IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Aaron King

September 14, 2023

Outline

Executive Summary

- Collected data from public SpaceX API and SpaceX Wikipedia page. Created labels column ‘class’ which classifies successful landings. Explored data using SQL, visualization, folium maps, and dashboards. Gathered relevant columns to be used as features. Changed all categorical variables to binary using one hot encoding. Standardized data and used GridSearchCV to find best parameters for machine learning models. Visualize accuracy score of all models.

- Four machine learning models were produced: Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors. All produced similar results with accuracy rate of about 83.33%. All models over predicted successful landings. More data is needed for better model determination and accuracy.

Introduction

- **Project Background**
  - Space X has best pricing ($62 million vs. $165 million USD that has been the bottom price for sending property into space.)
  - Largely due to ability to recover part of rocket (Stage 1)
  - Space Y wants to compete with Space X
- **Project Problem**
  - Space Y tasks us to train a machine learning model to predict successful Stage 1 recovery

SpaceX's Falcon 9 rocket launch – Popular Mechanics

Section 1

Methodology

Methodology

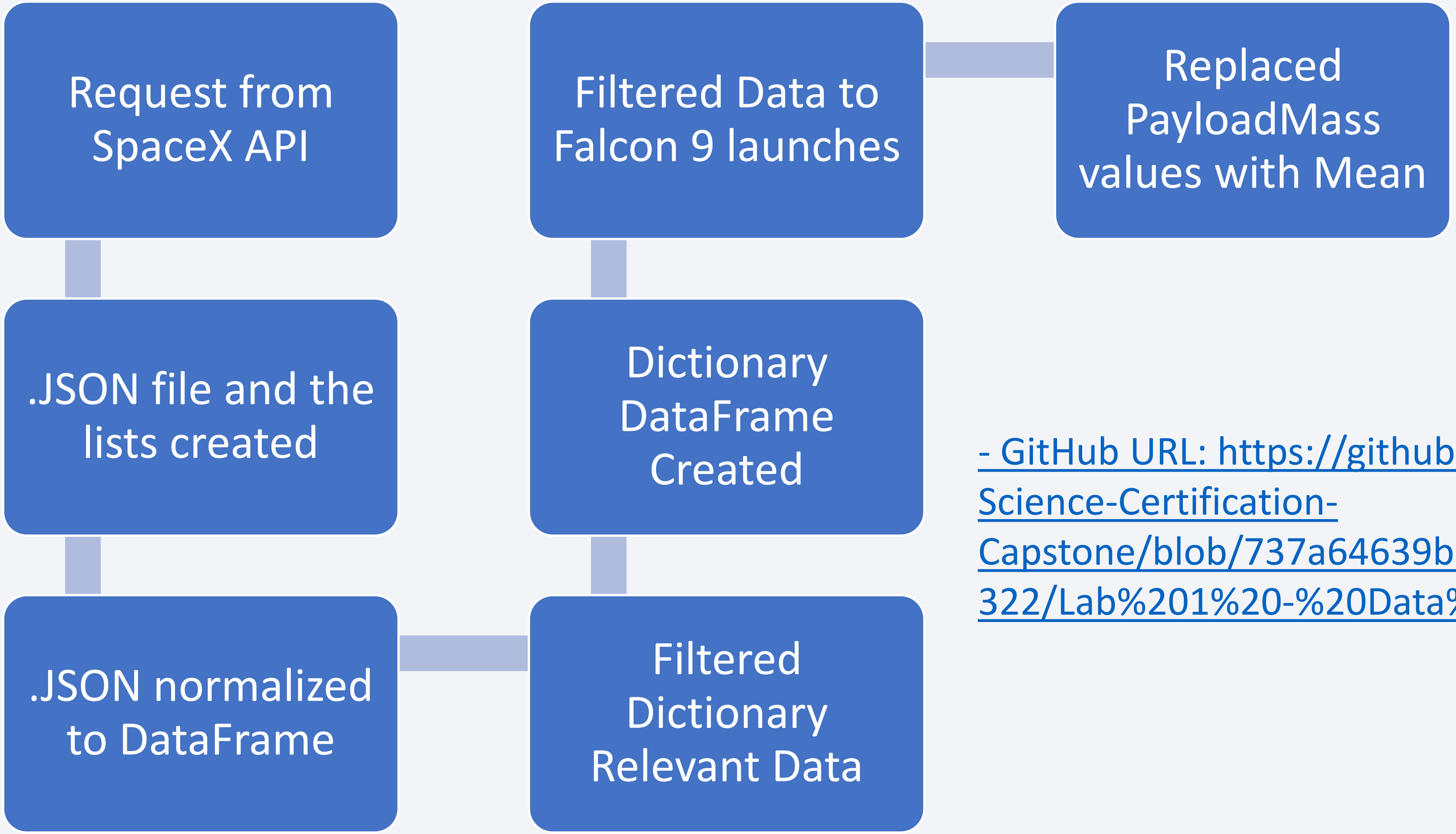
Executive Summary

- Data collection methodology:
  - Combined data from SpaceX public API and SpaceX Wikipedia page
- Perform data wrangling
  - Classifying true landings as successful and unsuccessful otherwise
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Tuned models using GridSearchCV

Data Collection

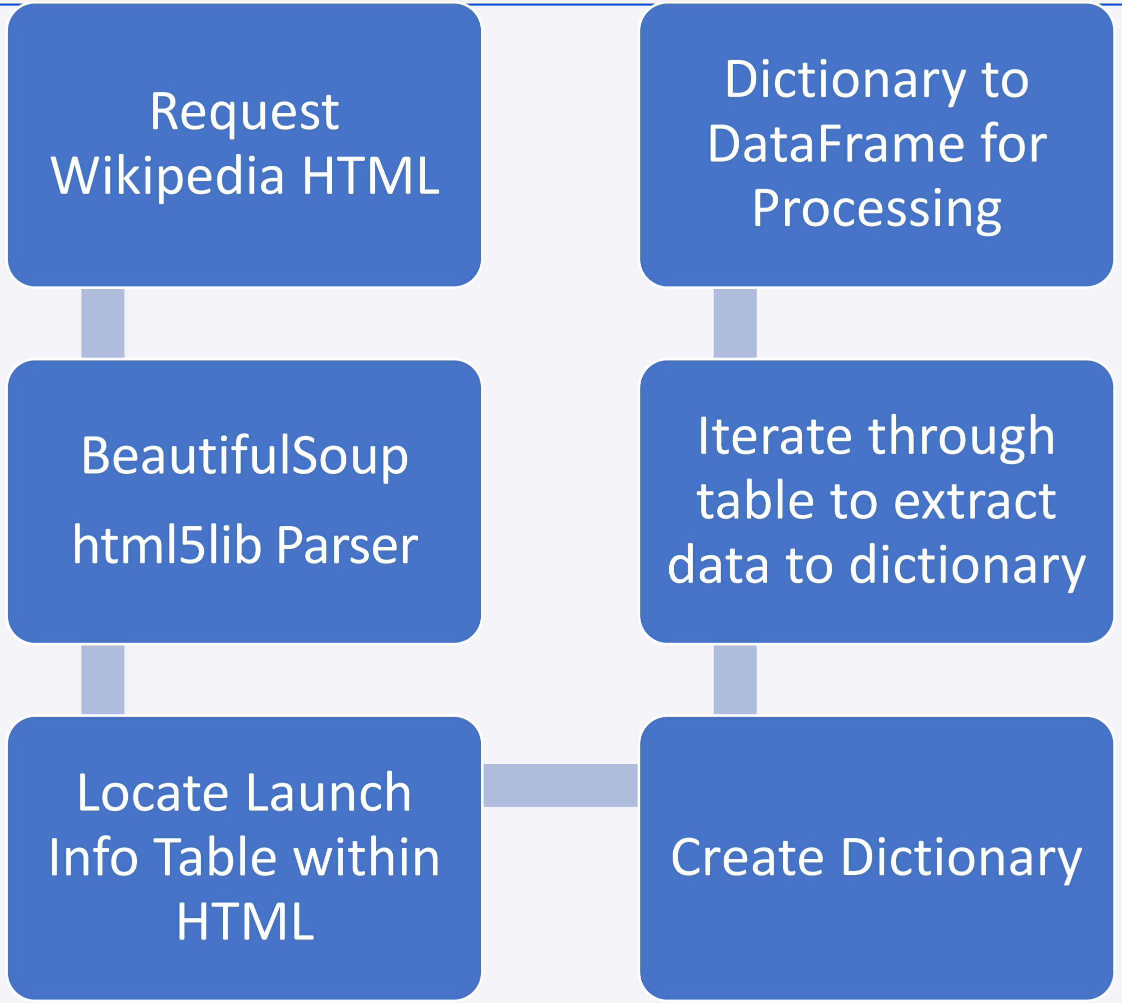
- Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.
- Space X API Data Columns:
  - FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude
- Wikipedia Webscrape Data Columns:
  - Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time
- The following slides will cover the flowchart of data collection from API and the process for webscraping.

