IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Khaled Youssef A. Hassoun
October 30, 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result

Executive Summary(Cont.)

- 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 and context

  Space X 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 Space X can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against space X for a rocket launch. This goal of the project is to create a machine learning pipeline to predict if the first stage will land successfully.

- Problems you want to find answers

  - What factors determine if the rocket will land successfully?
  - The interaction amongst various features that determine the success rate of a successful landing.
  - What operating conditions needs to be in place to ensure a successful landing program.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and web scrapping from Wikipedia.
- Perform data wrangling
  - One-hot encoding was applied to categorical features
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- The data was collected using various methods
  - Data collection was done using get request to the SpaceX API.
  - Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().
  - We then cleaned the data, checked for missing values and fill in missing values where necessary.
  - In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.
  - The objective was to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for future analysis.

Data Collection(Cont.)

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.

The next slide will show the flowchart of data collection from API and the one after will show the flowchart of data collection from webscraping.

Space X API Data Columns:

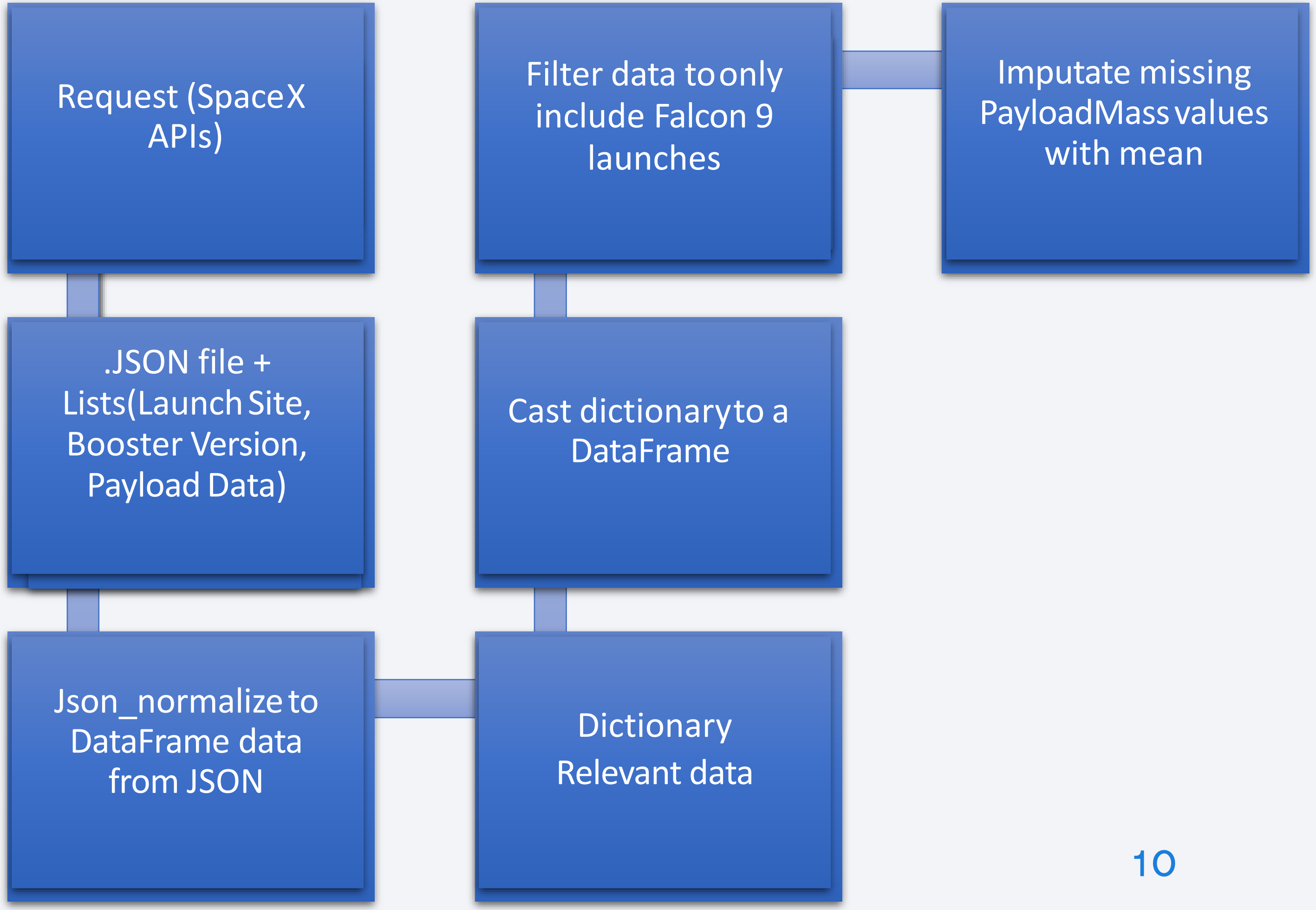
FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Wikipedia Webscrape Data Columns:

Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

- I used the get request to the SpaceX API to collect data, clean the requested data and did some basic data wrangling and formatting.
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/Data%20Collection%20API.ipynb



Request (SpaceX APIs)

.JSON file + Lists(Launch Site, Booster Version, Payload Data)

Json_normalize to DataFrame data from JSON

Dictionary Relevant data

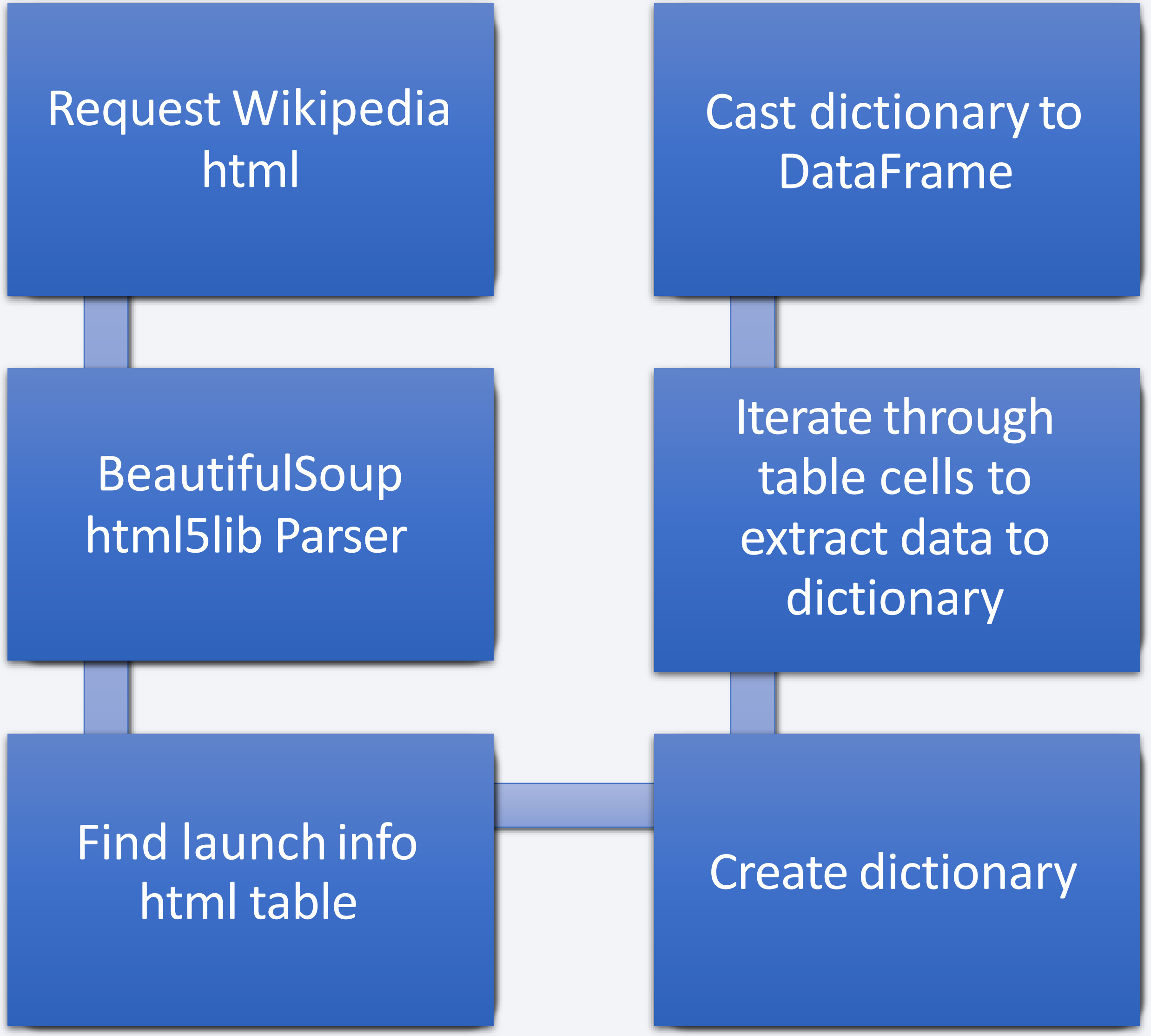
Cast dictionary to a DataFrame

Filter data to only include Falcon 9 launches

Imputate missing PayloadMass values with mean

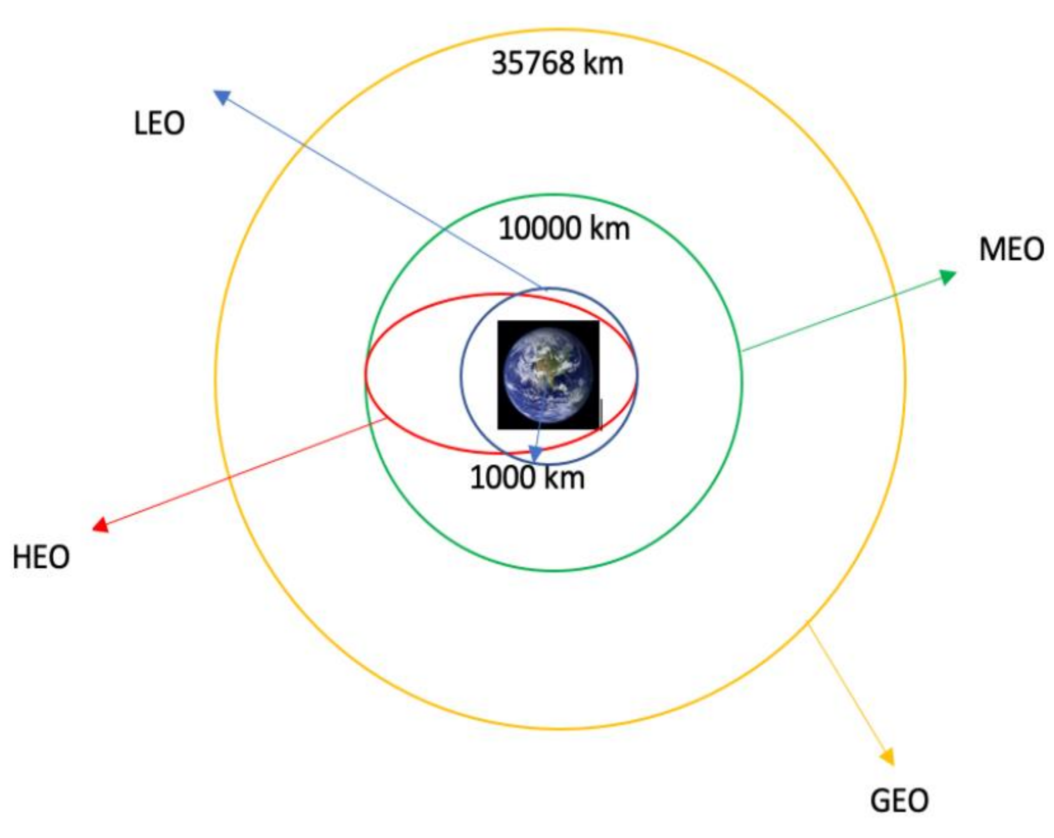
Data Collection - Scraping

- I applied web scrapping to web scrap Falcon 9 launch records with BeautifulSoup
- I parsed the table and converted it into a pandas dataframe.
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/Webscrapping.ipynb


Request Wikipedia html
BeautifulSoup html5lib Parser
Find launch info html table
Create dictionary
Iterate through table cells to extract data to dictionary
Cast dictionary to DataFrame

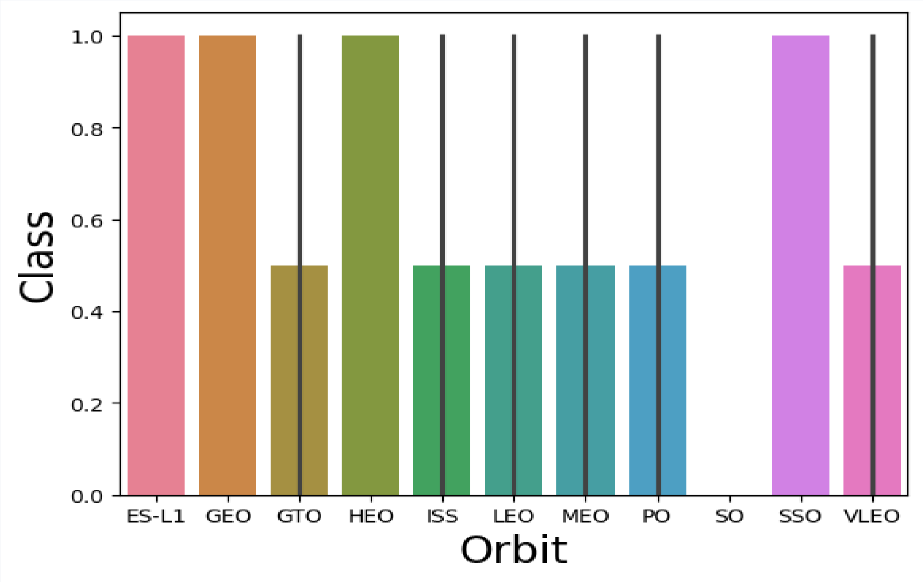
Data Wrangling

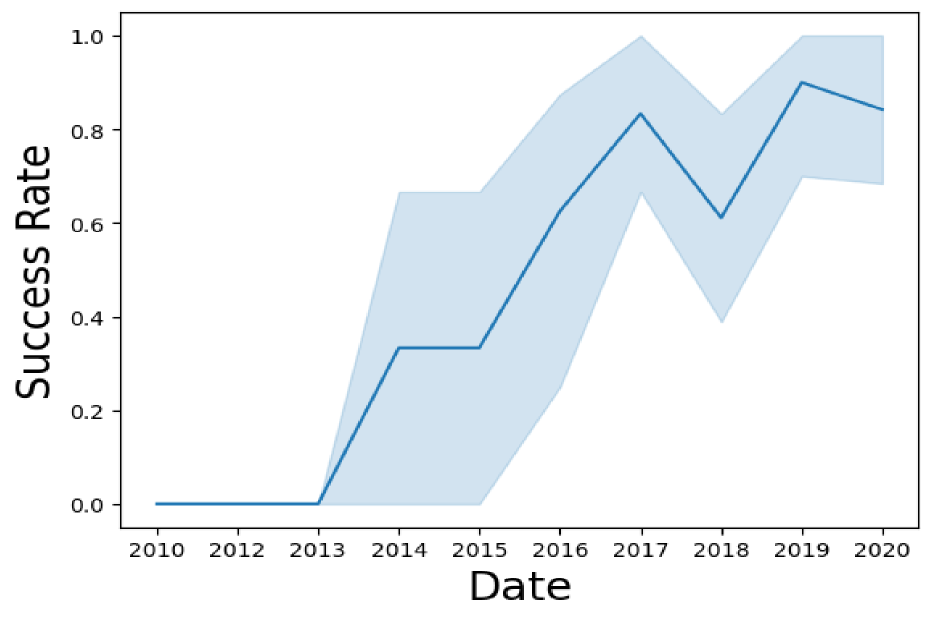
- I performed exploratory data analysis and determined the training labels where successful=1 and failure=0.
- I calculated the number of launches at each site, and the number and occurrence of each orbits
- I created landing outcome label from outcome column and exported the results to csv.
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/Data%20Wrangling.ipynb



EDA with Data Visualization

- I explored the data by visualizing the relationship between flight number and launch Site, payload and launch site, success rate of each orbit type, flight number and orbit type, the launch success yearly trend.

