IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mercy Chege

28th September 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through API and Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Dashboards with Plotly Dash
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis Results
  - Interactive maps and dashboards
  - Predictive Analytics Results

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. The difference in price is because SpaceX can reuse the first stage. Determining if the first stage will land, can enable us determine the cost of a launch. This information can be used by alternate companies that want to compete with SpaceX for a rocket launch.

- Problems you want to find answers
  - What attributes are correlated with a successful or failed landing ?
  - How does the various rocket variables affect the success or failure of the first stage landing?
  - How to achieve the best landing success rate ?

Section 1

Methodology

Methodology

Executive Summary

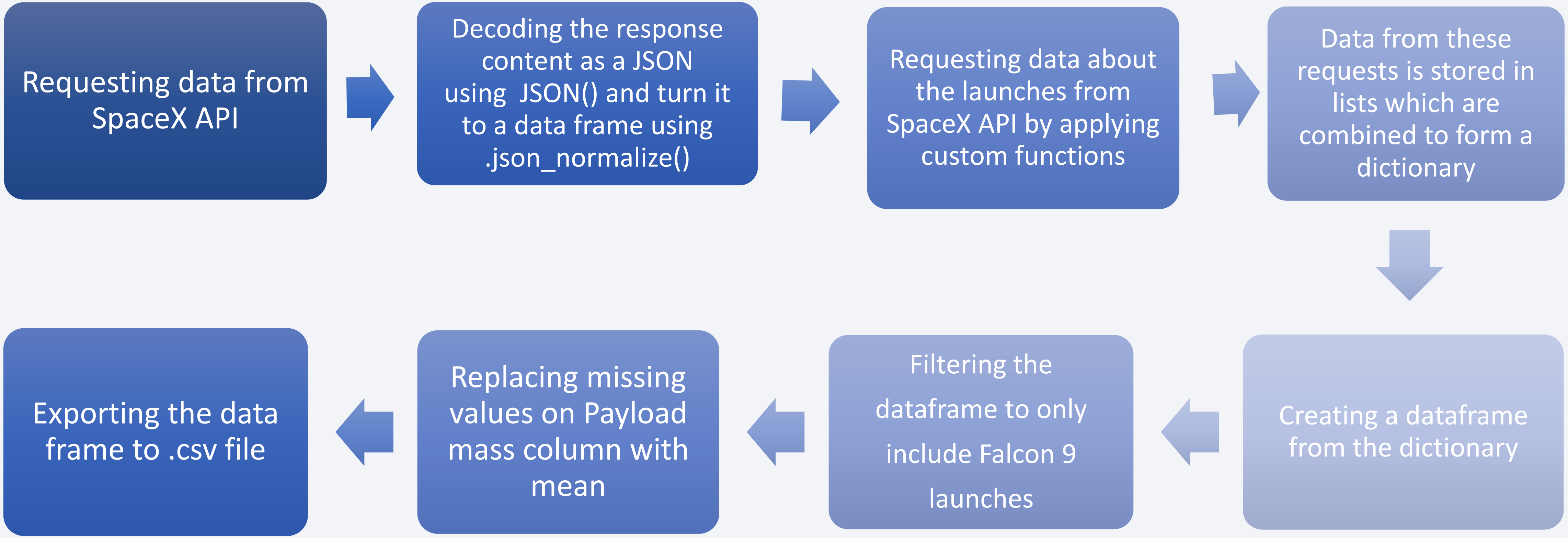
- Data collection methodology:
  - Data was collected using:
    - SpaceX REST API
    - Web Scrapping from Wikipedia
- Perform data wrangling
  - Filtering the data and Dealing with null values
  - One Hot Encoding for classification models
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, tuning and evaluation of classification models

Data Collection

- Data was collected from SpaceX REST API endpoints (https://api.spacexdata.com/v4/) and Web scrapping Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches)
- Data obtained from SpaceX REST API include:
  - FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite,Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount,Serial, Longitude, Latitude
- Data obtained from Wikipedia Web Scraping include:
  - Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

Data collection on SpaceX API involved the following processes:



Requesting data from SpaceX API

Decoding the response content as a JSON using JSON() and turn it to a data frame using .json_normalize()

Requesting data about the launches from SpaceX API by applying custom functions

Data from these requests is stored in lists which are combined to form a dictionary

Creating a dataframe from the dictionary

Filtering the dataframe to only include Falcon 9 launches

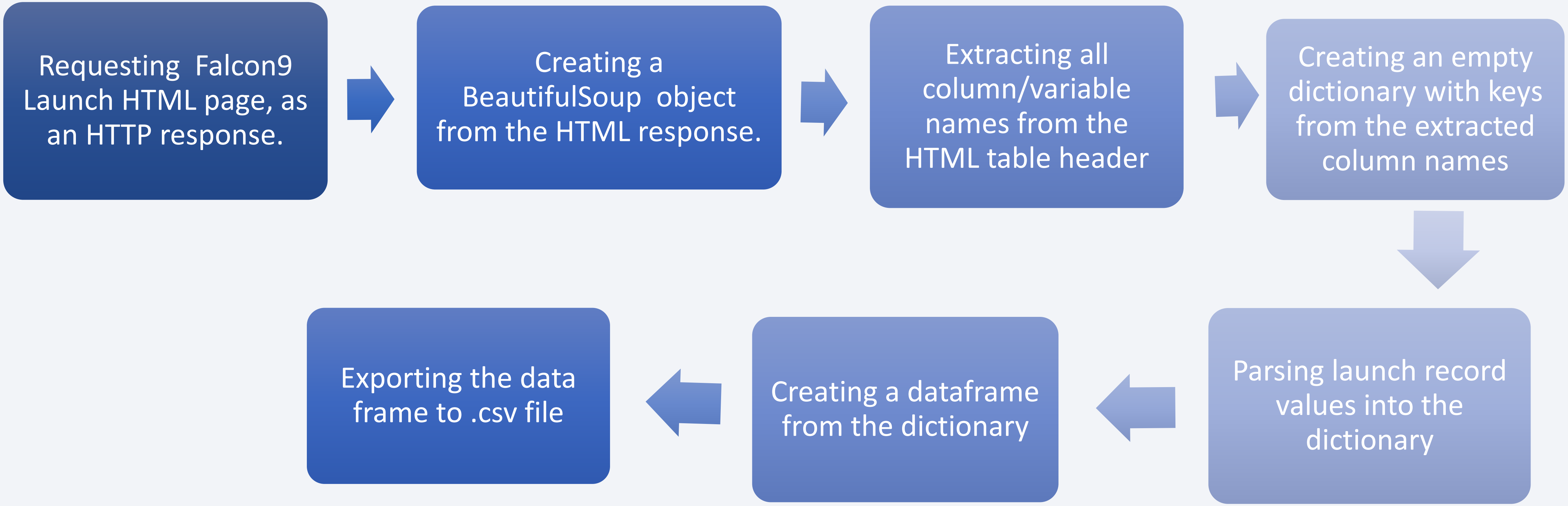
Replacing missing values on Payload mass column with mean

Exporting the data frame to .csv file

The link to the data collection notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X%20-%20Data%20Collection.ipynb

Data Collection – Scraping

Data collection on Web Scraping involved the following processes:



Requesting Falcon9 Launch HTML page, as an HTTP response.

Creating a BeautifulSoup object from the HTML response.

Extracting all column/variable names from the HTML table header

Creating an empty dictionary with keys from the extracted column names

Parsing launch record values into the dictionary

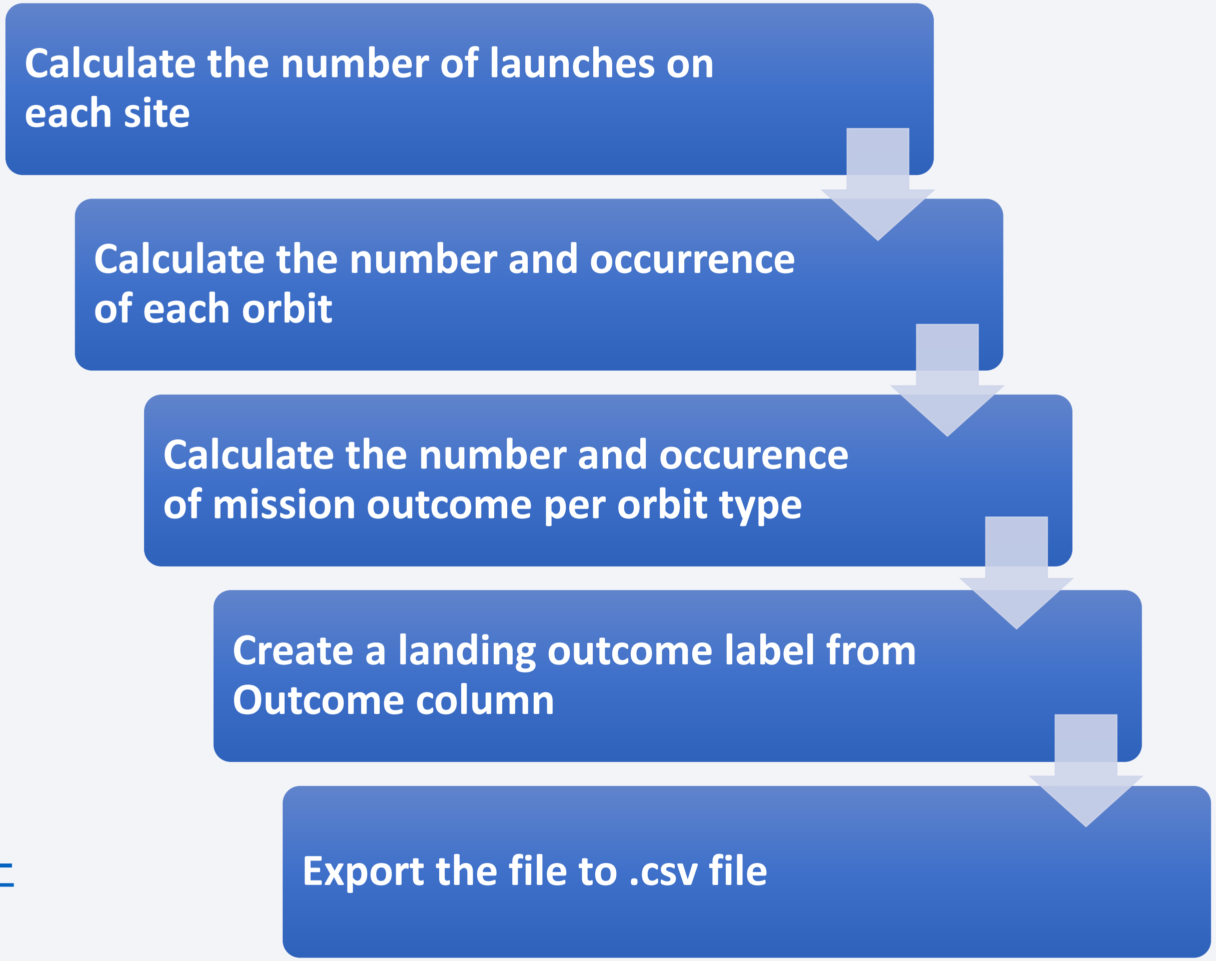
Creating a dataframe from the dictionary

Exporting the data frame to .csv file

The link to the notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X%20-%20Data%20Collection%20with%20Web%20Scraping.ipynb

Data Wrangling

- Data wrangling process involved conversions of the booster success or failed landing outcomes into **Training Labels** with **1** meaning the booster landed **successfully** and **0** meaning it was **unsuccessful.**
- The link to the notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X%20-%20Data%20Wrangling.ipynb



Calculate the number of launches on each site

Calculate the number and occurrence of each orbit

Calculate the number and occurence of mission outcome per orbit type

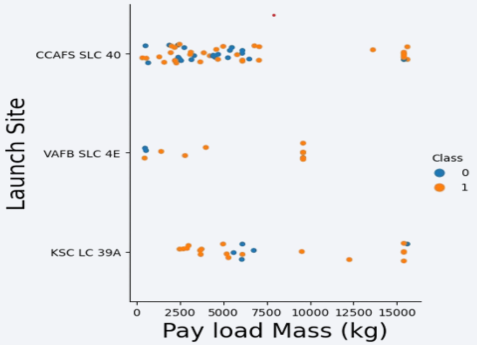
Create a landing outcome label from Outcome column

Export the file to .csv file

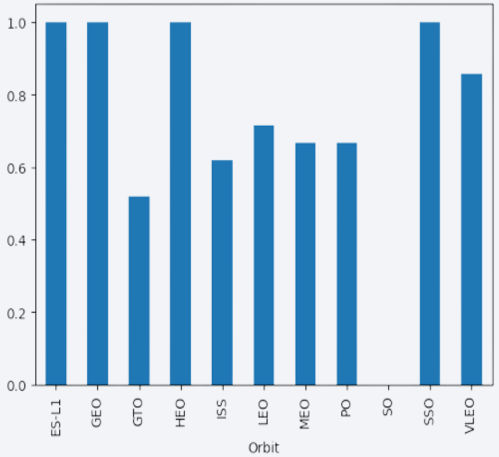
EDA with Data Visualization

- Charts plotted to visualize data included:

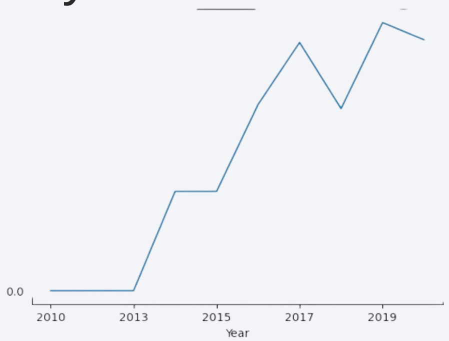
  1. Scatter Plots to show the relationship between variables.