Data Collection – SpaceX API



[- GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/737a64639bd997b816d3920e24b790664017f322/Lab%201%20-%20Data%20Collection.ipynb](https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/737a64639bd997b816d3920e24b790664017f322/Lab%201%20-%20Data%20Collection.ipynb)

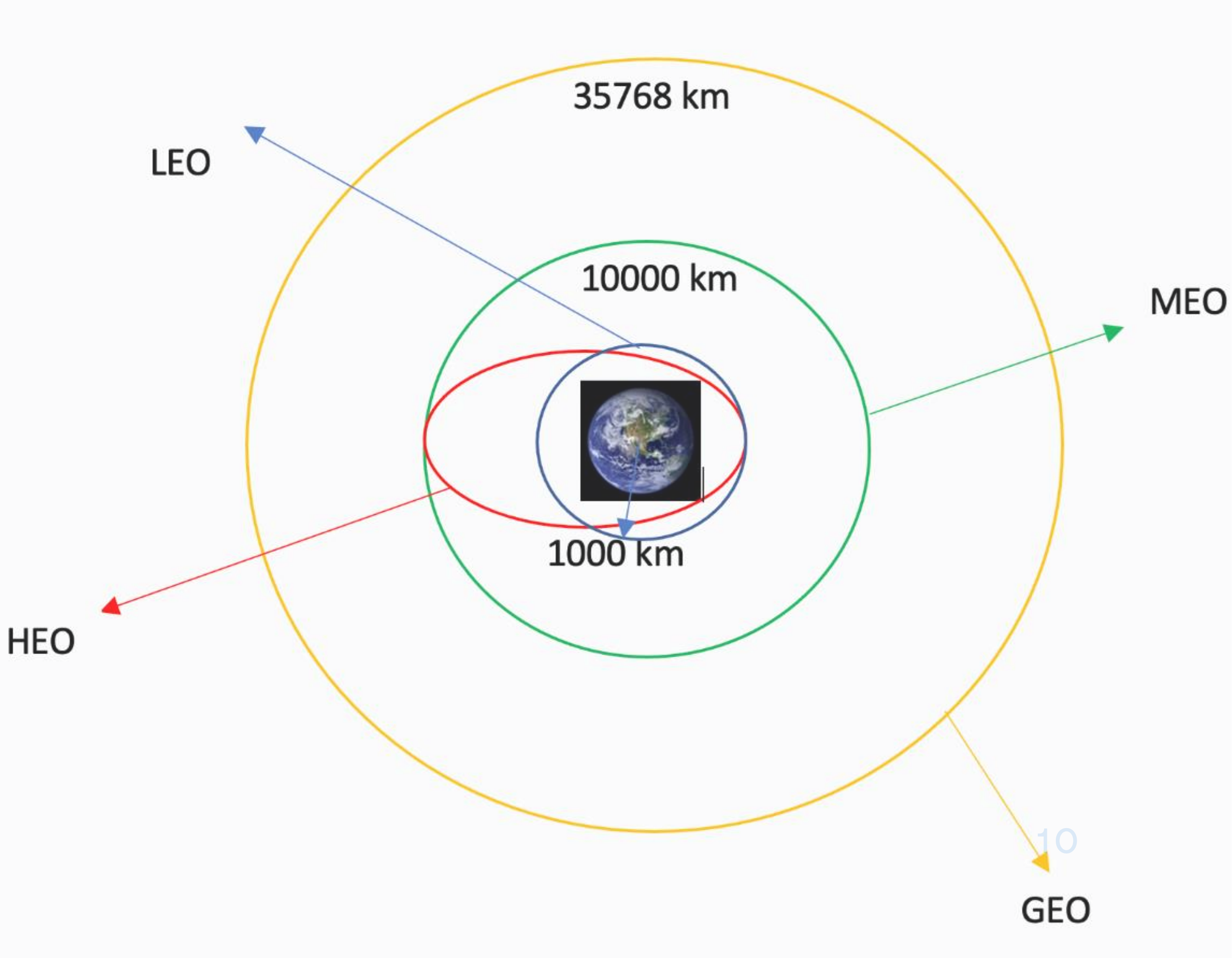
Data Collection - Scraping



- GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/219da2bd5d59c33814a67e13d68e415af630eb5e/Lab%201%20-%20Webscraping.ipynb

Data Wrangling

- Performed exploratory data analysis, determining the training labels for the data sets.
- Calculated the number of launches at each site, the number and occurrence of each orbit and launch.
- Created landing outcome variables from the outcome data column and exported those results to csv file format for later use.
- GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/a645279fe7da390f0bc7abccda18495178e015a3/Lab%202%20-%20Data%20Wrangling.ipynb



EDA with Data Visualization

- Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.
- Plots Used:
- Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend
- Scatter plots, line charts, and bar plots were used to compare relationships between variables to
- decide if a relationship exists so that they could be used in training the machine learning model
- GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/6f3b003b8bbc807304d4d01655910bcd1e50a3aa/Lab%204%20-%20EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

- Loaded SpaceX dataset into PostgreSQL database using jupyter notebook.
- Analyzed using SQL queries to get insight into the data. The following are some of the queries:
  - Display the names of the unique launch sites in the space mission
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first succesful landing outcome in ground pad was acheived.
- [GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/6f3b003b8bbc807304d4d01655910bcd1e50a3aa/Lab%203%20-%20SQL%20EDA.ipynb](https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/6f3b003b8bbc807304d4d01655910bcd1e50a3aa/Lab%203%20-%20SQL%20EDA.ipynb)

