IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Tanuj Joshi

15/04/2024

Outline

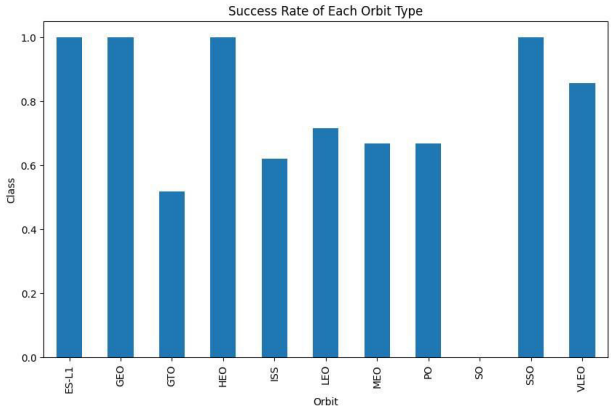
- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

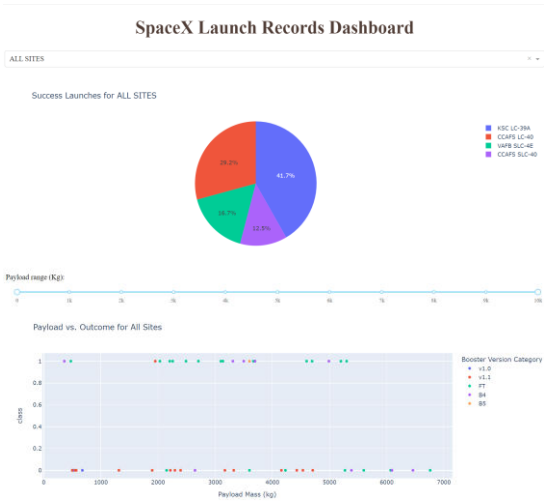
Executive Summary

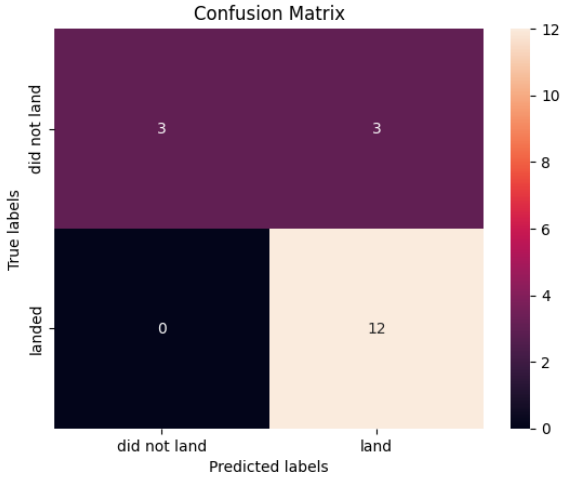
Summary of Methodologies:

This project follows these steps:

- Data Collection
- Data Wrangling
- Exploratory Data Analysis
- Interactive Visual Analytics
- Predictive Analysis (Classification)







Summary of Results:

This project produced following outputs and visualizations:

- Exploratory Data Analysis (EDA) results
- Geospatial analytics
- Interactive dashboard
- Predictive analysis of classification models

Introduction

- SpaceX launches Falcon 9 rockets at a cost of around $62m. This is considerably cheaper than other providers (which usually cost upwards of $165m), and much of the savings are because SpaceX can land, and then re-use the first stage of the rocket.
- If we can make predictions on whether the first stage will land, we can determine the cost of a launch, and use this information to assess whether or not an alternate company should bid and SpaceX for a rocket launch.
- This project will ultimately predict if the Space X Falcon 9 first stage will land successfully.
- **Important questions we want to find answers to:**

  -- How do variables such as payload mass, launch site, number of flights, and orbits affect the success of the first stage landing ?

  -- Does the rate of successful landings increase over the years ?

  -- What is the best algorithm that can be used for binary classification in this case ?

Section 1

Methodology

Methodology Summary

1. **Data Collection**
   - Making GET requests to the SpaceX REST API
   - Web Scraping

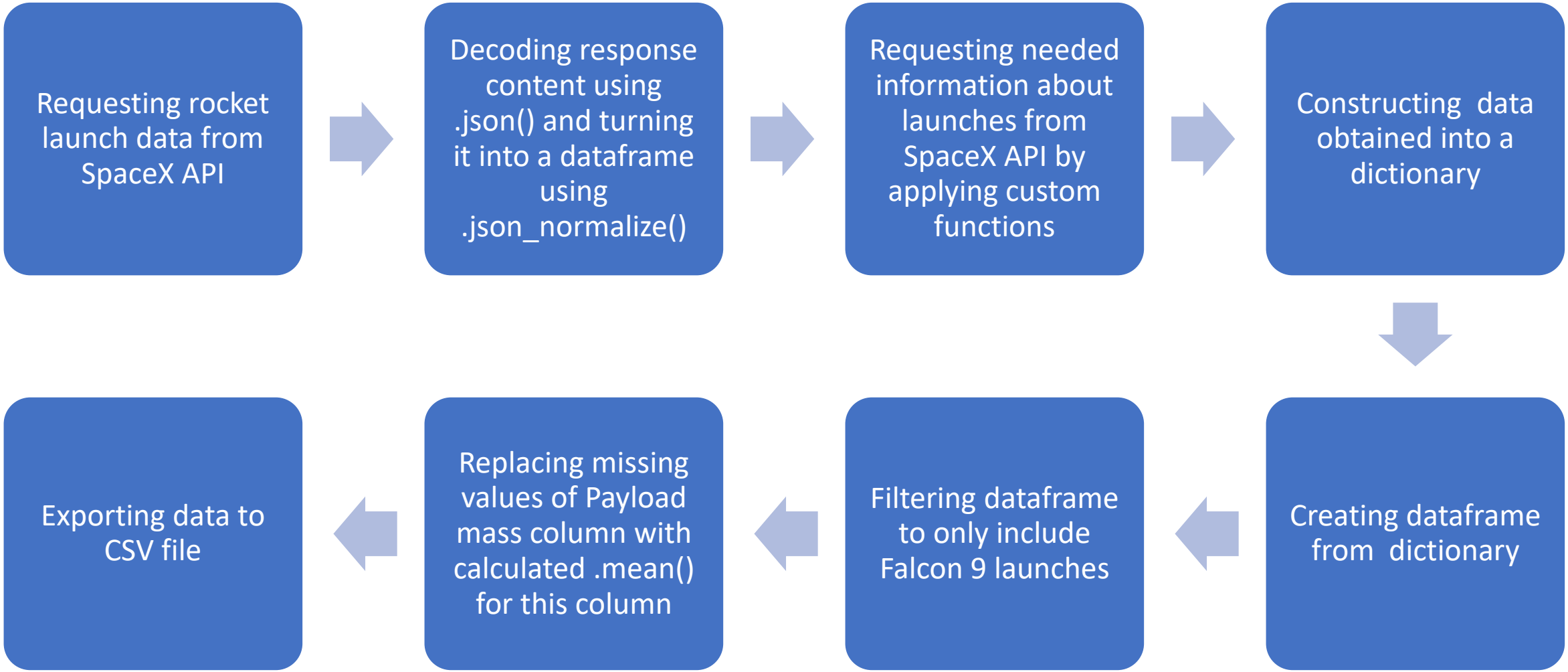
2. **Data Wrangling**
   - Using the .fillna() method to remove NaN values
   - Using the .value_counts() method to determine the following:
     - Number of launches on each site
     - Number and occurrence of each orbit
     - Number and occurrence of mission outcome per orbit type
   - Creating a landing outcome label that shows the following:
     - 0 when the booster did not land successfully
     - 1 when the booster did land successfully

3. **Exploratory Data Analysis**
   - Using SQL queries to manipulate and evaluate the SpaceX dataset
   - Using Pandas and Matplotlib to visualize relationships between variables, and determine patterns

4. **Interactive Visual Analytics**
   - Geospatial analytics using Folium
   - Creating an interactive dashboard using Plotly Dash

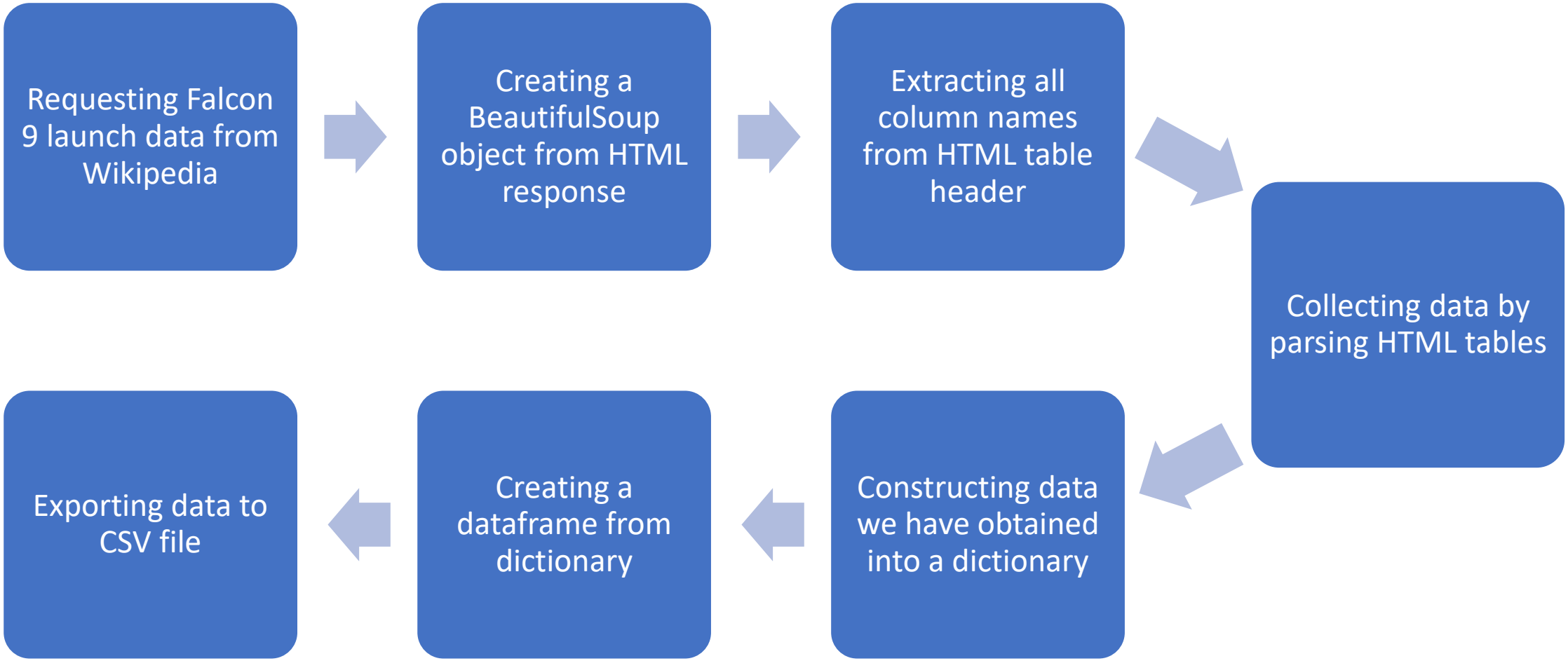
5. **Data Modelling and Evaluation**
   - Using Scikit-Learn to:
     - Pre-process (standardize) the data
     - Split the data into training and testing data using train_test_split
     - Train different classification models
     - Find hyperparameters using GridSearchCV
   - Plotting confusion matrices for each classification model
   - Assessing the accuracy of each classification model