  2. Bar Graph to show the relationship between categorical and numerical variables



  3. Line Graph to show trend analysis over time



The link to the notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X-%20EDA%20With%20Visualization%20.ipynb

EDA with SQL

- The following queries were performed:
  - Displaying the names of the unique launch sites in the space mission
  - Displaying 5 records where launch sites begin with the string ‘CCA‘
  - 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 when the first successful landing outcome in ground pad was achieved
  - Listing the names of the boosters which have success in drone ship 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 failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015
  - Ranking 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
- The link to the notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X%20-%20EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- The following map objects were created and added to folium map:
  - Markers to indicate launch sites and NASA Johnson Space Center
  - Circles to indicate highlighted areas around specific coordinates
  - Marker clusters to indicate groups of events in each coordinate, like launches in a launch site
  - Lines are used to indicate distances between two coordinates.

- The link to the notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X%20-%20Interactive%20Visual%20Analytics%20with%20Folium%20.ipynb

Build a Dashboard with Plotly Dash

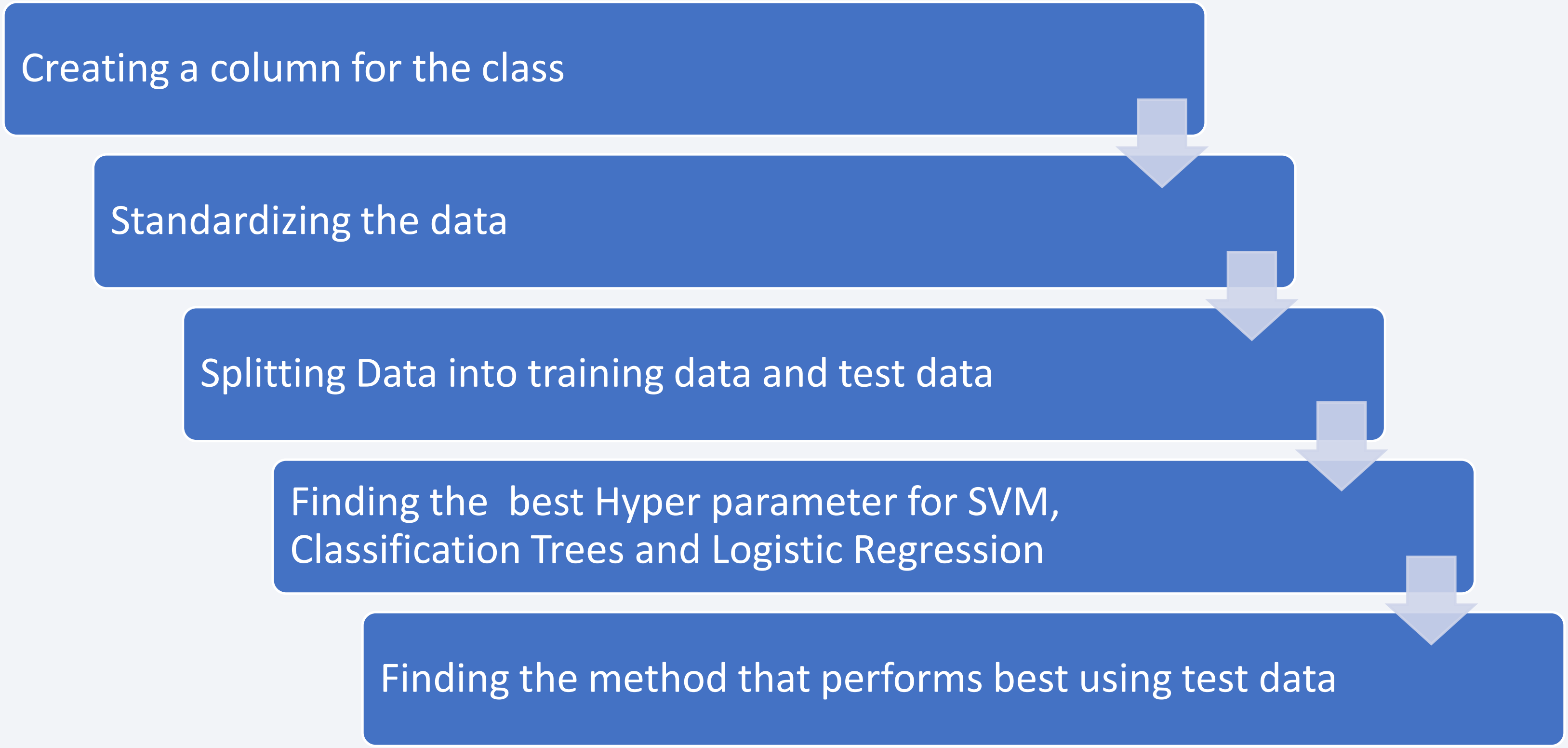
- The dashboard includes the following:

  1. Dropdown list to allow users select one or all launch sites
  2. Pie Chart showing the total successful launches count for all sites and the Success vs. Failed counts for the site selected
  3. Range slider to allow users select a payload mass in a given range
  4. Scatter Chart to show Payload Mass vs. Success Rate for the different Booster Versions

- GitHub URL : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Model development process involved:



Creating a column for the class

Standardizing the data

Splitting Data into training data and test data

Finding the best Hyper parameter for SVM, Classification Trees and Logistic Regression

Finding the method that performs best using test data

The link to the notebook is : https://github.com/mercychege/Applied-Data-Science-Capstone/blob/main/Space%20X%20-%20Machine%20Learning%20Prediction.ipynb

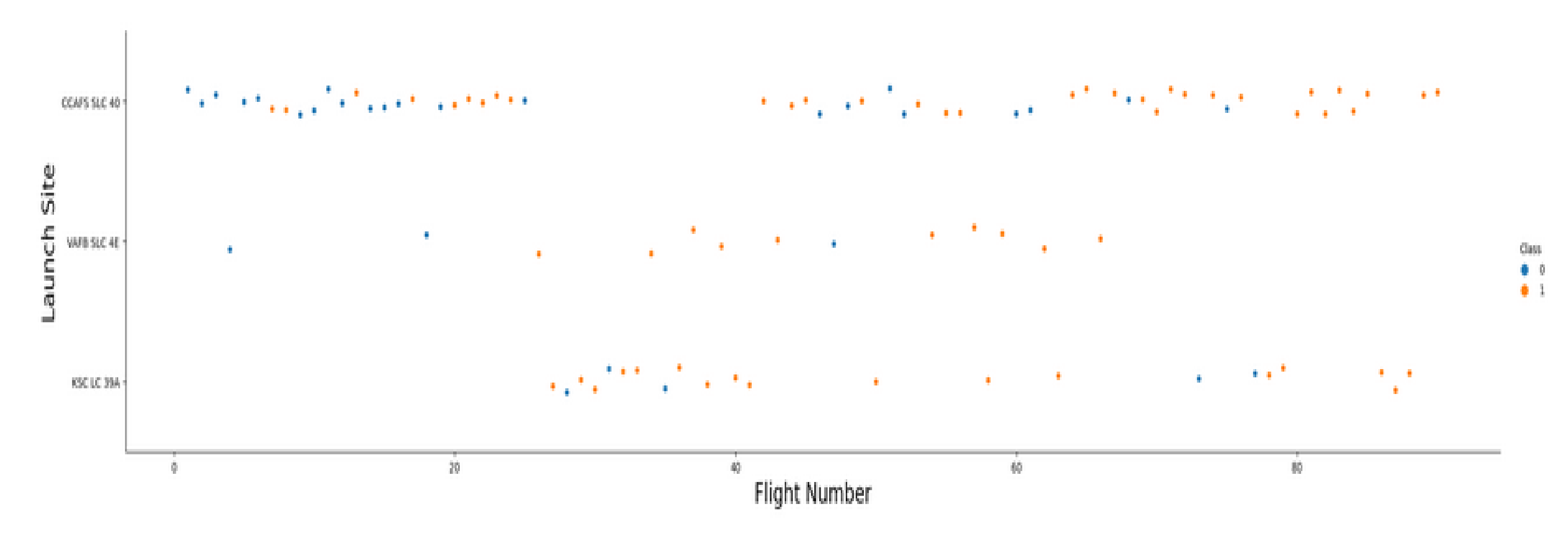
Results

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

Section 2

Insights drawn from EDA

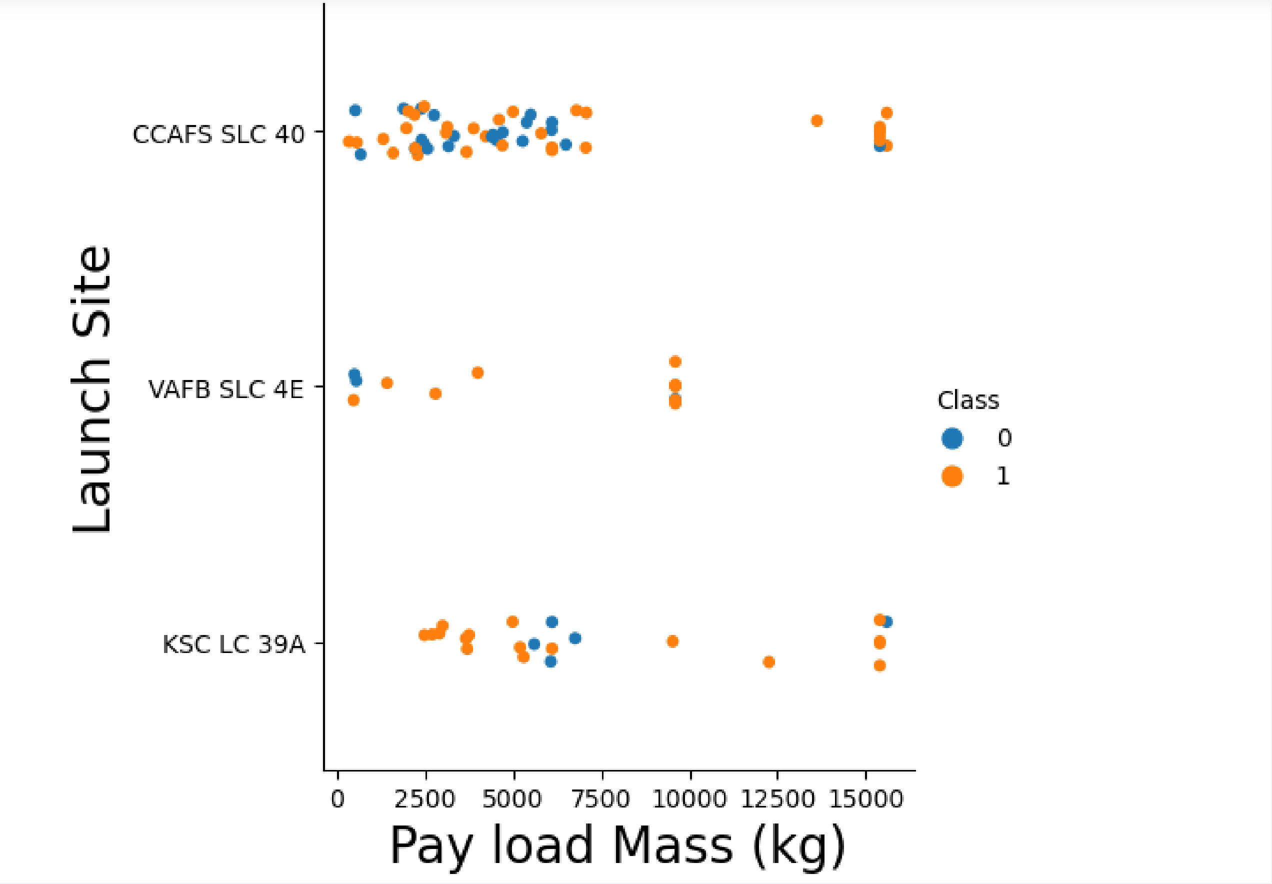
Flight Number vs. Launch Site



- We observe that the larger the flight number at a launch site, the greater the success rate at the launch site

