IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Olaoluwa Ikuesan
August 18, 2022

Outline

Table of Contents

Executive Summary

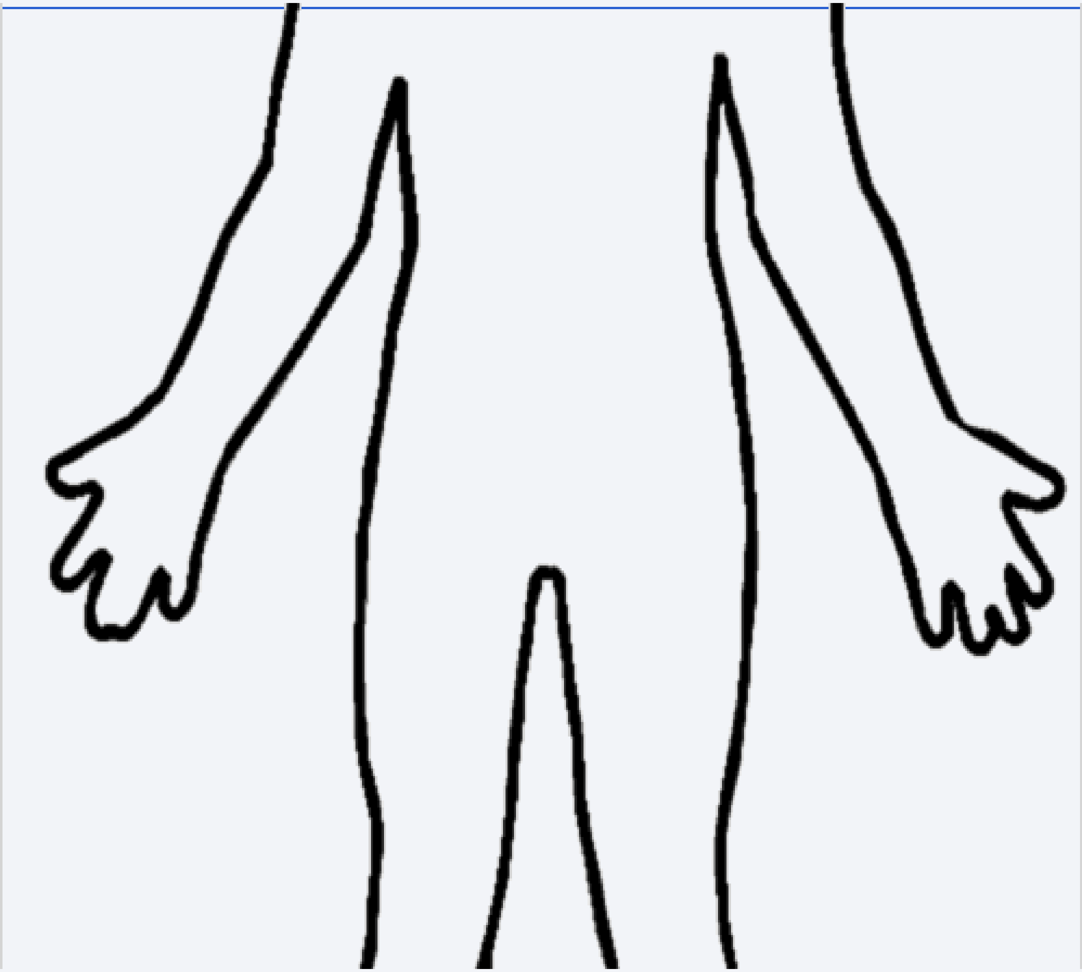
Introduction

Methodology

Results

Conclusion

Appendix



Executive Summary

Summary of Results

- Exploratory Data Analysis Results
- Interactive Analytics Results
- Predictive Analysis Results using Machine Learning

✓ Summary of Methodologies

- Data Collection from SpaceX API and Webscrapping
- Data Cleaning (Wrangling)
- Exploratory Data Analysis with SQL
- EDA using Data Visualization, Charts and Plots
- Building an interactive map with Folium
- Building a Dashboard using Plotly
- Predictive Analysis with Supervised Machine Learning Techniques

Introduction

- Project Background and Context

We predicted if the falcon 9 first stage will land succwssfully. SpaceX advertises falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards of 165 million dollars each, much of the savings is because SpaceX 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 SpaceX for a rocket launch.

- Problems that would be attempted to resolve

✓ What are the factors that influence a successful rocket launch and first stage landing?

✓ Are there factors that would always guarantee a successful first stage landing and the possible drawbacks.

Section 1

Methodology

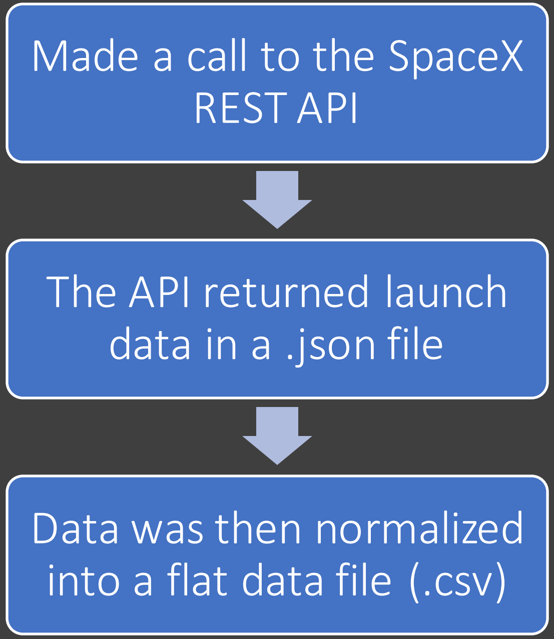
Methodology

- Executive Summary
- Data collection methodology:
  - SpaceX REST API
  - Web Scrapping (from Wikipedia)
- Perform data wrangling
  - Data Cleaning, dropping data irrelevant to the problem and feature engineering
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, Tuning and Evaluating Classification Models

Data Collection

SpaceX REST API

- Launch Data was collected from the SpaceX API. These include information like payload mass, booster version, landing site, launch site and the likes.



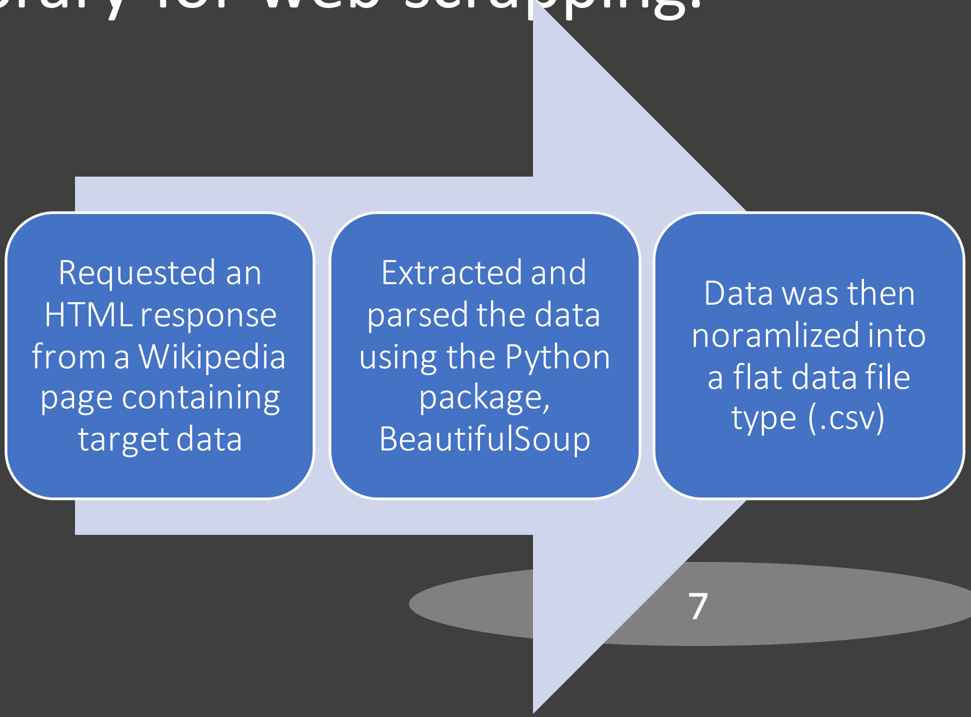
Made a call to the SpaceX REST API

The API returned launch data in a .json file

Data was then normalized into a flat data file (.csv)

Web Scrapping Wikipedia

- Additional Data was scrapped from Wikipedia using the Python Library, BeautifulSoup, a popular library for web scrapping.



Requested an HTML response from a Wikipedia page containing target data

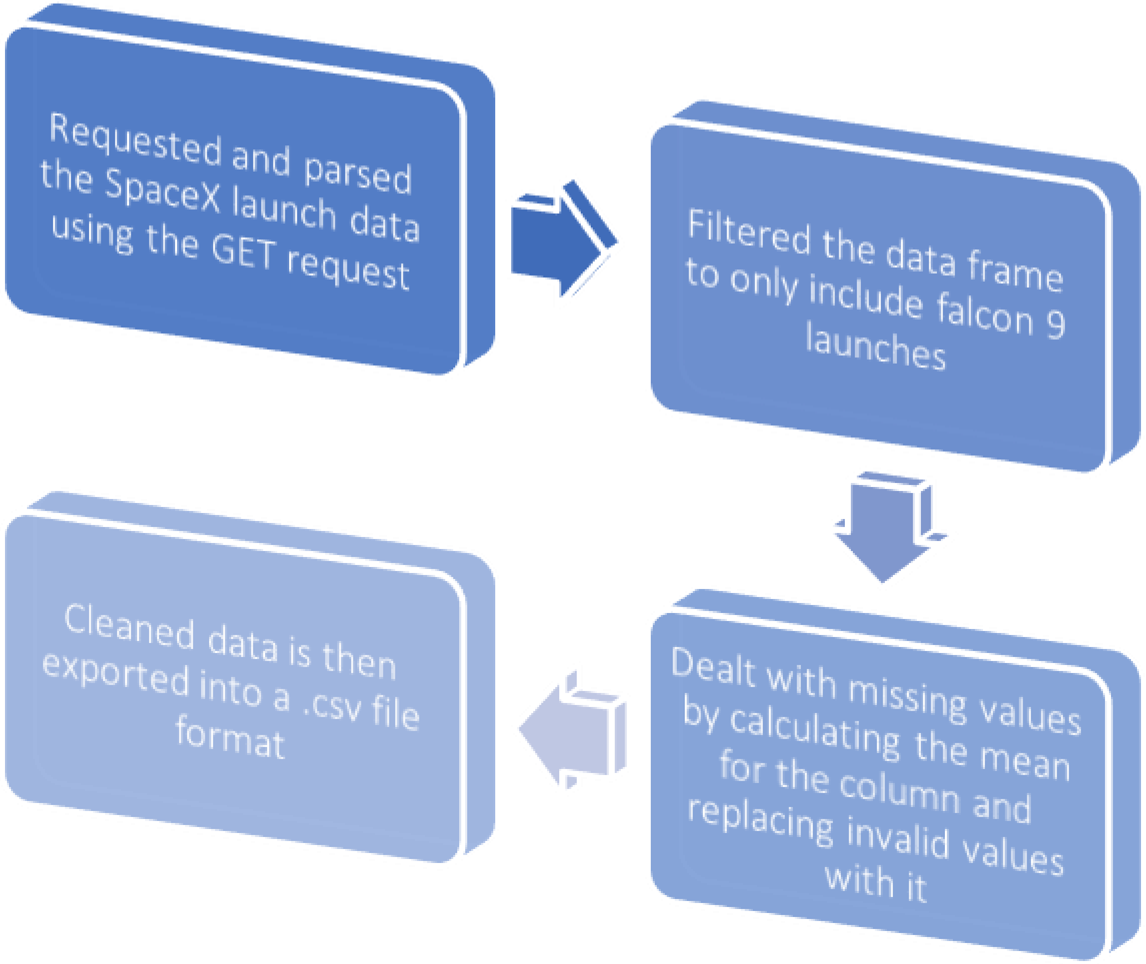
Extracted and parsed the data using the Python package, BeautifulSoup

Data was then noramlized into a flat data file type (.csv)

Data Collection – SpaceX API

API calls were made in a Jupyter Notebook that can be found in the url below

[Link to SpaceX API calls on GitHub]()



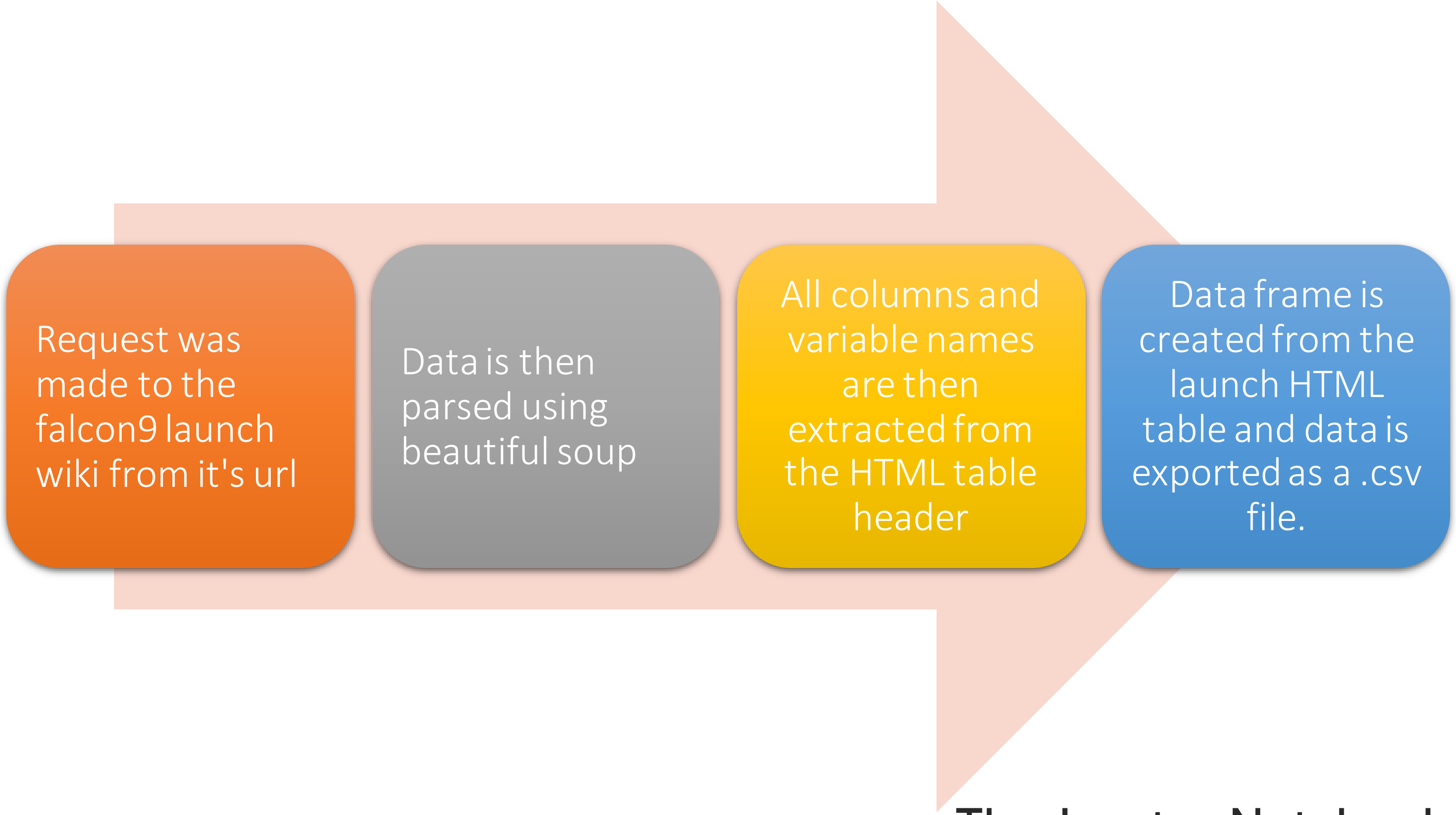
Requested and parsed the SpaceX launch data using the GET request

