IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Gary Hsieh
1 July 2022

Outline

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

Executive Summary

- Summary of methodologies

➢ Data Collection API

➢ Data Collection with Web Scraping

➢ Data Wrangling

➢ Exploratory Data Analysis (EDA) with SQL

➢ EDA with Data Visualization

➢ Interactive Visual Analytics with Folium

➢ Predictive analysis (Classification)

- Summary of all results

➢ Exploratory Data Analysis Result

➢ Interactive Analytics in Screenshots

➢ Predictive Analytics result from Machine Learning

Introduction

- Project background and context

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. 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 you want to find answers

○ Identifying factors that influence the landing outcome

○ The relationship between each variables and how it is affecting the outcome

○ The best condition needed to increase the probability of successful landing

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX REST API and web scraping from Wikipedia
- Perform data wrangling
  - Data was processed using one-hot encoding for 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
  - LR, KNN, SVM, DT models have been built and evaluated for the best classifier.

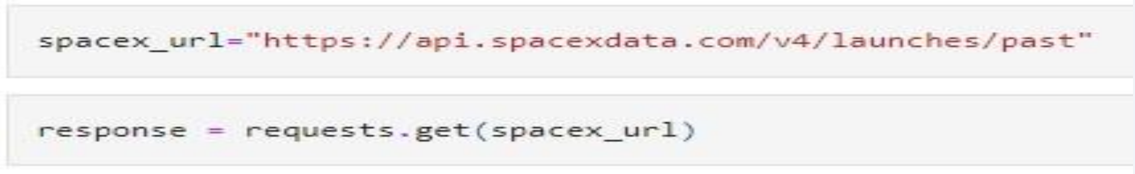
Data Collection

- Data collection is the process of gathering and measuring information on targeted variables in an established system, which then enables one to answer relevant questions and evaluate outcomes. As mentioned, the dataset was collected by REST API and Web Scraping from Wikipedia
- For Rest API, we started by using the Get request. Then, we decoded the response content as Json and turn it into a pandas dataframe using Json_normalize(). We then cleaned the data, checked for missing values and fill with whatever needed.
- For Web Scraping, we used the Beautifulsoup to extract the launch records as HTML table, parse the table and convert it to a Pandas dataframe for further analysis.

Data Collection – SpaceX API

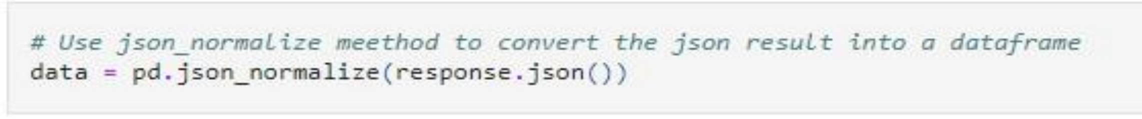
- Request rocket launch data from SpaceX API
- Decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`
- Do Data Cleaning
- Deal with missing values

https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/Data%20Collection%20API.ipynb

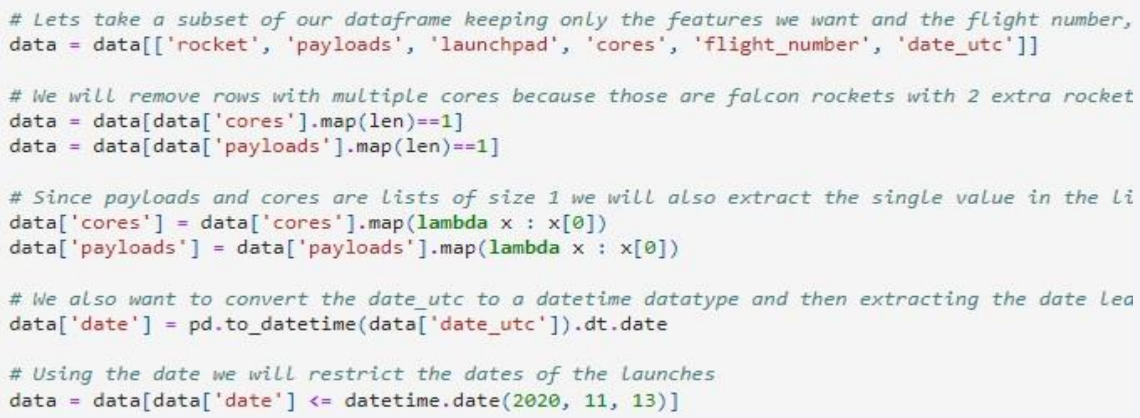


```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

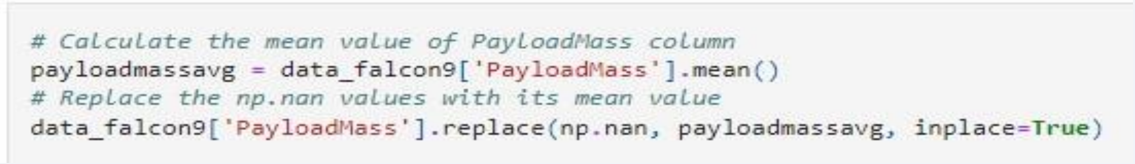
```
response = requests.get(spacex_url)
```



```
# Use json_normalize meethod to convert the json result into a dataframe
data = pd.json_normalize(response.json())
```



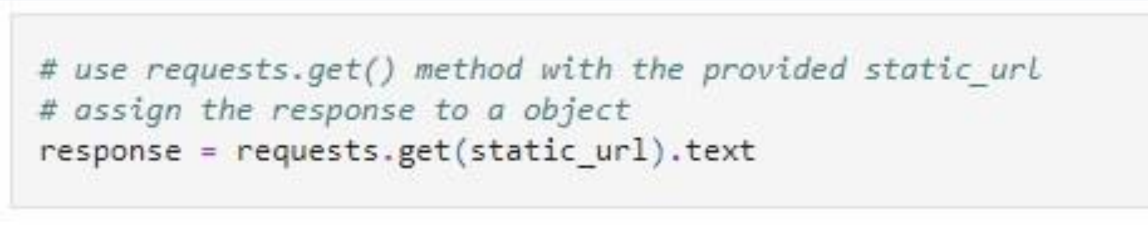
```
# Lets take a subset of our dataframe keeping only the features we want and the flight number,
data = data[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]

# We will remove rows with multiple cores because those are falcon rockets with 2 extra rocket
data = data[data['cores'].map(len)==1]
data = data[data['payloads'].map(len)==1]

# Since payloads and cores are lists of size 1 we will also extract the single value in the li
data['cores'] = data['cores'].map(lambda x : x[0])
data['payloads'] = data['payloads'].map(lambda x : x[0])

# We also want to convert the date_utc to a datetime datatype and then extracting the date lea
data['date'] = pd.to_datetime(data['date_utc']).dt.date

# Using the date we will restrict the dates of the launches
data = data[data['date'] <= datetime.date(2020, 11, 13)]
```



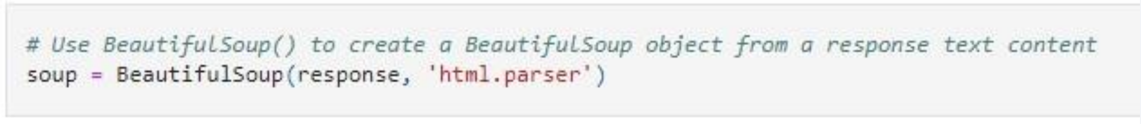
```
# Calculate the mean value of PayloadMass column
payloadmassavg = data_falcon9['PayloadMass'].mean()
# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(np.nan, payloadmassavg, inplace=True)
```

Data Collection - Scraping

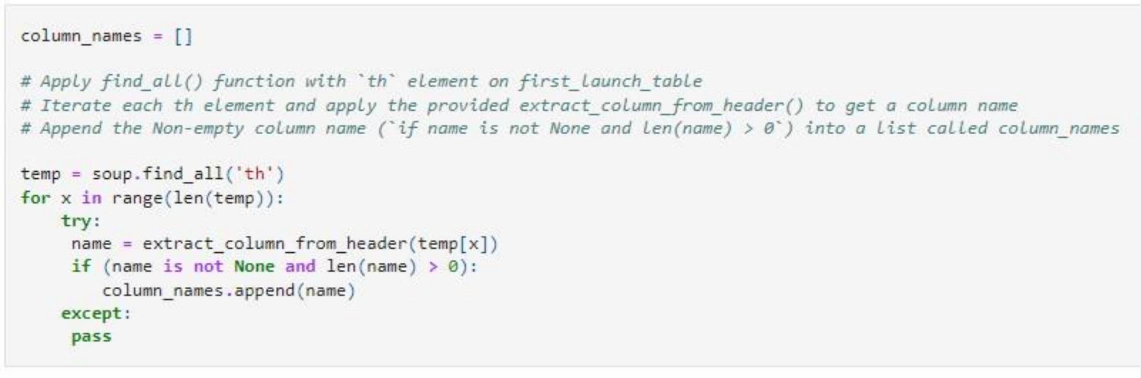
- Request the Falcon9 Launch Wiki page from its URL
- Create BeautifulSoup object from the HTML response
- Extract all column/variable names from the HTML table header