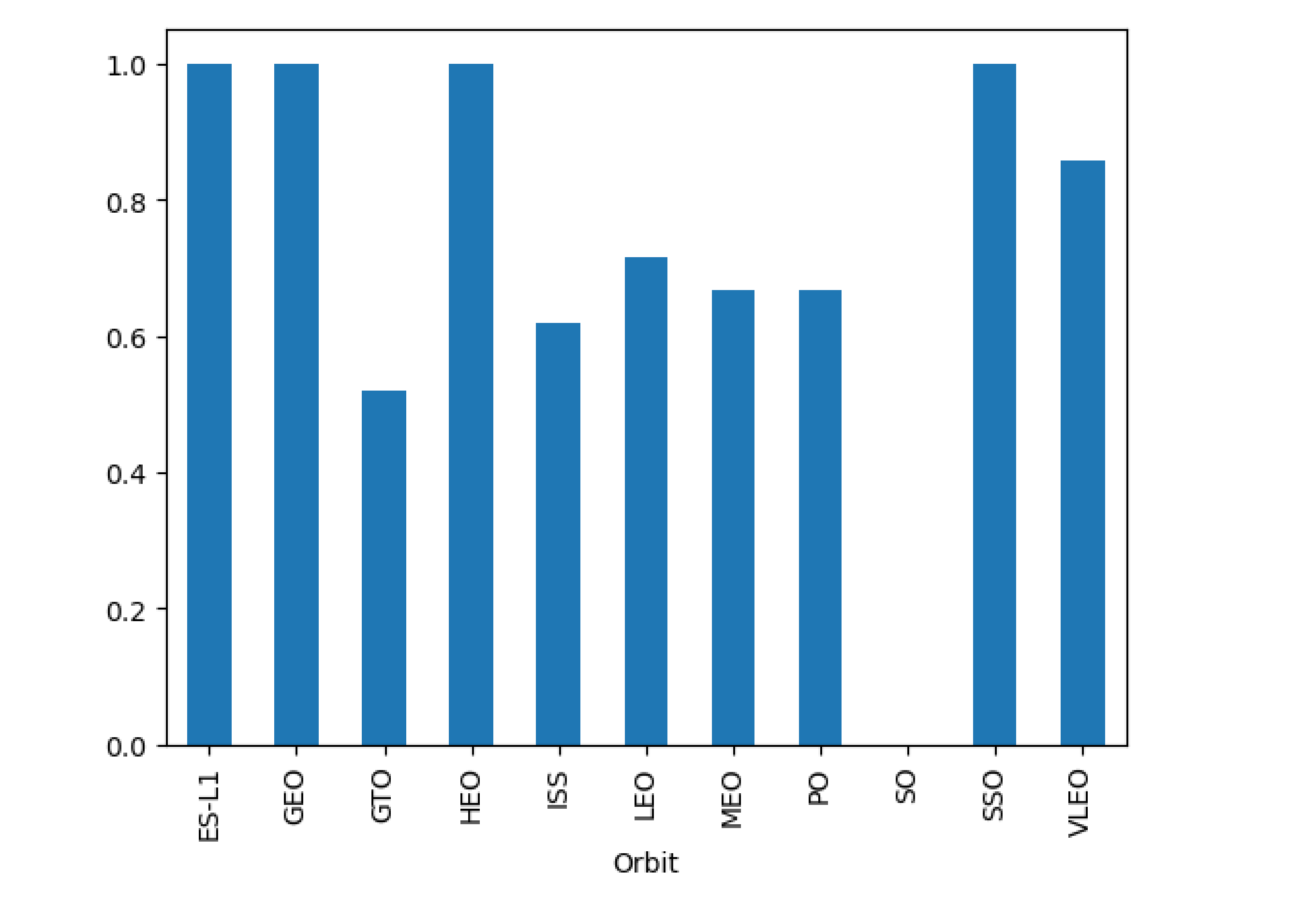
Payload vs. Launch Site

- The higher the payload mass, the higher the success rate.
- VAFB-SLC launch site had no rockets launched for heavy payload mass(greater than 10000).
- KSC LC 39A has a 100% success rate for payload mass under 5500 kg



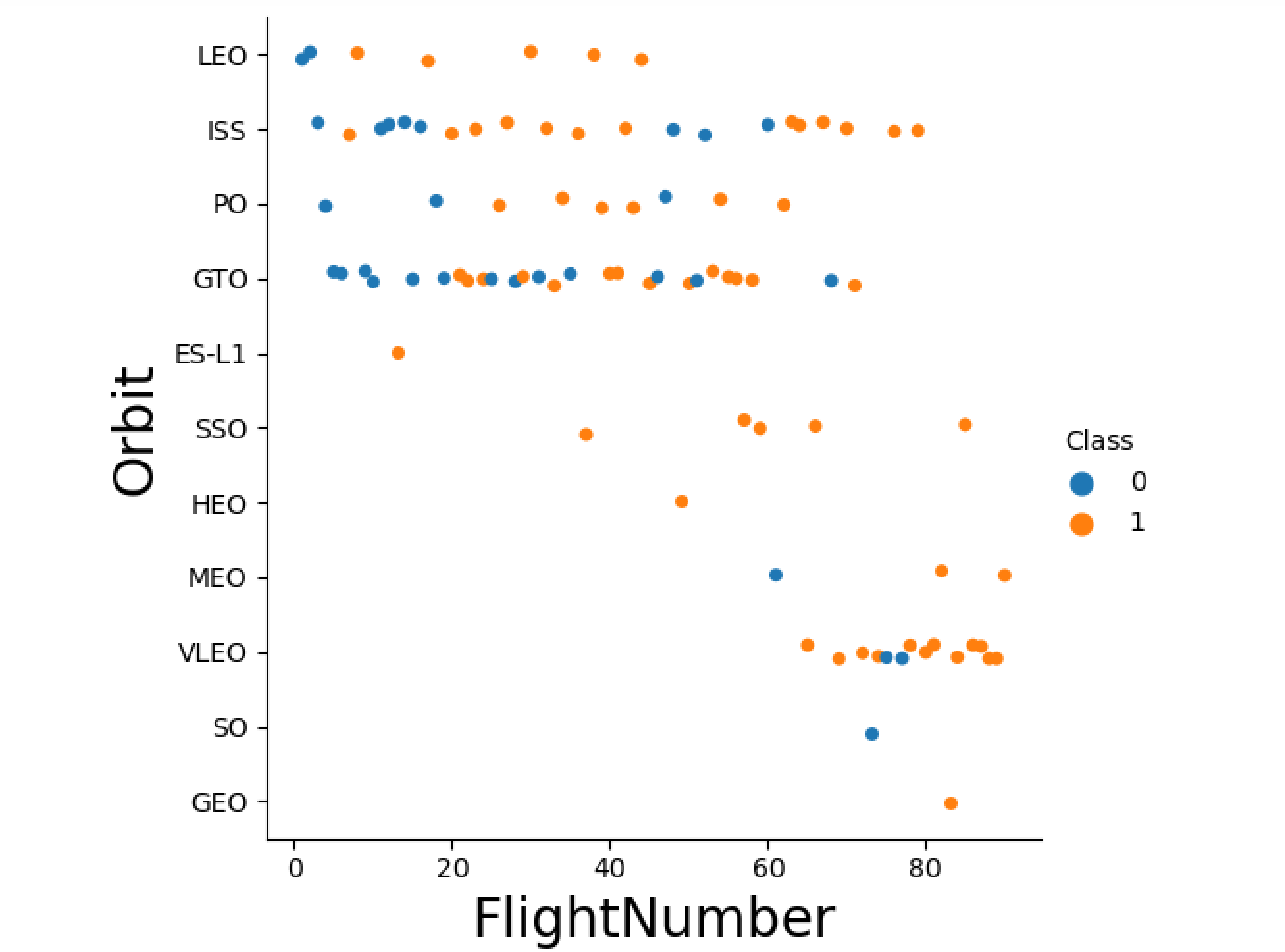
Success Rate vs. Orbit Type

- Orbit Type ES-L1, GEO,HEO,SSO had 100% success rate
- Orbit Type SO had 0% success rate.
- Orbit Type GTO, ISS,LEO,MEO,PO and VLEO had success rate between 50% and 85%



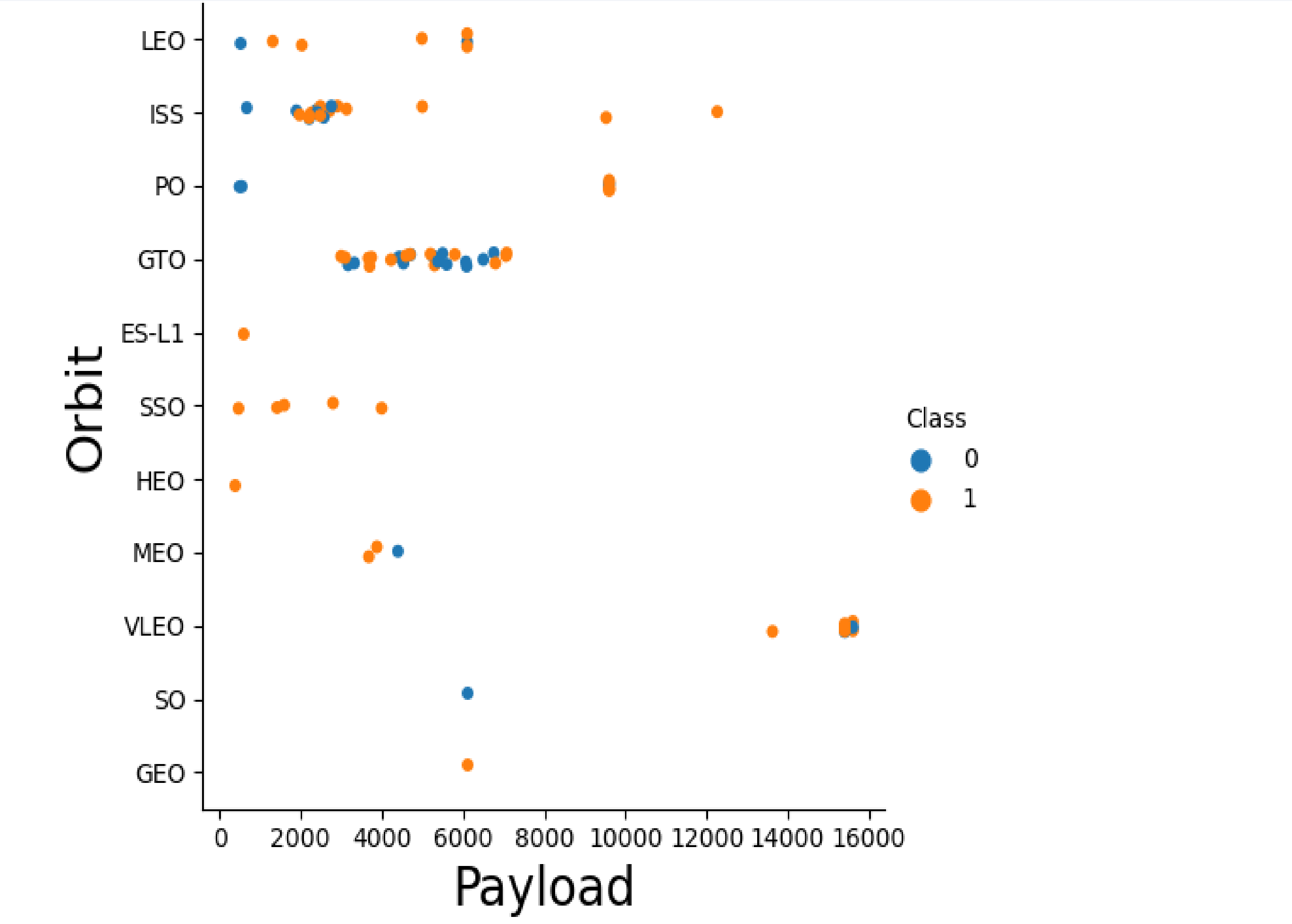
Flight Number vs. Orbit Type

- In the LEO orbit the Success rate increases by the flight number.
- In GTO orbit, there seems to be no relationship between success rate and flight number



