IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Naincy
17-01-2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- In this report, I have designed a model that will predict if the Falcon 9 will land successfully or not. We have used the SpaceX data since there are much cost effective landings because it reuses the first stage. Therefore if we can determine if the first stage will land, we can determine the cost of a launch. We collected the data from spaceX API and Wikipedia page. After collecting data for Falcon 9, data is cleaned, wrangled, handled carefully and visualized to fully understand the main parameters that will affect the landing. Booster Version, Launch site, Payload, Latitude, Longitude and Class works out to be the main parameters to predict the landing.
- After the Data visualisation, we found out that the launch sites have a certain distance from cities and highways but they are relatively close to railways, this does produce a opinion that launch site may contain some harmful launching effects that can cause disturbances in cities. We found out the launch sites which has largest number of successful launches and which has the highest number of success rate.
- After all the findings, a predictive analysis is done in which there are a number of machine learning algorithms which are trained to analyze which machine learning algorithm is best for this prediction. There are four machine learning algorithms used in this lab since it is a categorical prediction we have used classification algorithms: Logistic Regression, Support Vector Machine, K-Nearest Neighbor and Decision Trees.
- Out of these four algorithms, the best performing model is KNN with a accuracy of 83% and other models have accuracy of below 83%.

Introduction

- There are a lot companies who works for launching the rocket but the launching costs a lot since it requires a new first stage at every launch.
- SpaceX has gained worldwide attention for a series of historic milestones.
- It is the only private company ever to return a spacecraft from low-earth orbit, which it first accomplished in December 2010. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars whereas other providers cost upward of 165 million dollars each, much of the savings is because Space X 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 space X for a rocket launch.
- The cost of launch depends on the landing of the first stage on various platforms like sea, land etc. We will use every platform and connect it for our data findings to determine if the landing is done successfully or not for the first stage.
- In this lab, we will create a machine learning pipeline to predict if the first stage will land successfully or not using data from the Space X API and wikipedia.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- Describe how data sets were collected. Data for this project is collected from Space X API using request.get method and from Wikipedia using request.get and web scraping.
- From Space X, we will get the Falcon 9 data by filtering through the it and then deal with missing/NA values in this dataframe.
- We will collect Falcon 9 historical launch records from a Wikipedia page titled “List of Falcon 9 and Falcon Heavy launches” through web scraping. We will perform the HTTP GET method to request the Falcon 9 Launch HTML page and then scrape the data through each row to extract the important columns using Beautiful Soup.

Data Collection – SpaceX API

- Here, using the requests.get method, we get the data from space X API. To make the requested JSON results more consistent, we will use the following static response object. Then, Decode the response content as a Json using **.json()** and turn it into a Pandas dataframe using **.json_normalize()**.
- You can reach out to the SpaceX API calls notebook on this GitHub URL here: https://github.com/naincyv/Data-Science-Project/blob/main/spacex-data-collection-api.ipynb

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

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

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
```

```
response=requests.get(static_json_url)
response = response.json()
```

```
data = pd.json_normalize(response)
```

Data Collection - Scraping

- Here, we will request the HTML page from the wikipedia URL and perform a HTTP GET method to request the Falcon 9 Launch HTML page, as an HTTP response. Then, Create a BeautifulSoup object from the HTML response. We will find all tables on the wiki page first and then find the third table on the page.

- You can reach out to the web scraping notebook, on this Github URL: https://github.com/naincyv/Data-Science-Project/blob/main/webscraping.ipynb

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

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

```
bf = BeautifulSoup(response.text, 'html.parser')
html_tables = bf.find_all('table')
```

```
first_launch_table = html_tables[2]
```

Data Wrangling

- After collecting the data, we will process it and make it useful for our model to predict the landing. This process is done to find out the patterns in data and determine what would be the label for training supervised models.
- In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. Similarly, there are other terms: True RTLS(ground pad), False RTLS , True ASDS(drone ship) and False ASDS.
- Here, we mainly converted those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.
- Firstly, we will read CSV data through pandas and determine the number of launches on each Launch site, each Orbit, Outcomes.
- Using the Outcome, we will create a list where the element is zero if the corresponding row in Outcome is in the set bad_outcome; otherwise, it's one. Then assign it to the variable ‘class’. This class variable will represent the classification variable that represents the outcome of each launch.

- You can reach out to the GitHub URL of the complete data wrangling notebook here: https://github.com/naincyv/Data-Science-Project/blob/main/Data%20wrangling.ipynb

EDA with Data Visualization

- Firstly we will see that how the Flight Number (indicating the continuous launch attempts) and Payload variables would affect the launch outcome. We can plot out the Flight Number vs. Payload Mass and overlay the outcome of the launch. We will then plot the Flight Number and Payload Mass with respect to LaunchSite and use the hue as 'class'. We will then analyze the plotted bar chart to identify which orbits have the highest success rates. We will also plot the relationship between FlightNumber and Orbit type, Payload Mass and Orbit type and visualize the launch success yearly trends.
- By now, we should obtain some preliminary insights about how each important variable would affect the success rate, we will select the features that will be used in success prediction.
- Then, we will move forward to Feature Engineering and use the function get_dummies and features dataframe to apply OneHotEncoder to the column Orbits, LaunchSite, LandingPad, and Serial.
- You can reach out to the GitHub URL of the completed EDA with data visualization notebook for your reference: https://github.com/naincyv/Data-Science-Project/blob/main/data-visualisation.ipynb

EDA with SQL

- We can also use SQL to perform Exploratory Data Analysis.
- To do this, we have to first establish a connection with the database and load the dataset into the corresponding table in a Db2 database. Data is analysed and understood by displaying various columns using various filters such as displaying the name of Unique launch sites, display records where launch sites begin with the string 'CCA', total payload mass carried by boosters launched by NASA (CRS), average payload mass carried by booster version F9 v1.1, the date when the first successful landing outcome in ground pad was achieved, names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000 etc.

- You can view all the queries in my notebook, open this GitHub URL of my completed EDA with SQL notebook: https://github.com/naincyv/Data-Science-Project/blob/main/eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- The launch success rate may depend on various factors such as payload mass, orbit type, and so on. It may also depend on the location and proximities of a launch site, i.e., the initial position of rocket trajectories. Finding an optimal location for building a launch site certainly involves many factors and hopefully we could discover some of the factors by analyzing the existing launch site locations.
- Here we will be performing more interactive visual analytics using Folium.
- Folium is used to understand the data more deeply and easily, it gives the map view of our location and their features.
- Here, we will mark all launch sites on a map, mark the success/failed launches for each site on the map and calculate the distances between a launch site to its proximities.
- After building these maps, we will be able to find some geographical patterns about launch sites.
- We will create and add folium.Circle and folium.Marker to add a highlighted circle area with a text label on a specific coordinate. Using this we will add a circle for each launch site in data frame launch_sites.
- Also remember that a launch only happens in one of the four launch sites, which means many launch records will have the exact same coordinate. Marker clusters can be a good way to simplify a map containing many markers having the same coordinate
- You can reach out to this Github URL for the complete interactive map with Folium map : https://github.com/naincyv/Data-Science-Project/blob/main/launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- After all the data collection and visualisation, we will now present our data using Plotly Dash, and will be building a Plotly Dash application for users to perform interactive visual analytics on SpaceX launch data in real-time.
- This dashboard application contains input components such as a dropdown list and a range slider to interact with a pie chart and a scatter point chart.

- With the dashboard completed, we will be able to use it to analyze SpaceX launch data and have some findings such as KSC LC-39A site has the largest successful launches, KSC LC-39A has the highest launch success rate and F9 Booster version FT, B4 has the highest launch success rate.

- You can reach to this GitHub URL of my completed Plotly Dash lab, as an external reference: https://github.com/naincyv/Data-Science-Project/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- We will now create a machine learning pipeline to predict if the first stage will land given the data by first standardizing the data, splitting into training data and test data, finding best Hyperparameter for SVM, Classification Trees and Logistic Regression, and finally finding the method that performs best using test data.
- We will first load the data, standardize the data using **StandardScaler()** method of preprocessing and splitting the data into train and test using **train_test_split()**.
- We will then work with each algorithm, create a **logistic regression/SVM/KNN/tree** object then create a **GridSearchCV** object and then fit the object to find the best parameters from the dictionary parameters.