```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url).text
```



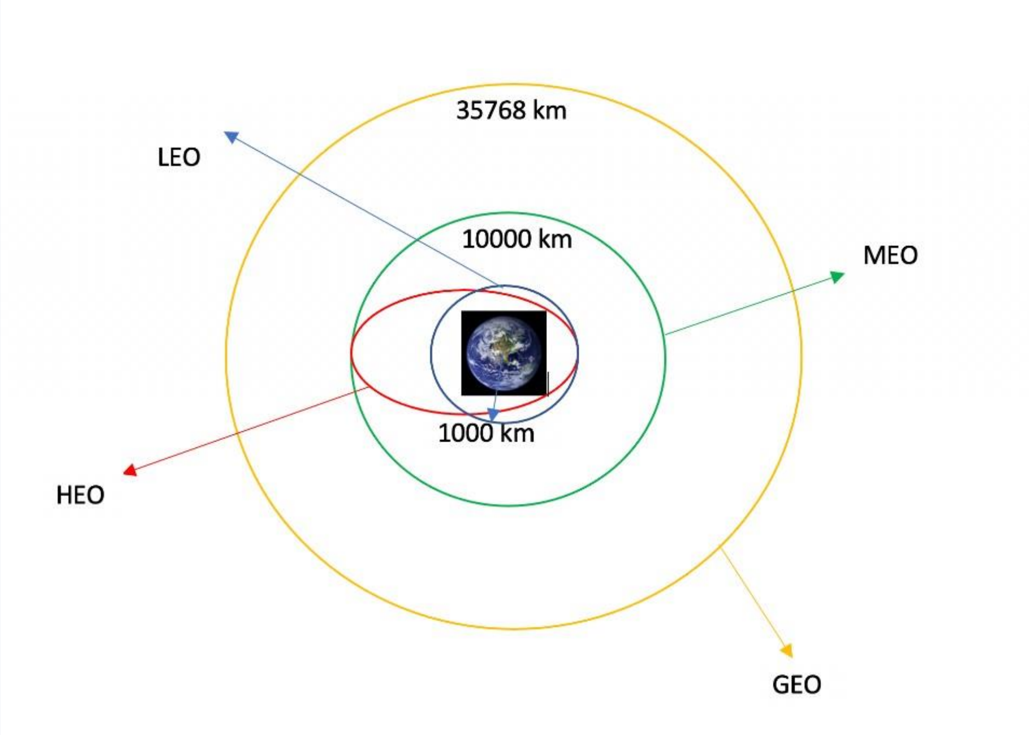
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response, 'html.parser')
```



```
column_names = []

# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided extract_column_from_header() to get a column name
# Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names

temp = soup.find_all('th')
for x in range(len(temp)):
    try:
     name = extract_column_from_header(temp[x])
     if (name is not None and len(name) > 0):
        column_names.append(name)
    except:
     pass
```

https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

Data Wrangling



Calculate the number of launches on each site

Calculate the number and occurence of mission outcome per orbit type

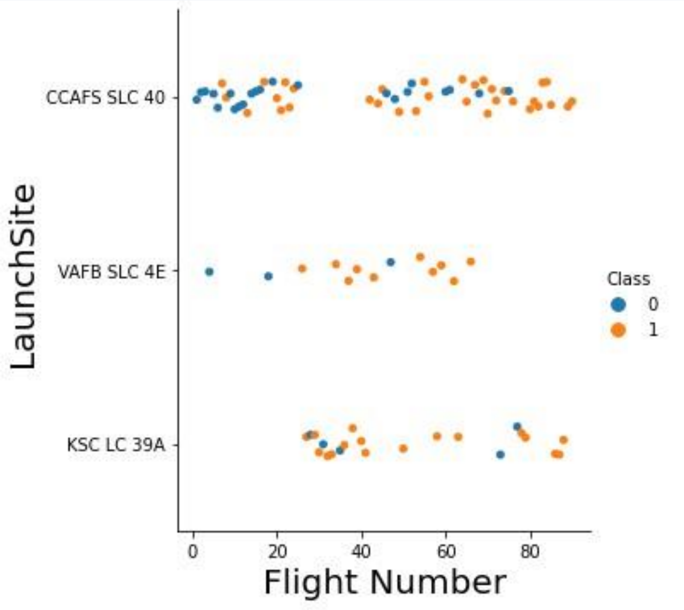
Create a landing outcome label from Outcome column

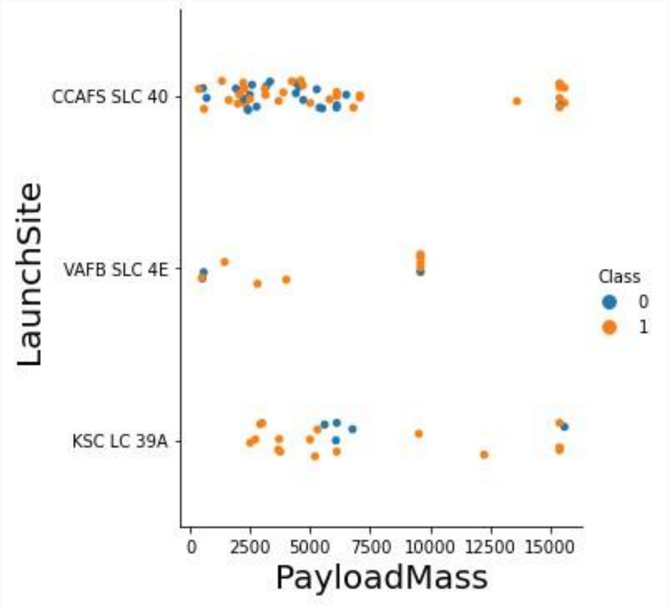
https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/EDA.ipynb

EDA with Data Visualization

Scatter plots were plotted to find the relationship between attributes

- Flight Number and Launch Site
- Payload and Launch Site
- Flight Number and Orbit type
- Payload and Orbit type

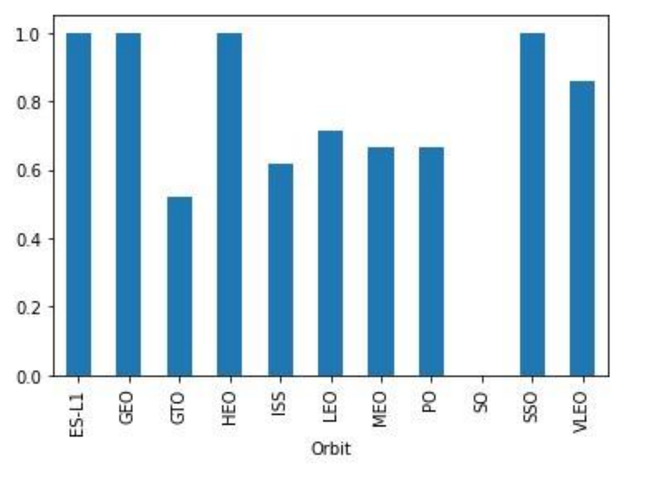




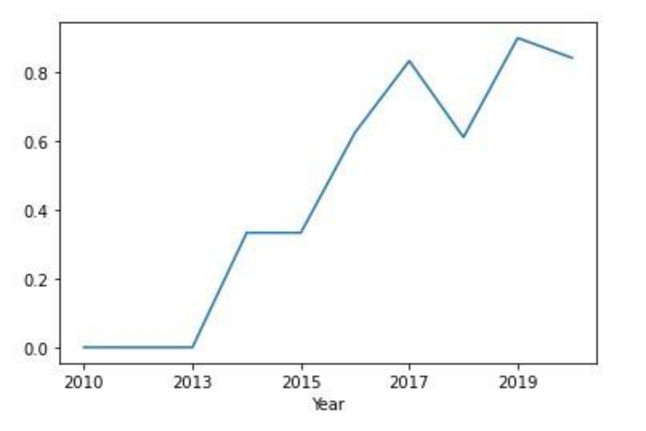
https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/EDA%20with%20Data%20Visualization.ipynb

EDA with Data Visualization

- Using Bar chart to find which orbits have high sucess rate.



- Using Line chart to get the average launch success trend.



https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

SQL queries performed:

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- 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 successful landing outcome in ground pad was acheived.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass
- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- 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

https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

Mark all launch sites using site's latitude and longitude coordinates and use folium.Circle to add highlighted circle areas with text label on specific coordinates

Enhance the map by adding the launch outcomes for each site with red and green markers, and see which sites have high success rates using MarkerCluster object

MousePosition were added to get coordinate for a mouse over a point on the map. As such, while you are exploring the map, you can easily find the coordinates of any points of interests

https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/Interactive%20Visual%20Analytics%20with%20Folium%20lab.ipynb

Build a Dashboard with Plotly Dash

- We build a Plotly Dash application for users to perform interactive visual analytics on SpaceX launch data in real-time.
- With the dashboard, we are able to use it to analyze SpaceX launch data, and answer the following questions:

1. Which site has the largest successful launches?
2. Which site has the highest launch success rate?
3. Which payload range(s) has the highest launch success rate?
4. Which payload range(s) has the lowest launch success rate?
5. Which F9 Booster version (`v1.0`, `v1.1`, FT, B4, B5, etc.) has the highest launch success rate?

https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

Build the Model

- Load the dataframe
- Standardize the data
- split the data into training and test data
- Create the model object

Evaluate the Model

- Calculate the accuracy for each model
- Examining the confusion matrix

Improve the Model

- create a GridSearchCV object
- Fit the object to find the best parameters from the dictionary parameters

Find the Best Model

- The model with the best accuracy score

https://github.com/Gary199309/Applied_Data_Science_Capstone/blob/master/Machine%20Learning%20Prediction.ipynb

Results

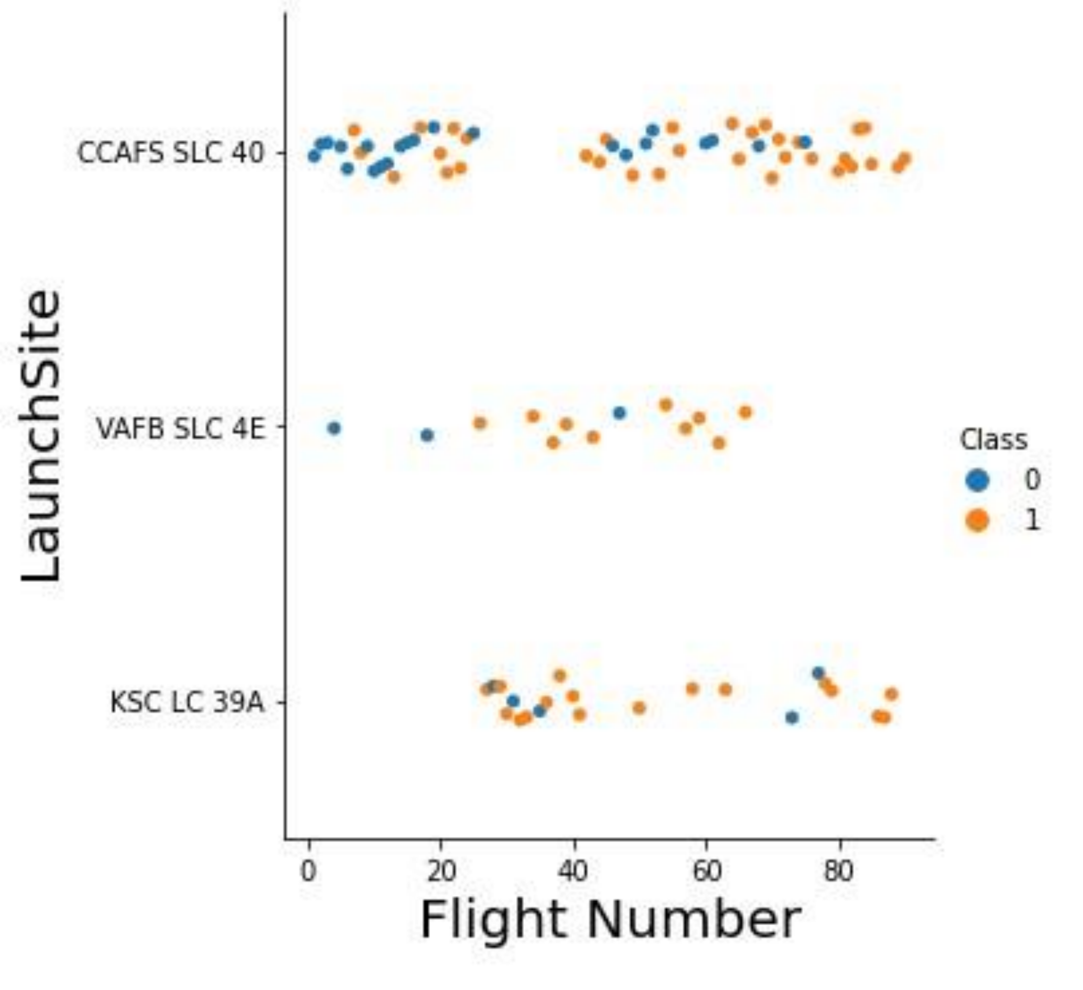
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