Data Collection – SpaceX API



GitHub URL: Data Collection API

Data Collection – Web Scraping

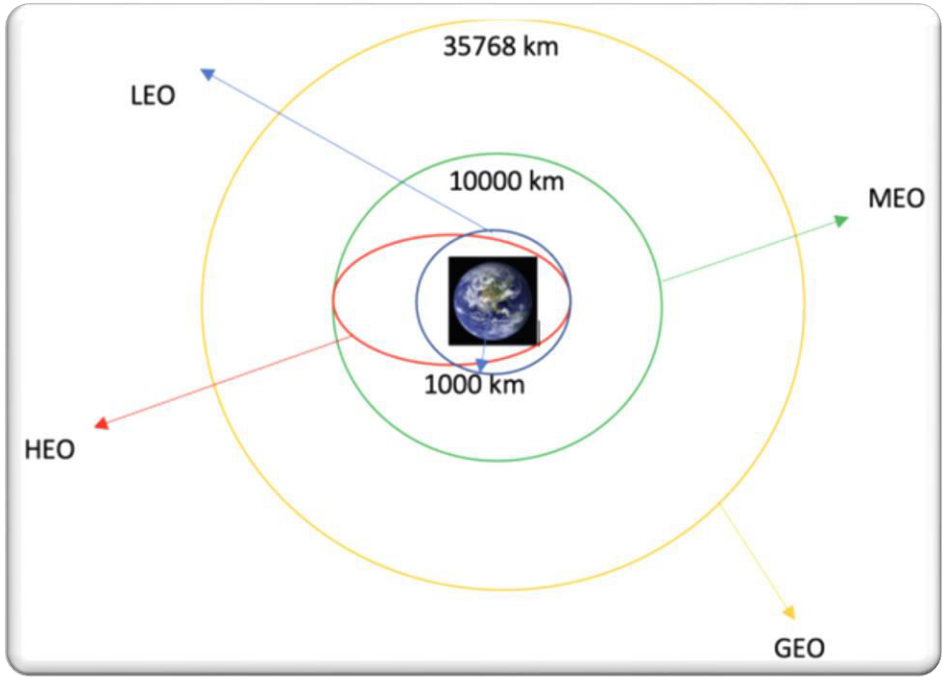


GitHub URL: Data Collection with Web Scraping

Data Wrangling - Pandas

Context

- The SpaceX dataset contains several Space X launch facilities, and each location is in the LaunchSite column.
- Each launch aims to a dedicated orbit, and some of the common orbit types are shown in the figure (right-side). The orbit type is in the Orbit column.



Initial Data Exploration:

Using the .value_counts() method to determine following:

1. Number of launches on each site
2. Number and occurrence of each orbit
3. Number and occurrence of landing outcome per orbit type

GitHub URL: Data Wrangling

Data Wrangling - Pandas

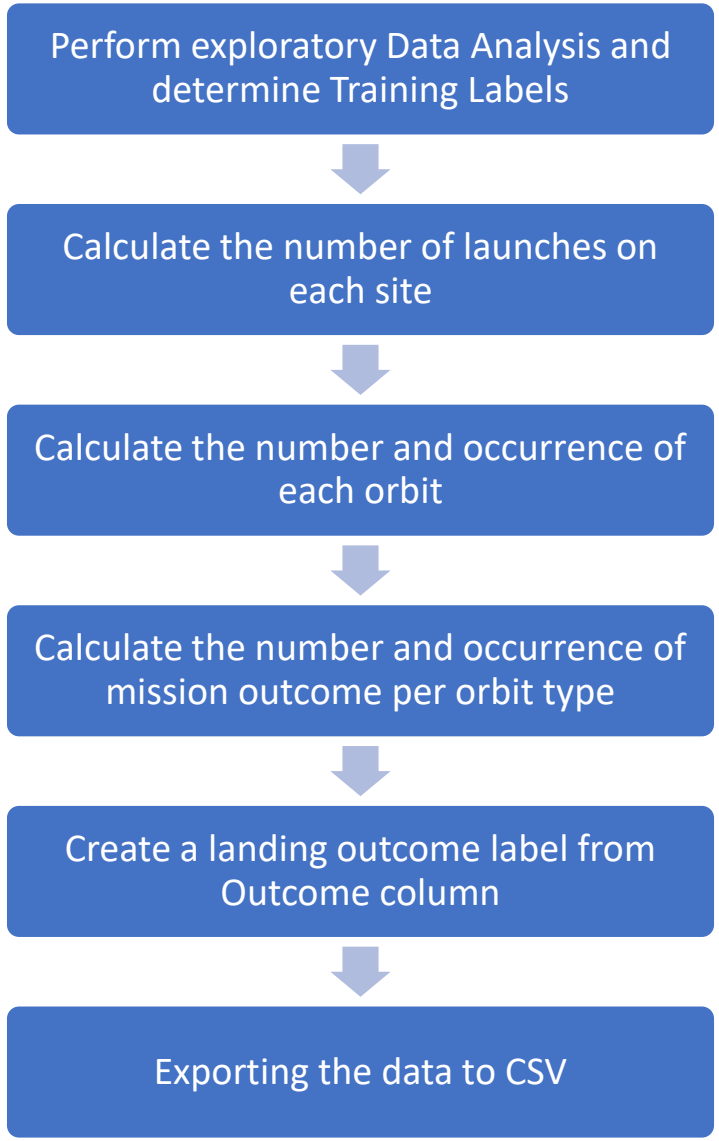
The landing outcome is shown in the Outcome column:

- True Ocean – the mission outcome was successfully landed to a specific region of the ocean
- False Ocean – the mission outcome was unsuccessfully landed to a specific region of the ocean.
- True RTLS – the mission outcome was successfully landed to a ground pad
- False RTLS – the mission outcome was unsuccessfully landed to a ground pad.
- True ASDS – the mission outcome was successfully landed to a drone ship
- False ASDS – the mission outcome was unsuccessfully landed to a drone ship.
- None ASDS and None None – these represent a failure to land.

Data Wrangling:

- To determine whether a booster will successfully land, it is best to have a binary column, i.e., where the value is 1 or 0, representing the success of the landing.
- This is done by:
  1. Defining a set of unsuccessful (bad) outcomes, bad_outcome
  2. Creating a list, landing_class, where the element is 0 if the corresponding row in Outcome is in the set bad_outcome, otherwise, it's 1.
  3. Create a Class column that contains the values from the list landing_class
  4. Export the DataFrame as a .csv file.

GitHub URL: Data Wrangling



Exploratory Data Analysis - Visualization

SCATTER CHARTS

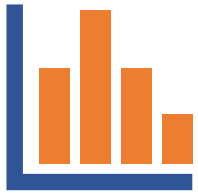
- Scatter charts were produced to visualize the relationships between:
- Flight Number and Launch Site
- Payload and Launch Site
- Orbit Type and Flight Number
- Payload and Orbit Type

- Scatter charts are useful to observe relationships, or correlations, between two numeric variables.

BAR CHART

- A bar chart was produced to visualize the relationship between:

  Success Rate and Orbit Type



- Bar charts are used to compare a numerical value to a categorical variable. Horizontal or vertical bar charts can be used, depending on the size of the data.

LINE CHARTS

- Line charts were produced to visualize the relationships between:
- Success Rate and Year (i.e. the launch success yearly trend)

- Line charts contain numerical values on both axes, and are generally used to show the change of a variable over time.

GitHub URL: Data Visualization

Exploratory Data Analysis - SQL

- **To gather some information about the dataset, some SQL queries were performed.**
- **The SQL queries performed on the data set were used to:**

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string ‘CCA’
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display the average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome on a ground pad was achieved
6. List the names of the boosters which had success on a drone ship and a payload mass between 4000 and 6000 kg
7. List the total number of successful and failed mission outcomes
8. List the names of the booster versions which have carried the maximum payload mass
9. List the failed landing outcomes on drone ships, their booster versions, and launch site names for 2015
10. Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

GitHub URL: EDA SQL

Geospatial Analysis - Folium

The following steps were taken to visualize the launch data on an interactive map:

1. **Mark all launch sites on a map**
   - Initialise the map using a Folium Map object
   - Add a folium.Circle and folium.Marker for each launch site on the launch map

2. **Mark the success/failed launches for each site on a map**
   - As many launches have the same coordinates, it makes sense to cluster them together.
   - Before clustering them, assign a marker colour of successful (class = 1) as green, and failed (class = 0) as red.
   - To put the launches into clusters, for each launch, add a folium.Marker to the MarkerCluster() object.
   - Create an icon as a text label, assigning the icon_color as the marker_colour determined previously.