- After all the analysis with the models, the model that works out best with test data is the **Decision Tree model** with the accuracy of **87.67%.**
- You can open this GitHub URL of my completed predictive analysis lab, as an external reference: https://github.com/naincyv/Data-Science-Project/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

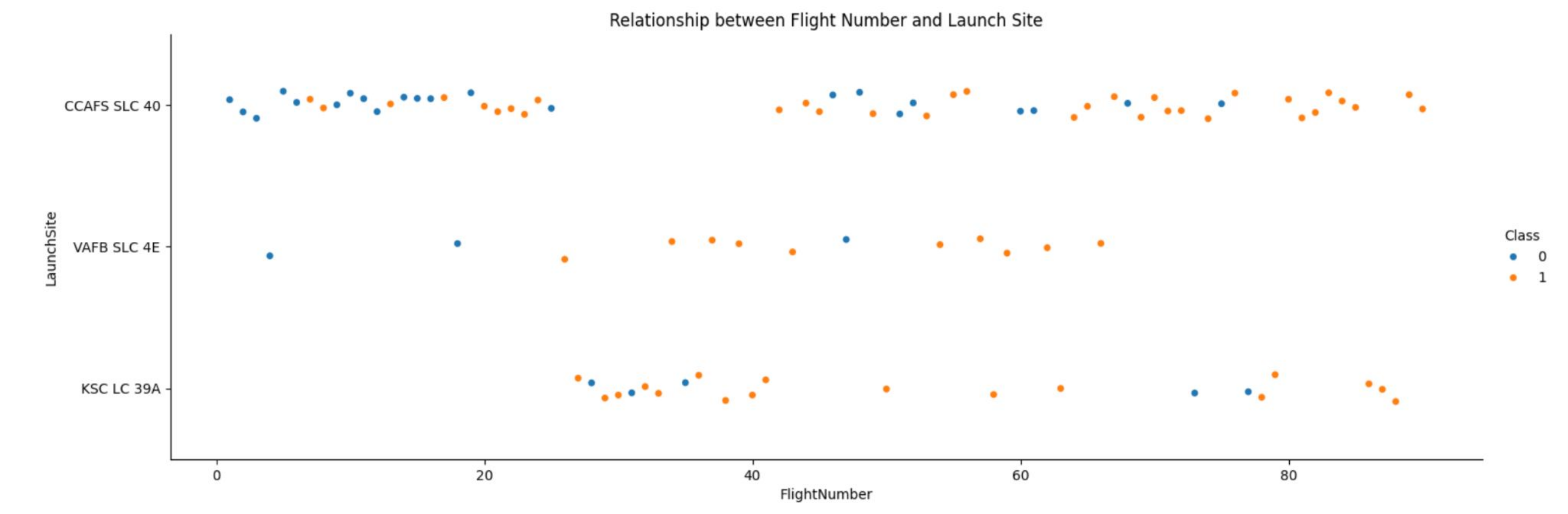
- From EDA analysis, we now know how each important variable would affect the success rate, we have selected the features “Booster Version, Launch site, Payload, Latitude, Longitude and Class” that will be used in success prediction that will affect the landing.
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

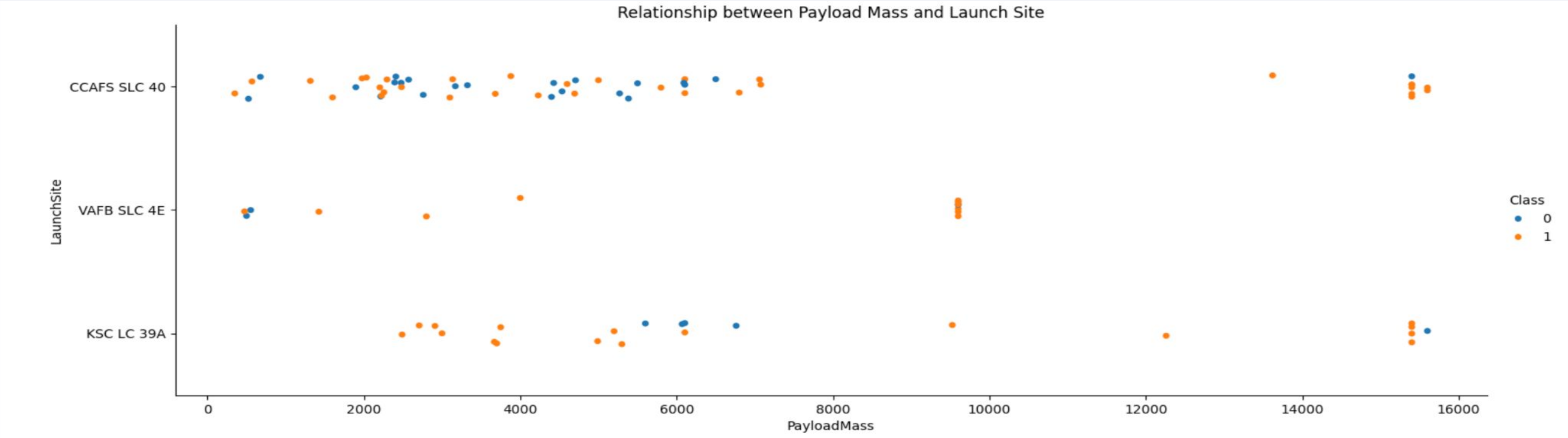
Flight Number vs. Launch Site

We can see in the below relationship between Flight Number and Launch site where as the flight number increases, more landings get successful for KSC LC 39A and CCAFS SLC 40, and for CAFB SLC 4E, there are no landings more than 65 flight number. There are much more flights for CCAFS SLC 40 compared to other flights.



Payload vs. Launch Site

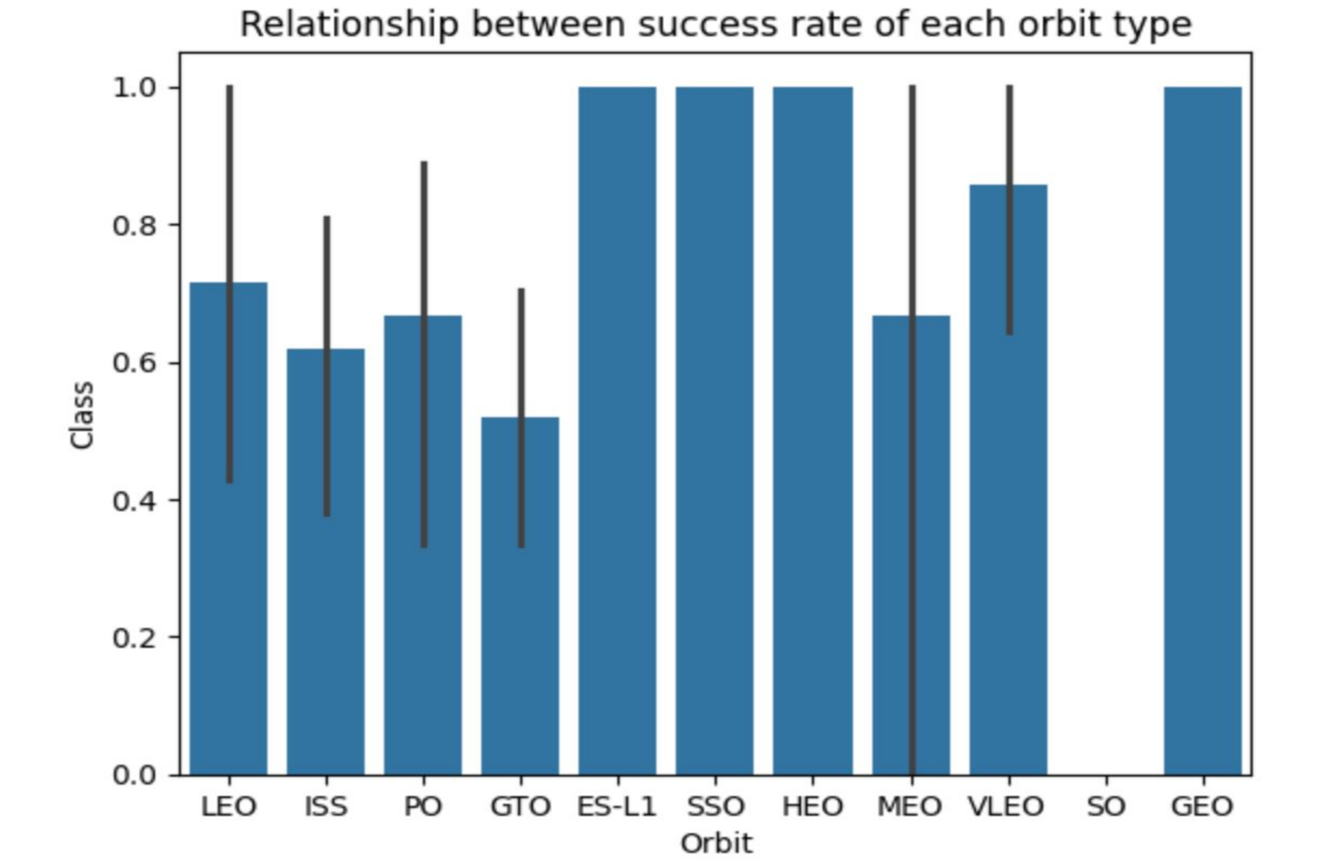
- We also want to observe if there is any relationship between launch sites and their payload mass.
- Now if you observe Payload Mass Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavy payload mass(greater than 10000).



Success Rate vs. Orbit Type

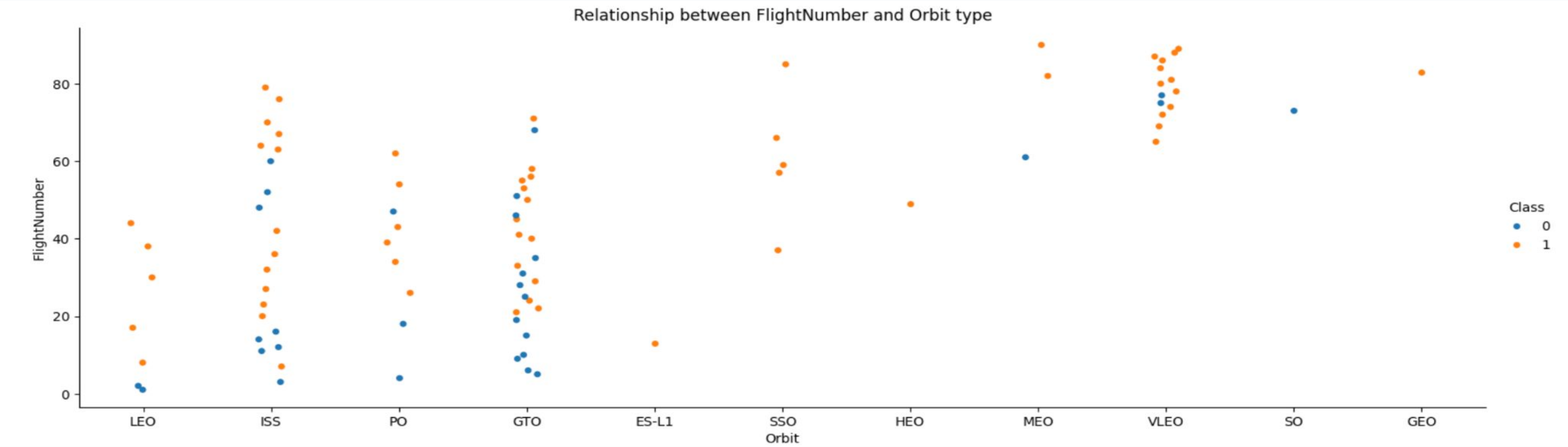
- This bar chart shows the relationship between Success rate and Orbit Type.