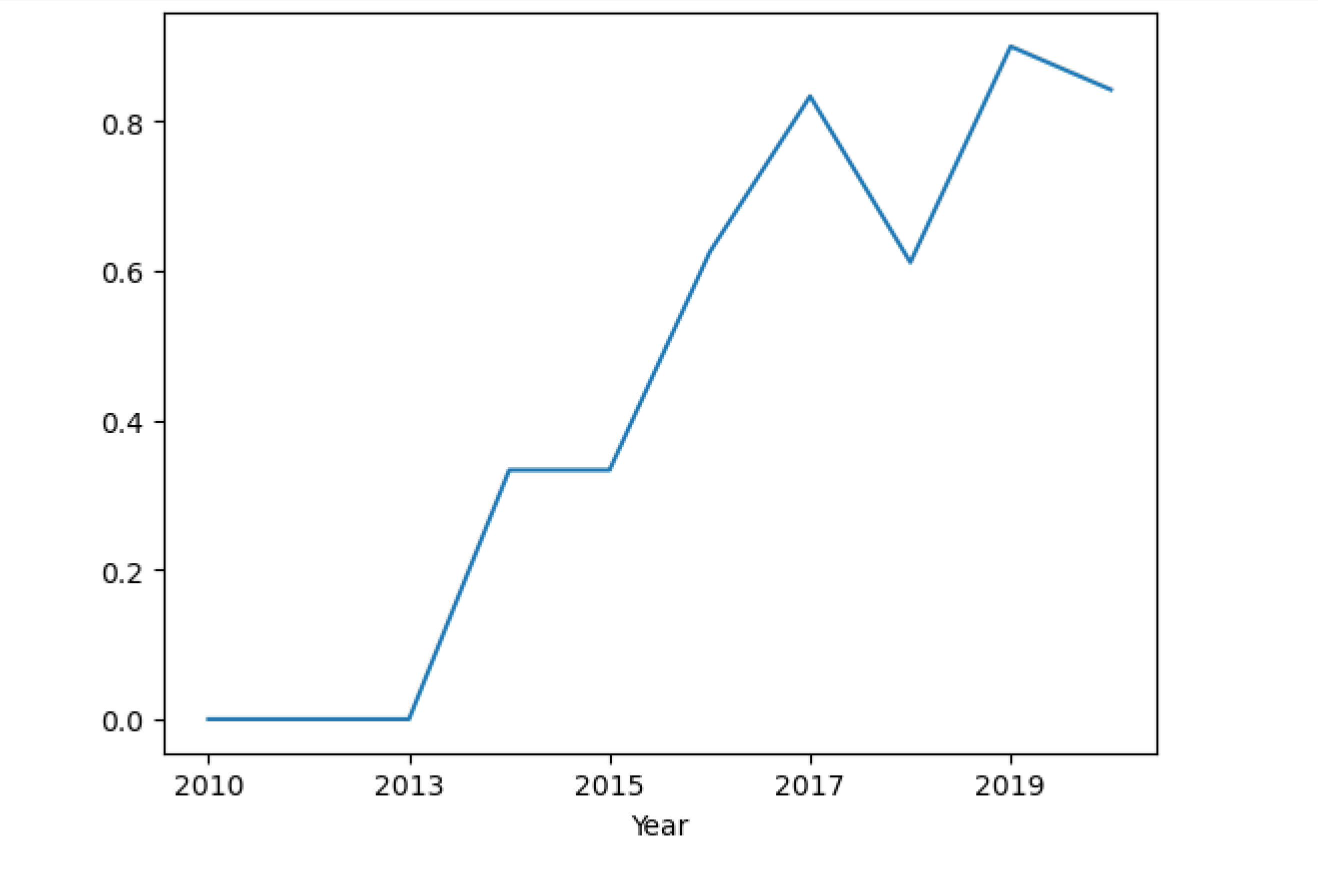
Payload vs. Orbit Type

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



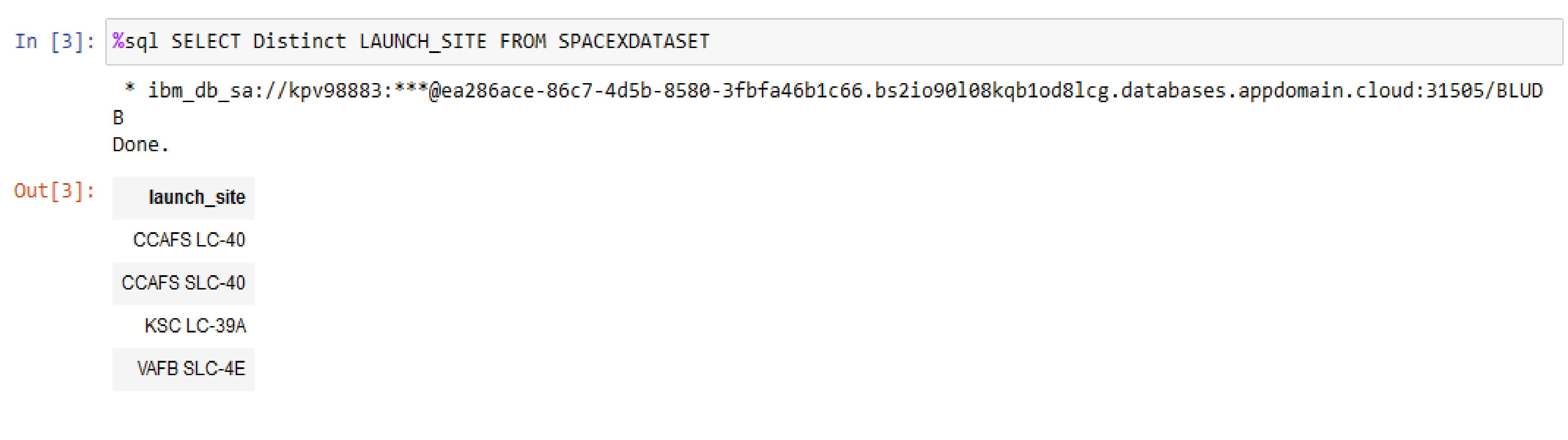
Launch Success Yearly Trend

- We observe that success rate since 2013 kept increasing till 2020



All Launch Site Names

- There are four unique launch sites.
- **DISTINCT** keyword is used in the query to show only unique launch sites from the data set.

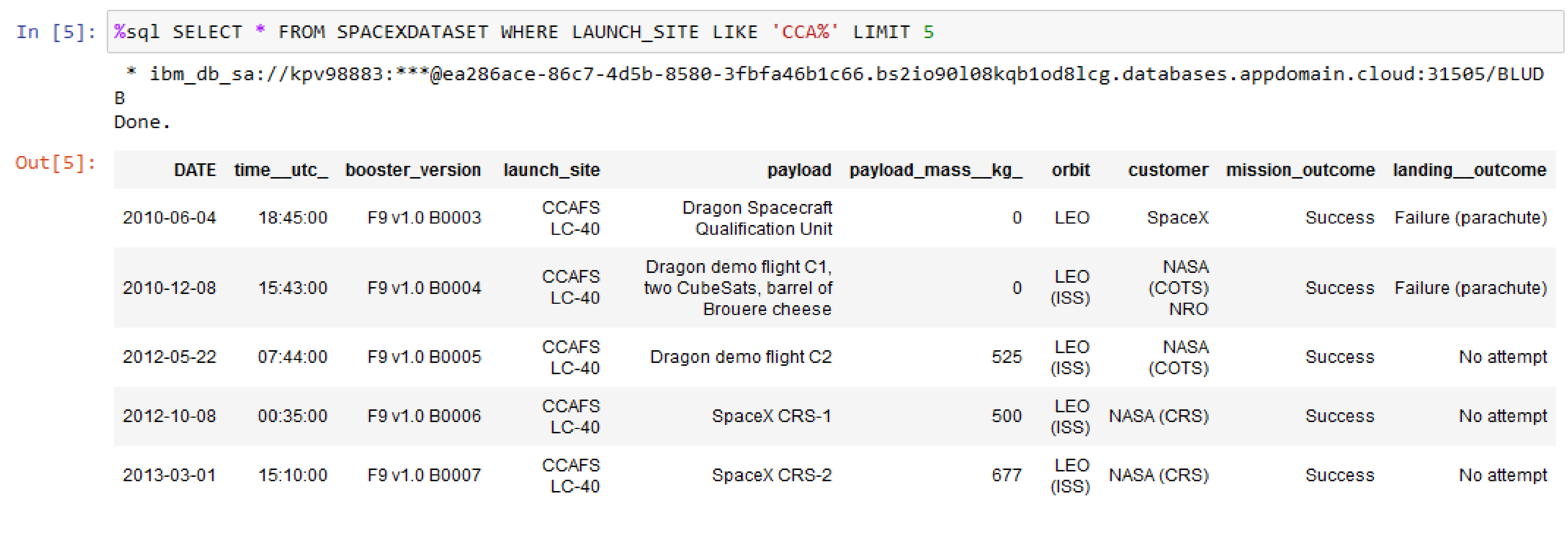


```
In [3]: %sql SELECT Distinct LAUNCH_SITE FROM SPACEXDATASET

 * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
B
Done.

Out[3]:
```

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

Launch Site Names Begin with 'CCA'

- To display 5 launch sites whose name begin with ‘CCA’, we use the **WHERE** clause with **LIKE** clause to filter launch sites that contain sub string ‘CCA’ and **LIMIT** clause to only display 5 records.



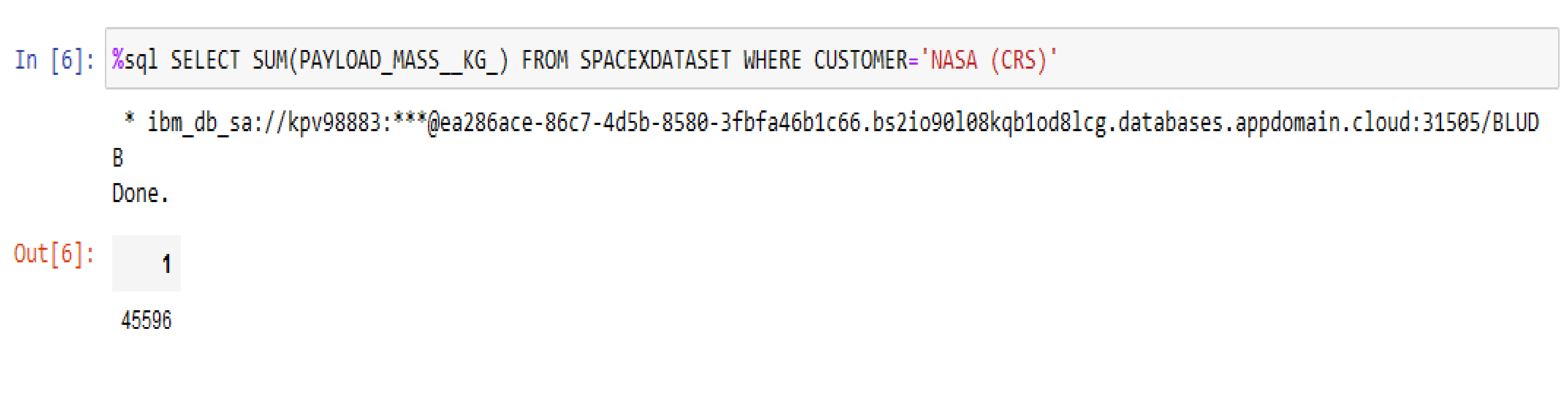
```
In [5]: %sql SELECT * FROM SPACEXDATASET WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5

         * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
        B
        Done.
```

Out[5]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

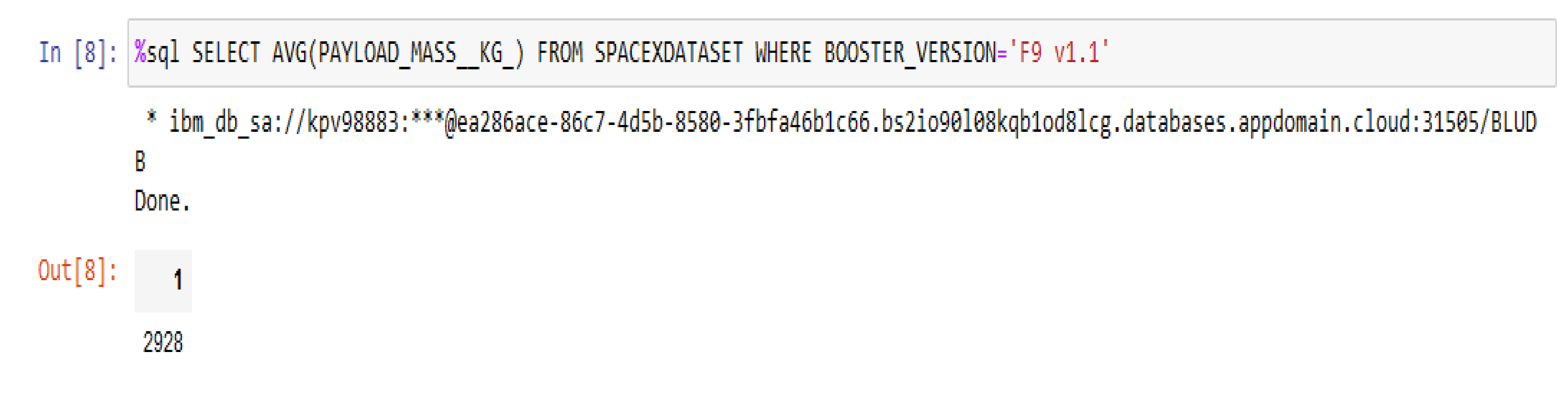
- The query sums payload mass where customer is ‘NASA (CRS)’



```
In [6]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXDATASET WHERE CUSTOMER='NASA (CRS)'

         * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
        B
        Done.

Out[6]:      1

         45596
```

