



IBM Developer  
SKILLS NETWORK

# Winning Space Race with Data Science

Sujata Singh  
6/14/2022



# 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

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.

- Problems you want to find answers

The Project Task is to predict if the first stage of the SpaceX Falcon 9 will land successfully



Section 1

# Methodology

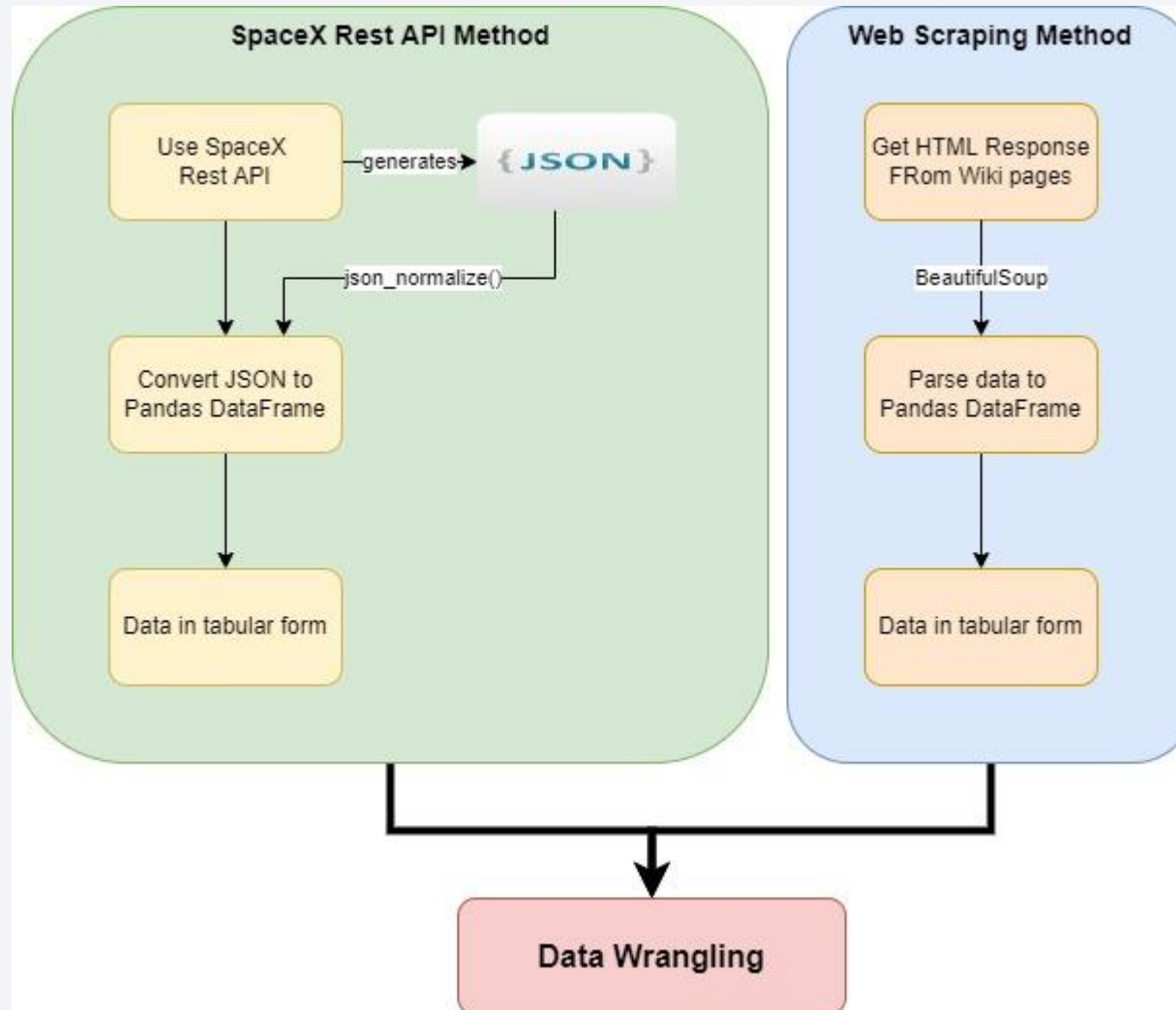
# Methodology

---

## Executive Summary

- Data collection methodology:
  - SpaceX Rest API
  - Web Scraping from Wikipedia
- Perform data wrangling
  - One Hot Encoding fields for Machine Learning
  - Data cleaning – null values, 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
  - Models built and evaluated : Linear Regression, KNN, SVM, Decision Tree
  - Best Classifier estimated

# Data Collection



## Two methods used for data collection :

- **From Space X API** – Specific endpoint to get the past launch data. Data obtained in json form is normalized to a flat table form
- **By Web Scraping related Wiki Pages** – web scrape some HTML tables that contain Falcon9 Launch records. Data is parsed into data frames for next steps.

# Data Collection – SpaceX API

- Data collection steps summary with SpaceX REST calls :

- GitHub URL for the complete code of SpaceX API calls  
[https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/O1\\_DataCollectionAPI.ipynb](https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/O1_DataCollectionAPI.ipynb)

## 1. Get Response from API and make it consistent

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

```
7) : response = requests.get(spacex_url)
```

```
8) [3] : static_json_url="https://tf-source-data.s3.us-east-1.amazonaws.com/TFM-050101781-0511/datasets/datasetx/API_Call_spacex_api.json"
```

We should see that the request was successful with the 200 status response code

```
[16] : response.status_code
```

```
0 [16] : 200
```

## 2. Decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

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

## 3. Apply custom functions to use specific data

```
# Lets take a subset of our dataframe keeping only the features we want and the flight number, and date_utc.  
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 boosters and row 1  
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 and replace the  
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 the time  
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, 15)]
```

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

Then, we need to create a Pandas data frame from the dictionary launch\_dict.

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

## 5. Filter dataframe to include only Falcon9 data

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

## 6. Data Wrangling

```
# Calculate the mean value of PayloadMass column  
df_mean=data_falcon9['PayloadMass'].mean()  
df_mean  
  
# Replace the np.nan values with its mean value  
data_falcon9['PayloadMass']=data_falcon9['PayloadMass'].replace(np.nan,df_mean)  
data_falcon9
```

