IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Chidili Chijioke
May 3, 2023

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
    - Exploratory Data Analysis Results
    - Interactive Analytics
    - Predictive Analysis

Introduction

- Project background and context
  - SpaceX advertises Falcon9 rocket launches on its website, with a cost of 62million dollars; other providers cost upward of 165million 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.

- Problems you want to find answers
  - In this capstone, we will predict if the Falcon 9 first stage will land successfully.

Section 1

Methodology

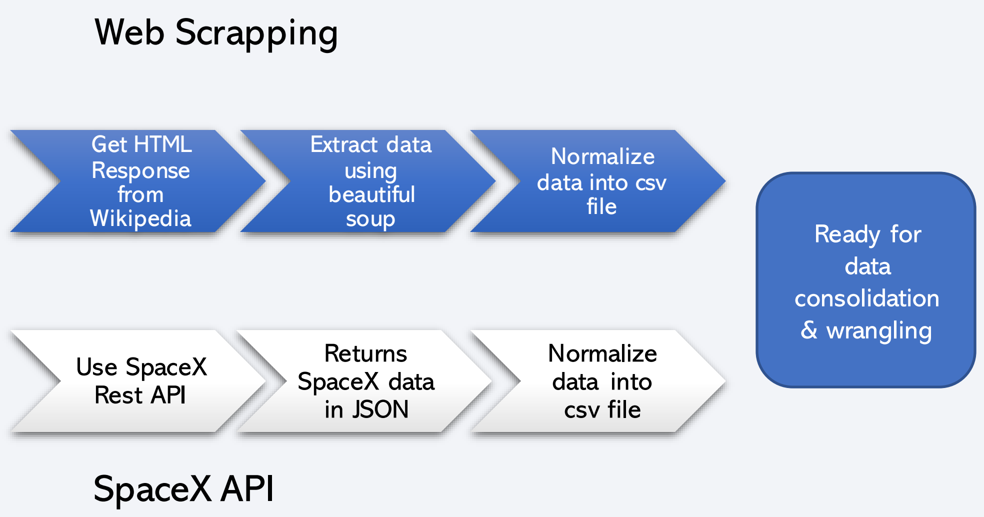
Methodology

Executive Summary

- Data collection methodology:
  - SpaceX Rest API
  - Web Scrapping data from Wikipedia
- Perform data wrangling
  - Data cleaning and removal of null values and irrelevant columns
  - Standardize and One hot encoding of data for Machine learning
- 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, SVM, KNN, DTC models built and evaluated for the best classified

Data Collection

- The following dataset was collected:
  - SpaceX launch data that is gathered from the SpaceX Rest API.
  - This API will give us data about launches that includes information about the rocket used, landing & launching specifications, payload delivered and landing outcomes.
  - 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 API

https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/Collecting-Data.ipynb

1.Getting Response from API

```
Spacex_url="https://api.spacedata.com/v4/launches/past"
Response=requests.get(Spacex_url)
```

2.Converting response to a json file

```
Data=pd.json_normalize(response.json())
```

3.Apply custom functions to clean data

```
getBoosterVersion(data)
getLaunchSite(data)
getPayLoadData(data)
getCoreData(data)
```

4. Assign list to Dict 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(launch_dict)
```

5. Filter dataframe & export file

```
df=df.drop(df['BoosterVersion']!
='Falcon9'].index,
inplace=False)

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

Data Collection - Scraping

- Web Scrapping from Wikipedia

https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/Webscraping.ipynb

1.Getting Response from HTML

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

2.Creating BeautifulSoup Object

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

3.Finding Tables

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

4.Getting column names

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

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 eac
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()
```

7. Converting dictionary to DataFrame

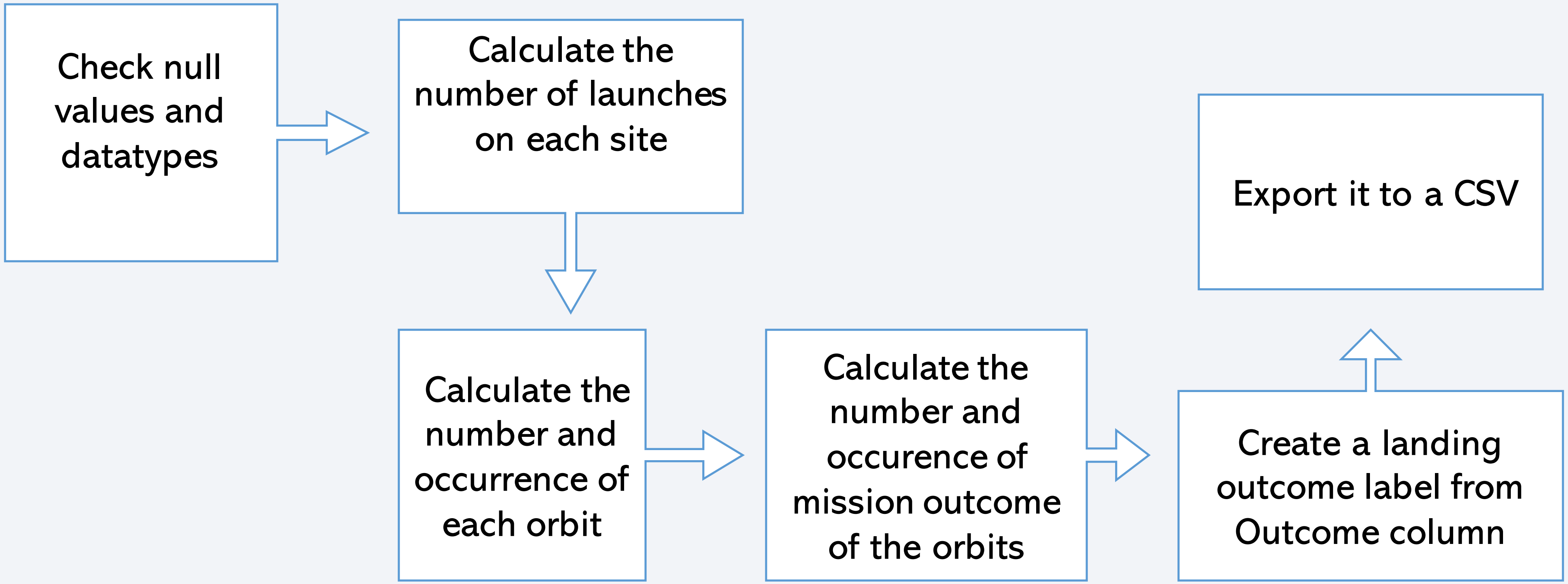
```
pad_dict_list(launch_dict,0)

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

8.DataFrame to .csv

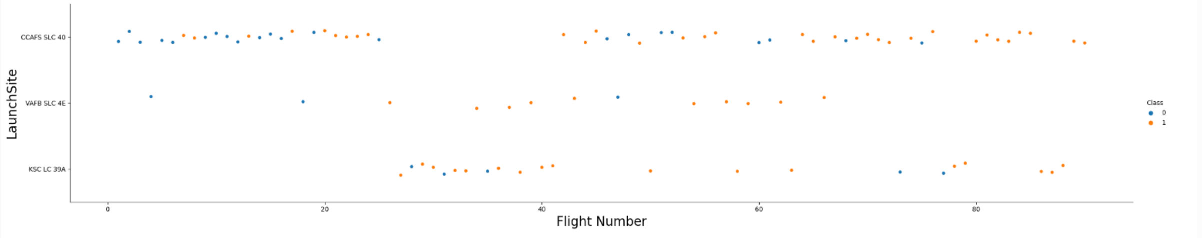
```
df.to_csv('spacex_web_scraped.csv', index=False)
```

