Winning Space Race with Data Science

Ye Myint Oo
18.6.2022

Outline

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

Executive Summary

Summary of Methodologies:

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

Summary of Results:

- Exploratory Data Analysis Results
- Geospatial Analytics
- Interactive Visual Analytics
- Predictive Analysis Results (Classification)

Introduction

- 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
- if we can determine if the first stage will land, we can determine the cost of a launch
- In this project, we will predict if the Falcon 9 first stage will land successfully

Section 1

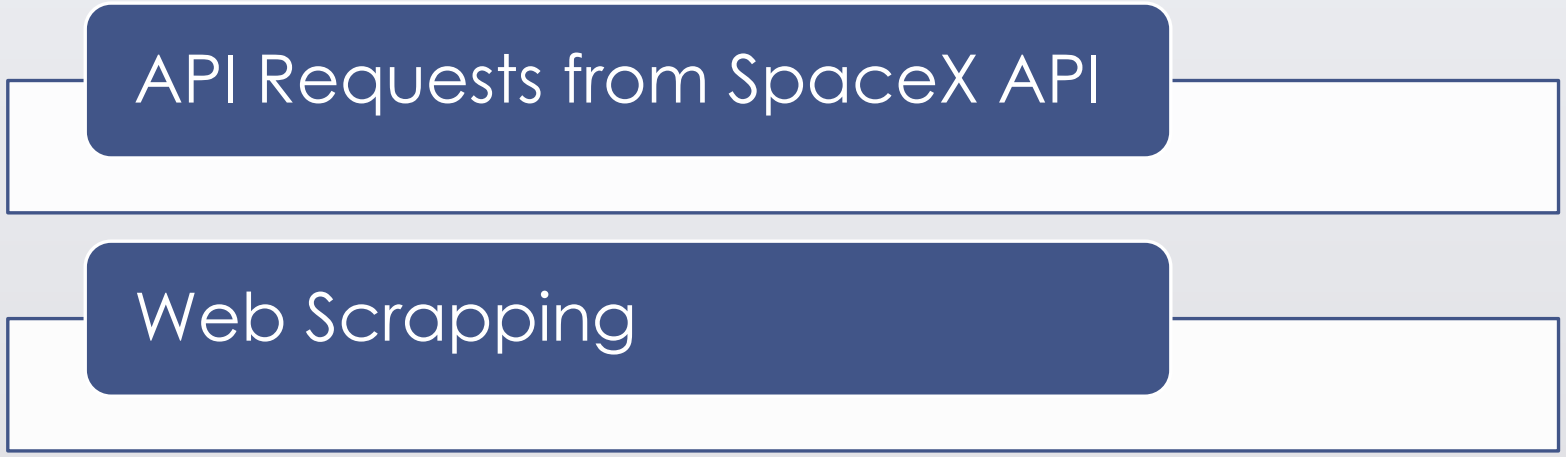
Methodology

Methodology

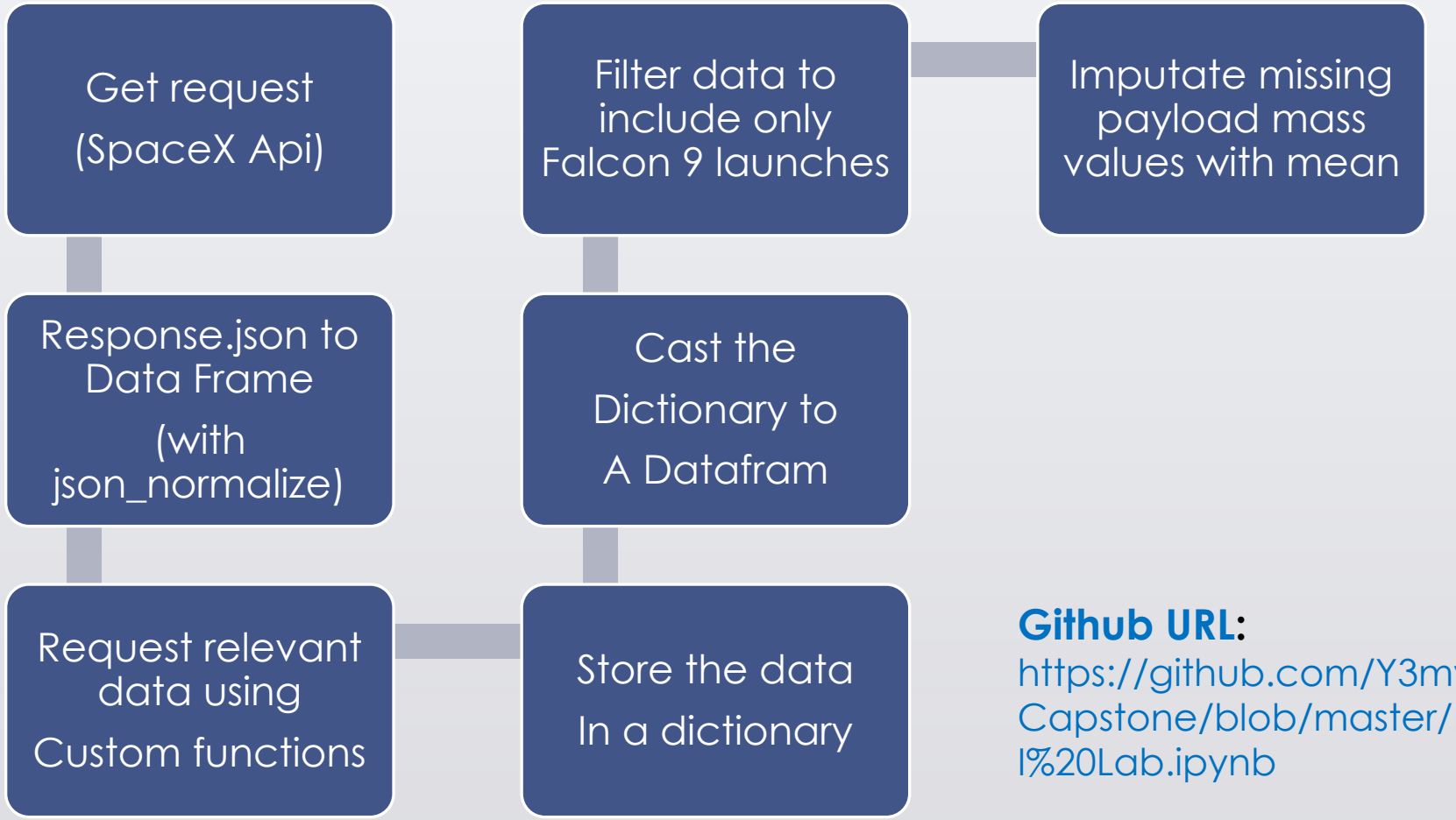
1. Data Collection
   - Making GET requests to the SpaceX REST API
   - Web Scraping
2. Data Wrangling
   - Using the .replace() method to remove NaN values
   - Using the .value_counts() method to determine the following:
     - Number of launches on each site
     - Number and occurrence of each orbit
     - Number and occurrence of mission outcome per orbit type
   - Creating a landing outcome label
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

- Data Collection involves the following processes:

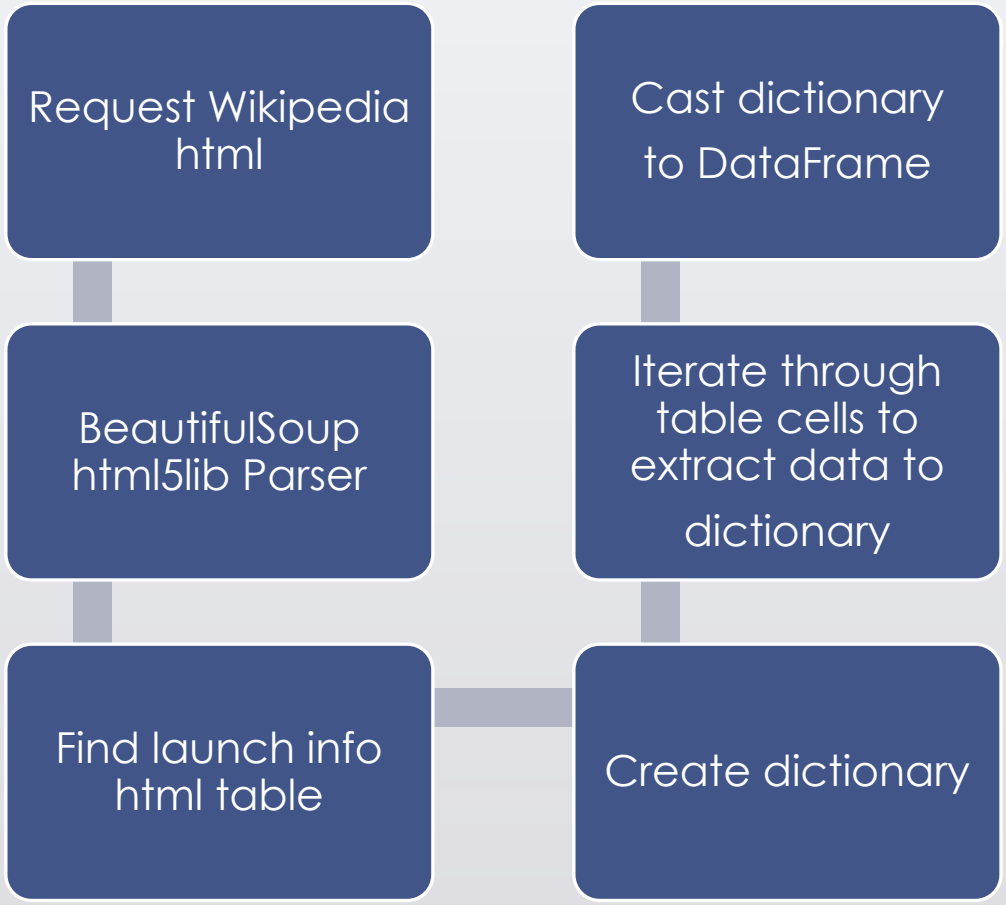


Data Collection – SpaceX API



Github URL:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/Data%20Collection%20API%20Lab.ipynb

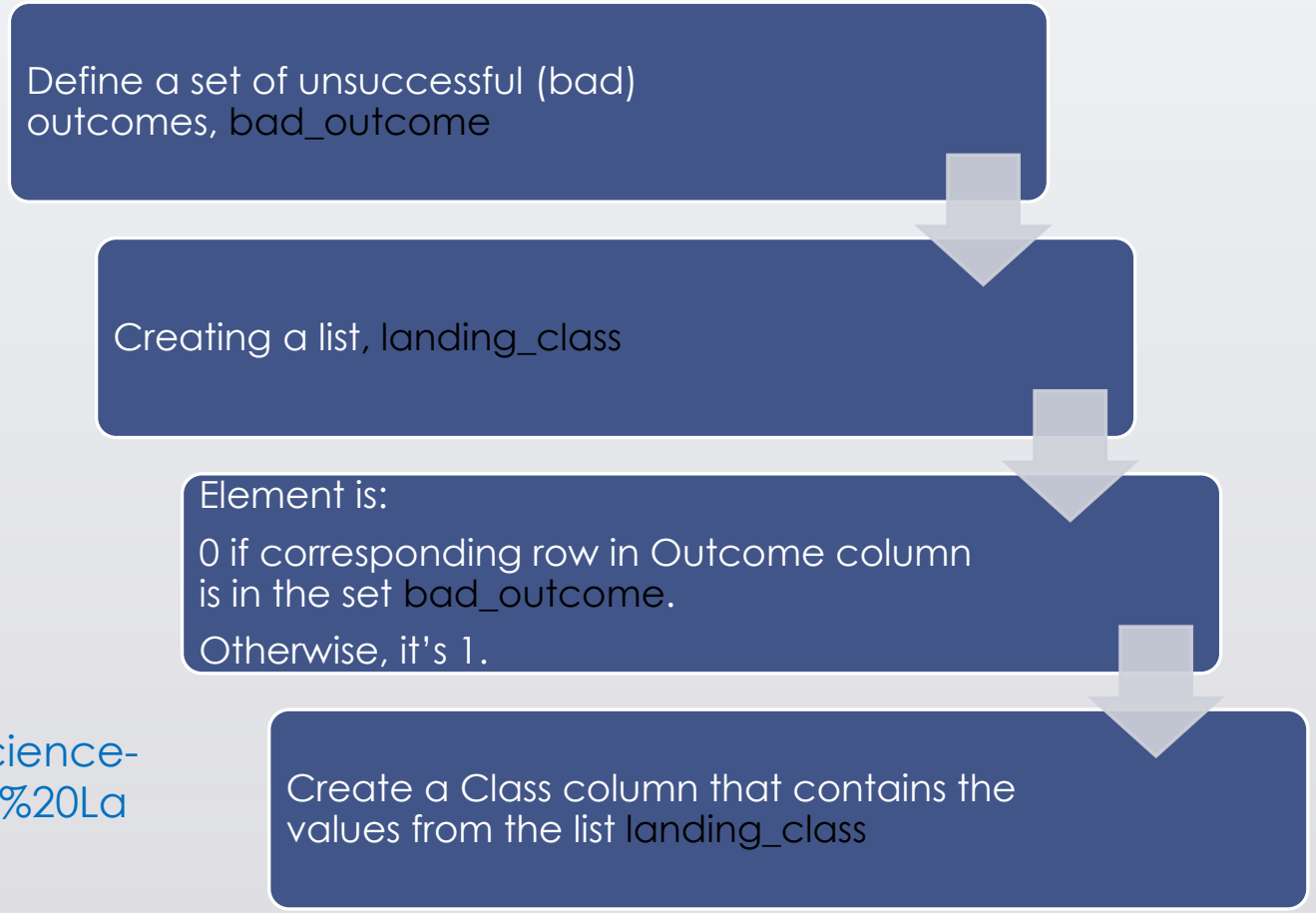
Data Collection - Scraping



Github URL:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/Data%20Collection%20Webscraping%20Lab.ipynb

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: Defining a set of unsuccessful (bad) outcomes, bad_outcome.



Github URL:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/Data%20Wrangling%20Lab.ipynb

EDA with Data Visualization

- Scatter plots, line charts and bar plots were used to visualize the relationship between:
  - Flight Number and Launch Site
  - Payload and Launch Site
  - Success rate of each orbit type
  - Flight Number and Orbit Type
  - Payload and Orbit Type
  - And to visualize the launch success yearly trend

Github URL:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/EDA%20with%20Visualization%20Lab.ipynb

EDA with SQL

- Loaded data set into IBM DB2 Database.
- Queried using SQL Python integration.
- Queries were made to get a better understanding of the dataset.
- Information about launch sites, payload mass, booster versions, mission outcomes, landing outcomes were queried.

GitHub url:

https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/EDA%20with%20SQL%20Lab.ipynb

Interactive Map with 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:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/Data%20Visualization%20with%20Folium.ipynb