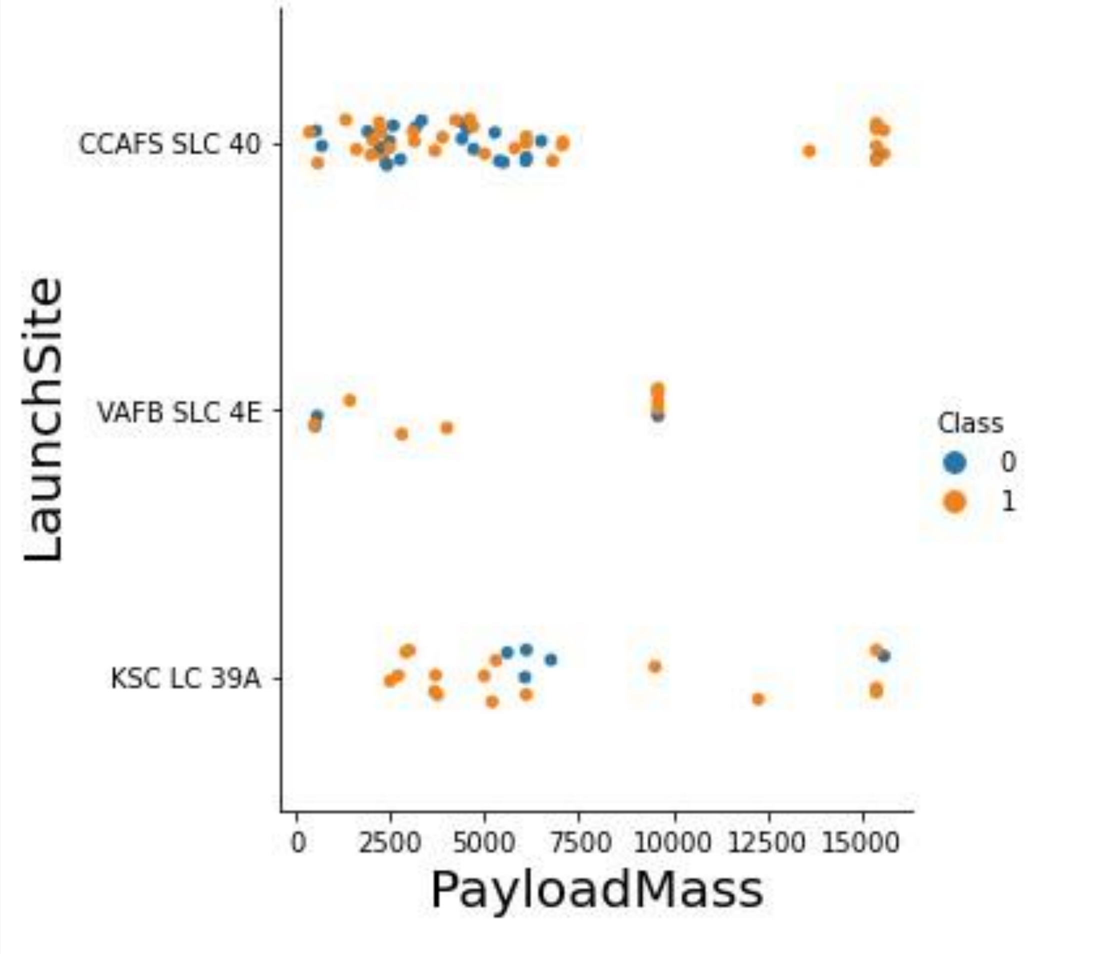
Flight Number vs. Launch Site

We see that as the flight number increases, the first stage is more likely to land successfully.



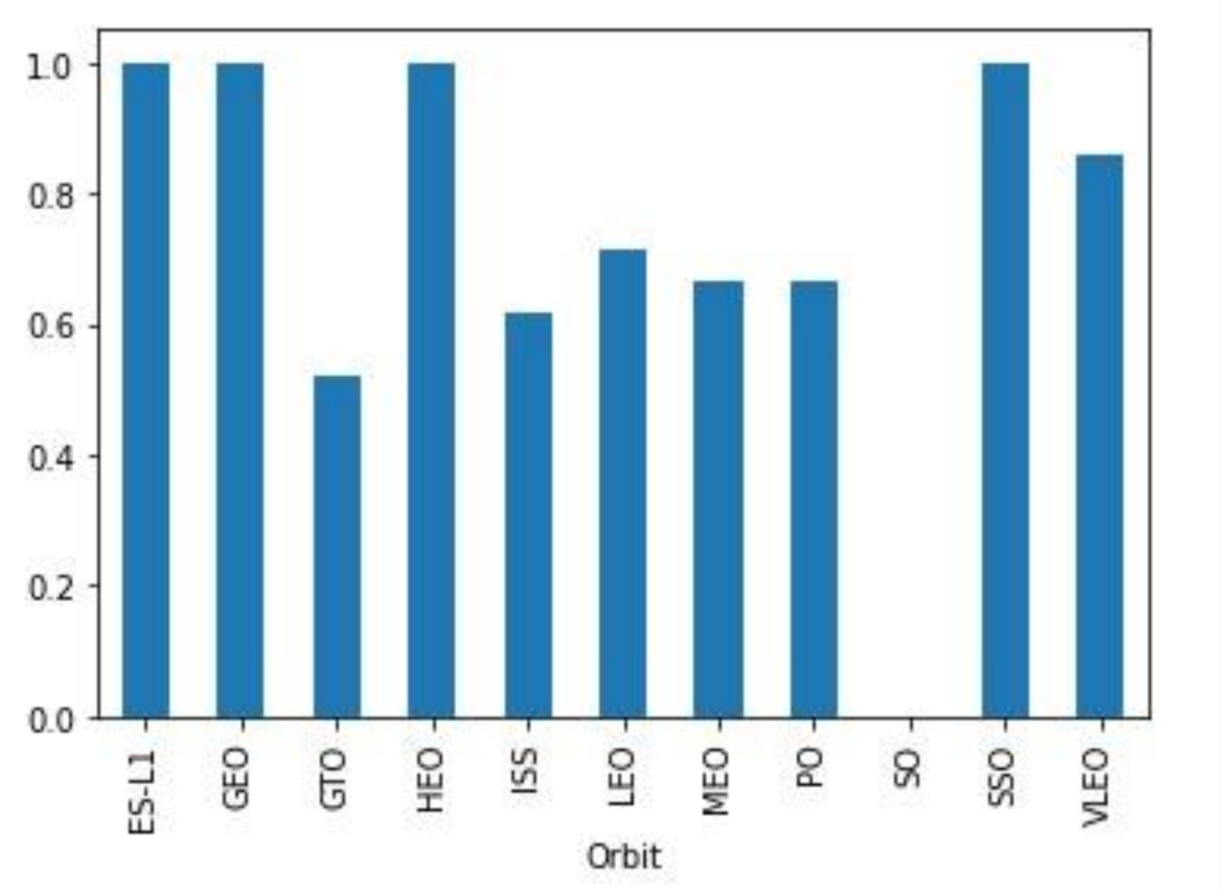
Payload vs. Launch Site

- For the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).



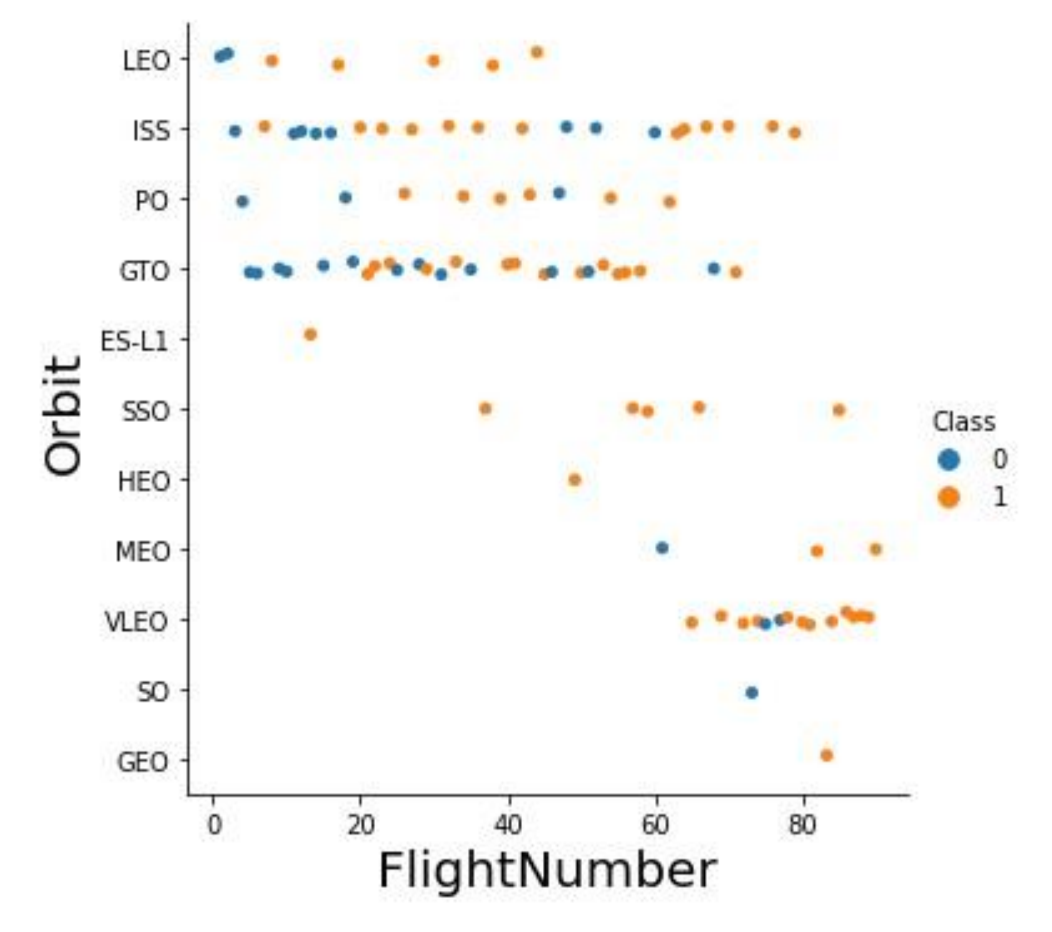
Success Rate vs. Orbit Type

- Orbit Type ES-L1, GEO, HEO, SSO, have high sucess rate.