- We can see here that the success rate is high for some orbits like ES-L1, SSO, HEO, GEO.



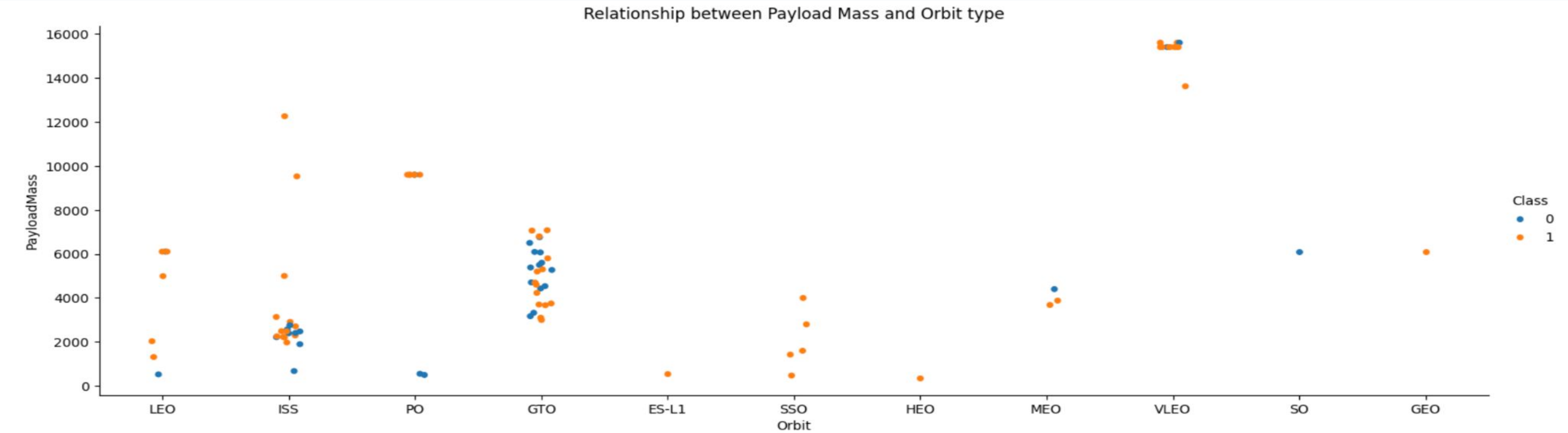
Flight Number vs. Orbit Type

- For each orbit, we want to see if there is any relationship between FlightNumber and Orbit type.
- We can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.



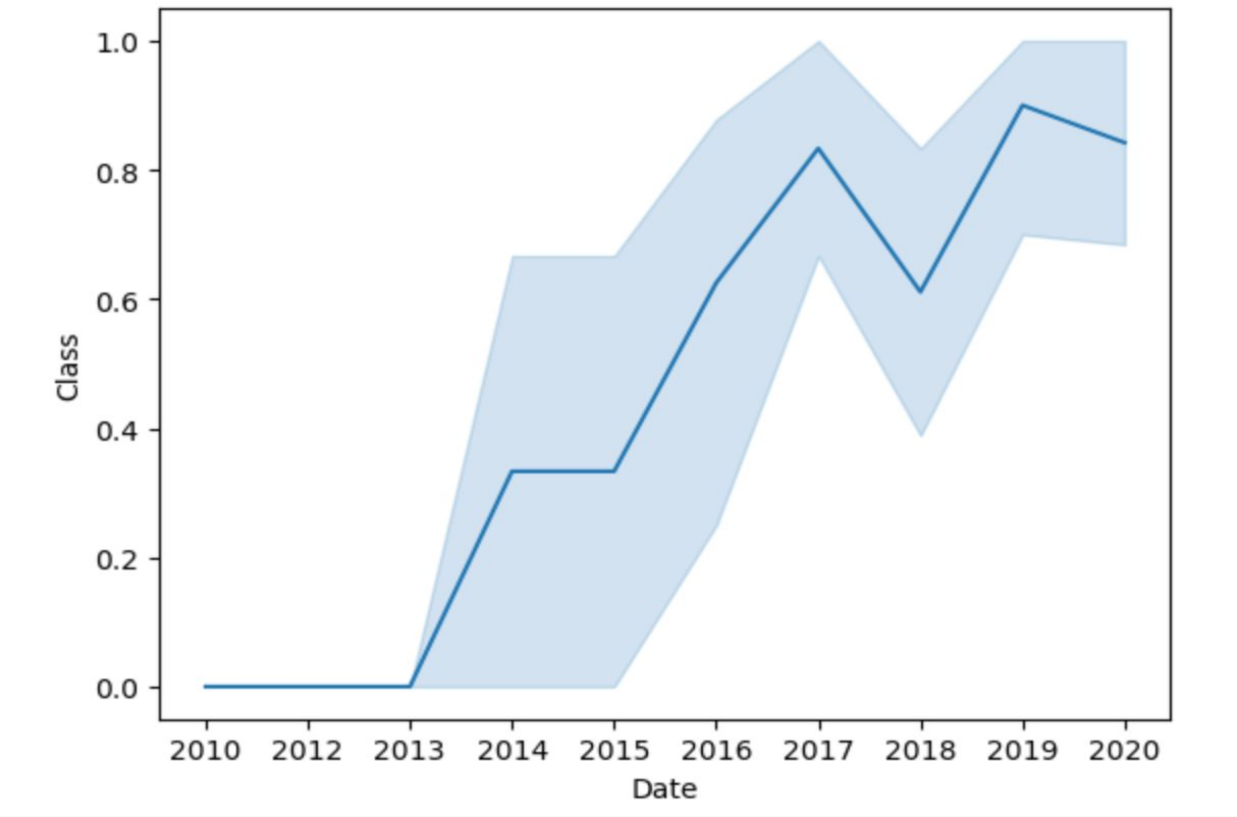
Payload vs. Orbit Type

- We have visualized the relationship between Payload Mass and Orbit type.
- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.



Launch Success Yearly Trend

- This shows a line chart with x axis to be Year and y axis to be average success rate, to get the average launch success trend.
- We can observe that the success rate since 2013 kept increasing till 2020.



All Launch Site Names

- Through SQL, we can find the names of the unique launch sites in the space mission.
- We will use this query:
- **%sql select distinct Launch_Site from spacextbl**
- The unique Launch sites are shown here:

Launch_Site

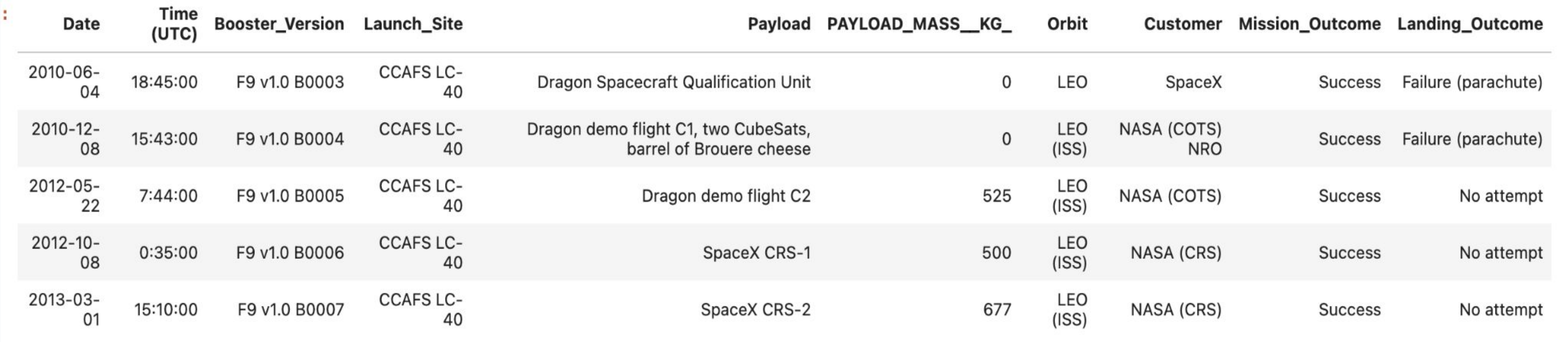
- **CCAFS LC-40**
- **VAFB SLC-4E**
- **KSC LC-39A**
- **CCAFS SLC-40**

Launch Site Names Begin with 'CCA'

- Now we will display 5 records where launch sites begin with `CCA`.