Dashboard with Plotly Dash

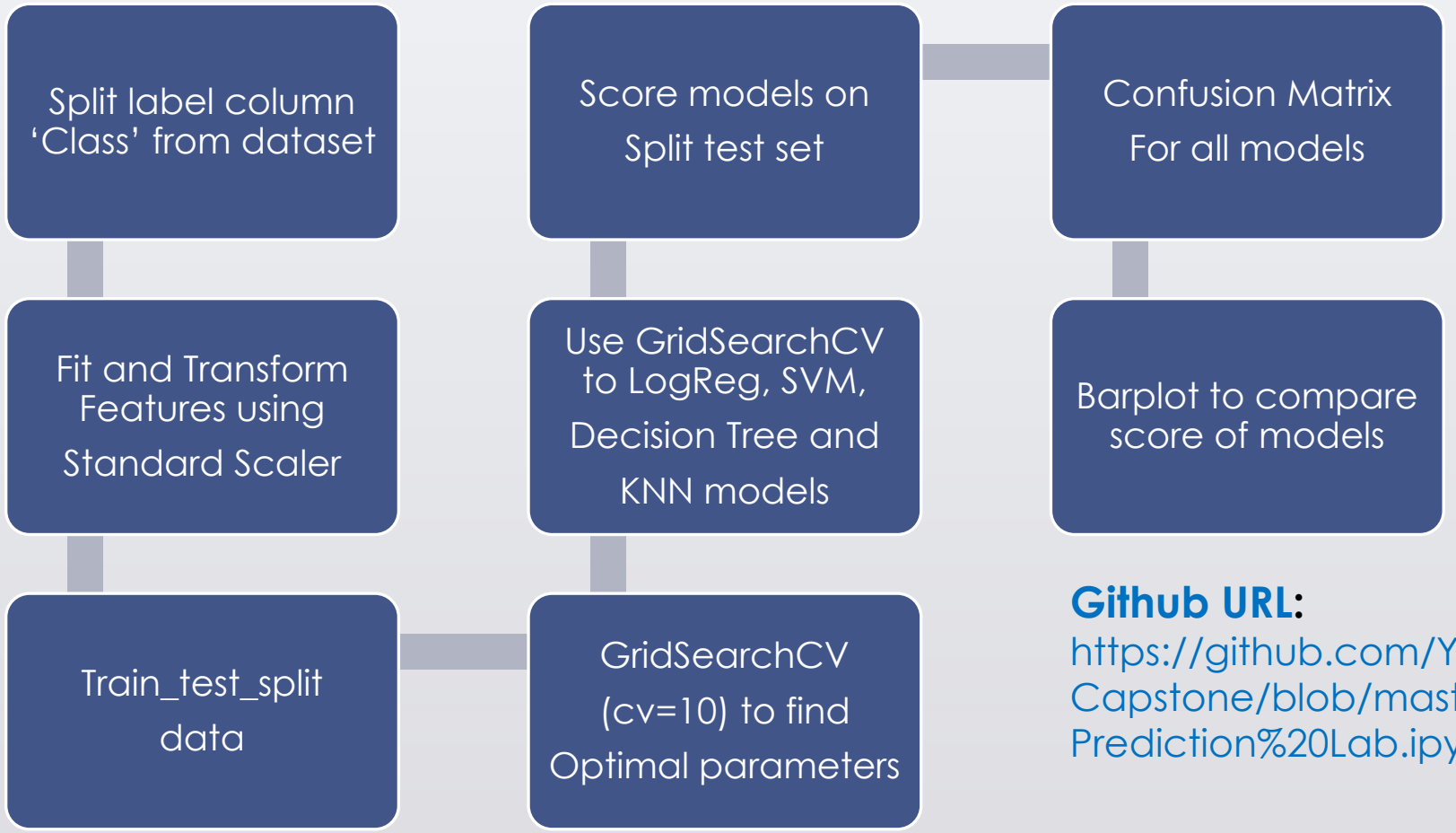
- The following plots were added to a Plotly Dash dashboard to have an interactive visualisation of the data:

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:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)



Github URL:
https://github.com/Y3myintoo/IBM-Data-Science-Capstone/blob/master/Machine%20Learning%20Prediction%20Lab.ipynb

Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

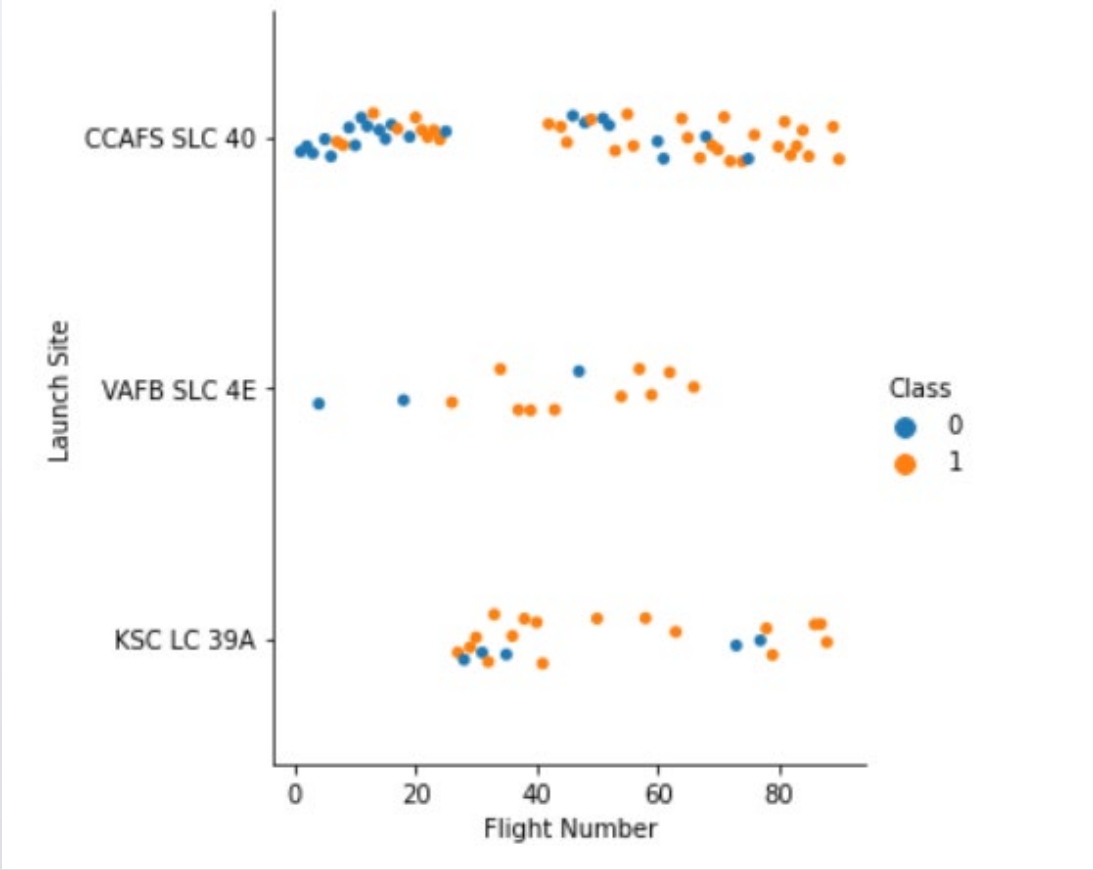
Section 2

Insights drawn from EDA

EDA with Visualization

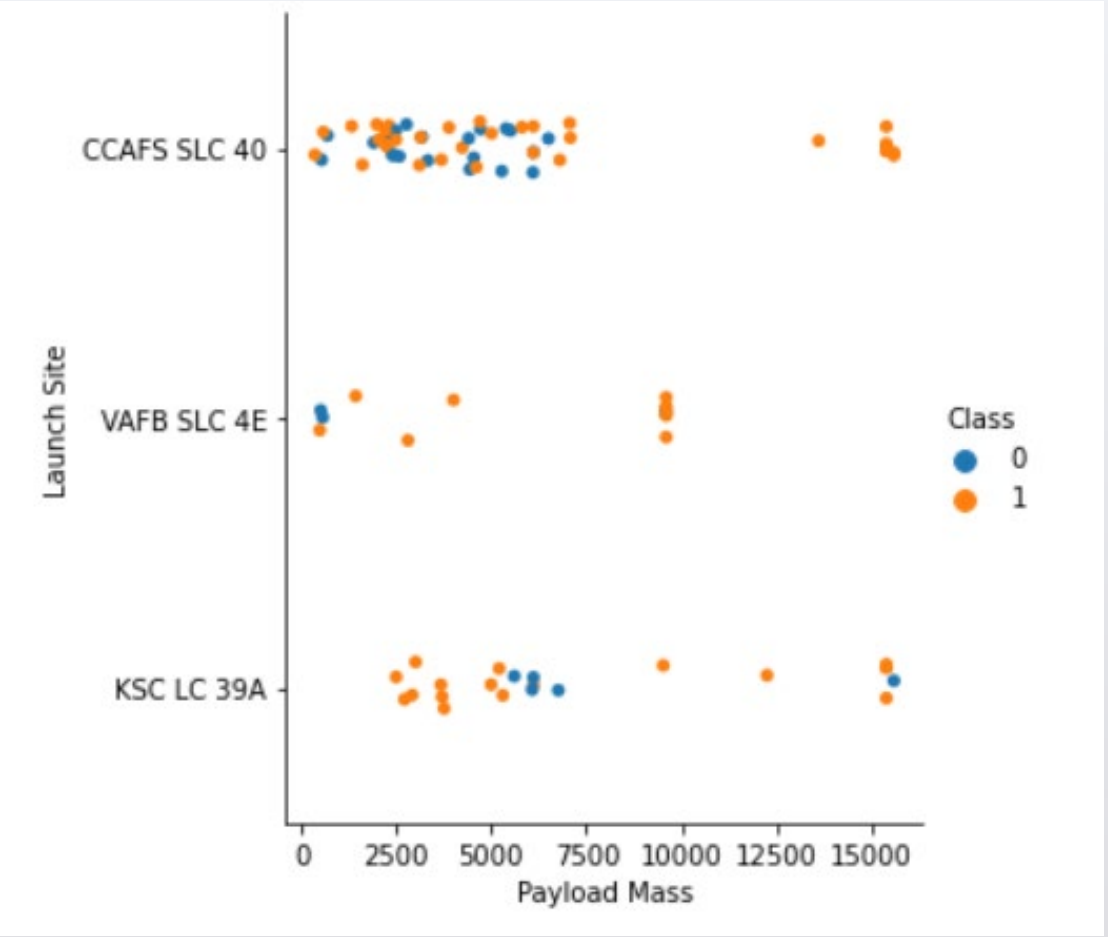
Flight Number vs. Launch Site

- With increase in flight number, the rate of success at a launch site increases
- Most of early flights(<30) were launched from CCAFS SLC 40 and were generally unsuccessful
- No early flights were launched from KSC LC 39A, so the launches from this site are more successful
- From above flight number of 30, there are significantly more successful landing outcomes



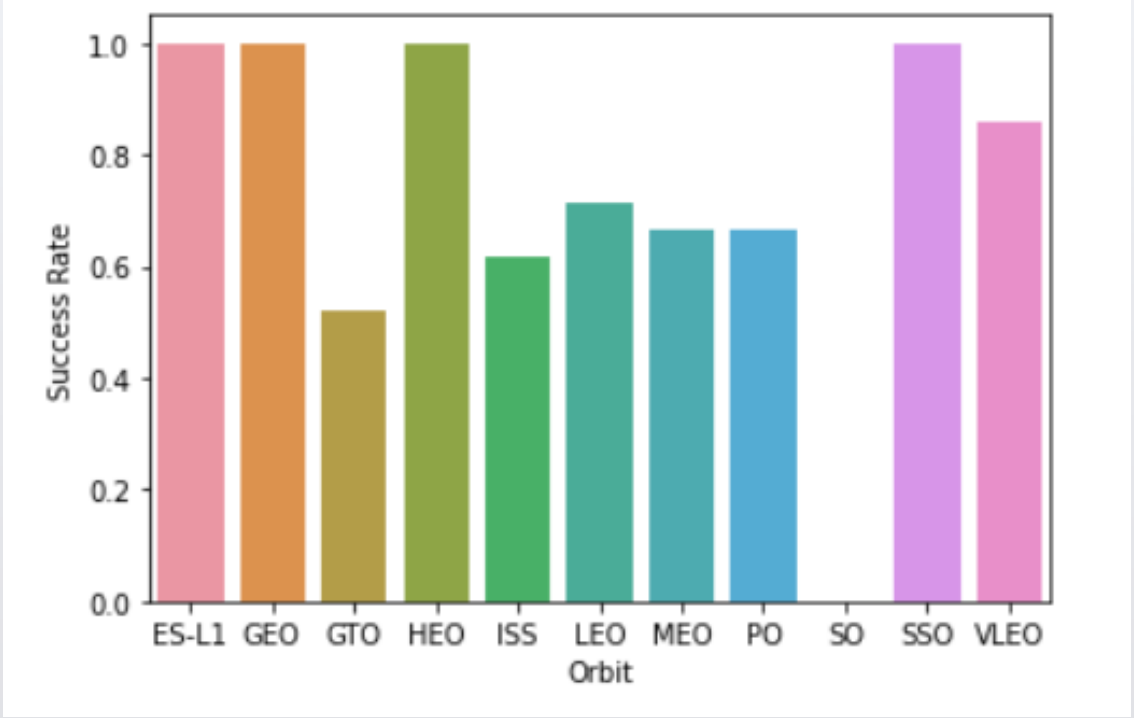
Payload vs. Launch Site

- Payload Mass appears to fall mainly between 0-7500 kg.
- All sites launched a variety of payload mass with most of the launches from VAFB SLC 4E having comparatively lighter payloads than other sites
- There is no clear correlation between success rate and payload mass for a given launch site



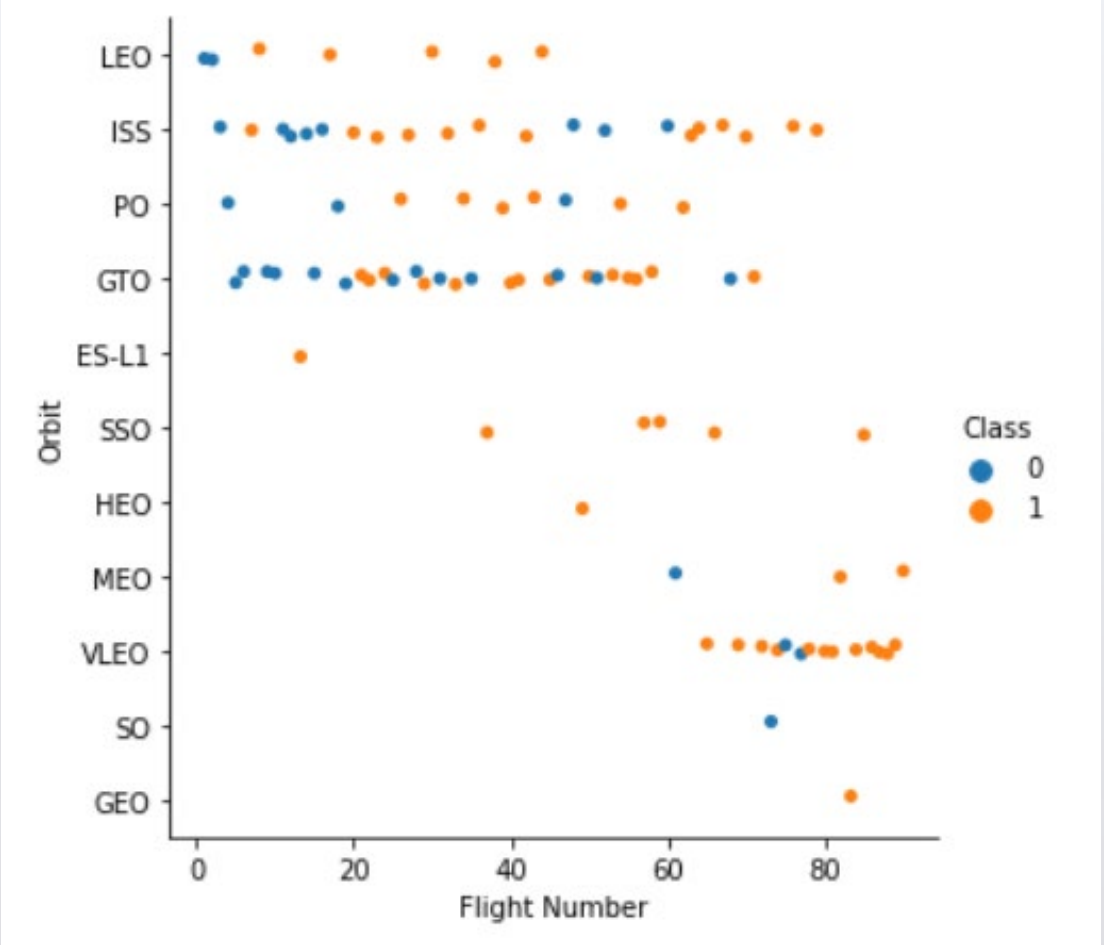
Success Rate vs. Orbit Type

- The bar chart of Success Rate vs Orbit Type shows that orbits with most success rate(100%) are:
  - ES-L1 (Earth-Sun First Lagrangian Point)
  - GEO (Geostationary Orbit)
  - HEO (High Earth Orbit)
  - SSO (Sun-synchronous Orbit)
- The orbit with lowest success rate(0%) is SO(Heliocentric Orbit).