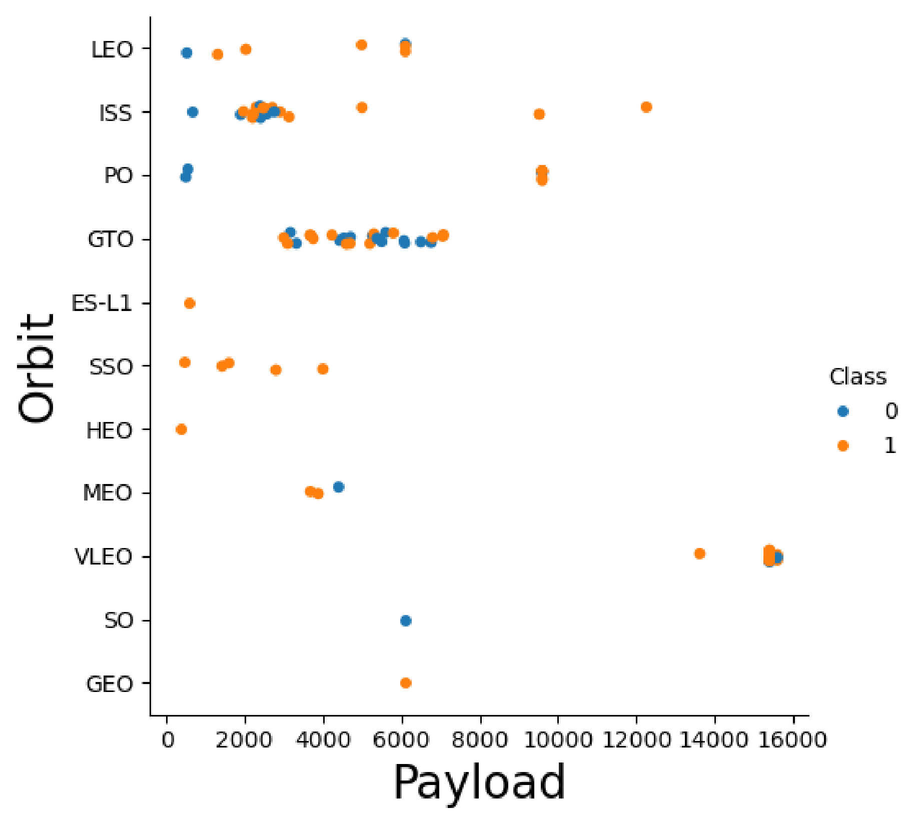




EDA with Data Visualization(Cont.)

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 I used to compare relationships between variables to decide if a relationship exists so that they could be used in training the machine learning model.



The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/Data%20Visualiztaion.ipynb

EDA with SQL

- I loaded the SpaceX dataset into the corresponding table in a Db2 Database.
- Execute SQL queries using SQL Python Integration.
- I applied EDA with SQL to get insight from the data. I wrote queries to find out for instance:
    - The names of unique launch sites in the space mission.
    - The total payload mass carried by boosters launched by NASA (CRS)
    - The average payload mass carried by booster version F9 v1.1
    - The total number of successful and failure mission outcomes
    - The failed landing outcomes in drone ship, their booster version and launch site names.
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/SQL%20Management.ipynb

Build an Interactive Map with Folium

Folium maps marks all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.

I assigned the feature launch outcomes (failure or success) to class 0 and 1.i.e., 0 for failure, and 1 for success.

Using the color-labeled marker clusters, I identified which launch sites have relatively high success rate.

We calculated the distances between a launch site to its proximities. I answered some question for instance:

-Are launch sites near railways, highways and coastlines.

-Do launch sites keep certain distance away from cities.

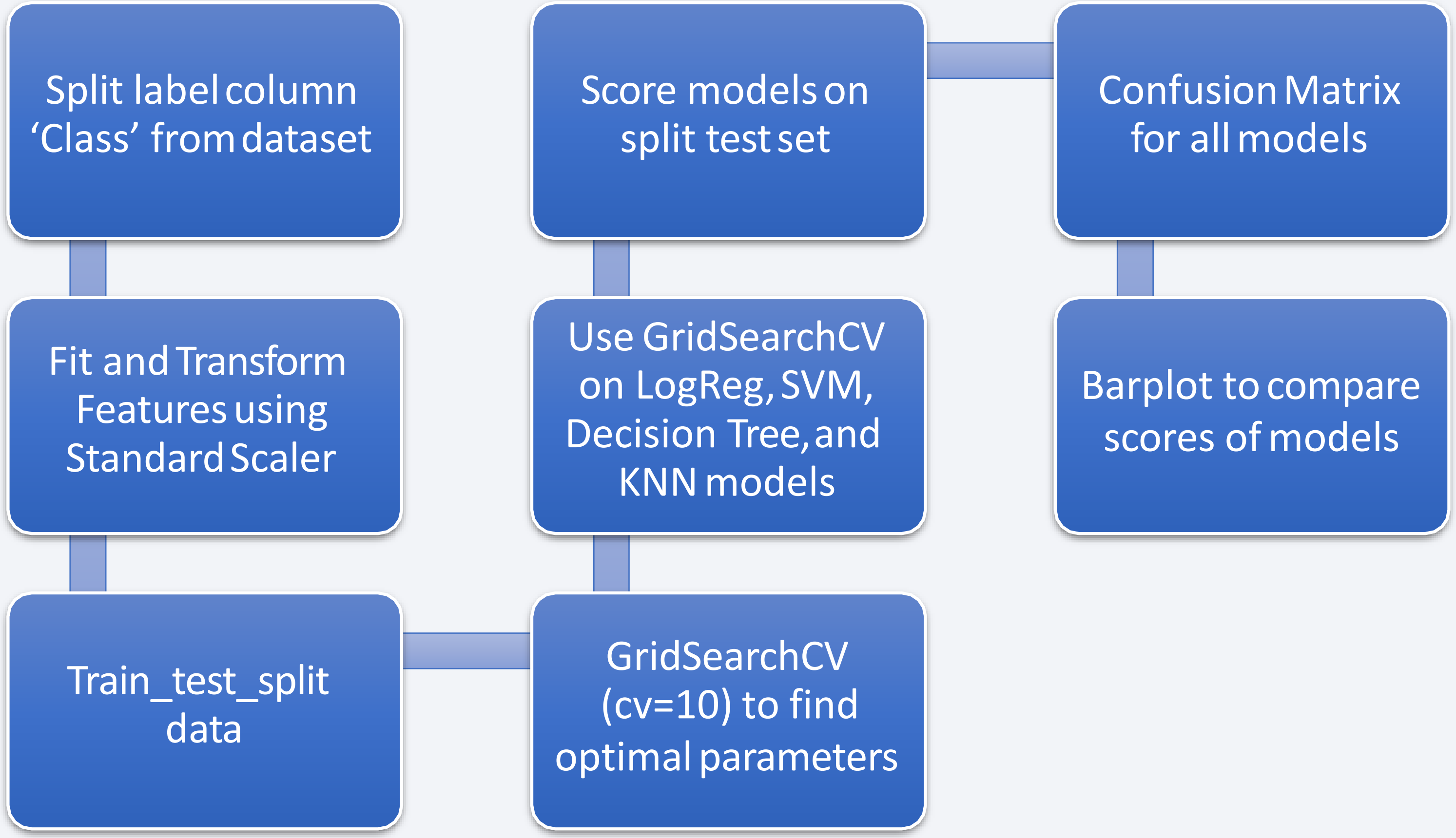
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/Exploratory%20Data%20Analysis.ipynb

Build a Dashboard with Plotly Dash

- I built an interactive dashboard with Plotly dash
- I plotted pie charts showing the total launches by a certain sites
- I plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version.
- Dashboard includes a pie chart and a scatter plot. Pie chart can be selected to show distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.
- Scatter plot takes two inputs: All sites or individual site and payload mass on a slider between 0 and 10000 kg. The scatter plot can help us see how success varies across launch sites, payload mass, and booster version category.
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- I loaded the data using numpy and pandas, transformed the data, split our data into training and testing.
- I built different machine learning models and tune different hyperparameters using GridSearchCV.
- I used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.
- I found the best performing classification model.
- The link to the notebook is https://github.com/Kyahpoots/Applied-Data-Science-Capstone/blob/main/Model%20Development.ipynb



Split label column ‘Class’ from dataset

Fit and Transform Features using Standard Scaler

Train_test_split data

GridSearchCV (cv=10) to find optimal parameters

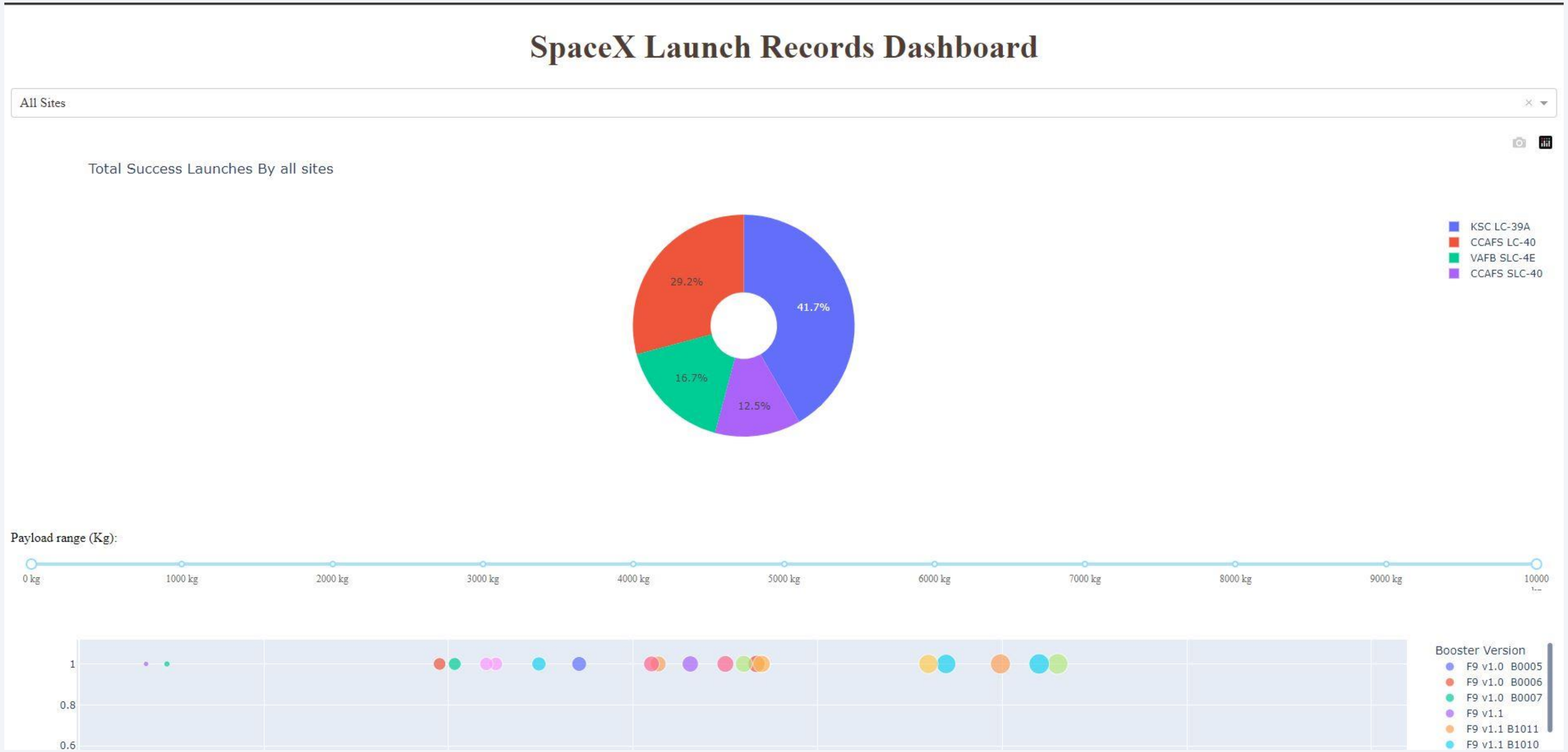
Use GridSearchCV on LogReg, SVM, Decision Tree, and KNN models