- We used this query:

  %sql select * from spacextbl where Launch_Site like 'CCA%' limit 5

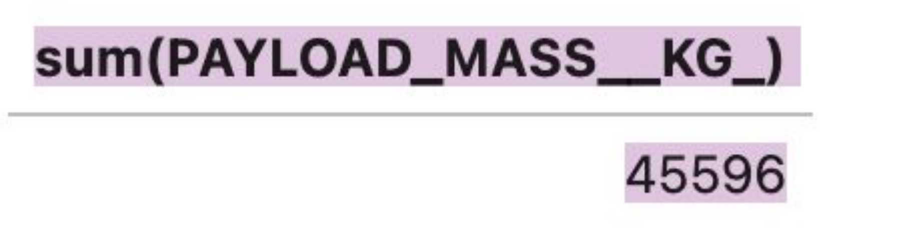
- We can see the result below which shows 5 records with all the launch sites which starts with ‘CCA’.



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

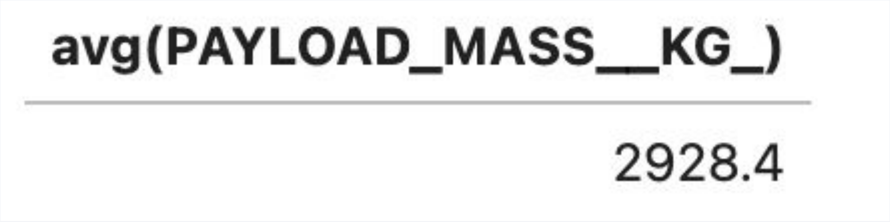
- Calculating the total payload carried by boosters from NASA.
- this query is used to calculate the total payload mass:
- **%sql select sum(PAYLOAD_MASS__KG_) from spacextbl where Customer='NASA (CRS)'**



sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

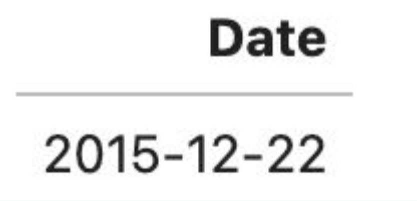
- Calculating the average payload mass carried by booster version F9 v1.1
- **%sql select avg(PAYLOAD_MASS__KG_) from spacextbl where Booster_Version='F9 v1.1'**



avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

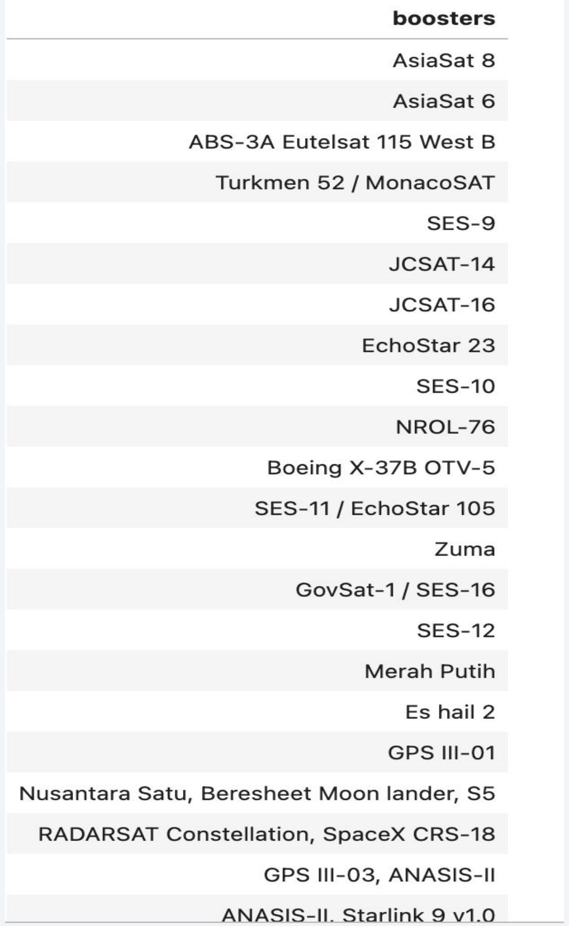
- Finding the dates of the first successful landing outcome on ground pad
- **%sql select date from spacextbl where Landing_Outcome='Success (ground pad)' limit 1**



Date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

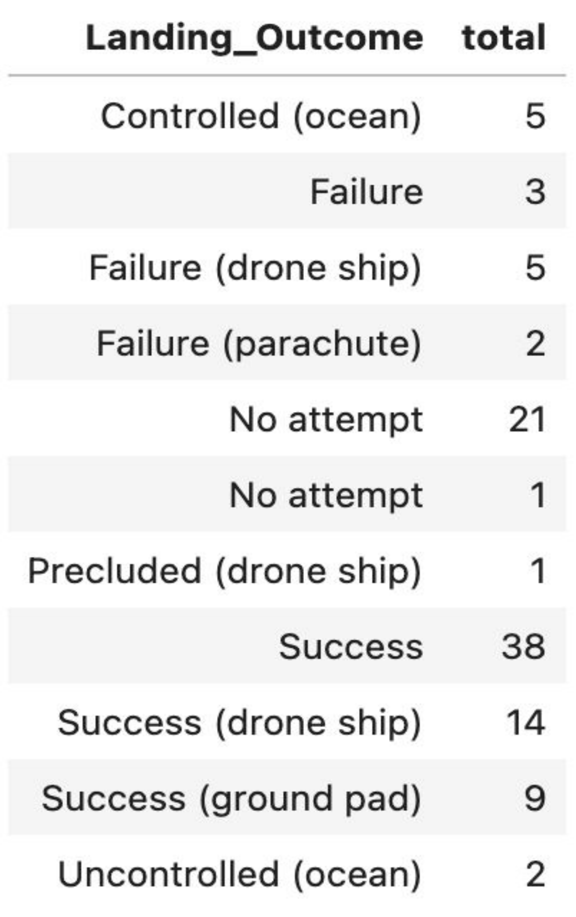
- Listing the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000.
- **%sql select Payload as boosters from spacextbl where PAYLOAD_MASS__KG_>4000 and PAYLOAD_MASS__KG_<6000**



boosters
AsiaSat 8
AsiaSat 6
ABS-3A Eutelsat 115 West B
Turkmen 52 / MonacoSAT
SES-9
JCSAT-14
JCSAT-16
EchoStar 23
SES-10
NROL-76
Boeing X-37B OTV-5
SES-11 / EchoStar 105
Zuma
GovSat-1 / SES-16
SES-12
Merah Putih
Es hail 2
GPS III-01
Nusantara Satu, Beresheet Moon lander, S5
RADARSAT Constellation, SpaceX CRS-18
GPS III-03, ANASIS-II
ANASIS-II, Starlink 9 v1.0

Total Number of Successful and Failure Mission Outcomes

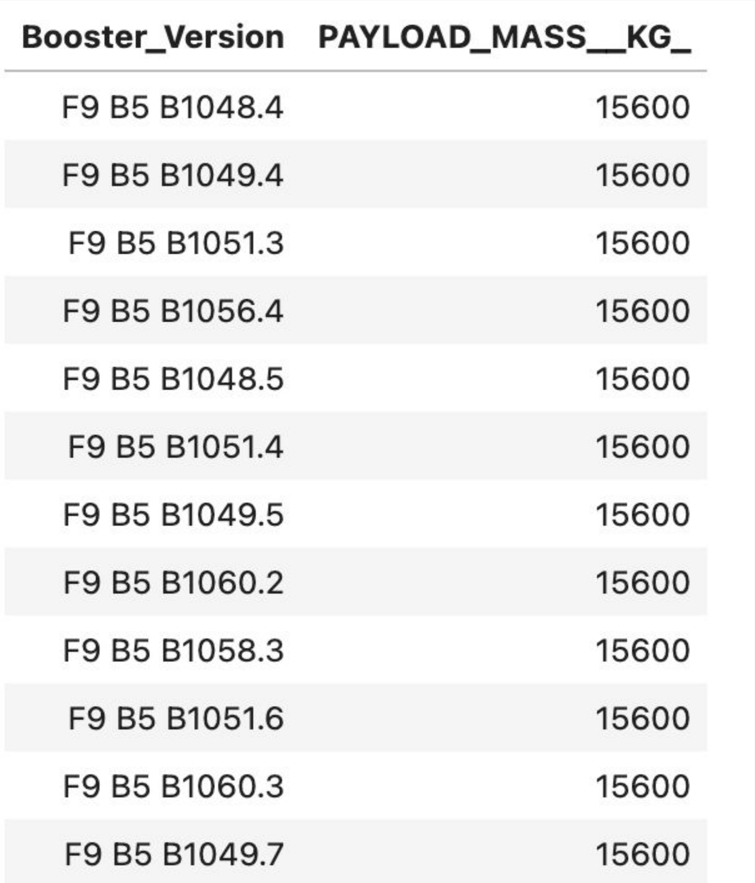