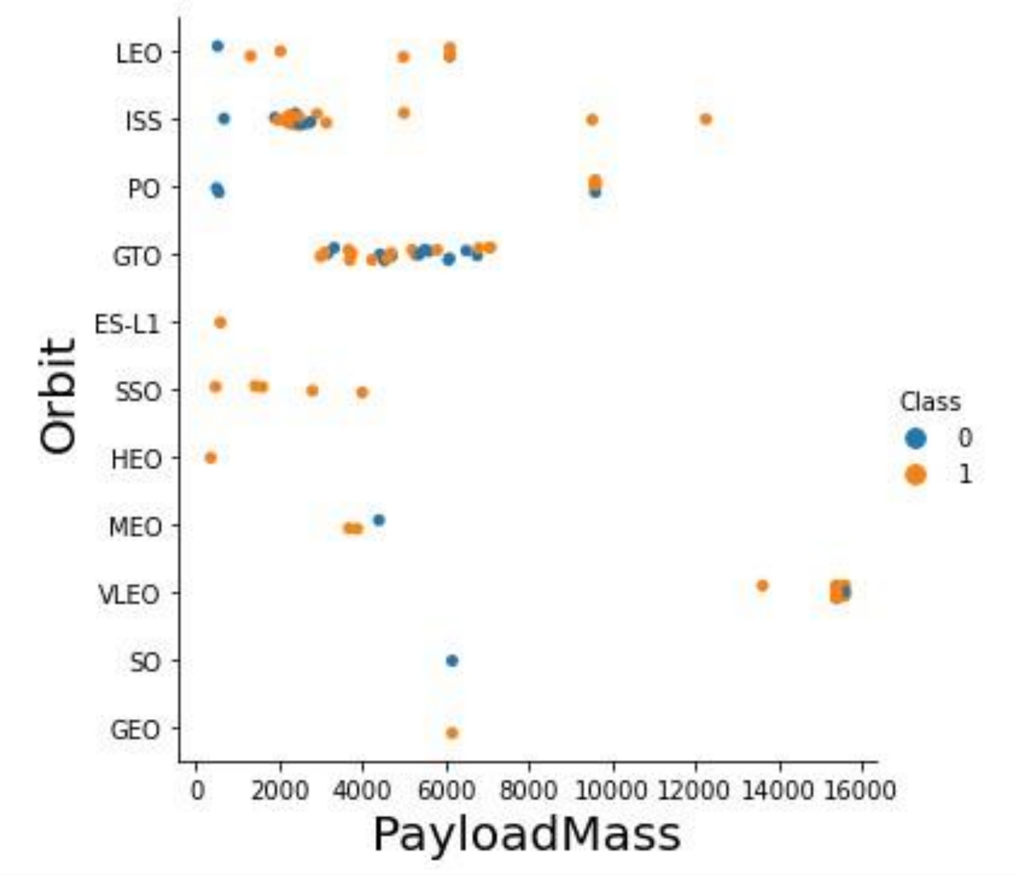
Flight Number vs. Orbit Type

- We can see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



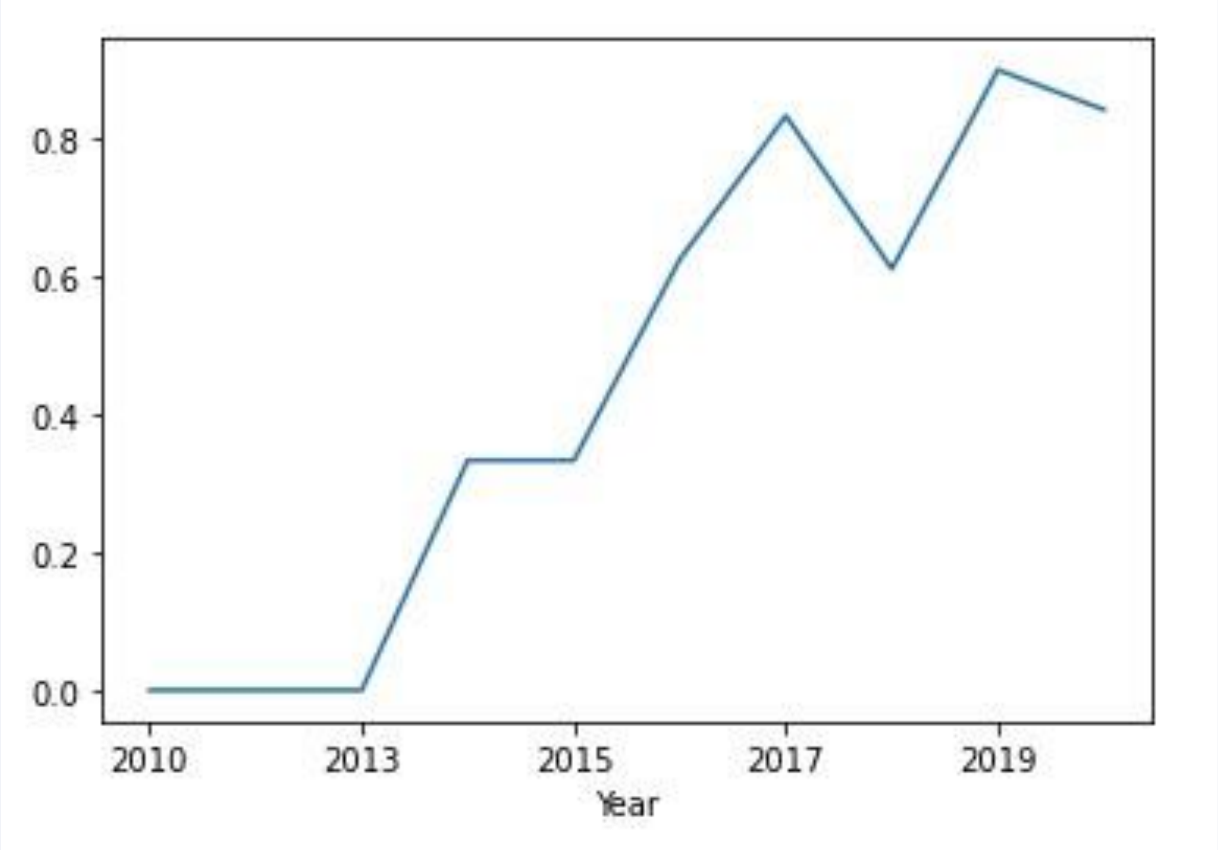
Payload vs. Orbit Type

- 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.