Score models on split test set

Confusion Matrix for all models

Barplot to compare scores of models

Results



SpaceX Launch Records Dashboard

All Sites

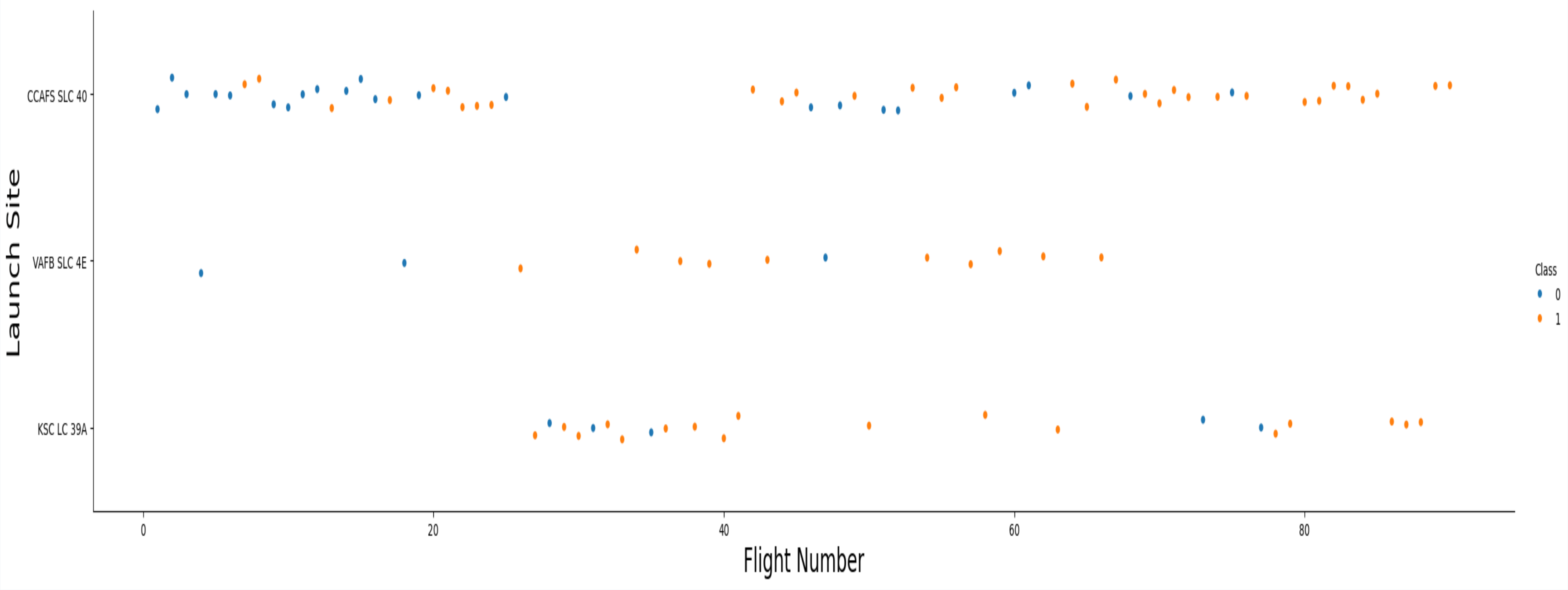
Payload range (Kg):

- This is a preview of the Plotly dashboard. The following sides will show the results of EDA with visualization, EDA with SQL, Interactive Map with Folium, and finally the results of our model with about 83% accuracy.

Section 2

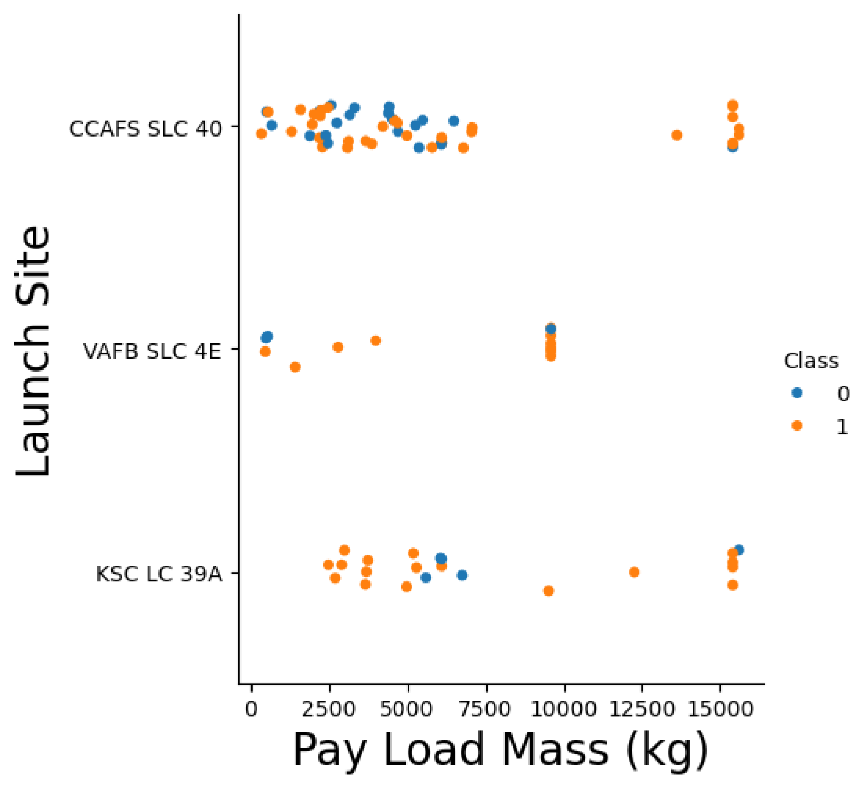
Insights drawn from EDA

Flight Number vs. Launch Site



- From the plot, we found that the larger the flight amount at a launch site, the greater the success rate at a launch site. CCAFS appears to be the main launch site as it has the most volume. Orange indicates successful launch; blue indicates unsuccessful launch.

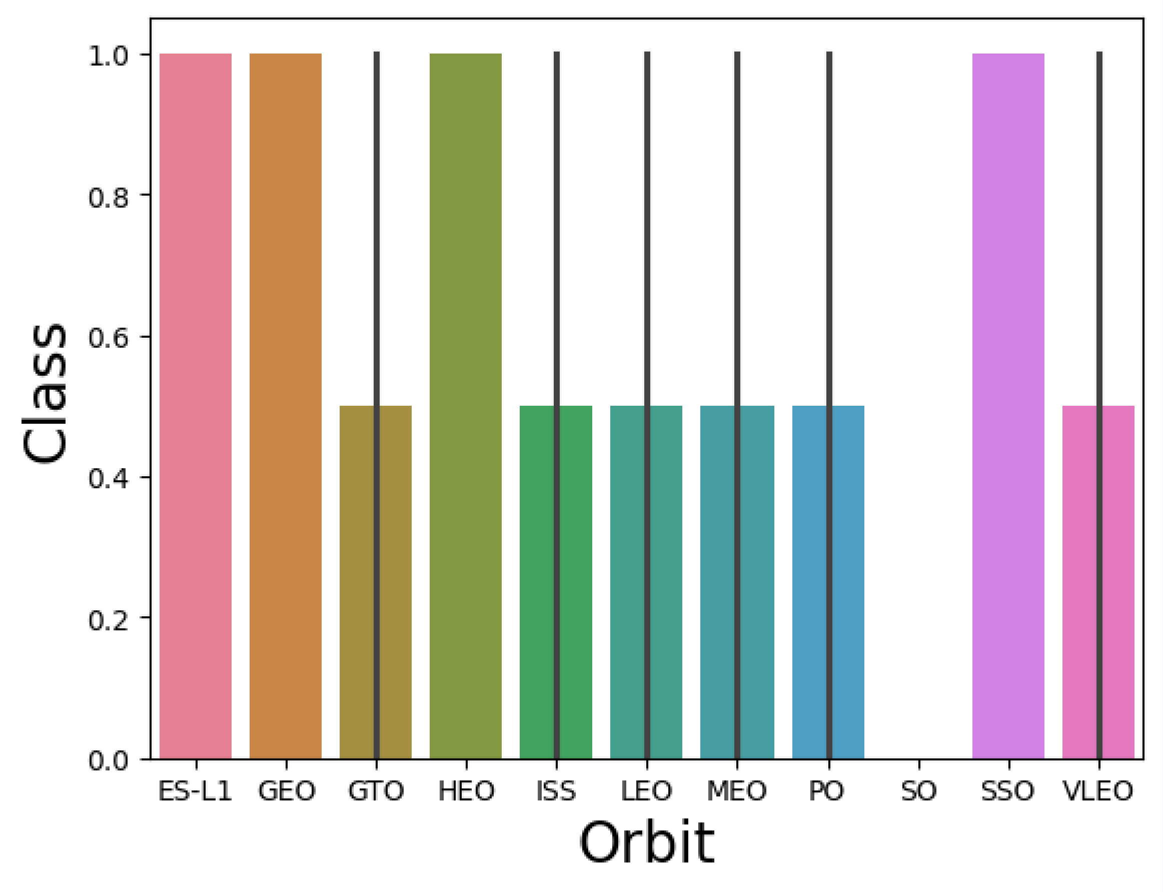
Payload vs. Launch Site



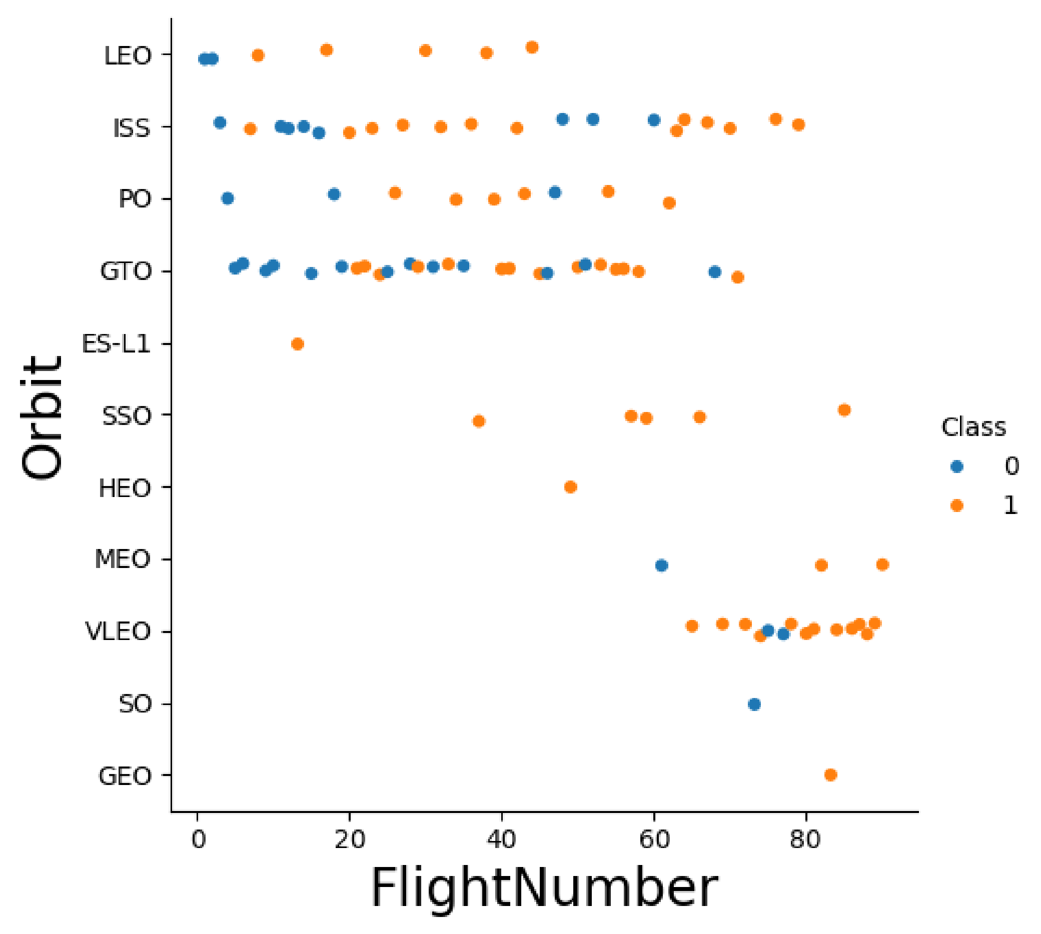
- From the plot, the greater the payload mass for the launch site the higher the success rate for the rocket. Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass. Orange indicates successful launch; blue indicates unsuccessful launch.

Success Rate vs. Orbit Type

- From the plot, we can see that ES-L1, GEO, HEO, SSO had the most success rate.
- While GTO, ISS, LEO, MEO, PO, VLEO had 50% success rate