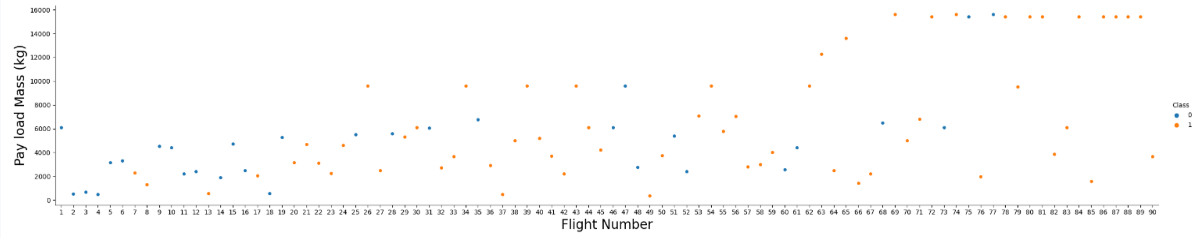
Data Wrangling

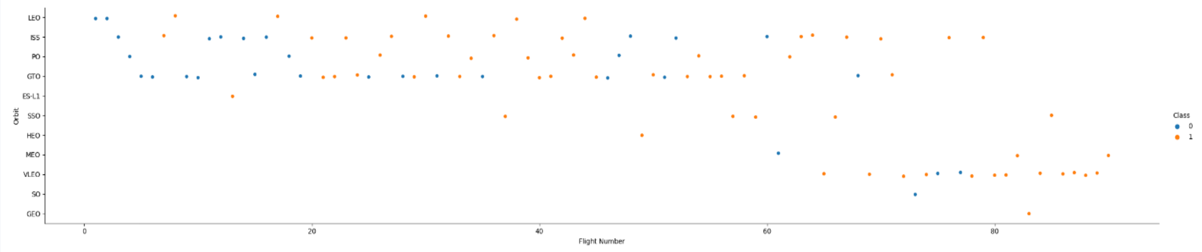


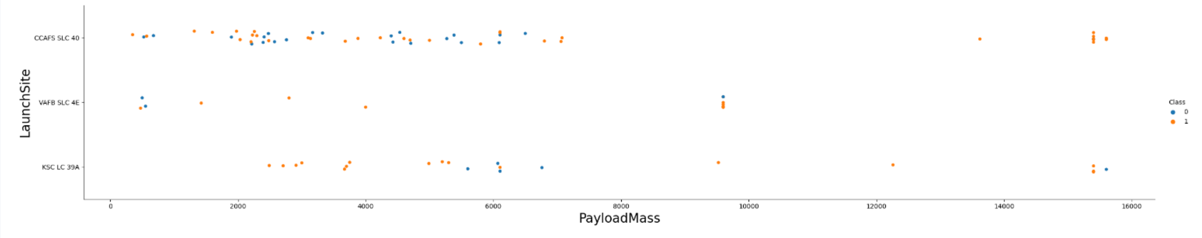
https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/Data%20Wrangling-2.ipynb

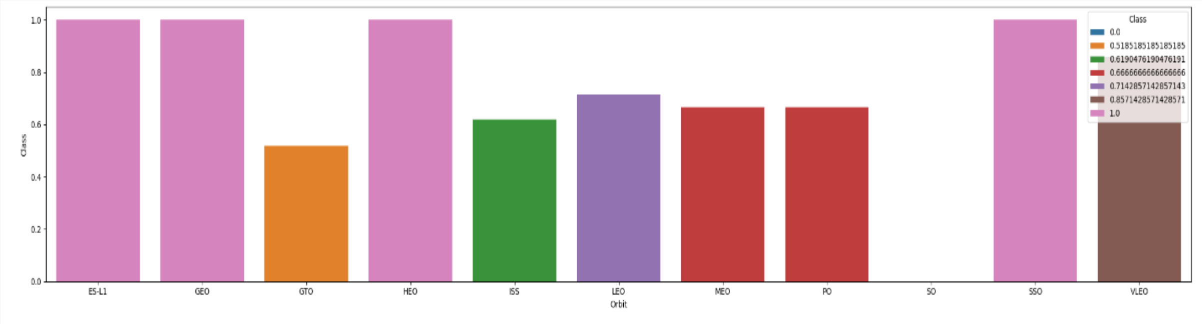
EDA with Data Visualization

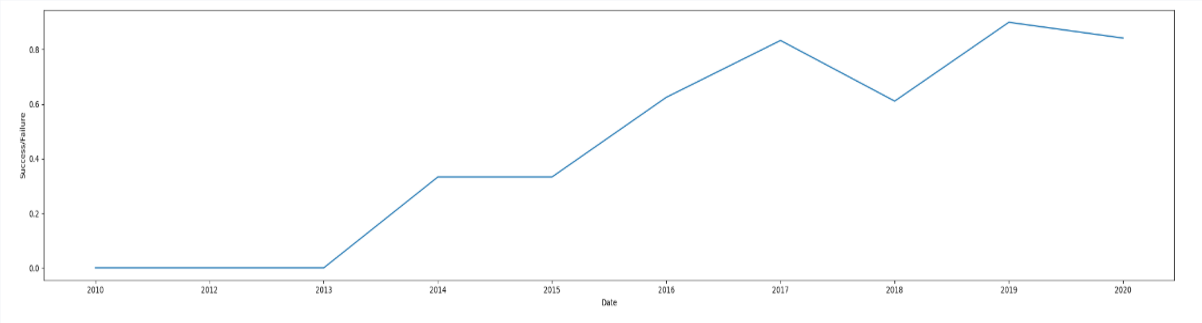












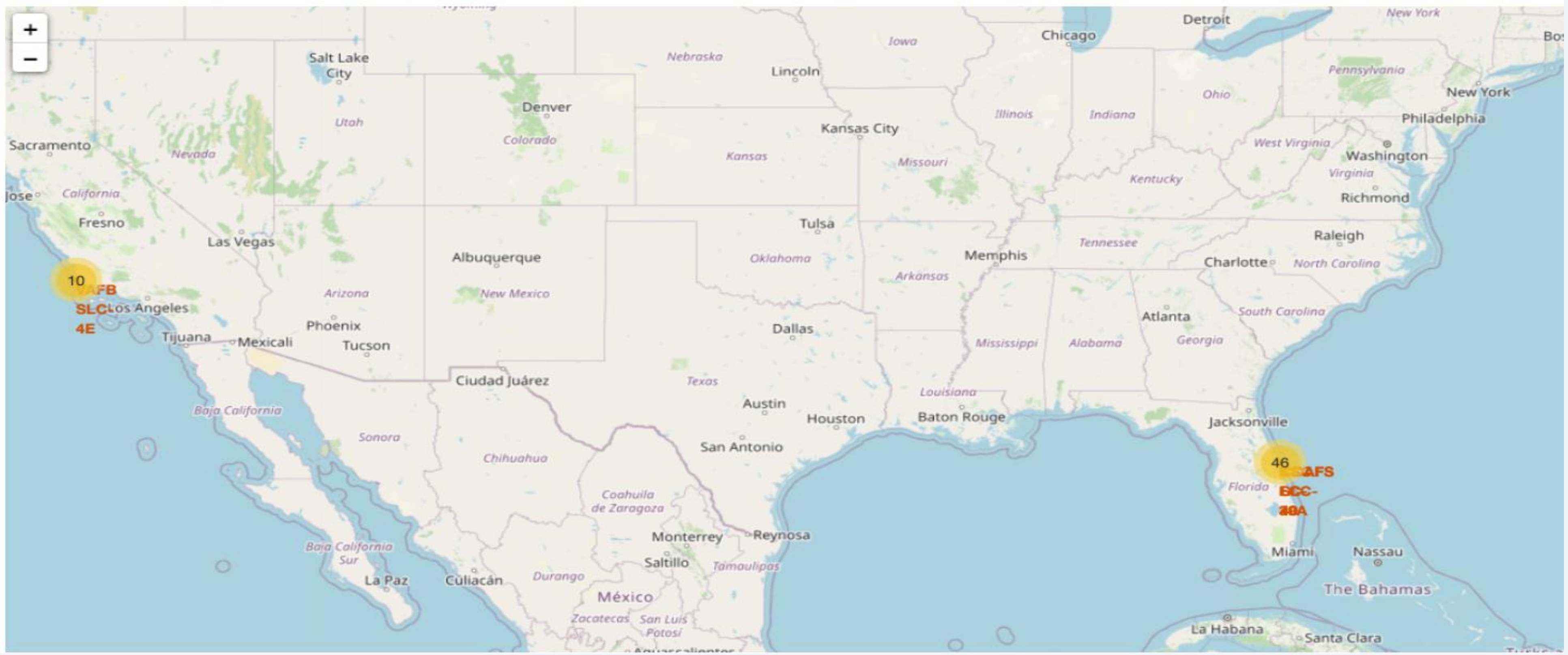
https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/EDA-DataViz.ipynb