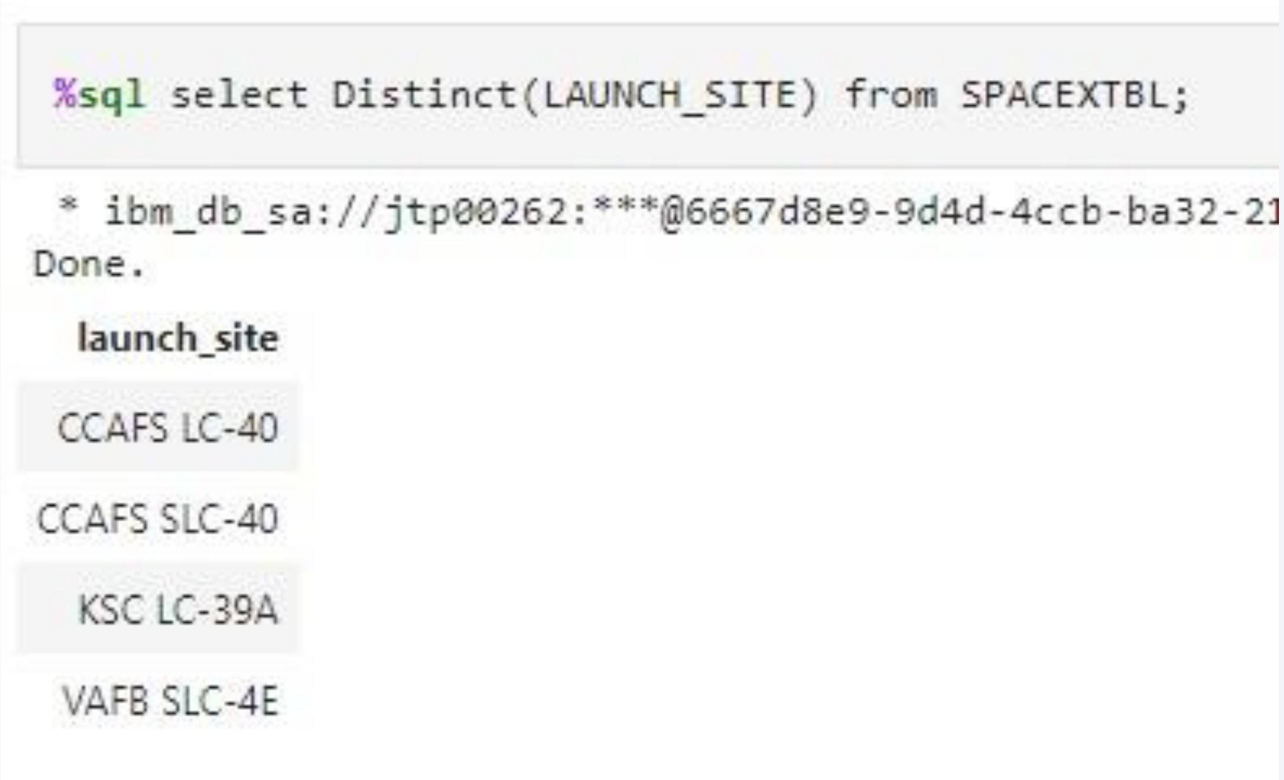
Launch Success Yearly Trend

- We can observe that the sucess rate since 2013 kept increasing till 2020



All Launch Site Names

- Find the names of the unique launch sites



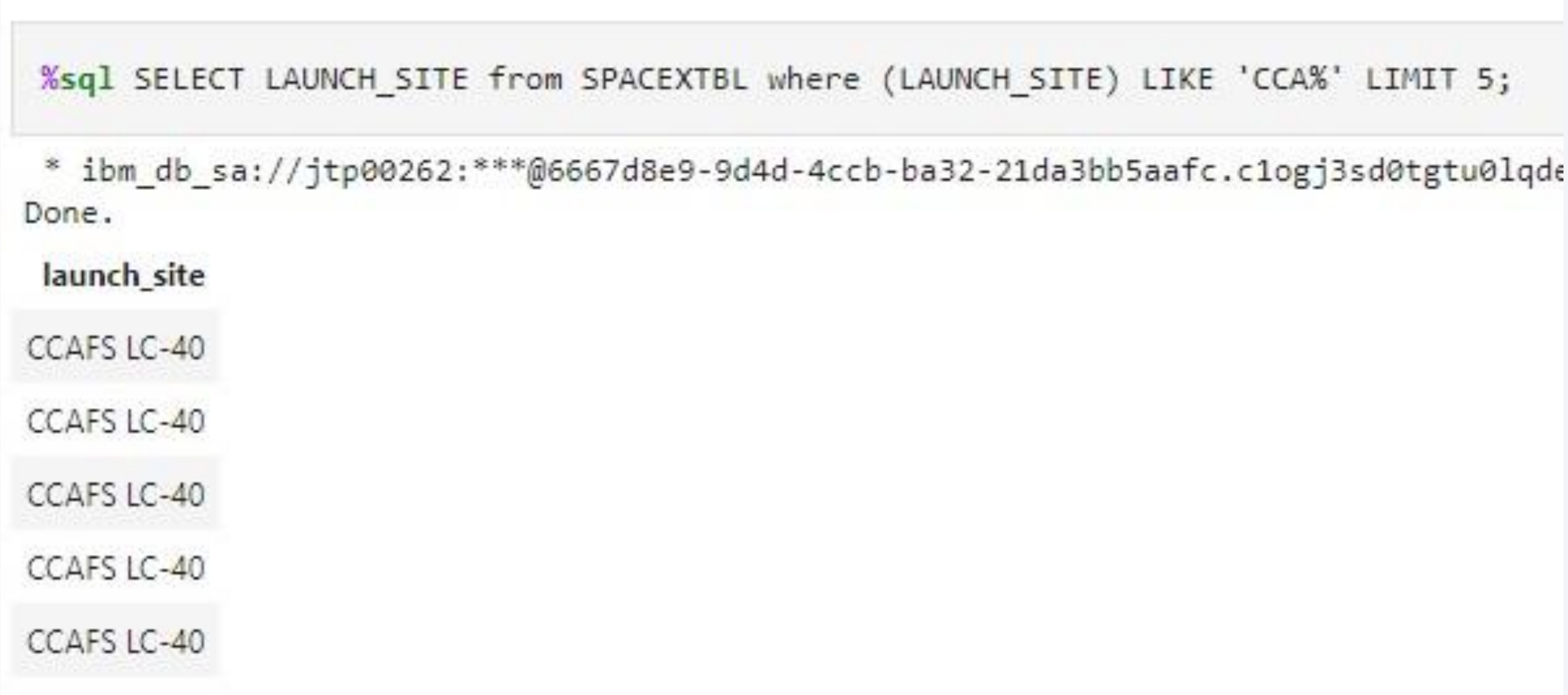
```
%sql select Distinct(LAUNCH_SITE) from SPACEXTBL;
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21
Done.
```

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

Launch Site Names Begin with 'CCA'

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



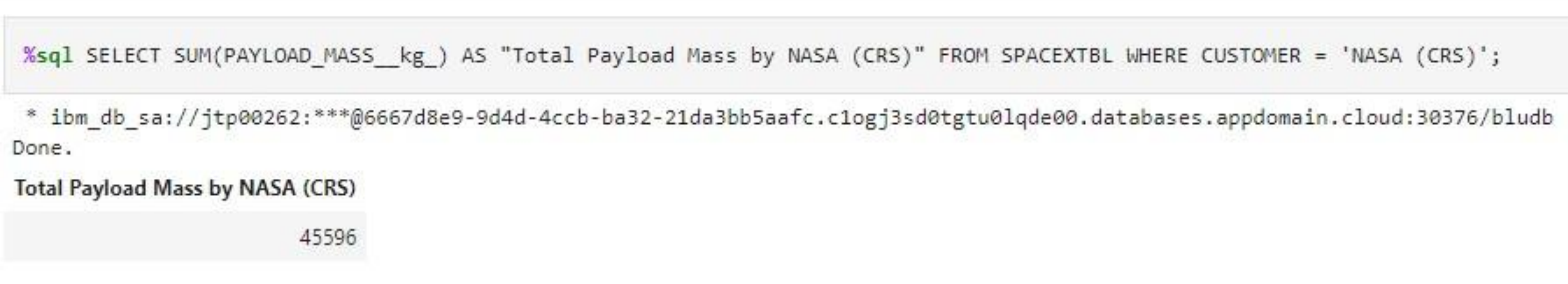
```
%sql SELECT LAUNCH_SITE from SPACEXTBL where (LAUNCH_SITE) LIKE 'CCA%' LIMIT 5;
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde
Done.
```

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA



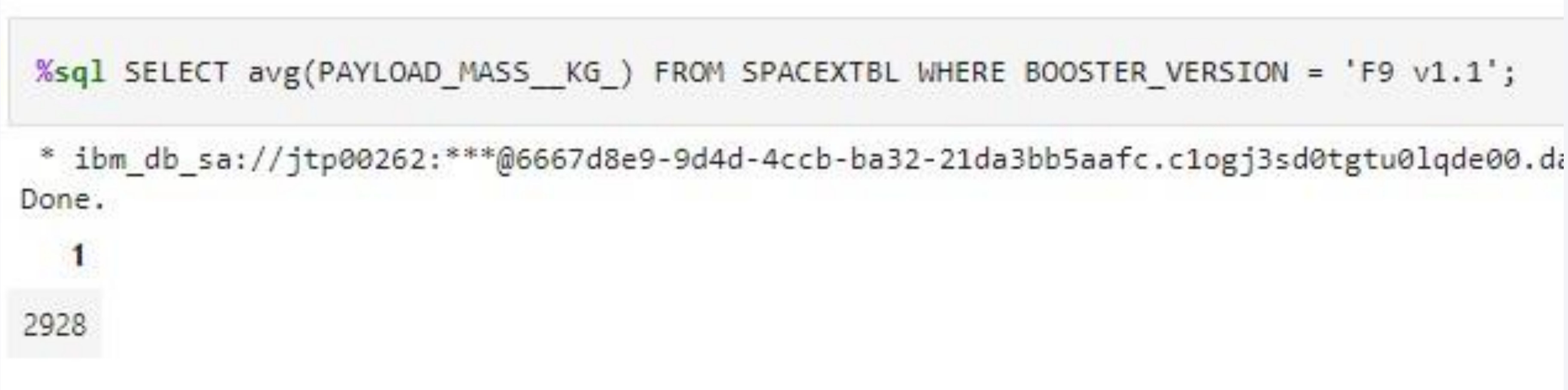
```
%sql SELECT SUM(PAYLOAD_MASS__kg_) AS "Total Payload Mass by NASA (CRS)" FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)';
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Total Payload Mass by NASA (CRS)
45596

Average Payload Mass by F9 v1.1

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

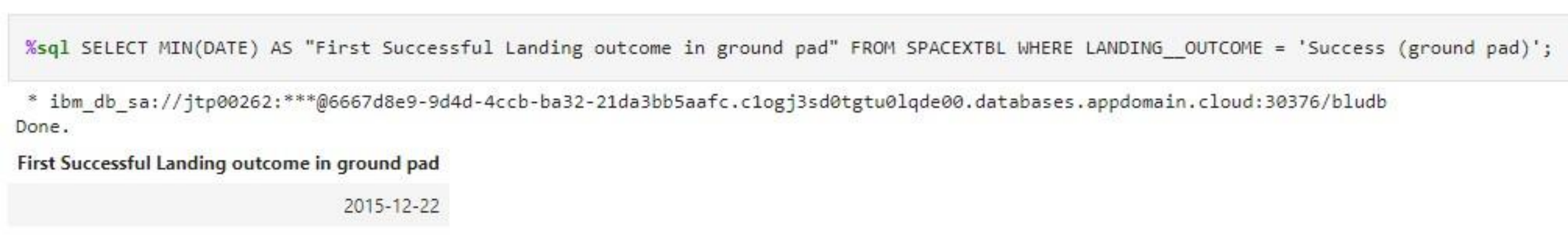


```
%sql SELECT avg(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.da
Done.
   1
2928
```

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- *Use min() function to find the result*



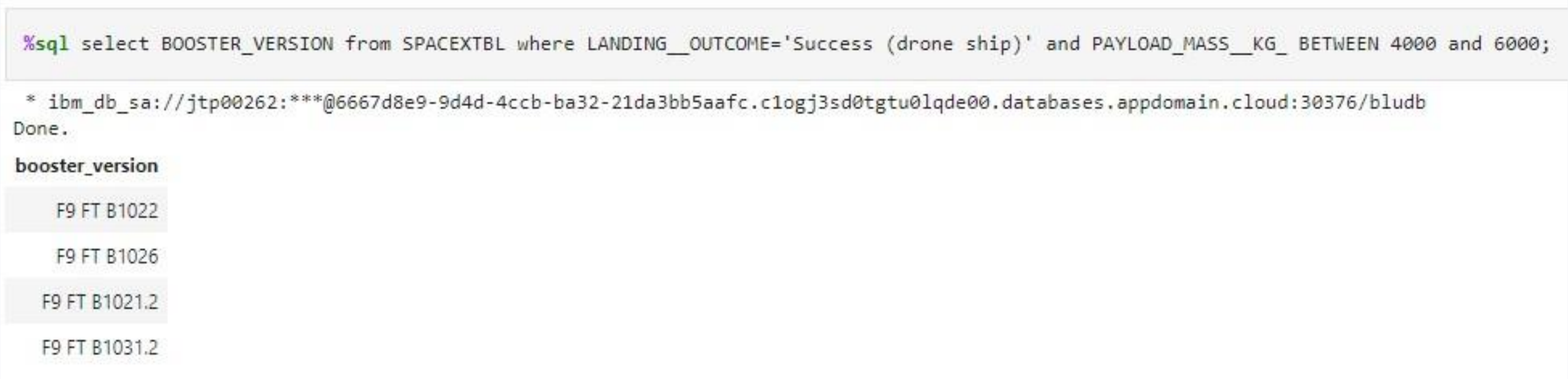
```
%sql SELECT MIN(DATE) AS "First Successful Landing outcome in ground pad" FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