Filtered the data frame to only include falcon 9 launches

Dealt with missing values by calculating the mean for the column and replacing invalid values with it

Cleaned data is then exported into a .csv file format

Data Collection - Scraping



Request was made to the falcon9 launch wiki from it's url

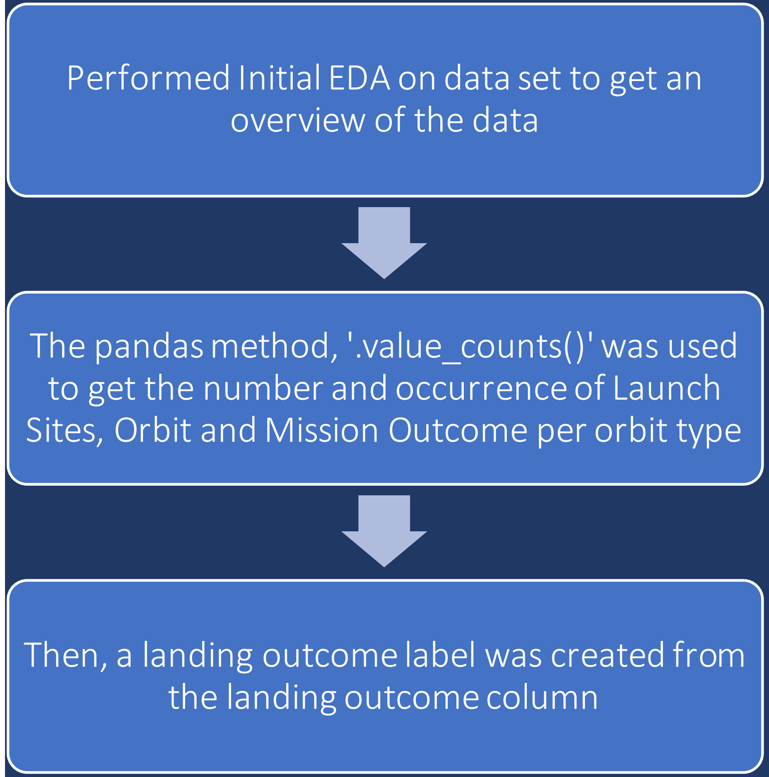
Data is then parsed using beautiful soup

All columns and variable names are then extracted from the HTML table header

Data frame is created from the launch HTML table and data is exported as a .csv file.

- The Jupyter Notebook containing the web scraping process can be found on GitHub through the link below.
- [Web Scraping Process on GitHub]()

Data Wrangling



Performed Initial EDA on data set to get an overview of the data

The pandas method, '.value_counts()' was used to get the number and occurrence of Launch Sites, Orbit and Mission Outcome per orbit type

Then, a landing outcome label was created from the landing outcome column

Data Wrangling Jupyter Notebook on GitHub

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship. We mainly convert those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.

EDA with Data Visualization

- Scatter Plots: Scatter plots were used to show the correlation between two variables. This is done in order to visually get an idea of the variables that affect each other the most. Plots include Flight Number vs Payload, Flight Number vs Launch Site, Orbit vs Flight Number, etc.
- Bar Graphs: A bar graph makes it easy to compare sets of data between different groups at a glance. The graph represents categories on one axis and a discrete value in the other. The goal is to show the relationship between the two axes. Bar charts can also show big changes in data over time. Mean vs Orbit was plotted on a Bar Graph
- Line Plots: Line graphs are useful in that they show data variables and trends very clearly and can help to make predictions about the results of data not yet recorded. Success Rate vs Year was plotted on a Line Graph.

- EDA with Data Visualization on GitHub

EDA with SQL

Queries were made to a database containing the SpaceX data set with the aim of finding out some information about the data before further analysis. The queries made to the database gave us an initial overview of what the data set is about and set us on course for our analysis. The Jupyter Notebook to the SQL queries can be found in the link below.

EDA with SQL Jupyter Notebook on GitHub

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'KSC’
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date where the successful landing outcome in drone ship was achieved.
- Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes
- Listing the names of the booster_versions which have carried the maximum payload mass.
- Listing the records which will display the month names, failed landing outcomes in ground pad, booster versions, launch site in year 2015
- Ranking the count of successful landing outcomes between the date 2010-06-04 and 2017-03-20 in descending order

Build an Interactive Map with Folium

- To Visually interact with our data, we got the location of our Launch Sites in Longitude and Latitude by hovering over the sites in the map and with it, added circle objects to the map, for visibility.

- Markers were then added to the different missions that happened at each site, green for successful mission outcomes and red for failed missions.

- The distance of the various Launch sites to certain Landmarks were calculated. Distance to cities, railway, highway and the coast line was measured and a line was drawn on the map to calculate these distances. We did these to measure patterns and to visually see factors to consider when building a launch site.

- The Jupyter Notebook to the interactive map with Folium can be found

ere.

Build a Dashboard with Plotly Dash

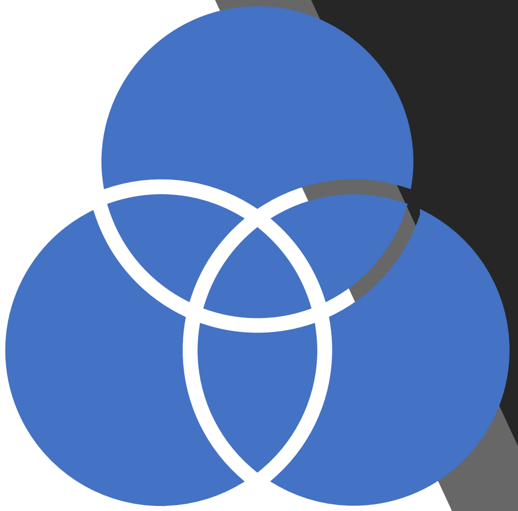
Plotly Python Script on GitHub

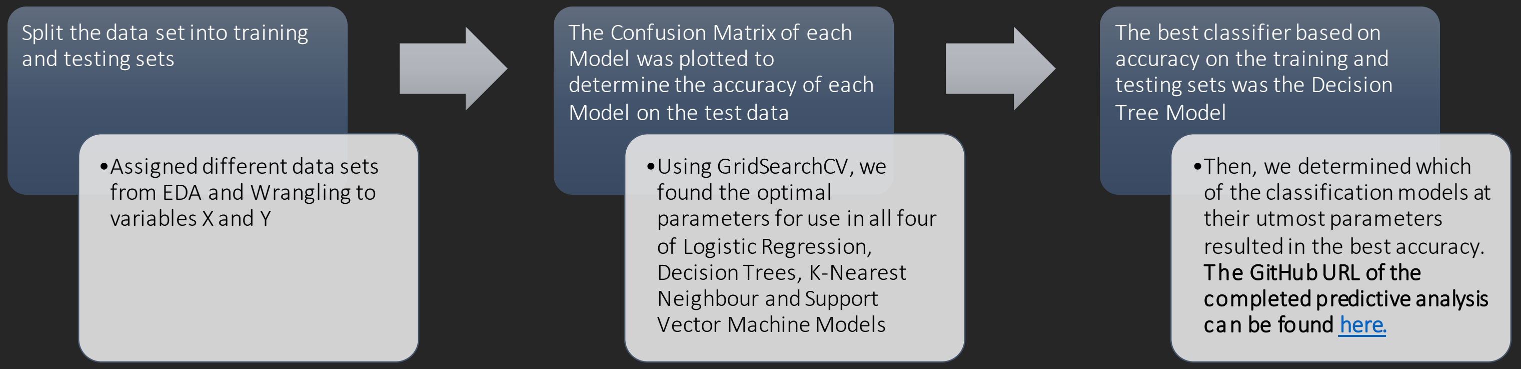
For the interactive Dashboard, a Pie Chart and a Scatter Plot were built

The Pie Chart contains information about the success rate of Launches carried out on the different Launch sites

The Scatter Plot shows the success rate for the different launch sites when compared against the payload mass and the booster version.

Predictive Analysis (Classification)





Split the data set into training and testing sets

- Assigned different data sets from EDA and Wrangling to variables X and Y

The Confusion Matrix of each Model was plotted to determine the accuracy of each Model on the test data

- Using GridSearchCV, we found the optimal parameters for use in all four of Logistic Regression, Decision Trees, K-Nearest Neighbour and Support Vector Machine Models

The best classifier based on accuracy on the training and testing sets was the Decision Tree Model

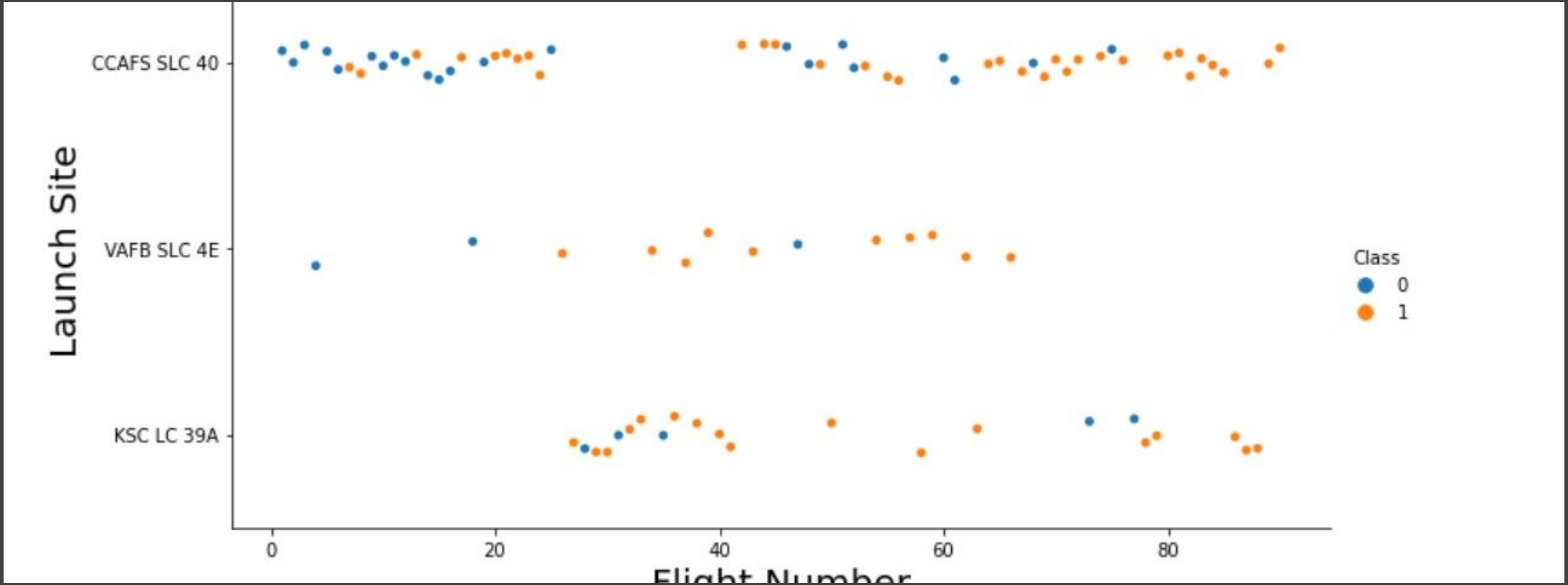
- Then, we determined which of the classification models at their utmost parameters resulted in the best accuracy. The GitHub URL of the completed predictive analysis can be found here.

