IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Santiago Quintanilla
26th March 2024

Outline

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

Executive Summary

- In this analysis we have used the data provided by SpaceX in order to predict the probability of landing for the Falcon 9, therefore getting more accurate data on the true cost of launch due to its potential reusability.
- Through different methods we have determined the most important factors in order to predict the landing success for the Falcon 9. After which we have used those factors to elaborate a model for the most accurate landing prediction.

Introduction

- SpaceX advertises the Falcon 9 rocket launches with a cost of 62 million dollars. This is in contrast to other providers which can cost upward of 165 million dollars each. This difference in price is in the most part due to the savings SpaceX can make reusing the first stage of the launch.
- Because of the potential reusability of the first stage of the launch, the cost of the Falcon 9 can remain much lower than its competitors. However, the price advertised is dependent on the outcome of the first stage, therefore needing a successful landing of this stage. With that in mind, we want to determine if it will land in order to determine the cost of the launch.

Section 1

Methodology

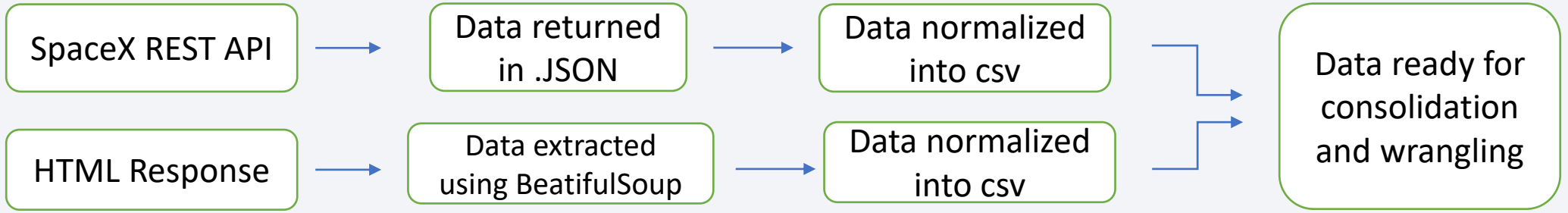
Methodology

Executive Summary

- Data collection methodology:
  - We first requested and cleaned the SpaceX API. After that we extracted the Falcon 9 launch records and converted them into a pandas dataframe.
- Perform data wrangling
  - We performed some exploratory data analysis with the data collected and used it to determine the training labels which would later be used.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - After the standardization for the data, we proceed to split into training data and test data, using the test data to find out which model performs best.

Data Collection

- For the purposes of this analysis, we needed to first collect the data which we would later use in order to get the answers we are searching for. For the data collection stage of our project, we have used two different methods. The first of these is a request to an API, specifically the SpaceX API. The second method being web scraping, searching the internet for some Falcon 9 launch records and extracting them in order to use them in our analysis.
- Before having the data ready for consolidation and wrangling, we used both methods previously specified. The following flowchart summarizes the steps taken in this stage:



Data Collection – SpaceX API

- The flowchart shows the calls used on the SpaceX REST
- The GitHub URL of the completed SpaceX API notebook is: https://github.com/saquza/testrepo/blob/master/Data%20Collection%20API.ipynb

1. Getting response from API

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
```

2. Converting response to .json file

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

3. Use functions to apply outputs to the variables

```
getBoosterVersion(data)  getPayloadData(data)
getLaunchSite(data)      getCoreData(data)
```

4. Assign the list into a dictionary, then a 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 and export to csv

```
data_falcon9= df[df['BoosterVersion']!= 'Falcon 1']
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping

- The flowchart shows the web scraping process

- The GitHub URL of the completed web scraping notebook is: https://github.com/saquza/testrepo/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

1. Getting HTML response and creating BeatifulSoup object

```
page= requests.get(static_url)
soup= BeautifulSoup(page.content, 'html.parser')
```

2. Finding tables

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

3. Getting column names

```
column names = []
th_col_names_text= first_launch_table.find_all('th')
for name in th_col_names_text:
    col_names_text= extract_column_from_header(name)
    if col_names_text is not None and len(col_names_text) > 0:
        column_names.append(col_names_text)
```

4. Creation of dictionary

```
launch_dict= dict.fromkeys(column_names)
del launch_dict['Date and time ( )']
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
launch_dict['Version booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

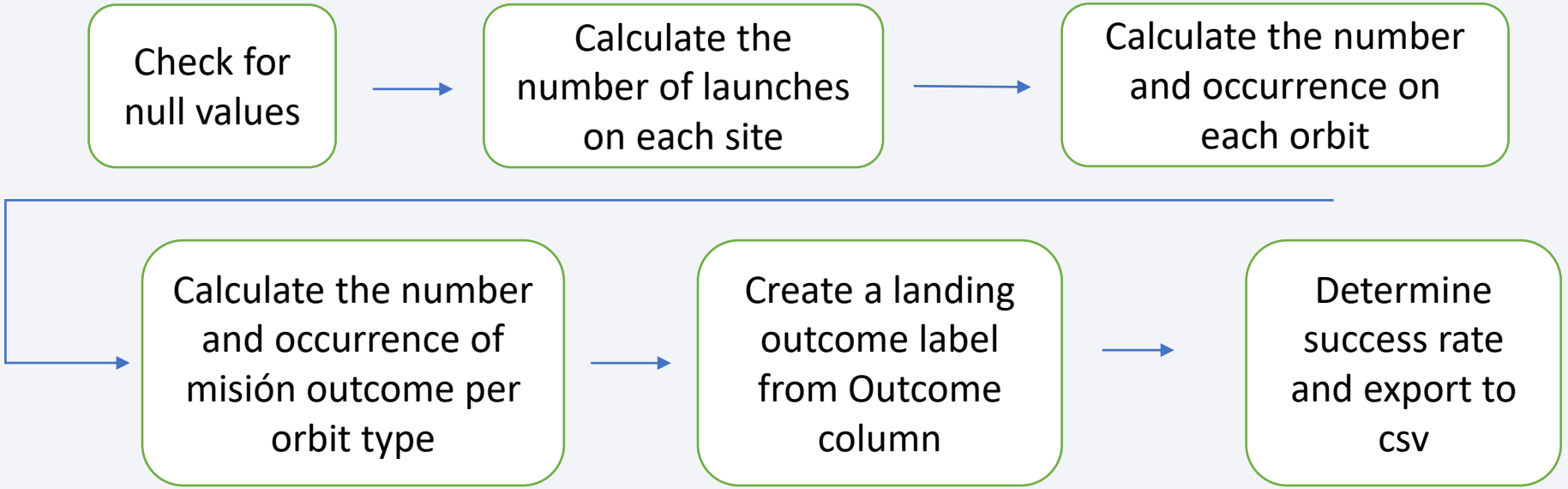
5. Fill the dictionary with launch records