EDA with SQL

- SQL queries performed:
  - 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 successful 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.
  - 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 successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/EDA-SQL.ipynb

Build an Interactive Map with Folium

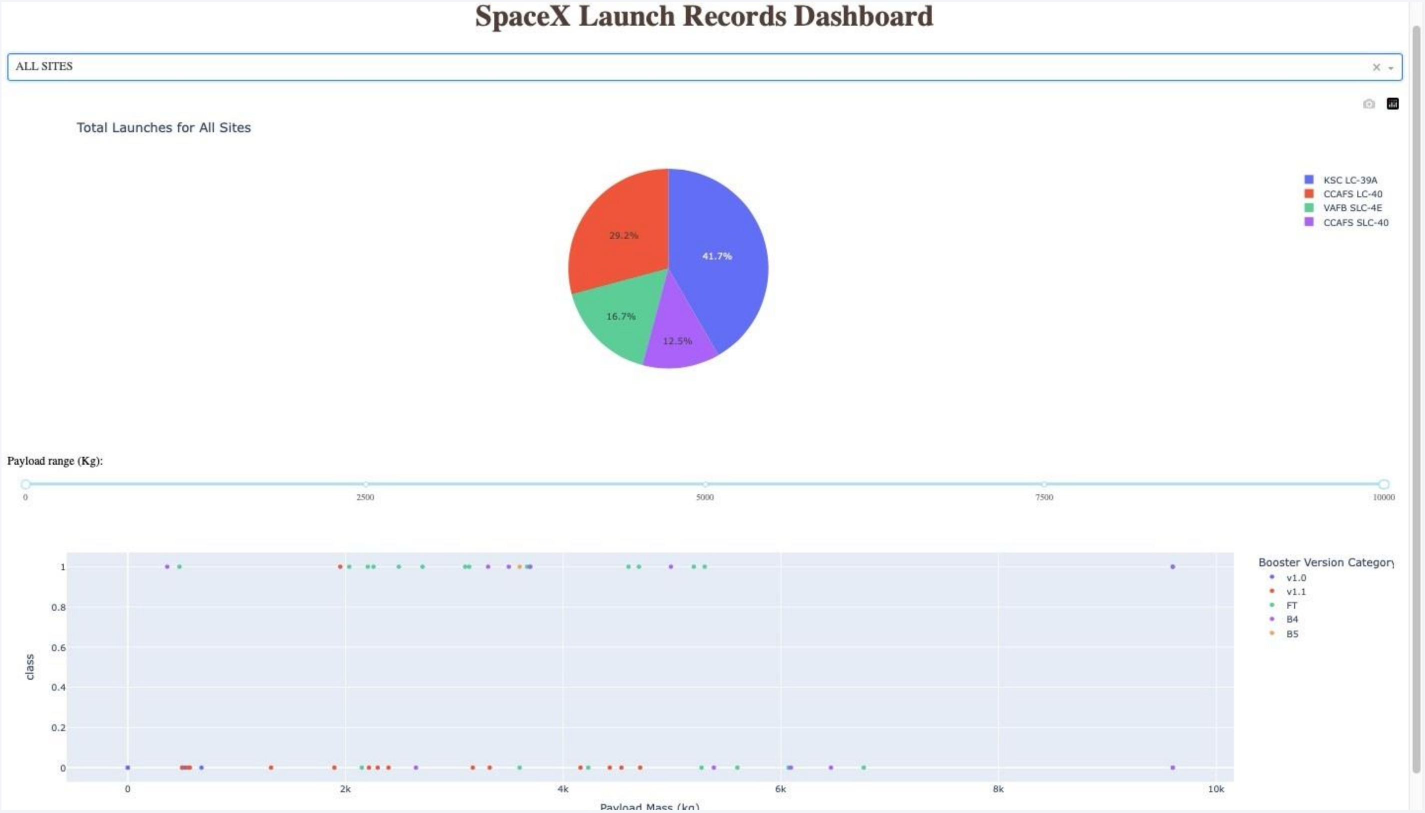


Map markers were added to the map with the aim to finding the optimal location for building a launch site

`

https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/Launch-sites-location.ipynb

Build a Dashboard with Plotly Dash

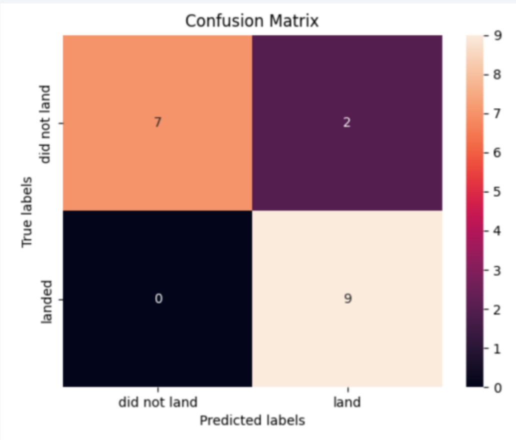


https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/spacex_dash_app.py

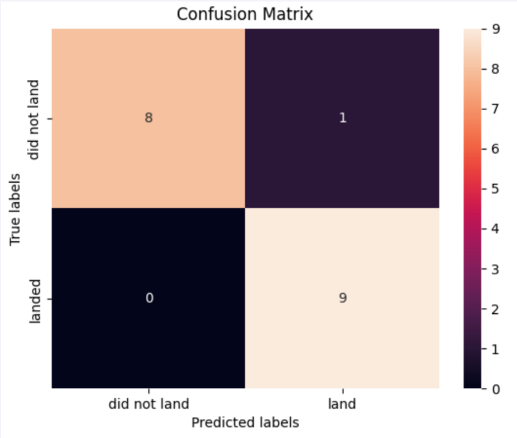
Predictive Analysis (Classification)

- After training and testing the models. The SVM model achieved the highest accuracy of 94.4% in terms of Area Under the Curve. Logistic Regression and KNeighbors models had an accuracy of 88.9%, and Decision tree model with the lowest accuracy of 83.3%.