3. **Calculate the distances between a launch site to its proximities**
   - To explore the proximities of launch sites, calculations of distances between points can be made using the Lat and Long values.
   - After marking a point using the Lat and Long values, create a folium.Marker object to show the distance.
   - To display the distance line between two points, draw a folium.PolyLine and add this to the map.

GitHub URL: Geospatial Analysis - Folium

Interactive Dashboard – Plotly Dash

The following plots were added to a Plotly Dashboard to have an interactive visualisation of the data:

1. Pie chart (px.pie()) showing the total successful launches per site
    - This makes it clear to see which sites are most successful
    - The chart could also be filtered (using a dcc.Dropdown() object) to see the success/failure ratio for an individual site

2. Scatter graph (px.scatter()) to show the correlation between outcome (success or not) and payload mass (kg)
    - This could be filtered (using a RangeSlider() object) by ranges of payload masses
    - It could also be filtered by booster version

GitHub URL: Interactive Dashboard – Plotly dash

Predictive Analysis (Classification)

Model Development

- To prepare the dataset for model development:
  - Load dataset
  - Perform necessary data transformations (standardise and pre-process)
  - Split data into training and test data sets, using train_test_split()
- Decide which type of machine learning algorithms are most appropriate
- For each chosen algorithm:
  - Create a GridSearchCV object and a dictionary of parameters
  - Fit the object to the parameters
  - Use the training data set to train the model

Model Evaluation

- For each chosen algorithm:
  - Using the output GridSearchCV object:
    - ➢ Check the tuned hyperparameters (best_params_)
    - ➢ Check the accuracy (score and best_score_)
- Plot and examine the Confusion Matrix

Finding the Best Classification Model

- Review the accuracy scores for all chosen algorithms
- The model with the highest accuracy score is determined as the best performing model

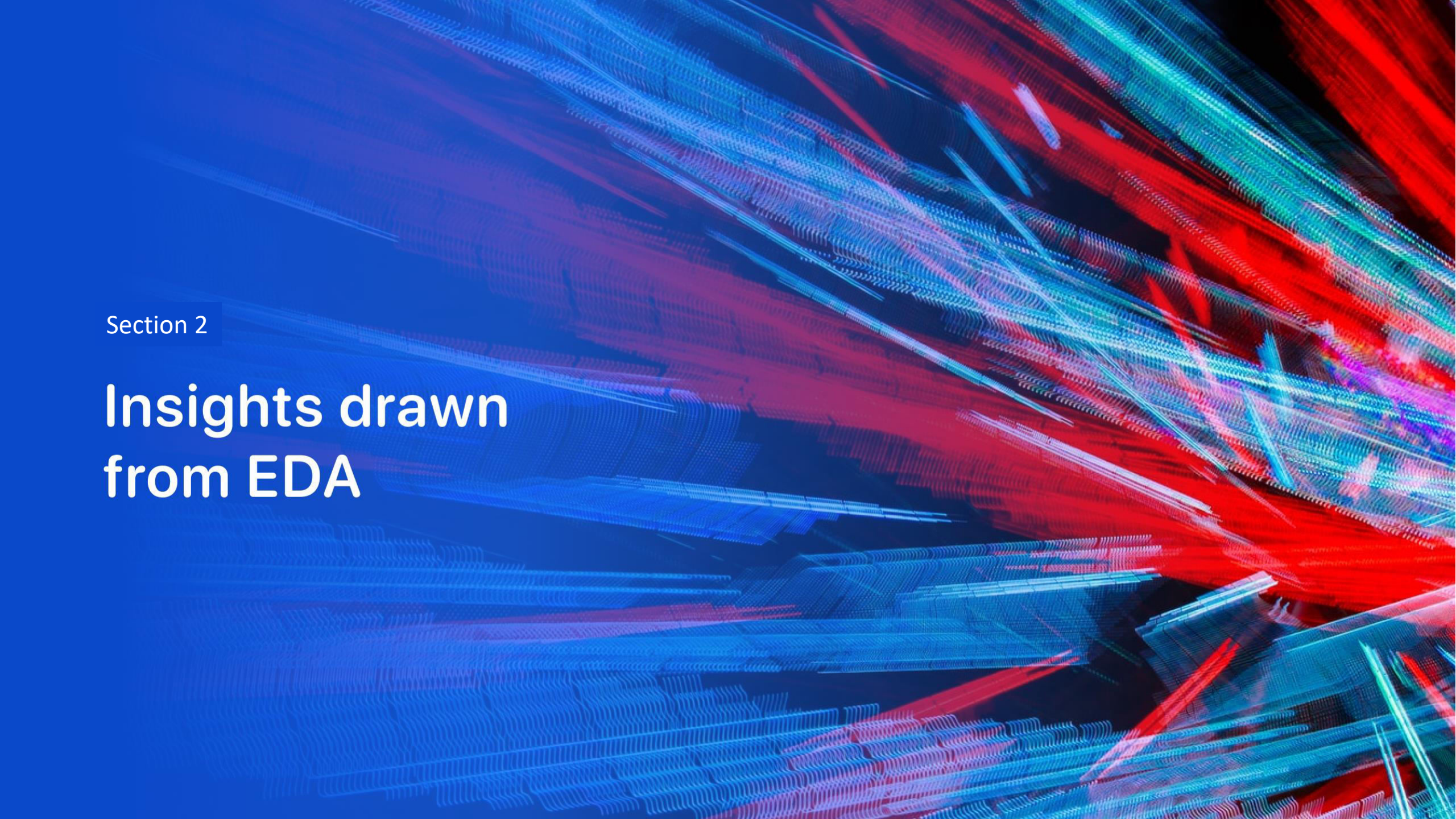
GitHub URL: Predictive Analysis

Results

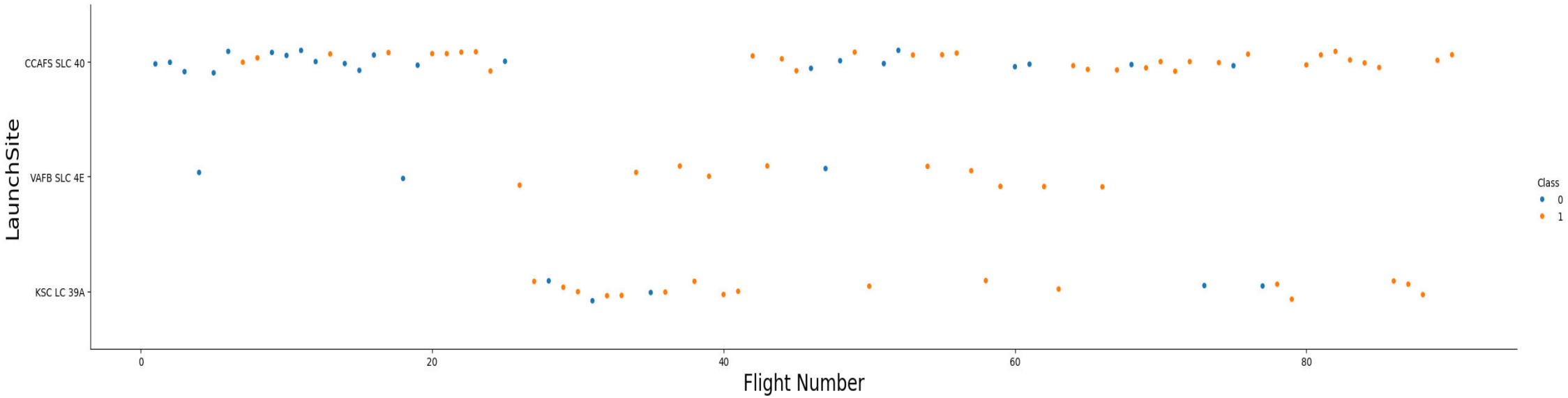
Exploratory Data Analysis

Interactive Analytics

Predictive Analysis

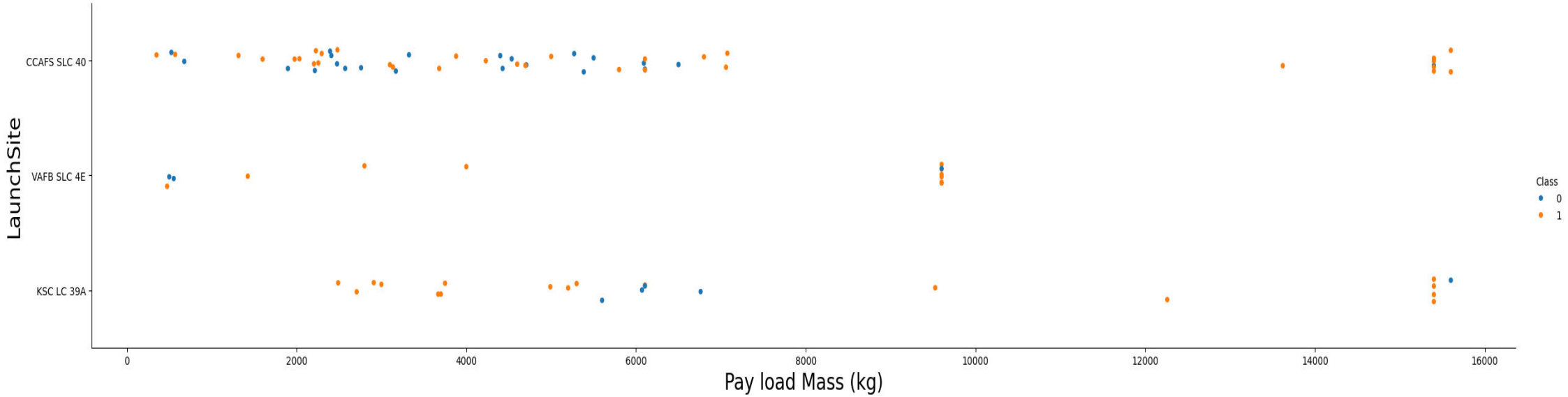

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



- As the number of flights increases, the rate of success at a launch site increases.
- Most of the early flights (flight numbers < 30) were launched from CCAFS SLC 40, and almost 50% of them were unsuccessful.
- The flights from VAFB SLC 4E also show this trend, that later flights were more successful than earlier fights
- No early flights were launched from KSC LC 39A, so the launches from this site are more successful.
- Above a flight number of around 30, there are significantly more successful landings (Class = 1).