```
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"):
```

6. Convert dictionary to dataframe and export to csv

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

Data Wrangling

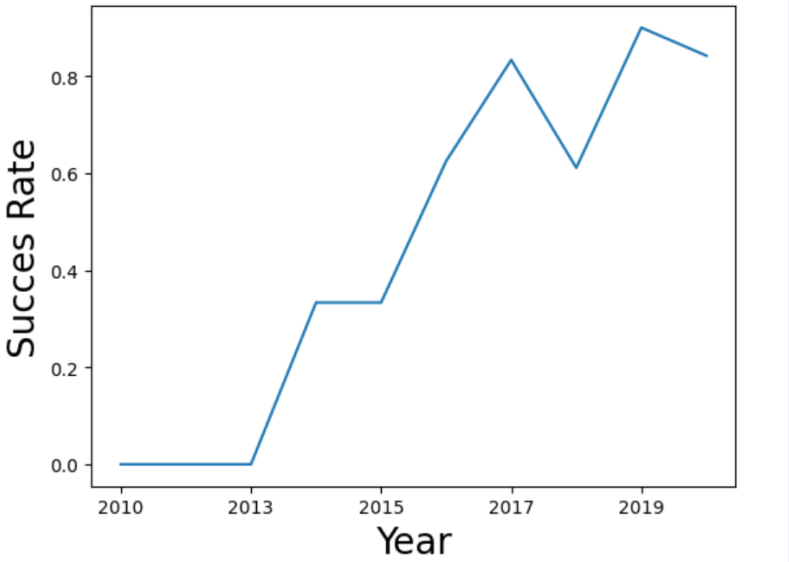


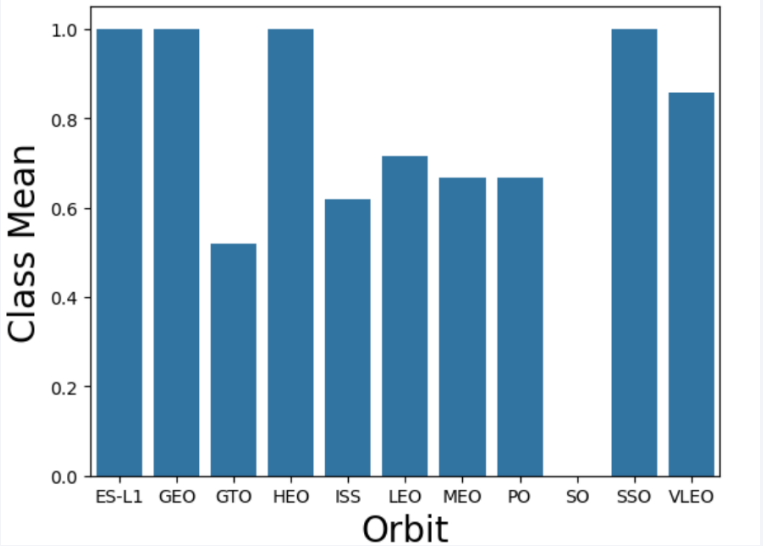
- The GitHub URL of the completed data wrangling notebook is: https://github.com/saquza/testrepo/blob/master/Data%20Wrangling.ipynb

EDA with Data Visualization

- In this exploratory data analysis we plotted some charts to help us see the relationship between flight number and payload mass; flight number and launch site; launch site and payload mass; orbit type and success rate; flight number and orbit type; payload mass and orbit type; and year and success rate in order to see if there are any trends.

- The GitHub URL of the completed EDA with data visualization notebook is: https://github.com/saquza/testrepo/blob/master/EDA%20with%20Data%20Visualization.ipynb





EDA with SQL

- **We used SQL to help explore our data and answer the following questions:**
- Display the names of the unique launch sites
- Display 5 records where launch sites begin with ‘KSC’
- Display the total payload mass carried by boosters launched by NASA
- Display average payload mass carried by booster version F9 v1.1
- List the date when the successful landing outcome in drone ship was achieved
- List the names of the names of the successful boosters in ground pad with payload mass between 4000 and 6000
- List the total amount of successful and failure mission outcomes
- List the booster versions which have carried the maximum payload mass