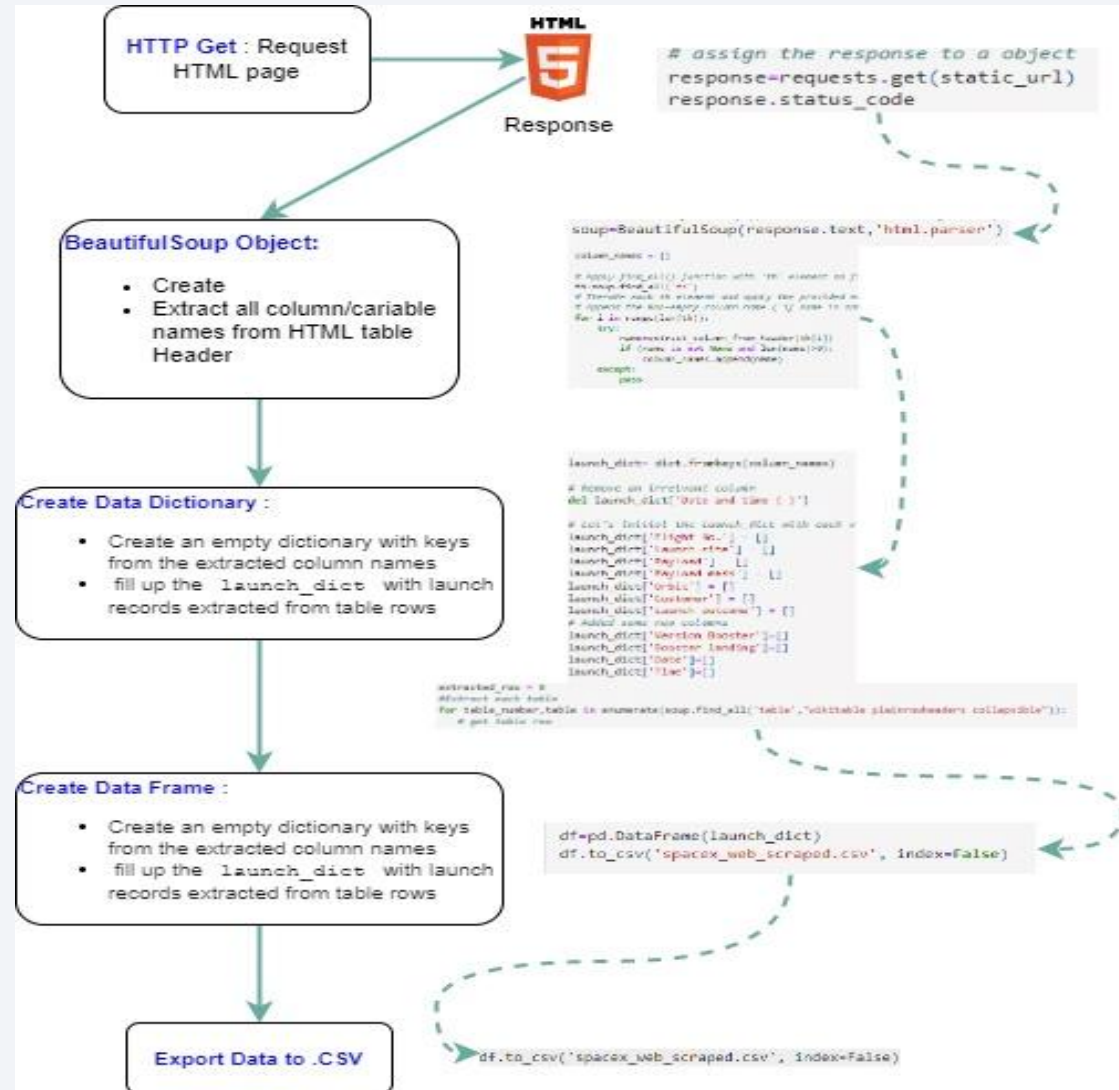
## 7. Export data to .csv

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

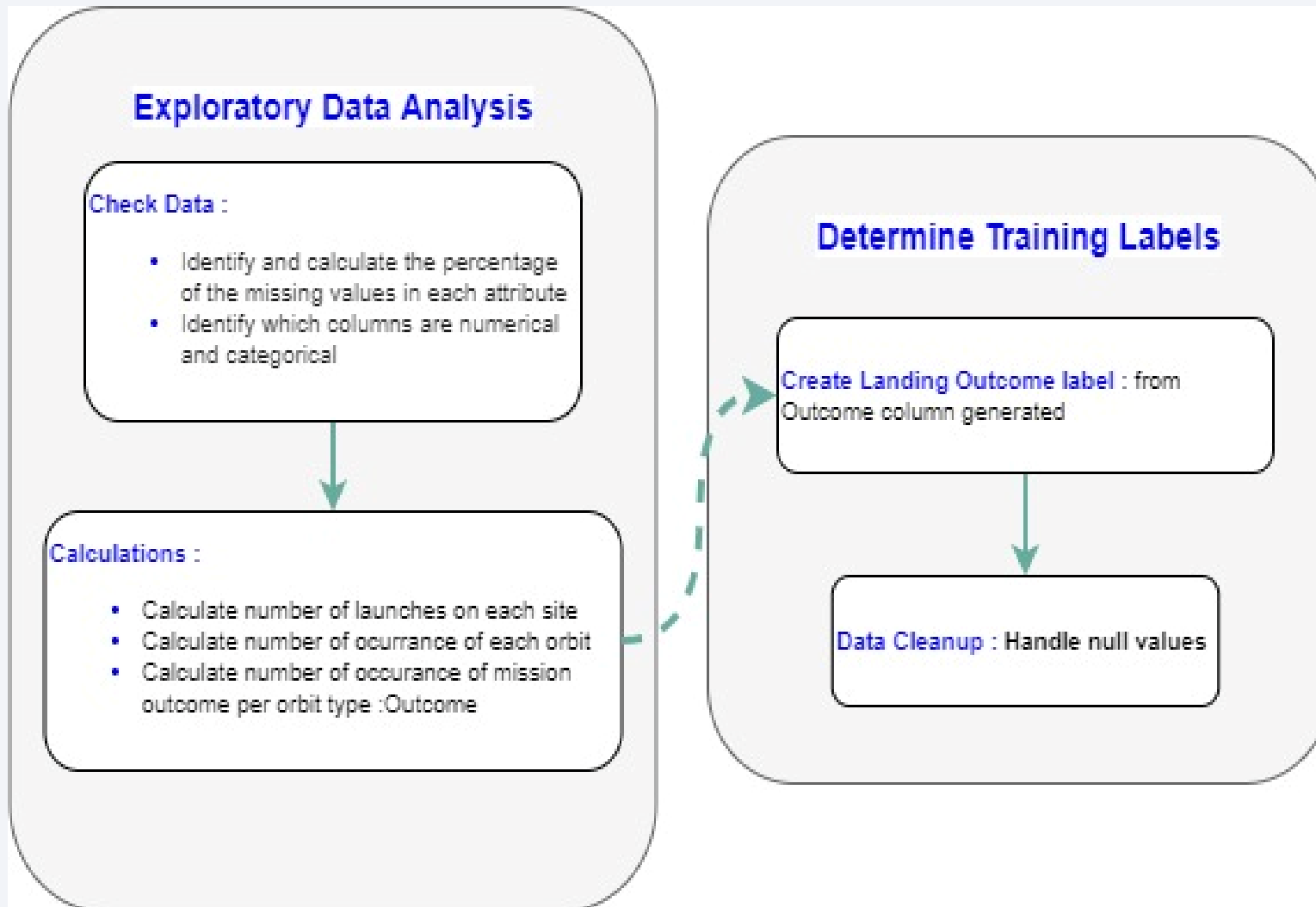


# Data Collection Scrapping

- Web Scrapping from Wikipedia
- GitHub URL for the complete code of SpaceX Web Scrapping  
<https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/O2-DataCollectionWebScrapping.ipynb>

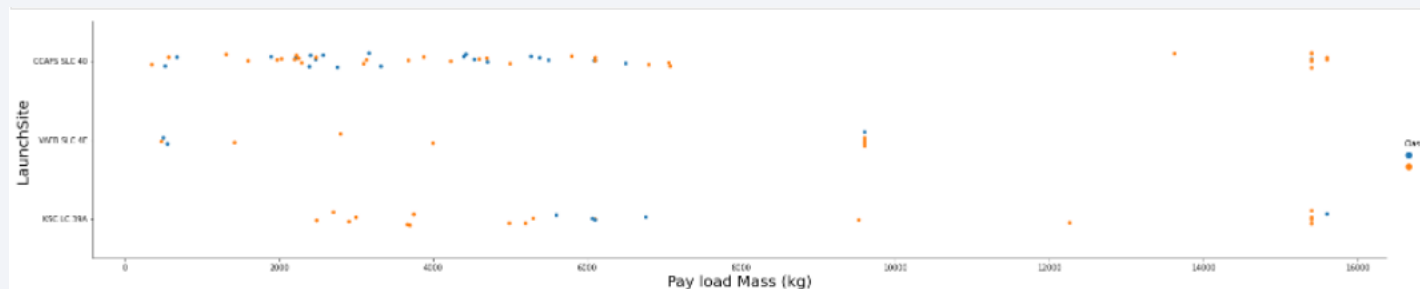
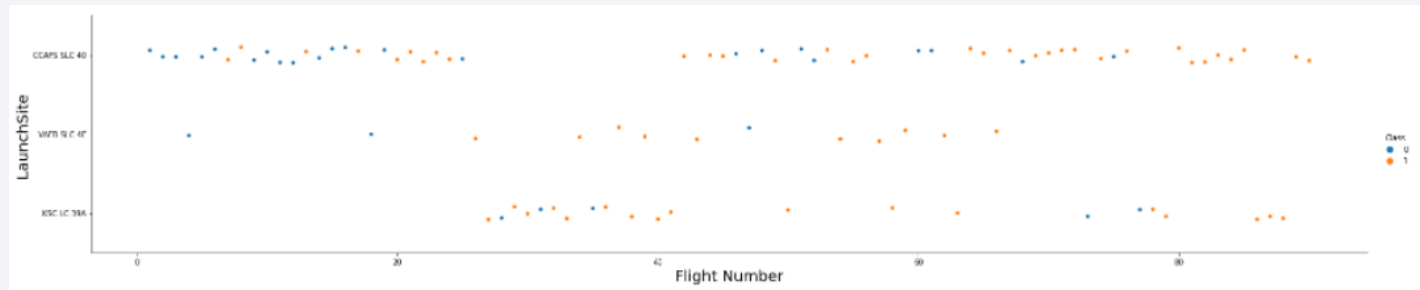
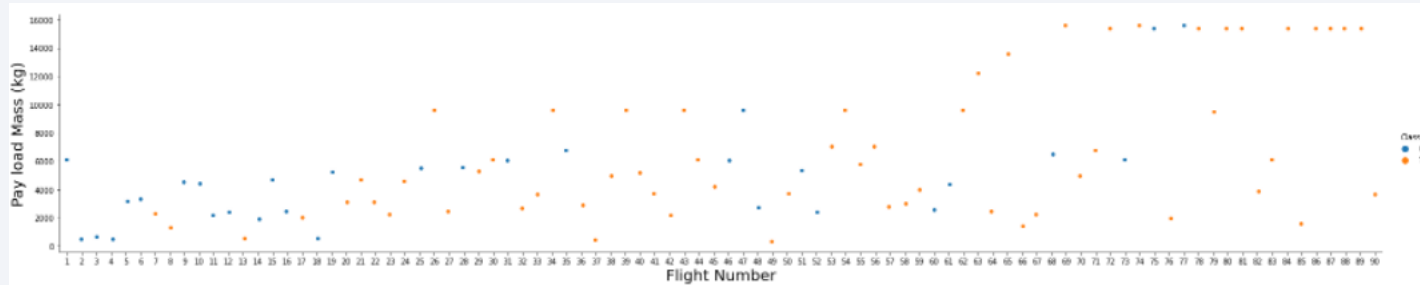


# Data Wrangling



- Data Wrangling steps
- GitHub URL for the complete code of SpaceX Data Wrangling  
<https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/O3-DataCollectionDataWrangling.ipynb>