Average Payload Mass by F9 v1.1

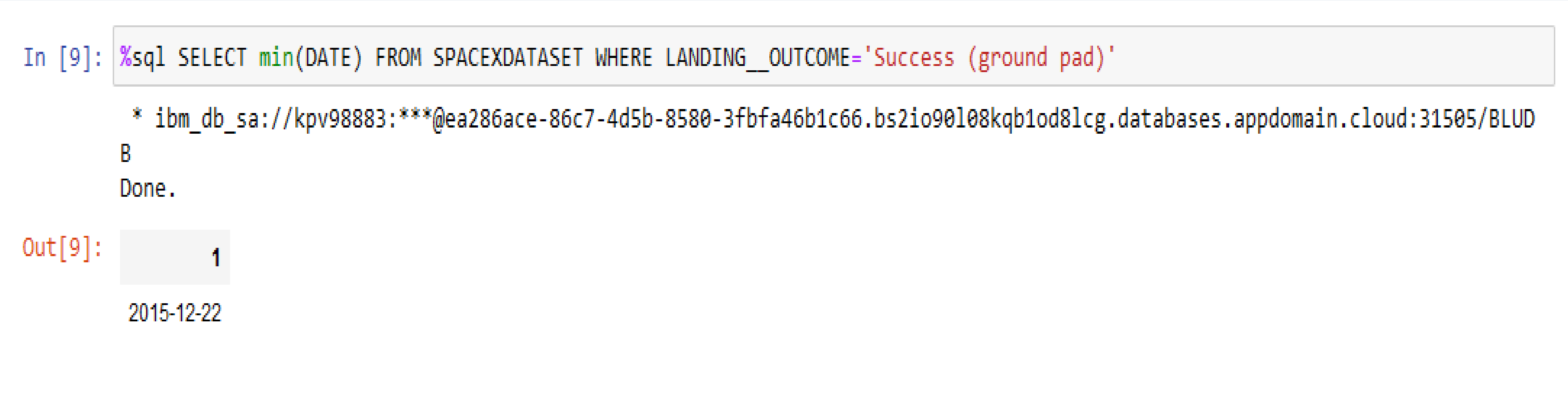
- The query calculates the average payload mass carried by booster version F9 v1.1



```
In [8]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXDATASET WHERE BOOSTER_VERSION='F9 v1.1'

         * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
        B
        Done.

Out[8]:    1
        2928
```

First Successful Ground Landing Date

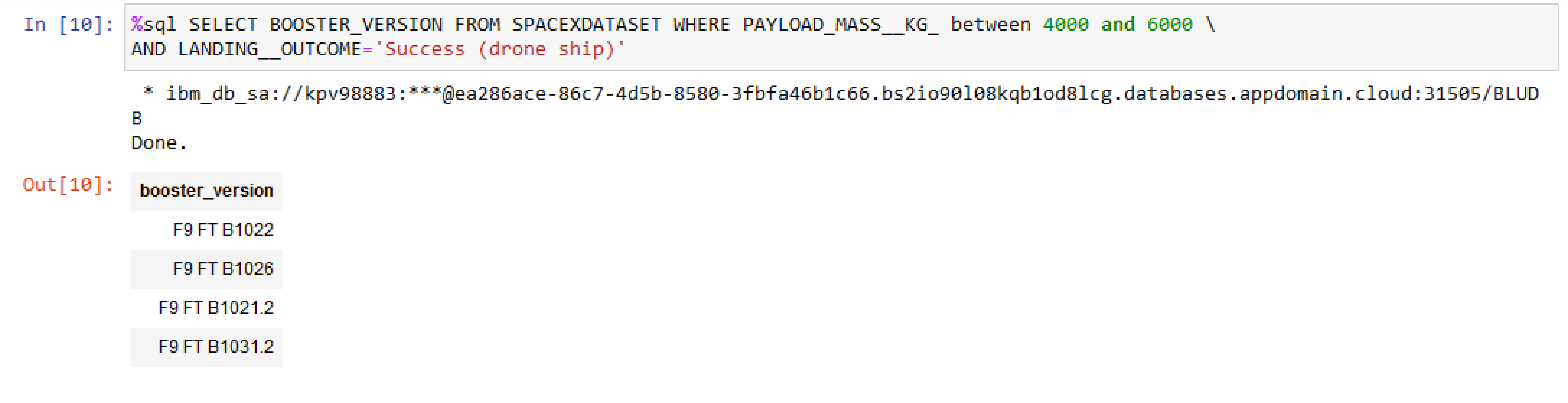
- The query filter successful landing Outcome on Ground pad and gets the first successful date using MIN keyword on the Date column.



```
In [9]: %sql SELECT min(DATE) FROM SPACEXDATASET WHERE LANDING__OUTCOME='Success (ground pad)'

         * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
        B
        Done.

Out[9]:          1
         2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

- The query uses the **WHERE** clause to filters for boosters which have payload mass greater than 4000 but less than 6000 and the **AND** condition to determine those that successfully landed on drone ship.



In [10]:
```
%sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 \
AND LANDING__OUTCOME='Success (drone ship)'
```

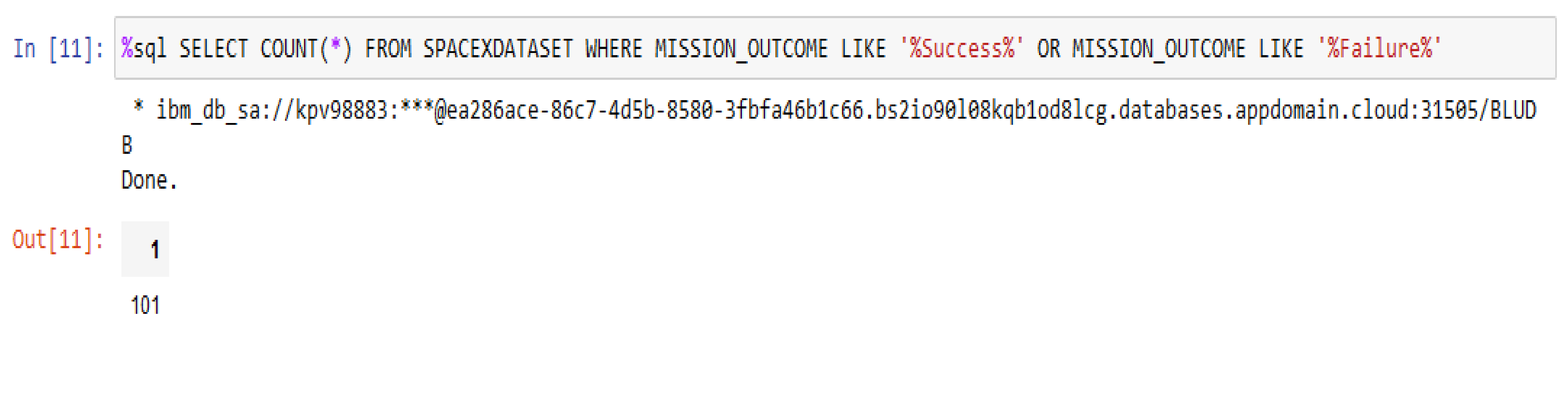
```
 * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
B
Done.
```

Out[10]:

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

Total Number of Successful and Failure Mission Outcomes

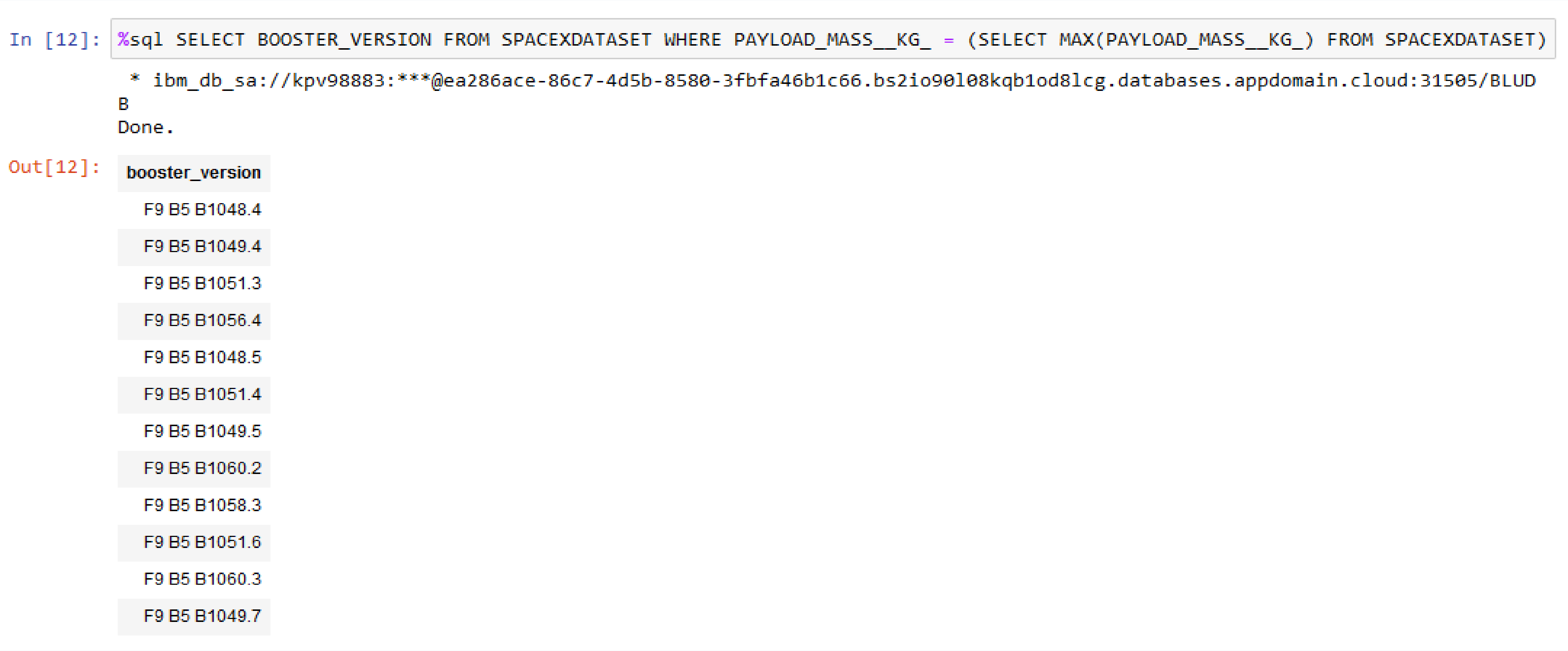
- The query lists the total number of successful and failure mission outcomes



```
In [11]: %sql SELECT COUNT(*) FROM SPACEXDATASET WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'

          * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
         B
         Done.

Out[11]:   1
          101
```

Boosters Carried Maximum Payload

- The query determines the booster that have carried the maximum payload using a subquery



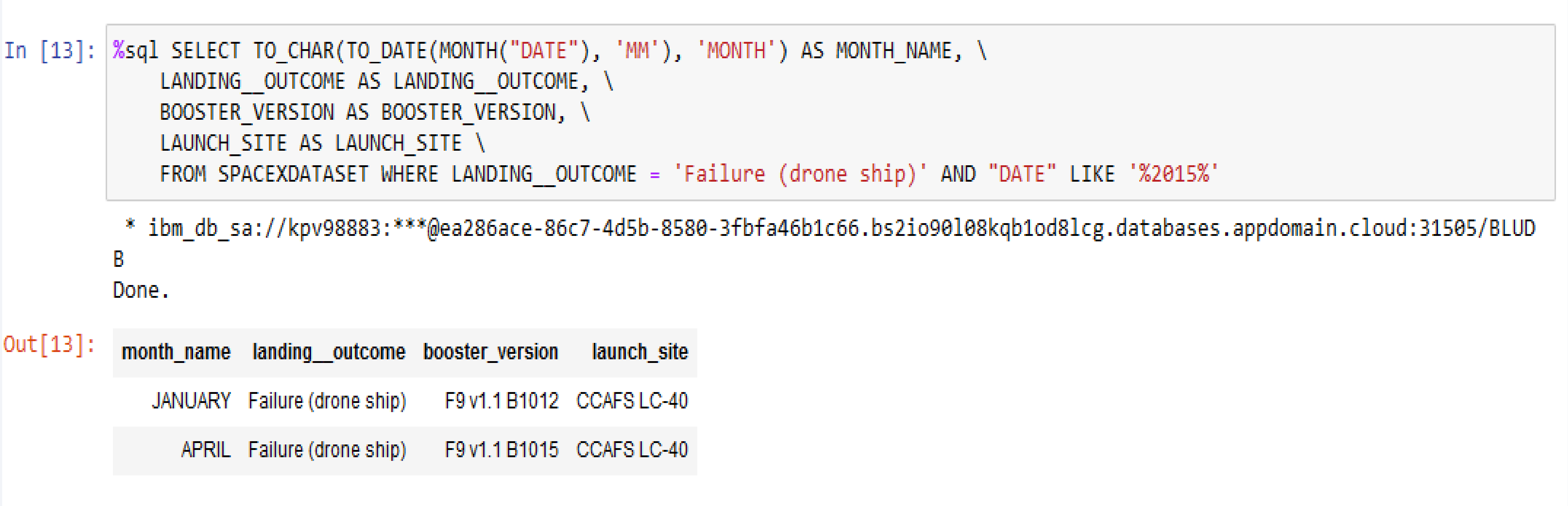
```
In [12]: %sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET)

          * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
         B
         Done.
```

Out[12]:

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

2015 Launch Records

- The query returns the month, landing outcome, booster version and launch site that occurred during 2015



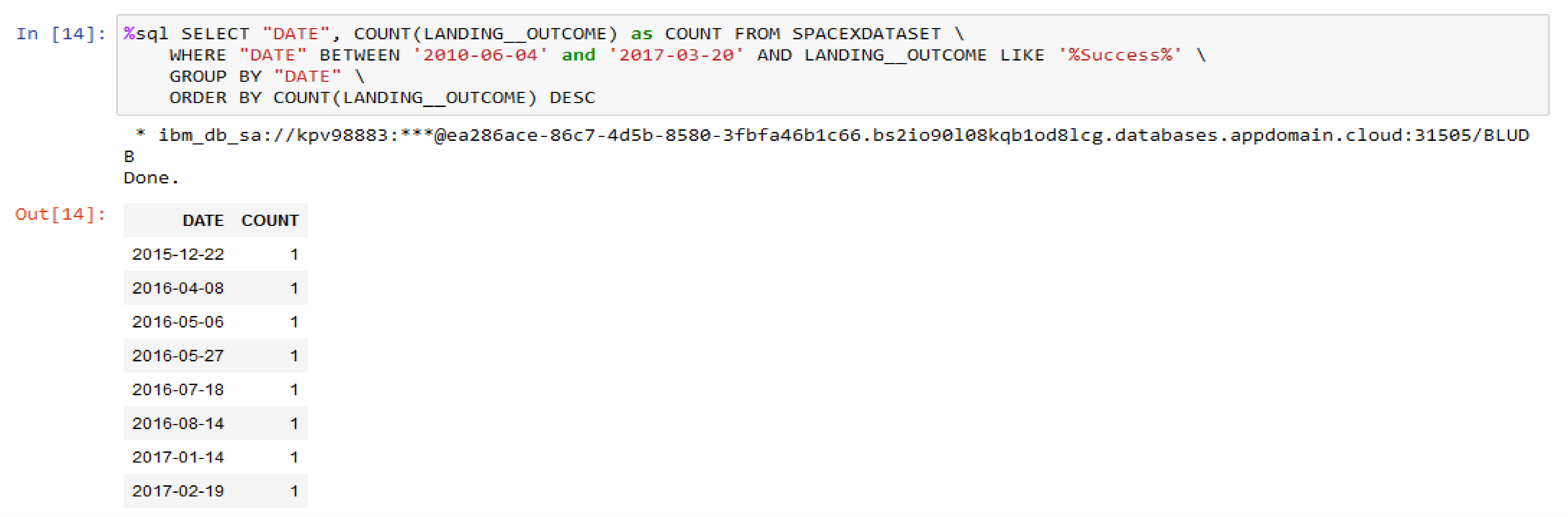
```
In [13]: %sql SELECT TO_CHAR(TO_DATE(MONTH("DATE"), 'MM'), 'MONTH') AS MONTH_NAME, \
             LANDING__OUTCOME AS LANDING__OUTCOME, \
             BOOSTER_VERSION AS BOOSTER_VERSION, \
             LAUNCH_SITE AS LAUNCH_SITE \
             FROM SPACEXDATASET WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND "DATE" LIKE '%2015%'

          * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
         B
         Done.
```

Out[13]:

month_name	landing__outcome	booster_version	launch_site
JANUARY	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
APRIL	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Ranking 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 [14]: %sql SELECT "DATE", COUNT(LANDING__OUTCOME) as COUNT FROM SPACEXDATASET \
             WHERE "DATE" BETWEEN '2010-06-04' and '2017-03-20' AND LANDING__OUTCOME LIKE '%Success%' \
             GROUP BY "DATE" \
             ORDER BY COUNT(LANDING__OUTCOME) DESC

          * ibm_db_sa://kpv98883:***@ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31505/BLUD
         B
         Done.
```

Out[14]:

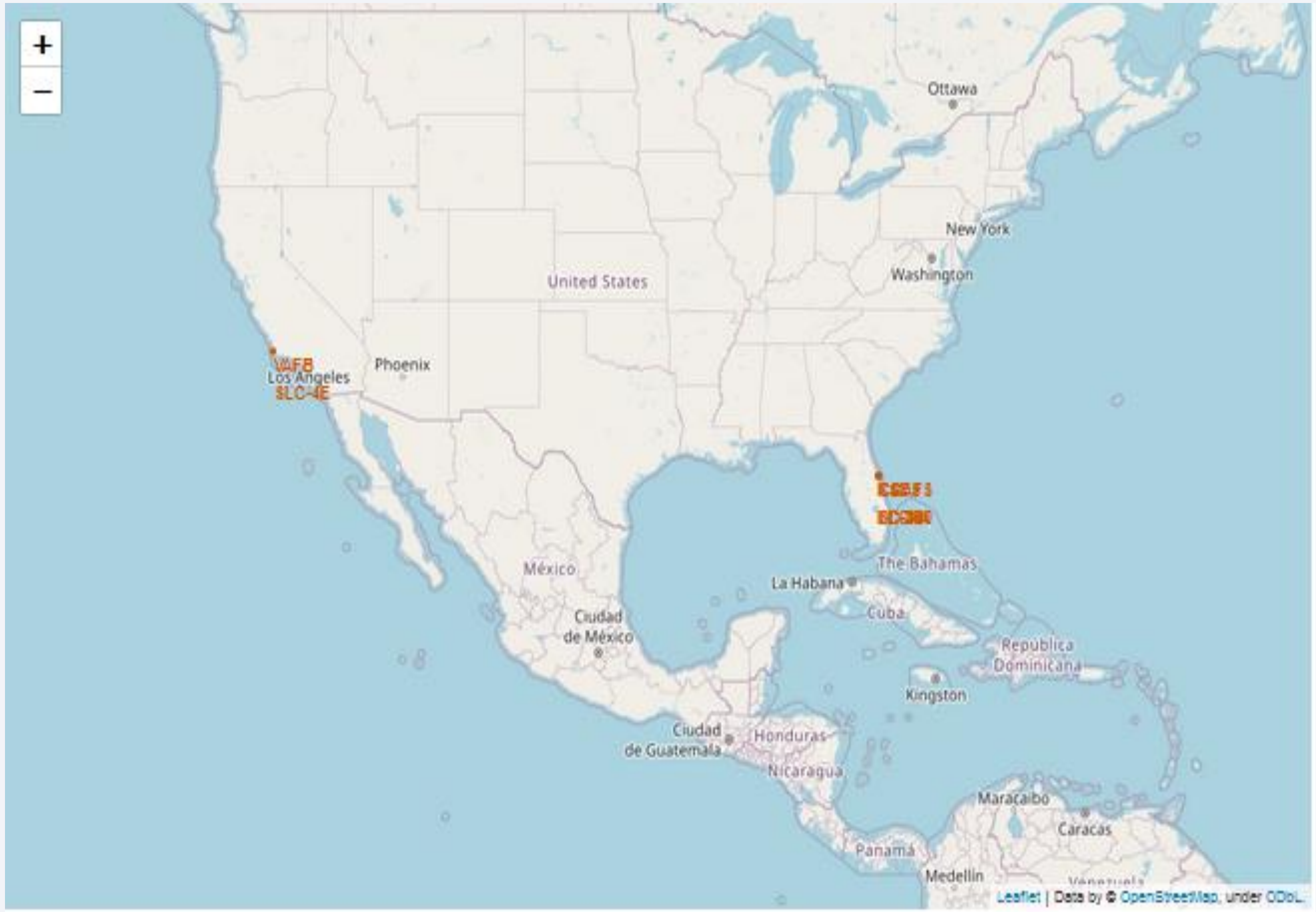
DATE	COUNT
2015-12-22	1
2016-04-08	1
2016-05-06	1
2016-05-27	1
2016-07-18	1
2016-08-14	1
2017-01-14	1
2017-02-19	1

Section 3

Launch Sites Proximities Analysis

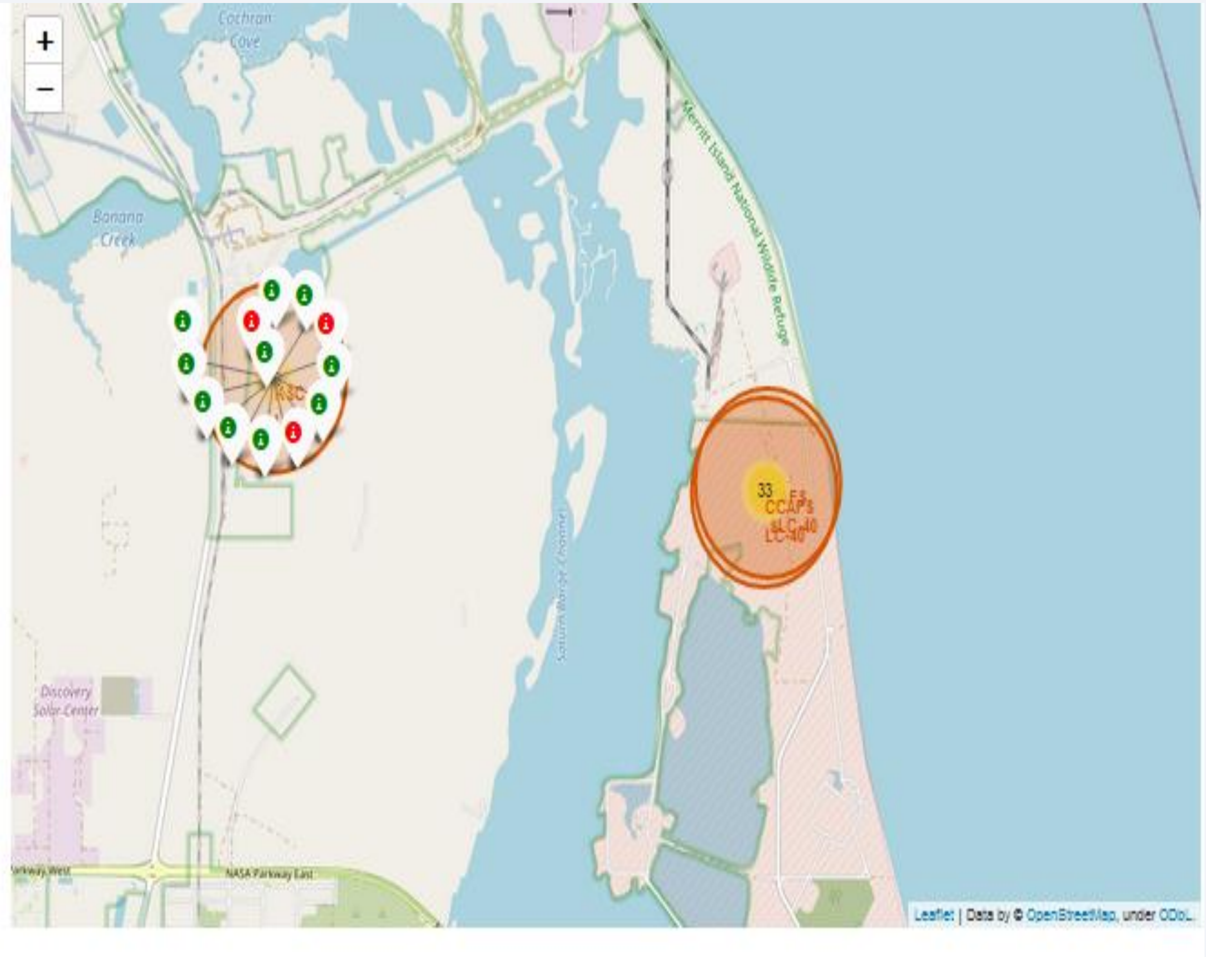
All Launch Sites' Location Markers on a global map

- All launch sites are in very close proximity to the coast.
- Most of Launch sites are in proximity to the Equator line.



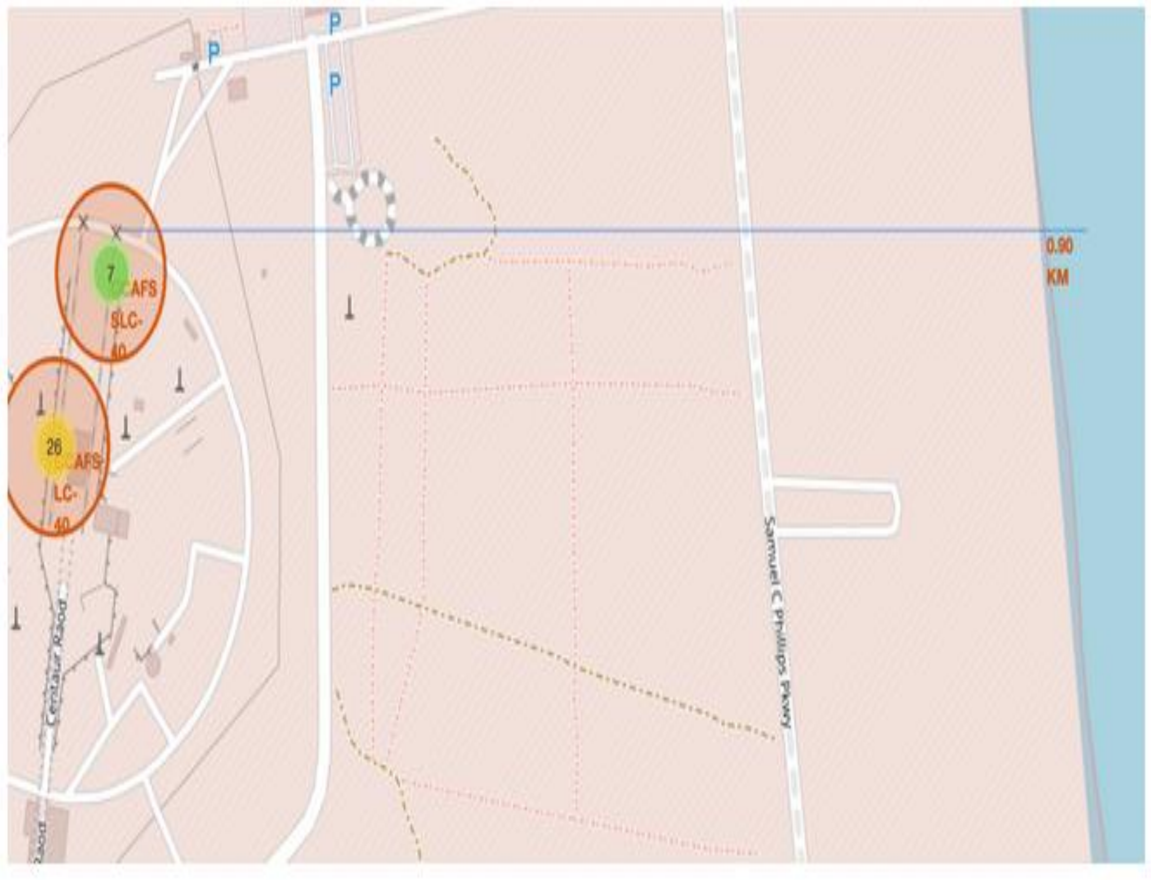
Color-labeled launch outcomes on the Map

- Green Markers represents successful launches
- Red markers represents unsuccessful launches.
- KSC LC-39A has a higher launch success rate.