First Successful Landing outcome in ground pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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



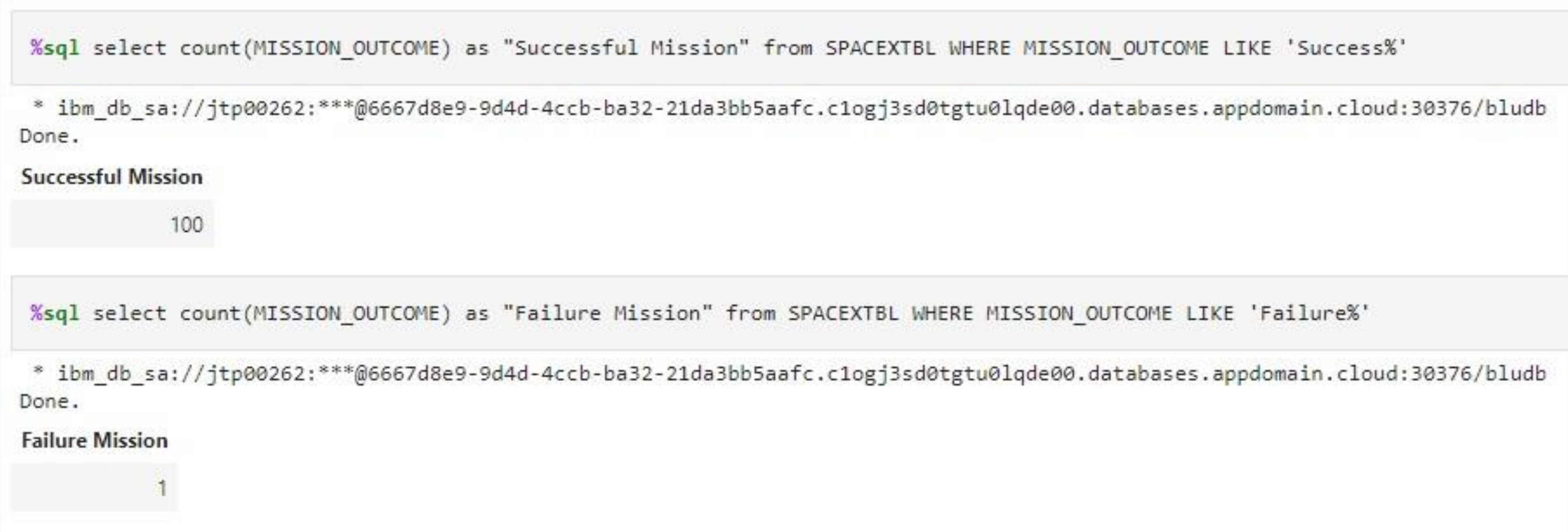
```
%sql select BOOSTER_VERSION from SPACEXTBL where LANDING__OUTCOME='Success (drone ship)' and PAYLOAD_MASS__KG_ BETWEEN 4000 and 6000;
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes



```
%sql select count(MISSION_OUTCOME) as "Successful Mission" from SPACEXTBL WHERE MISSION_OUTCOME LIKE 'Success%'
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Successful Mission
100

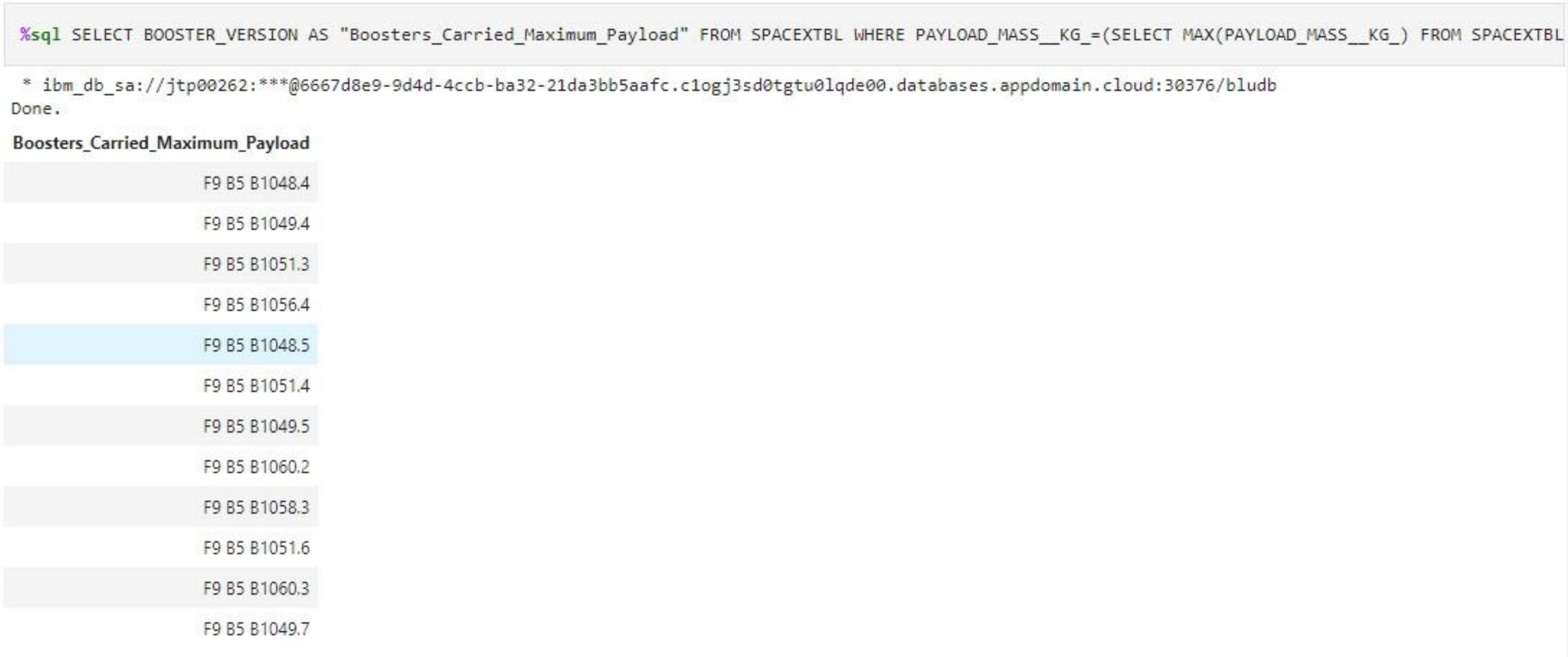
```
%sql select count(MISSION_OUTCOME) as "Failure Mission" from SPACEXTBL WHERE MISSION_OUTCOME LIKE 'Failure%'
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Failure Mission
1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass



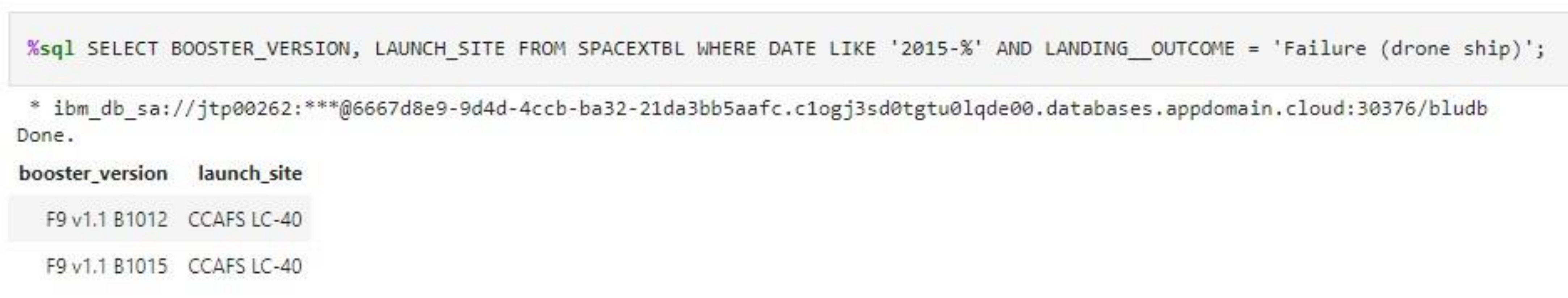
```
%sql SELECT BOOSTER_VERSION AS "Boosters_Carried_Maximum_Payload" FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_=(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Boosters_Carried_Maximum_Payload
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