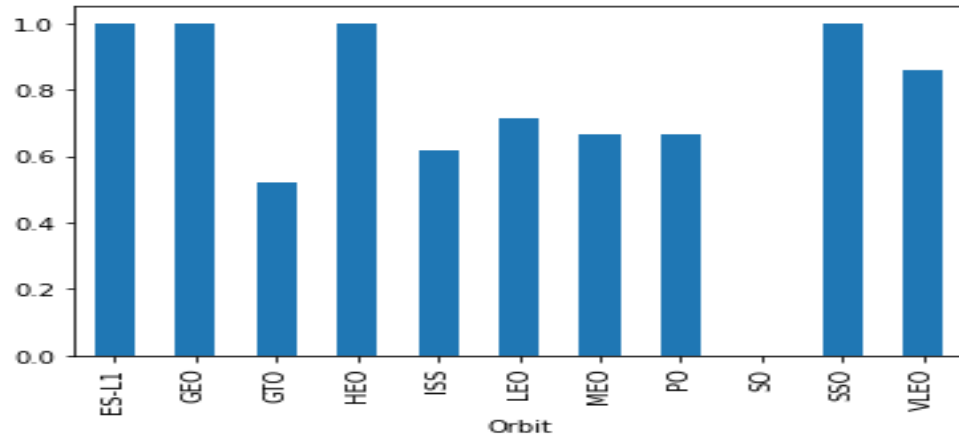
# EDA with Data Visualization



Scatter graph to find the relationship between the attributes such as between:

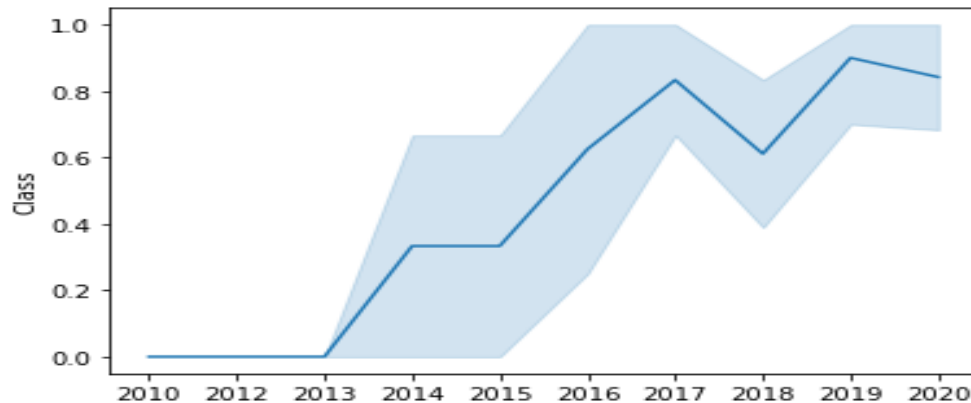
- Payload and Flight Number.
- Flight Number and Launch Site.
- Payload and Launch Site.
- Flight Number and Orbit Type.
- Payload and Orbit Type

# EDA with Data Visualization



Bar graph to determine which orbits have the highest probability of success

Line graph to show a trends or pattern of the attribute over time which in this case, is used for see the launch success yearly trend.



Next, obtain some preliminary insights about impact on success rate by each important variable. This will help to select the features that will be used in success prediction.

GitHub URL for the complete code of Data Visualization supported EDA <https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/05-EDADDataVisualization.ipynb>

# EDA with SQL

---

## Using SQL to get better understanding of the dataset:

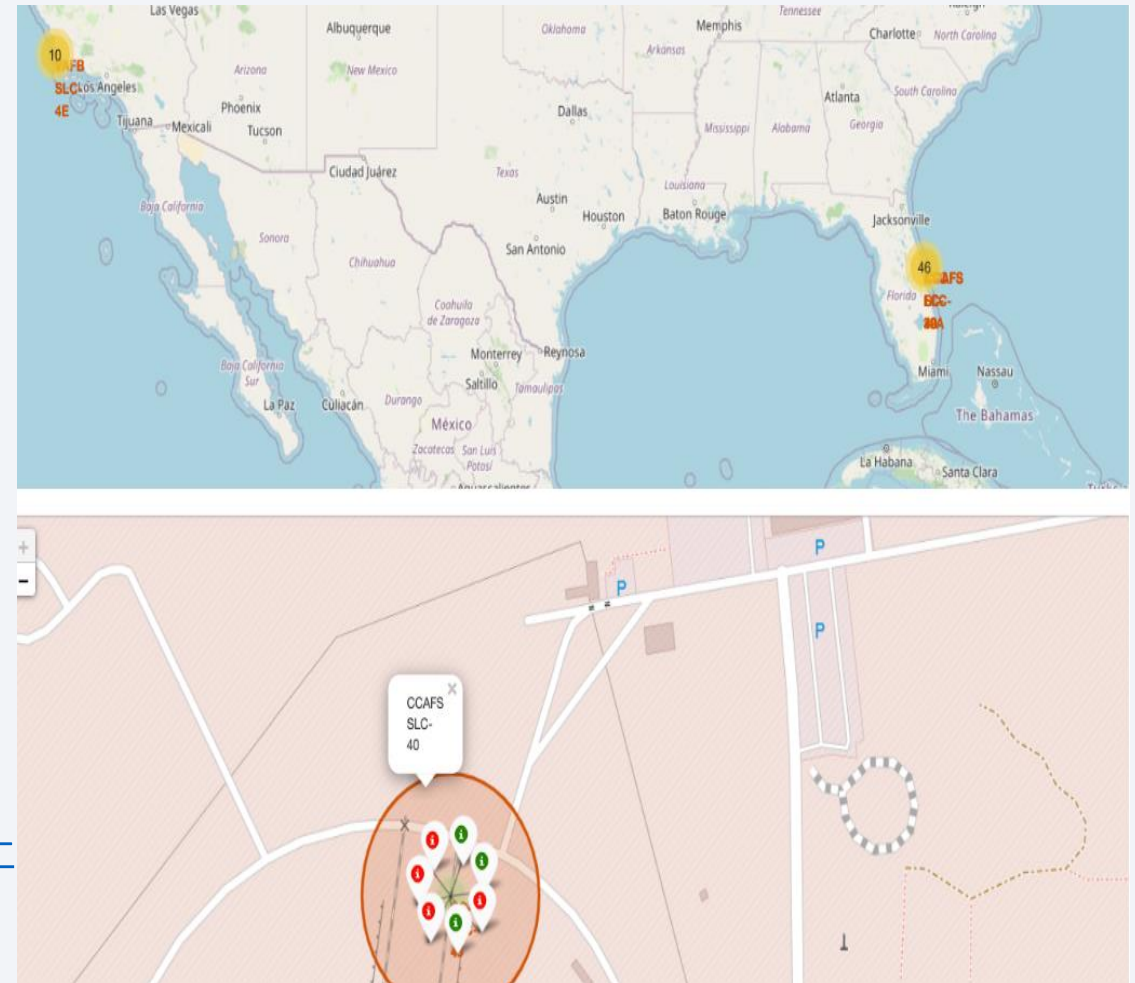
- Displaying the names of the launch sites.
  - Displaying 5 records where launch sites begin with the string 'KSC'.
  - Displaying the total payload mass carried by booster launched by NASA (CRS).
  - Displaying the average payload mass carried by booster version F9 v1.1.
  - Listing the date when the first successful landing outcome in drone ship was achieved.
  - Listing the names of the boosters which have success in drone ship 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 leveraging subquery.
  - List the records which will display the month names, succesful landing\_outcomes in ground pad ,booster versions, launch\_site for the months in year 2017.
  - Rank the count of successful landing\_outcomes between the date 2010-06-04 and 2017-03-20 in descending order.
- 
- GitHub URL for the complete code of SQL supported EDA