Results

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

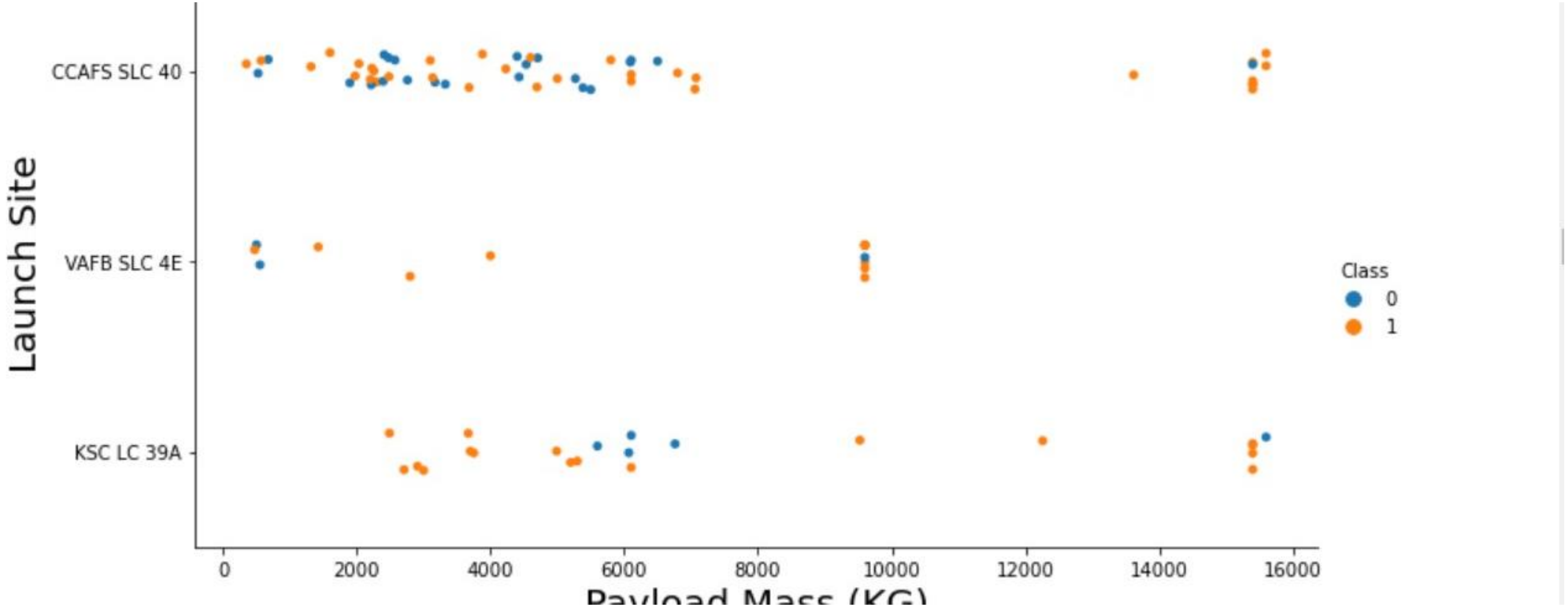
Section 2

Insights drawn from EDA



Flight Number vs. Launch Site

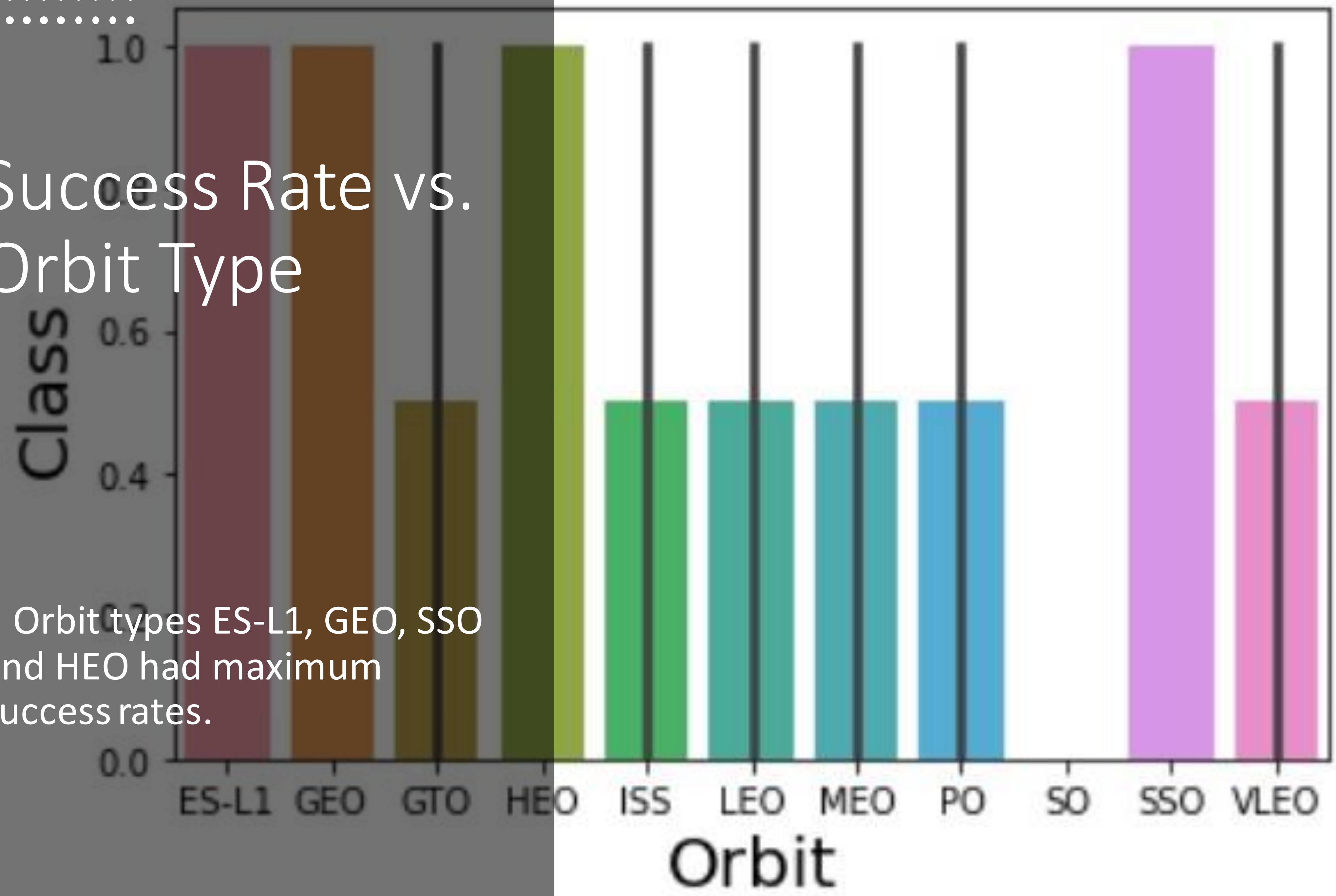
- Flights Numbers larger than twenty(20) started seeing an increase in Success rates. Meanwhile, Launch Site CCAFS SLC-40 was were all but 2 or 3 of the first 20 flights were launched, hence its lower success rate (about 60%) compared to the other two sites



Payload vs. Launch Site

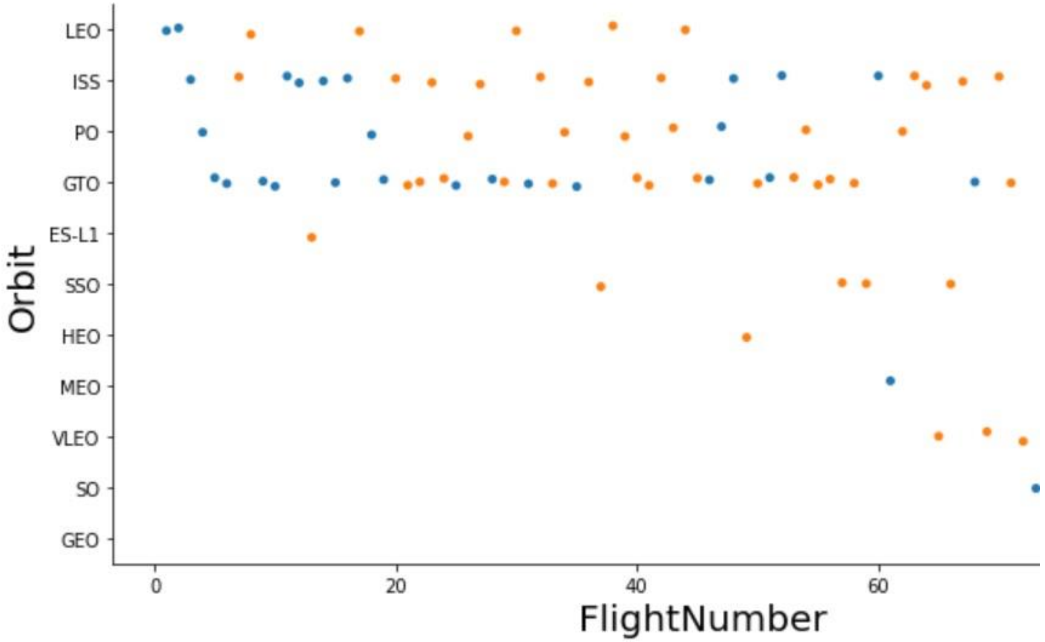
- From the scatter plot, there are no rocket for heavy payload mass at the VAFB SLC 4E launch site. For the other two launch sites, there seemed to be a relatively high success rate for payload mass greater than 10000

Success Rate vs. Orbit Type



- Orbit types ES-L1, GEO, SSO and HEO had maximum success rates.

Flight Number vs. Orbit Type

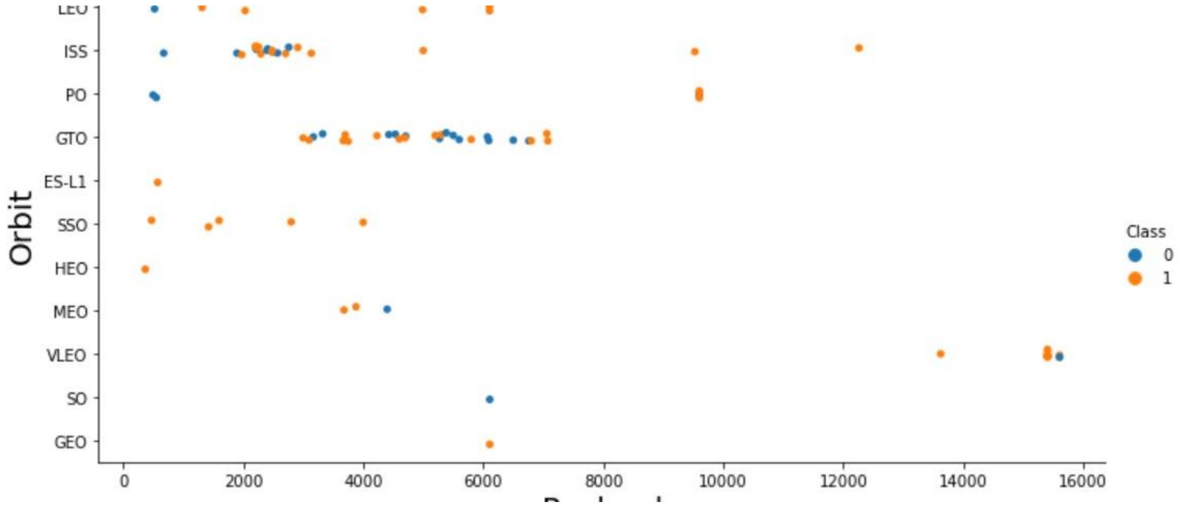


- In the LEO orbit the Success appears related to the number of flights; on the omther hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type

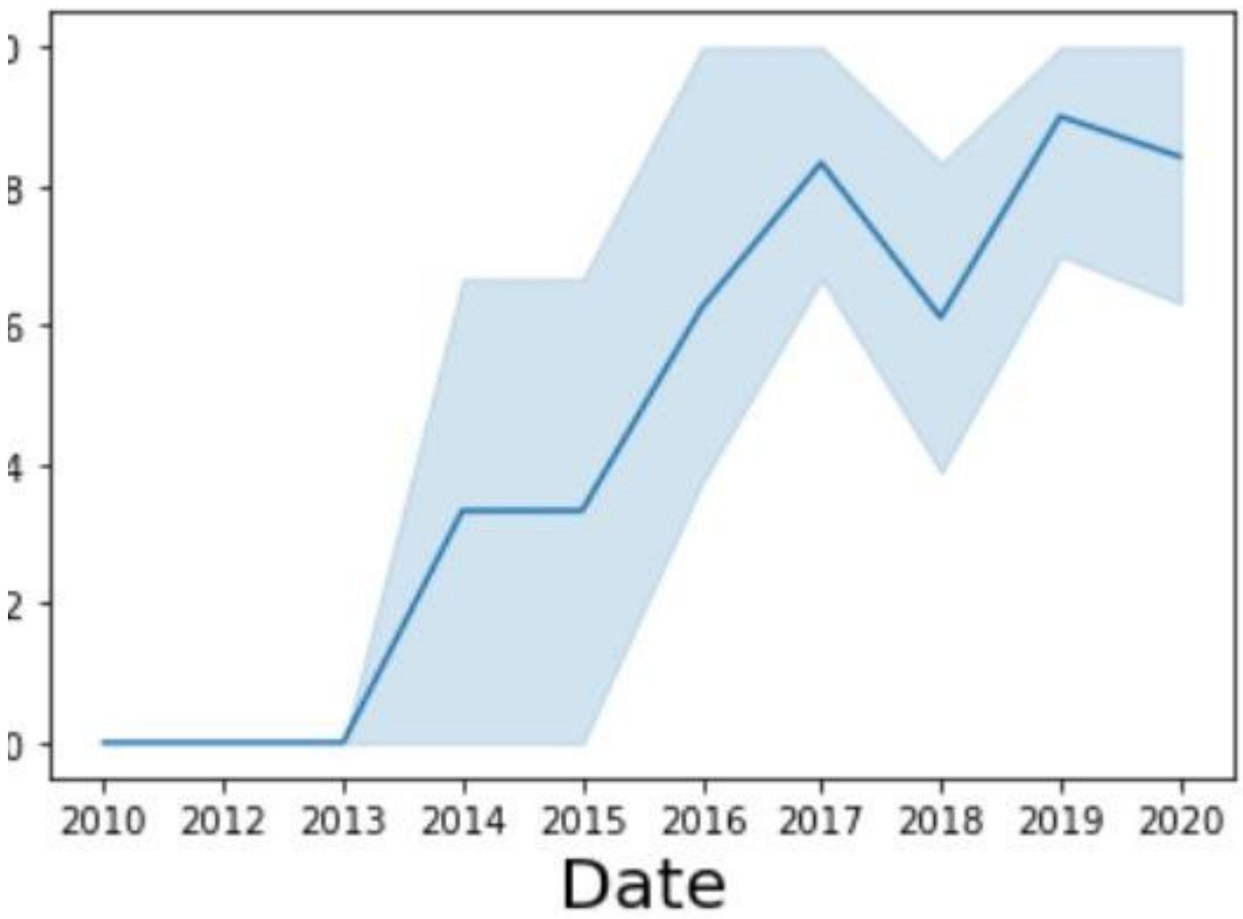
With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

- However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.