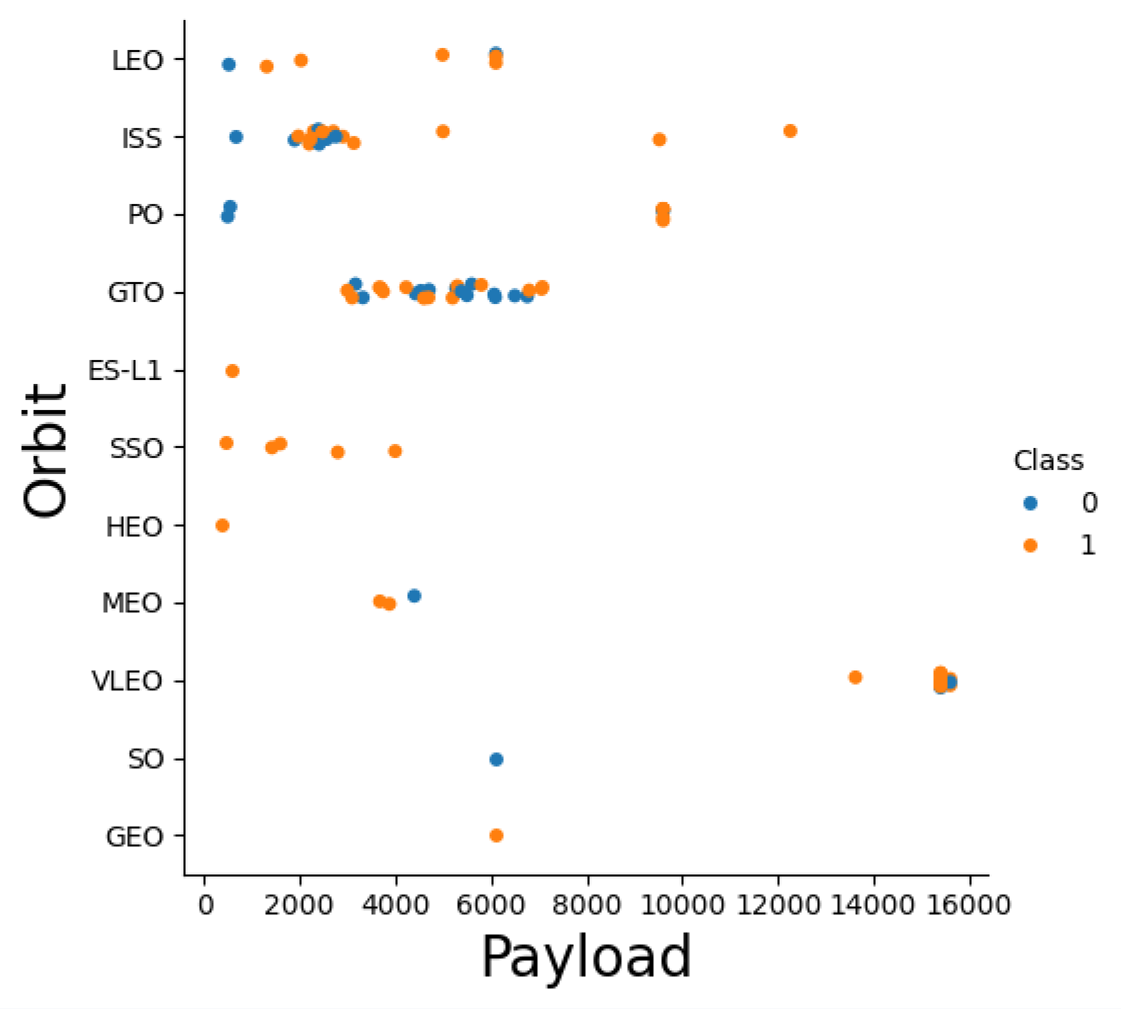
Flight Number vs. Orbit Type



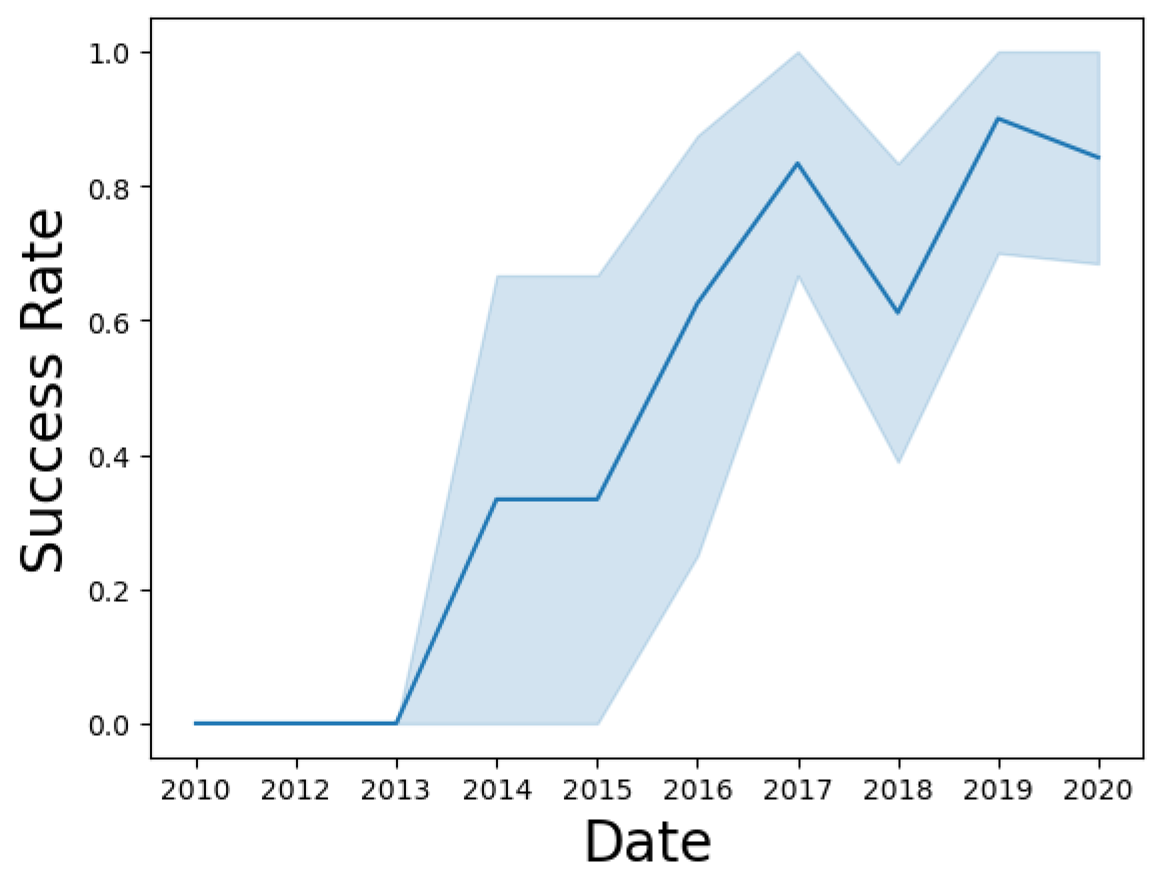
- The plot shows the Flight Number vs. Orbit type. We observe that in the LEO orbit, success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.
- Launch Orbit preferences changed over Flight Number. Launch Outcome seems to correlate with this preference.

Payload vs. Orbit Type

- We can observe that with heavy payloads, the successful landing are more for PO, LEO and ISS orbits.
- Payload mass seems to correlate with orbit.
- The most successful orbit is VLEO only has payload mass values in the higher end of the range.

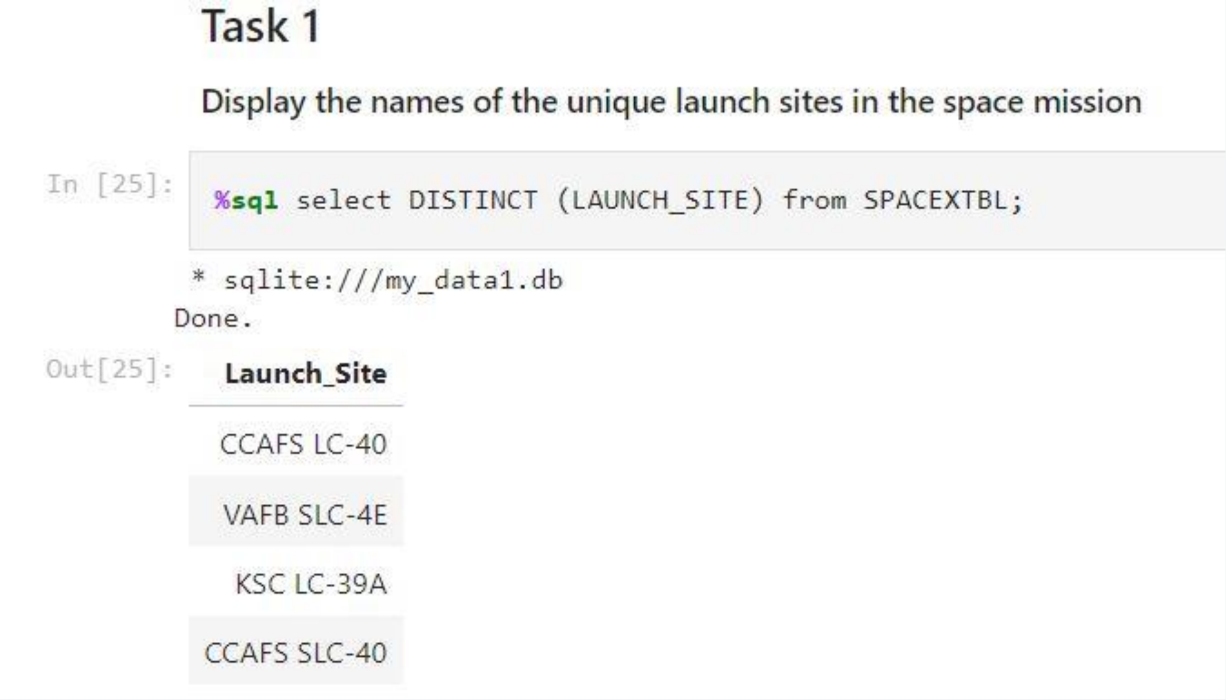


Launch Success Yearly Trend



- From the plot, we can observe that success rate generally increases over time since 2013 with a slight dip in 2018 kept on increasing till 2020.
- 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 [25]: %sql select DISTINCT (LAUNCH_SITE) from SPACEXTBL;
```

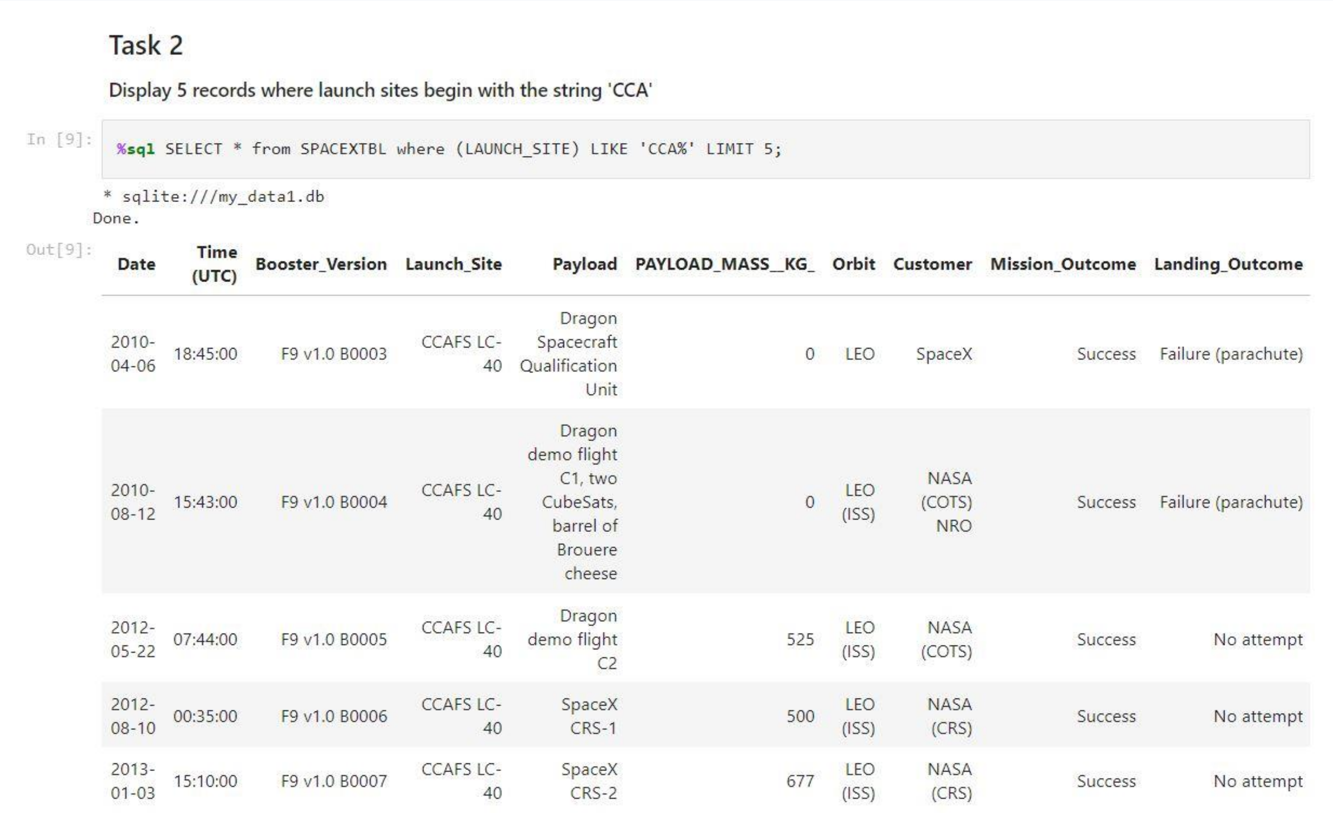
```
 * sqlite:///my_data1.db
Done.
```

Out[25]:

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

- I used the key word **DISTINCT** to show only unique launch sites from the SpaceX data.

Launch Site Names Begin with 'CCA'



Task 2

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

```
In [9]:  %sql SELECT * from SPACEXTBL where (LAUNCH_SITE) LIKE 'CCA%' LIMIT 5;
```

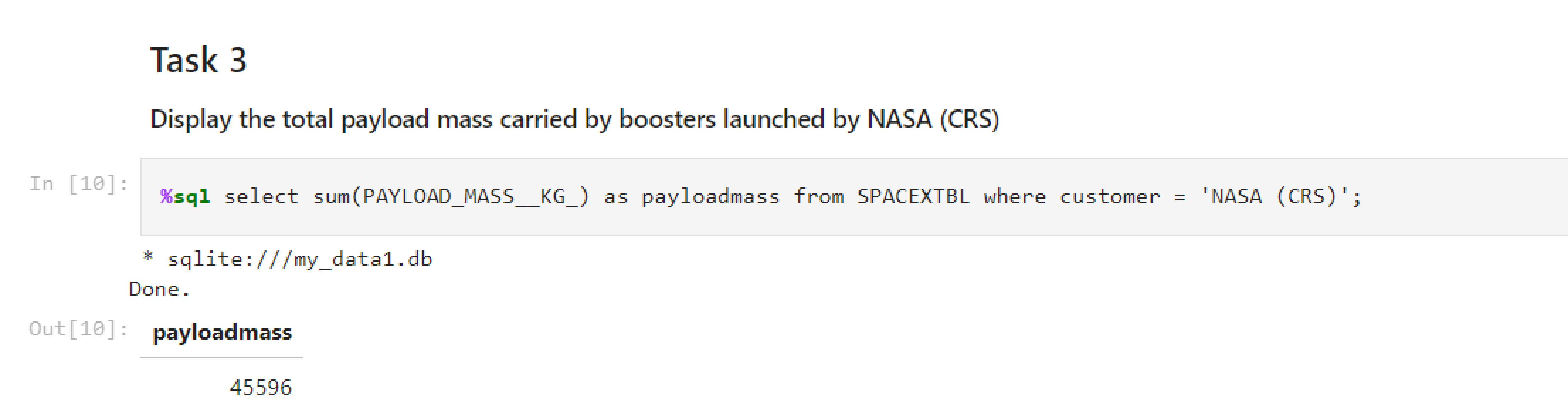
```
 * sqlite:///my_data1.db
Done.
```

Out[9]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- I used the query above to display 5 records where launch sites begin with `CCA`

Total Payload Mass



Task 3

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

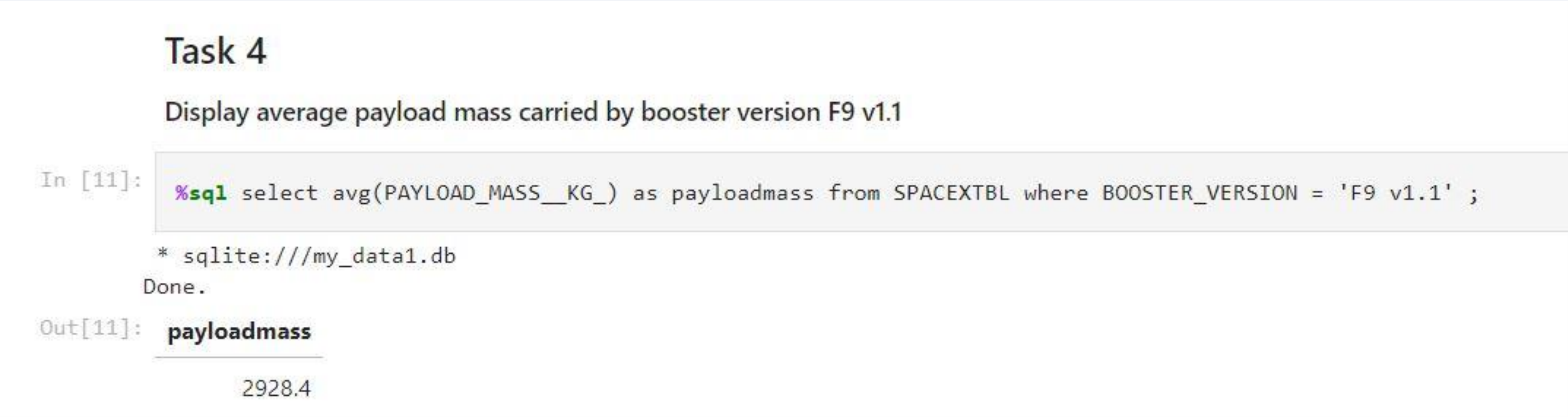
```
In [10]:  %sql select sum(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTBL where customer = 'NASA (CRS)';

  * sqlite:///my_data1.db
 Done.