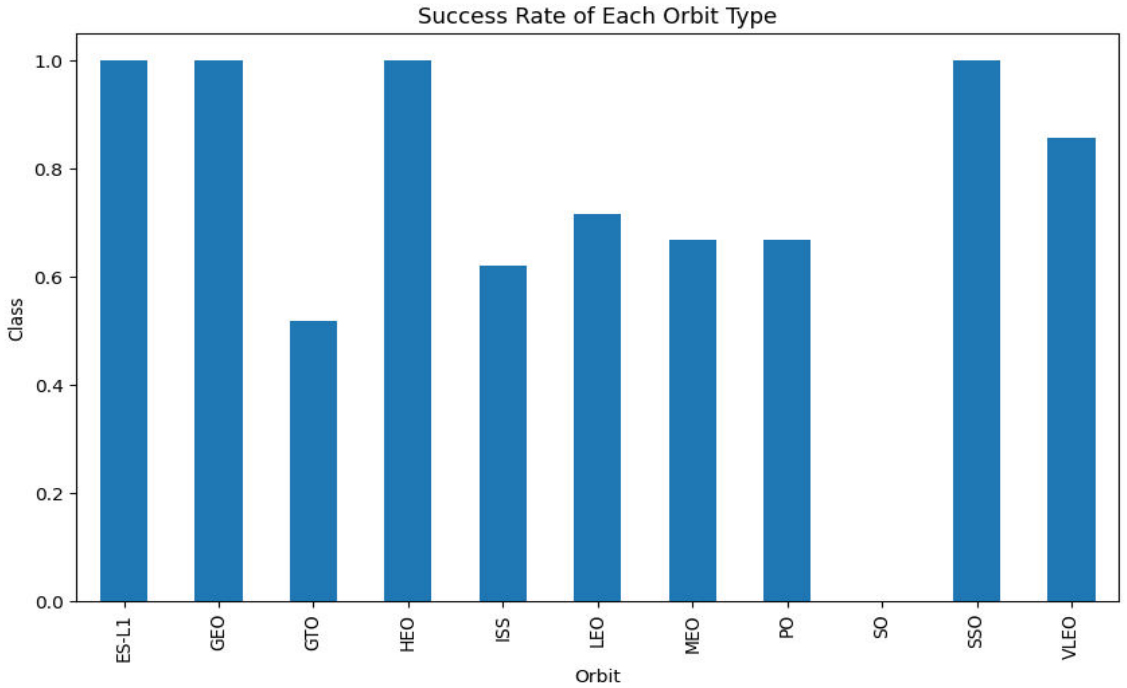
Payload vs. Launch Site



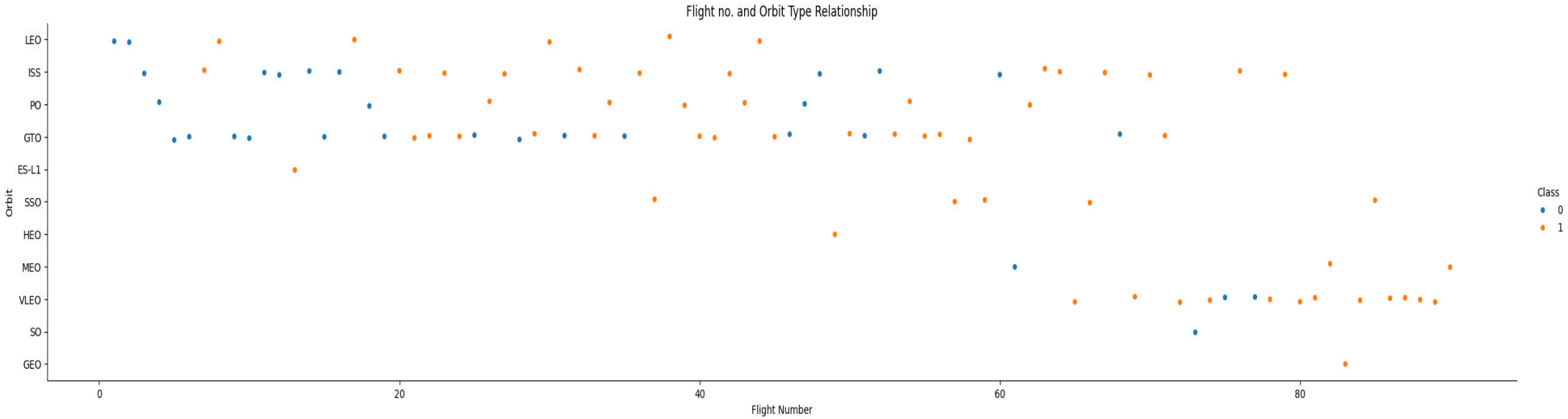
- Above a payload mass of around 7000 kg, there are very few unsuccessful landings, but there is also far less data for these heavier launches.
- There is no clear correlation between payload mass and success rate for a given launch site.
- All sites launched a variety of payload masses, with most of the launches from CCAFS SLC 40 being comparatively lighter payloads (with some outliers).

Success Rate vs. Orbit Type

- The bar chart shows that following orbits have the highest (100%) success rate:
  - ES-L1 (Earth-Sun First Lagrangian Point)
  - GEO (Geostationary Orbit)
  - HEO (High Earth Orbit)
  - SSO (Sun-synchronous Orbit)
- The orbit with the lowest (0%) success rate is ‘SO’ (Heliocentric Orbit)
- Orbits with success rate between 50% and 85%: - ‘GTO’, ‘ISS’, ‘LEO’, ’MEO’, ‘PO’

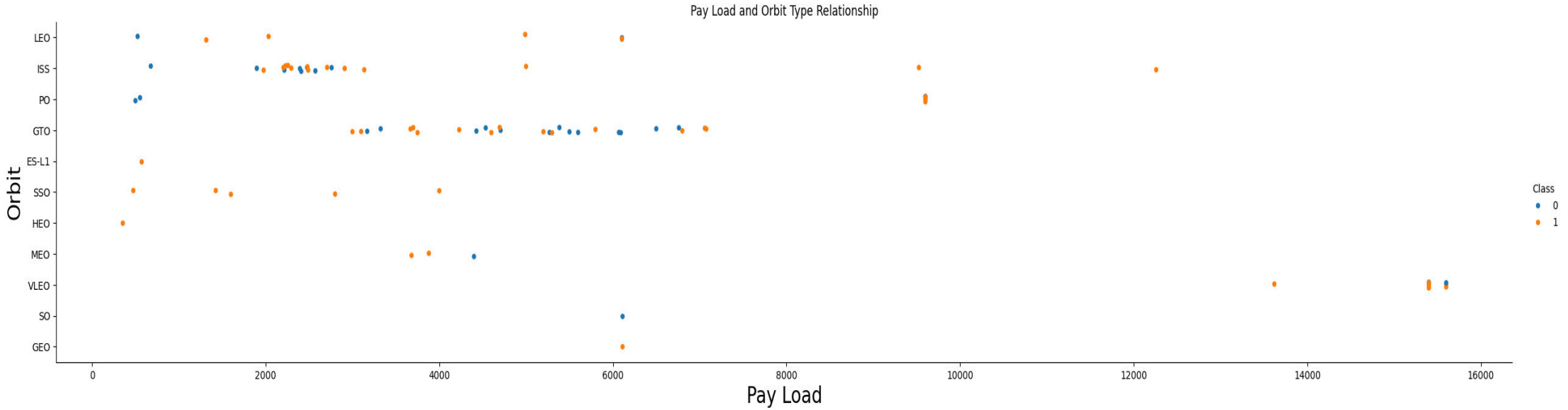


Flight Number vs. Orbit Type



- The 100% success rate of GEO, HEO, and ES-L1 orbits can be explained by only having 1 flight into the respective orbits.
- The SSO has 100% success rate, with 5 successful flights.
- There is little relationship between Flight Number and Success Rate for GTO.
- Generally, as Flight Number increases, the success rate increases. This is most extreme for LEO, where unsuccessful landings only occurred for the low flight numbers (early launches).

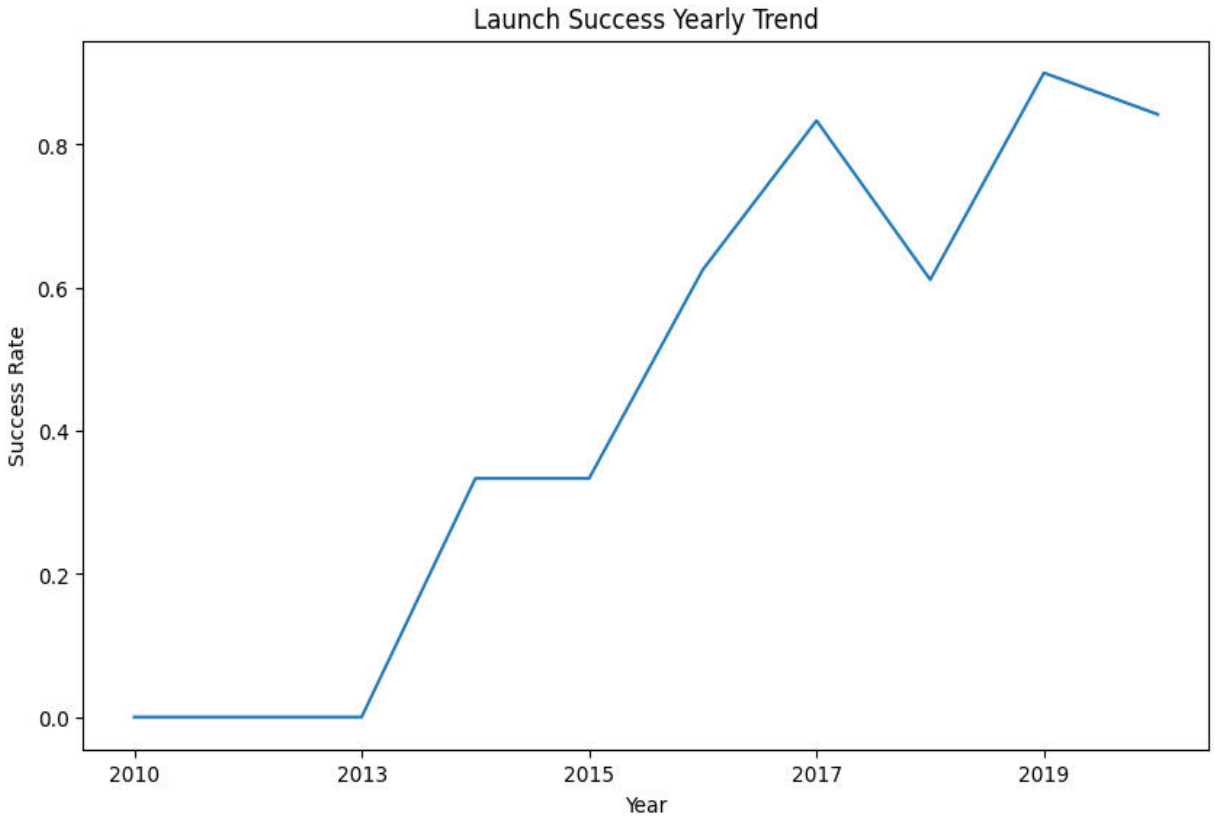
Payload vs. Orbit Type



- The following orbit types have more success with heavy payloads:
  - PO (although the number of data points is small)
  - ISS
  - LEO
- For GTO, the relationship between payload mass and success rate is unclear.
- VLEO (Very Low Earth Orbit) launches are associated with heavier payloads.