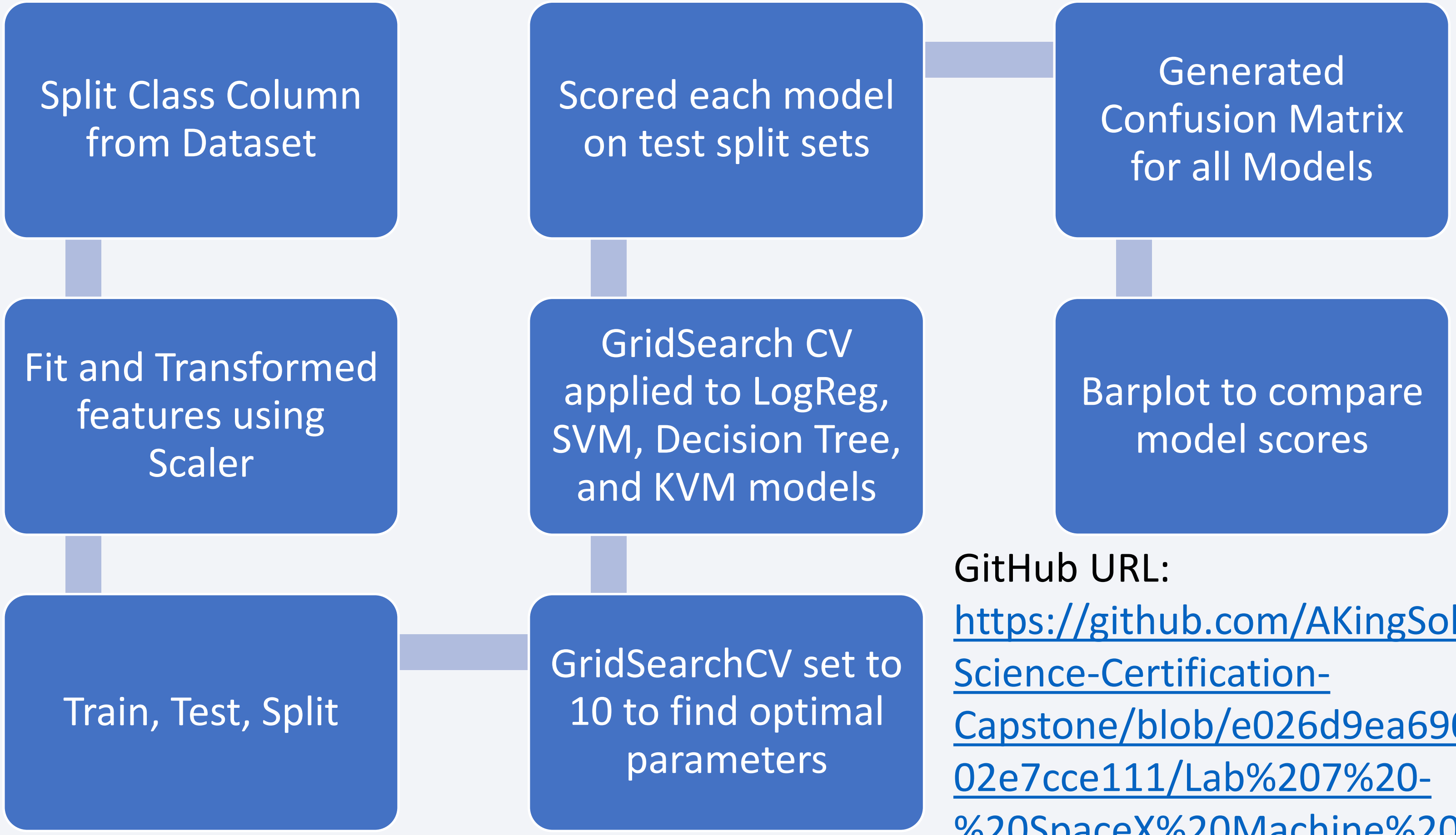
Build an Interactive Map with Folium

- Each launch site was marked with a map marker, circles for indicating site location, and then lines were incorporated to pinpoint success and failure of launches for each launch site using the folium map.
- By assigning success and failure rates to each launch site, color labeled marker clusters could be assigned to each map launch site to identify the success rate of each launch site.
- Finally, distances were calculated relative to launch sites to judge objects within its proximity.
  - Distance from railways, highways, and coastlines.
  - Distance from major cities.
- GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/d4436c947916099e27eb3fdda5b84d4672811e1f/Lab%205%20-%20Launch%20Site%20Analysis%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

- Using Plotly Dash, a dashboard was generated to be an interactive design capable of allowing a user to observe different pieces of information on Launch Sites for Space X.
- Pie charts were generated showing total launches from each site and giving a comparison to the total launches of all sites.
- And finally, a scatter plot was generated showing the relationship between Outcome and Payload Mass for different booster versions. A slide adjustment was added to allow a user to observe the changes between different payload masses.
- GitHub URL: https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/1f31c05903f718a027ede2f5d0b1fa06bb2d30c2/SpaceX%20Dashboard%20App.py

Predictive Analysis (Classification)



GitHub URL:
https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone/blob/e026d9ea690c5a6f07f409aff84ee602e7cce111/Lab%207%20-%20SpaceX%20Machine%20Learning%20Predictions.jupyterlite.ipynb

Results

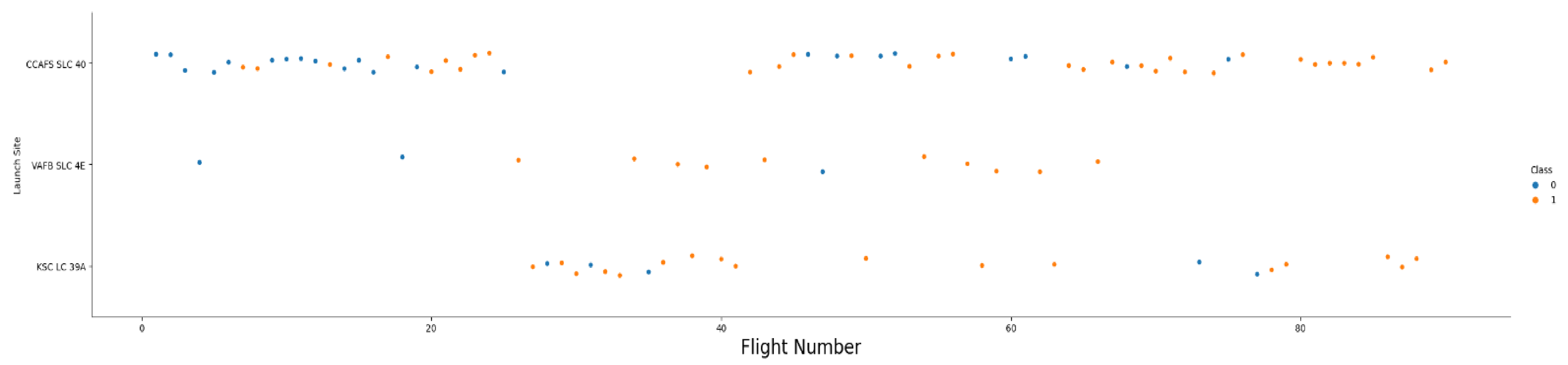
The following slides will show the following:

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

Section 2

Insights drawn from EDA

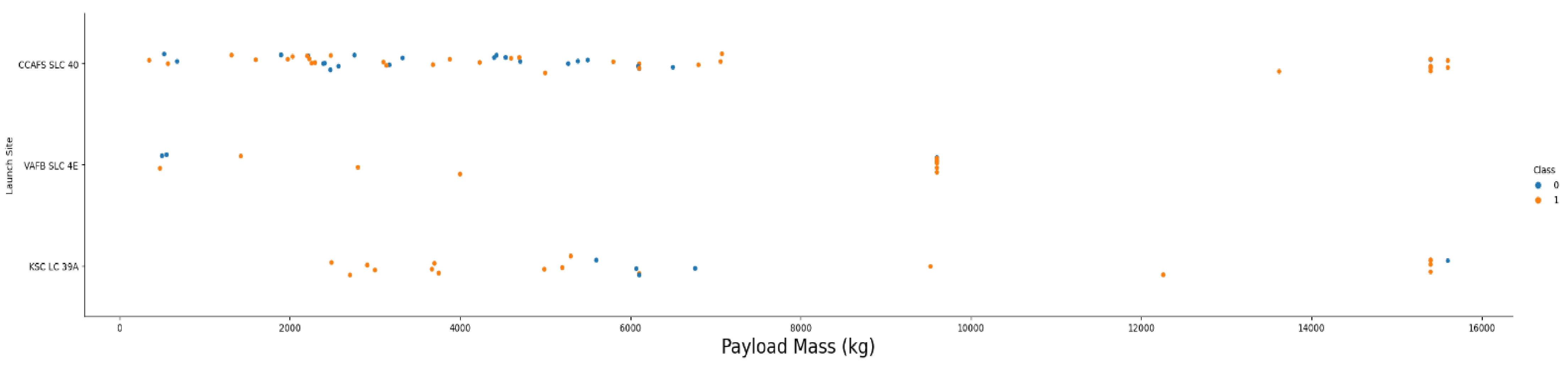
Flight Number vs. Launch Site



Blue indicates unsuccessful launch while orange indicates successful launches.

The graph indicates an increase in success rate over time. As the number of launches increases, the more those launches were written as a success. The graph also seems to indicate that CCAFS SLC 40 launch site is the favored launch site by volume as the majority of launches are performed there.