- Calculate the total number of successful and failure mission outcomes
- **%sql select Landing_Outcome, count(*) as total from spacextbl group by Landing_Outcome**



Landing_Outcome	total
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

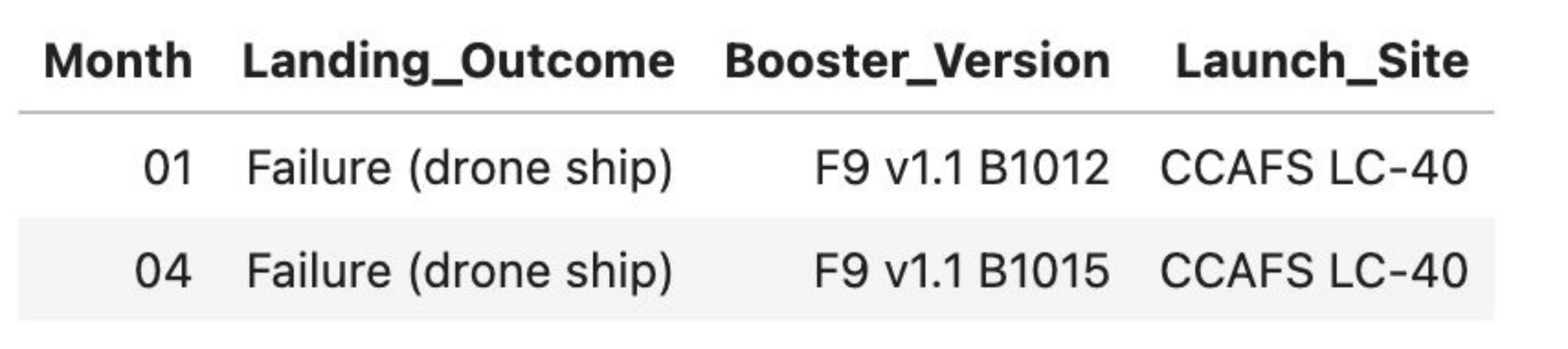
- Listing the names of the booster which have carried the maximum payload mass
- **%sql select Booster_Version, PAYLOAD_MASS__KG_ from spacextbl where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from spacextbl)**



Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

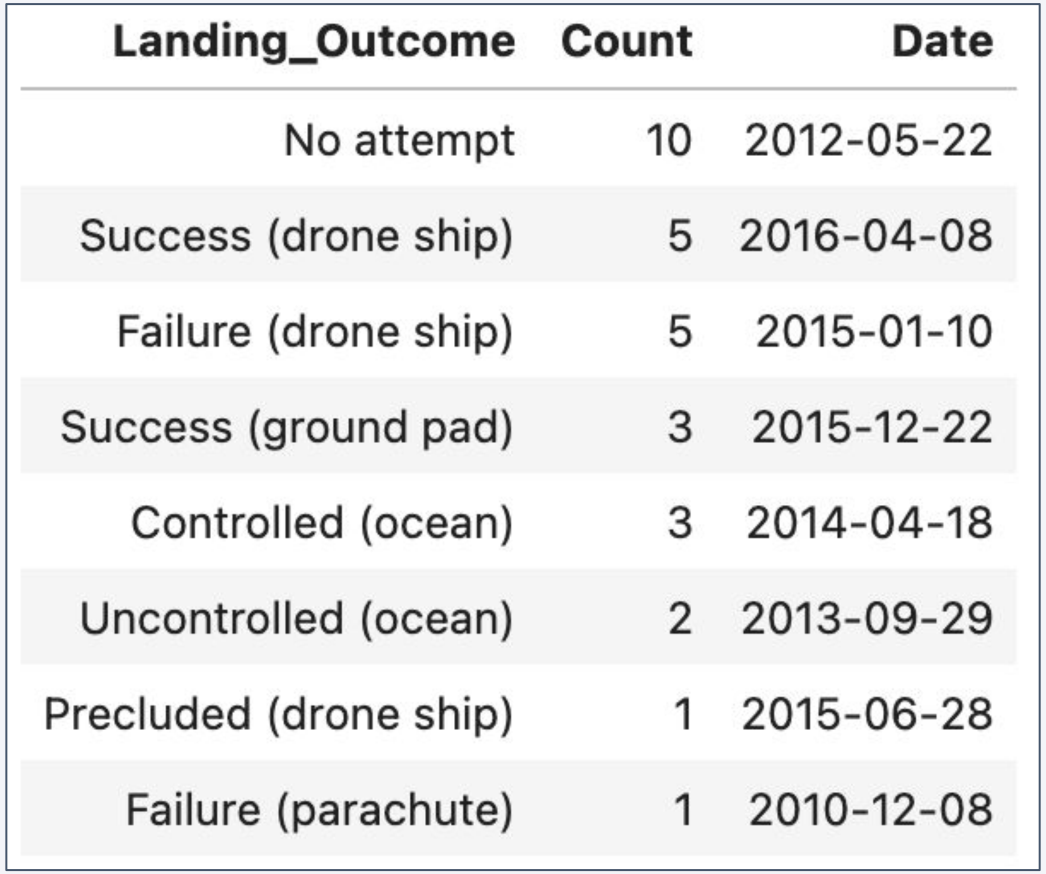
- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- **%sql select substr(Date, 6,2) as Month, Landing_Outcome, Booster_Version, Launch_Site from spacextbl where substr(Date,0,5)='2015' and Landing_Outcome like '%Failure%'**



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

- 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
- **%sql select Landing_Outcome, count(Landing_Outcome) as Count, Date from spacextbl where Date>'2010-06-04' and Date<'2017-03-20' group by Landing_Outcome order by Count desc**



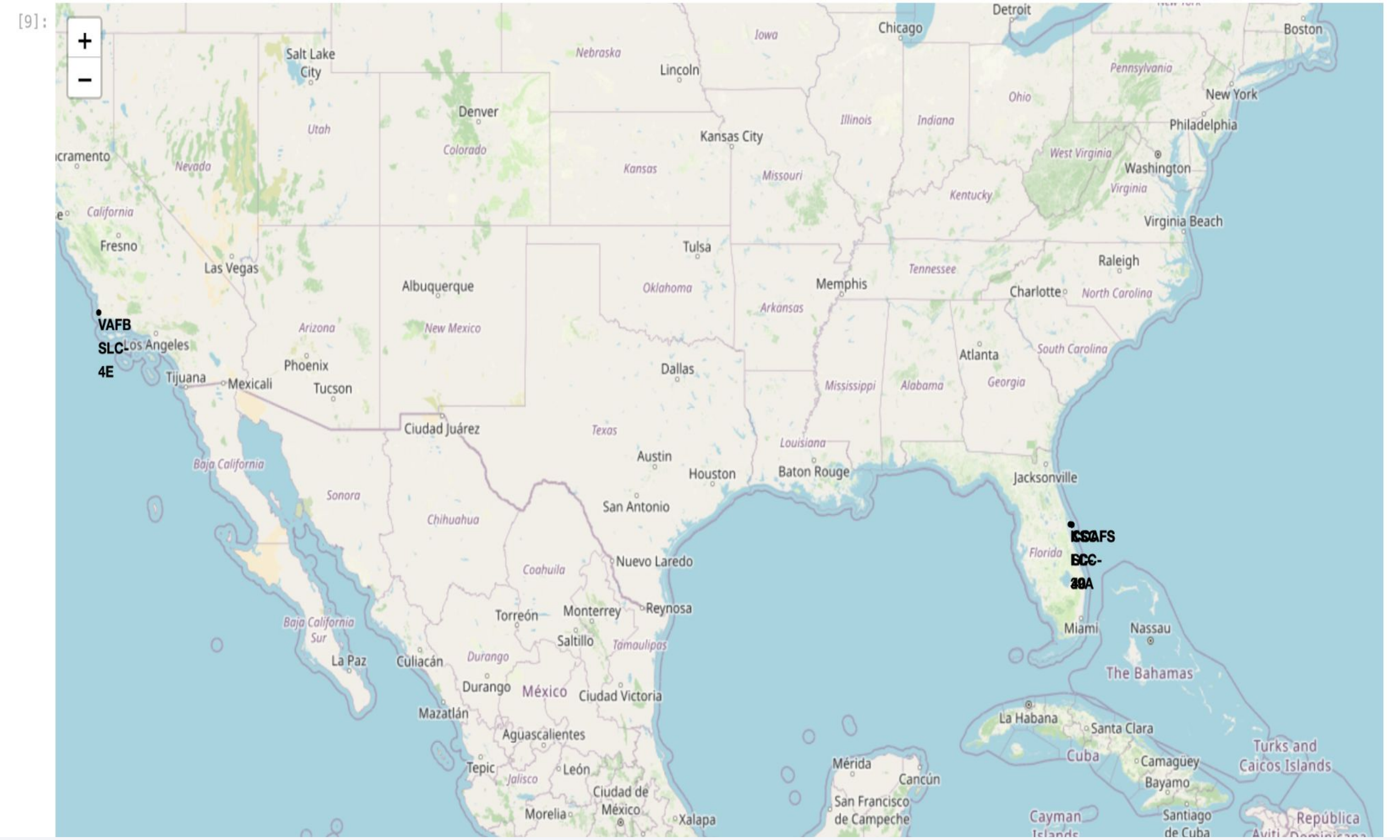
Landing_Outcome	Count	Date
No attempt	10	2012-05-22
Success (drone ship)	5	2016-04-08
Failure (drone ship)	5	2015-01-10
Success (ground pad)	3	2015-12-22
Controlled (ocean)	3	2014-04-18
Uncontrolled (ocean)	2	2013-09-29
Precluded (drone ship)	1	2015-06-28
Failure (parachute)	1	2010-12-08

Section 3

Launch Sites Proximities Analysis

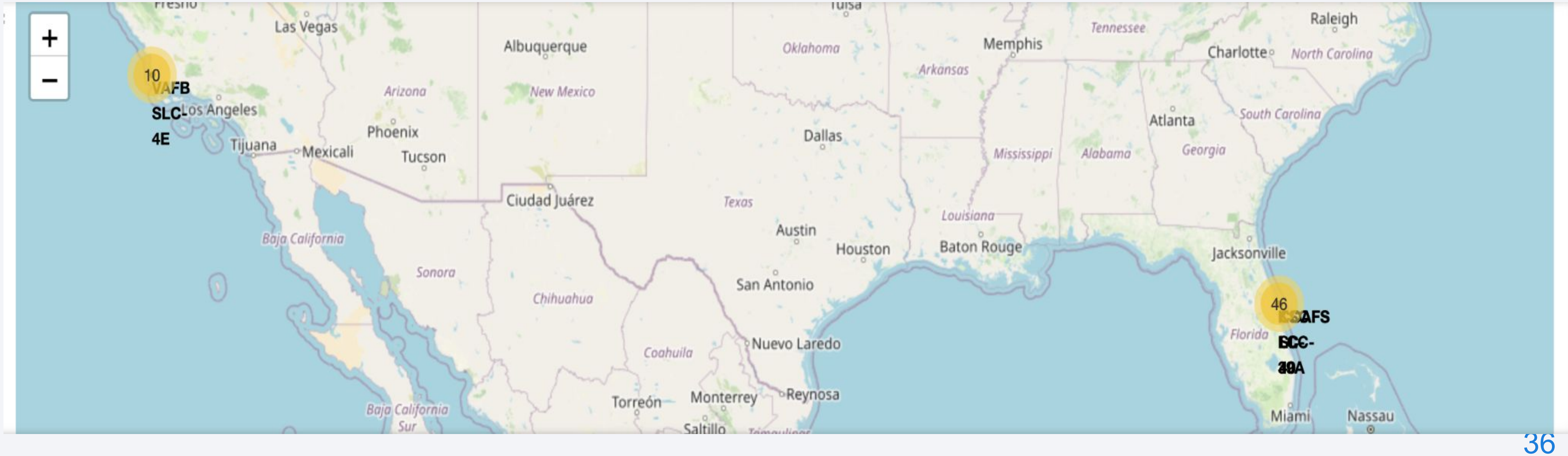
Visualisation of launch sites on a map

- Here, we are marking all launch sites on a map using folium.Map()
- We can conclude that all the launch sites are not close to the equator lines but to the coast which can be seen in the map itself.