<https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/04-EDASQL.ipynb>



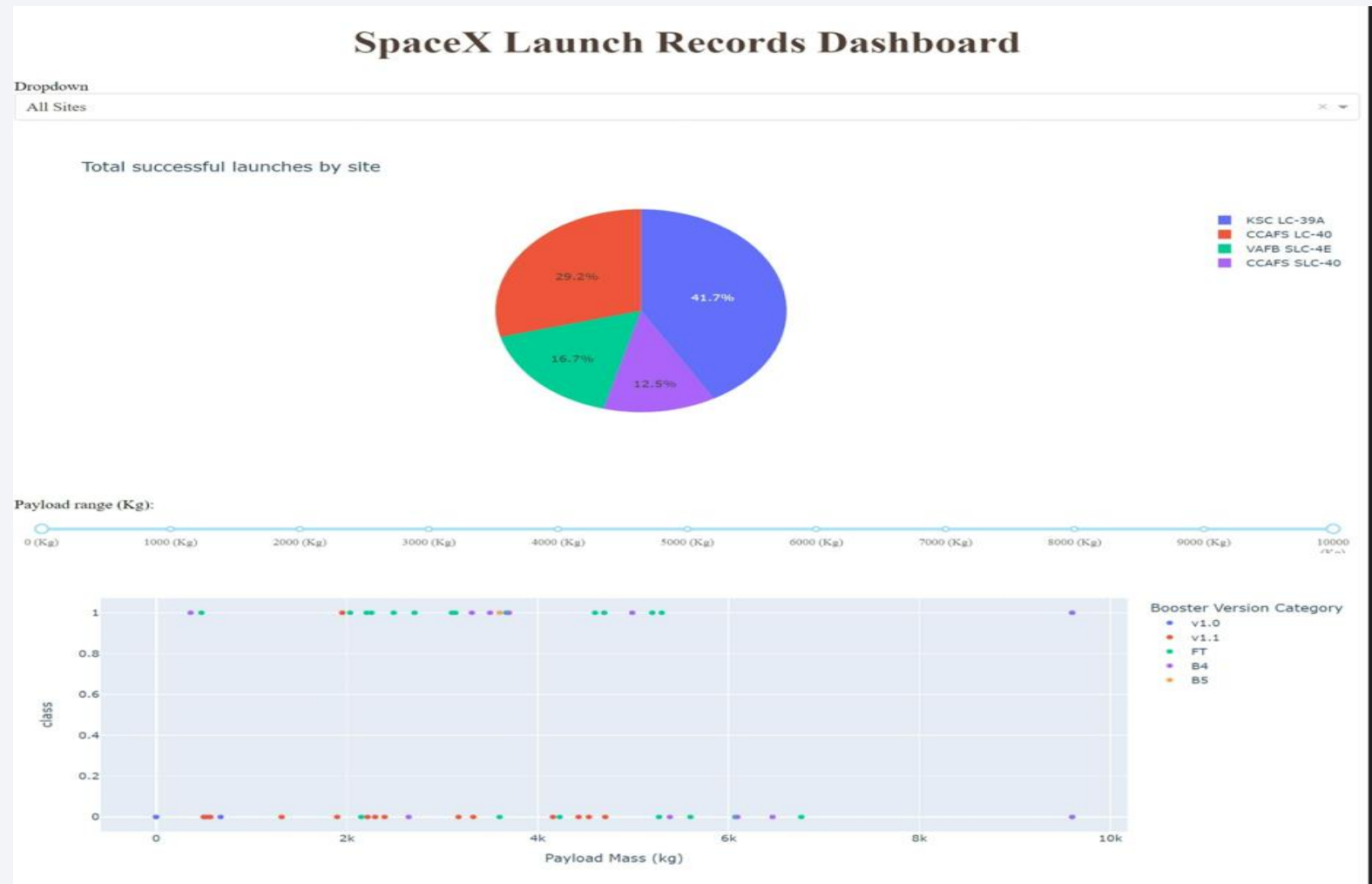
# Build an Interactive Map with Folium

- Circles and Markers were added for each launch site on the site map to highlight the launch sites
- The Launch Outcomes were added to the map with color coding to easily identify which launch sites have relatively high success rates
- Lines are drawn on maps to measure the distance to landmarks to find various trends as :
  - Are launch sites in close proximity to railways? No
  - Are launch sites in close proximity to highways? No
  - Are launch sites in close proximity to coastline? Yes
  - Do launch sites keep certain distance away from cities? Yes
- GitHub URL for the complete code for Interactive Map with Folium <https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/06-VisualAnalyticsFolium.ipynb>



# Build a Dashboard with Plotly Dash

- A pie chart to show the total successful launches count for all sites. If a specific launch site was selected, show the Success vs. Failed counts for the site.
- A callback function for `site-dropdown` and `payload-slider` as inputs, `success-payload-scatter-chart` as output
- GitHub URL for the complete code for Plotly Dashboard  
[https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/07-InteractiveDashboardPlotly\\_app.py](https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/07-InteractiveDashboardPlotly_app.py)



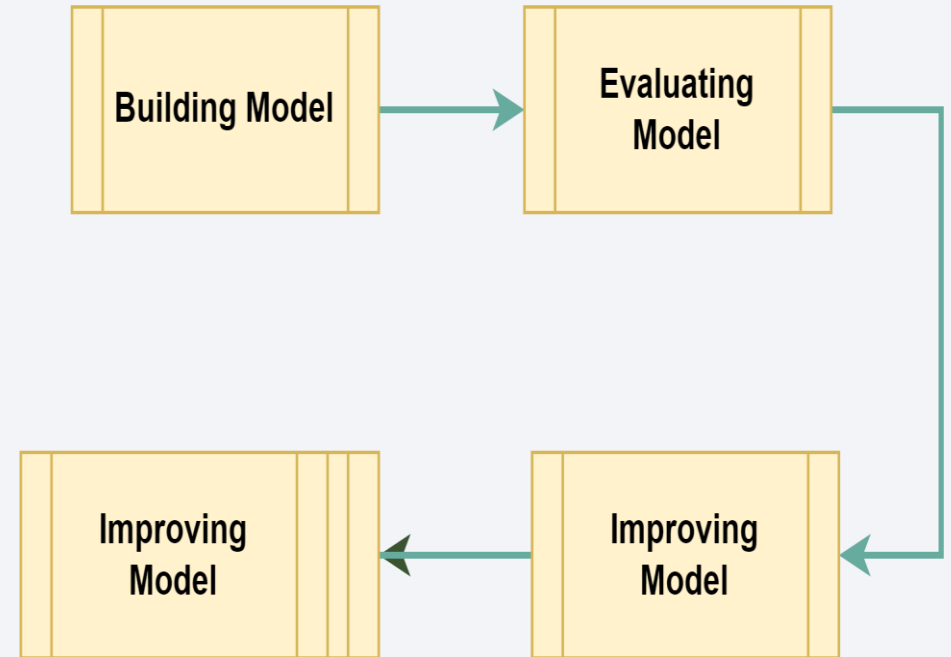
# Predictive Analysis (Classification)

---

Using the best hyperparameter values, the model with the best accuracy using the training data is determined.

The following tests are done : Logistic Regression, Support Vector machines, Decision Tree Classifier, and K-nearest neighbors. Finally, the confusion matrix is produced.

- GitHub URL for the complete code for Classification  
<https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/08-MLPredictiveAnalysisClassification.ipynb>



# Results

---

- SVM, KNN and Logistic Regression models are the best for prediction accuracy for this dataset
- Low weighted payloads perform better than the heavier payloads
- KSC LC 39A has the most successes from all sites
- Orbit GEO, HEO, SSO , ES L1 has the best success rates