Launch Success Yearly Trend

- Between 2010 and 2013, all landings were unsuccessful (success rate is 0).
- After 2013, the success rate generally increased, despite small dips in 2018 and 2020.
- After 2016, there was always a greater than 50% chance of success.



All Launch Site Names

```
%sql select DISTINCT Launch_Site from SPACEXTABLE
```

```
 * sqlite:///my_data1.db
Done.
```

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

The word DISTINCT returns only unique values from the Launch_Site column of the SPACEXTABLE table

Launch Site Names begin with ‘CCA’

```
%sql select * from SPACEXTABLE where Launch_Site like 'CCA%' LIMIT 5
```

Python

```
 * sqlite:///my_data1.db
Done.
```

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

LIMIT 5 fetches only 5 records, and the like keyword is used with the wild card ‘CCA%’ to retrieve string values beginning with ‘CCA’

Total Payload Mass

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer = 'NASA (CRS)'
```

```
 * sqlite:///my_data1.db
Done.
```

sum(PAYLOAD_MASS__KG_)
45596

The sum keyword is used to calculate the total of the Launch column, and the sum keyword (and the associated condition) filters the results to only boosters from ‘NASA (CRS)’

Average Payload Mass by F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like 'F9 v1.1'
```

```
 * sqlite:///my_data1.db
Done.
```

avg(PAYLOAD_MASS__KG_)
2928.4

The avg keyword is used to calculate the average of the PAYLOAD_MASS__KG_ column, and the where keyword (and the associated condition) filters the results to only the ‘F9 v1.1’ booster version

First Successful Ground Landing Date

```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'
```

```
 * sqlite:///my_data1.db
Done.
```

min(Date)
2015-12-22

The min keyword is used to calculate the minimum of the Date column, i.e. the first date, and the where keyword (and the associated condition) filters the results to only the successful ground pad landings.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select Booster_Version from SPACEXTABLE where Landing_Outcome = 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

```
 * sqlite:///my_data1.db
Done.
```

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

The where keyword is used to filter the results to include only those that satisfy both conditions in the brackets (as the and keyword is also used). The and keyword allows for 4000 < x < 6000 values to be selected.

Total Number of Successful and Failure Mission Outcomes

```
%sql select Mission_Outcome, count(Mission_Outcome) as total_number from SPACEXTABLE group by Mission_Outcome
```

✓ 0.0s

```
 * sqlite:///my_data1.db
Done.
```

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

The count keyword is used to calculate the total number of mission outcomes, and the group by keyword is also used to group these results by the type of mission outcome

Boosters Carried Maximum Payload

```
%sql select distinct(Booster_Version) from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

✓ 0.0s

```
 * sqlite:///my_data1.db
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

A subquery is used here. The select statement within the brackets finds the maximum payload, and this value is used in the where condition. The distinct keyword is then used to retrieve only distinct /unique booster versions

2015 Launch Records

```
%sql select substr(Date, 6, 2) as Month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where Date like '2015%' and Landing_Outcome = 'Failure (drone ship)'
```

```
 * sqlite:///my_data1.db
Done.
```

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

The where keyword is used to filter the results for only failed landing outcomes,

and only for the year of 2015

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

```
%sql select Landing_Outcome, count(*) as no_of_outcomes,  rank() over (order by count(*) desc) as out_rank from SPACEXTABLE where \
Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by no_of_outcomes desc
```

✓ 0.0s

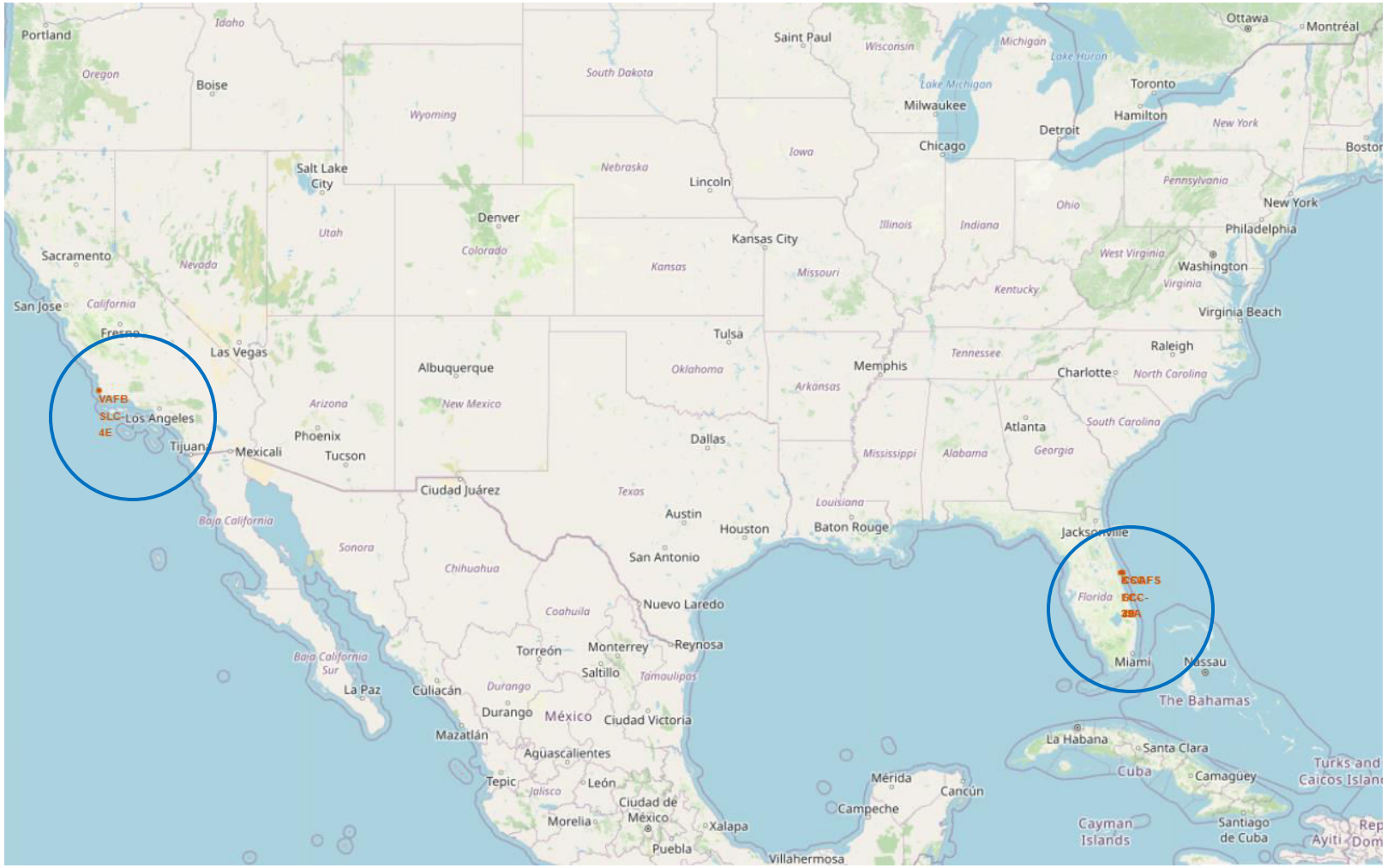
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	no_of_outcomes	out_rank
No attempt	10	1
Success (drone ship)	5	2
Failure (drone ship)	5	2
Success (ground pad)	3	4
Controlled (ocean)	3	4
Uncontrolled (ocean)	2	6
Failure (parachute)	2	6
Precluded (drone ship)	1	8

The where keyword is used with the between keyword to filter the results to dates only within those specified. The results are then grouped and ordered, using the keywords group by and order by, respectively, where desc is used to specify the descending order