Out[10]:  payloadmass
          -----------
                45596
```

- This query sums the total payload mass in kg. I calculated the total payload carried by boosters from NASA as 45596.

Average Payload Mass by F9 v1.1



Task 4

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

```
In [11]: %sql select avg(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1' ;
```

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

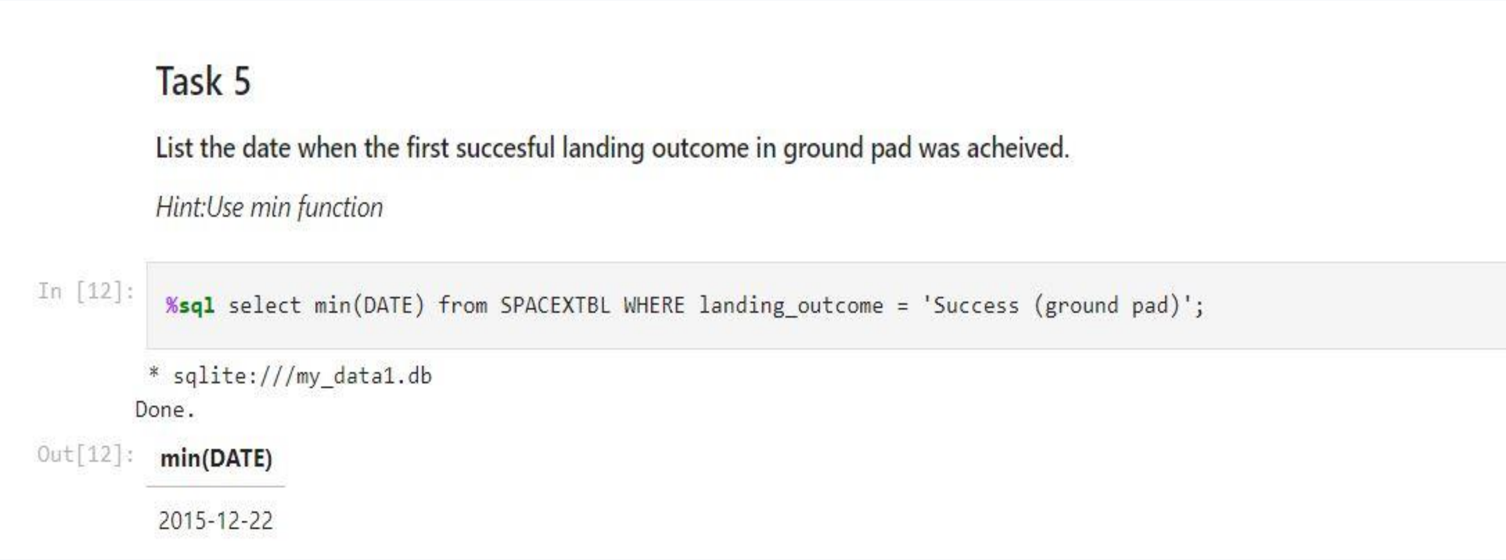
Out[11]:

payloadmass
2928.4

- I calculated the average payload mass carried by booster version F9 v1.1 as 2928.4
- This query calculates the average payload mass or launches which used booster version F9 v1.1
- Average payload mass of F9 1.1 is on the low end of our payload mass range

First Successful Ground Landing Date

- We observed that the dates of the first successful landing outcome on ground pad was 22nd December 2015
- This query returns the first successful ground pad landing date.
- First ground pad landing wasn't until the end of 2015.
- Successful landings in general appear starting 2014.



Task 5

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

Hint:Use min function

In [12]:

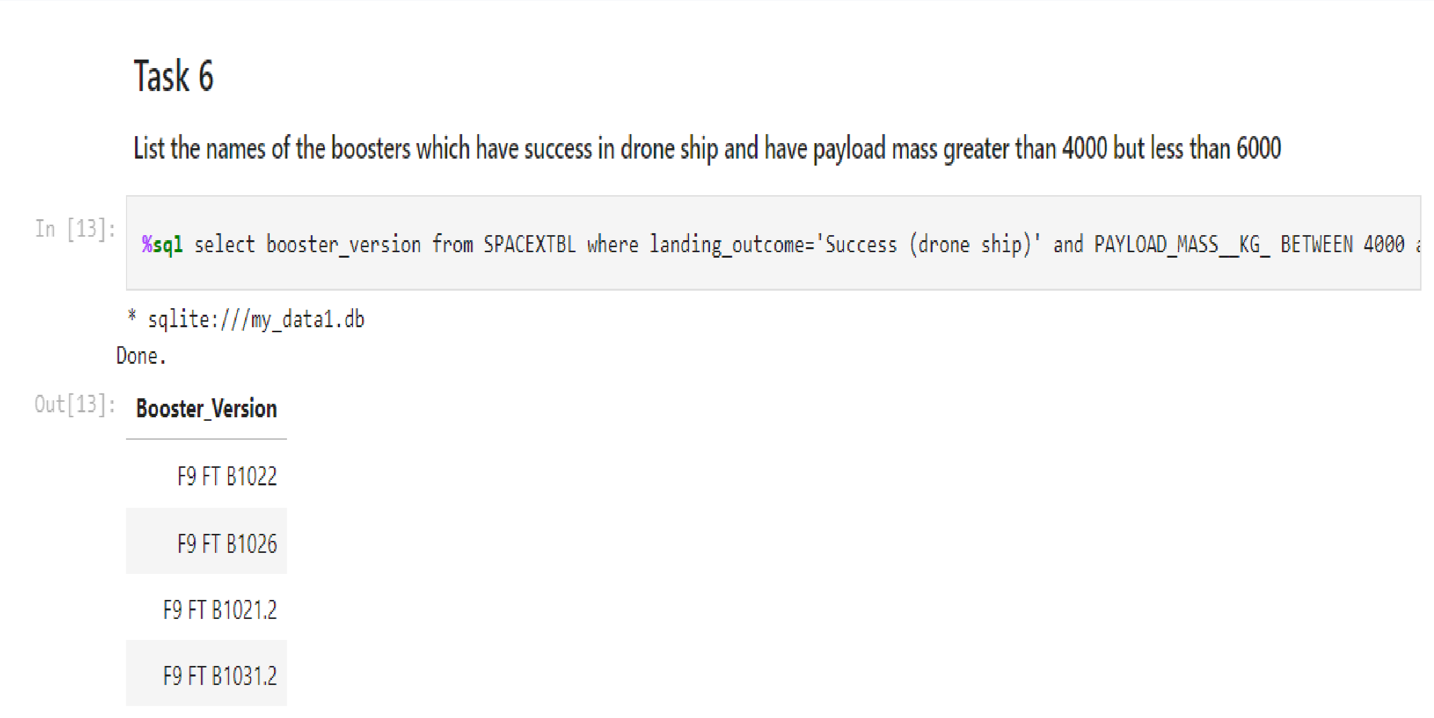
```
%sql select min(DATE) from SPACEXTBL WHERE landing_outcome = 'Success (ground pad)';
```

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

Out[12]:

min(DATE)
2015-12-22

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 [13]: %sql select booster_version from SPACEXTBL where landing_outcome='Success (drone ship)' and PAYLOAD_MASS__KG_ BETWEEN 4000 a
```

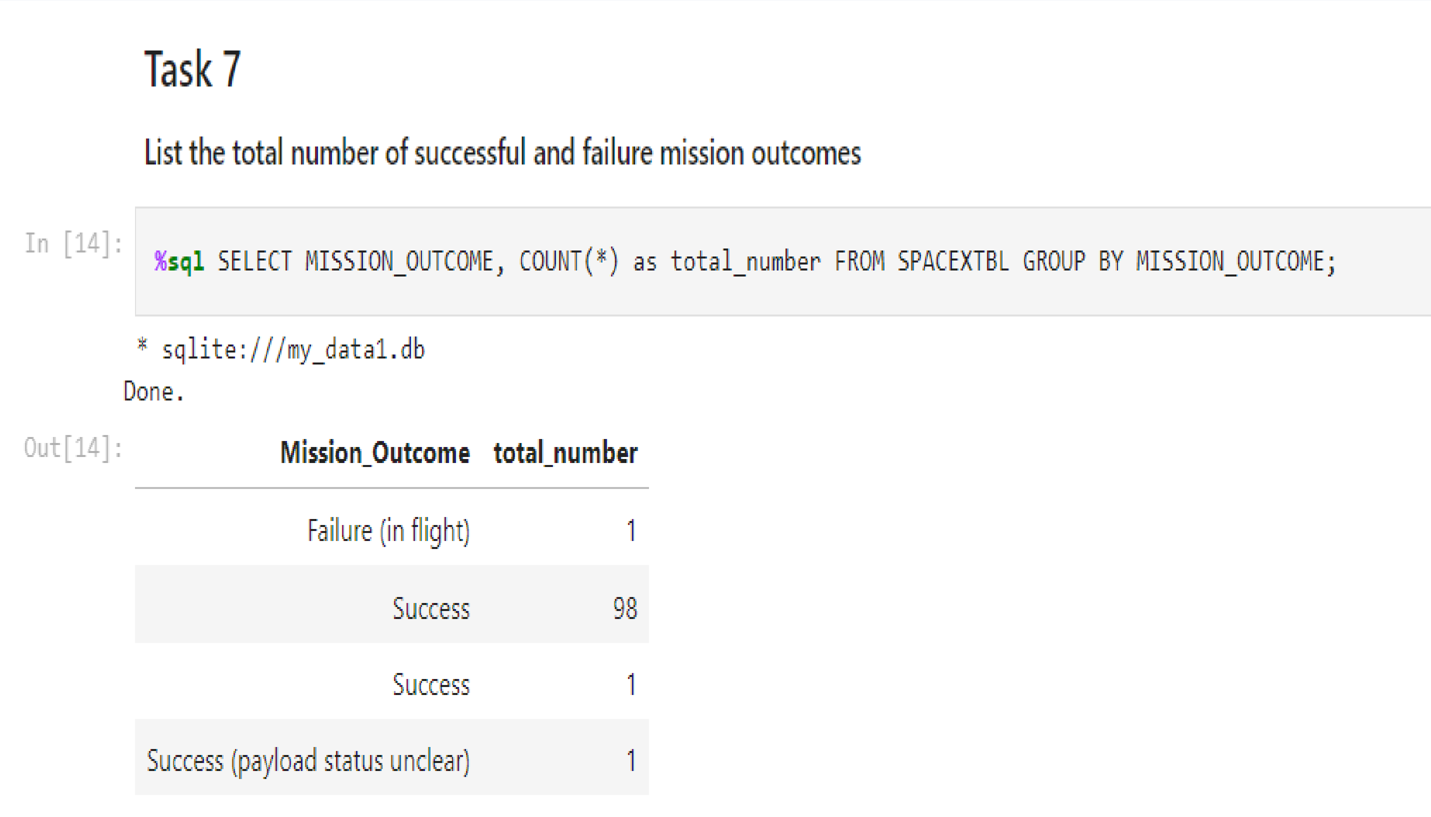
```
 * sqlite:///my_data1.db
Done.
```

Out[13]:

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

- I used the query **WHERE** clause to filter for booster versions which have successfully landed on drone ship and applied the **AND** condition to determine successful landing with payload mass greater than 4000 but less than 6000.