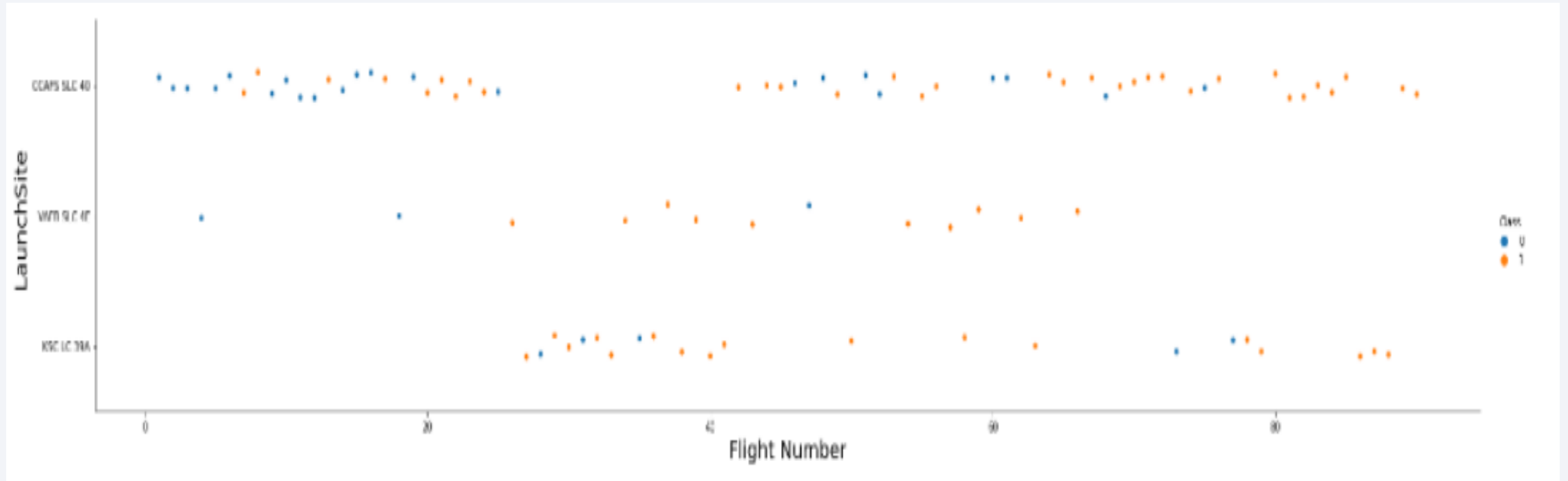


Section 2

# Insights drawn from EDA

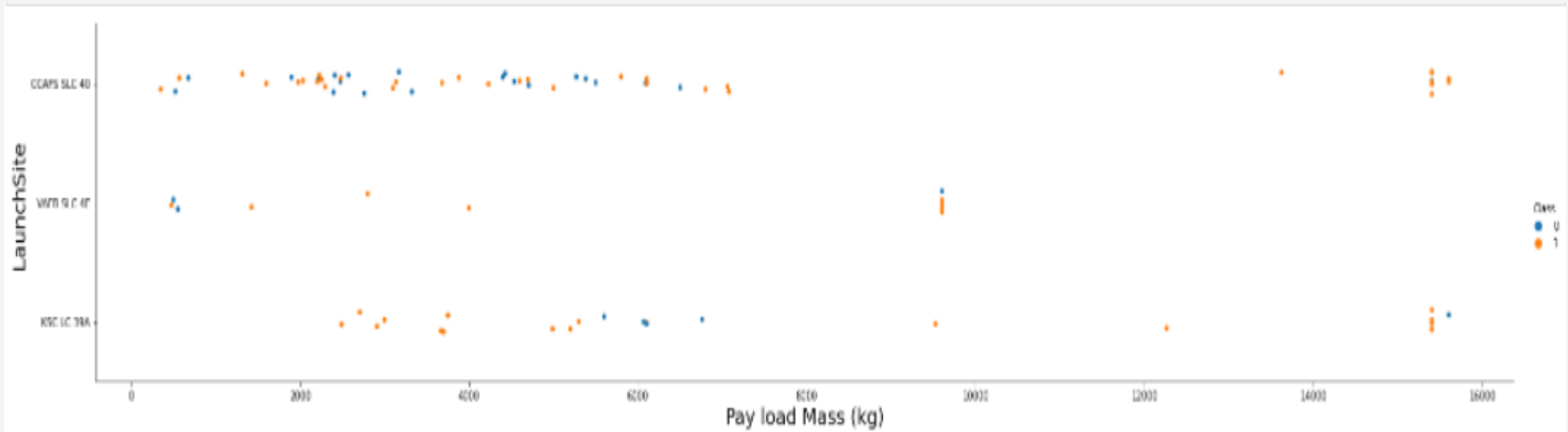


# Flight Number vs. Launch Site



- Launches from the site CCAFS SLC 40 are slightly higher than other sites

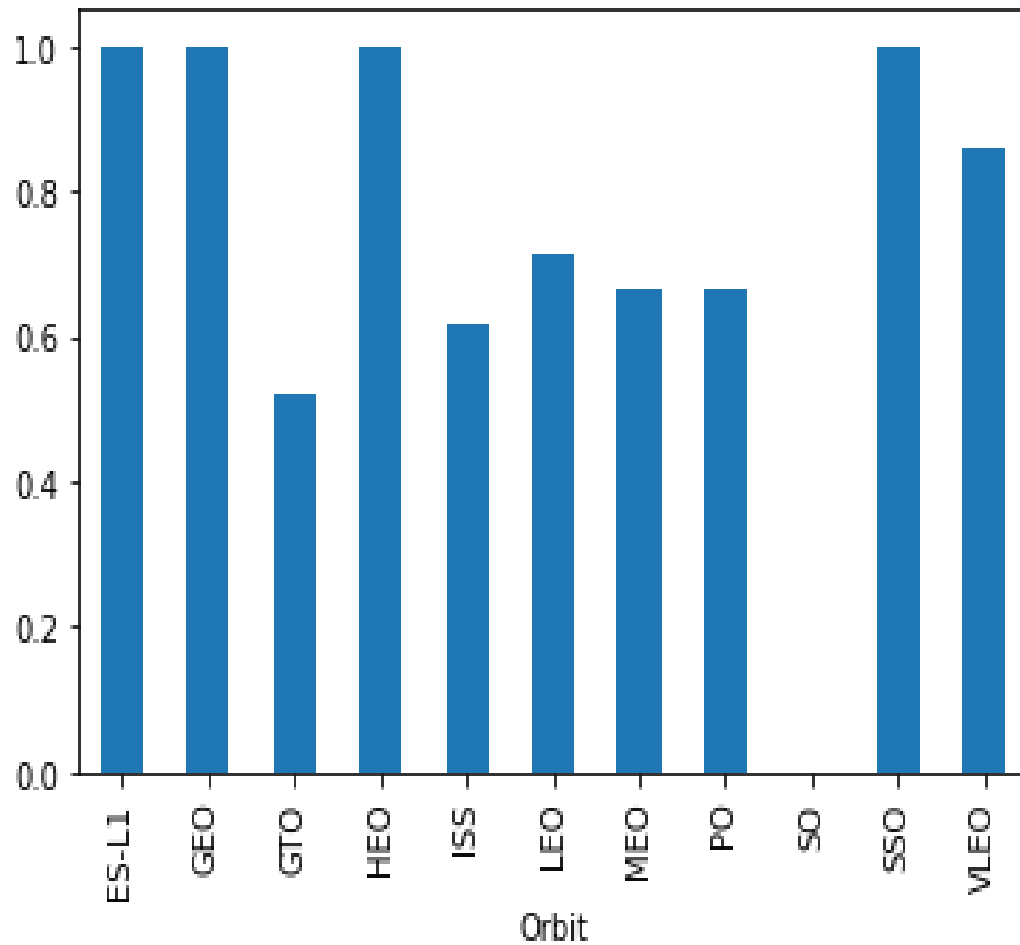
# Payload vs. Launch Site



- Mostly lower mass Payloads have been launched from CCAFS SLC 40

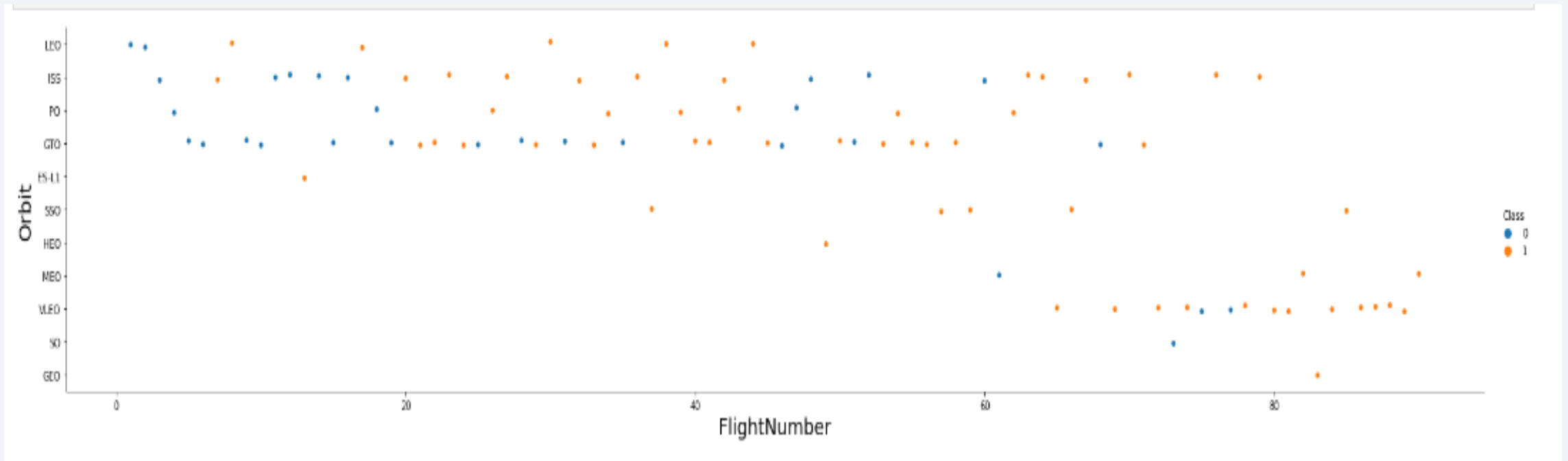
# Success Rate vs. Orbit Type

---



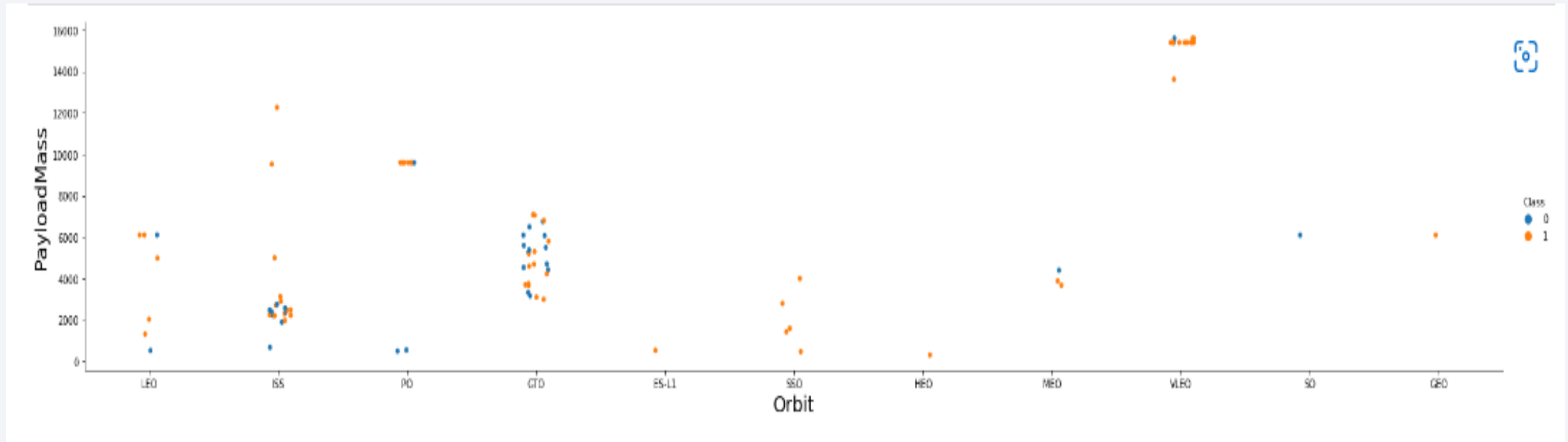
- Highest success rates are for orbit types of ES-L1, GEO, HEO, SSO

# Flight Number vs. Orbit Type



- In the LEO orbit the Success appears related to the number of flights; where as , there seems to be no relationship between flight number when in GTO orbit.