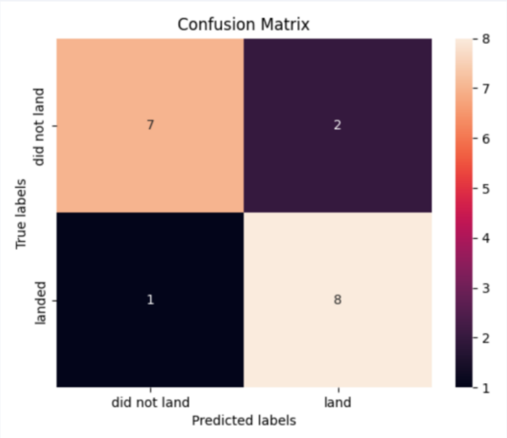
Logistic Regression Model



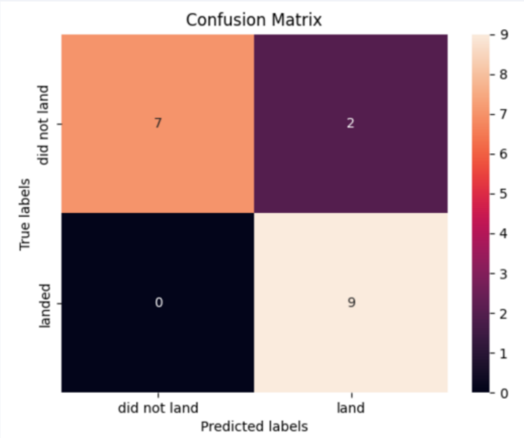
SVM Model



DTC Model



KNN Model



https://github.com/Oscardafuscar/Applied-Capstone-Project/blob/main/Model-Prediction.ipynb

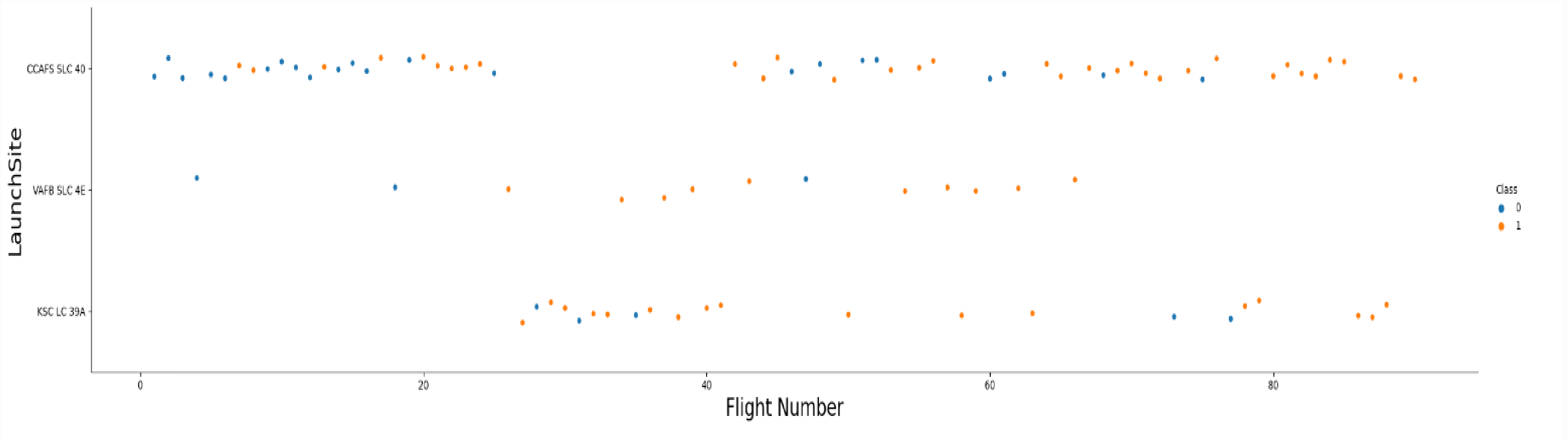
Results

- The Success rate for SpaceX launches corresponds to time. This explains perfect launches in the future.
- The Orbits ES-L1, GEO, HEO, SSO has the best Success Rate.
- Heavier payloads Performed poorly compared to low weighted payloads.
- The site with the most successful launches is KSC-LC-39A.
- The SVM Model is the best in terms of prediction accuracy.

Section 2

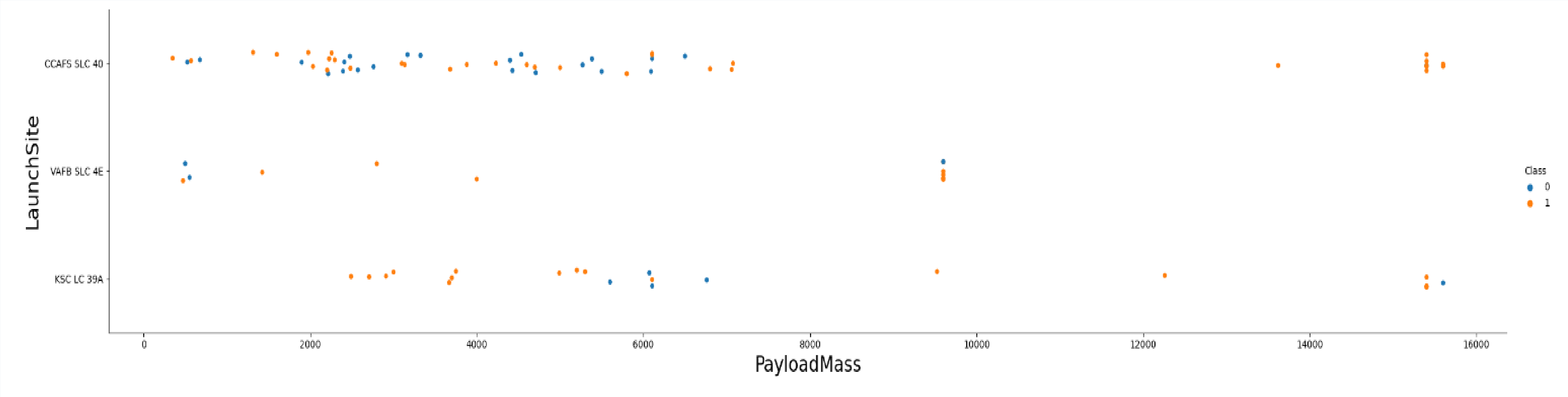
Insights drawn from EDA

Flight Number vs. Launch Site



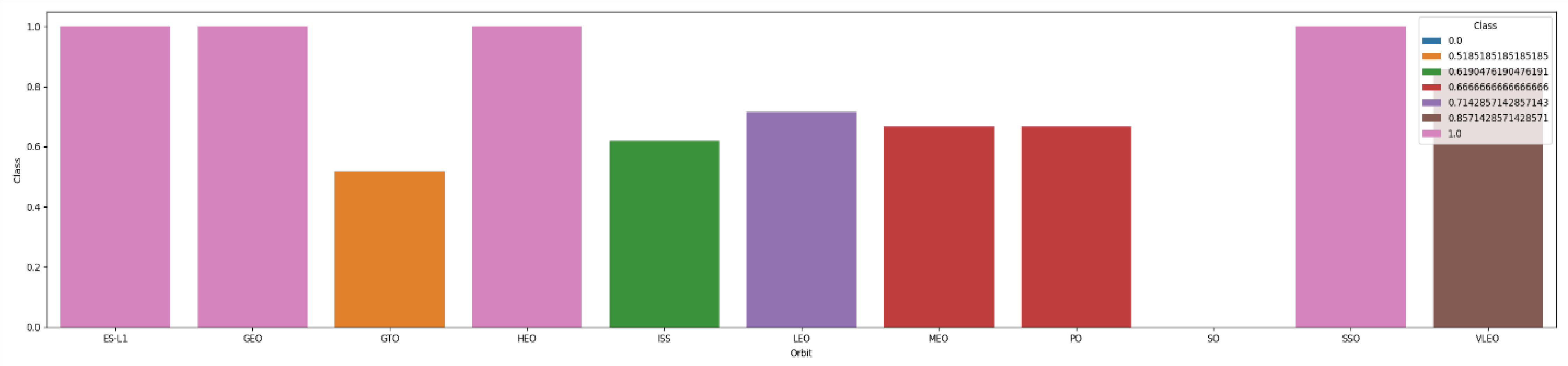
- Launches from the sites of CCAFS-SLC-40 are significantly higher than launches from KSC-LC-39A and VAFB-SLC-4E.