Section 3
Launch Sites
Proximities Analysis

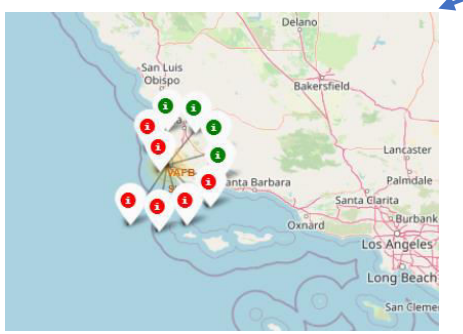
All Launch Sites on Map



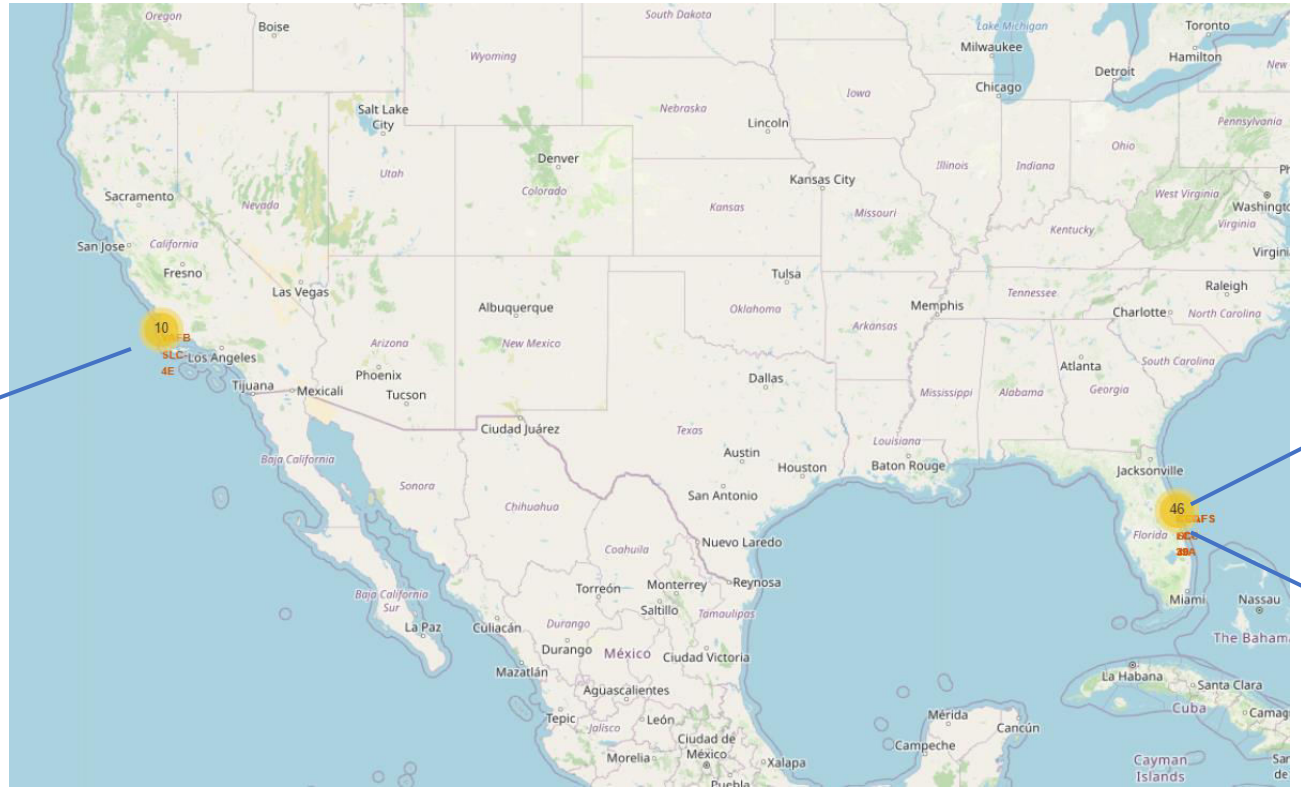
All SpaceX launch sites are on coasts of the United States of America, specifically **Florida** and **California**.

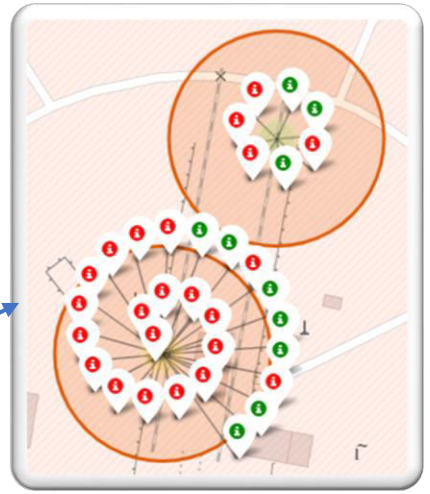
Success/Failed Launches from each Site

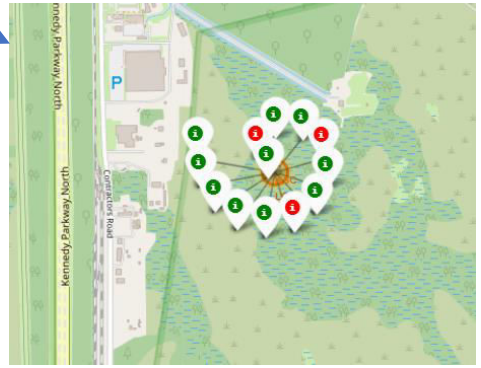
CCAFS SLC-40 and
CCAFS LC-40



VAFB SLC-4E







KSC LC-39A

- Launches have been grouped into clusters, and annotated with green icons for successful launches, and red icons for failed launches
- Launch Site KSC LC-39A has a very high Success Rate

Proximity of Launch Sites to other points of Interest

Using the CCAFS SLC-40 launch site as an example site, we can understand more about the placement of launch sites.

Are launch sites in close proximity to railways?

- Yes. The coastline is only 0.87 km due East

Are launch sites in close proximity to highways?

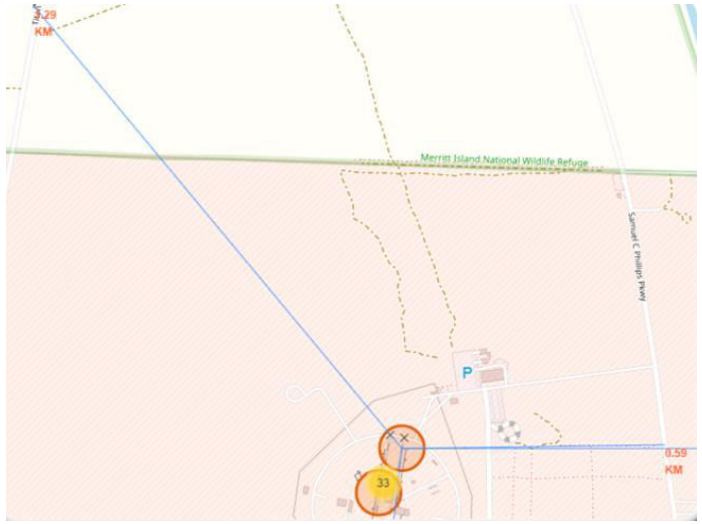
- Yes. The nearest highway is only 0.59km away.

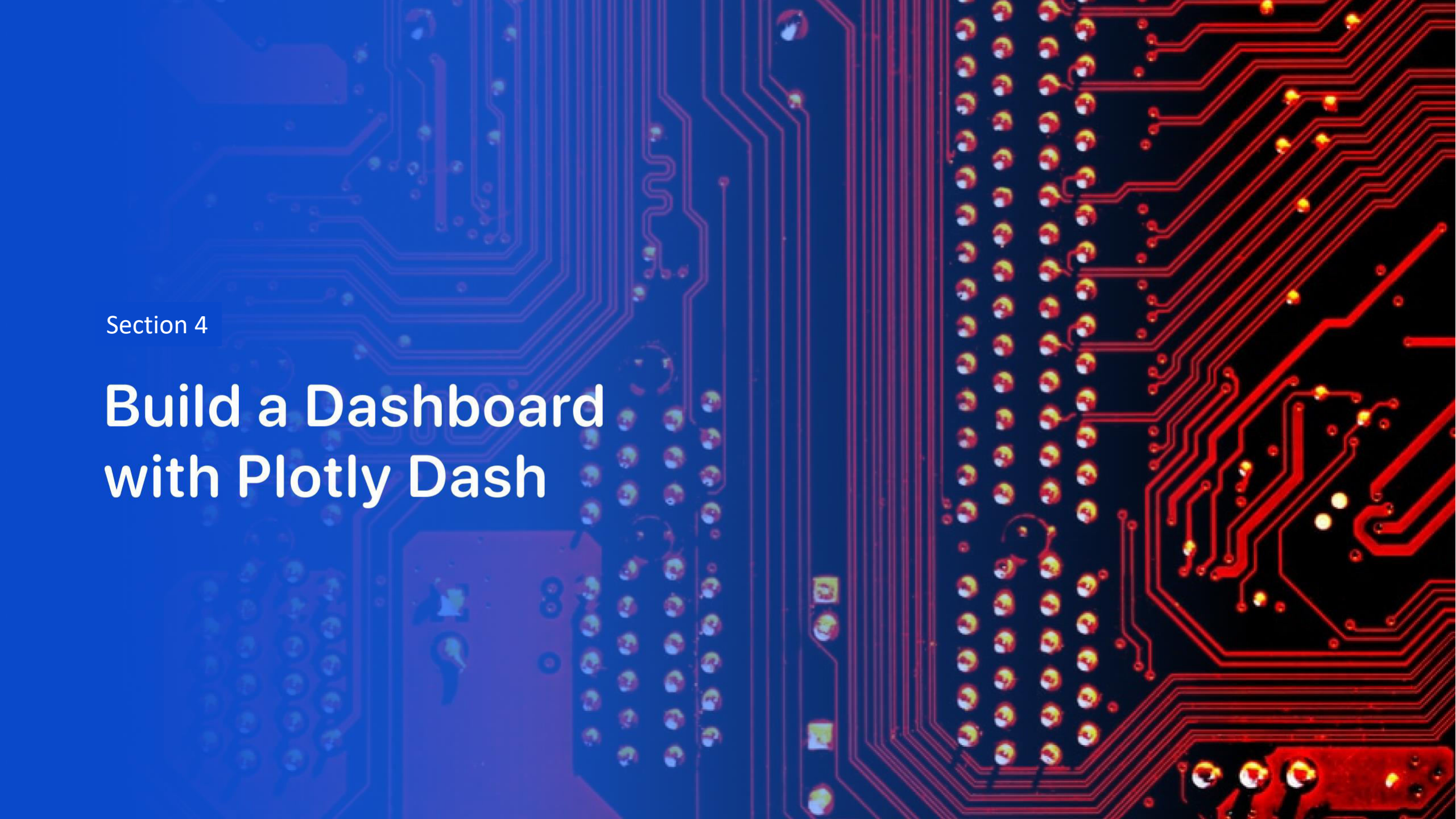
Are launch sites in close proximity to railways?