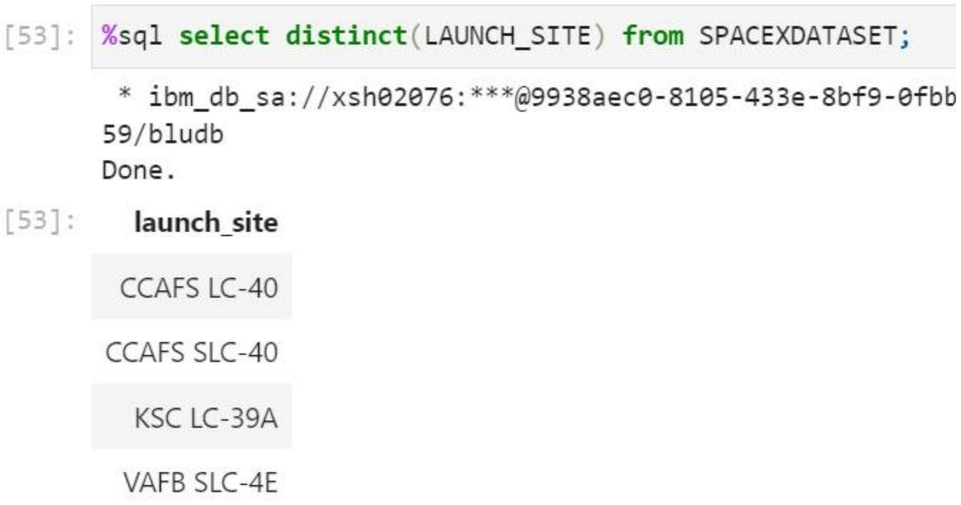
Launch Success Yearly Trend

- The Line plot shows the success rate progression with time. There is an increase in Success rate from year 2013, which halted in 2014 and increased again in 2015. There was a dip in 2017 which took an upward trajectory again in 2018. Generally, success rate kept increasing from 2013 till 2020.



EDA with SQL

All Launch Site Names



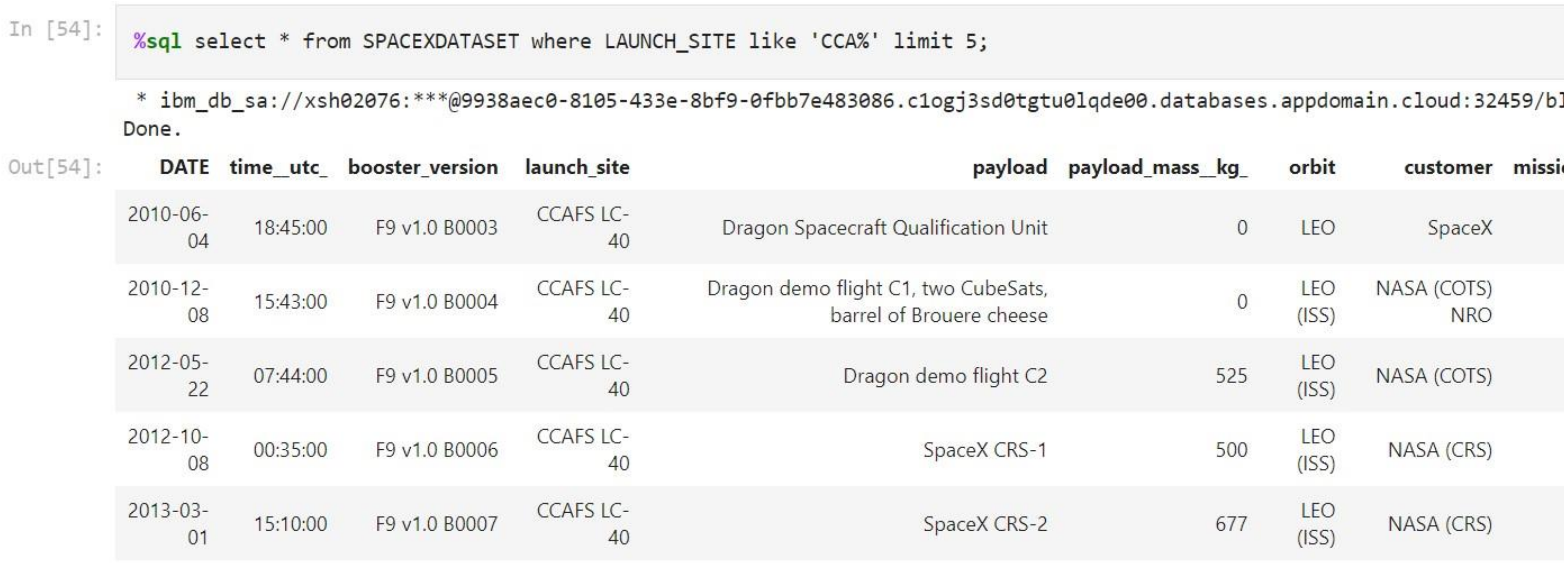
```
[53]: %sql select distinct(LAUNCH_SITE) from SPACEXDATASET;

       * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb
      59/bludb
      Done.
```

[53]:

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

- QUERY EXPLANATION

Using the Keyword 'DISTINCT' displays the unique site names from the launch site column.



```
In [54]:  %sql select * from SPACEXDATASET where LAUNCH_SITE like 'CCA%' limit 5;

 * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:32459/bl
Done.
```

Out[54]:

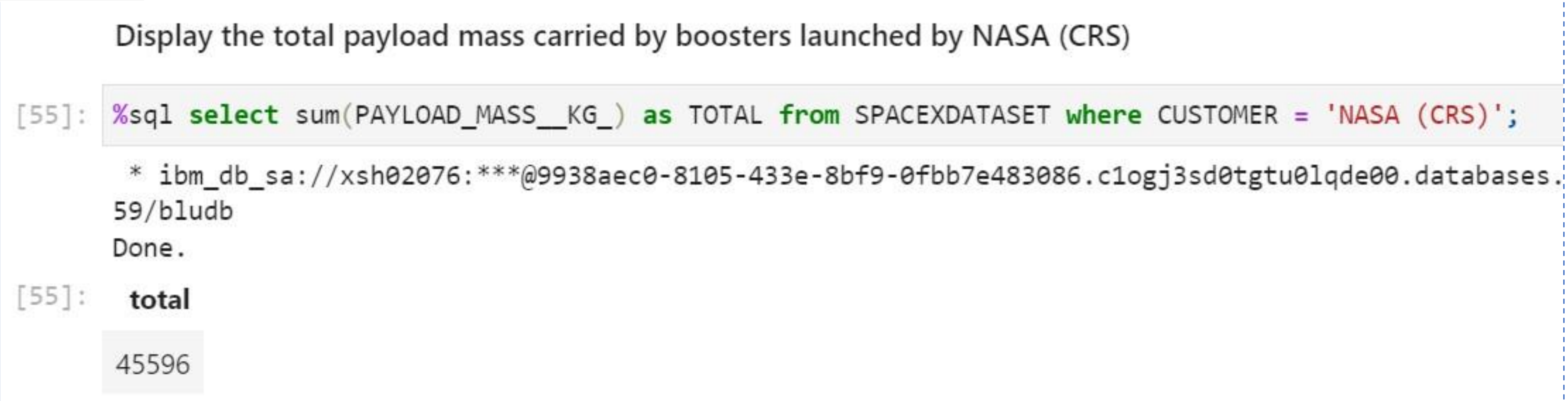
DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	missi
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	

Launch Site Names Begin with 'CCA'

Using the Keyword 'LIKE' and placing the '%' sign in front of CCA tells the query to search for words starting with 'CCA'. Keyword 'LIMIT 5' limits the output to 5.

Total Payload Mass

Using the aggregate function 'SUM' adds up the values of the integer values of the desired column(s). The as keyword creates a new column to store the added values.



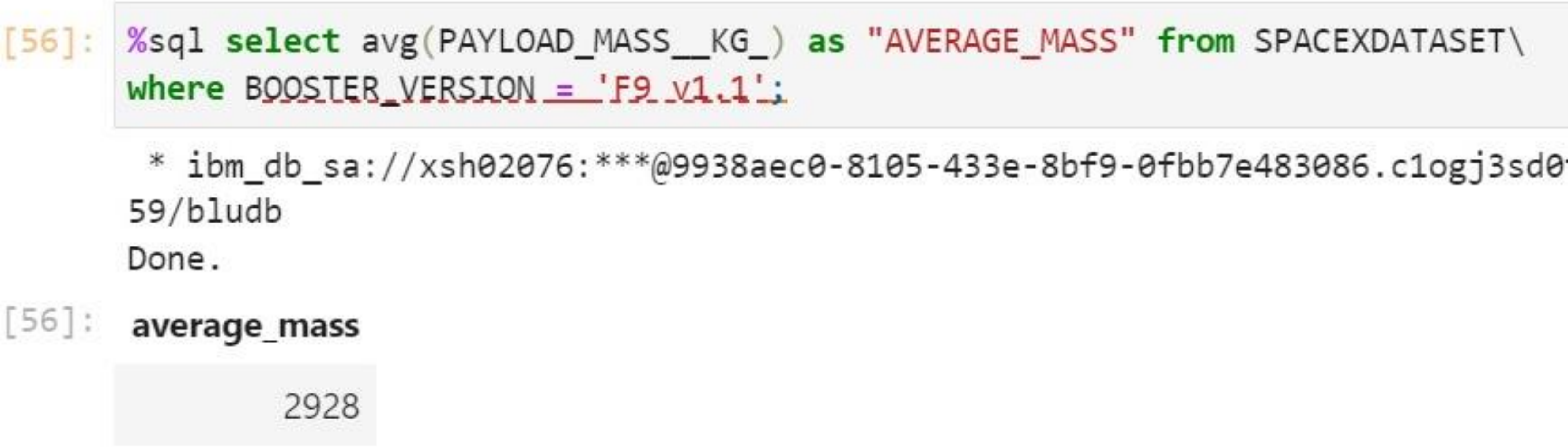
Display the total payload mass carried by boosters launched by NASA (CRS)

```
[55]: %sql select sum(PAYLOAD_MASS__KG_) as TOTAL from SPACEXDATASET where CUSTOMER = 'NASA (CRS)';
```

```
 * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0lqde00.databases.
59/bludb
Done.
```

[55]:

total
45596



```
[56]: %sql select avg(PAYLOAD_MASS__KG_) as "AVERAGE_MASS" from SPACEXDATASET\
      where BOOSTER_VERSION = 'F9 v1.1';
```

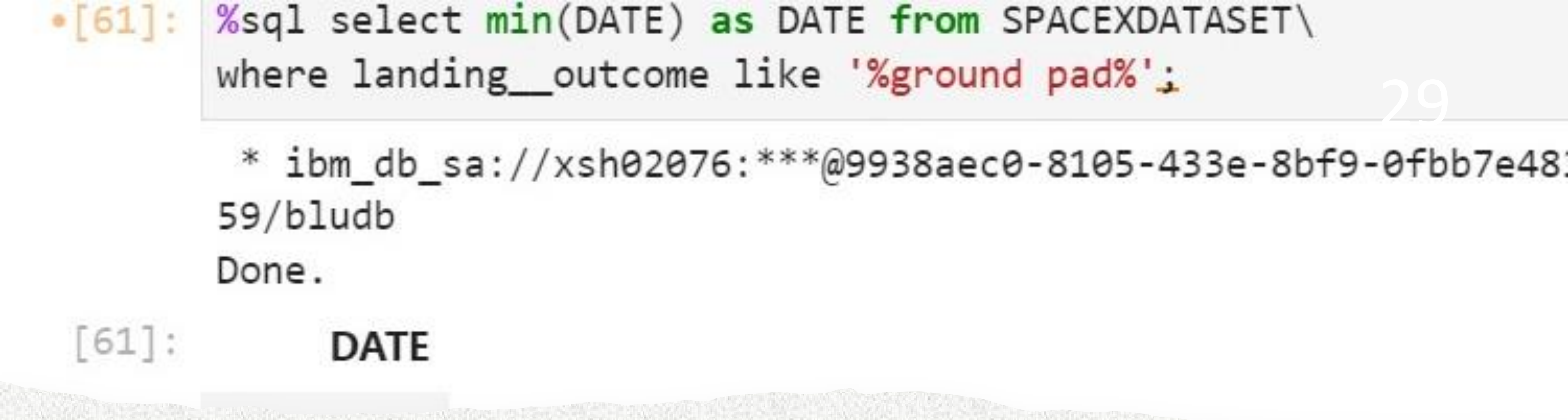
```
 * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0
59/bludb
Done.
```