Success/Failed launches for each site on the map using Folium

- We have tried to enhance the map by adding the launch outcomes for each site, and see which sites have high success rates.
- Note that a launch only happens in one of the four launch sites, which means many launch records will have the exact same coordinate. Marker clusters can be a good way to simplify a map containing many markers having the same coordinate.

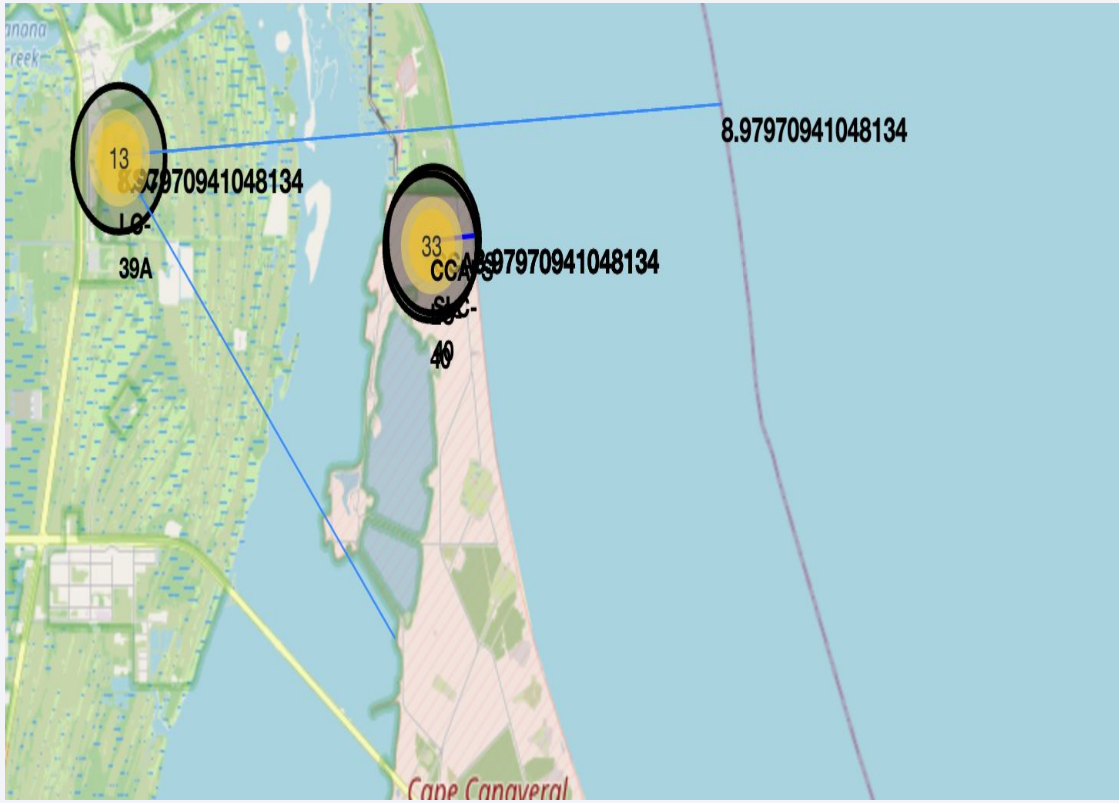


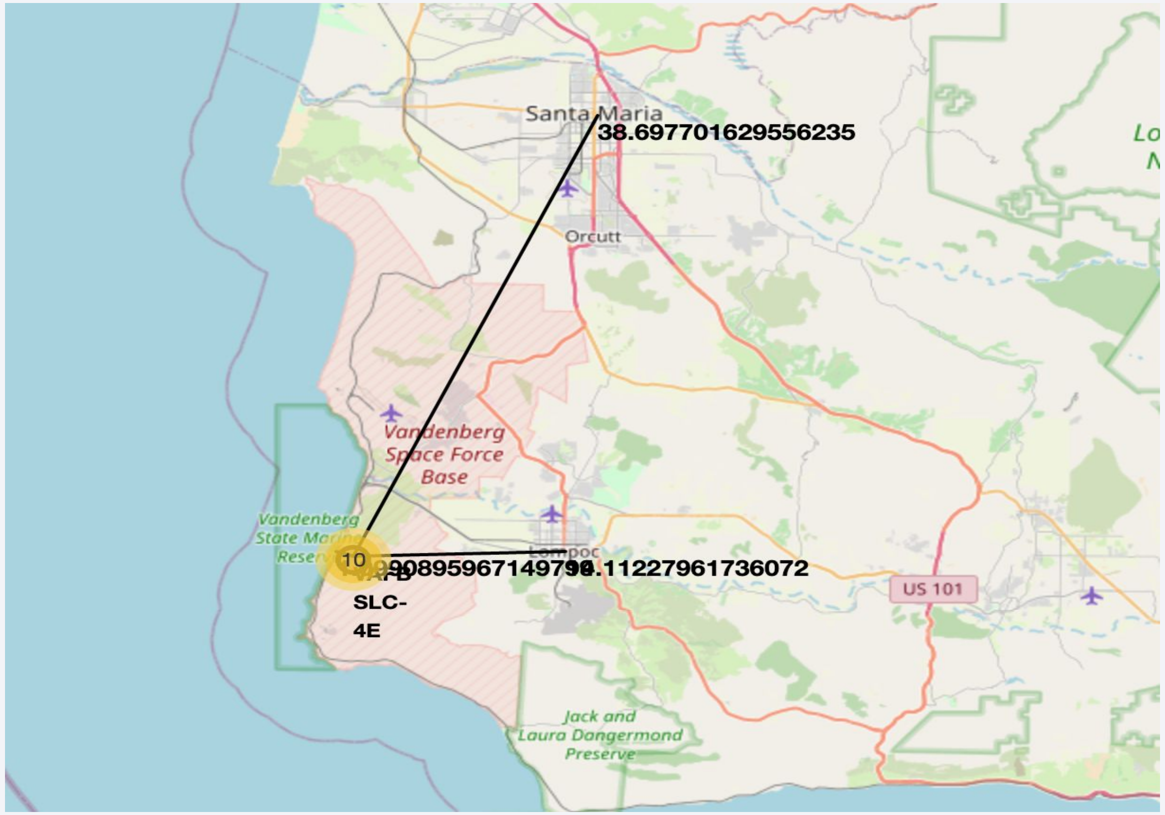
<Folium Map Screenshot 2>

+
−
7
CCAFS
SLC-
40
26
CCAFS
LC-
40
Falcon
9 launchpad

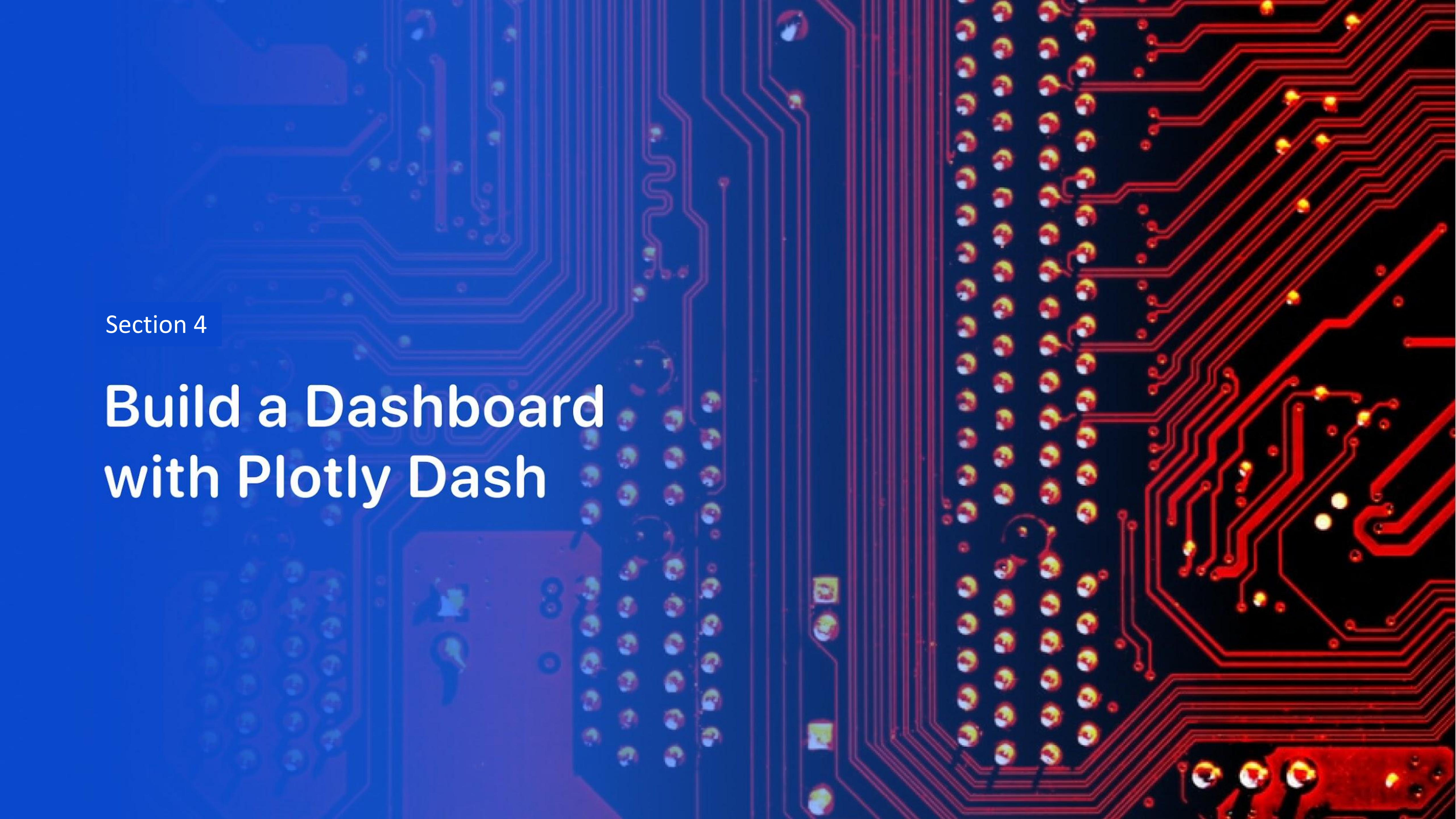
Close view of the launch sites.