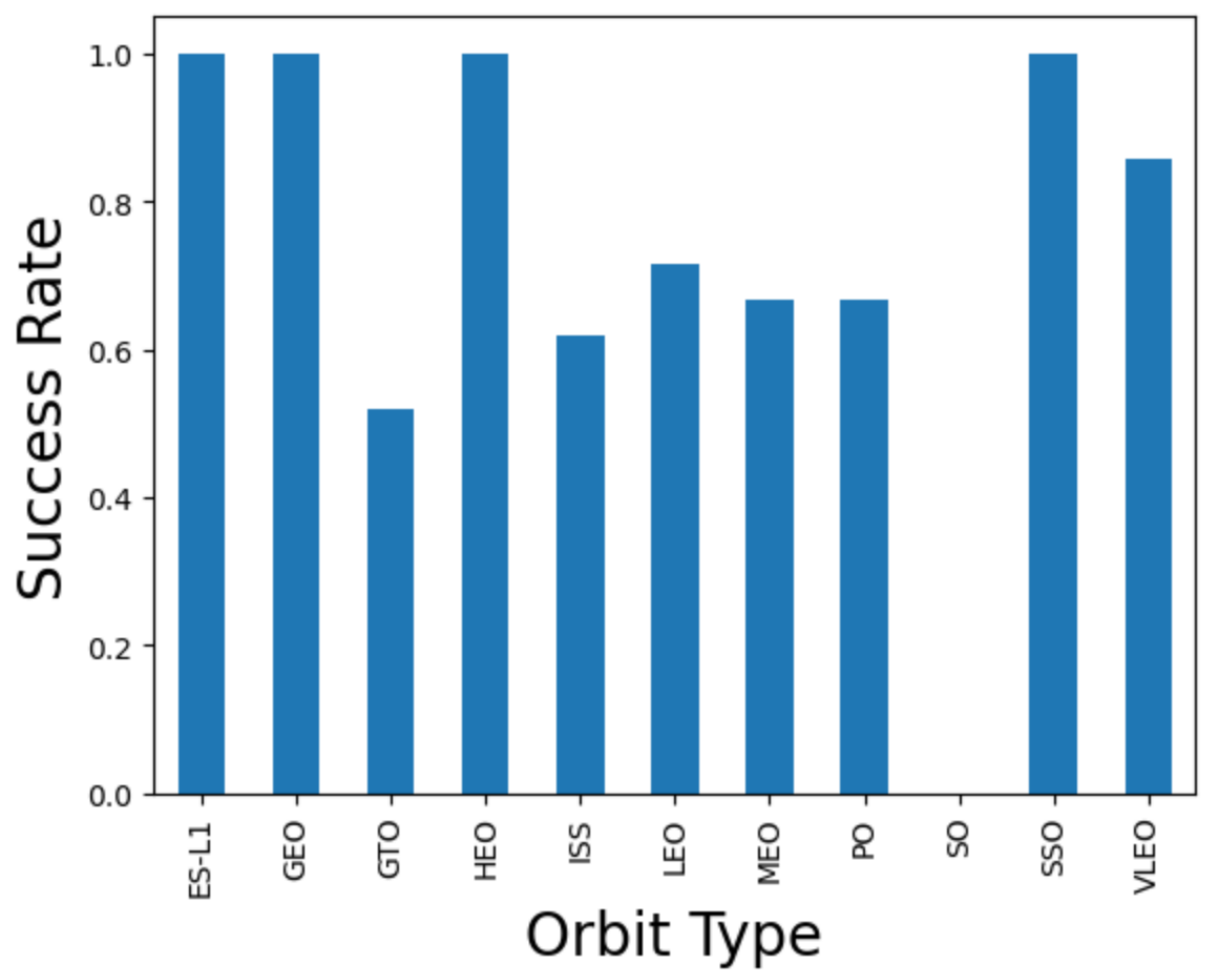
Payload vs. Launch Site



Blue indicates unsuccessful launch while orange indicates successful launches.

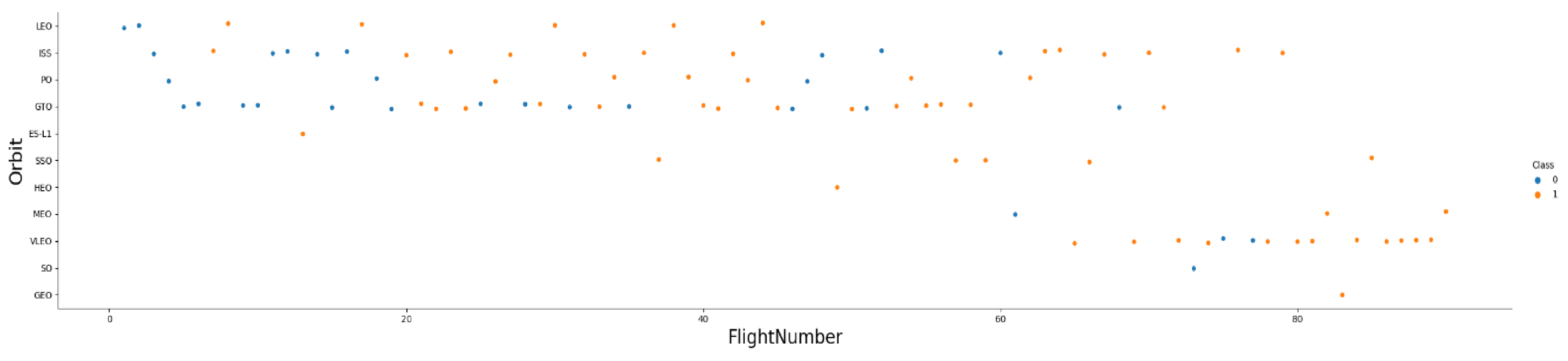
Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass.

Success Rate vs. Orbit Type



- Success Rate of ES-L1, GEO, HEO, SSO are 100%
- GTO has the second lowest success rate behind S) which has a 0% success rate.

Flight Number vs. Orbit Type

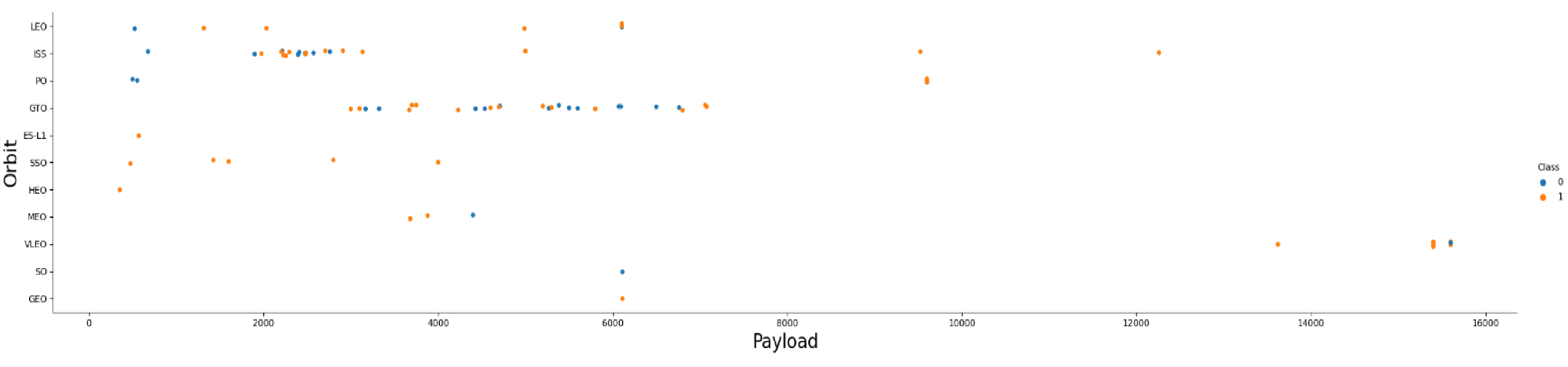


Blue indicates unsuccessful launch while orange indicates successful launches.

Launch Orbit preferences changed over Flight Number. Launch Outcome seems to correlate with this preference.

SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches
SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

Payload vs. Orbit Type



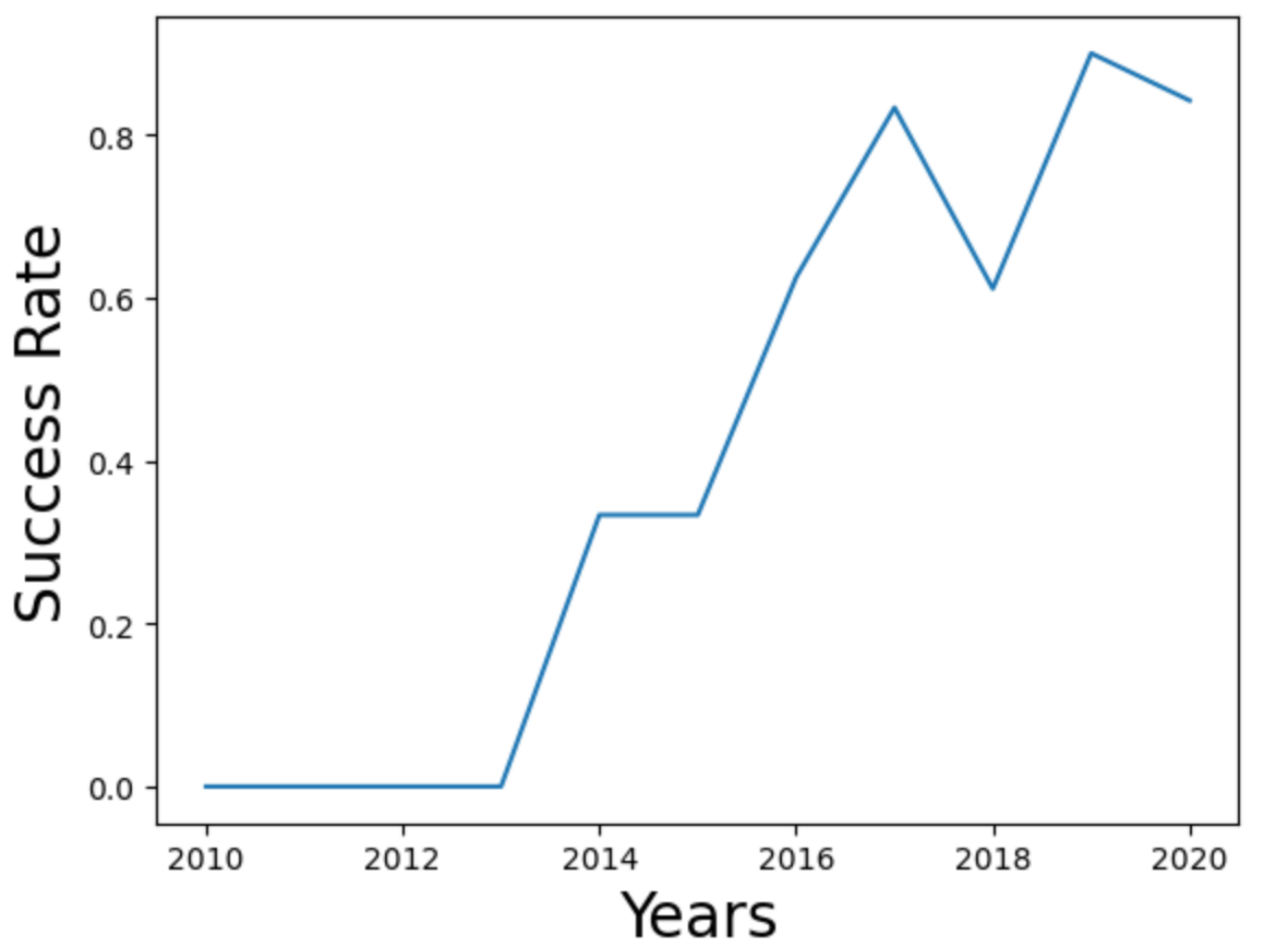
Blue indicates unsuccessful launch while orange indicates successful launches.