[56]:

average_mass
2928

Average Payload Mass by F9 v1.1

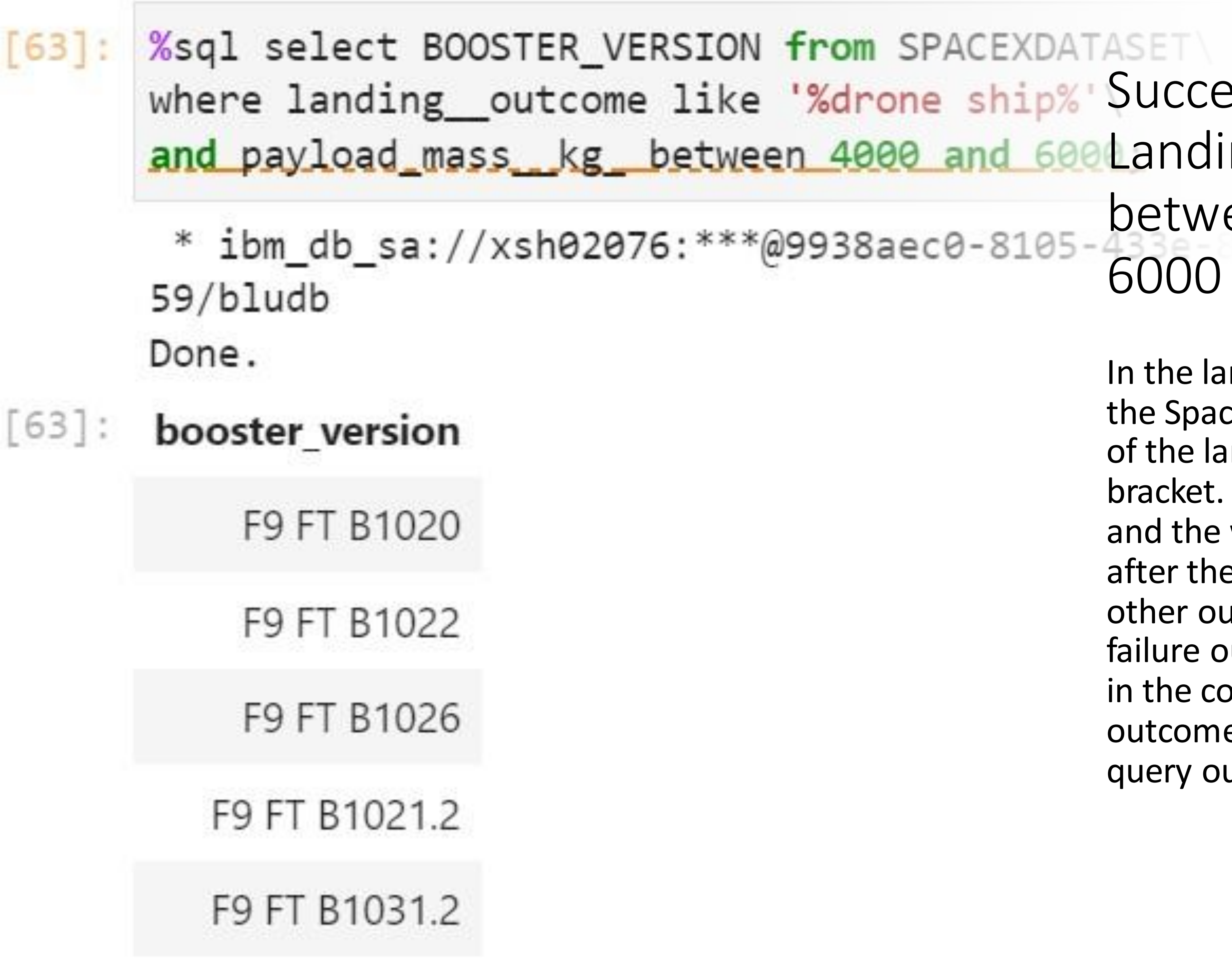
Using the aggregate function 'AVG' outputs the average of the selected column. As before, the 'AS' Keyword creates a new column for the output.



```
•[61]: %sql select min(DATE) as DATE from SPACEXDATASET\
       where landing__outcome like '%ground pad%';

        * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb7e48
       59/bludb
       Done.

[61]:       DATE
```

First Successful Ground Landing Date

Using the aggregate function 'MIN' gets the least, or in this case, earliest date from the column 'DATE' and outputs the condition specified in the 'WHERE' clause.

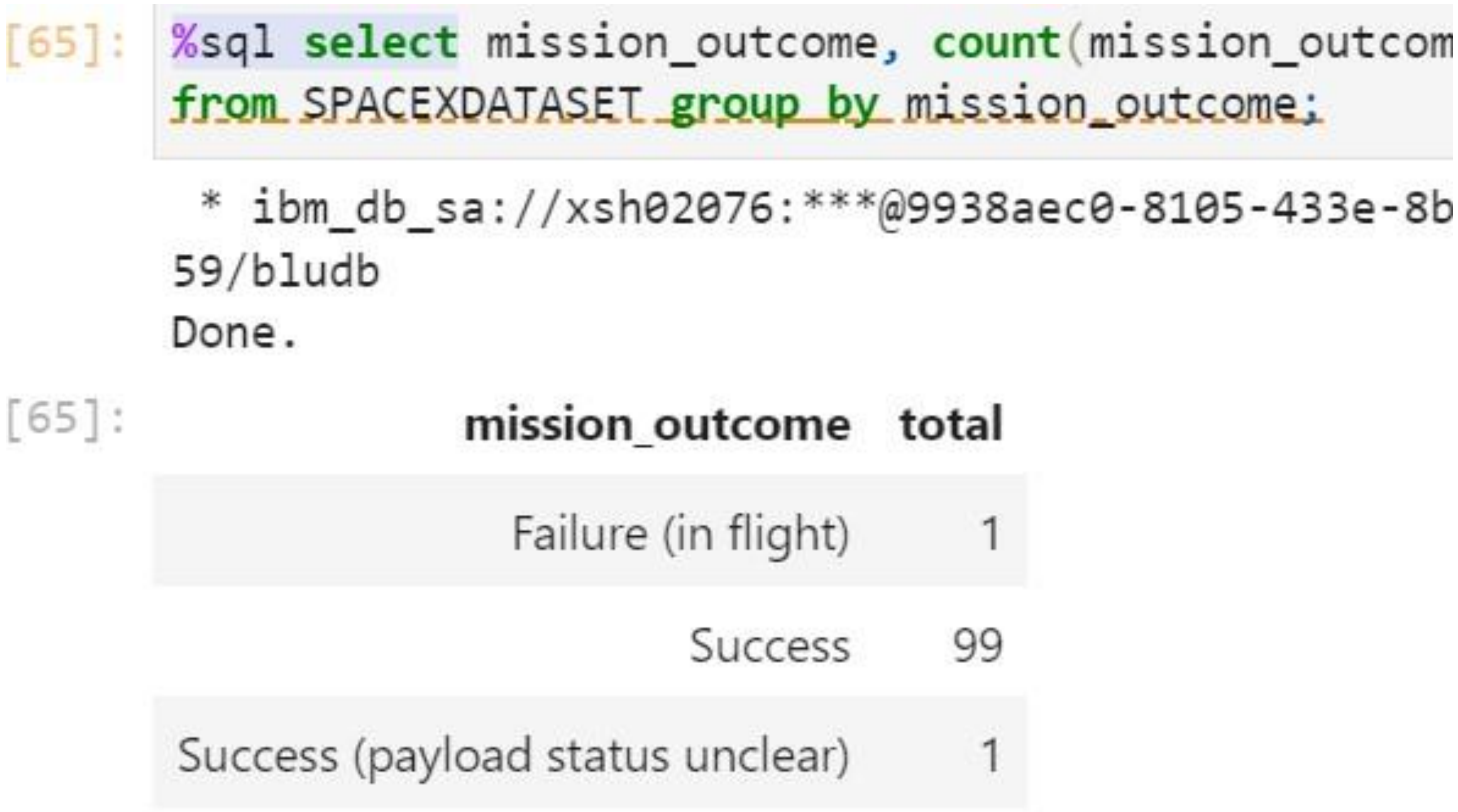


```
[63]: %sql select BOOSTER_VERSION from SPACEXDATASET\
      where landing__outcome like '%drone ship%' \
      and payload_mass__kg_ between 4000 and 6000

       * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8
      59/bludb
      Done.

[63]: booster_version
          F9 FT B1020
          F9 FT B1022
          F9 FT B1026
        F9 FT B1021.2
        F9 FT B1031.2
```

Successful Drone Ship Landing with Payload between 4000 and 6000

In the landing outcome column of the SpaceX Data Set, the location of the landing is specified in a bracket. Using the Keyword 'LIKE' and the wildcard '%' before and after the drone ship filters out al other outcomes. Location of the failure outcome are not specified in the column and as such, failure outcome won't be included in the query output.

Total Number of Successful and Failure Mission Outcomes

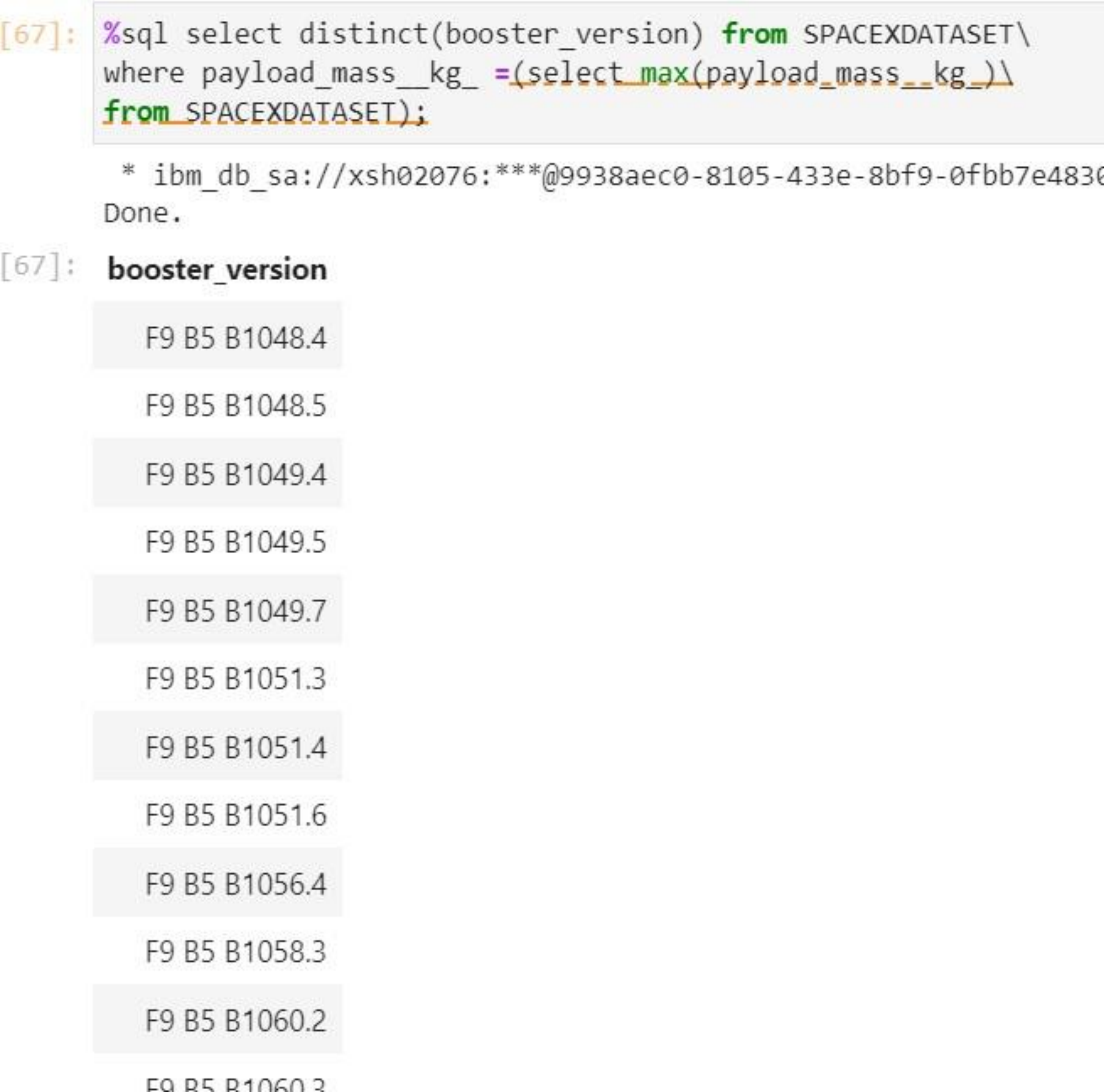


```
[65]: %sql select mission_outcome, count(mission_outcom
      from SPACEXDATASET group by mission_outcome;

       * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8b
      59/bludb
      Done.
```

[65]:

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

The 'COUNT' Keyword counts the distinct number of elements in the selected column, mission_outcome. 'AS' Keyword creates a new column to store the output and the 'GROUP BY' clause does the grouping, in this case, into success and failure outcomes as available in the data set.



```
[67]: %sql select distinct(booster_version) from SPACEXDATASET\
      where payload_mass__kg_ =(select max(payload_mass__kg_)\
      from SPACEXDATASET);

       * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb7e483
      Done.
```

[67]:

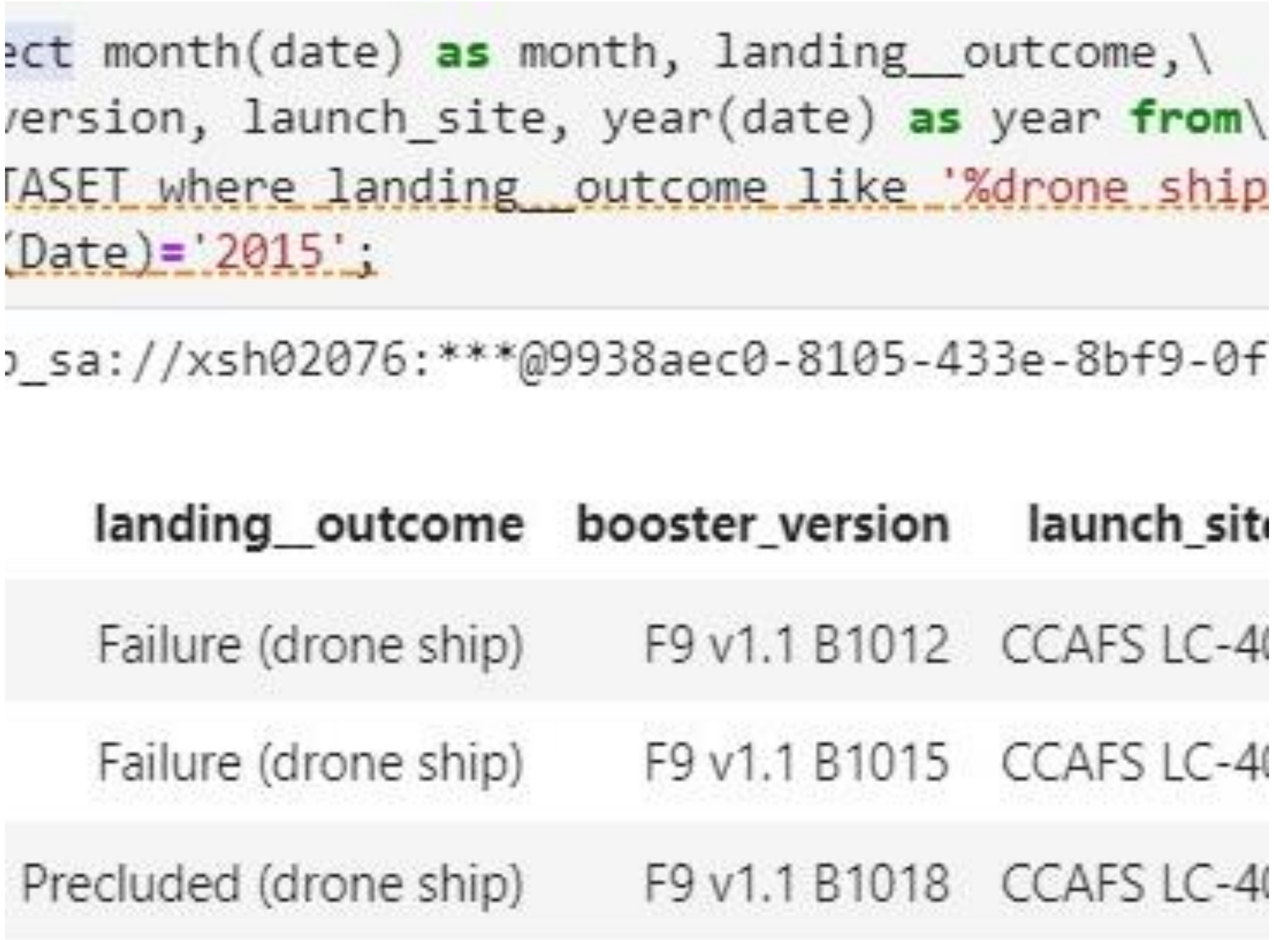
booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

Boosters Carried Maximum Payload

Using the 'DISTINCT' keyword on the column ensures that no booster name is repeated. The aggregate function 'MAX' was used in a nested query.