Total Number of Successful and Failure Mission Outcomes



Task 7

List the total number of successful and failure mission outcomes

```
In [14]: %sql SELECT MISSION_OUTCOME, COUNT(*) as total_number FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

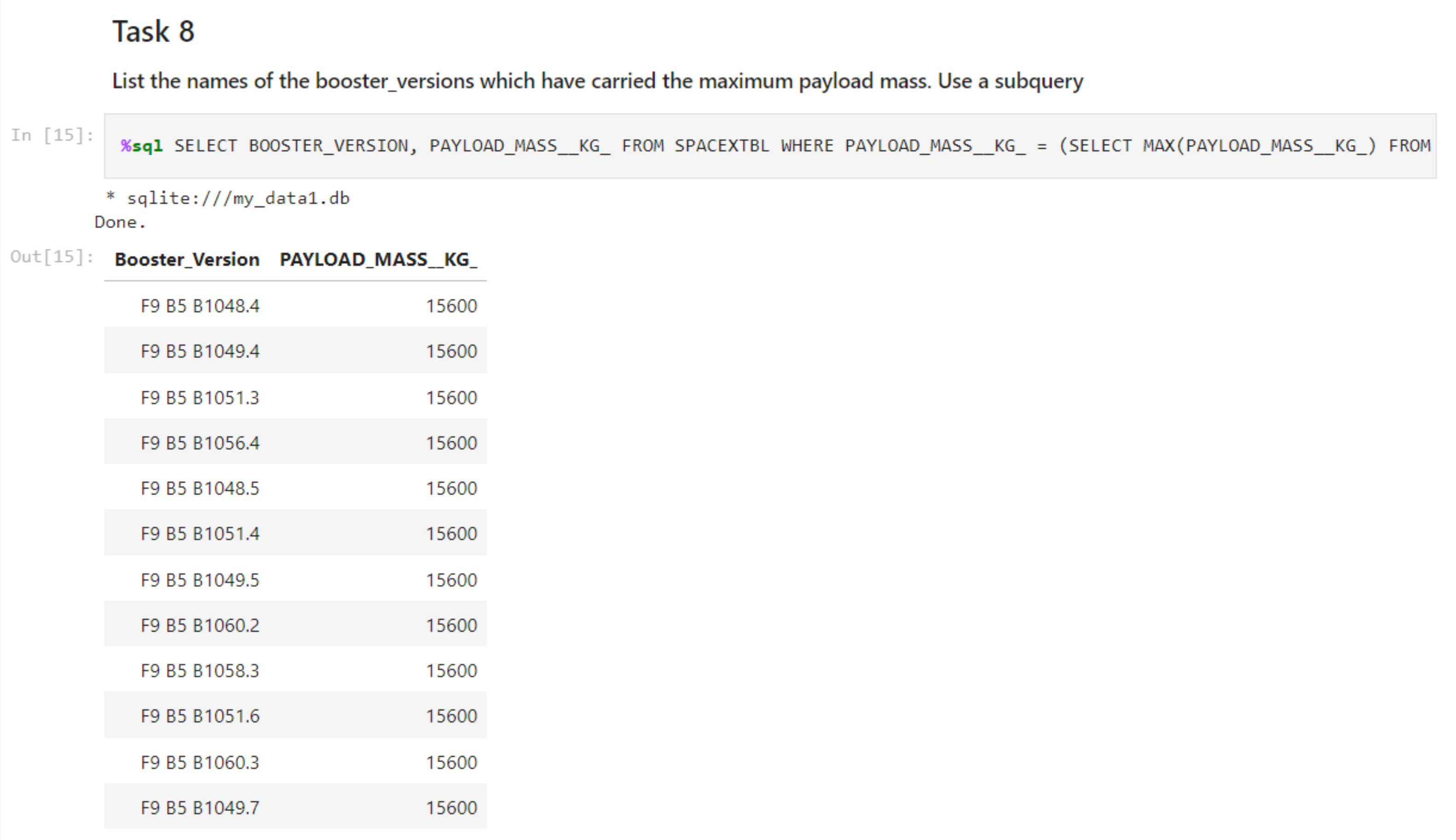
```
 * sqlite:///my_data1.db
Done.
```

Out[14]:

Mission_Outcome	total_number
Failure (in flight)	1
Success	98
Success	1
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



Task 8

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
In [15]:  %sql SELECT BOOSTER_VERSION, PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM
```

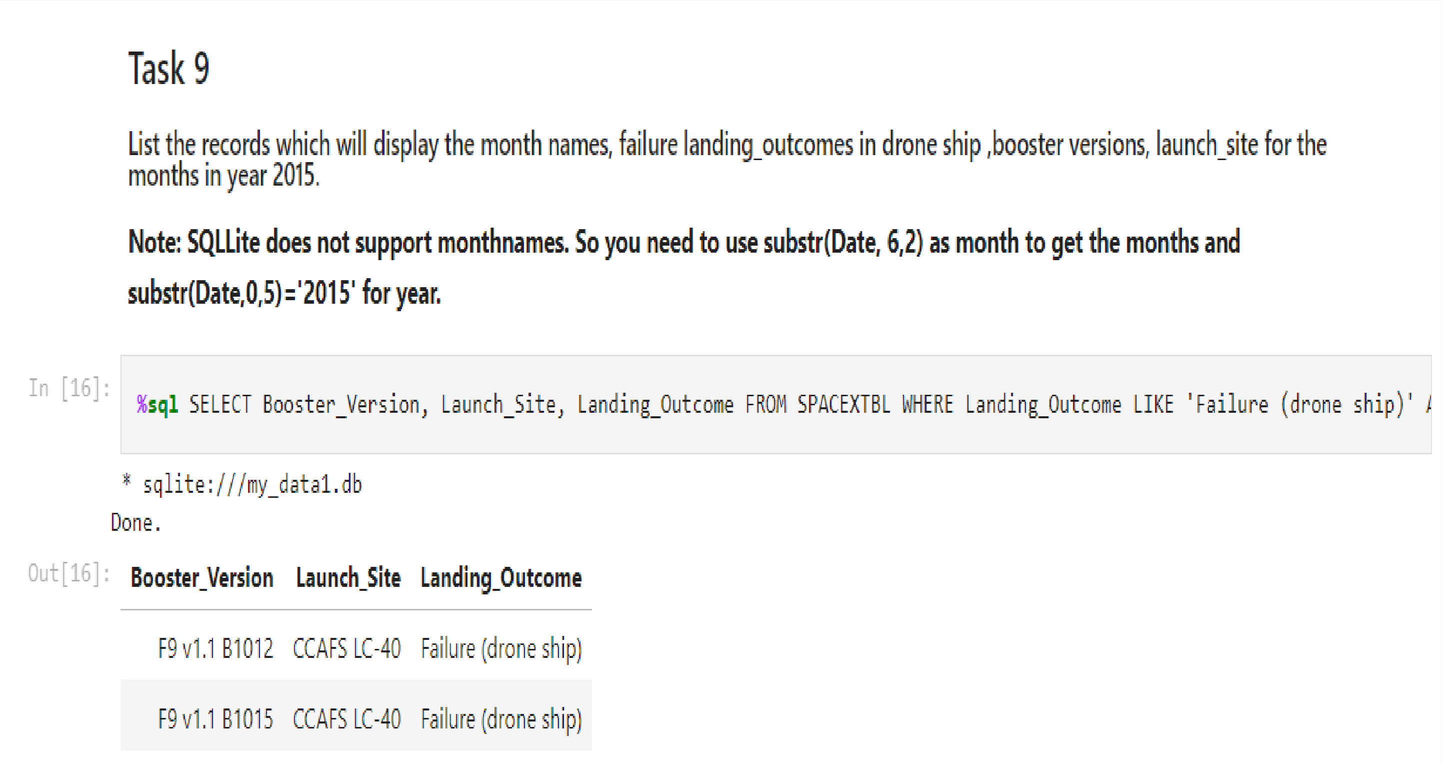
```
 * sqlite:///my_data1.db
Done.
```

Out[15]:

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.
- I determined the booster that have carried the maximum payload using a subquery in the **WHERE** clause and the **MAX()** function.

2015 Launch Records



Task 9

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

In [16]:

```
%sql SELECT Booster_Version, Launch_Site, Landing_Outcome FROM SPACEXTBL WHERE Landing_Outcome LIKE 'Failure (drone ship)' A
```

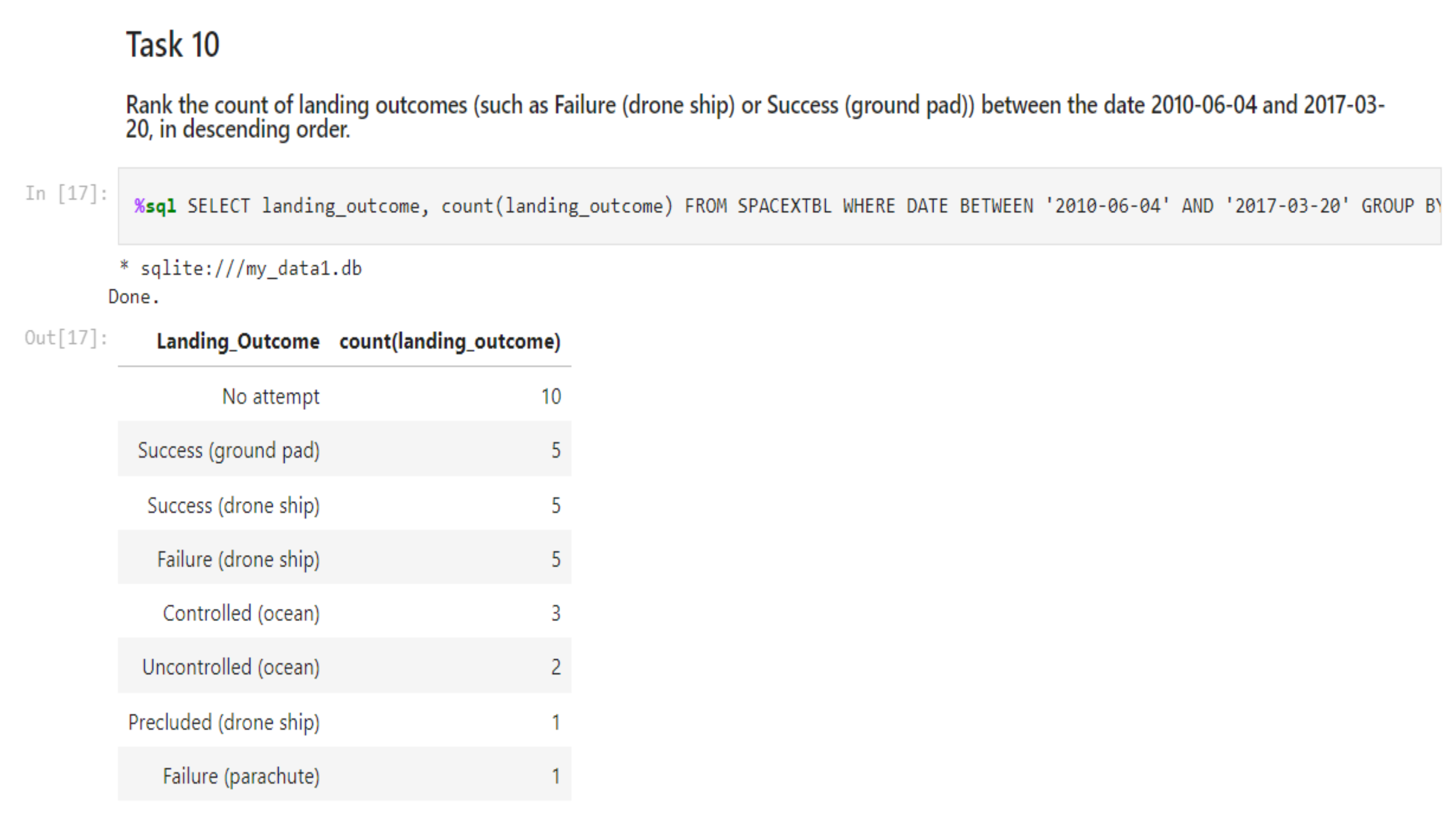
```
 * sqlite:///my_data1.db
Done.
```

Out[16]:

Booster_Version	Launch_Site	Landing_Outcome
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

- I used a combinations of the **WHERE** clause, **LIKE**, **AND**, and **BETWEEN** conditions to filter for failed landing outcomes in drone ship, their booster versions, and launch site names for year 2015
- This query returns the Booster Version, Landing Outcome, and Launch site of 2015 launches where stage 1 failed to land on a drone ship.
- There were two such occurrences.

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



Task 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.

```
In [17]: %sql SELECT landing_outcome, count(landing_outcome) FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY
```

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

Out[17]:

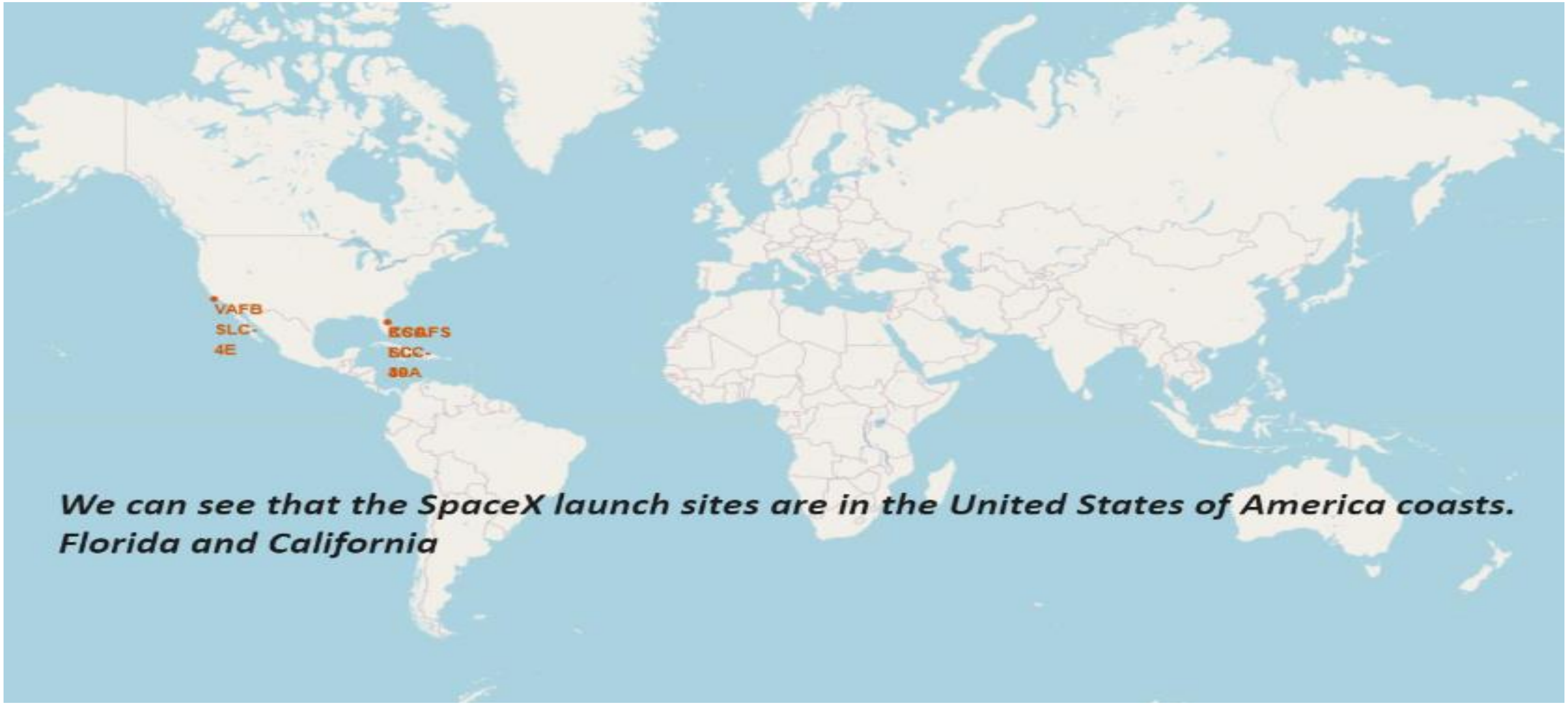
Landing_Outcome	count(landing_outcome)
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

- We selected Landing outcomes and the **COUNT** of landing outcomes from the data and used the **WHERE** clause to filter for landing outcomes **BETWEEN** 2010-06-04 to 2017-03-20.
- We applied the **GROUP BY** clause to group the landing outcomes and the **ORDER BY** clause to order the grouped landing outcome in descending order.