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



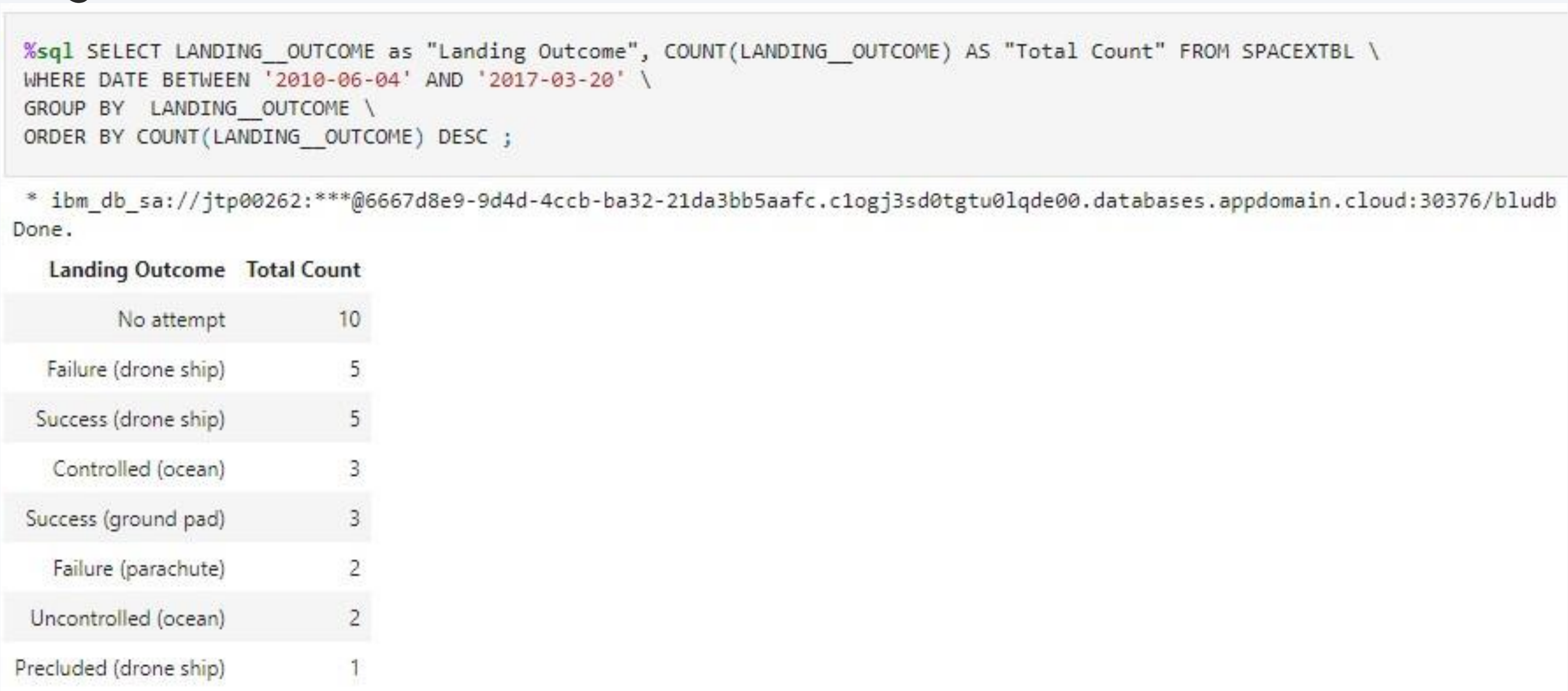
```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE DATE LIKE '2015-%' AND LANDING__OUTCOME = 'Failure (drone ship)';
```

```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

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

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order



```
%sql SELECT LANDING__OUTCOME as "Landing Outcome", COUNT(LANDING__OUTCOME) AS "Total Count" FROM SPACEXTBL \
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
GROUP BY  LANDING__OUTCOME \
ORDER BY COUNT(LANDING__OUTCOME) DESC ;
```

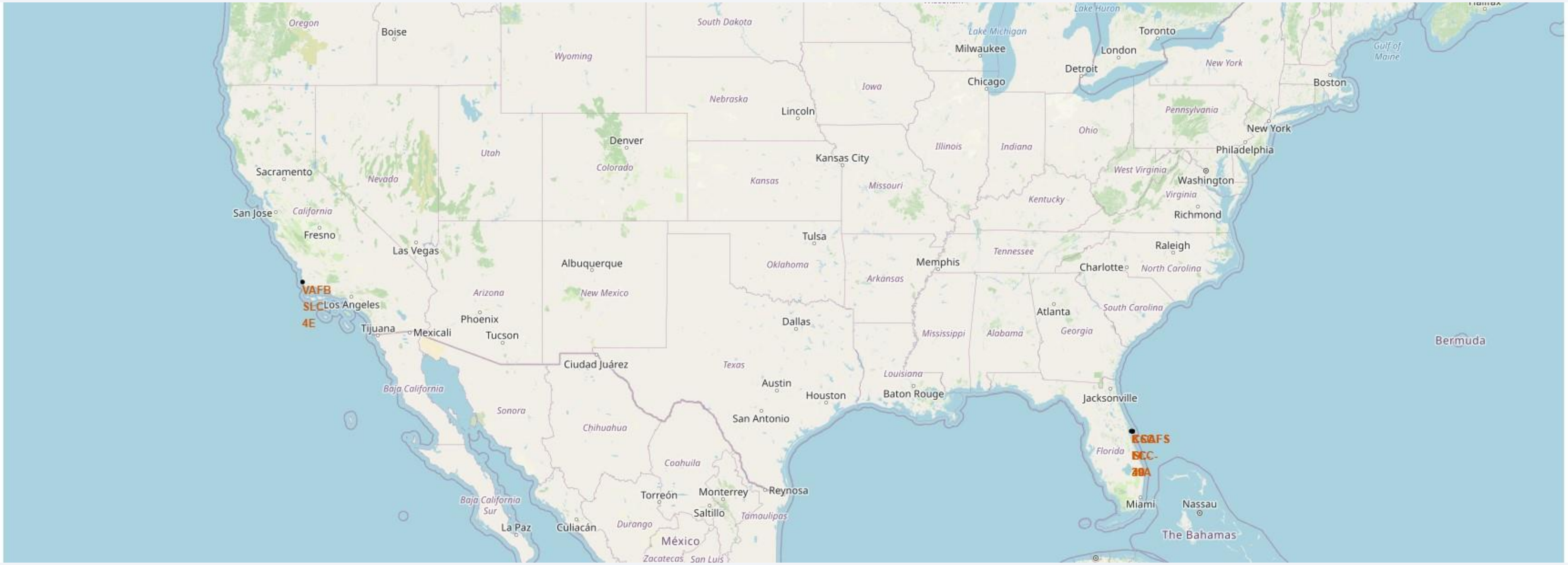
```
 * ibm_db_sa://jtp00262:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Landing Outcome	Total Count
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 3