2015 Launch Records

Using the wildcard '%' filters out failure landing outcome in drone ships. Using the datetime function 'YEAR' selects the year from the date column in the data set and makes it able to issue query on data in a particular year without necessarily having a separate column for year.



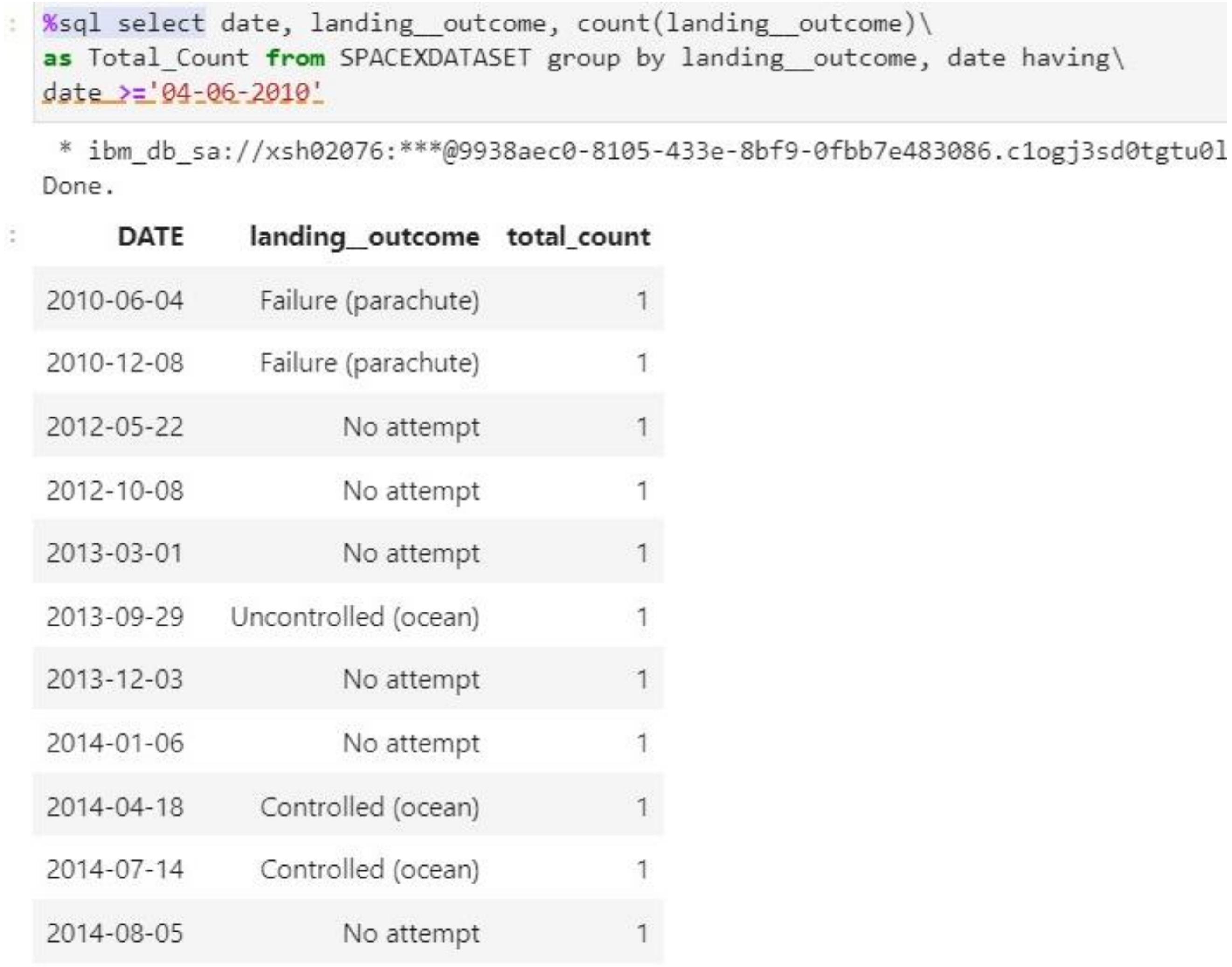
```
ect month(date) as month, landing__outcome,\
ersion, launch_site, year(date) as year from\
TASET where landing__outcome like '%drone ship
(Date)='2015';
```

```
_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0f
```

landing__outcome	booster_version	launch_sit
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-4
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-4
Precluded (drone ship)	F9 v1.1 B1018	CCAFS LC-4

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

Using the keyword 'COUNT' counts the total landing outcome. The keyword 'GROUP BY' groups the distinct landing outcomes and the 'HAVING' clause specifies the date.



```
%sql select date, landing__outcome, count(landing__outcome)\
as Total_Count from SPACEXDATASET group by landing__outcome, date having\
date >='04-06-2010'
```

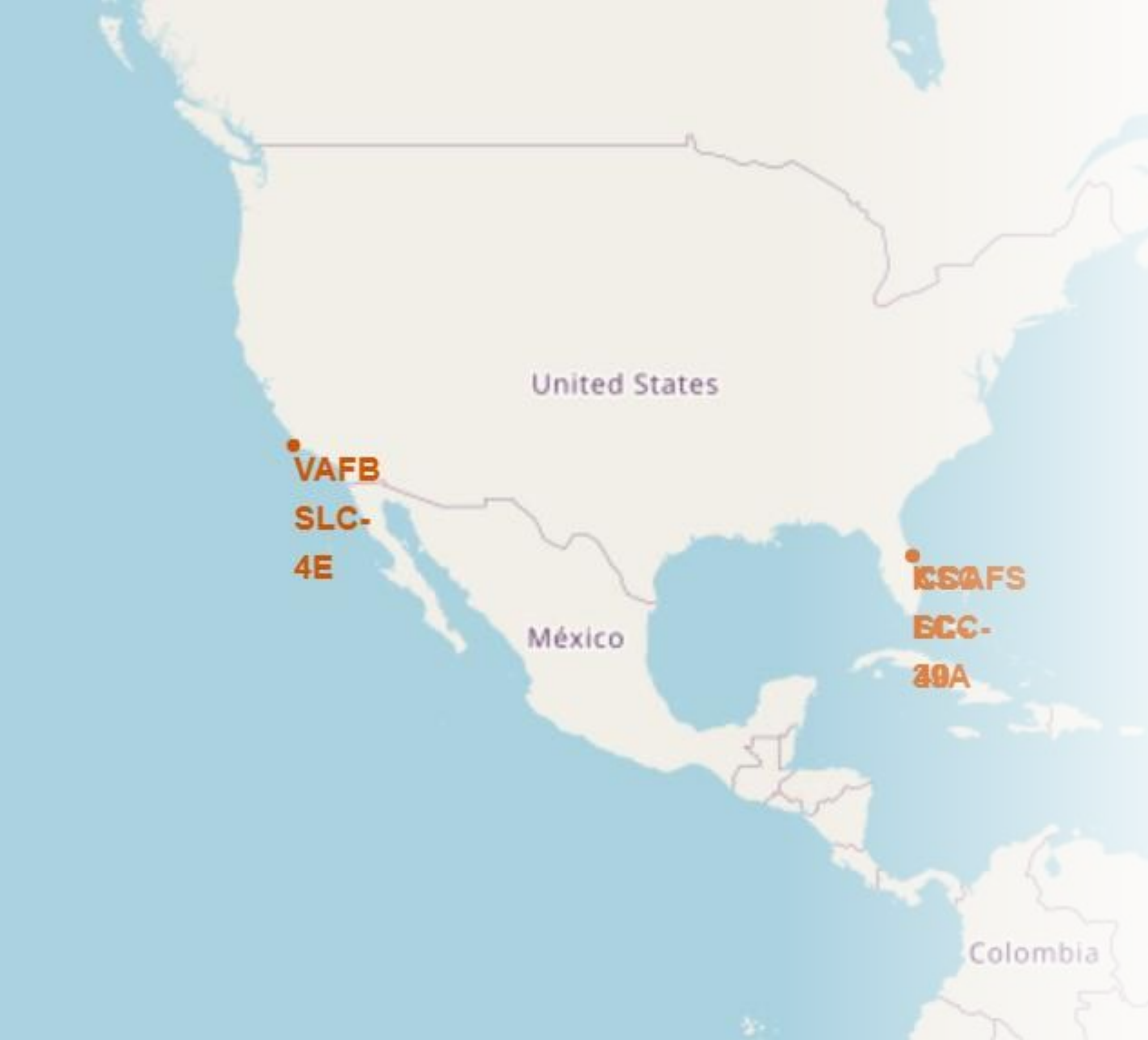
```
 * ibm_db_sa://xsh02076:***@9938aec0-8105-433e-8bf9-0fbb7e483086.c1ogj3sd0tgtu0l
Done.
```

DATE	landing__outcome	total_count
2010-06-04	Failure (parachute)	1
2010-12-08	Failure (parachute)	1
2012-05-22	No attempt	1
2012-10-08	No attempt	1
2013-03-01	No attempt	1
2013-09-29	Uncontrolled (ocean)	1
2013-12-03	No attempt	1
2014-01-06	No attempt	1
2014-04-18	Controlled (ocean)	1
2014-07-14	Controlled (ocean)	1
2014-08-05	No attempt	1

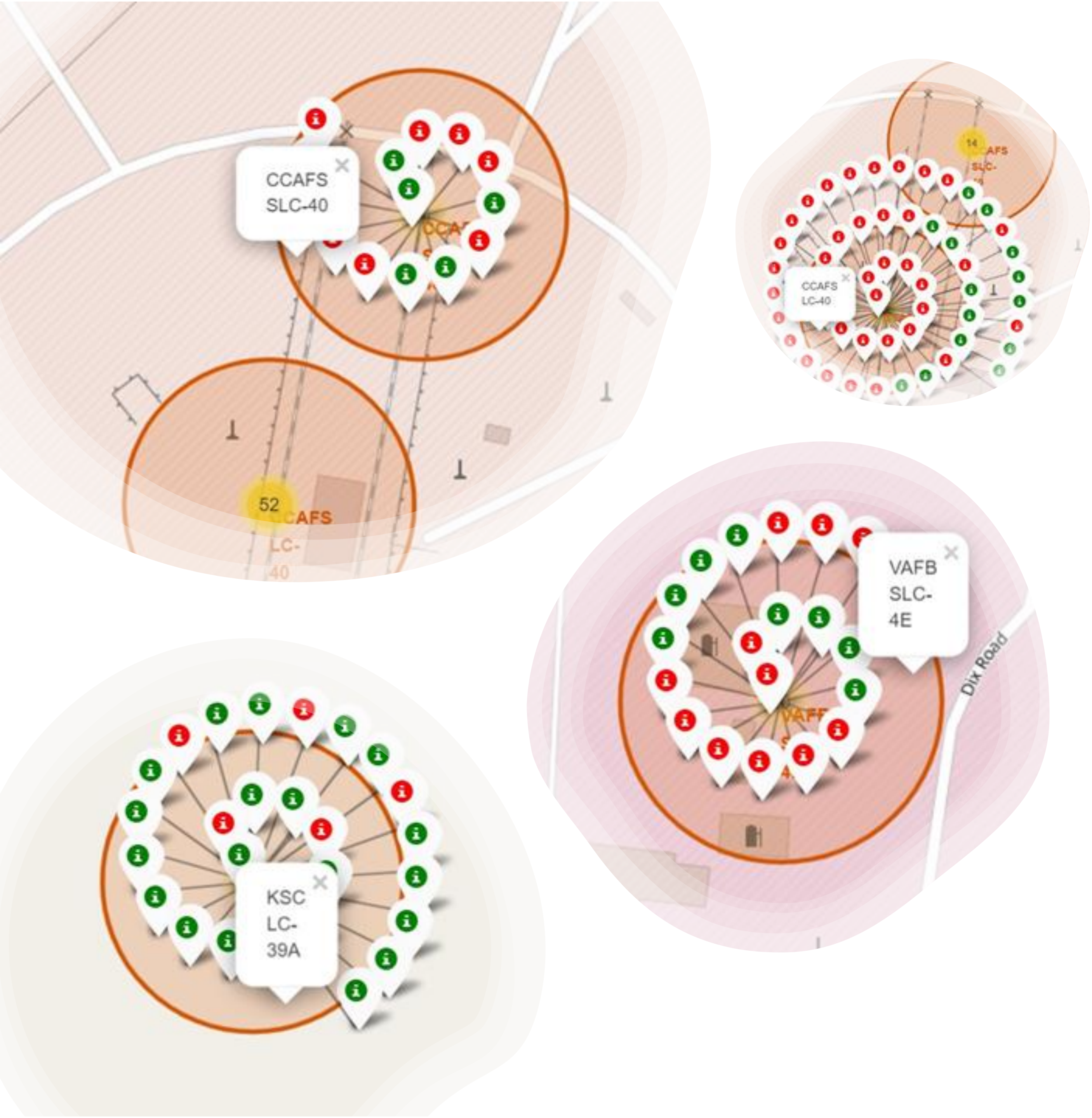
Section 3

Launch Sites Proximities Analysis

Launch Sites Global Map Markers



Launch sites are located on the coasts of the United States of America. VAFB SLC-4E to the South West of the USA and the other Launch sites to the South West of the USA, in Florida.