- List the monthly records for 2017
- Counting the successful landings between june 2010 and march 2017
- The GitHub URL of the completed EDA with SQL notebook is: https://github.com/saquza/testrepo/blob/master/EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- We first marked all the launch sites on the map, after which we proceeded to mark the successful and failed launches for each site marked earlier. Lastly we calculated the distances between a launch site to its proximities.
- The markers, and their respective success ratio and distance to some of the landmarks was used in order to help us find the optimal location for a launch site.
- The GitHub URL of the completed interactice map with Folium notebook is: https://github.com/saquza/testrepo/blob/master/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

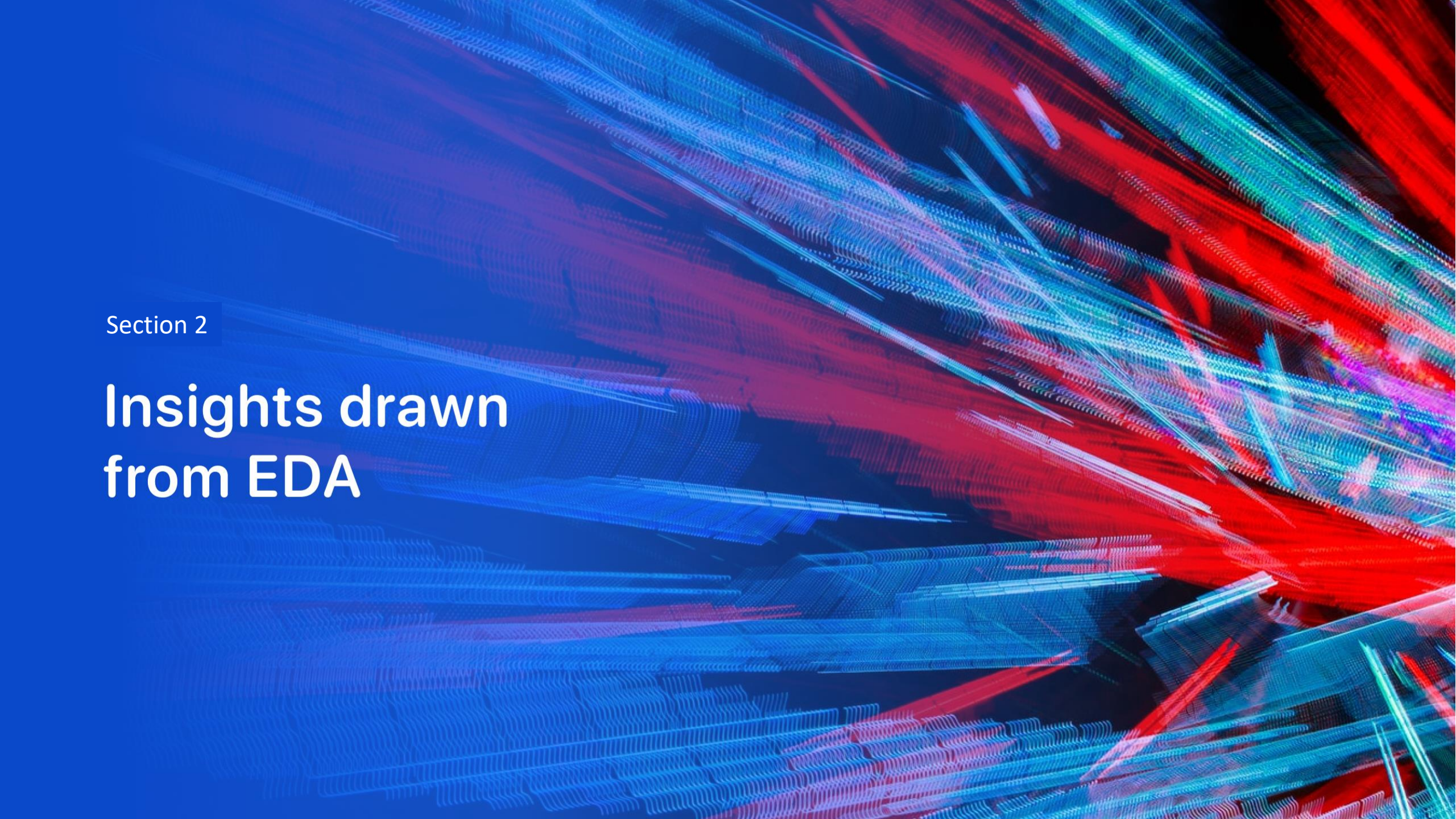
- We plotted a pie chart containing the success rate by launch site with an interactive dropdown that lets us select which launch site we want to check the rate for.
- We also plotted a scatter graph showing the outcomes (success/fail) for different payload masses by booster version.
- These charts and graphs help find the best combination of launch site and payload mass in order to help us predict the outcome of the mission.
- The GitHub URL of the completed Plotly Dash lab is: https://github.com/saquza/testrepo/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

- We loaded the data using numpy and pandas, transformed it and separated into training and testing data.
- We built different machine learning models and tuned different hyperparameters using GridSearchCV.