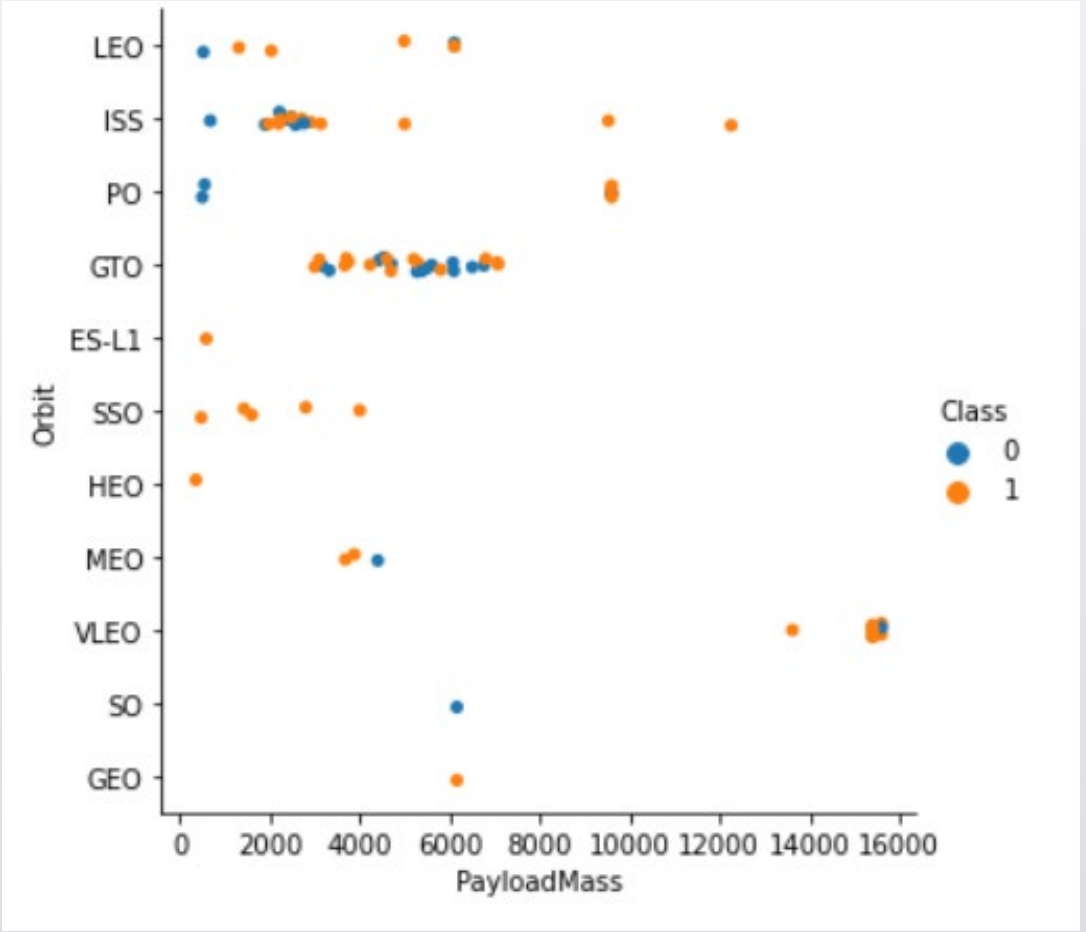
Flight Number vs. Orbit Type

- The 100% success rate of ES-L1, HEO,GEO orbits can be explained by having only one flight into respective orbits
- The success rate of SSO(100%) with 5 flights to the orbit is more impressive
- Generally, as flight number increases, the success rate also increases. This is more obvious for LEO where unsuccessful landings only occur for low flight numbers.



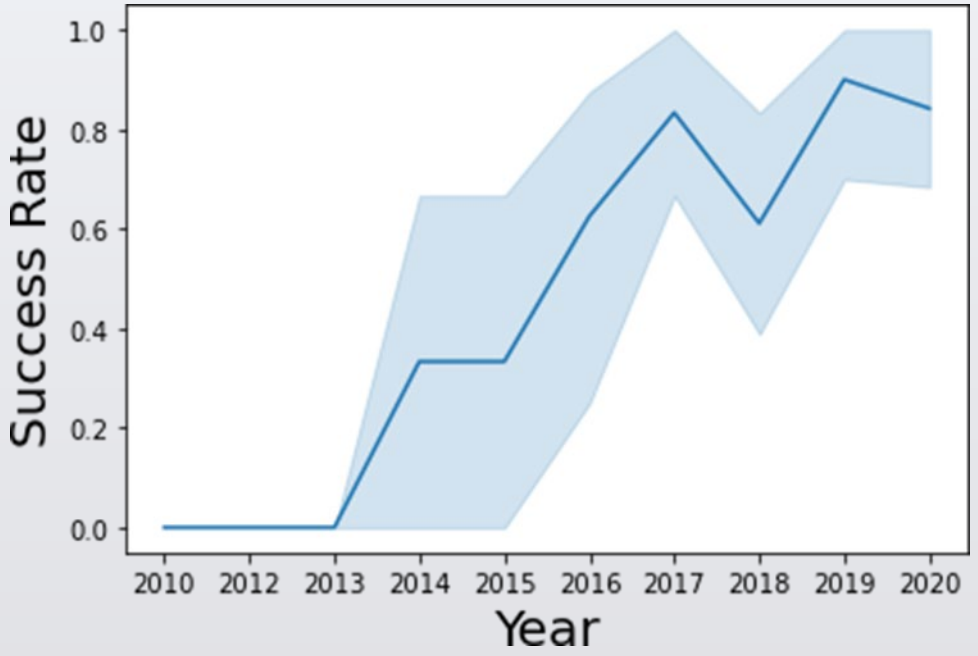
Payload vs. Orbit Type

- Payload mass seems to correlate with Orbit Type.
  - Orbits like LEO, GTO, SSO have lower payload mass.
  - Some orbits like VLEO have higher payload mass.
- As for GTO, there is no clear correlation between payload mass and orbit type.



Launch Success Yearly Trend

- Success rate generally increases over time since 2013 with a slight dip in 2018.
- Success rate in recent years is around 80%



95% confidence interval
(light blue shading)

EDA with SQL

All Launch Site Names

- Find the names of the unique launch sites.

```
%%sql
SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL;
```

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

- The SQL keyword ‘Distinct’ returns unique values from a column.
- CCAFS LC-40 is a previously used name. So, there are probably only 3 unique launch sites, which are:
  1. CCAFS SLC-40
  2. KSC LC-39A
  3. VAFB SLC-4E

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`.

```
%sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu
Done.
```

launch_site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

- The keyword ‘LIMIT 5’ fetches only 5 records.
- ‘LIKE’ is used to retrieve launch sites that begin with ‘CCA’.

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)';
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lq
Done.
```

1
45596

The ‘SUM’ keyword was used to get total values of Payload Mass and then result was filtered to include only ‘NASA’.

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS FROM SPACEXTBL\
WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tg
Done.
```

average_payload_mass
2928

We calculate the average payload mass with ‘AVG’. Then, we filter the booster version to include only F9 v1.1.

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

```
%%sql
SELECT MIN(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success (ground pad)';
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lq
Done.
```

1
2015-12-22

'Min' keyword is used to get the date for first successful landing outcome on ground pad.
Here we can see that the first successful landing on a ground pad was in 2015.

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

```
%%sql
SELECT DISTINCT(BOOSTER_VERSION) FROM SPACEXTBL WHERE
(LANDING_OUTCOME = 'Success (drone ship)') AND (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000) ;
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databa
Done.
```

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

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 BETWEEN keyword allows for $4000 < x < 6000$ values to be selected

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

```
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdoma
Done.
```

mission_outcome	total
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

This query returns a count of each mission outcome.