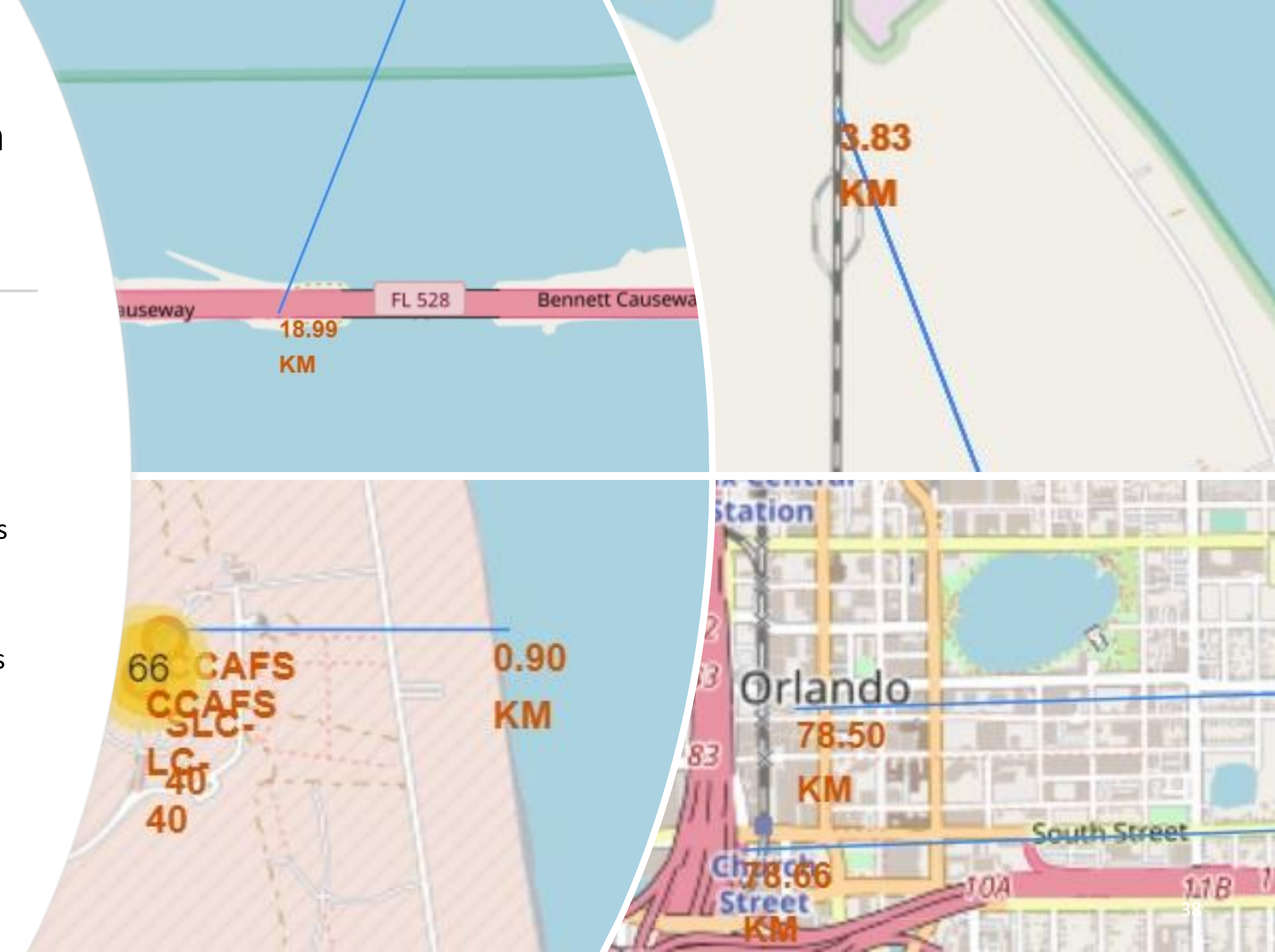


Launch Outcomes on Map

- GREEN MARKERS shows signifies successful launch outcomes while RED MARKERS signifies failed launch outcomes.
- All Launch sites Except VAFB SLC-4E which is situated off the ccoast of California are located off the coast of Florida.

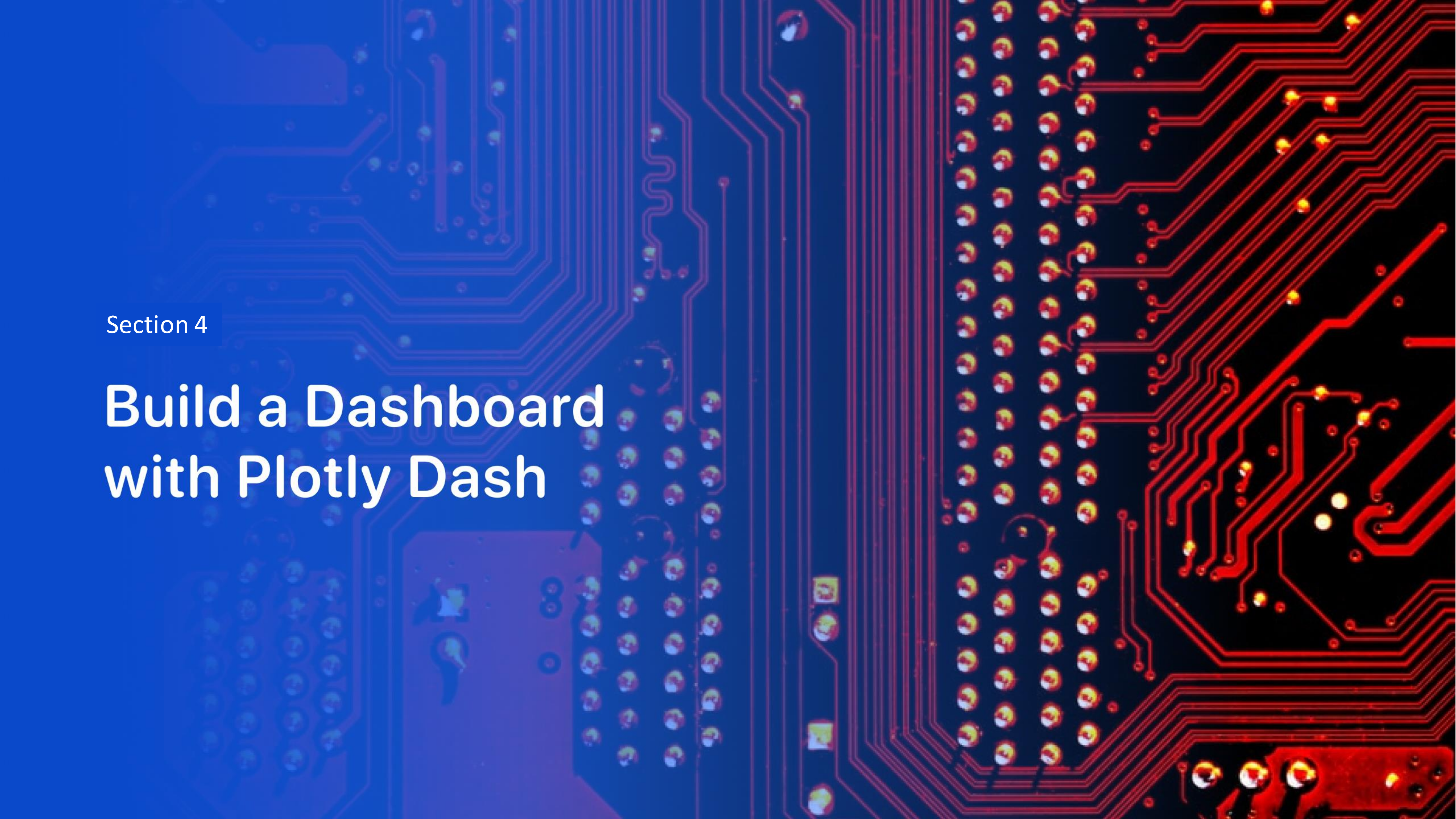
Distance of Launch Site CCAFS SLC-40 to its Proximities

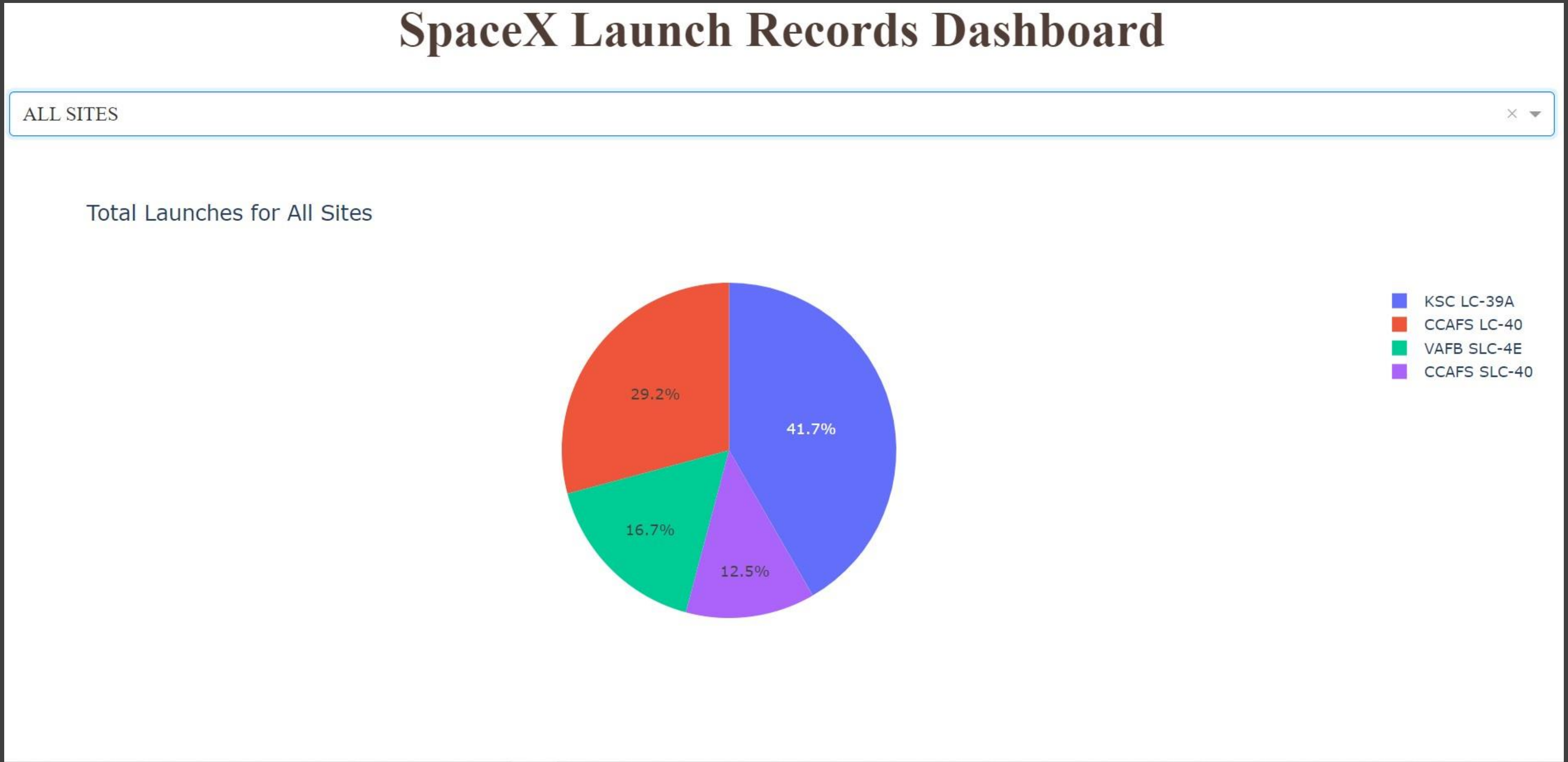
The screenshots are of the CCAFS SLC-40 Launch site to its proximities. The Launch site is close to the coastline and the NASA railway with a distance of 0.90KM and 3.83KM respectively. But it is a certain distance from Highways with a distance of 18.99KM from the Bennett Causeway. Also, Launch sites are very far away from the cities as it is approximately 78KM away from one of the closest city, Orlando.



Section 4

Build a Dashboard with Plotly Dash





SpaceX Launch Records Dashboard

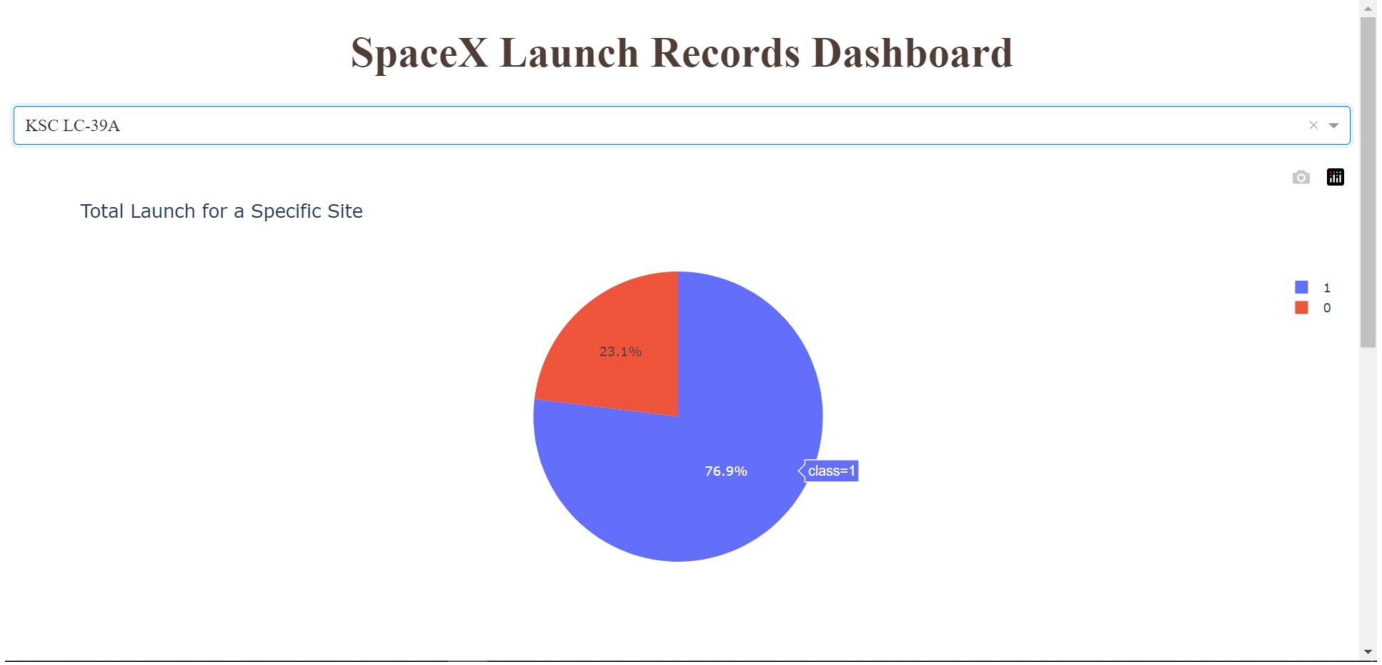
ALL SITES

Launch Success Count for All Sites Dashboard

From the pie chart, it is discovered that the highest success count was at Launch site KSC LC-39A, with a percentage of 41.7%. CCAFS LC-40 came second in success launch count with 29.2% and the least success count was recorded at CCAFS LC-40 with 12.5%.