# Payload vs. Orbit Type

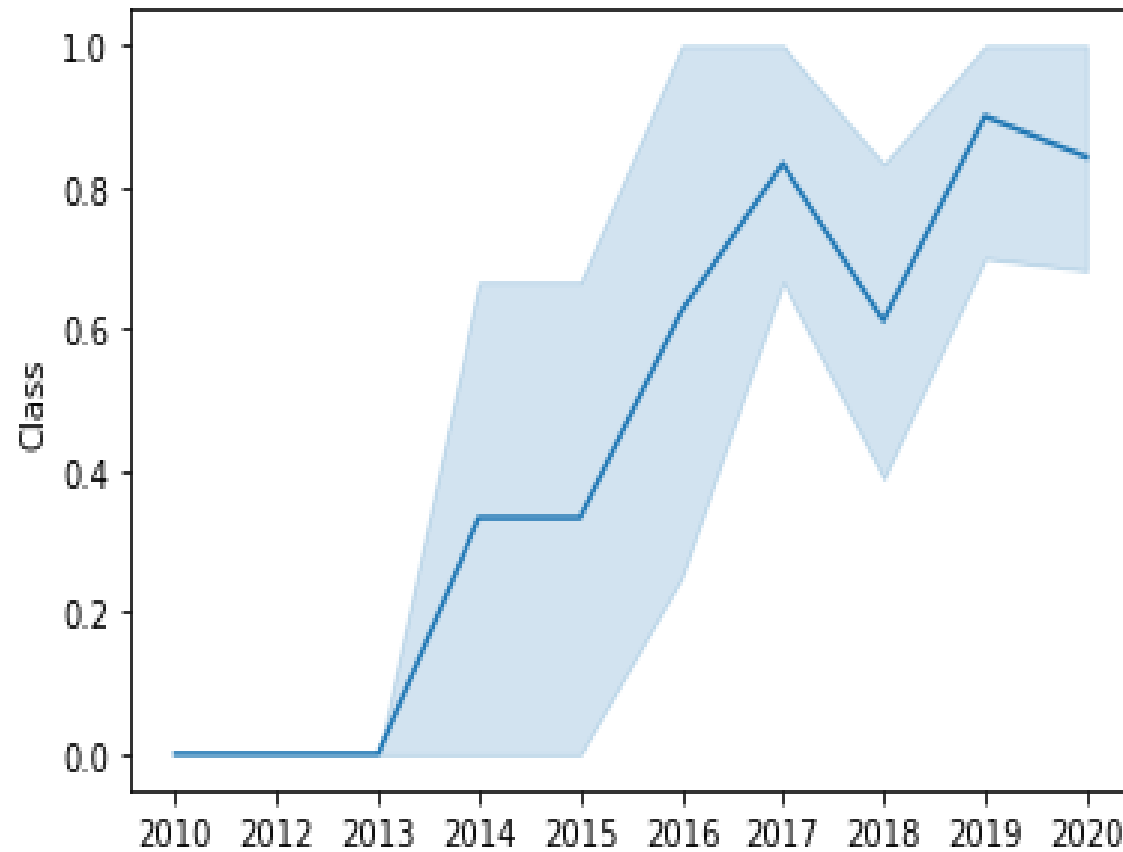


- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccesful mission) are both there here.



# Launch Success Yearly Trend

---



- The success rate since 2013 kept increasing till 2020

# All Launch Site Names

---

```
%sql select distinct(launch_site) from spacex
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

```
: launch_site
```

```
CCAFS LC-40
```

```
CCAFS SLC-40
```

```
KSC LC-39A
```

```
VAFB SLC-4E
```

# Launch Site Names Begin with 'KSC'

```
%sql select * from spacex where launch_site like 'KSC%' limit 5
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/blddb  
Done.
```

DATE	time_utc	booster_version	launch_site	payload	payload_mass_kg	orbit	customer	mission_outcome	landing_outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	06:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

# Total Payload Mass

---

```
%sql select sum(PAYLOAD_MASS_KG_) from spacex where customer='NASA (CRS)'
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

```
1
```

```
45596
```

# Average Payload Mass by F9 v1.1

---

```
%sql select avg(PAYLOAD_MASS_KG_) from spacex where Booster_Version='F9 v1.1'
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

1

2928



# First Successful Drone Ship Landing Date

---

```
%sql select min(date) from spacex where Landing_Outcome='Success (drone ship)'
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

```
1
```

```
2016-04-08
```

## Successful Ground Pad Boosters with Payload between 4000 and 6000

```
%sql select Booster_Version from spacex where Landing_Outcome='Success (ground pad)' and PAYLOAD_MASS_KG >4000 and PAYLOAD_MASS_KG <6000
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
```

Done.

**booster\_version**

F9 FT B1032.1

F9 B4 B1040.1

F9 B4 B1043.1

# Total Number of Successful and Failure Mission Outcomes

```
%sql select Mission_Outcome, count(1) from spacex group by Mission_Outcome
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

mission\_outcome 2

Failure (in flight) 1

Success 99

Success (payload status unclear) 1

# Boosters Carried Maximum Payload

```
%sql select booster_version from spacex where PAYLOAD_MASS_KG = ( select max(PAYLOAD_MASS_KG) from spacex )
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
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 monthname(date),landing__outcome,booster_version,Launch_site from  
spacex where year(date)='2017' and landing__outcome ='Success (ground pad)'
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

1	landing__outcome	booster_version	launch_site
February	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
May	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
June	Success (ground pad)	F9 FT B1035.1	KSC LC-39A
August	Success (ground pad)	F9 B4 B1039.1	KSC LC-39A
September	Success (ground pad)	F9 B4 B1040.1	KSC LC-39A
December	Success (ground pad)	F9 FT B1035.2	CCAFS SLC-40

# Rank Landing Outcomes Between 20100604 and 20170320

---

```
%sql select landing__outcome, count(1) cnt from spacex where landing__outcome like  
'Success%' and date between '2010-06-04'and'2017-03-20' group by landing__outcome
```

```
* ibm_db_sa://ygn69416:***@125f9f61-9715-46f9-9399-c8177b21803b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb  
Done.
```

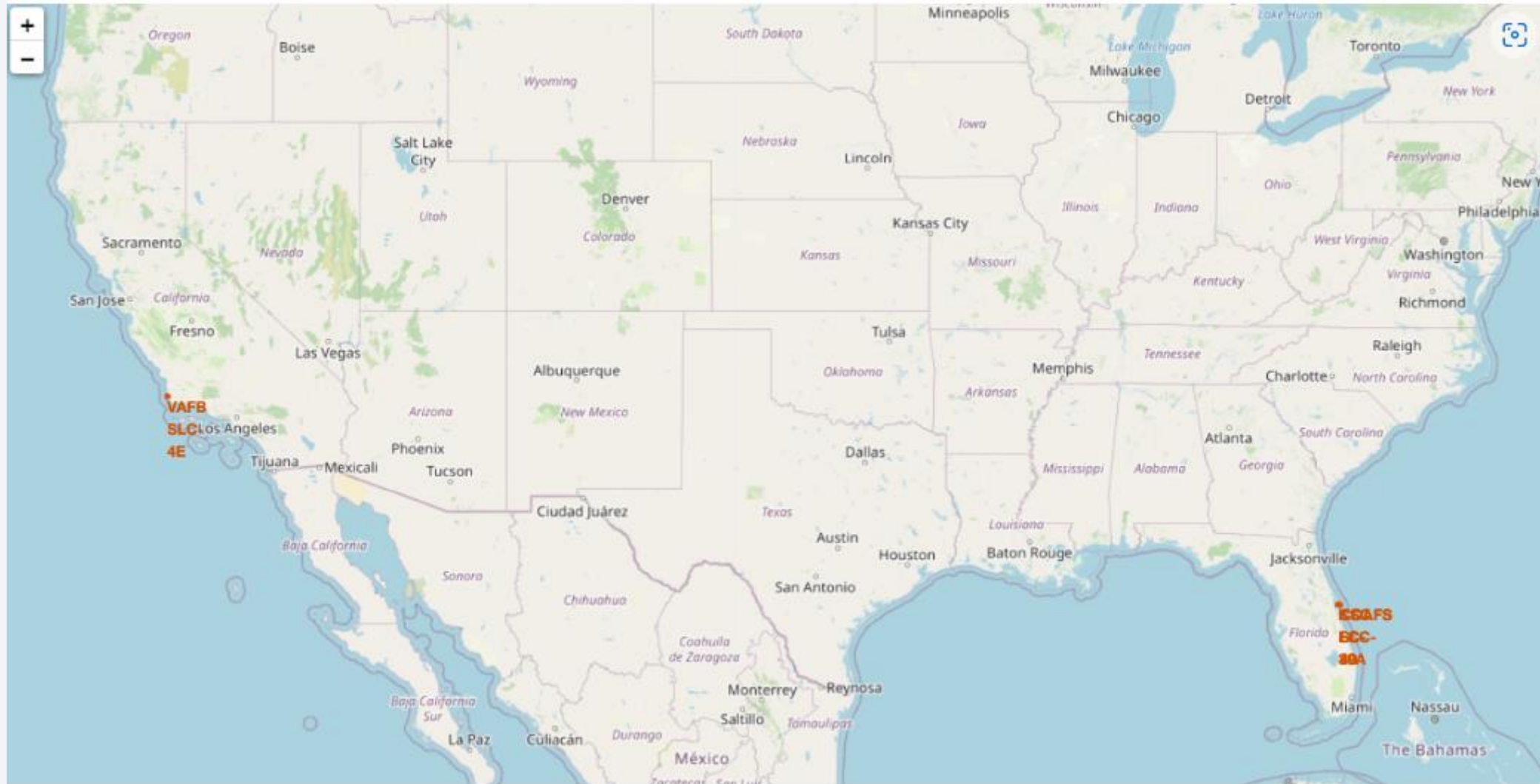
<u>landing_outcome</u>	cnt
Success (drone ship)	5
Success (ground pad)	3