Payload vs. Launch Site



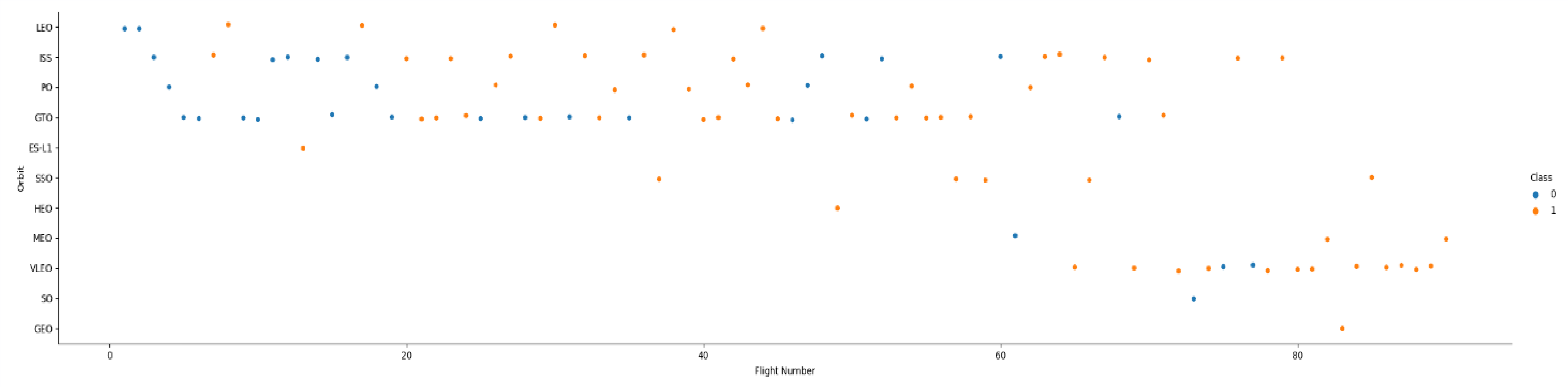
- Majority of launches with low weighted payload were performed at the CCAFS-SLC-40 launch site.
- No heavy weighted payload launch was performed at the VAFB-SLC-4E.

Success Rate vs. Orbit Type



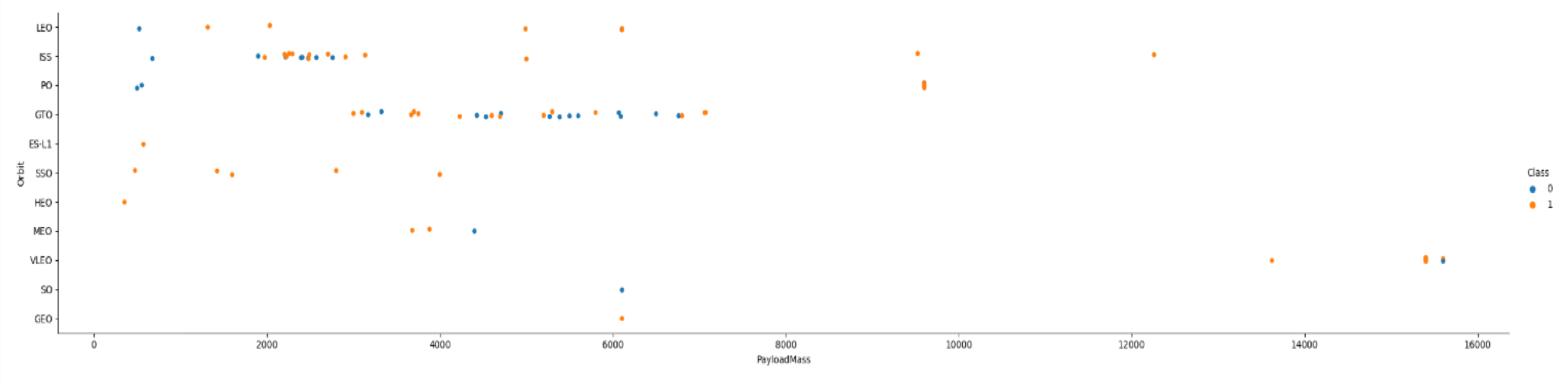
- The Orbit types ES-L1, GEO, HEO, SSO has the best Success Rate.
- No launch to SO orbit.

Flight Number vs. Orbit Type



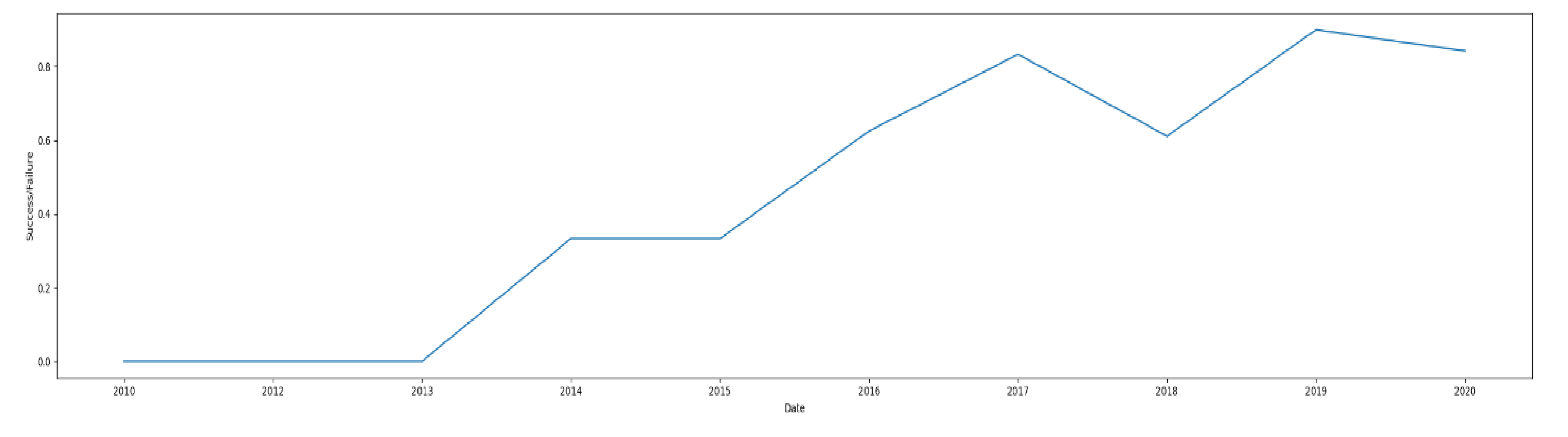
- A trend can be observed of shifting to VLEO launches as the number of launches increases with time.

Payload vs. Orbit Type



- There are strong correlation between ISS and Payload at the range between 2000-4000, also GTO and Payload at the range between 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.