- We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.
- We found the best performing classification model.
- The GitHub URL for the completed predictive analysis lab is: https://github.com/saquza/testrepo/blob/master/Machine%20Learning%20Prediction%20Lab.ipynb

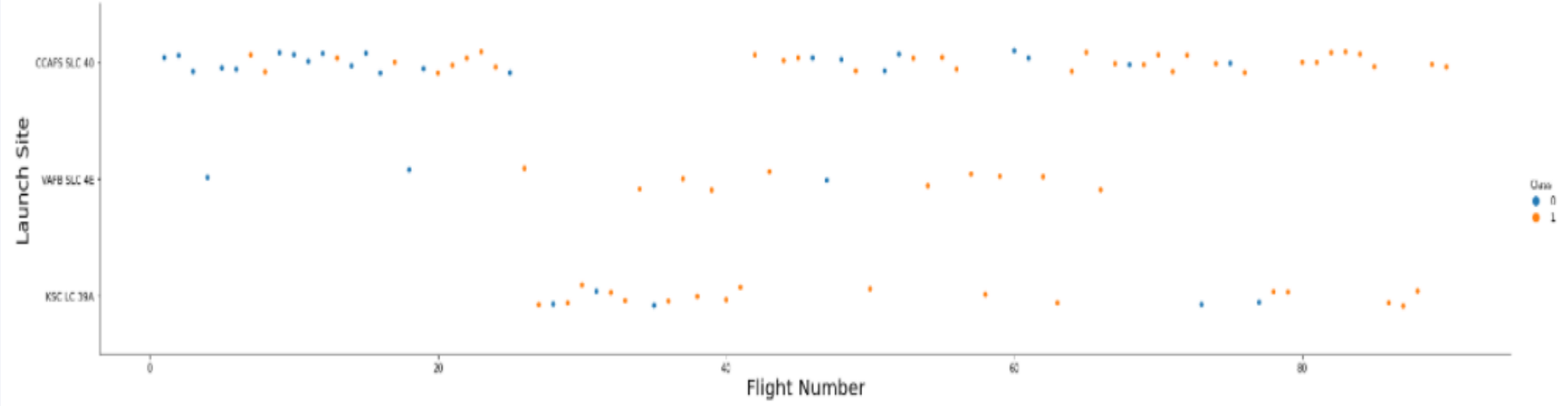
Results

- The models which best predict the outcome with our data are SVM, KNN and Logistic Regression models.
- Lighter payloads perform better than heavier ones.
- The launch site with the highest success rate is KSC LC 39A.
- The orbits ES-L1, GEO, HEO, SSO have the highest success rate.


Section 2
Insights drawn from EDA

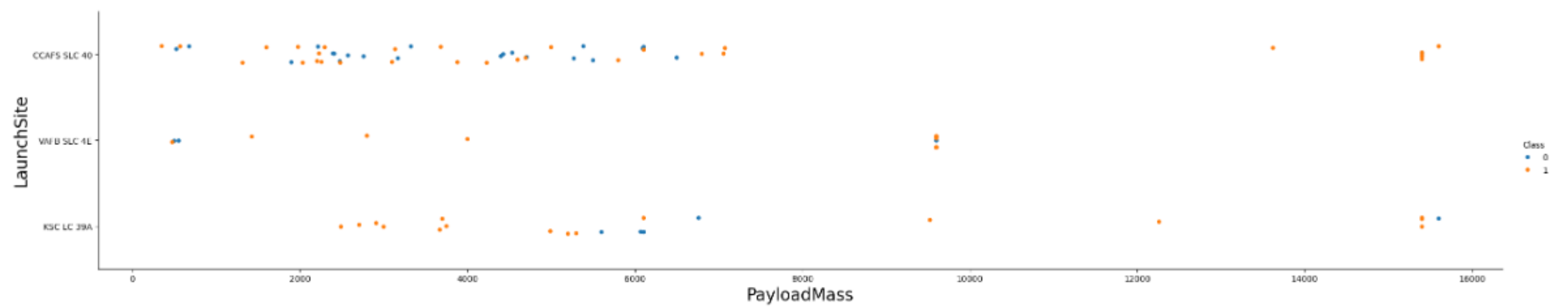
Flight Number vs. Launch Site

- As seen below, there are far more launches from the CCAFS SLC 40 launch site than any of the others. In the graph below we can also find that the success rate for the VAFB SLC 4E launch site seems to be the highest.



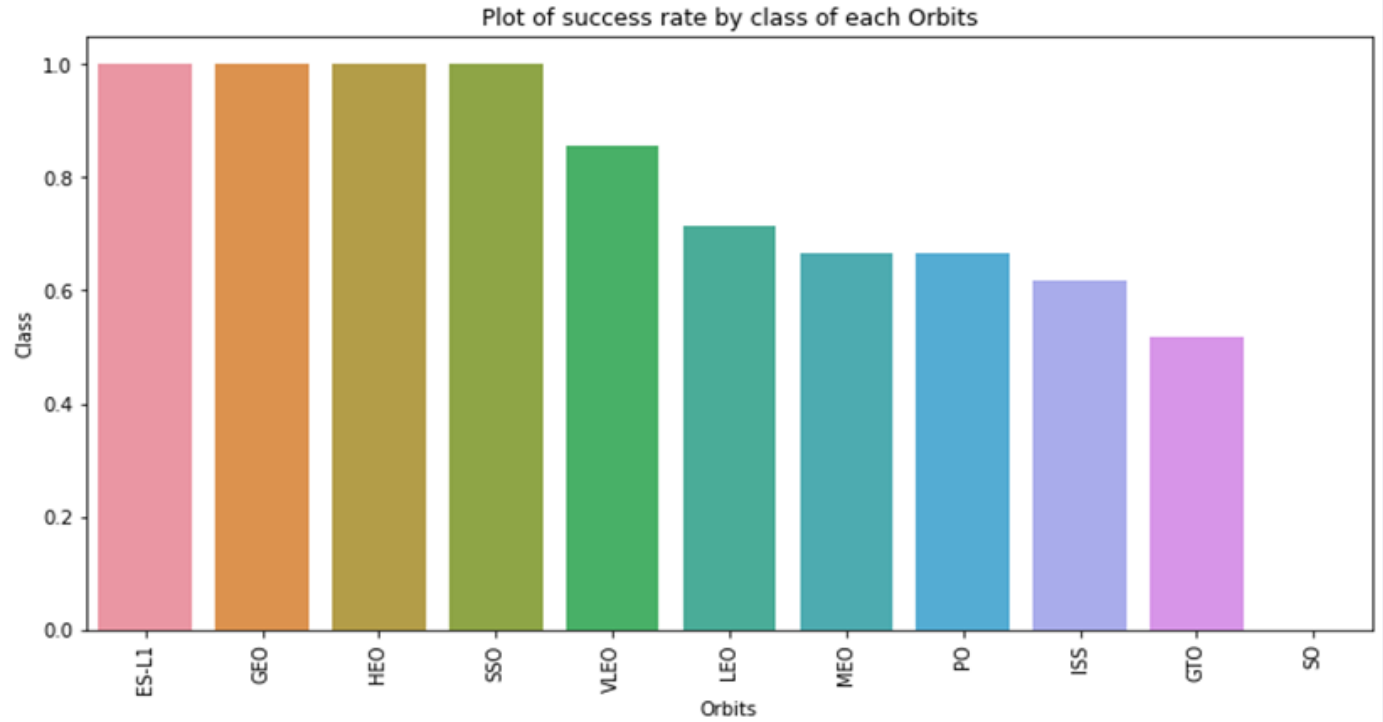
Payload vs. Launch Site

- The site with the highest amount of the lighter payload launches is CCAFS SLC 40. There also seems to be a correlation in which, the heavier the payload, the highest the success rate.



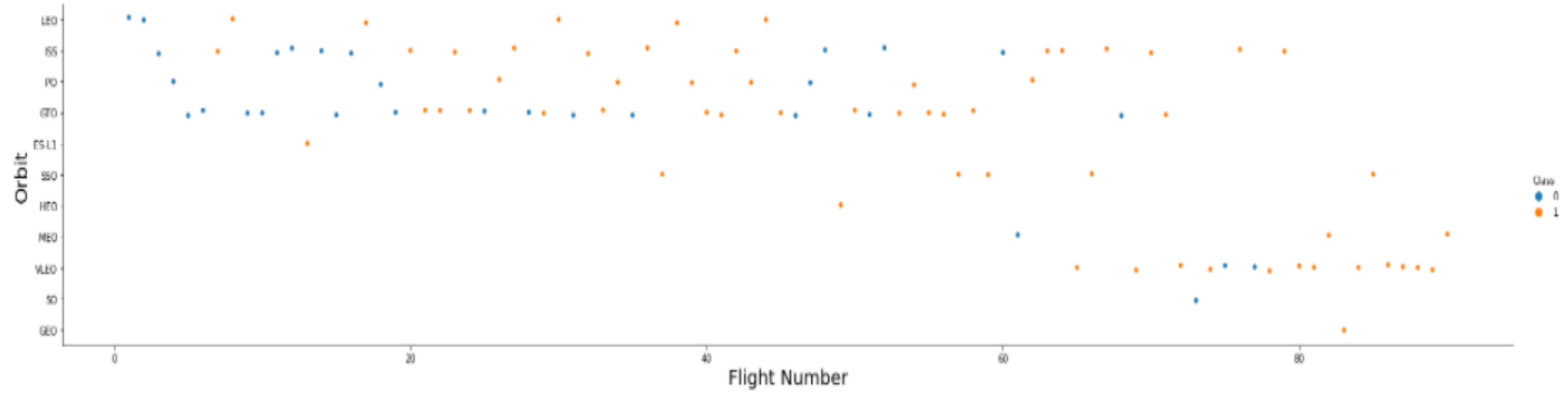
Success Rate vs. Orbit Type

- As seen on the chart, the orbits with the highest success rate are ES-L1, GEO, HEO and SSO, followed by VLEO with a slightly lower rate.



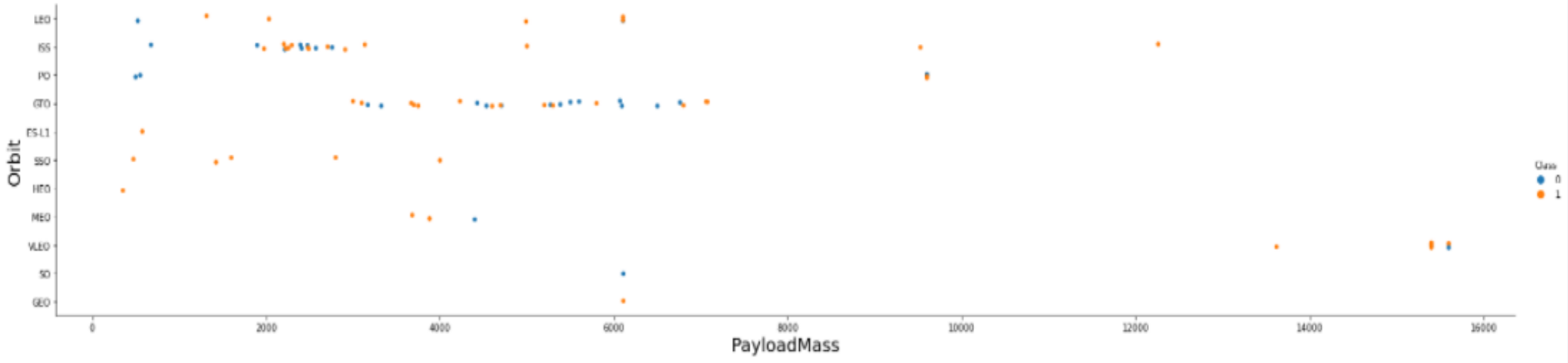
Flight Number vs. Orbit Type

- There seems to be a shift in the preferred orbit type in the latest launches, the trend seems to be to move on to newer orbits such as VLEO or SSO among others.