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

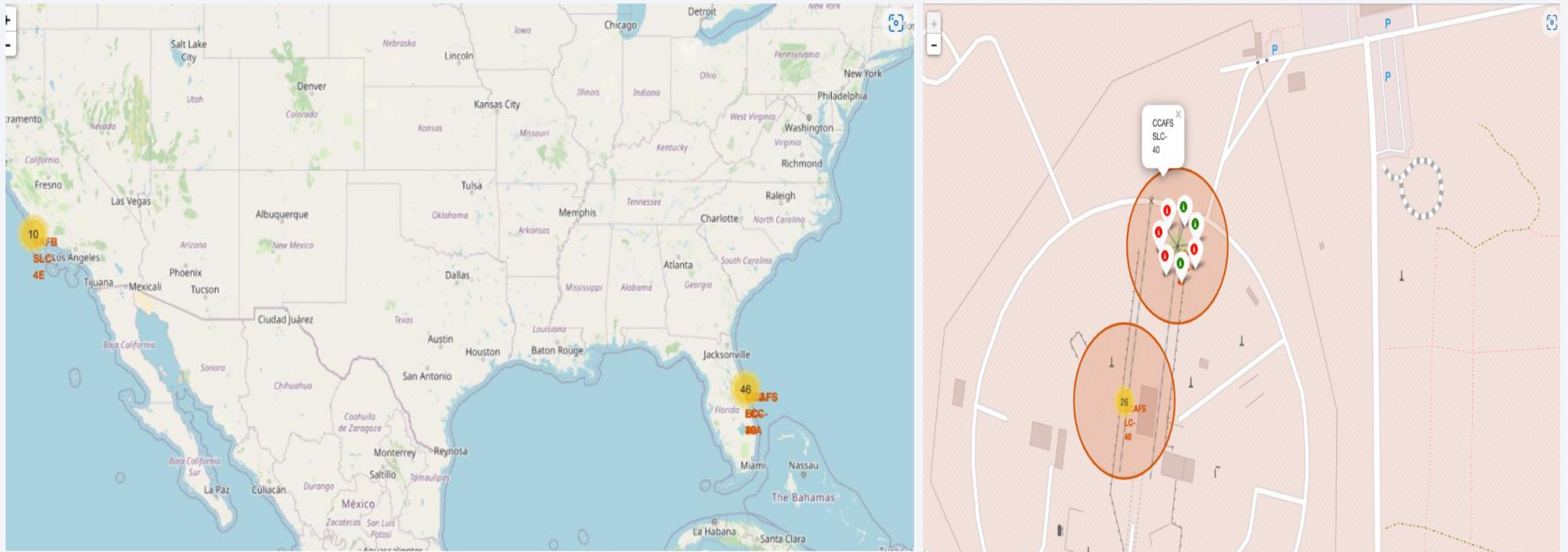
# Launch Sites Proximities Analysis

# Launch Sited Marked On A Map





# Success/Failed Launches for Each Site on the Map



From the color-labeled markers in marker clusters, one can easily identify which launch sites have relatively high success rates.