Payload mass seems to correlate with orbit

LEO and SSO seem to have relatively low payload mass

The other most successful orbit VLEO only has payload mass values in the higher end of the range

Launch Success Yearly Trend



Success generally increases over time since 2013 with a slight dip in 2018

Success in recent years at around 80%

All Launch Site Names

Task 1

Display the names of the unique launch sites in the space mission

In [11]:
```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTABLE;
```

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

Out[11]:

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

CCAFS SLC-40 and CCAFS LC-40 are the same site either entered incorrectly or the name changed.

Launch Site Names Begin with 'CCA'

Display 5 records where launch sites begin with the string 'CCA'

In [14]:
```
%sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
```

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

Out[14]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success
2010-08-12	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
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success

Each record uses the launch site that begins with ‘CCA’

Total Payload Mass

Task 3

Display the total payload mass carried by boosters launched by NASA (CRS)

In [16]:
```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer='NASA (CRS)';
```

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

Out[16]:

SUM(PAYLOAD_MASS__KG_)
45596

Mass payload was calculated by summing the column labeled PAYLOAD_MASS__KG_

Average Payload Mass by F9 v1.1

Task 4

Display average payload mass carried by booster version F9 v1.1

In [17]:
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version='F9 v1.1'
```

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

Out[17]:

AVG(PAYLOAD_MASS__KG_)
2928.4

Average Payload Mass was calculated by taking the average of the PAYLOAD_MASS__KG_ column sorted by the specific booster type of F9 v1.1

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

In [22]:
```
%sql SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome='Success (ground pad)';
```

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

Out[22]:

MIN(Date)
2015-12-22

The minimum date was selected from the Date column in order to obtain the first date where a rocket landed on a ground pad successfully.

Successful Drone Ship Landing with Payload between 4000 and 6000

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [30]:
```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome='Success (drone ship)' AND PAYLOAD_MASS_
```

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

Out[30]:

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

- The query uses two where parameters that take in the success type and the payload mass drawing the Booster_Versions that had a payload mass between 4000 and 6000 KG and had a successful landing on a drone ship.

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [32]:
```
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTABLE GROUP BY MISSION_OUTCO
```

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

Out[32]:

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

Mission outcome types and the number of those missions were returned.

Boosters Carried Maximum Payload

In [33]:

```
%%sql
SELECT DISTINCT BOOSTER_VERSION
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ = (
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTBL);
```

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

Out[33]:

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

- Using a subquery, max payload mass and is able to sort Booster version into the boosters that have transported the max payload mass.

2015 Launch Records

In [67]:
```
%%sql SELECT substr(Date, 6, 2) as month, DATE, BOOSTER_VERSION, LAUNCH_SITE, Landing_Outcome
FROM SPACEXTABLE
where Landing_Outcome = 'Failure (drone ship)' and substr(Date,1,4)='2015'
```

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

Out[67]:

month	Date	Booster_Version	Launch_Site	Landing_Outcome
10	2015-10-01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

This gives the month, date, booster version, launch site for landing outcomes that were a failure with drone ship landings.

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

In [58]:

```
%%sql SELECT Landing_Outcome, count(*) as Count_Outcomes
FROM SPACEXTABLE
WHERE DATE between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count_outcomes DESC;
```

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

Out[58]:

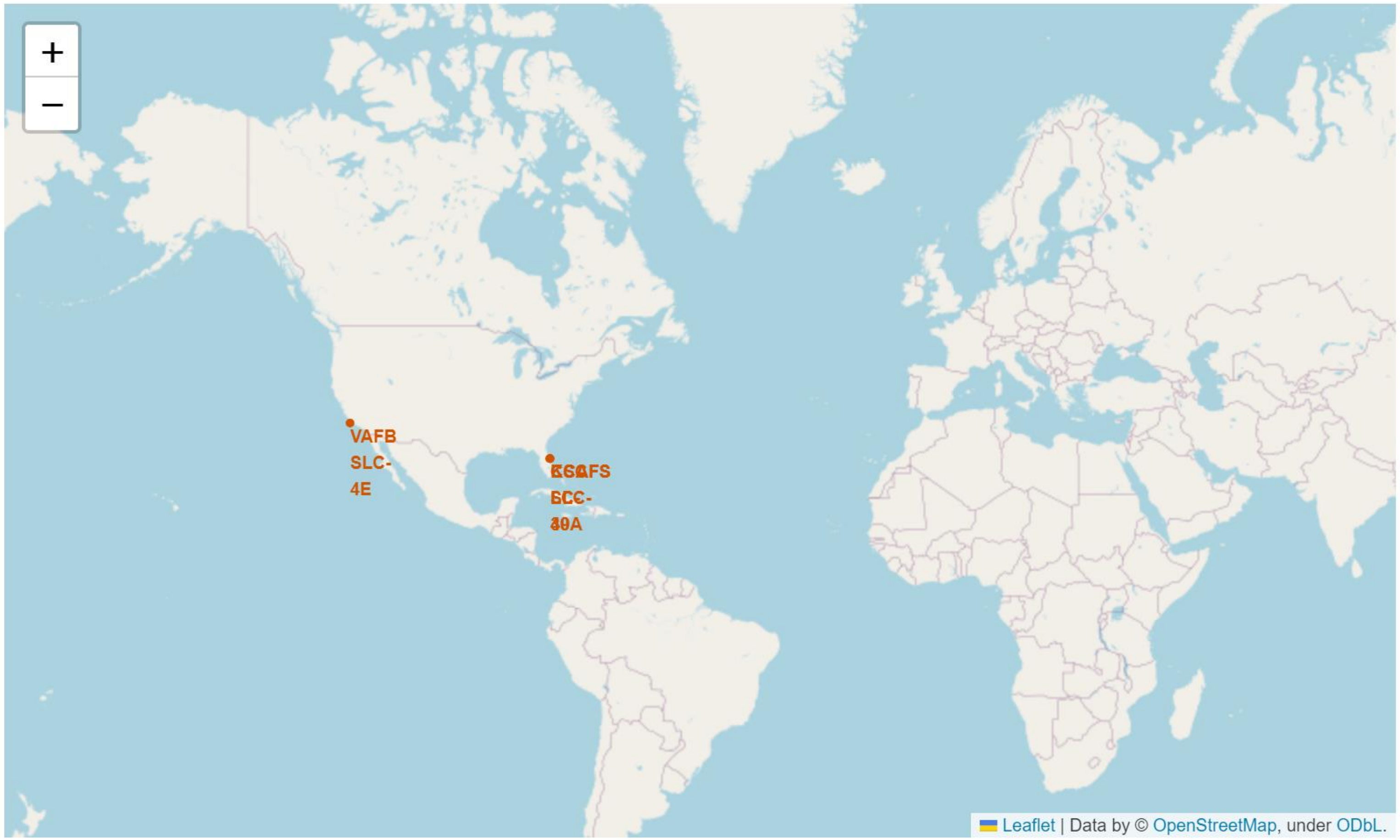
Landing_Outcome	Count_Outcomes
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

- This shows the count of landing outcomes for the time period between 6-4-2010 and 3-20-2017. No attempt was the highest, showing that most landings were not attempted during this time.

Section 3

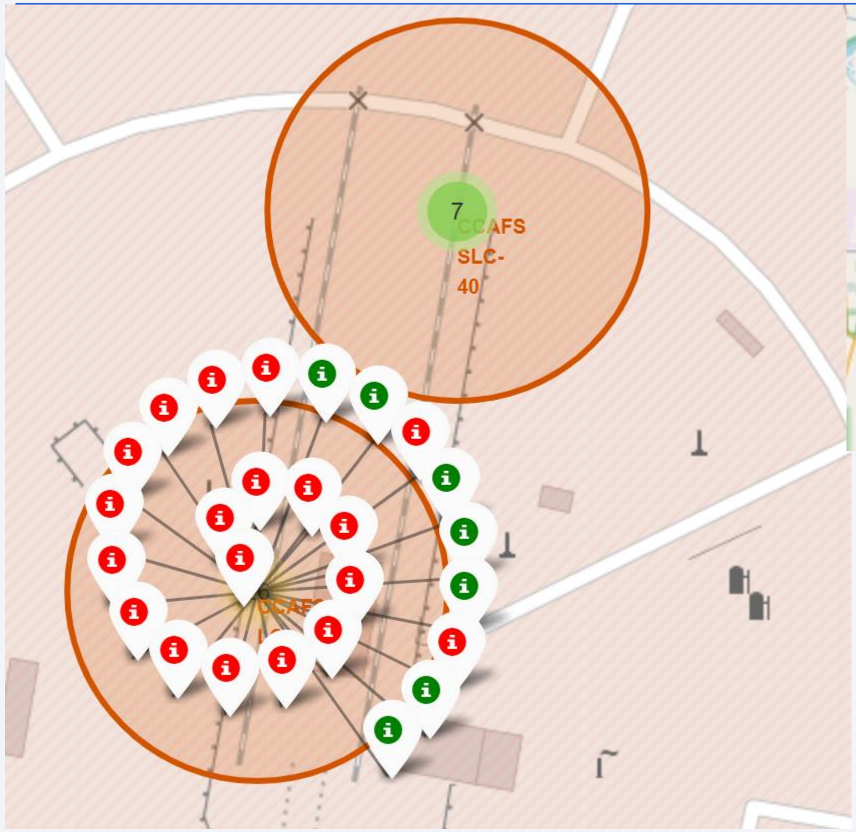
Launch Sites Proximities Analysis

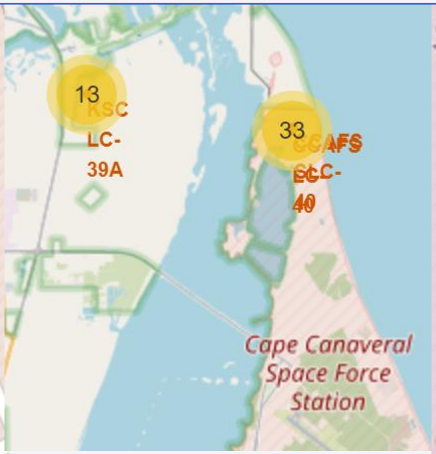
Launch Site Locations

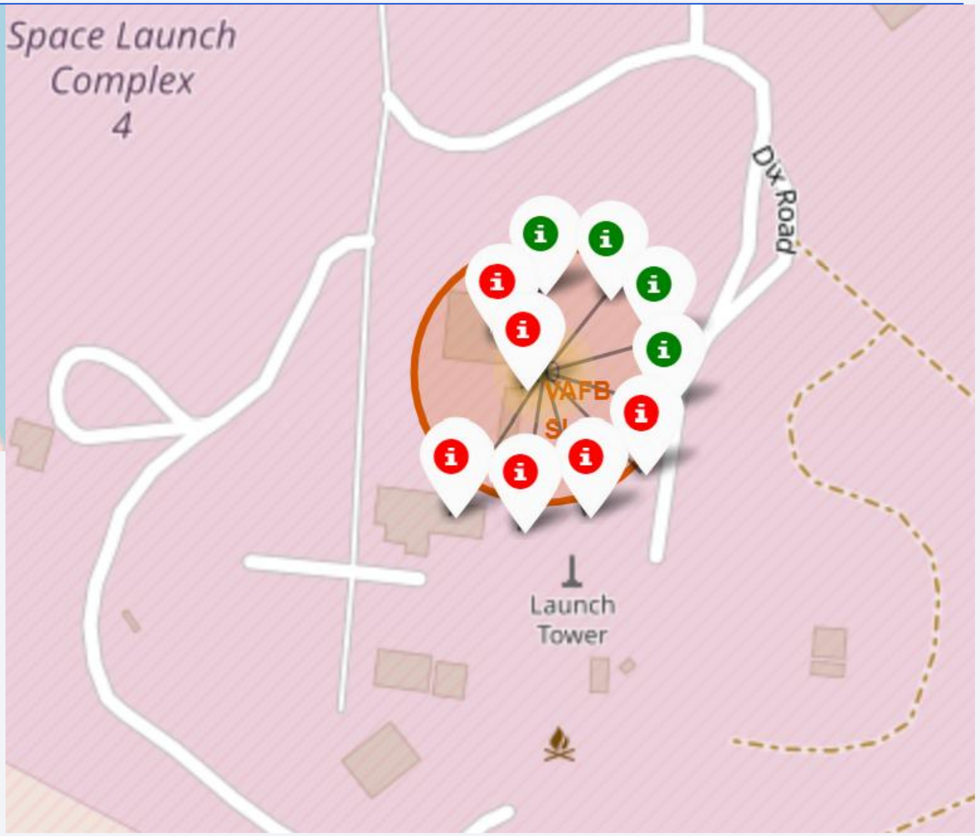


All SpaceX launches are made from the United States in primarily Florida and California coastal areas.

Color Code Launch Markers

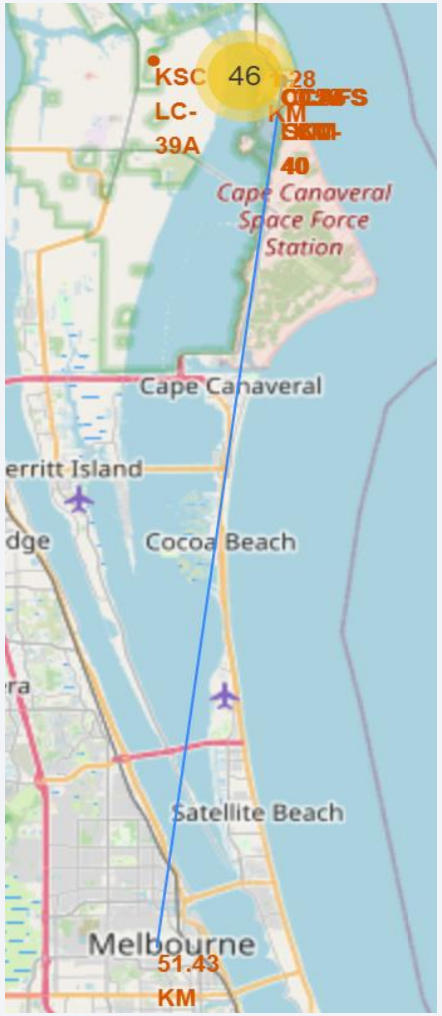






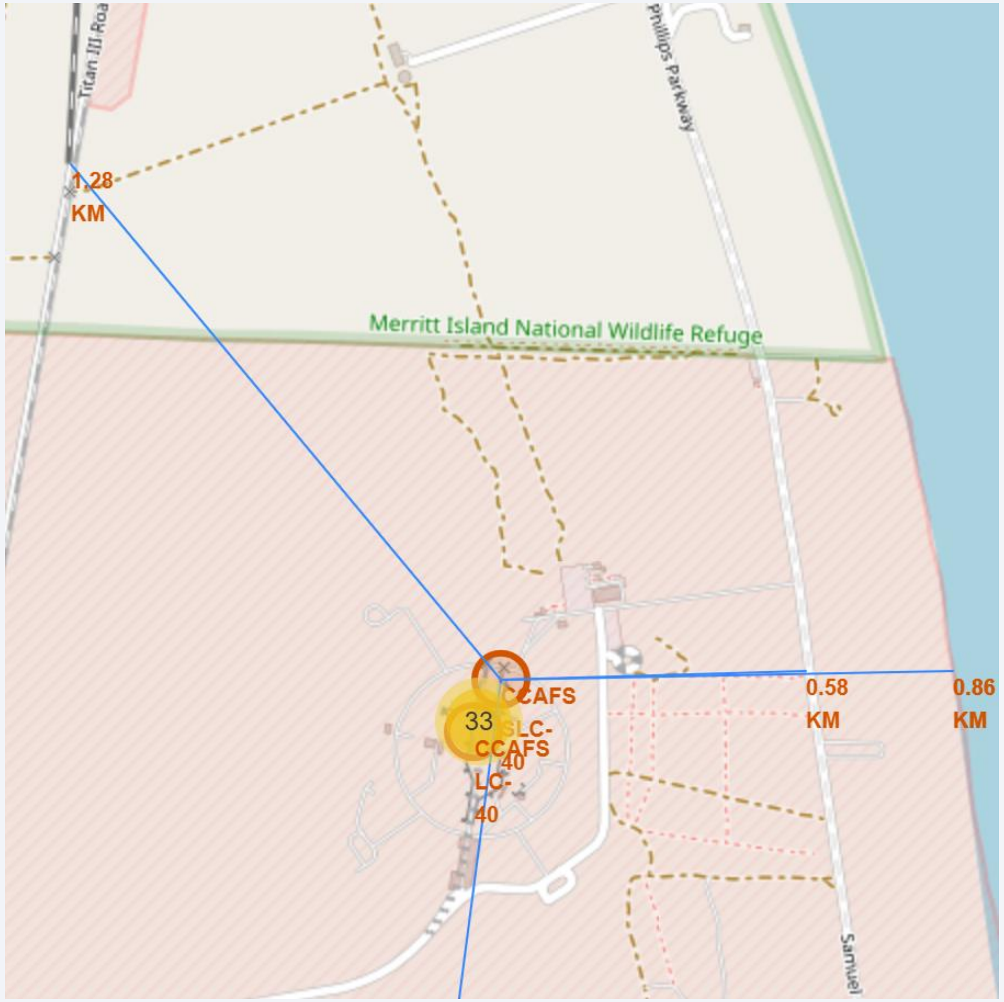
Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed landing (red icon). The Yellow circles near Cape Canaveral launch sites indicate the launch sites and the number of launches performed there.