SpaceX appears to achieve its mission outcome nearly 99% of the time.

This means that most of the landing failures are intended.

Interestingly, one launch has an unclear payload status and unfortunately one failed in flight.

Boosters Carried Maximum Payload

- List 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);
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1og
Done.
```

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

This query returns the booster versions that carried the highest payload mass of 15600 kg.

These booster versions are very similar and all are of the F9 B5 B10xx.x variety.

This likely indicates payload mass correlates with the booster version that is used.

2015 Launch Records

- List the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%%sql
SELECT LANDING_OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE
(LANDING_OUTCOME = 'Failure (drone ship)') AND (YEAR(DATE) = '2015');
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tg
Done.
```

landing_outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- This query list the failed landing outcomes in drone ship, their booster versions and launch site names in year 2015.
- There are 2 such cases.

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

- 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

```
%sql SELECT LANDING_OUTCOME, COUNT(LANDING_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL \
    WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
    GROUP BY LANDING_OUTCOME \
    ORDER BY TOTAL_NUMBER DESC;
```

```
 * ibm_db_sa://cvw90686:***@ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.da
Done.
```

landing_outcome	total_number
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

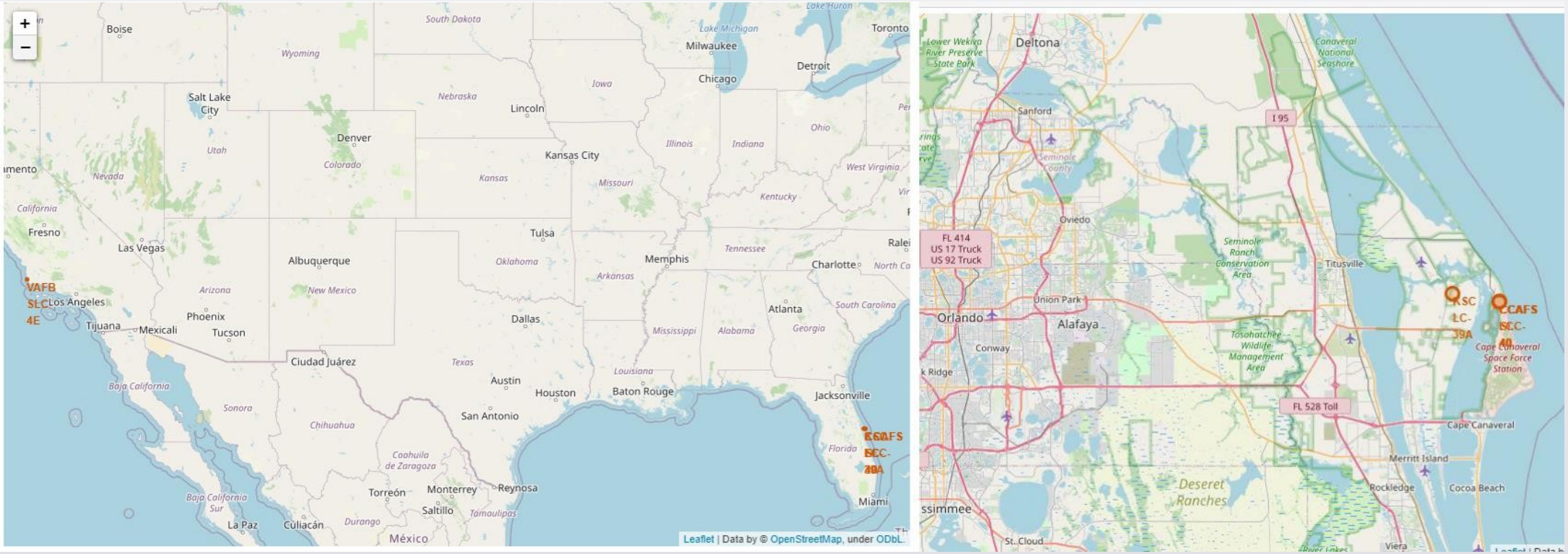
Between 2010-06-04 and 2017-03-20,

- Most of the landings are not attempted or on to the drone ship.
- The landings on ground pad, ocean and with parachute are lesser.

Section 3

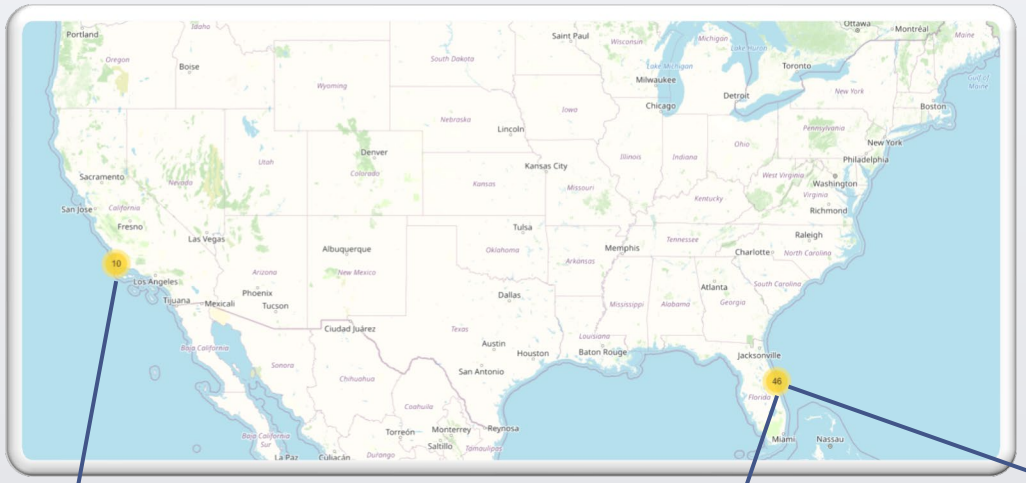
Launch Sites Proximities Analysis

Launch Site Locations



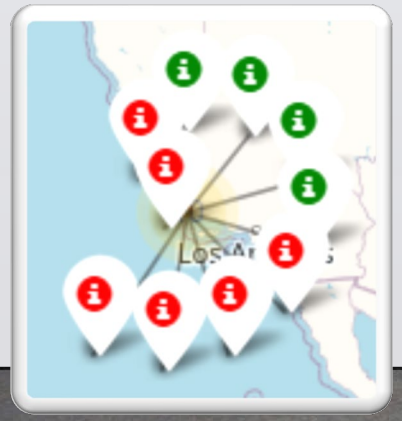
The left map shows all launch sites in relation to US map. The right map shows 2 Florida launch sites since they are very close to one another. All launch sites are located near the ocean.

Success/Failed Launches for each site

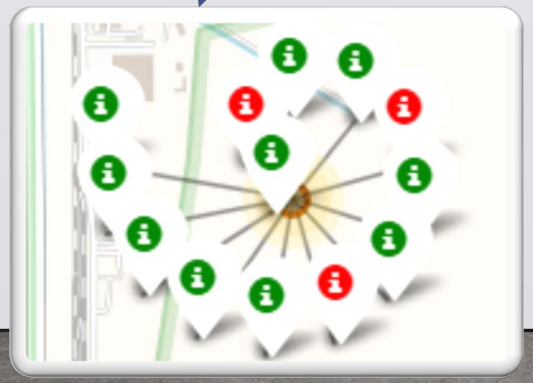


- Launches have been grouped into clusters, and annotated with green icons for successful launches, and red icons for failed launches.