# Distances Between a Launch Site to its Proximities



Map with distance line

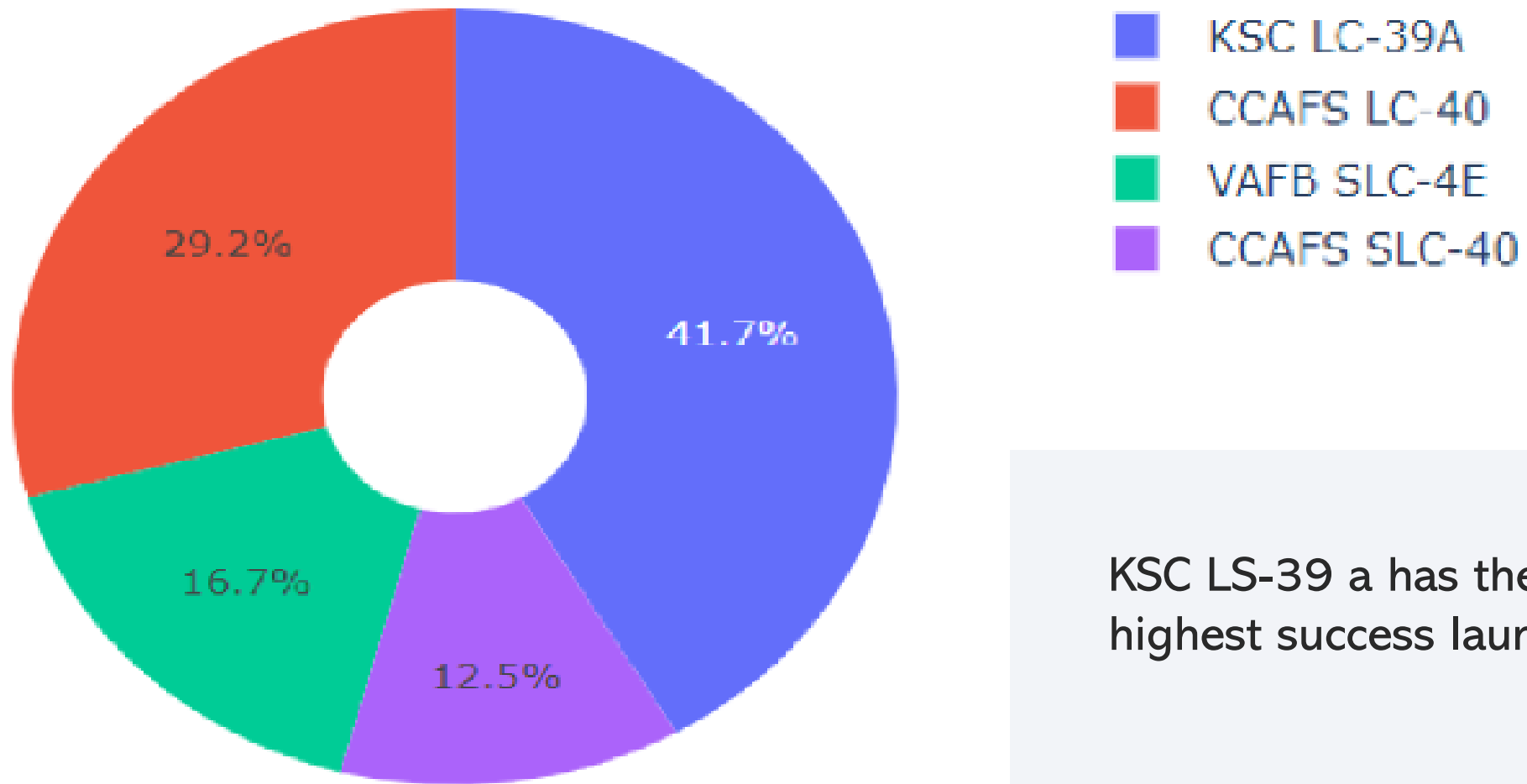




Section 4

# Build a Dashboard with Plotly Dash

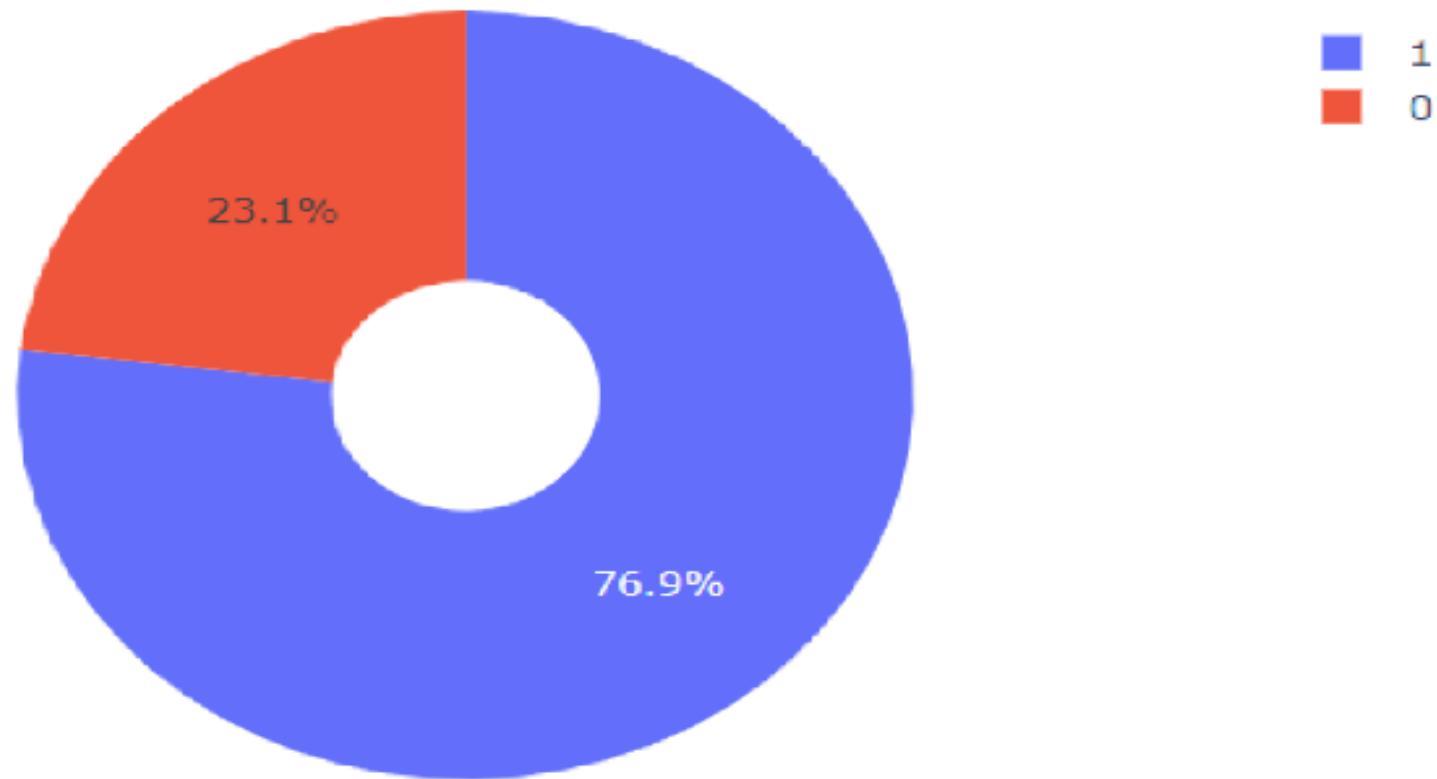
# Total Success Launches Site Wise



KSC LS-39 a has the highest success launches

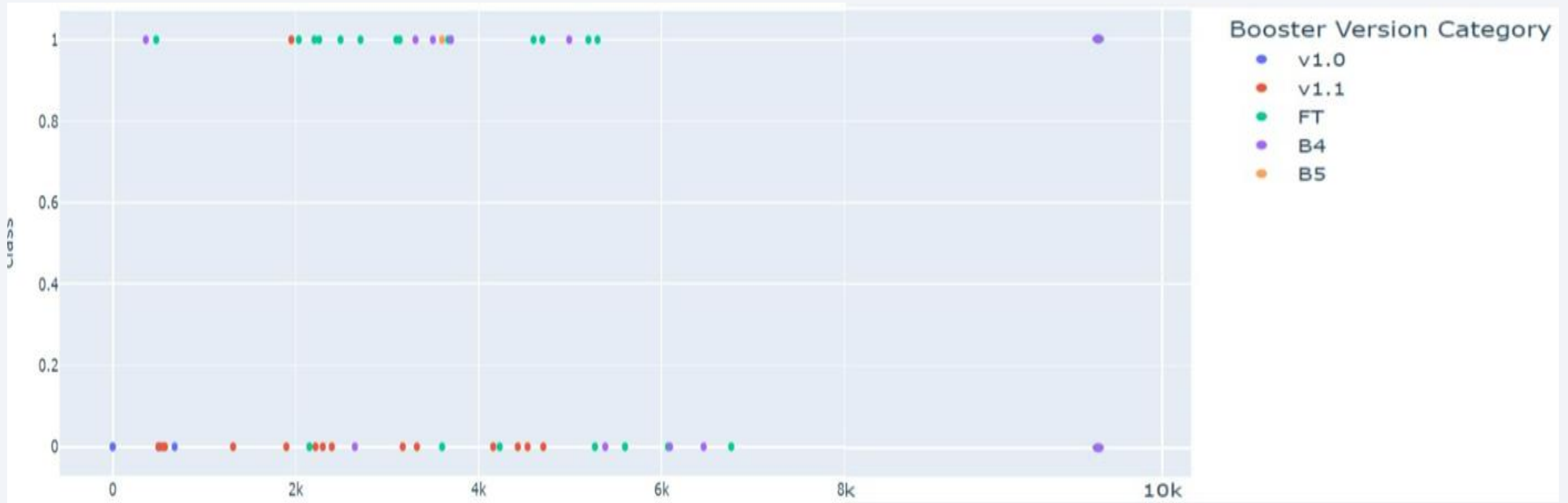
# Launch Site with highest Success Ratio

---



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

## <Dashboard Screenshot 3>



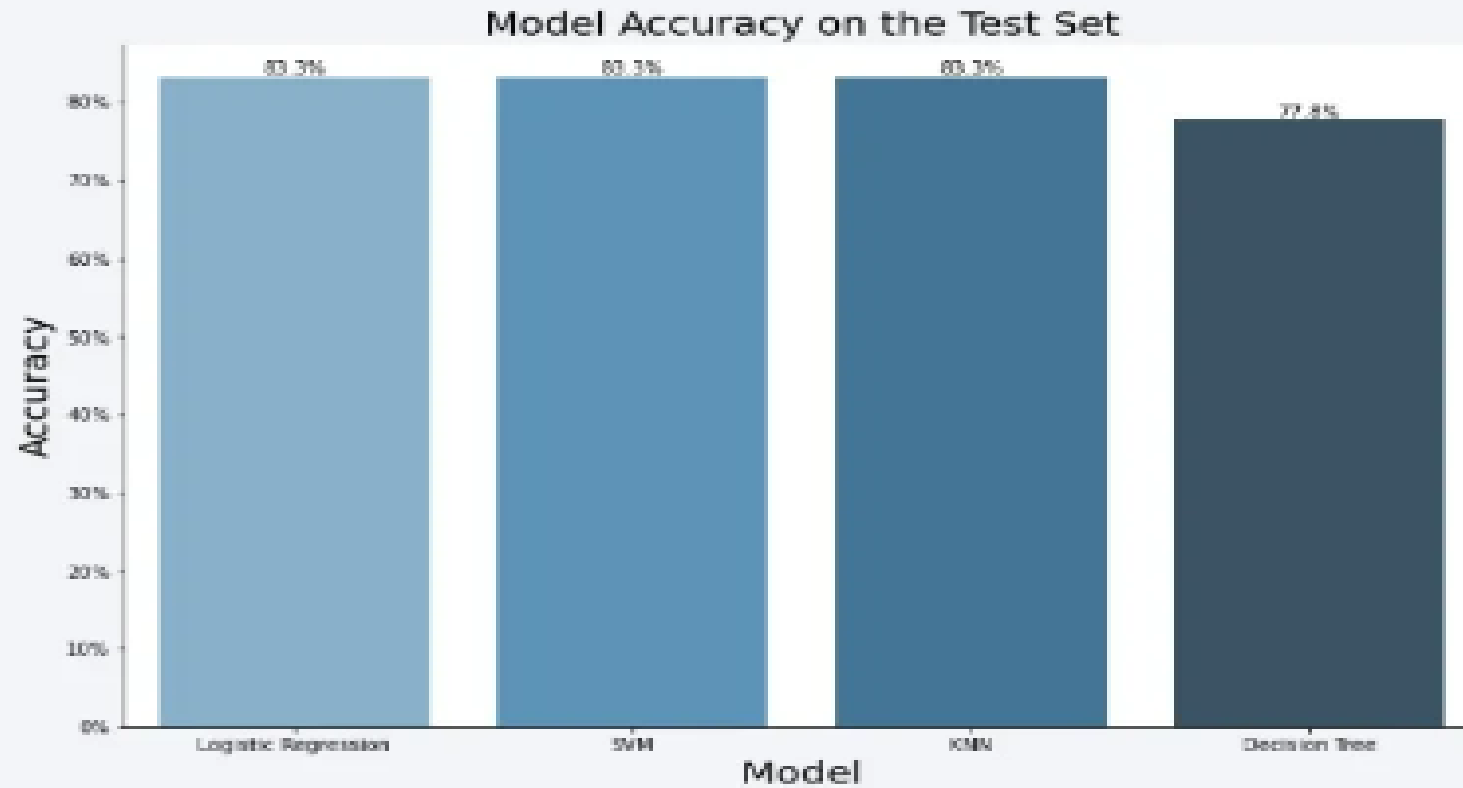
Low weighted payloads have higher success rates

Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

---





# Confusion Matrix



# Conclusions

---

- Decision Tree Algorithm is the best for this dataset for prediction
- Low weighted payloads have higher success rates than the heavy payloads
- With increasing years, success rates has increased. With more years, they will perfect the launches
- KSC LC – 39A has the best success launches
- Orbits with best success rates are – GEO, HEO, SSO, ES-L1

# Appendix

- Sample dataset\_part\_3.csv created for predictions : [https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/dataset\\_part\\_3.csv](https://github.com/sujasing/ds-ml-capstone-spacex/blob/main/dataset_part_3.csv)

FlightNum	PayloadMass	Flights	GridFins	Reused	Legs	Block	ReusedCore	Orbit_ES-L	Orbit_GE	Orbit_GTC	Orbit_HE	Orbit_ISS	Orbit_LEO	Orbit_ME	Orbit_PO	Orbit_SO	Orbit_SSO	Orbit_VLE
1	6104.959	1	FALSE	FALSE	FALSE	1	0	0	0	0	0	0	1	0	0	0	0	0
2	525	1	FALSE	FALSE	FALSE	1	0	0	0	0	0	0	1	0	0	0	0	0
3	677	1	FALSE	FALSE	FALSE	1	0	0	0	0	0	1	0	0	0	0	0	0
4	500	1	FALSE	FALSE	FALSE	1	0	0	0	0	0	0	0	0	1	0	0	0
5	3170	1	FALSE	FALSE	FALSE	1	0	0	0	1	0	0	0	0	0	0	0	0
6	3325	1	FALSE	FALSE	FALSE	1	0	0	0	1	0	0	0	0	0	0	0	0
7	2296	1	FALSE	FALSE	TRUE	1	0	0	0	0	0	1	0	0	0	0	0	0
8	1316	1	FALSE	FALSE	TRUE	1	0	0	0	0	0	0	1	0	0	0	0	0
9	4535	1	FALSE	FALSE	FALSE	1	0	0	0	1	0	0	0	0	0	0	0	0
10	4428	1	FALSE	FALSE	FALSE	1	0	0	0	1	0	0	0	0	0	0	0	0
11	2216	1	FALSE	FALSE	FALSE	1	0	0	0	0	0	1	0	0	0	0	0	0
12	2395	1	TRUE	FALSE	TRUE	1	0	0	0	0	0	1	0	0	0	0	0	0
13	570	1	TRUE	FALSE	TRUE	1	0	1	0	0	0	0	0	0	0	0	0	0
14	1898	1	TRUE	FALSE	TRUE	1	0	0	0	0	0	1	0	0	0	0	0	0

- Project Code Repository : <https://github.com/sujasing/ds-ml-capstone-spacex>

Thank you!