- Yes. The nearest railway is only 1.29 km away.

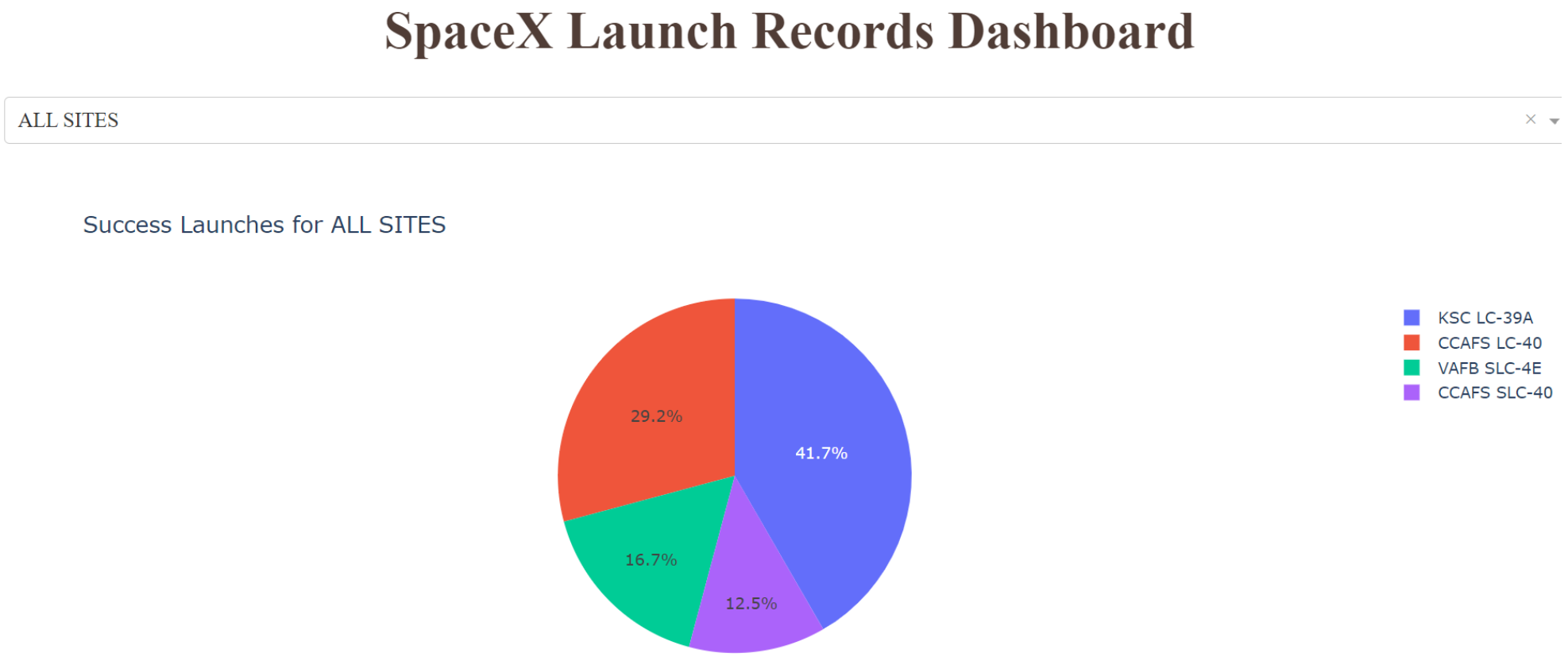
Do launch sites keep certain distance away from cities?

- Yes. The nearest city is 51.74 km away.



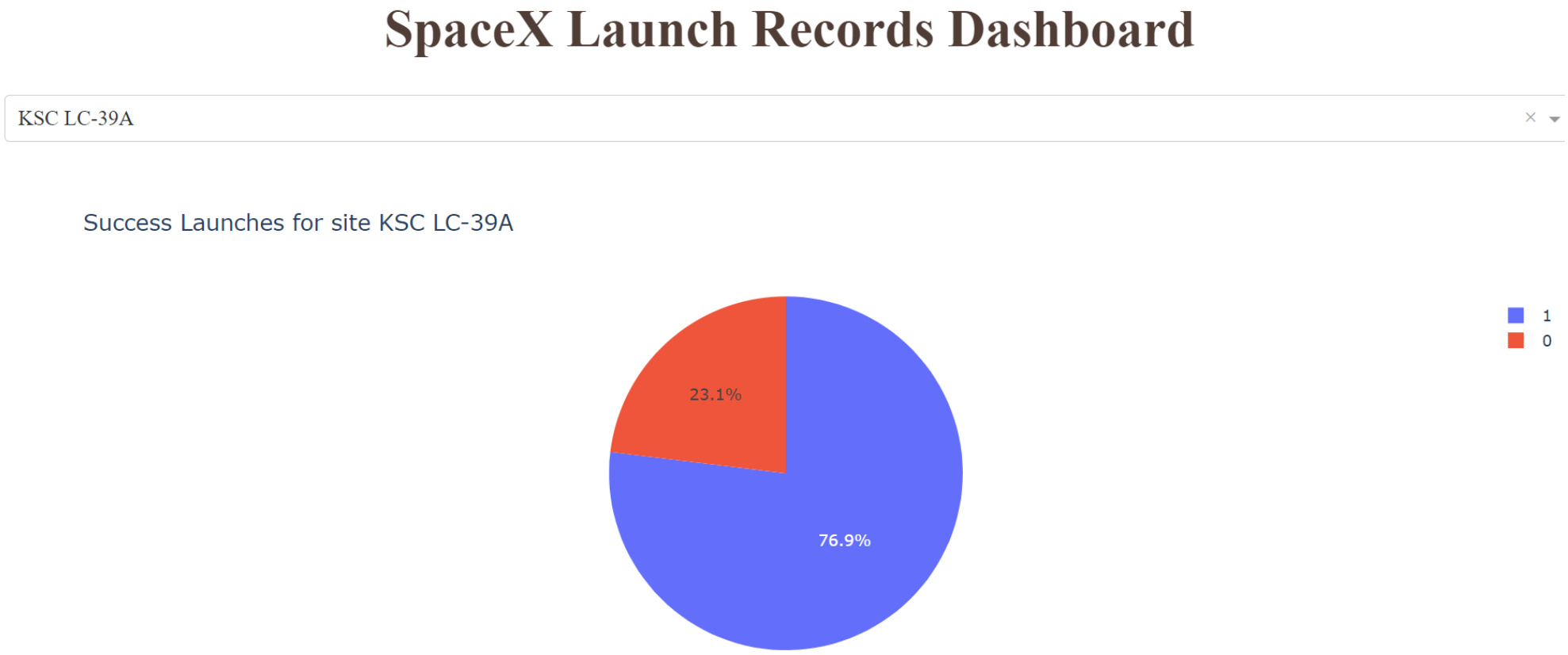

Section 4
Build a Dashboard with Plotly Dash

Launch Success counts for All Sites



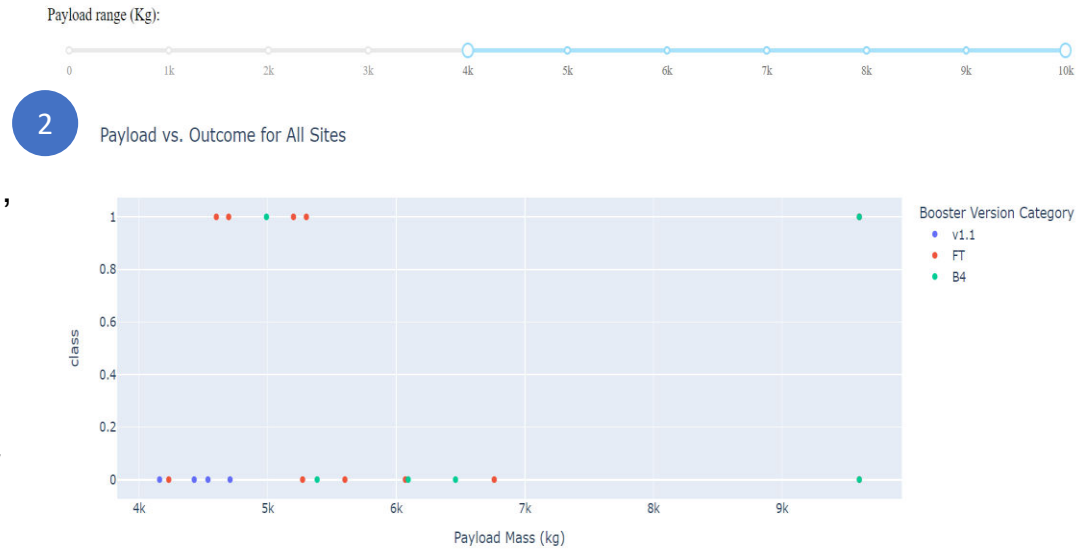
The launch site KSC LC-39 A had the most successful launches, with 41.7% of the total successful launches

Pie chart for the launch site with highest launch success ratio



The launch site KSC LC-39 A had the highest launch successful rate (76.9%) with 10 successful and only 3 failed landings.