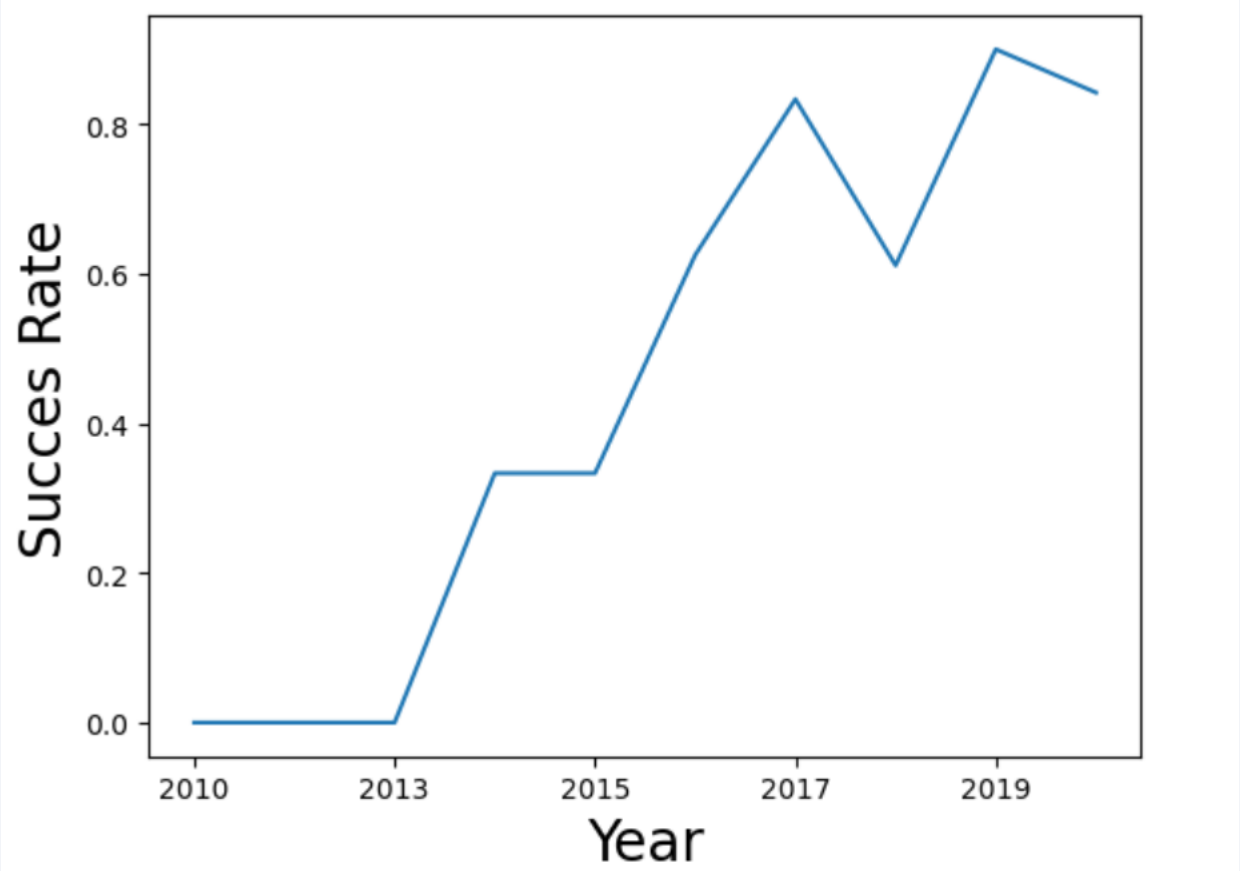
Payload vs. Orbit Type

- We can see that for heavier payloads, the ISS, PO and VLEO orbits are preferred. There seems to be no preference for payloads under 6000.



Launch Success Yearly Trend

- As we can see on the graph, since the year 2013 the success rate kept increasing, having only a slight dip on the year 2018. However the trend is positive and so is the correlation for these two variables.



All Launch Site Names

- The first step in our EDA was to familiarize ourselves with the data. In order to do that we wanted to first get to know the different launch sites, with that in mind we performed a DISTINCT query on our data. The result is as follows:

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

Launch Site Names Begin with 'KSC'

- The next step in our EDA with SQL was to get an idea of the amount of data we were working with. In order to have a first impression of this, we selected one of the launch sites seen previously and tried to get at least 5 different results for it. Something to keep in mind is that the full result couldn't be shown due to space constraint. Nevertheless, the result is as follows:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	lar
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	
2017-03-16	06:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	

Total Payload Mass

- Next we wanted to get and overall idea of the payload mass carried by the boosters launched by NASA. In order to do so, we used a query in which we filtered the results by their CRS tag. The result is as follows:

total_payload_mass_carried_by_nasa
111268

Average Payload Mass by F9 v1.1

- Regarding the payload mass, another aspect of our data we wanted to explore was the average payload mass carried by the booster version F9 v1.1. in order to do so, we filtered the result by the booster version and proceeded to ask for the average in the query. The result is as follows:

avg_payload_mass
2928

First Successful Ground Landing Date

- For the next step on our EDA we wanted to know the date for the first successful ground pad landing. In order to do so, we filtered the result on our query by the landing outcome. The result is as follows:

firstsuccessfull_landing_date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- For the next step in our EDA we wanted to find out the names of the boosters which have successfully landed o a drone ship and carried a payload with a weight between 4000 and 6000. In order to get the result we wanted, our query needed to have two different filters, one by payload and the other by the landing outcome. The result is as follows:

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

Total Number of Successful and Failure Mission Outcomes

- Next, we wanted to find out the total number of successful and failed mission outcomes. In order to do so, we filtered the result to our query by the outcome. We also wanted to show both the successes and failures, so we created a different result for each. The result is as follows:

successful_outcomes	failure_outcomes
61	10

Boosters Carried Maximum Payload

- Next, we wanted to find out the names of the boosters which have carried the maximum payload mass. In order to do so, we first needed to put on a DISTINCT query, filtering the results later by the boosters which have carried out the max amount of payload mass. The result is as follows:

booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2017 Launch Records

- For the next step on our EDA, we wanted to display the month names, successful landing outcomes in ground pad, booster versions and launch sites for each month in the year 2017. for this query we used only a filter by the outcome, but this time asking for 4 different columns in our query. The result is as follows:

MONTH	landing__outcome	booster_version	launch_site
December	Success (ground pad)	F9 FT B1019	CCAFS LC-40
July	Success (ground pad)	F9 FT B1025.1	CCAFS LC-40
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
January	Success (ground pad)	F9 B4 B1043.1	CCAFS SLC-40

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

- For the final step of our EDA with SQL, we wanted to rank the landing outcomes between june 2010 and march 2017. In order to do so, we needed to add two filters to our query, filtering it by date and outcome. Afterwards, we needed to use a group_by clause in order to have them in a descending order. The result is as follows:

landing_outcome	2
Success (drone ship)	5
Success (ground pad)	3

Section 3

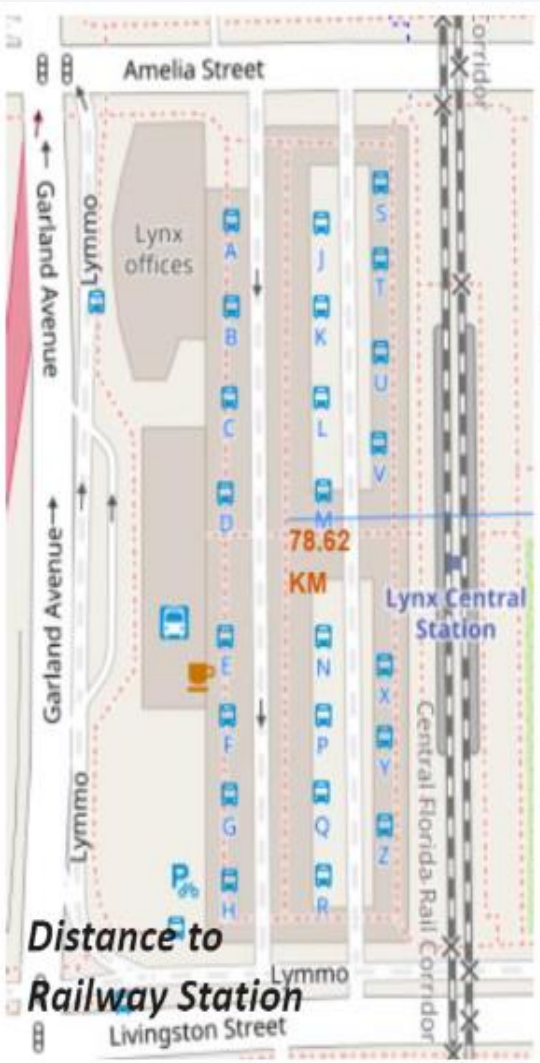
Launch Sites Proximities Analysis

All launch sites

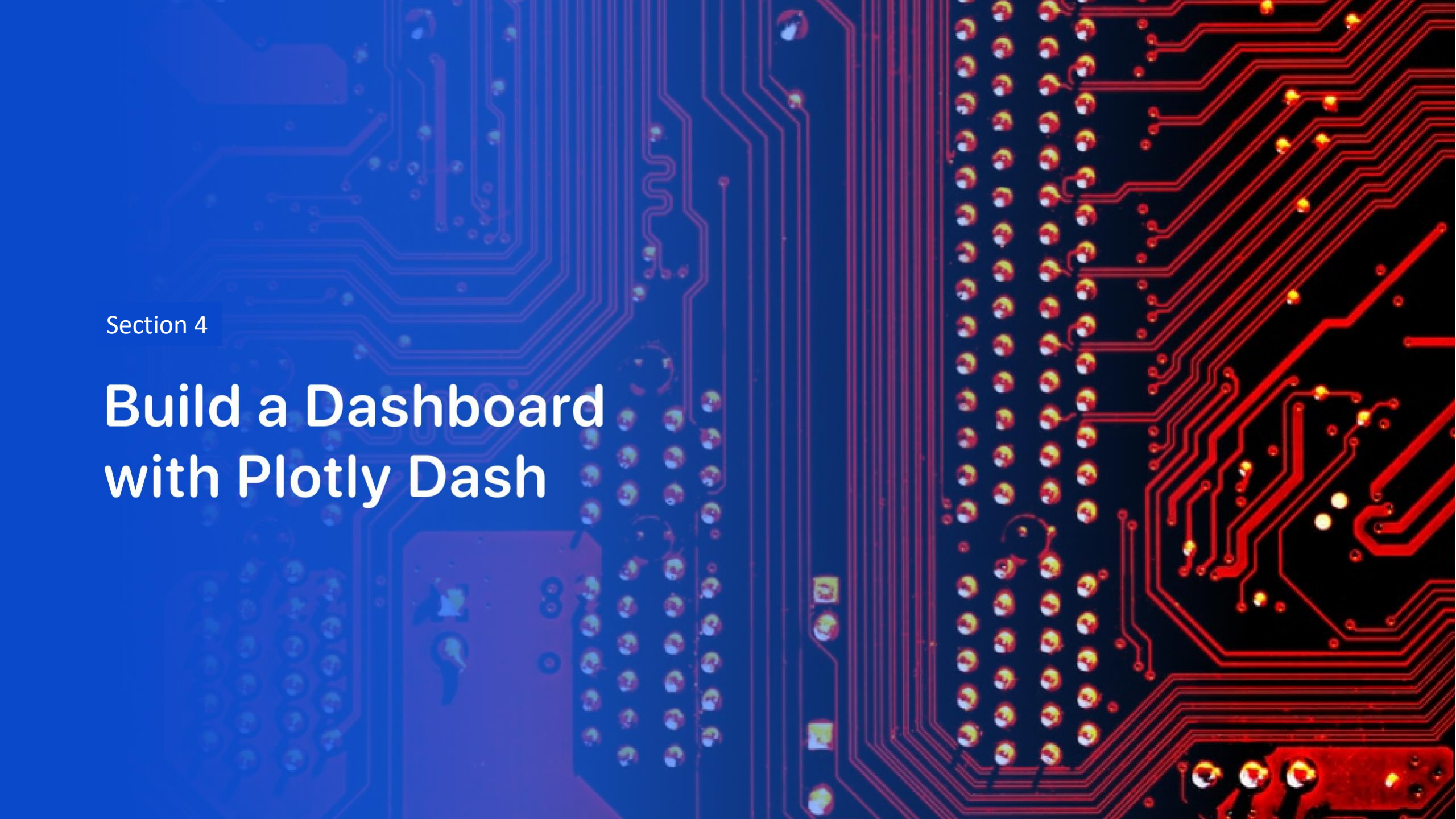
We can see that the SpaceX launch sites are in the United States of America coasts. Florida and California