All Launch Site Names

- **%sql** select distinct("launch_site") from SPACEXTBL

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;
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	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_) as ["payload_mass"] from SPACEXTBL where "Customer"="NASA (CRS)";

"payload_mass"
45596

Average Payload Mass by F9 v1.1

- **%sql** select avg(PAYLOAD_MASS__KG_) as ['payload_mass(avg)'] from SPACEXTBL\ where "Booster_Version" = "F9 v1.1";

'payload_mass(avg)'
2928.4

First Successful Ground Landing Date

- **%sql** select min(DATE) from SPACEXTBL where "Landing _Outcome" = "Success (ground pad)";

min(DATE)
01-05-2017

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;

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") from SPACEXTBL GROUP BY "Mission_Outcome";

Mission_Outcome	count("Mission_Outcome")
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);

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, 4, 2) as month, "Landing _Outcome", "Booster_Version", "launch_site" from SPACEXTBL **where** "Landing _Outcome" **=** 'Failure (drone ship)' **and** substr(Date,7,4)='2015';

month	Landing _Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- **%sql** SELECT "Landing _Outcome", count(*) as count_outcomes FROM SPACEXTBL WHERE DATE between '04-06-2010' **and** '20-03-2017' group by "Landing _Outcome" order by count_outcomes DESC;

Landing _Outcome	count_outcomes
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Failure	3
Controlled (ocean)	3
Failure (parachute)	2
No attempt	1

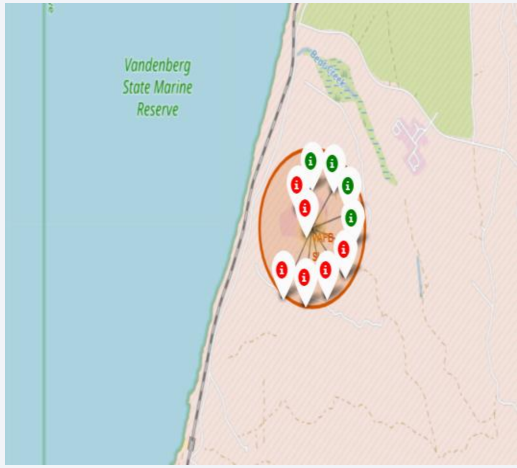
Section 3

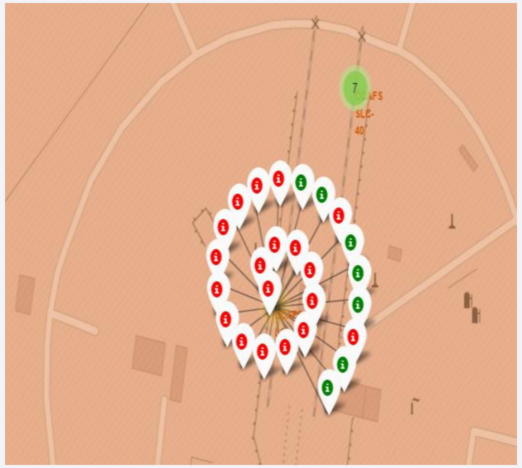
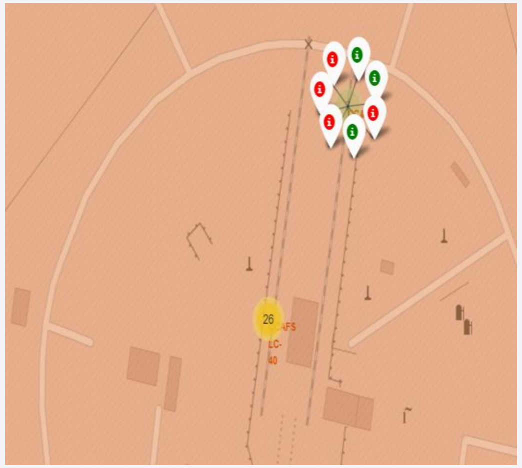
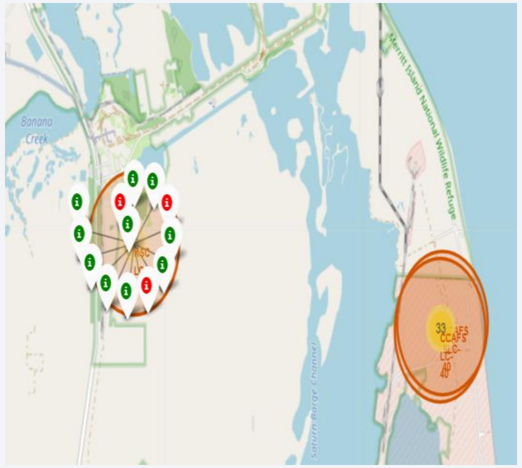
Launch Sites Proximities Analysis

All launch sites location on Map

- All launch sites have a close proximity to the coastline.