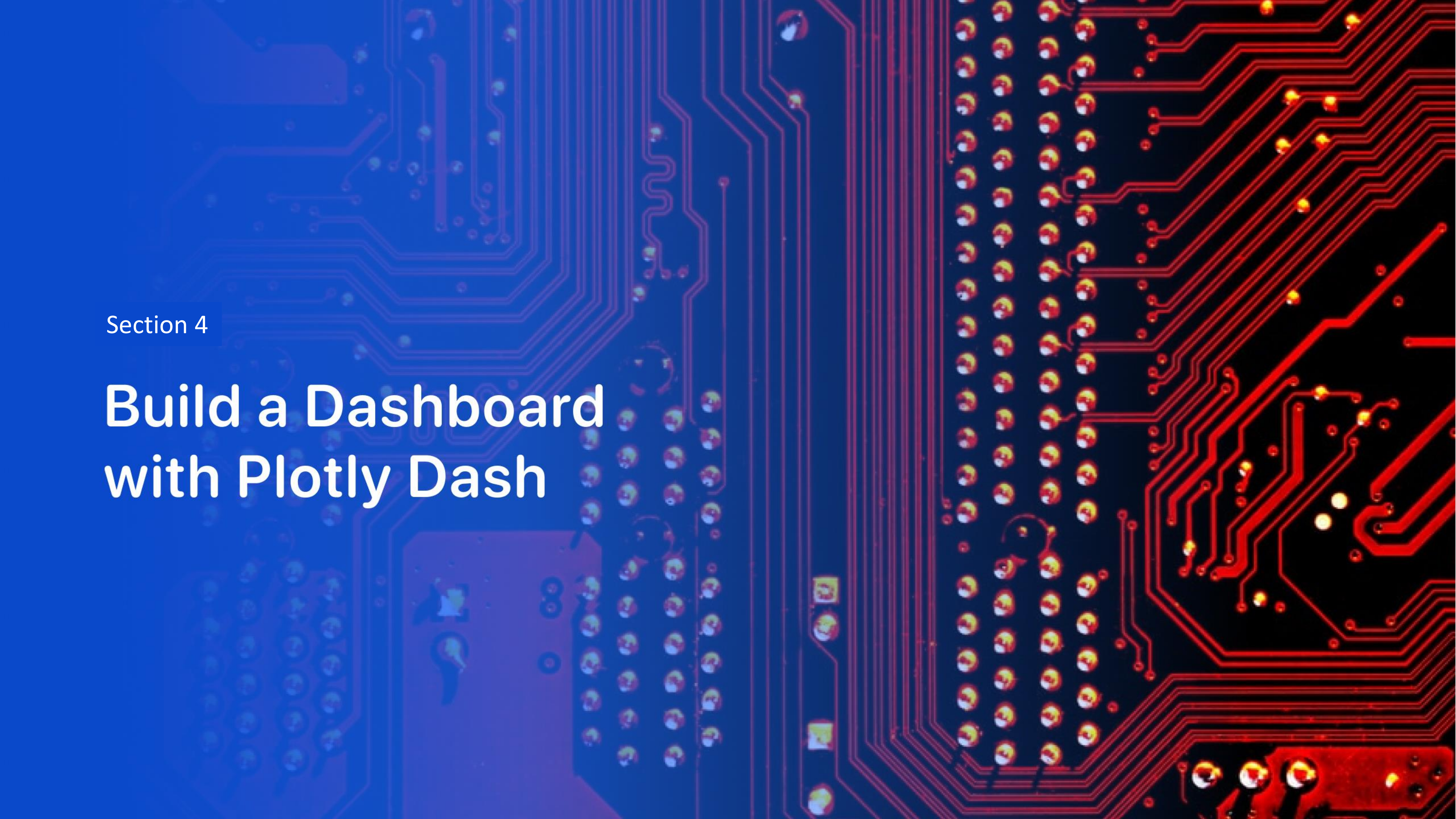
Distance from the launch site KSC LC-39A to its proximities

- Distance lines to the proximities help answer the following questions:
  - Is the launch site in close proximity to railways?
  - Is the launch site in close proximity to highways?
  - Is the launch site in close proximity to coastline?
  - Does the launch site keep certain distance away from cities?

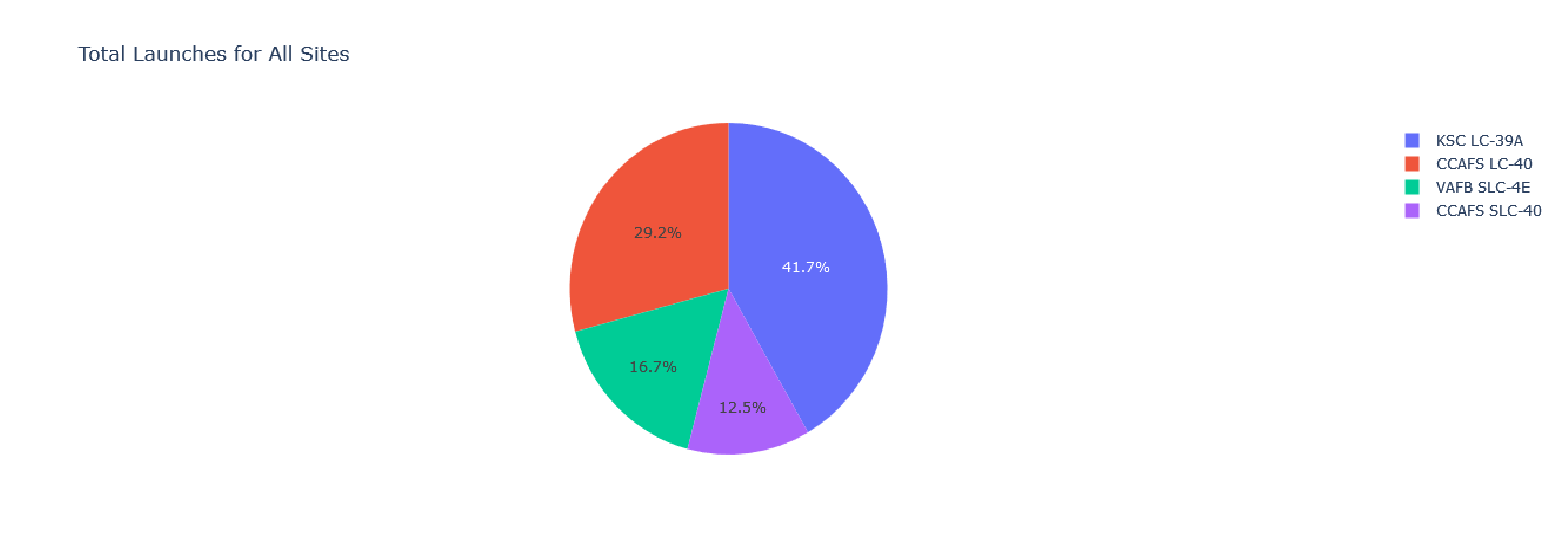


Section 4

Build a Dashboard with Plotly Dash

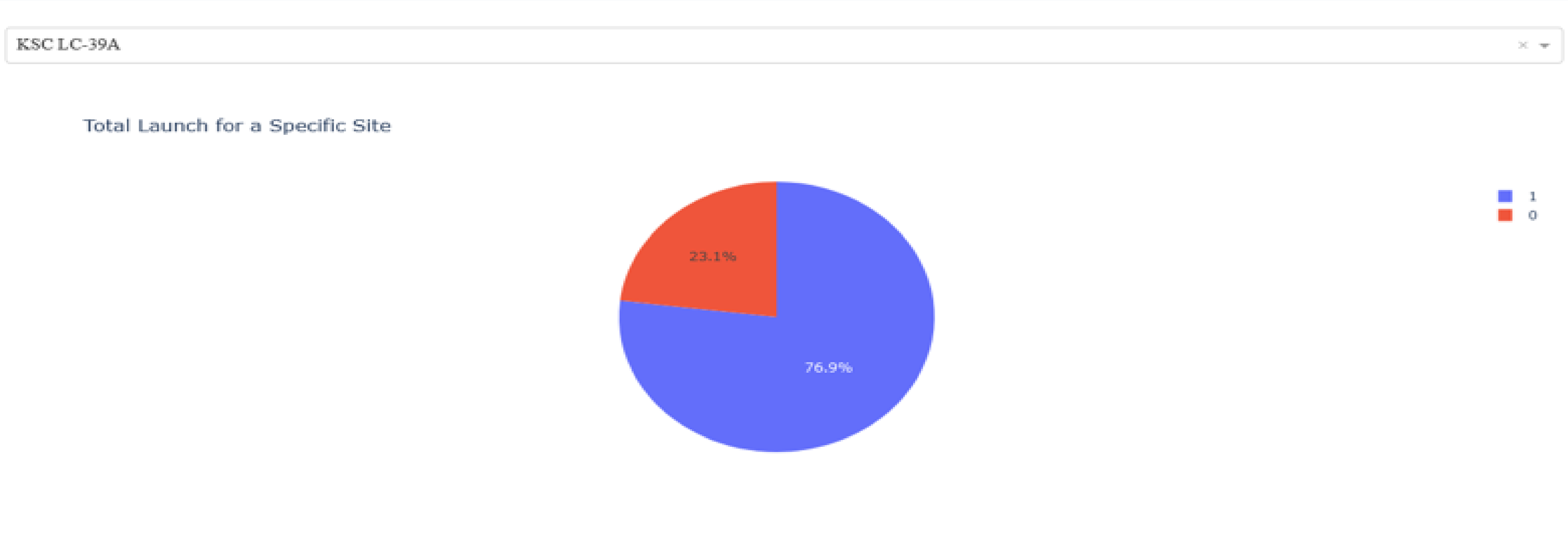


Launch success count for all sites



- The pie chart shows that KSC LC-39A had highest successful launches.