Successful and failed outcomes marked on launch sites

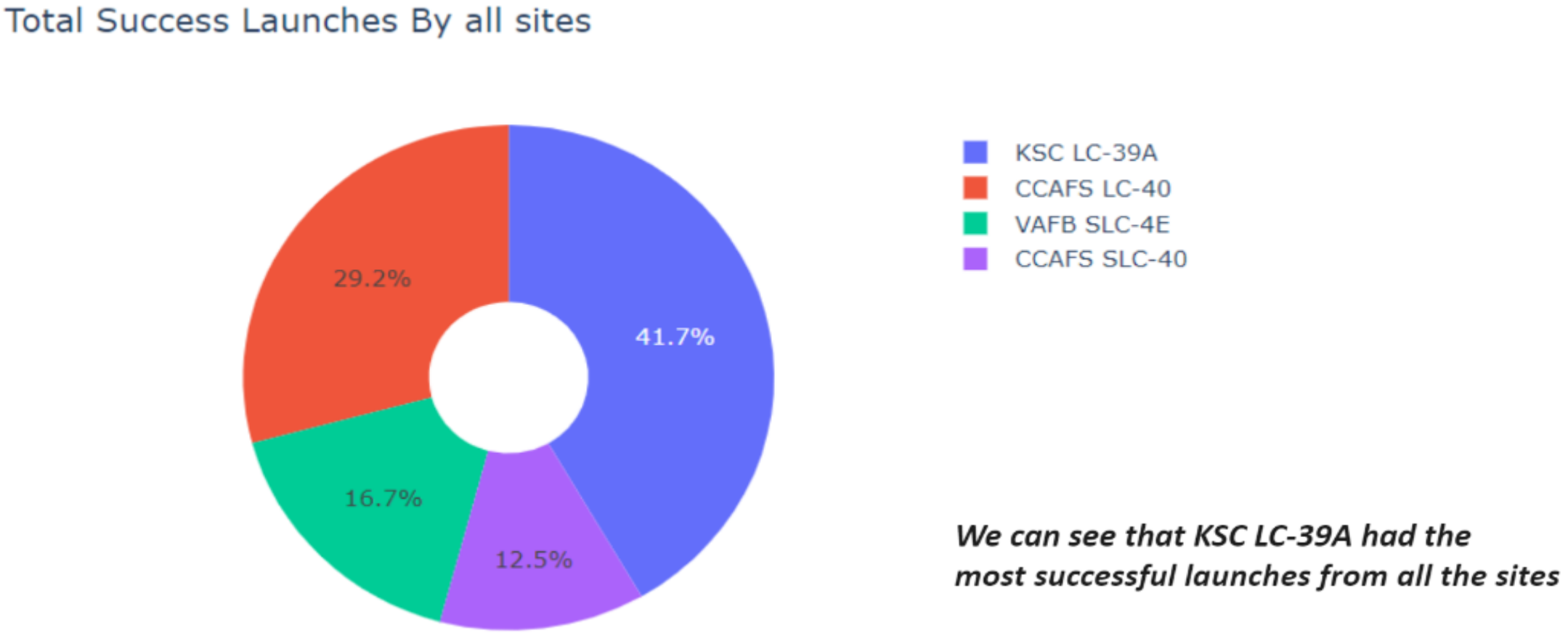
Distance from launch sites to landmarks



- 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

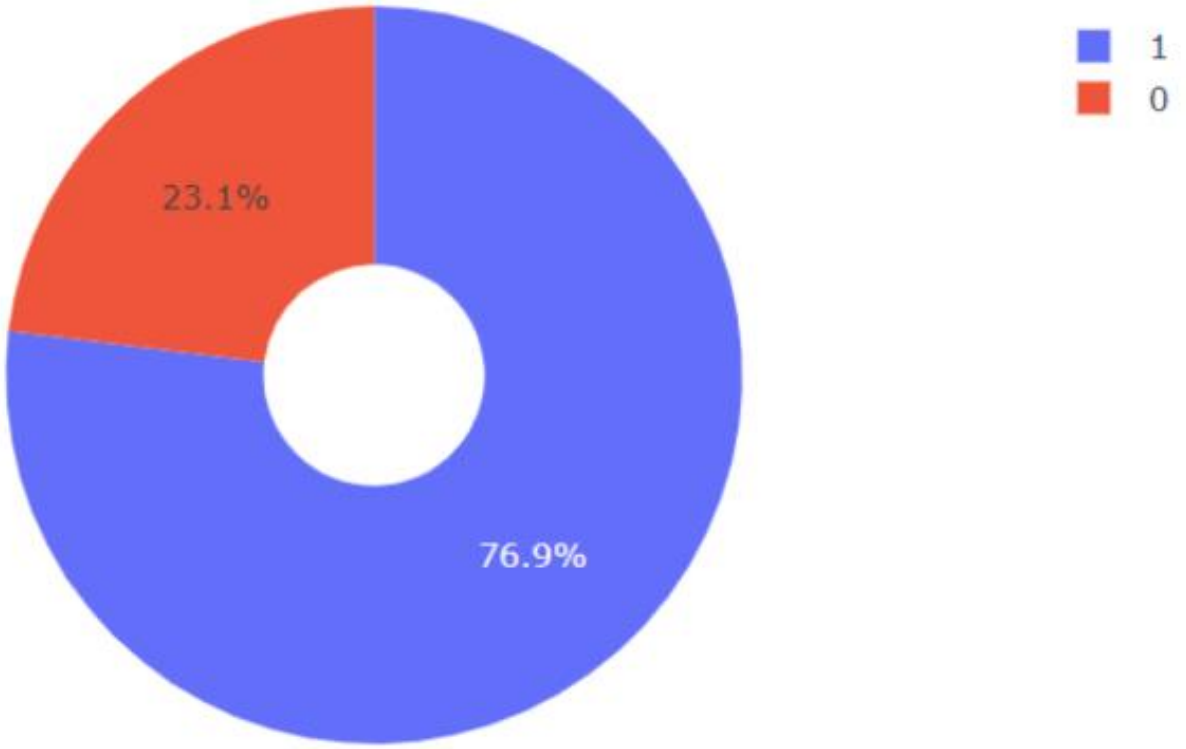

Section 4
Build a Dashboard with Plotly Dash

Success percentage by launch site



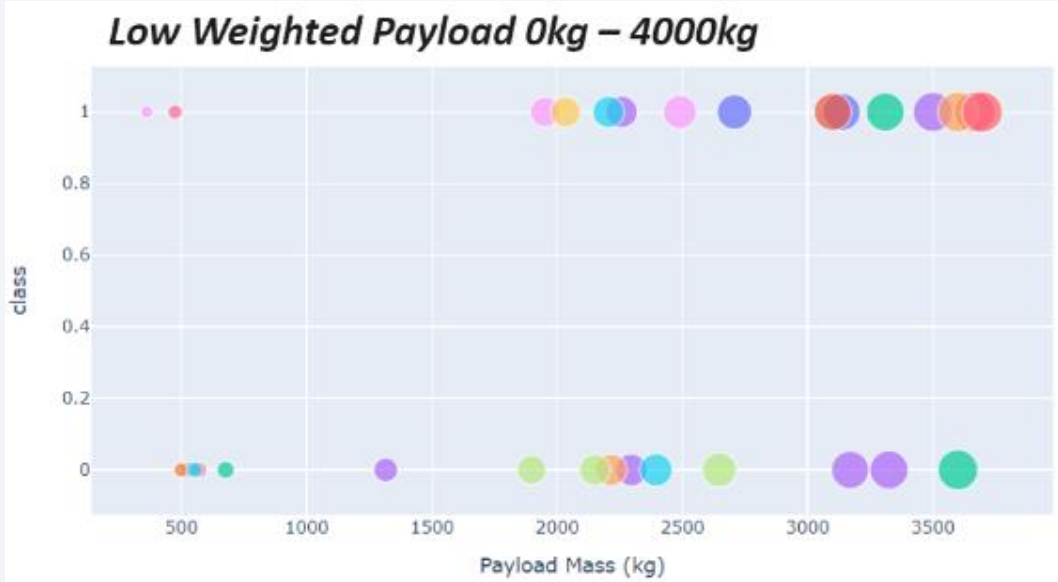
We can see that KSC LC-39A had the most successful launches from all the sites

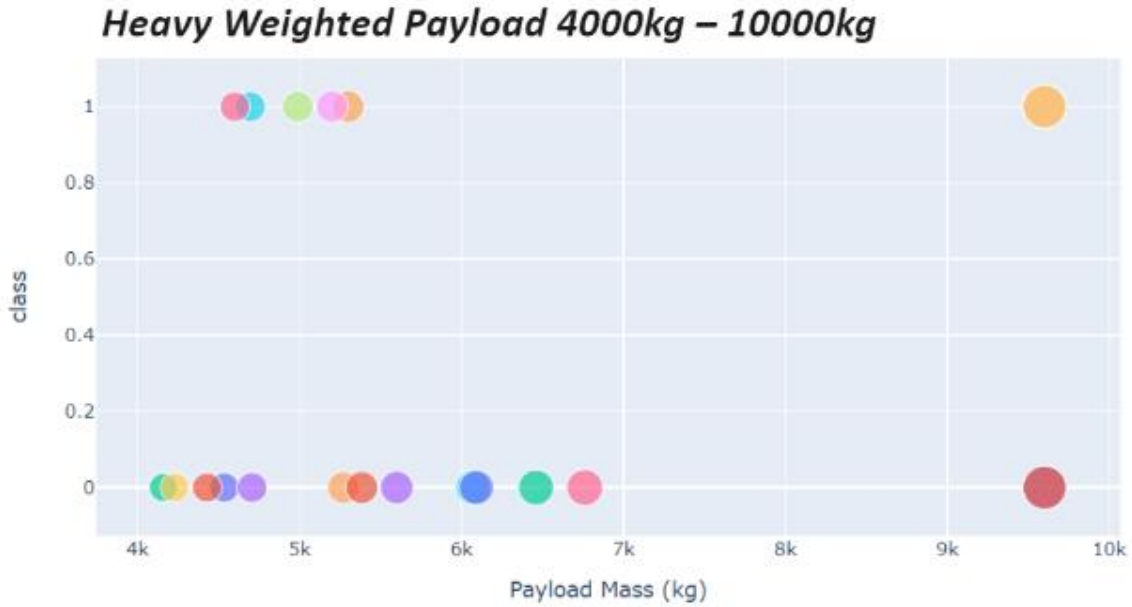
Highest success rate by a single launch site



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

Payload vs Launch Outcome for all sites





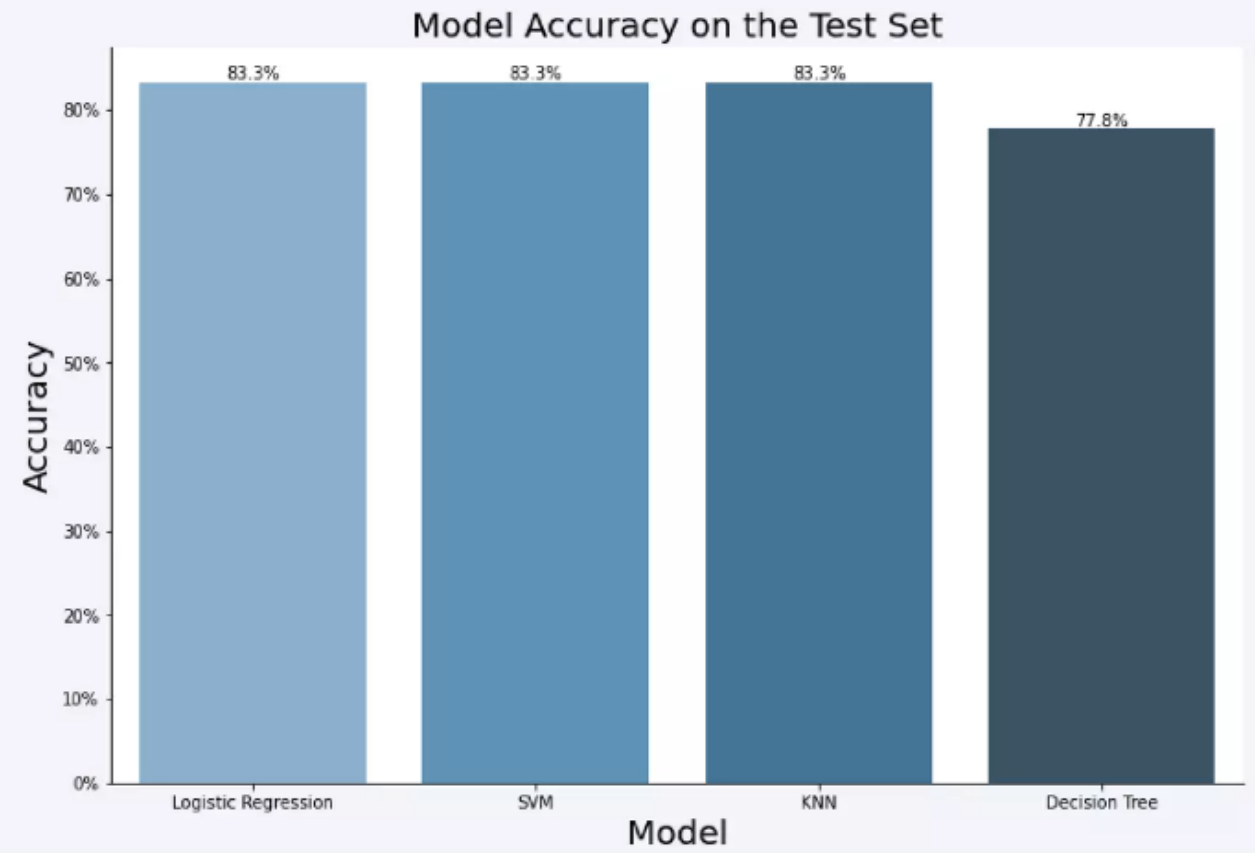
We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

Predictive Analysis (Classification)

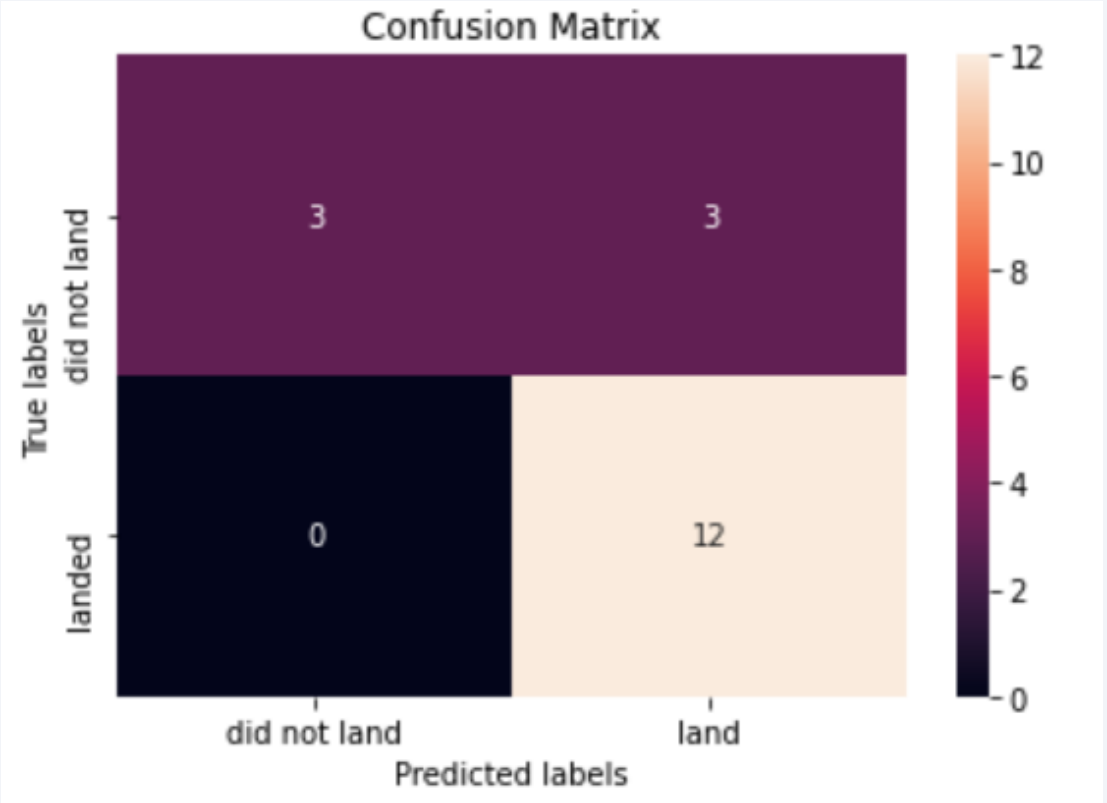
Classification Accuracy

- On the graph to the right we can see the accuracy of each model. As can be seen, there is a tie for the highest accuracy model between logistic regression, SVM and KNN.



Confusion Matrix

- Since there was a tie for the best performing model, there is not just one answer for the confusion matrix. However, all of the matrices show the same result which can be seen on the image.



Conclusions

- We can see a tie for the best prediction model. There is no correct answer since they give virtually the same result.
- Lighter payloads perform better than heavier ones.
- The launch site with the highest success rate is KSC LC 39A.
- The orbits ES-L1, GEO, HEO, SSO have the highest success rate.
- The launches are getting perfected over time and the trend is positive.

Thank you!