<Folium Map Screenshot 3>





- Here, we have visualised the distance of the launch sites from railway, highway, cities and coastal areas. The first diagram shows the distance of launch sites from coast and cities and second shows the distance of launch sites from highway and railways.

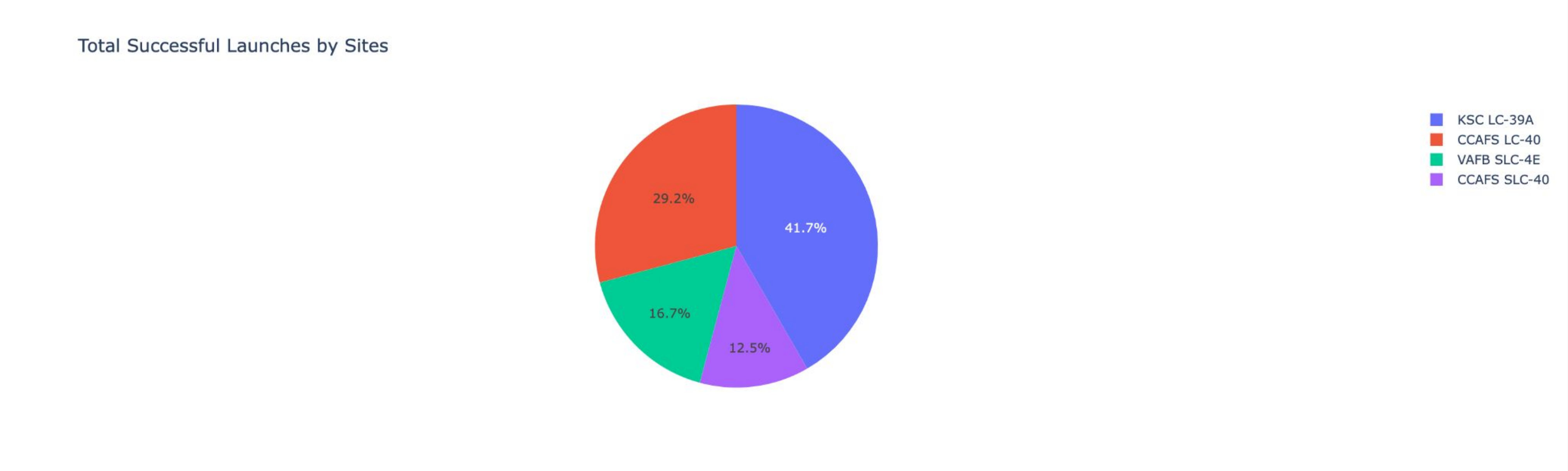


Section 4

Build a Dashboard with Plotly Dash

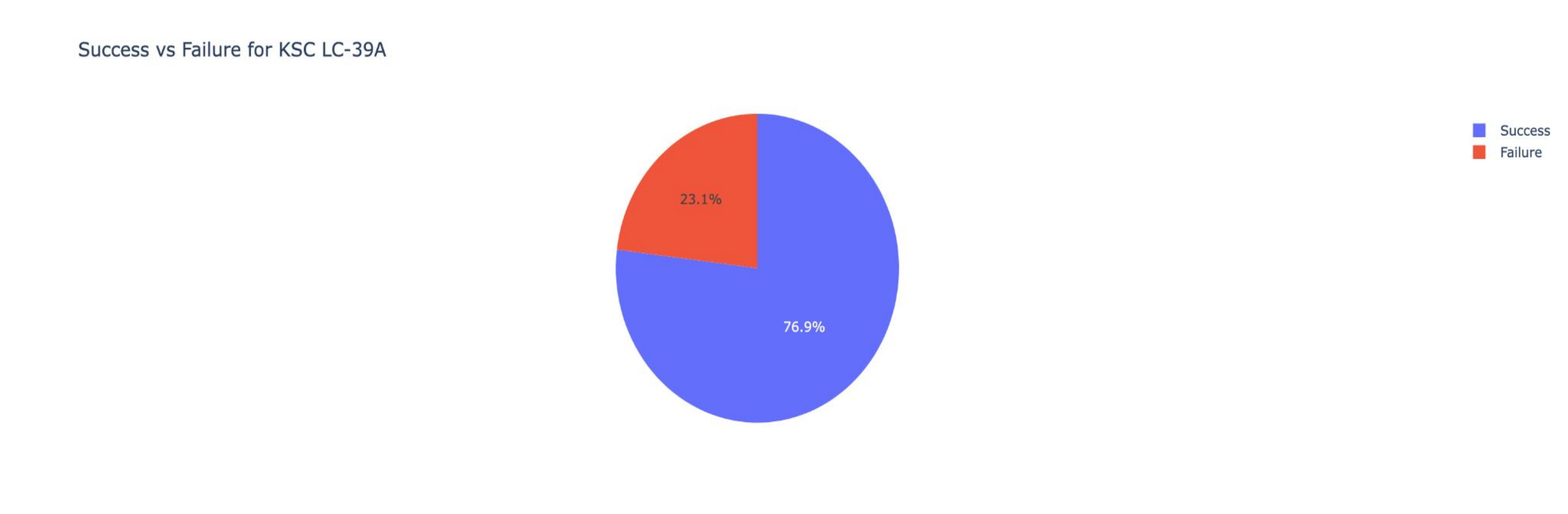
Total number of successful launches by all the sites

- This Pie chart shows the total number of successful launches by all the sites.
- By seeing this chart, we can determine the launch site with highest number of successful launches and that is KSC-LC-39A.



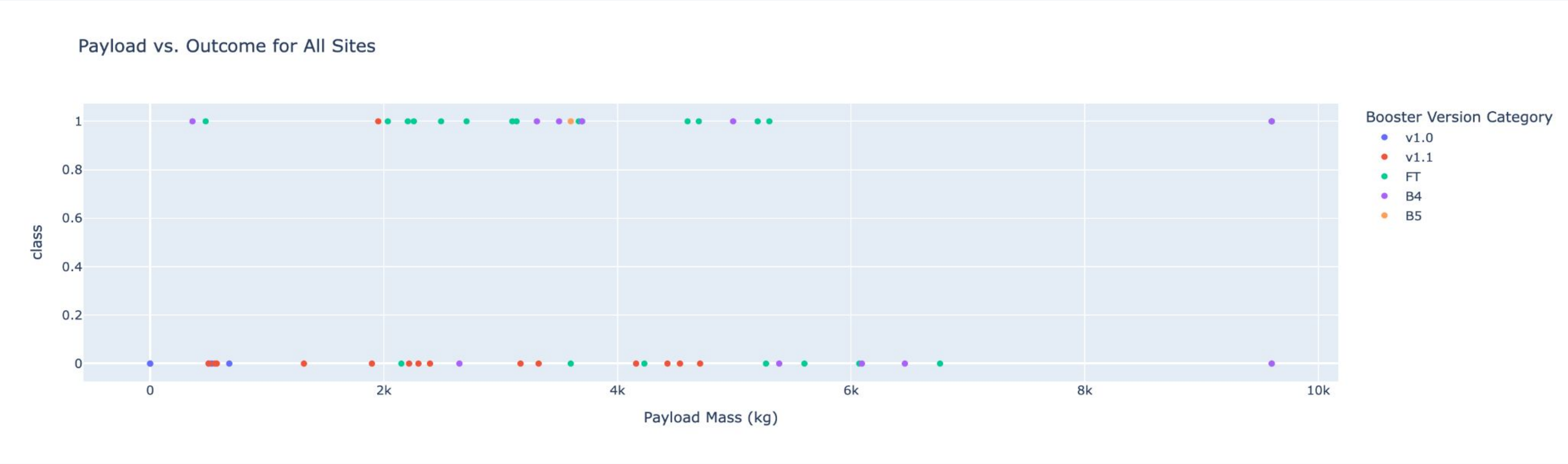
Success vs Failure of KSC LC-39A

This Pie Chart shows the Success vs Failure of the launch site KSC LC-39A with the highest number of successful launches. It shows the number of success launches by this launch site is 76.9%.