Launch success ratio for KSC LC-39A



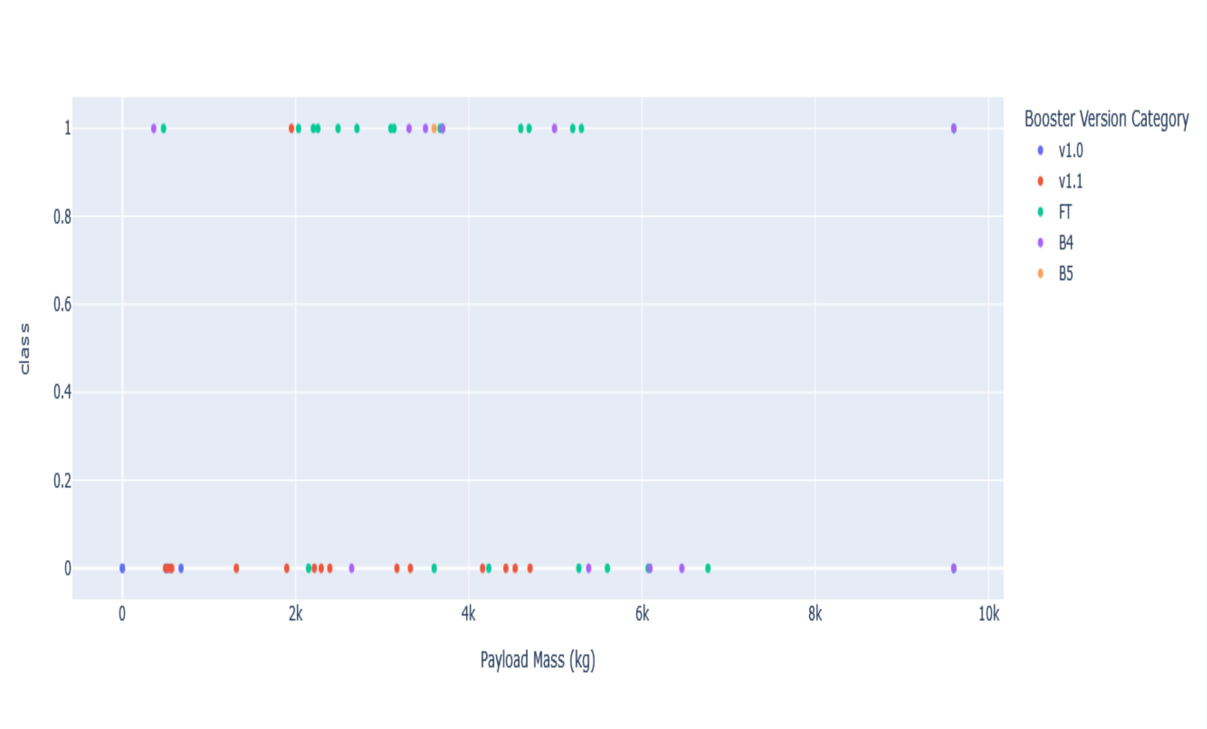
KSC LC-39A

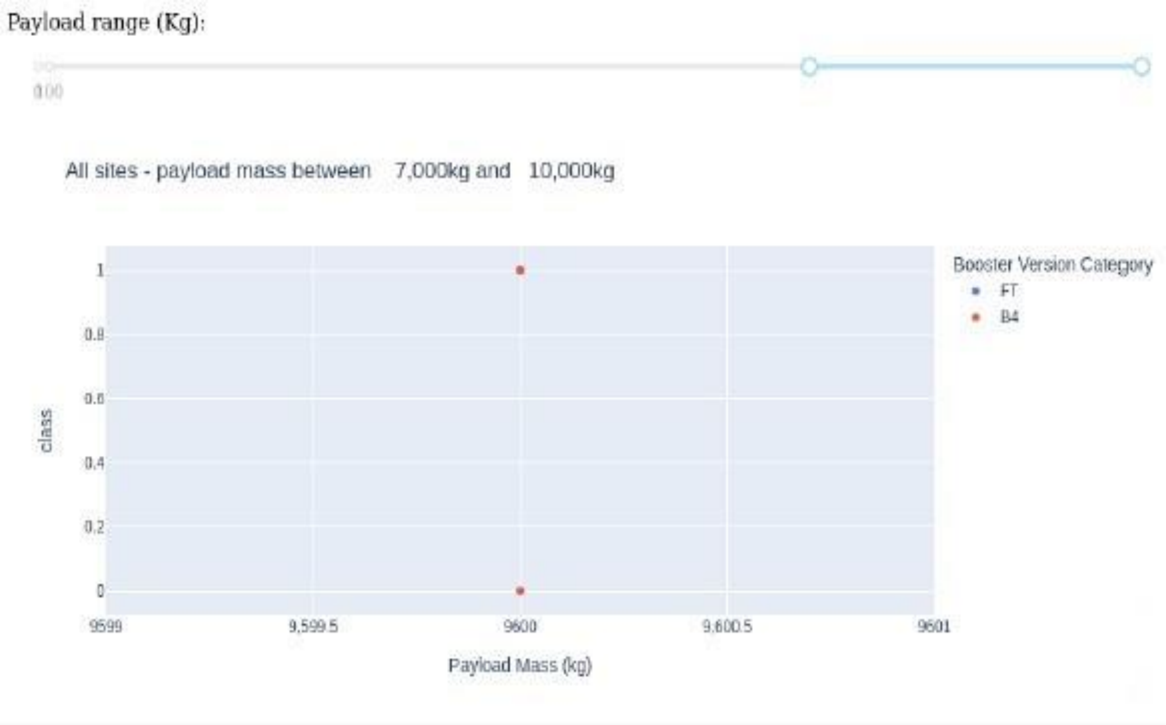
- KSC LC-39A had 76.9% success rate and 23.1% failure rate.

Payload vs. Launch Outcome scatter plot for all sites

- The charts show that payloads between 0 and 10000 kg for all sites

- The charts show that payloads between 7000 and 10000 kg for all sites





Payload range (Kg):

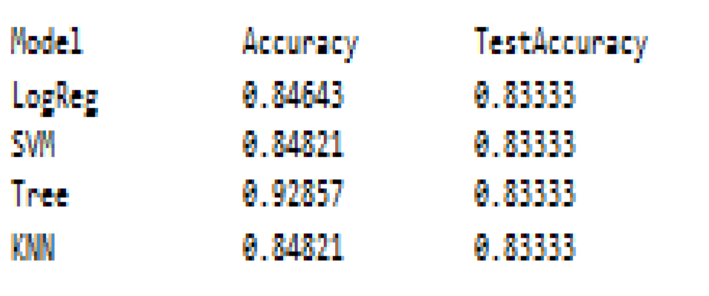
All sites - payload mass between 7,000kg and 10,000kg

Section 5

Predictive Analysis (Classification)

Classification Accuracy

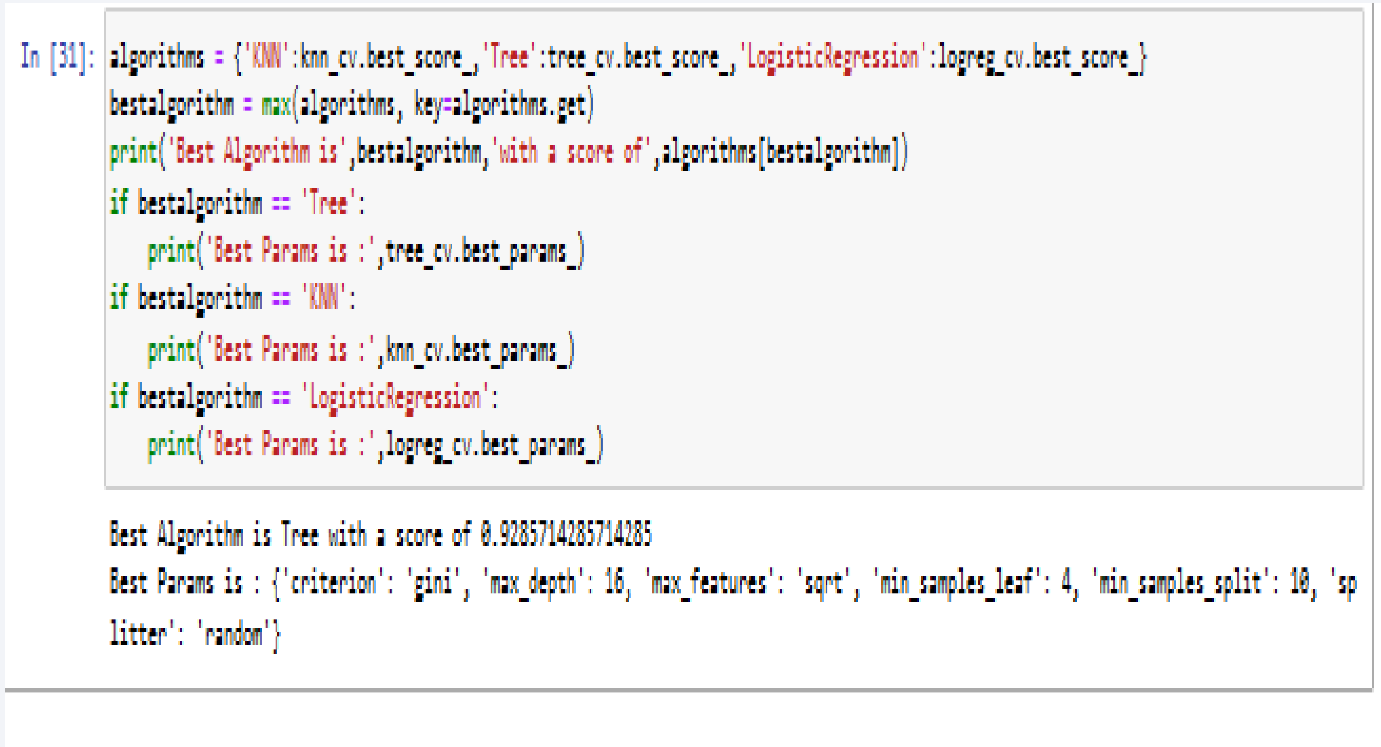
- Based on test accuracy, all methods performed the same.



```
Model      Accuracy    TestAccuracy
LogReg     0.84643     0.83333
SVM        0.84821     0.83333
Tree       0.92857     0.83333
KNN        0.84821     0.83333
```

- Same Test Set scores may be due to the small test sample size (18 samples)

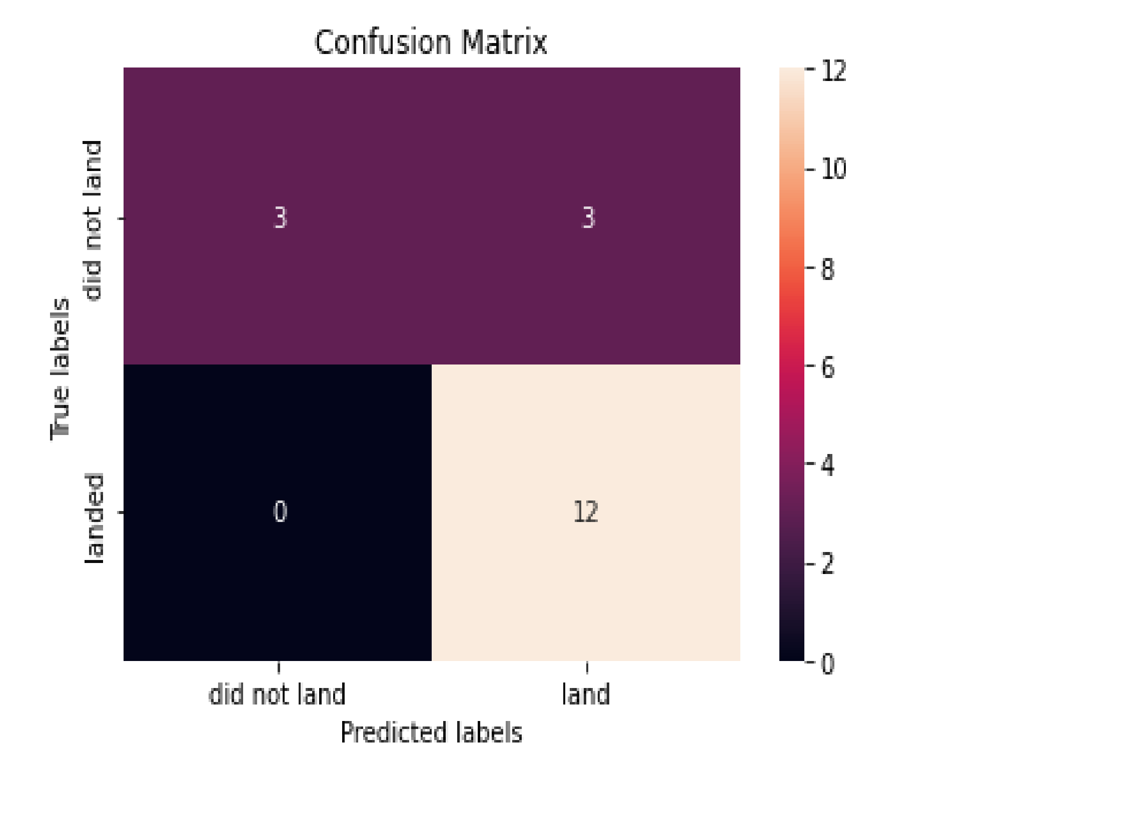
- The decision tree model had the highest classification accuracy.



```
In [31]: algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_}
         bestalgorithm = max(algorithms, key=algorithms.get)
         print('Best Algorithm is',bestalgorithm,'with a score of',algorithms[bestalgorithm])
         if bestalgorithm == 'Tree':
             print('Best Params is :',tree_cv.best_params_)
         if bestalgorithm == 'KNN':
             print('Best Params is :',knn_cv.best_params_)
         if bestalgorithm == 'LogisticRegression':
             print('Best Params is :',logreg_cv.best_params_)

         Best Algorithm is Tree with a score of 0.9285714285714285
         Best Params is : {'criterion': 'gini', 'max_depth': 16, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 10, 'sp
         litter': 'random'}
```

Confusion Matrix

- Confusion matrix of Decision Tree classifier has a big number of true positive and true negative compared to the false ones.



Conclusions

- Decision Tree Model is the best algorithm.
- KSC LC-39A has the highest success rate of the launches from all the sites
- OrbitsES-L1, GEO,HEO and SSO have 100% success rate.
- Launch success rate started to increase in 2013 till 2020
- The larger the flight number at a launch site, the greater the success rate
- Launch sites are in close proximity to the coast and all sites are in proximity to the Equator line.

Appendix

Special Thanks to:

- Coursera Instructors
- IBM

Thank you!