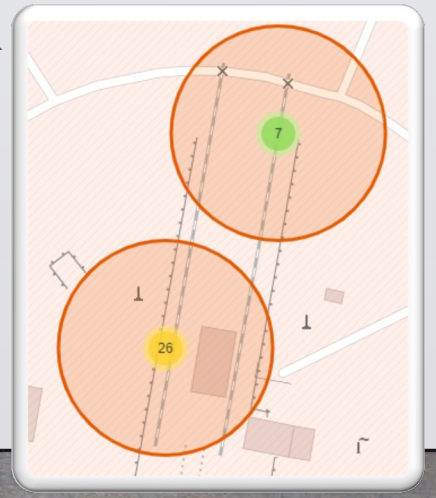
VAFB SLC-4E



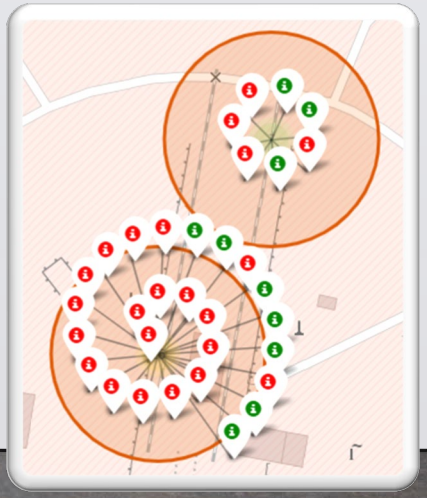
KSC LC-39A



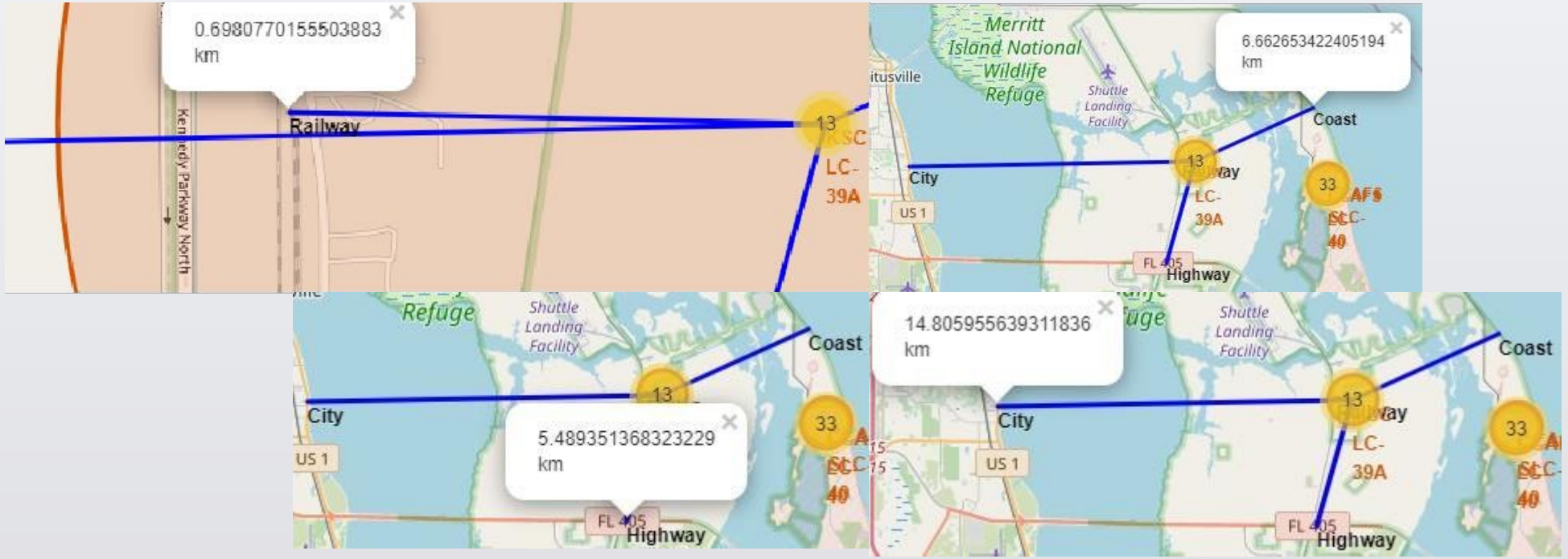
CCAFS SLC-40 and CCAFS LC-40



=



Key Location Proximities

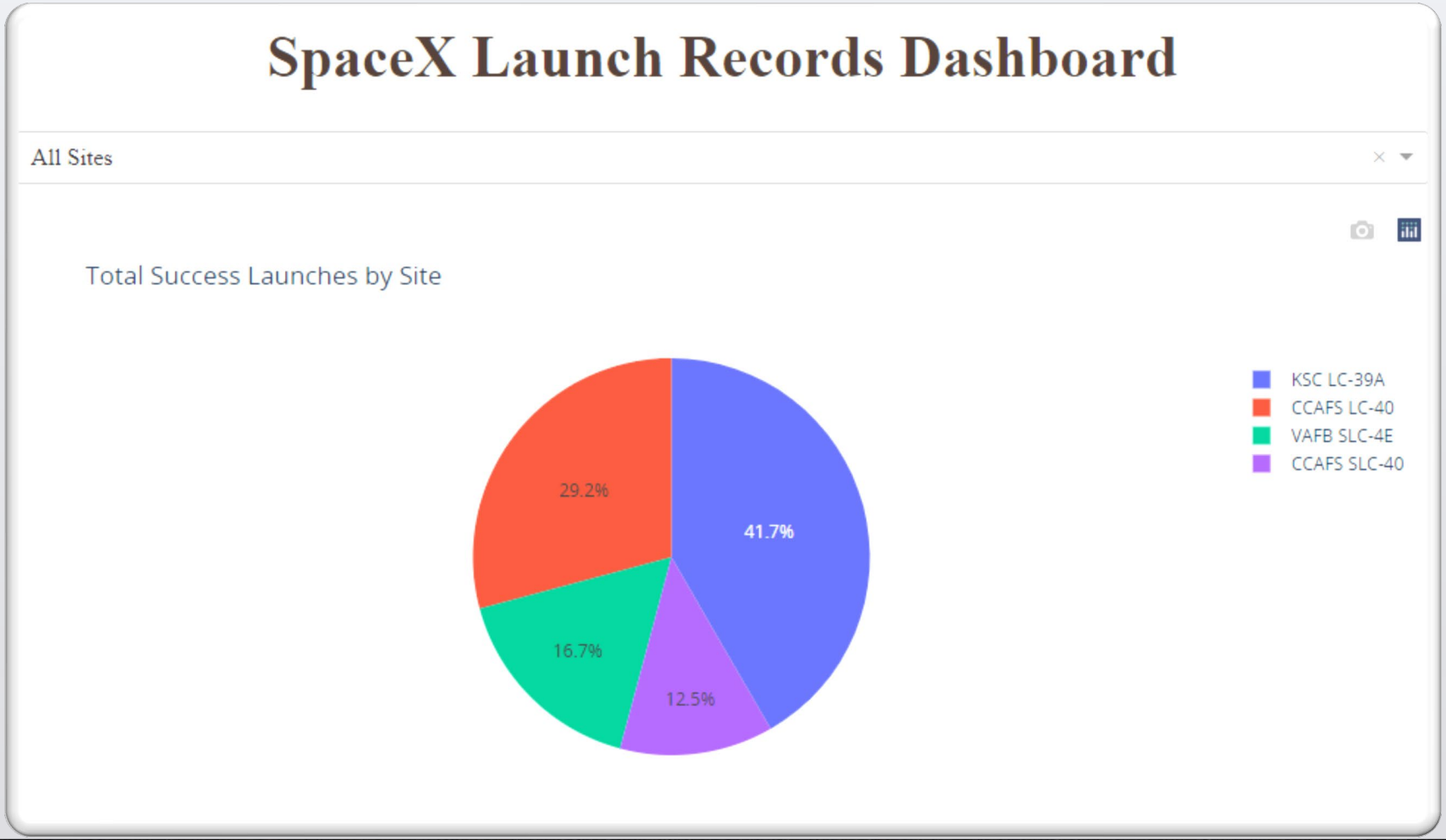


Using KSC LC-39A as an example, launch sites are very close to railways for large part and supply transportation. Launch sites are close to highways for human and supply transport. Launch sites are also close to coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.