Section 3

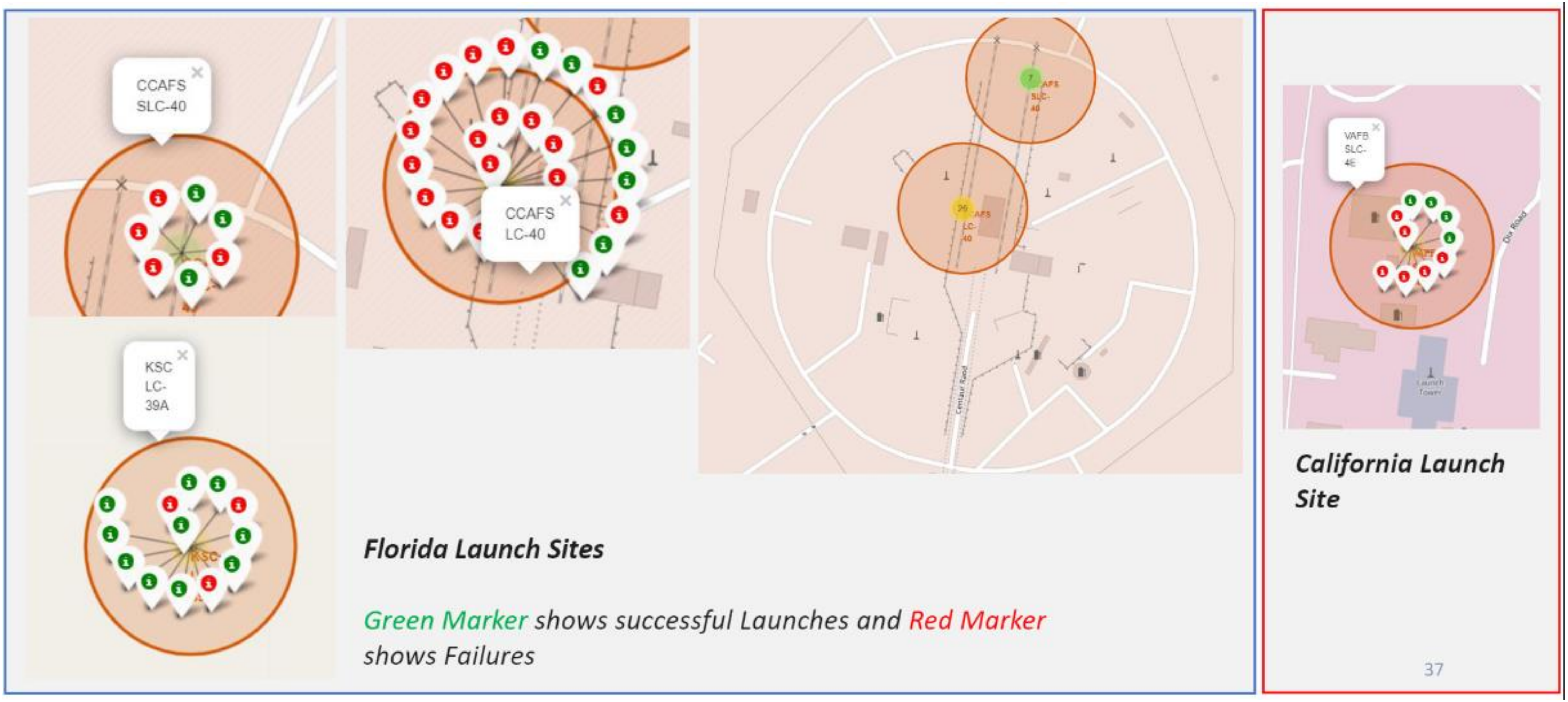
Launch Sites Proximities Analysis

All launch sites global map markers



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

Markers showing launch sites with color labels



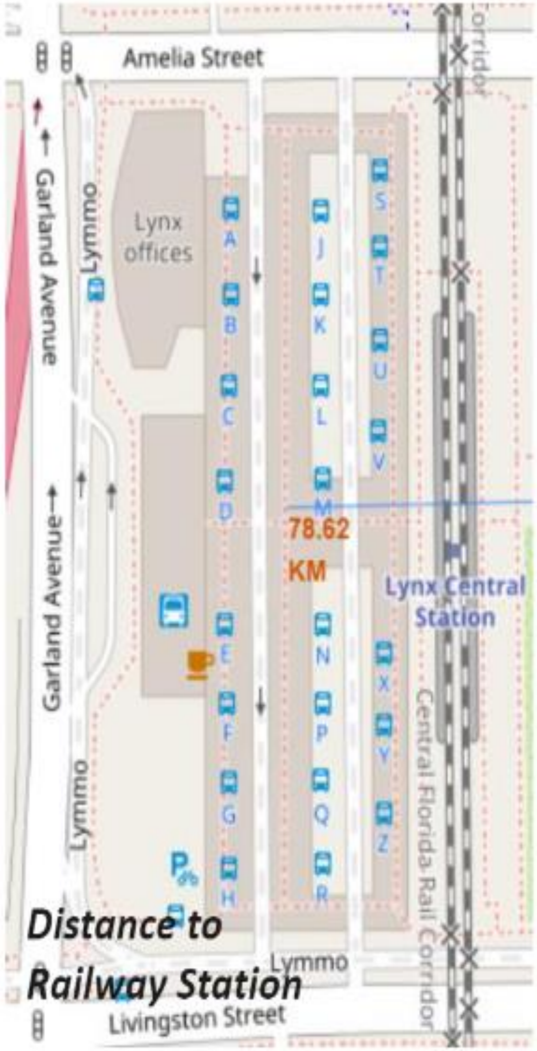
Florida Launch Sites

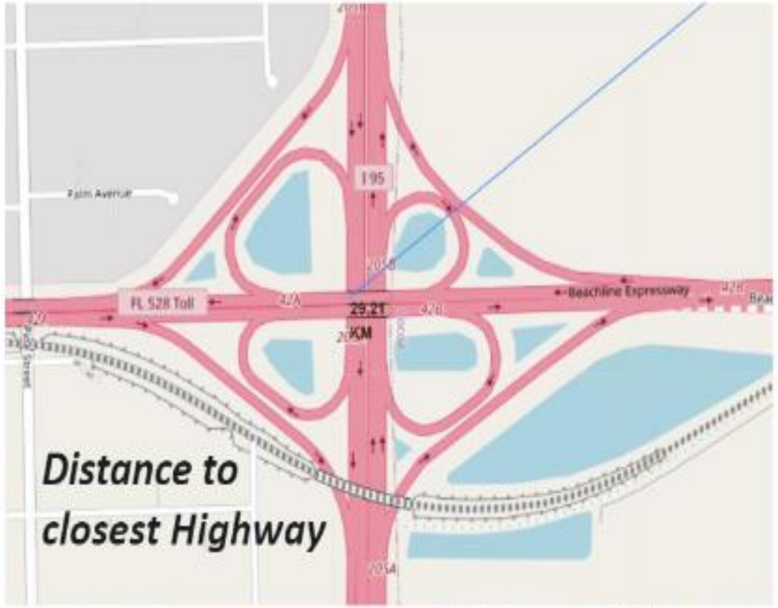
Green Marker shows successful Launches and Red Marker shows Failures