Launch Outcome vs. Payload Scatter Plot for All Sites



- Plotting the launch outcome vs. payload for all sites shows a gap around 4000 kg, so it makes sense to split the data into 2 ranges:
  - 0 – 4000 kg (smaller payloads)
  - 4000 – 10000 kg (bigger payloads)

- From these 2 plots, it can be shown that, **success for bigger payloads is lower than that for smaller payloads.**

Section 5

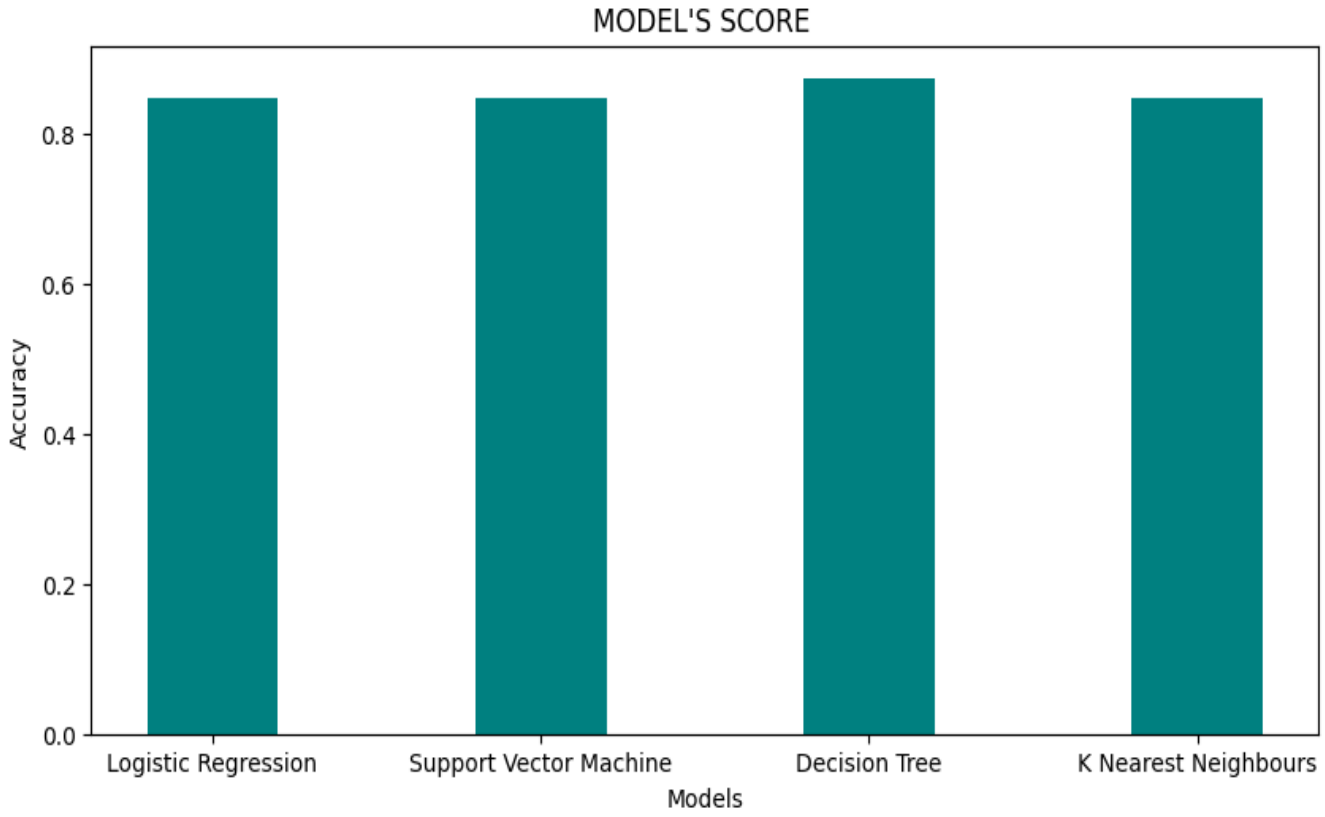
Predictive Analysis (Classification)

Classification Accuracy

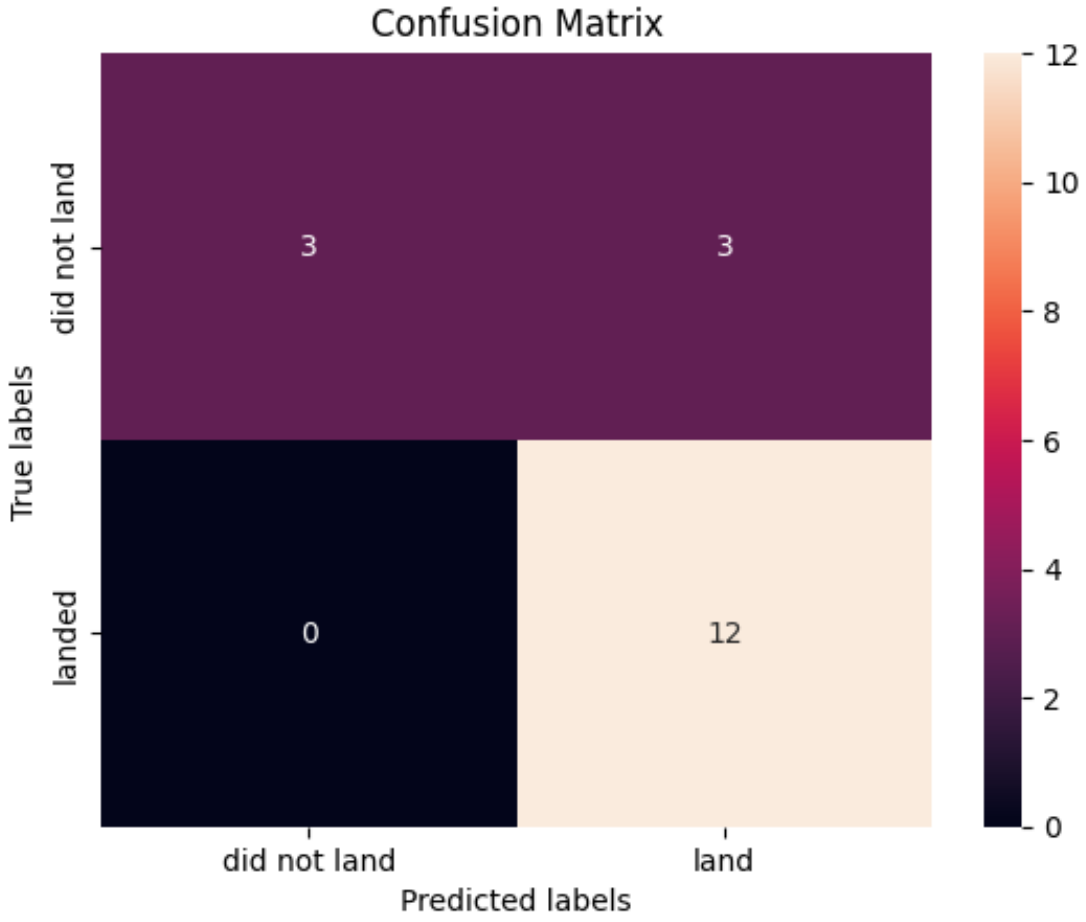
Plotting the Accuracy Score and Best Score for each classification algorithm produces the following result:

- The Decision Tree model has the highest training accuracy of 87.32 %
- All the models have same accuracy score of 83.33% on test set

	Algorithm	Accuracy Score	Best Score
0	Logistic Regression	0.833333	0.846429
1	Support Vector Machine	0.833333	0.848214
2	Decision Tree	0.833333	0.873214
3	K Nearest Neighbours	0.833333	0.848214



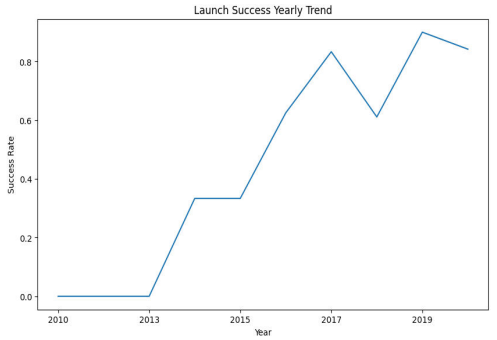
Confusion Matrix

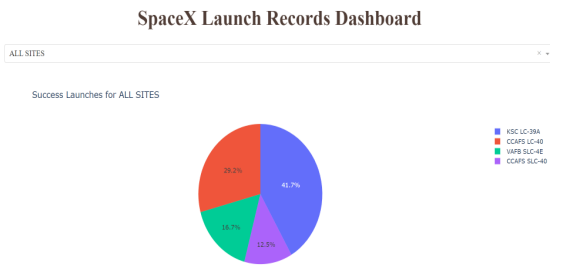


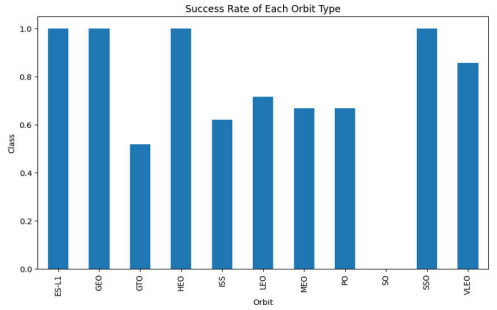
- As shown previously, best performing classification model is the Decision Tree model, with an accuracy of 87.32%
- Confusion matrix, is same for all models.
- It shows 3 out of 18 total results classified incorrectly (a false positive, shown in the top-right corner)
- The other 15 results are correctly classified (3 did not land, 12 did land)

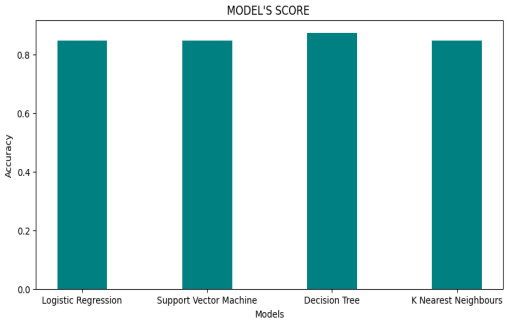
Conclusions

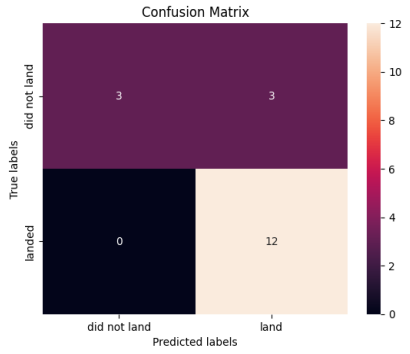
- As the number of flights increases, the rate of success at a launch site increases, with most early flights being unsuccessful. i.e., with more experience, the success rate increases.
- Orbit types ES-L1, GEO, HEO, and SSO, have the highest (100%) success rate.
- The launch site KSC LC-39 A had the most successful launches, with 41.7% of the total successful launches, and also the highest rate of successful launches, with a 76.9% success rate.
- The success for bigger payloads (over 4000kg) is lower than that for smaller payloads.
- The best performing classification model is the Decision Tree model, with an accuracy of 87.32%.











Thank you!