Section 4

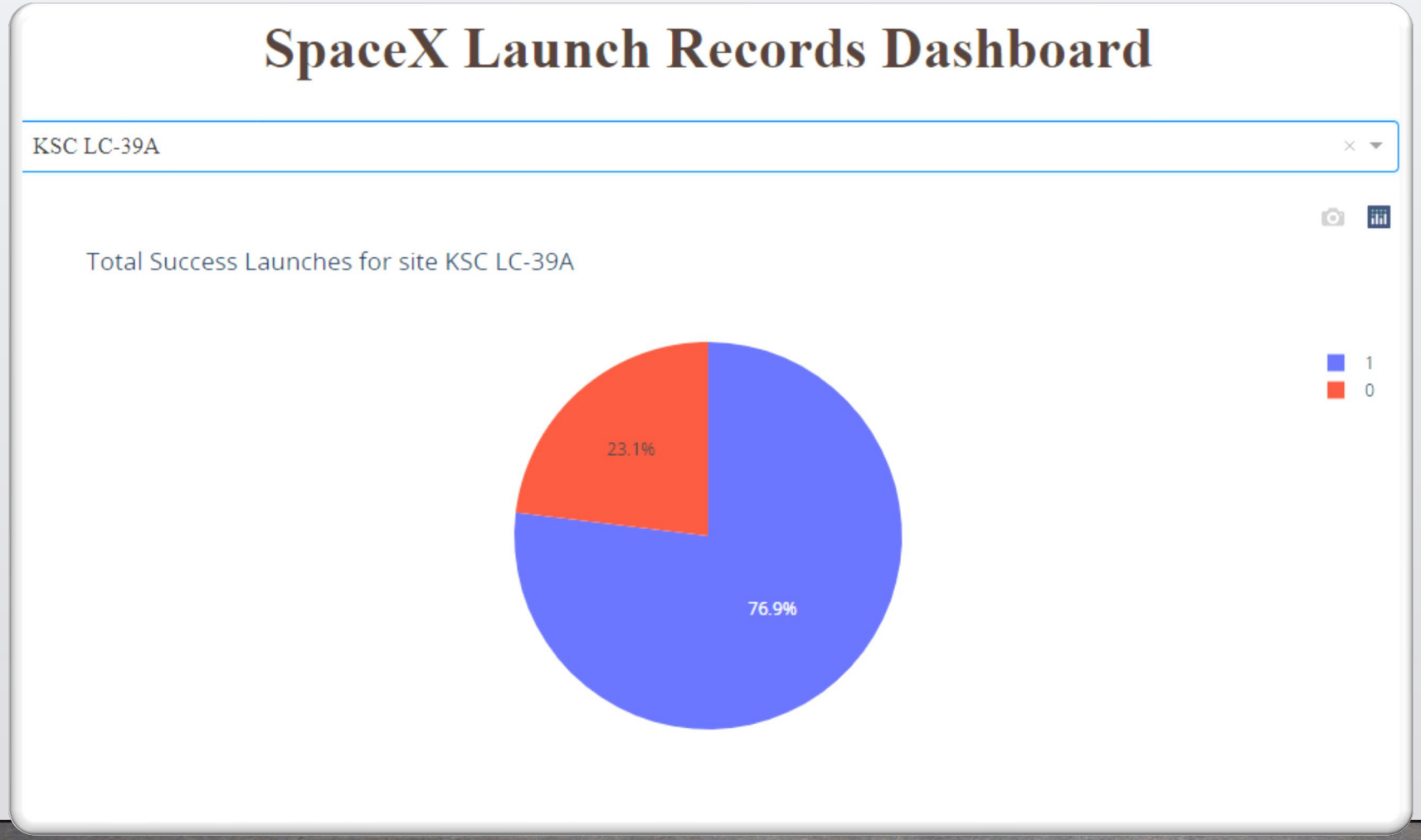
Build a Dashboard with Plotly Dash

Launch Success Count for All Sites



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

Launch Site with Highest Launch Success Ratio



The launch site KSC LC-39 A also had the highest rate of successful launches, with a 76.9% success rate.

Payload vs Launch Outcome for All Sites

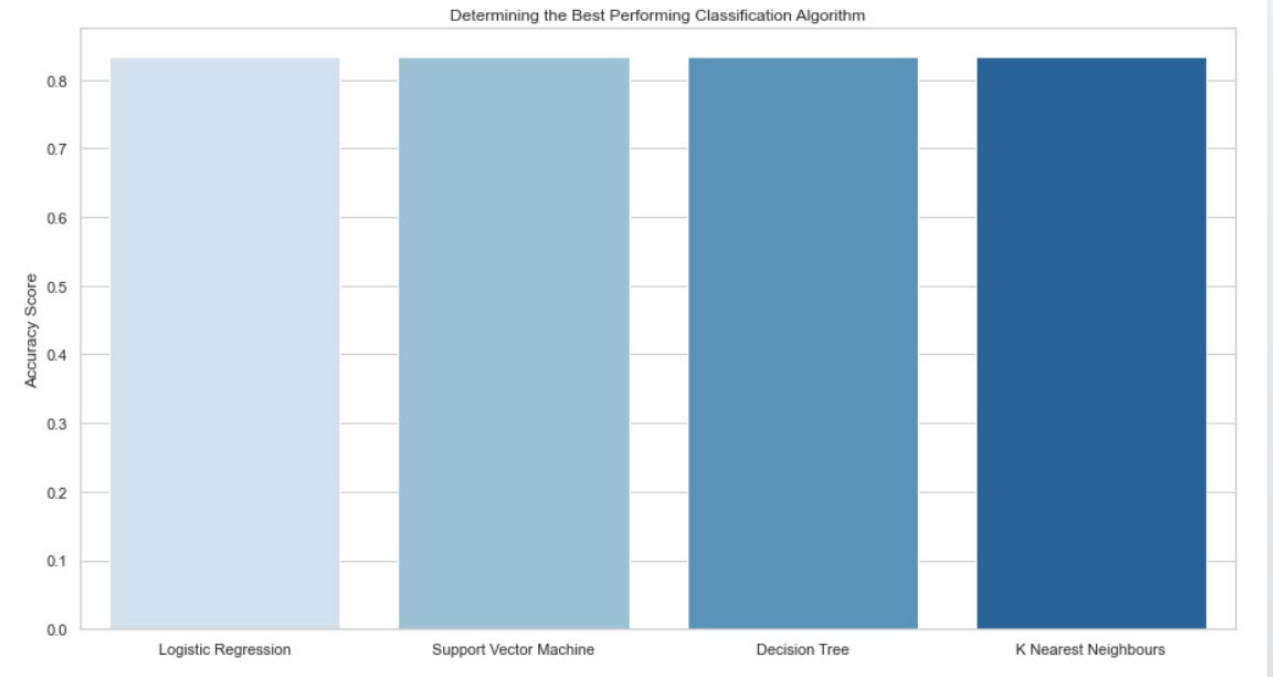
- Plotting the payload vs launch outcome for all sites shows a gap around 4000 kg, so it makes sense to split the data into 2 ranges:
  - 0 – 4000 kg (low payloads)
  - 4000 – 10000 kg (massive payloads)
- From these 2 plots, it can be shown that the success for massive payloads is lower than that for low payloads.
- It is also worth noting that some booster types (v1.0 and B5) have not been launched with massive payloads.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- All models had virtually the same accuracy on the test set at 83.33% accuracy.
- It should be noted that test size is small at only sample size of 18.
- This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.
- We likely need more data to determine the best model.



Confusion Matrix

- Since all models performed the same for the test set, the confusion matrix is the same across all models.
- The models predicted 12 successful landings when the true label was successful landing.
- The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.
- The models predicted 3 successful landings when the true label was unsuccessful landings (false positives).
- Our models over predict successful landings.

Correct predictions are on a diagonal from top left to bottom right.

Conclusions

- Our task: to develop a machine learning model for Space Y who wants to bid against SpaceX
- The goal of model is to predict when Stage 1 will successfully land to save ~$100 million USD
- Used data from a public SpaceX API and web scraping SpaceX Wikipedia page
- Created data labels and stored data into a DB2 SQL database
- Created a dashboard for visualization
- We created a machine learning model with an accuracy of 83%
- We can use this model to predict with relatively high accuracy whether a launch will have a successful Stage 1 landing before launch to determine whether the launch should be made or not.
- If possible more data should be collected to better determine the best machine learning model and improve accuracy.

Thank you!