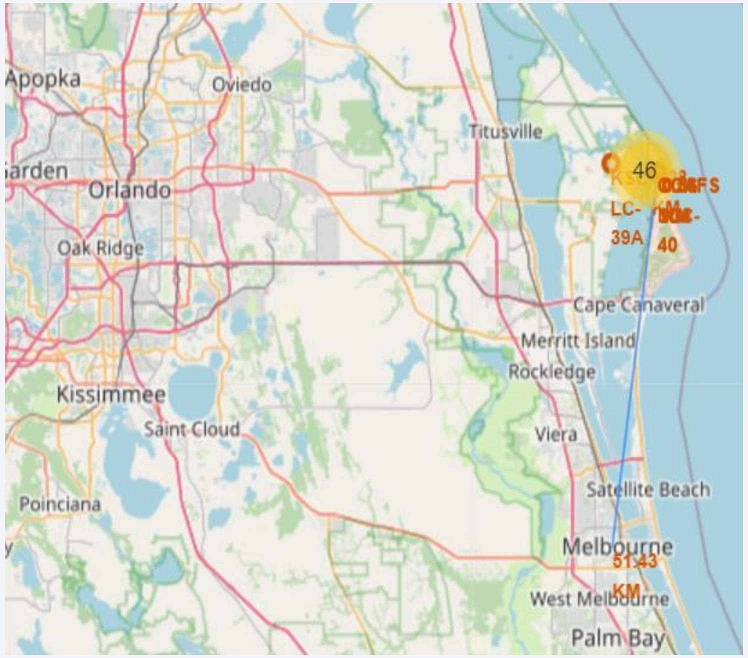
Launch outcomes of each site

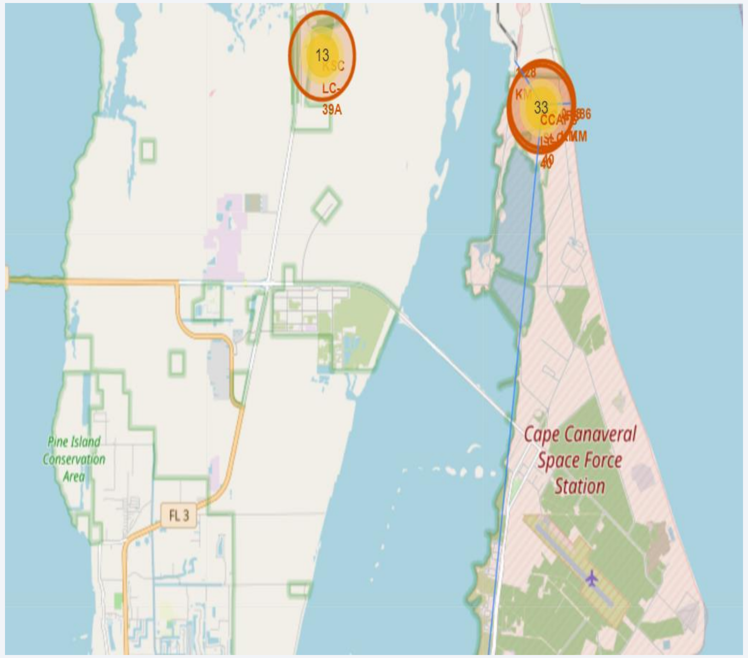


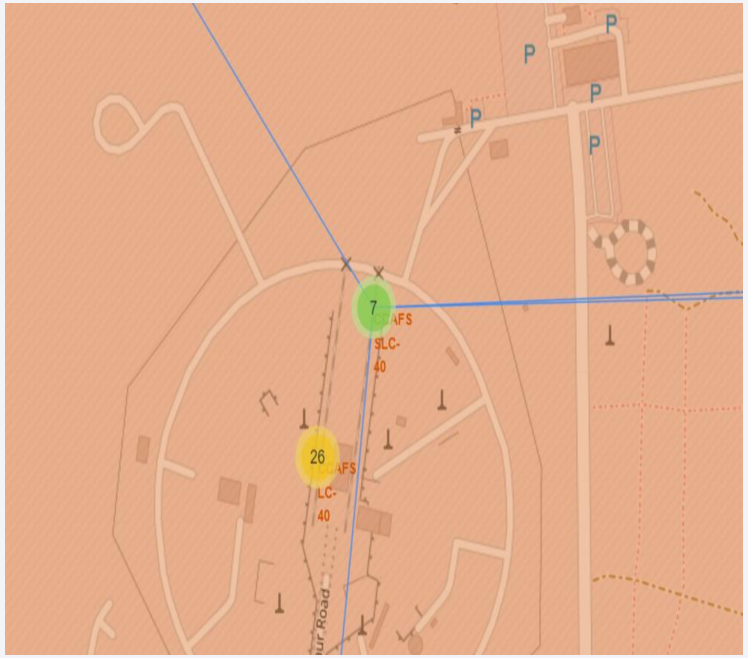




- These are the outcomes of launch sites; VAFB-SLC-4E, CCAFS-SLC-40, CCAFS LC-40 and KSC-LC-39A respectively.
- The green icons indicates a successful launch while the red icon indicates otherwise.

Distance between a launch site and its proximities





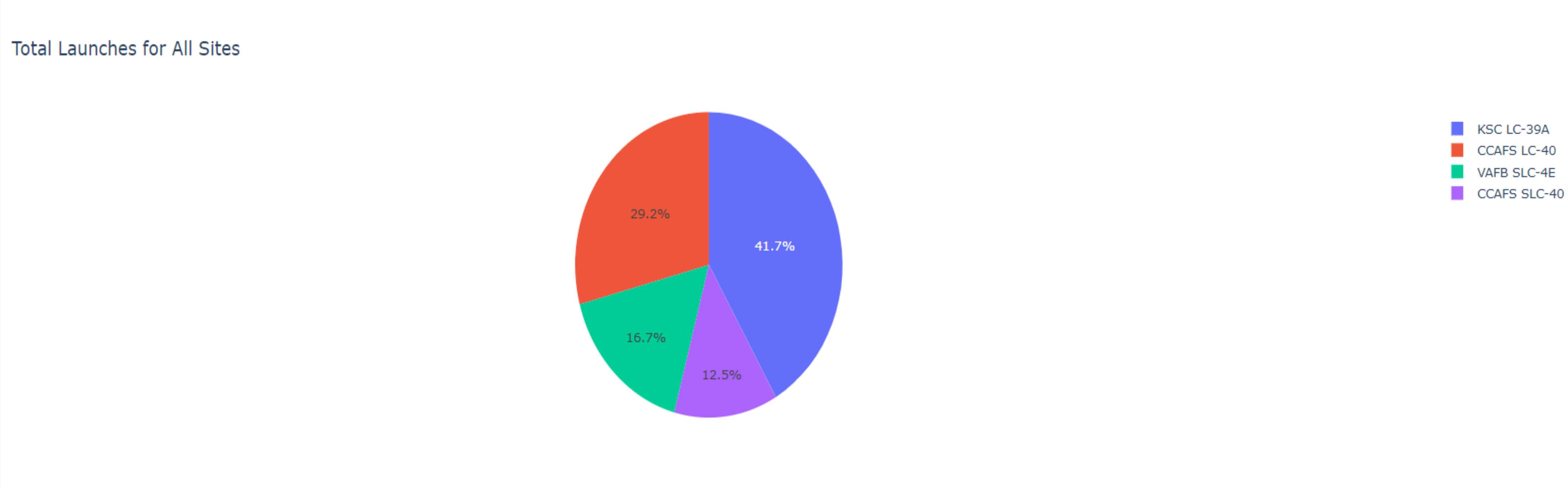


- Each map zooms in better to show the launch site CCFA-LC-40.
- This shows launch sites are far from human population and also closer to the coastline.

Section 4

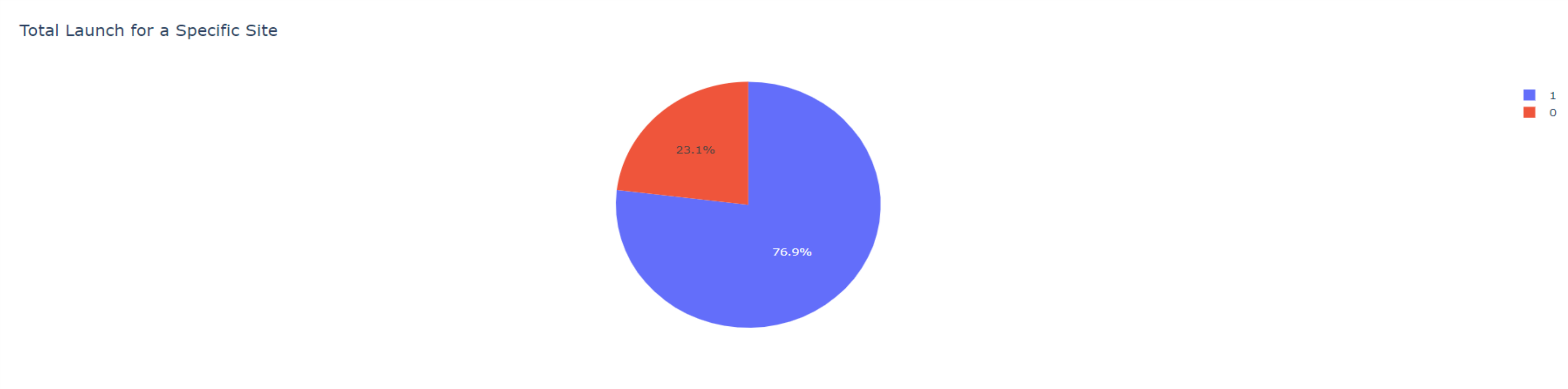
Build a Dashboard with Plotly Dash

Launch success for all sites



- KSC-LC-39A had the most successful launches from all the sites.
- CCAFS-SLC-40 had the least successful launches from all the sites.

Success rate for KSC-LC-39A launch site



- KSC-LC-39A achieved a 76.9% success rate and a 23.1% failure rate.
- This makes it the most successful launch site out of all launch sites.