Launch Site Distance to Land Markers



Here is an example of the map performing a distance calculation to the city of Melbourne, FL from CCAFS SLC – 40.

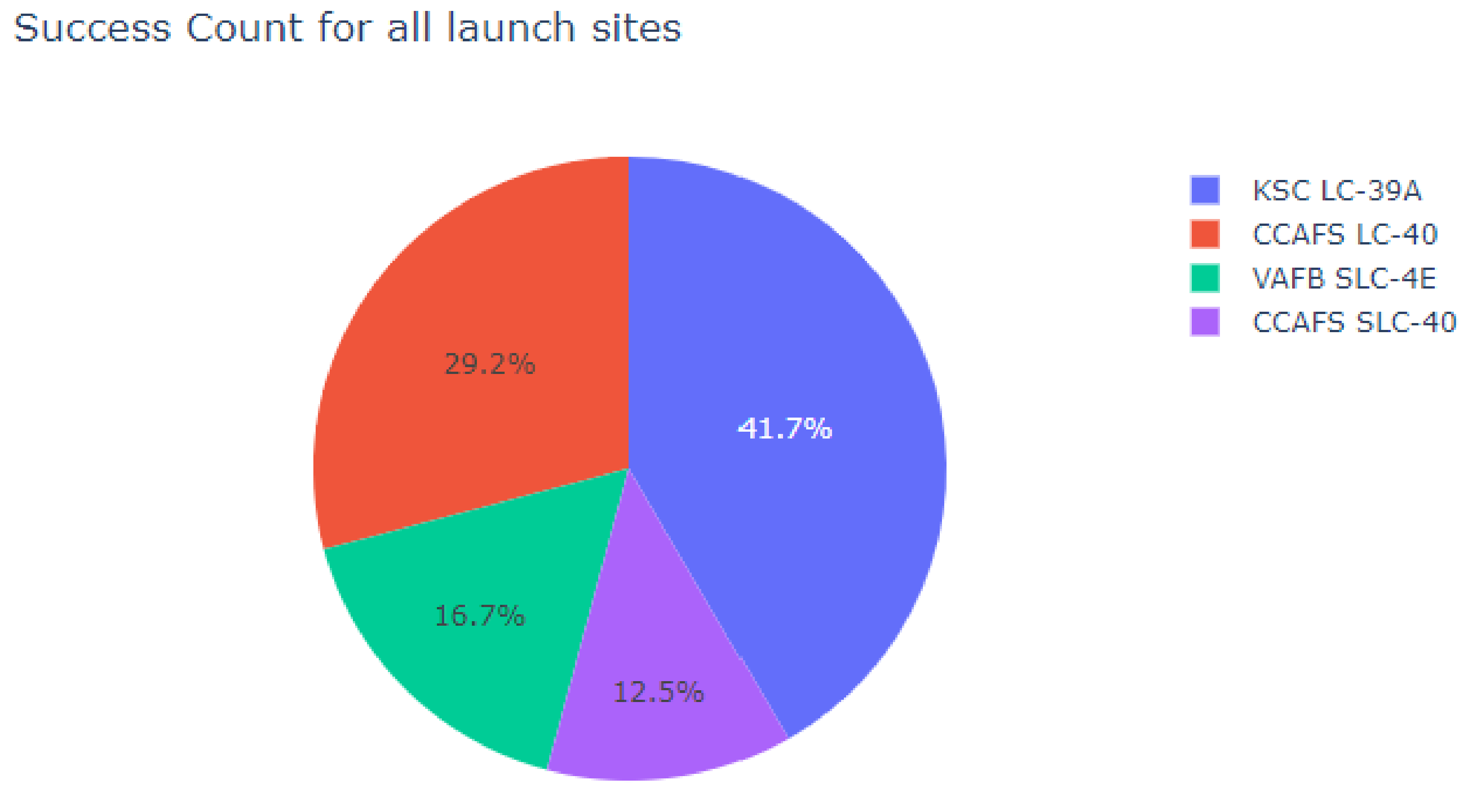
Here we can see this function working closer to CCAFS SLC – 40. It shows the distance to local railroads, highways, and the coast.



Section 4

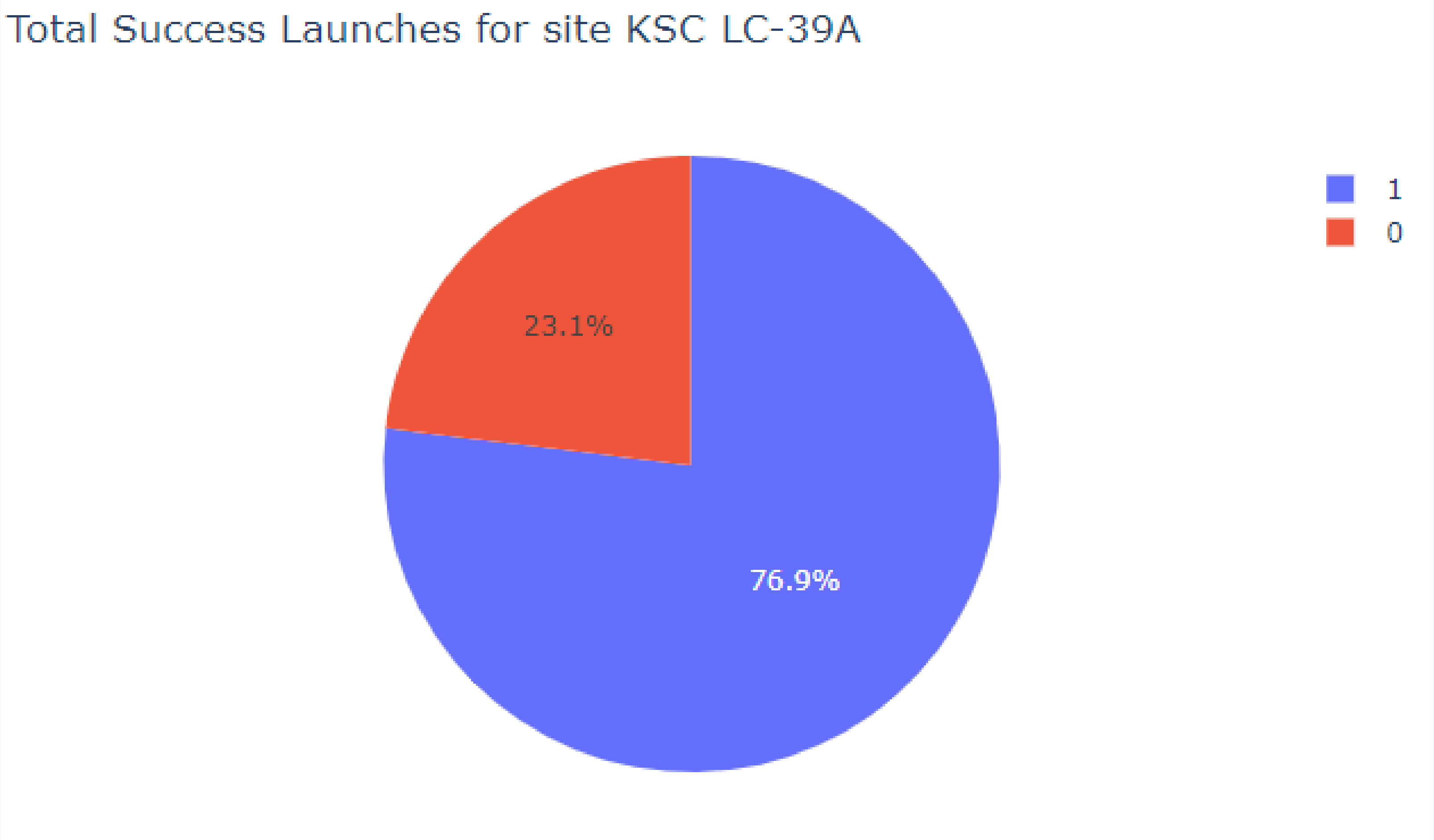
Build a Dashboard with Plotly Dash

Success Count for All Launch Sites – Pie Charts



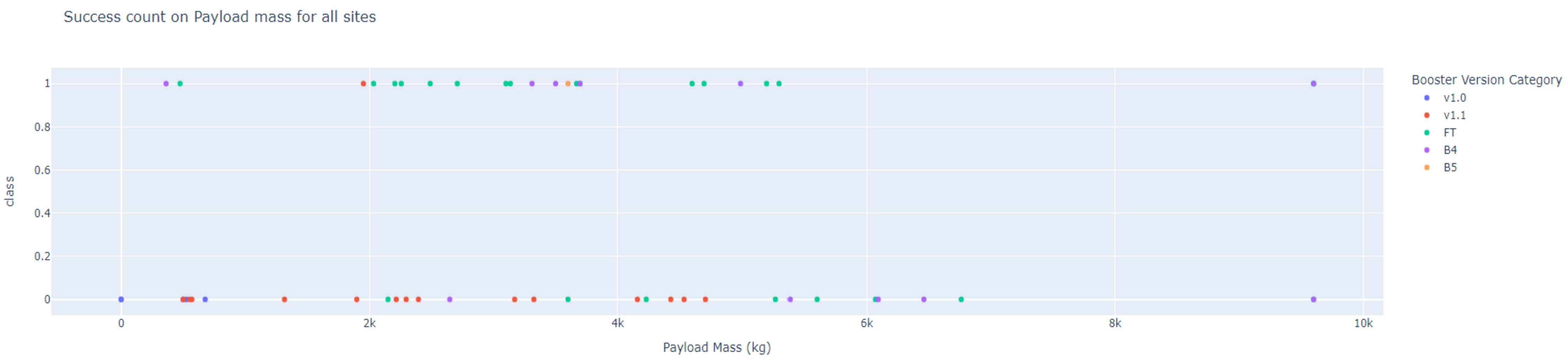
41.7% of Successful launches came from KSC LC-39A while only 12.5% of all successful launches came from CCAFS SLC-40.

SpaceX Launch Success for Site KSC LC – 39A



76.9% of launches from KSC LC-39A were successful.

Successful Launches for All Launch Sites

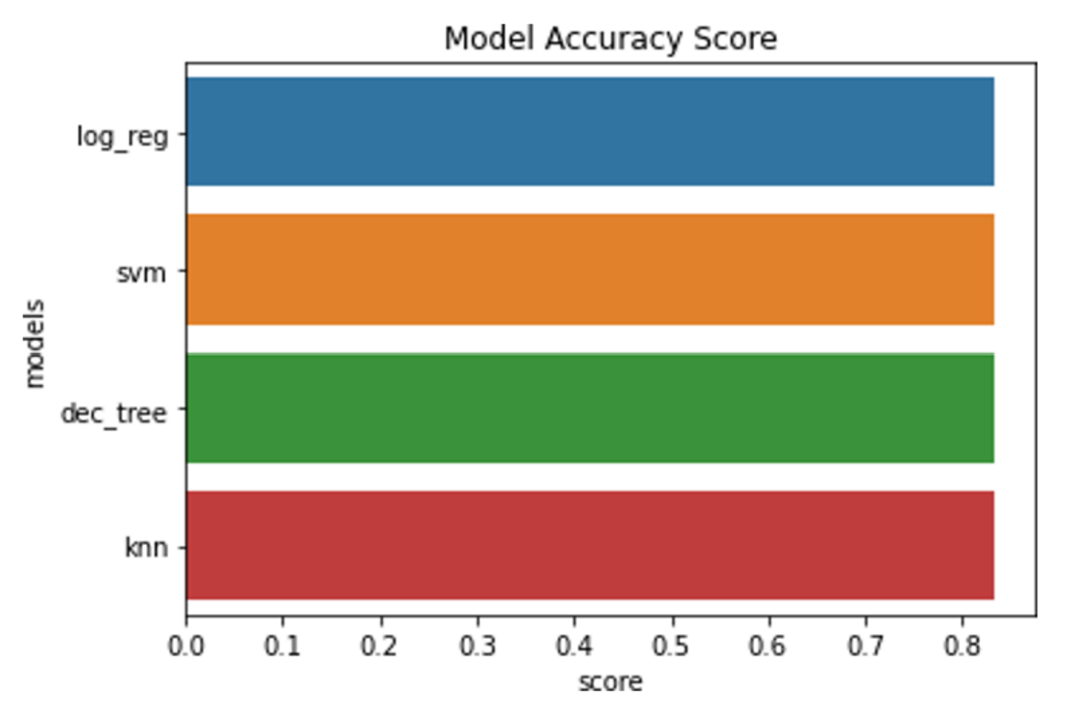


Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size. In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

Section 5

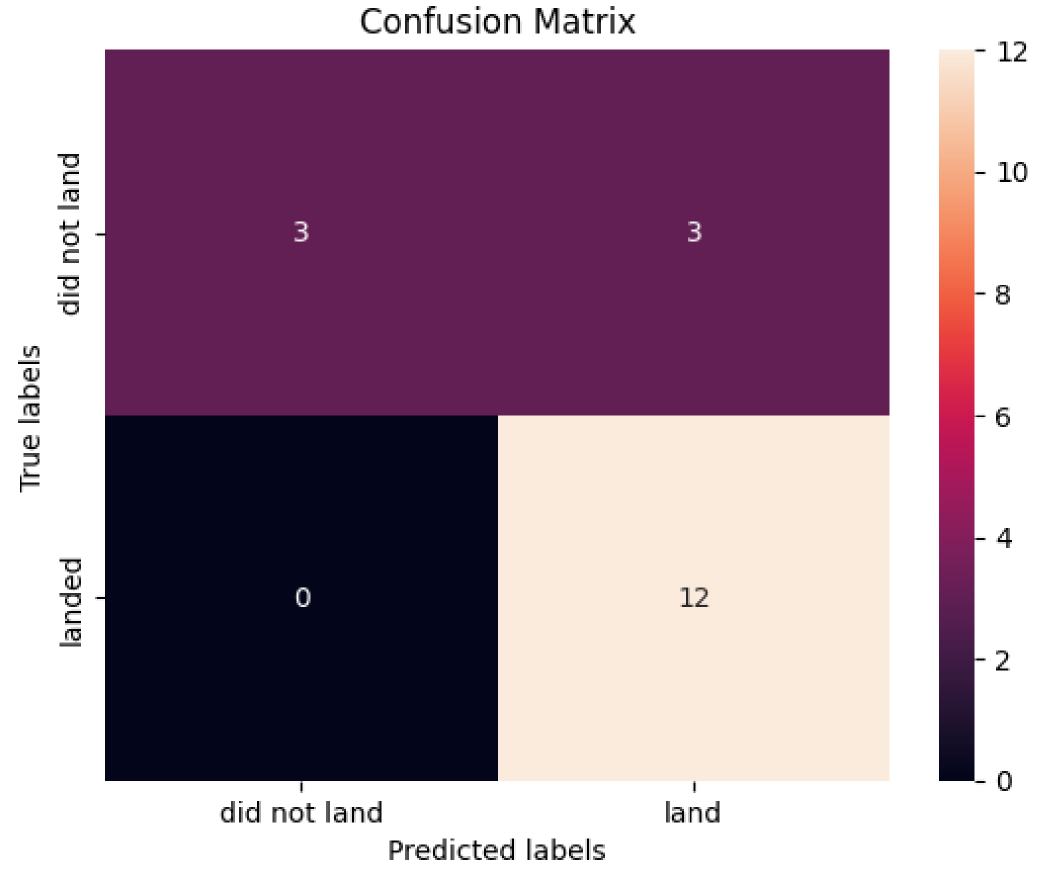
Predictive Analysis (Classification)

Classification Accuracy



Each model had virtually the same accuracy of 83.33%. Likely due to the small sample size of only 18.

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, 3 unsuccessful landings when the true label was unsuccessful landings, 3 successful landings when the true label was unsuccessful landings (false positives). The models over predicted landings.

Conclusions

- Task: to develop a machine learning model for Space Y – a company trying to out perform SpaceX
- The goal of the model is to predict when Stage 1 will successfully land to save approx. $100 million.
- Using data from public SpaceX API and web scraping SpaceX Wikipedia page:
    - Created data labels and stored data into a SQL database
    - Created a dashboard for visualization
    - created a machine learning model with an accuracy of 83%
- SpaceY 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
- Since this model was built off a small data set, more data should be collected to determine the best model to use to have a higher accuracy of predictability.

Appendix

- GitHub Respository for Capstone
  - https://github.com/AKingSolutions/IBM-Data-Science-Certification-Capstone
- Thanks to All Instructors!

Thank you!