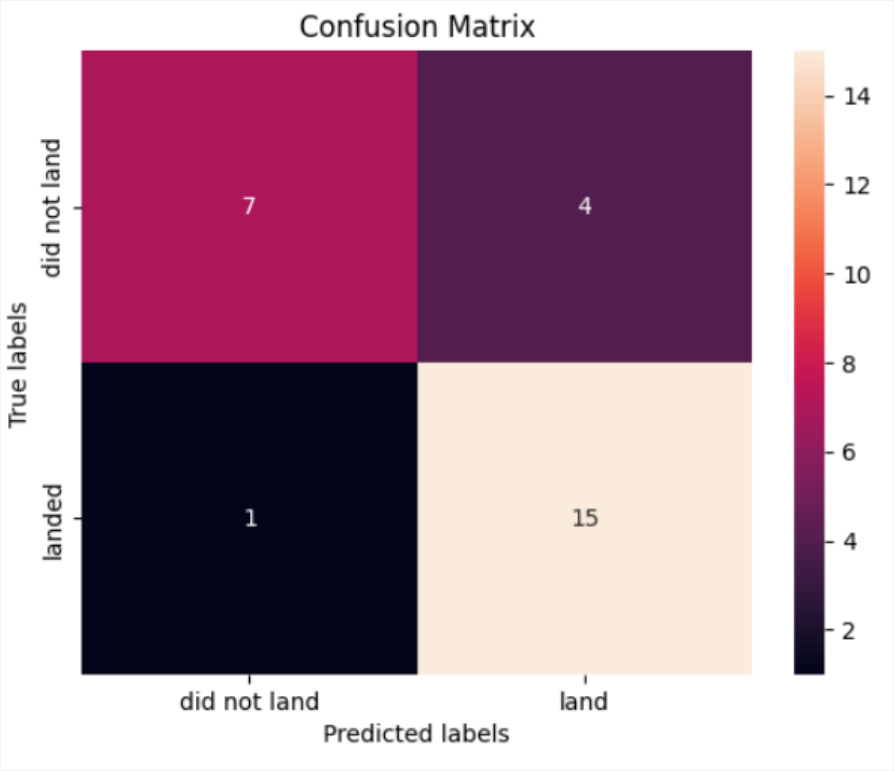


Confusion Matrix

We see that our model is able to correctly predict that 7 of the 11 testing points that fail to land

SVM model can correctly predict that all 15 of 16 rockets that land will land

- Sensitivity: 7/11=63.64%
- Specificity = 15/16 = 93.75%
- Accuracy: (7+15)/(7+4+1+15)=81.48%



Conclusions

- ✓ Support Vector Machine model shows the best accuracy 81.84% for predicting Falcon Rocket landing.
- ✓ Low weighted payloads perform better than the heavier payloads.
- ✓ KSC LC-39A Launch Site has the highest probability of success rate.
- ✓ ES-L1, SSO, HEO and GEO Orbits have the highest rate of landing successfully.
- ✓ Success landing rate has improved generally during years 2013 to 2020.

Thank you!