Payload vs. Launch outcome

Low weighted payload 0kg-5000kg

Heavy weighted payload 5000kg-10000kg

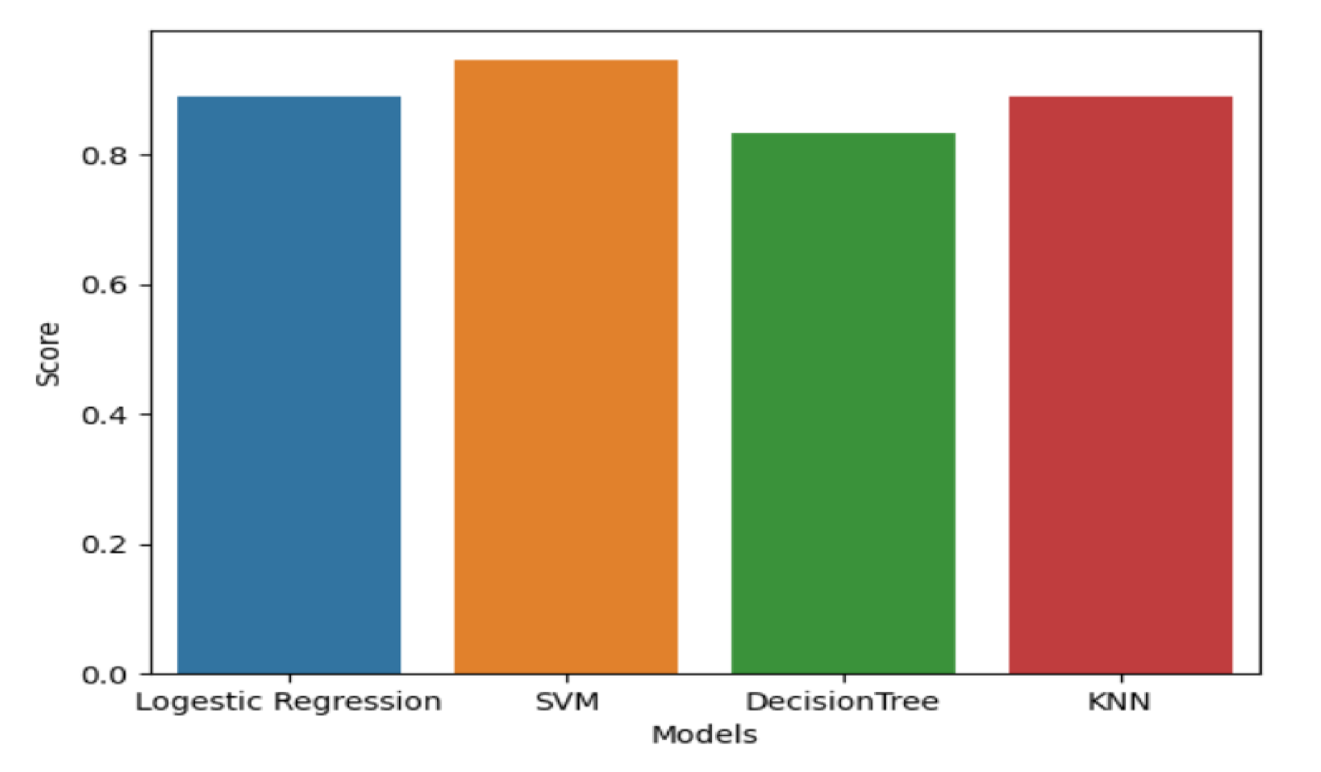
- CCAFS LC-40
- VAFB SLC-4E
- KSC LC-39A
- CCAFS SLC-40

- Success rate for low weighted payload is higher than the heavy weighted payloads.
- Majority of the booster version used for heavy weighted payloads are FT.

Section 5

Predictive Analysis (Classification)

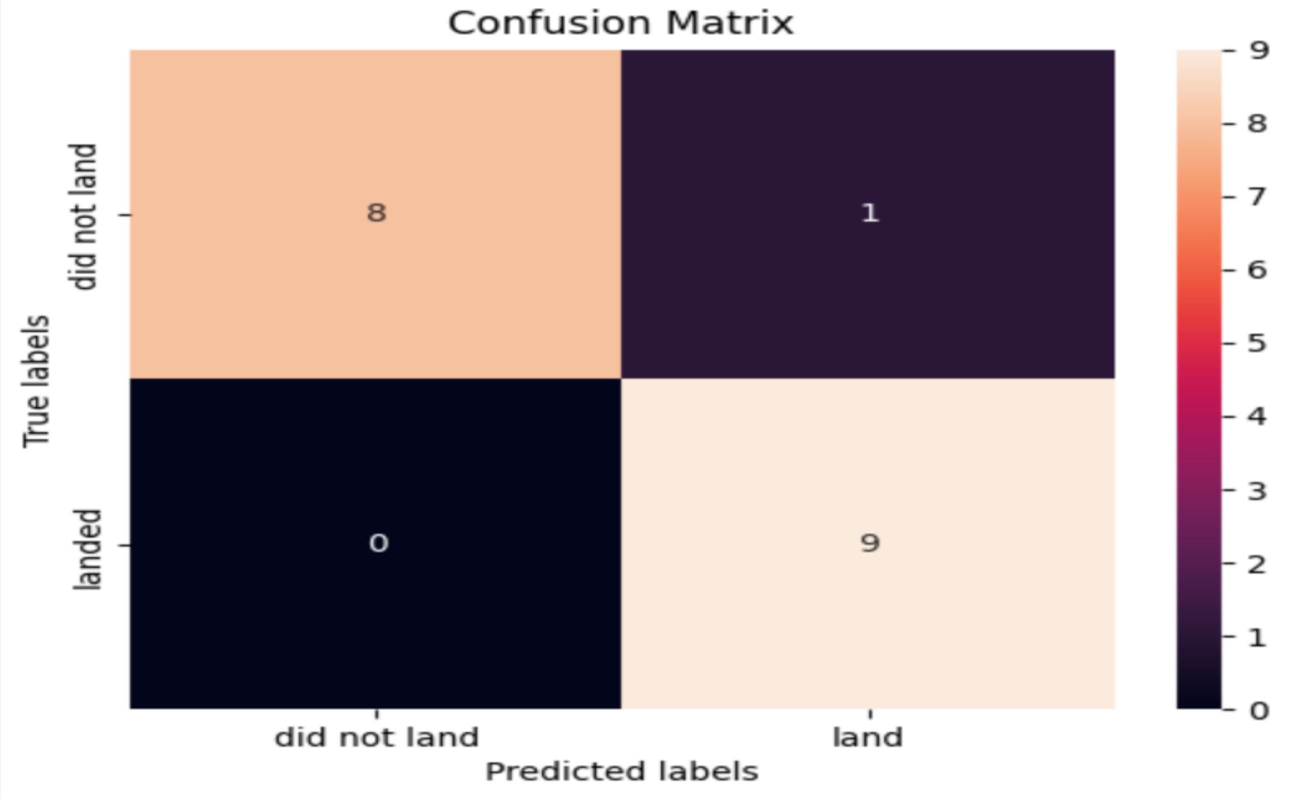
Classification Accuracy



Models	Score
SVM	0.944444
Logestic Regression	0.888889
KNN	0.888889
DecisionTree	0.833333

- The SVM model had the highest accuracy of 94.4%.

Confusion Matrix



- This shows the confusion matrix for SVM model. The model predicted accurately launch outcome(landed) accurately while the launch outcome(

Conclusions

- The Success rate for SpaceX launches corresponds to time. This explains perfect launches in the future.
- The Orbits ES-L1, GEO, HEO, SSO has the best Success Rate.
- Heavier payloads Performed poorly compared to low weighted payloads.
- The site with the most successful launches is KSC-LC-39A.
- The SVM Model is the best in terms of prediction accuracy.

Thank you!