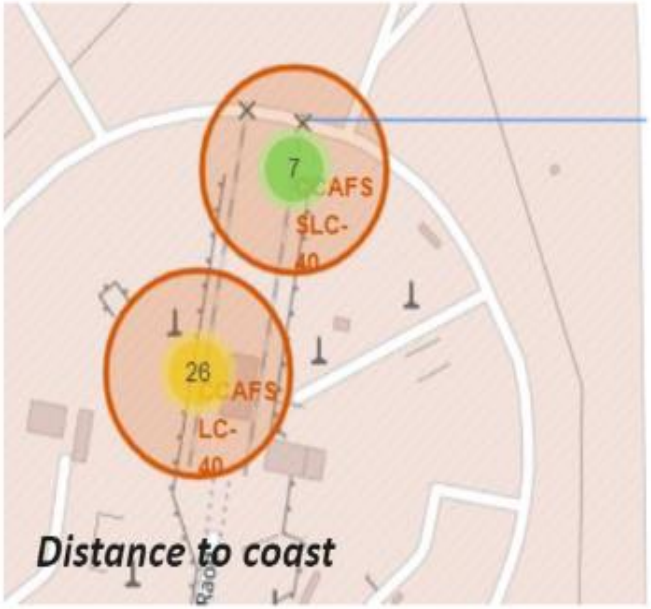
California Launch Site

37

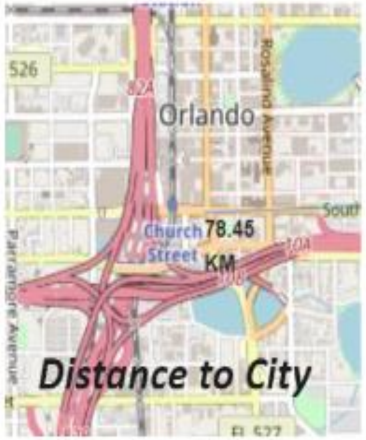
Launch Site distance to landmarks



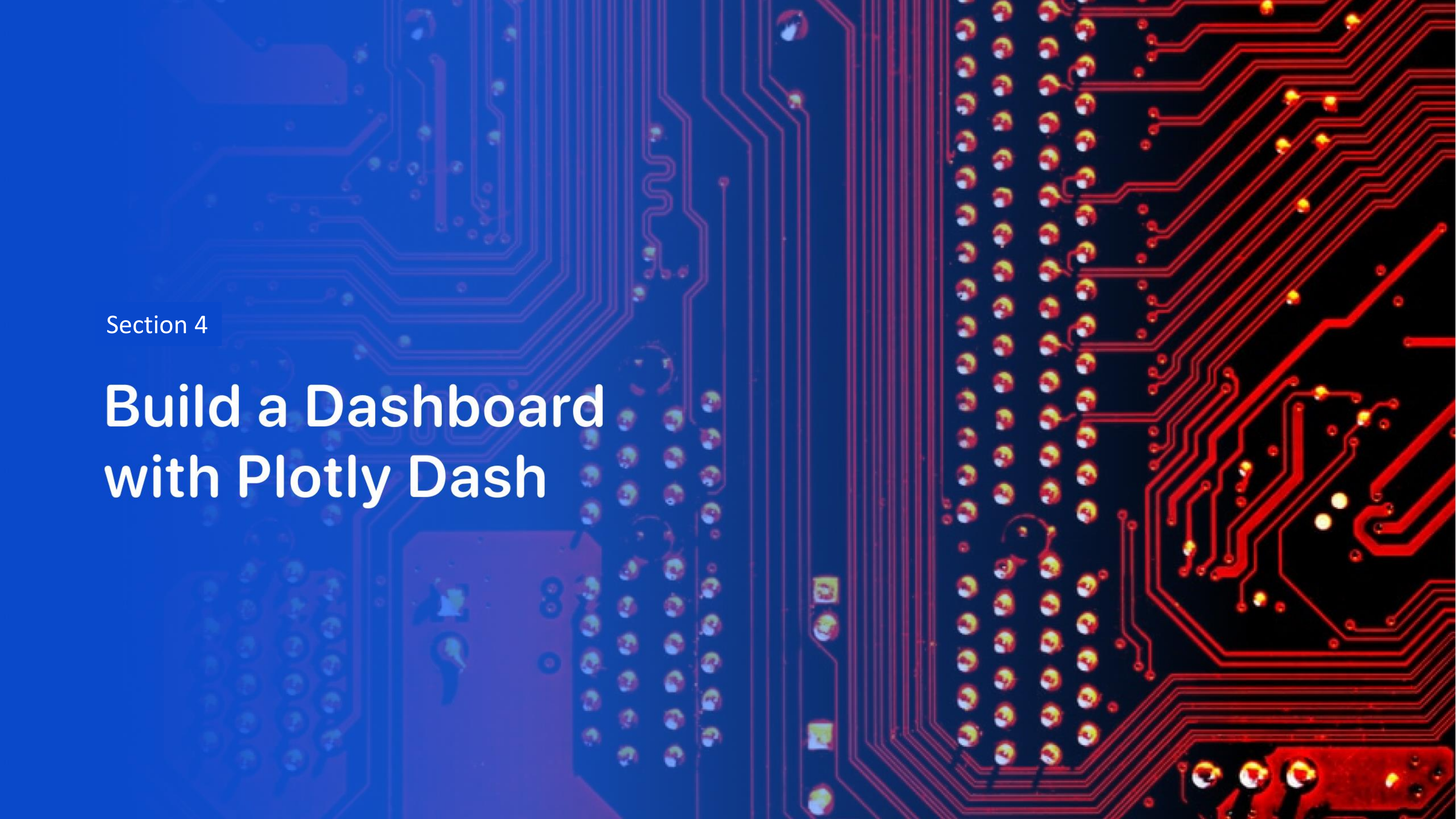




0.90 KM
Distance to Coastline



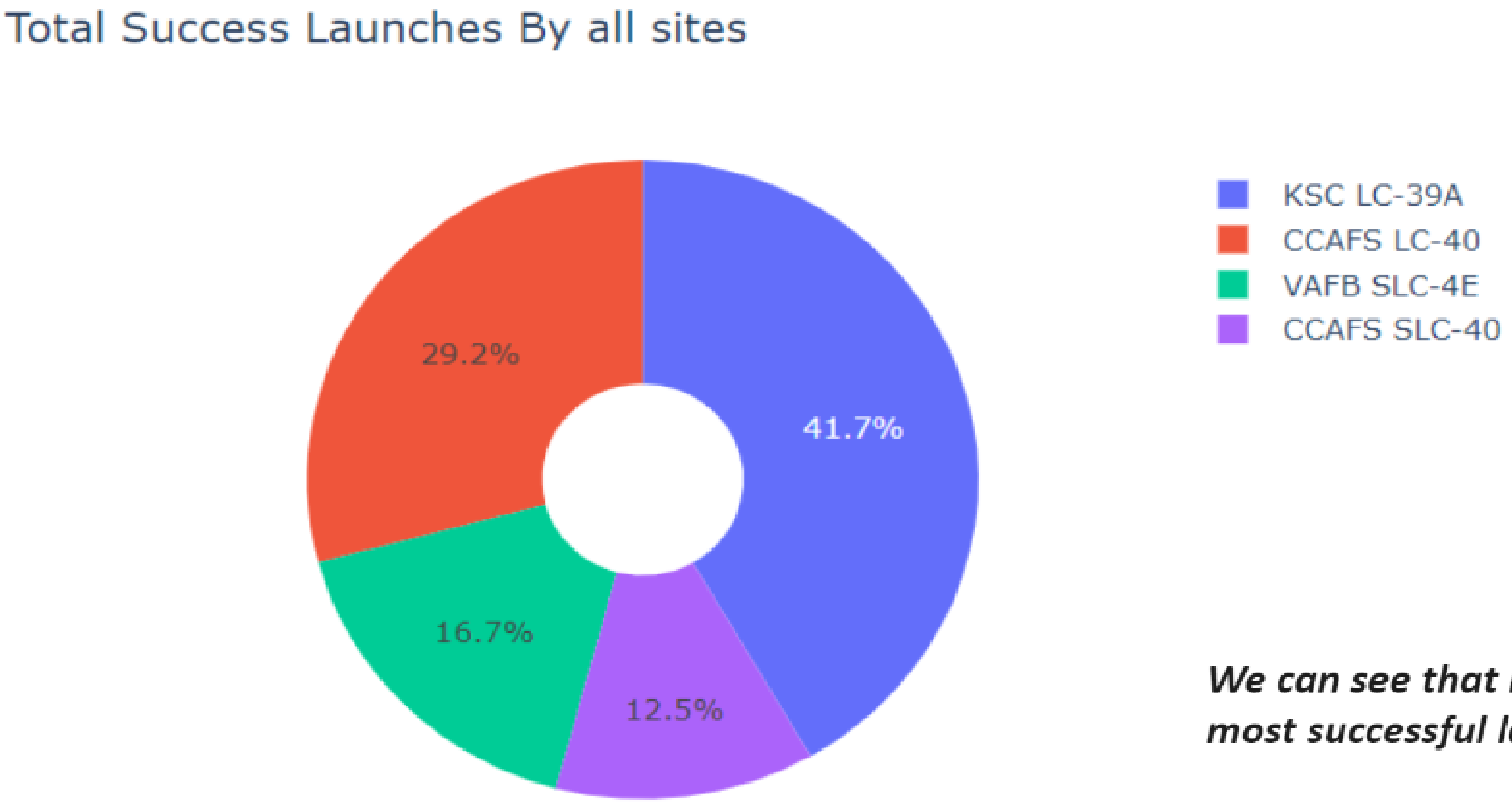
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes



Section 4

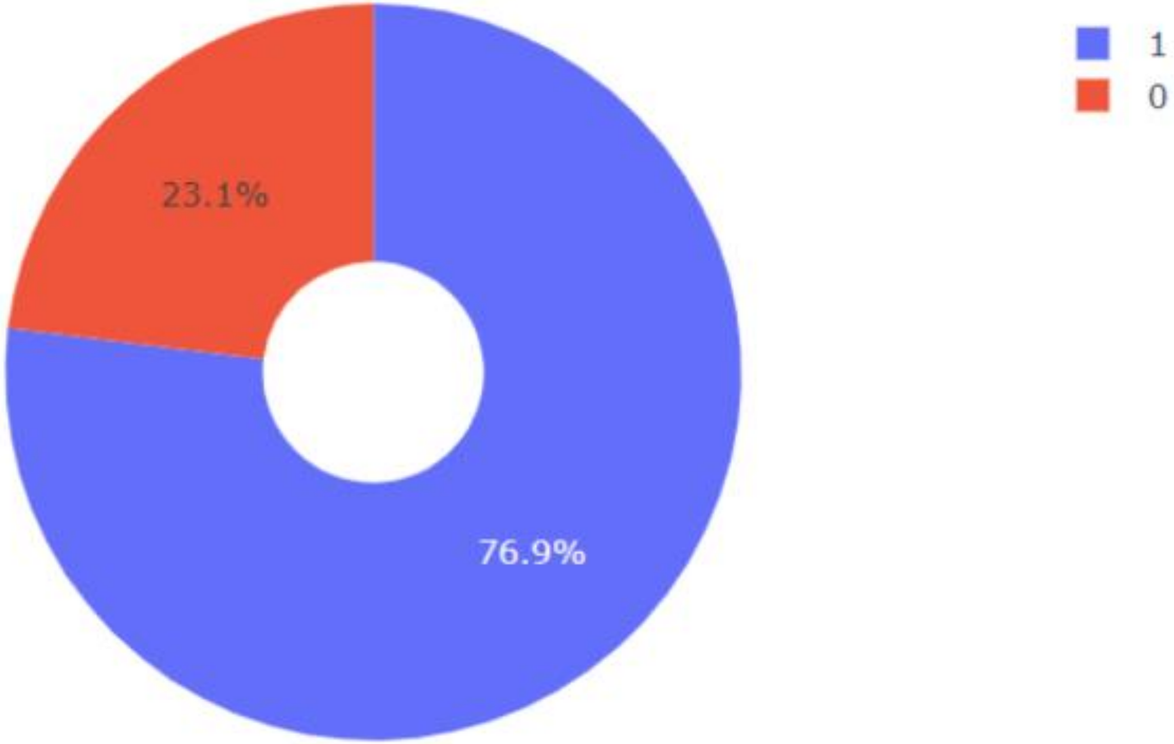
Build a Dashboard with Plotly Dash

Pie chart showing the success percentage achieved by each launch site



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

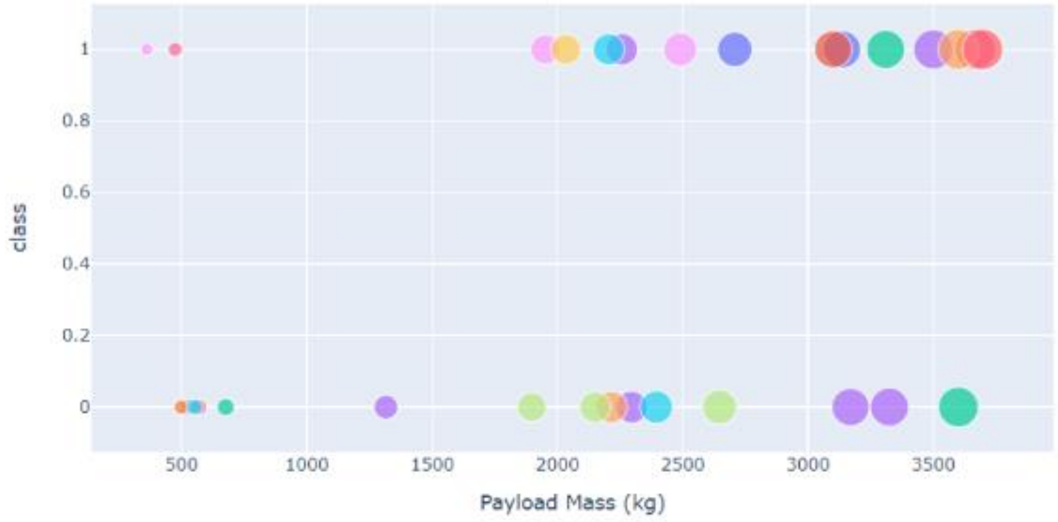
Pie chart showing the Launch site with the highest launch success ratio



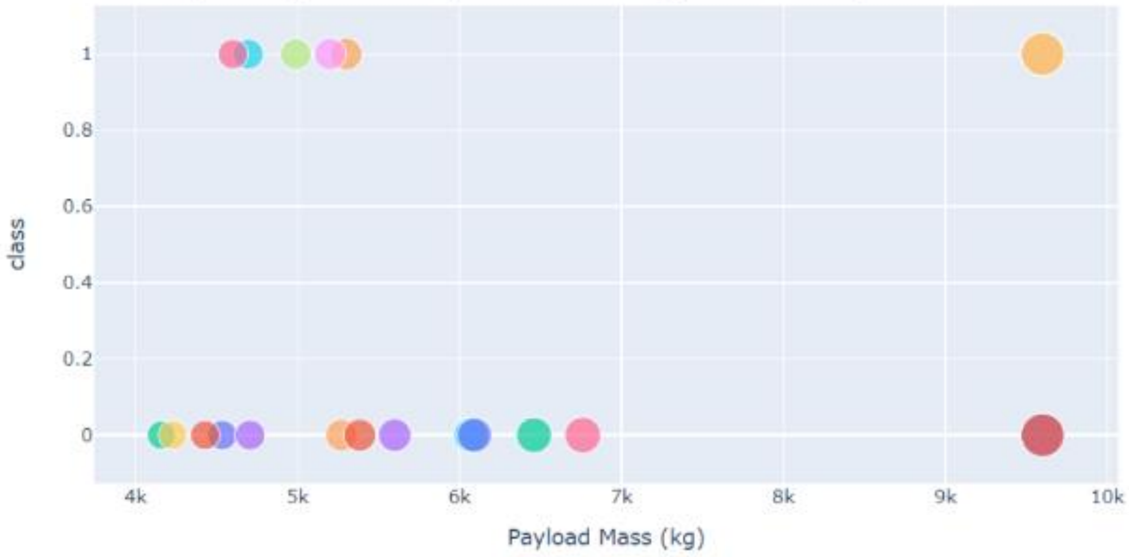
KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Scatter plot of Payload vs Launch Outcome for all sites, with different payload selected in the range slider

Low Weighted Payload 0kg – 4000kg



Heavy Weighted Payload 4000kg – 10000kg



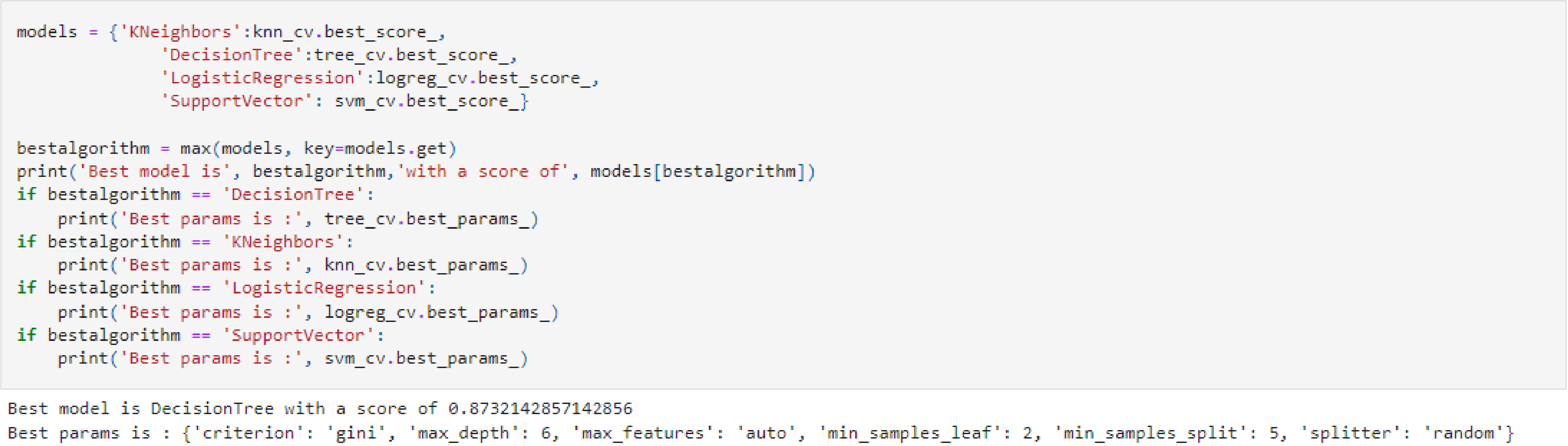
We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The decision tree classifier is the model with the highest classification accuracy



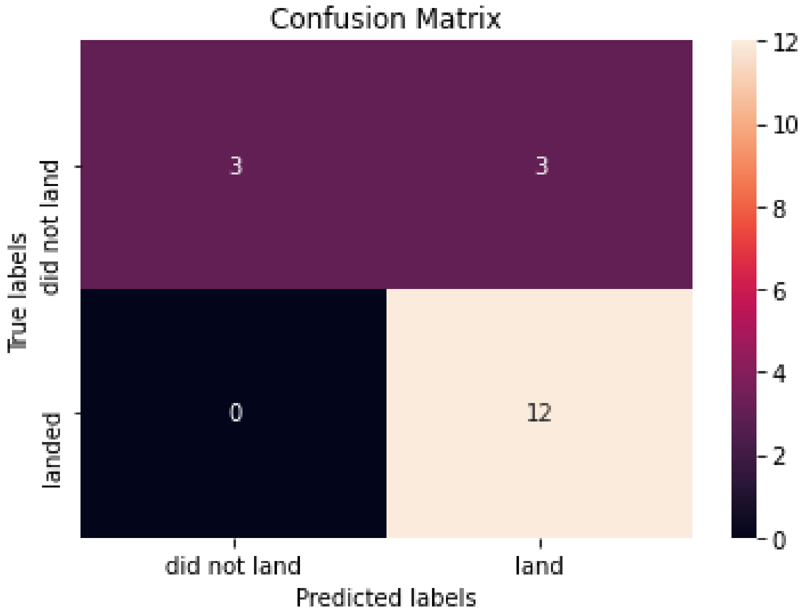
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8732142857142856
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 5, 'splitter': 'random'}
```

Confusion Matrix

- The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.



Conclusions

We can conclude that:

- The larger the flight amount at a launch site, the greater the success rate at a launch site. Launch success rate started to increase in 2013 till 2020. Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate. KSC LC-39A had the most successful launches of any sites. The Decision tree classifier is the best machine learning algorithm for this task.
- 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%
- Allon Mask of 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
- If possible, more data should be collected to better determine the best machine learning model and improve accuracy

Appendix

GitHub repository url:

- https://github.com/Kyahpoots/Applied-Data-Science-Capstone/

Special Thanks to all The Instructors:

- ***Instructors*: Dr. Pooja, Romeo Kienzler, Joseph Santacangelo, Polong Lin, Alex Aklson, Rav Ahuja, Saishruthi Swaminathan, Saeed Aghabozorgi, Hima Vasudevan, Azim Hirjani, Aije Egwaikhide, Yan Luo, Svetlana Levitan**

- https://coursera.org/professional-certificates/ibm-data-science

Thank you!