Plot between success rate and Payload wrt Booster Version

This is the scatter plot between success rate and Payload wrt Booster Version, Here we can see that the highest number of successful launch is done by Booster Version FT and highest number of unsuccessful launches is done by Booster Version v1.1.

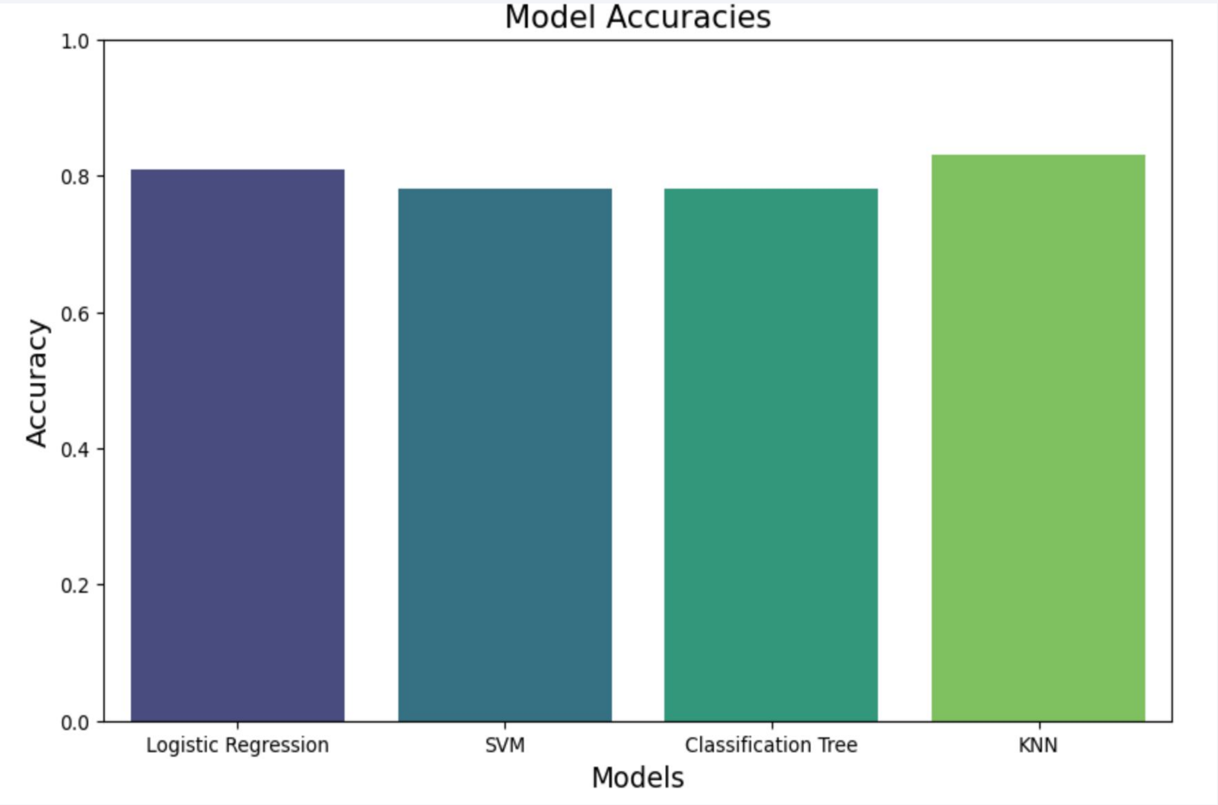


Section 5

Predictive Analysis (Classification)

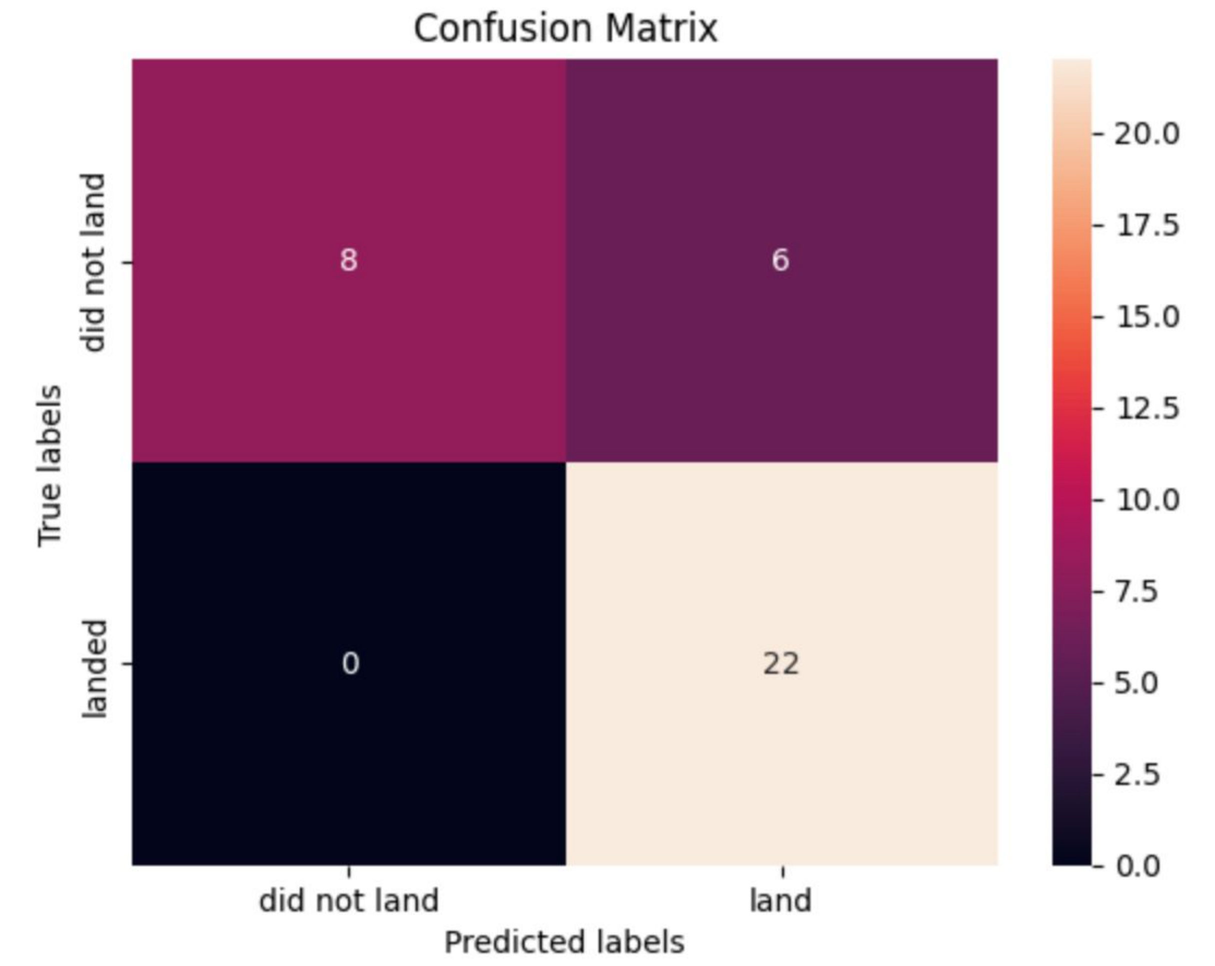
Classification Accuracy

- We have visualized the built model accuracy for all built classification models, in a bar chart. The models are Logistic regression, SVM, Classification tree and KNN
- Find the model KNN, it has the highest classification accuracy.



Confusion Matrix

- Confusion matrix of the best performing model i.e., KNN is shown here.
- Here we can analyze the confusion matrix.
- The model predicted 8 as did not land and in original it is correct, and 22 as the correct landed number of launches, 6 launches were misinterpreted as landed but it was not landed according to True Data.



Conclusions

- Hereby, I conclude by saying that the the best model has been figured out to predict the successful launch of a rocket based on it’s launch site, payload, location and orbit type.
- The best model can always change because the machine learning algorithm are performing continuously more accurate with time, by the use of various measures like scaling the data, handling the data, categorizing the data into dummies and much more, we can find the best model for our prediction.
- Based on the data, the models are chosen and their respective measures are taken to make it more reliable and accurate with time.
- This project was such a great learning experience for me and I got to learn so many things and i did them by myself. I hope to do projects like this in the future.

Thank you!