Launch Site with Highest Success Ratio

KSC LC-39A had the highest launch success ratio of all the four launch sites, with a percentage success ratio of 76.9%.



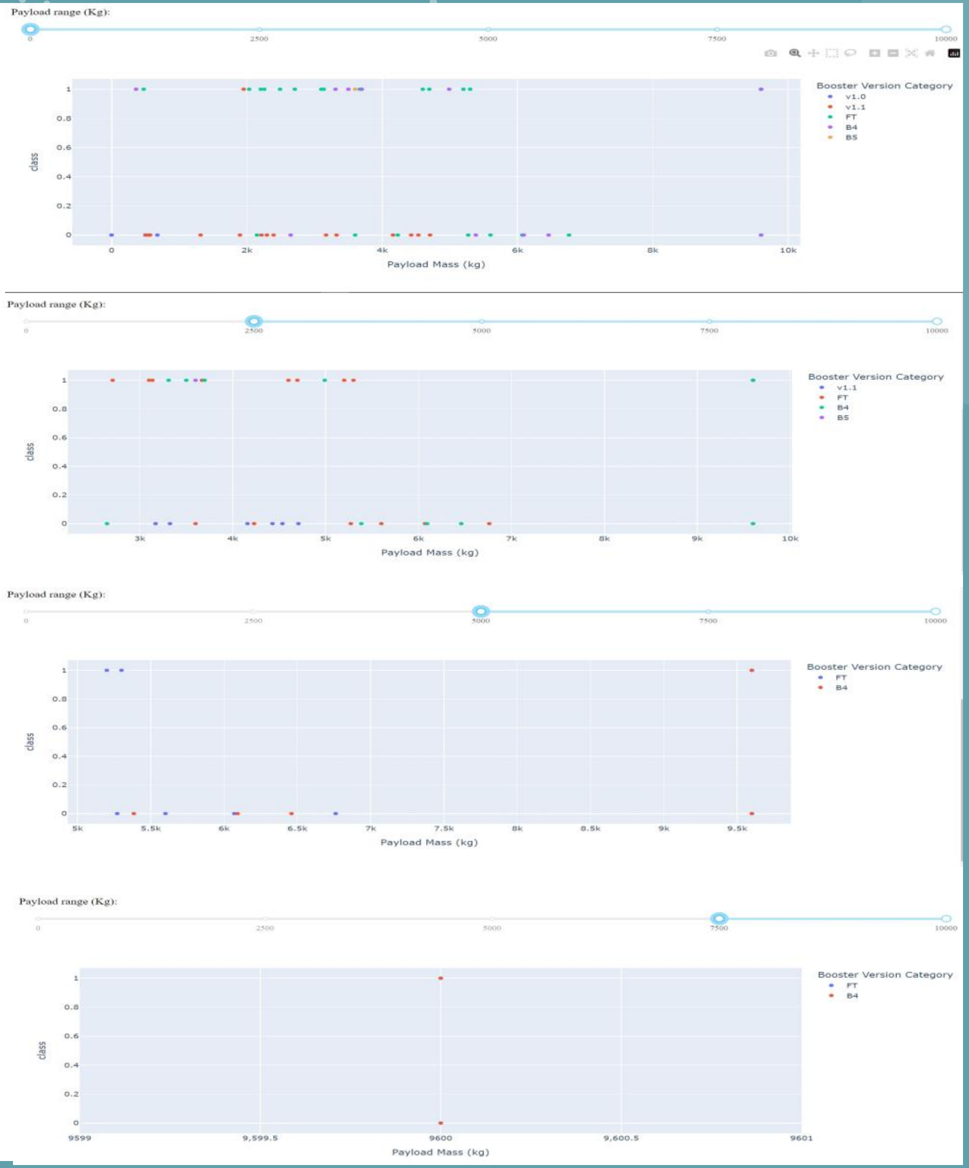
SpaceX Launch Records Dashboard

KSC LC-39A

Total Launch for a Specific Site

Payload vs Launch Outcome Scatter Plot

Success rate for payload lesser than 5000 was more than for heavy weighted payloads (>5000). This is probably due to the fact that fewer launches occurred with a heavy payload and compared to the low weighted payload launches which used booster versions from five different categories, heavier payloadd launches employed two categories of booster versions. This further suggests that the booster versions, as well as the payload is correlated to the success rate of a launch.



Section 5

Predictive Analysis (Classification)

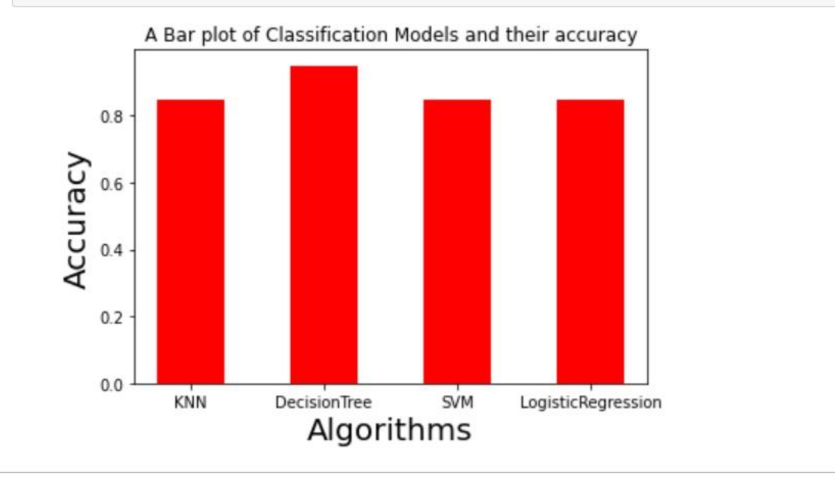
Classification Accuracy on Training Data

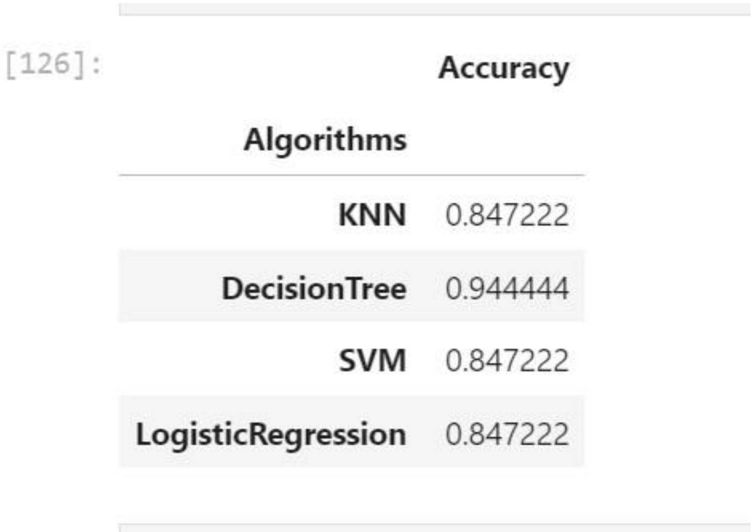
The Decision Tree Model has the highest classification accuracy, having a 95% accuracy on the training data.

However, the Decision Tree Model had an accuracy of approximately 67% on the test data set.

The other three classification Models had an accuracy of approximately 85% on the training data and 83% on the test data set. All other three Models are tied on accuracy.

Based on these facts, The Decision Tree Model edges out.

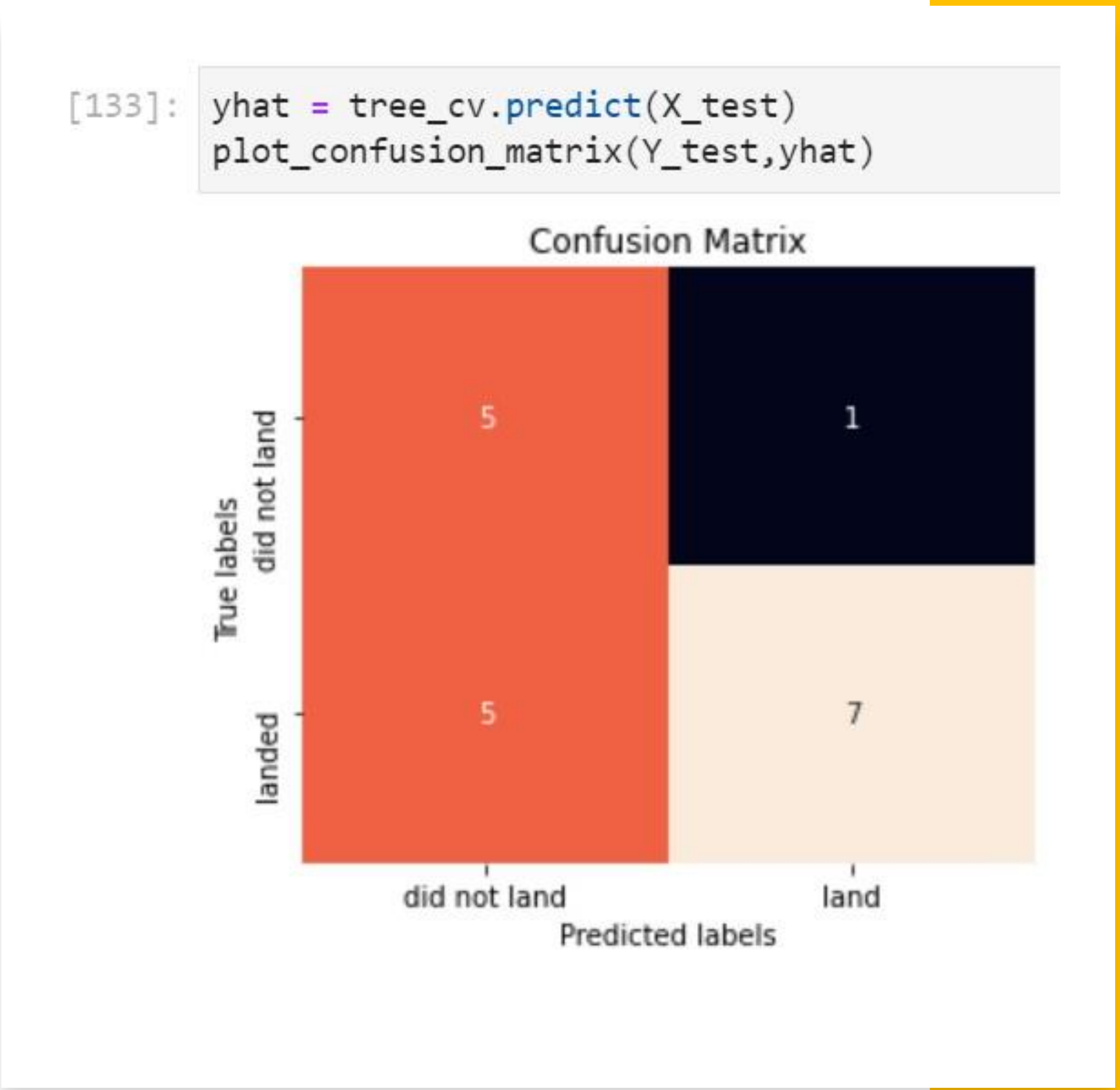



[126]:

	Accuracy
Algorithms	
KNN	0.847222
DecisionTree	0.944444
SVM	0.847222
LogisticRegression	0.847222

Confusion Matrix

The Confusion Matrix of the Decision Tree Model shows that the model was relatively accurate on the True Positve and True Negative results. The problem that needs fixing is the False Negative (Type 2) error.



```
[133]: yhat = tree_cv.predict(X_test)
       plot_confusion_matrix(Y_test,yhat)
```

Conclusions

The highest Success Rate was Recorded in Orbits HEO, SSO, ESL1 and GEO.

Lower payload mass had higher success rates than higher payload masses

Launches from the KSC LC-39A were the most successful with a success rate of 76.9%

Launch success rate increased with time, with our analysis showing us an increase from 2013 until 2020

The Decision Tree classifier gave the best accuracy on the training data set with 95% but did relatively low on test data set (67%) compared to the other models

Targeting Launches to HEO, GEO, SSO and ESL1 orbits with a lower payload mass from the KSC LC-39A Launch site might increase the success rate exponentially.

Appendix

The below code snippet was used to generate the bar plot for the best classifier.

```
accuracy = [knn_cv.best_score_, tree_cv.best_score_, svm_cv.best_score_, logreg_cv.best_score_]
classifications = ['KNN', 'DecisionTree', 'SVM', 'LogisticRegression']
plot_data = pd.DataFrame(accuracy, classifications, columns=['Accuracy'])
plot_data.index.name = 'Algorithms'
plot_data
plt.bar(classifications, plot_data['Accuracy'], color='red', width=0.5)

plt.xlabel("Algorithms", fontsize=20)
plt.ylabel("Accuracy", fontsize=20)
plt.title("A Bar plot of Classification Models and their accuracy")

plt.show()
```

Thank you!