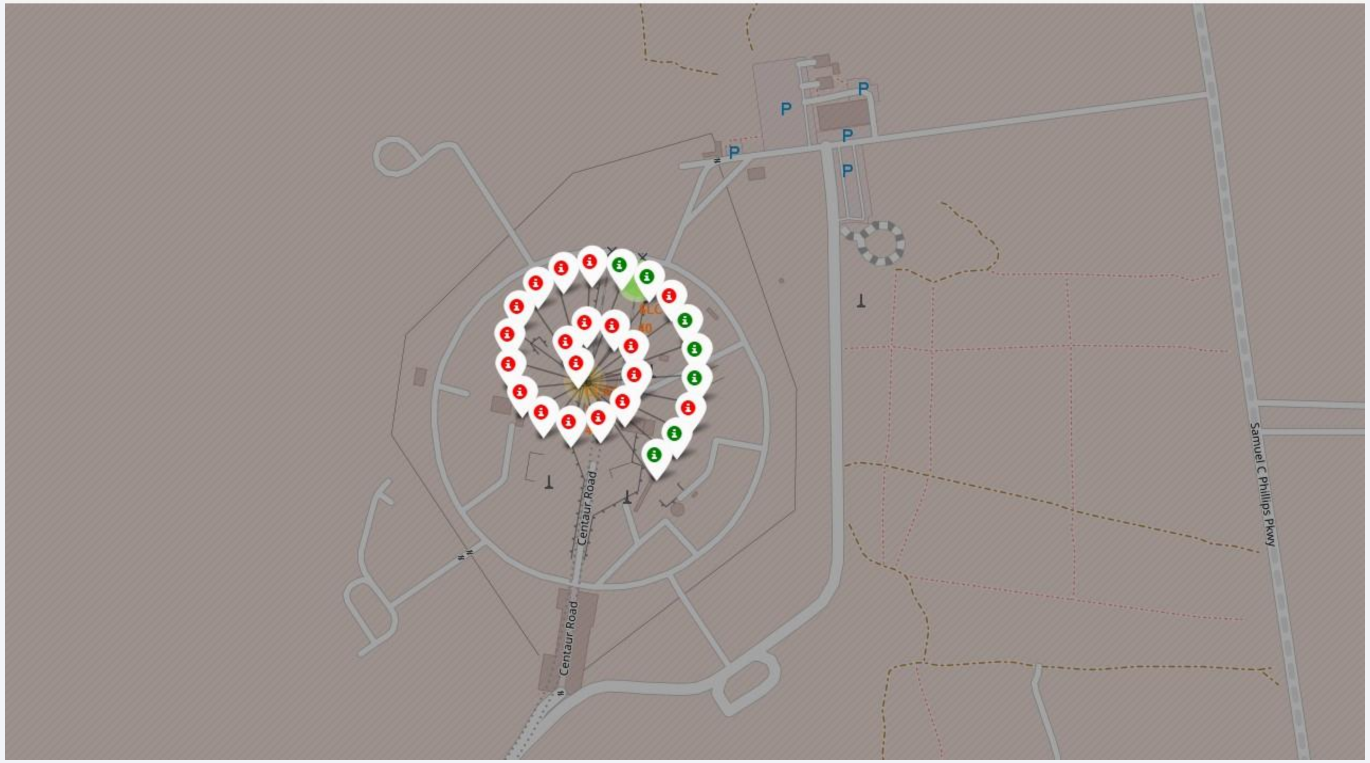
Launch Sites Proximities Analysis

Location of the Launch Sites



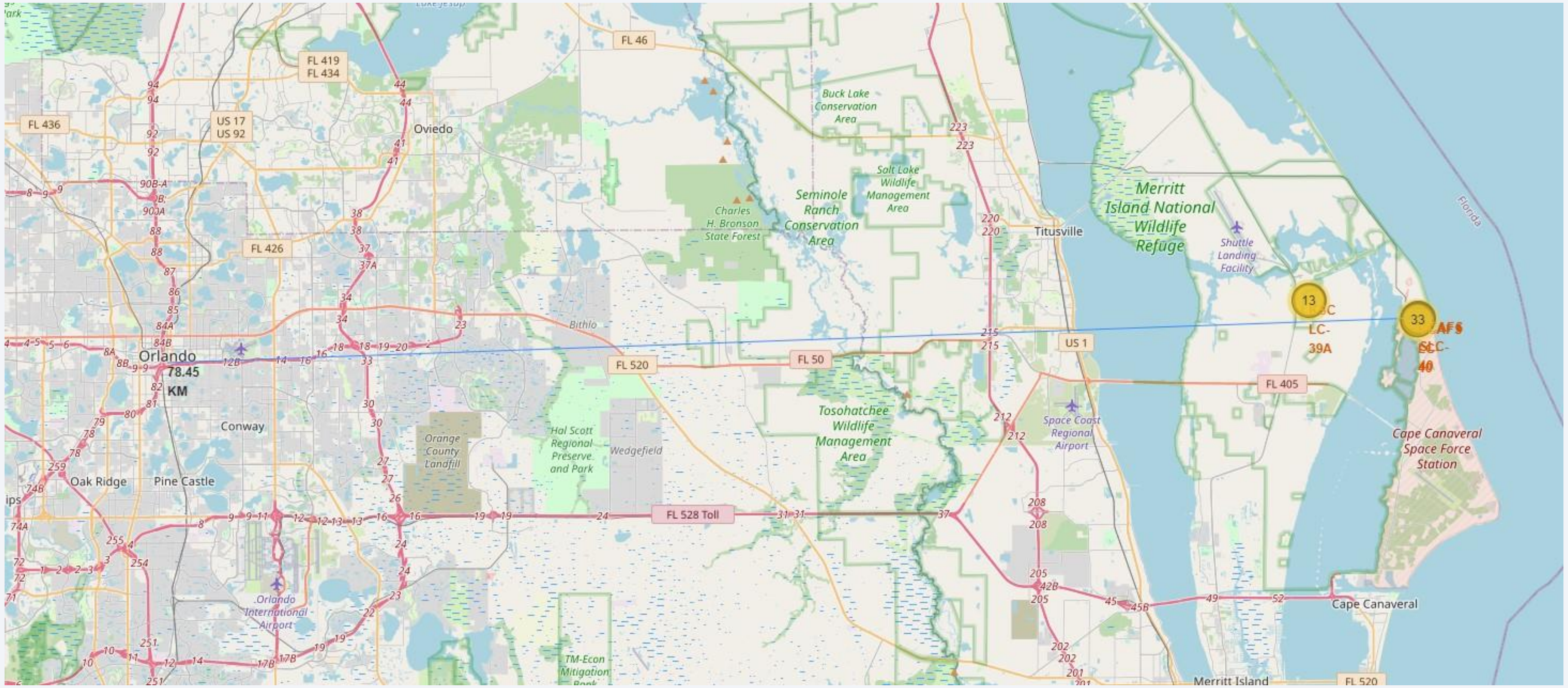
We can see that all the launch sites are close to the coastline in the US

Markers showing launch sites with color labels



Green Markes show success launches and red markes show failure launches

Launch site distance to proximities

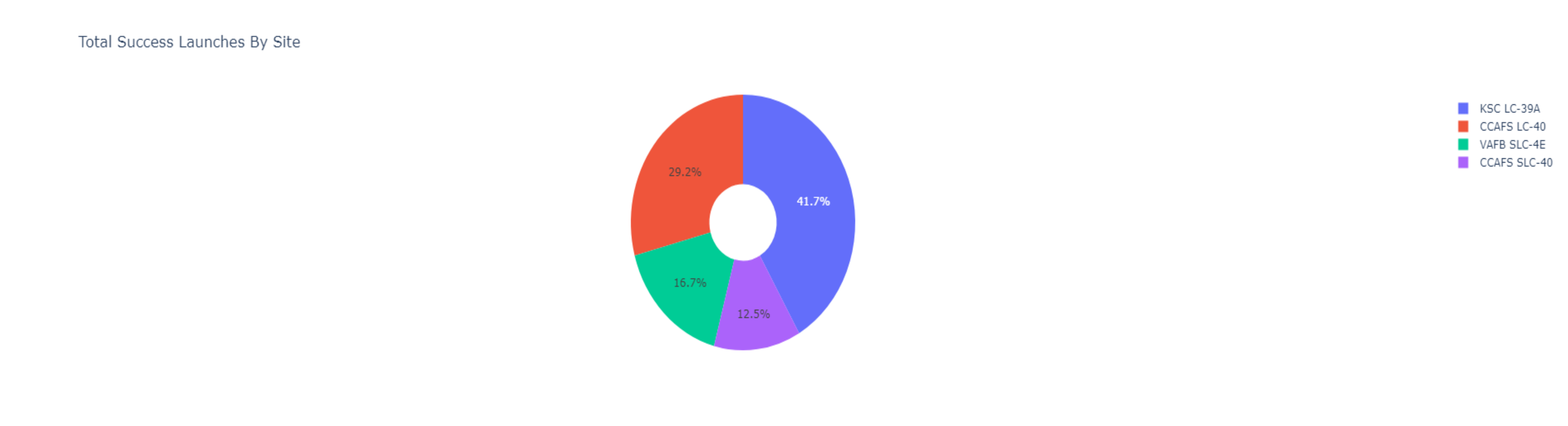


selected launch site to its proximities with distance calculated and displayed

Section 4

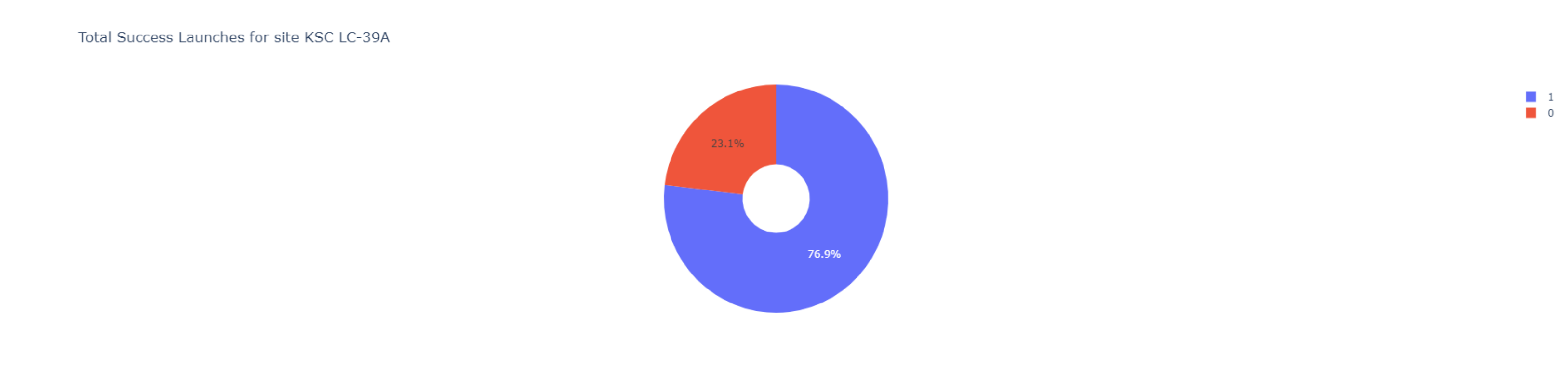
Build a Dashboard with Plotly Dash

Successful launches for all sites



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

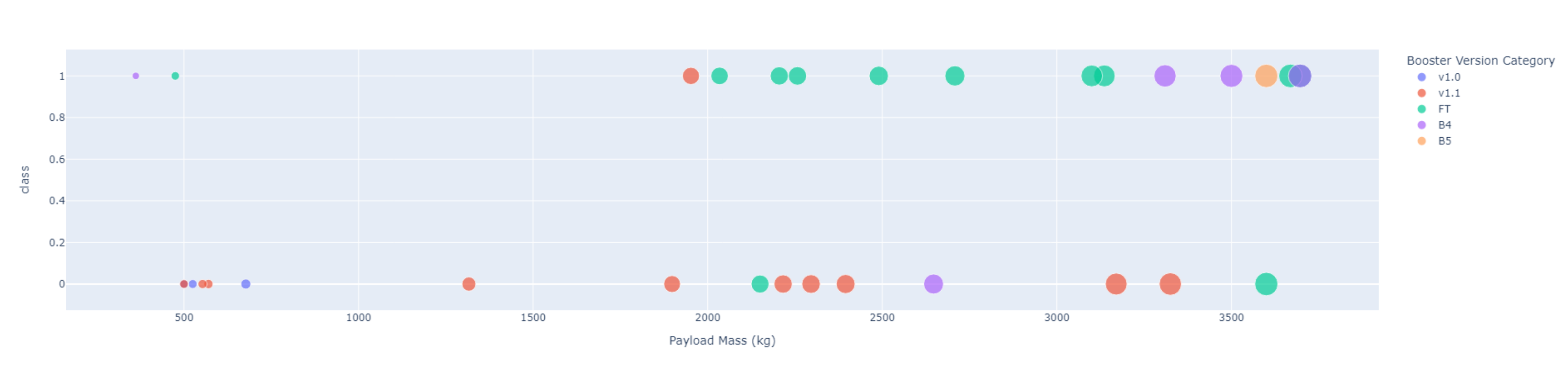
<Dashboard Screenshot 2>

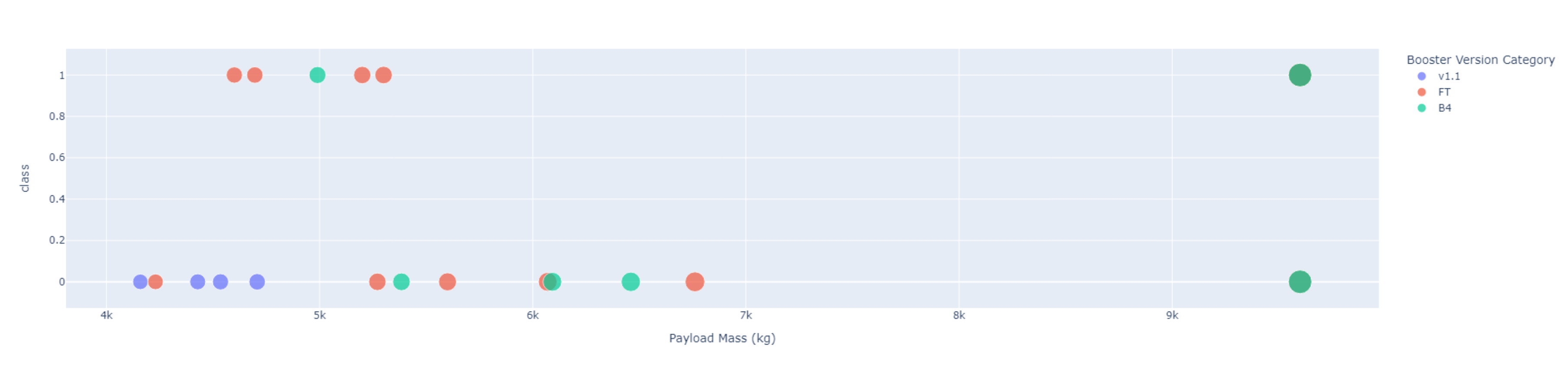


Total Success Launches for site KSC LC-39A

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

Payload VS Launch outcome



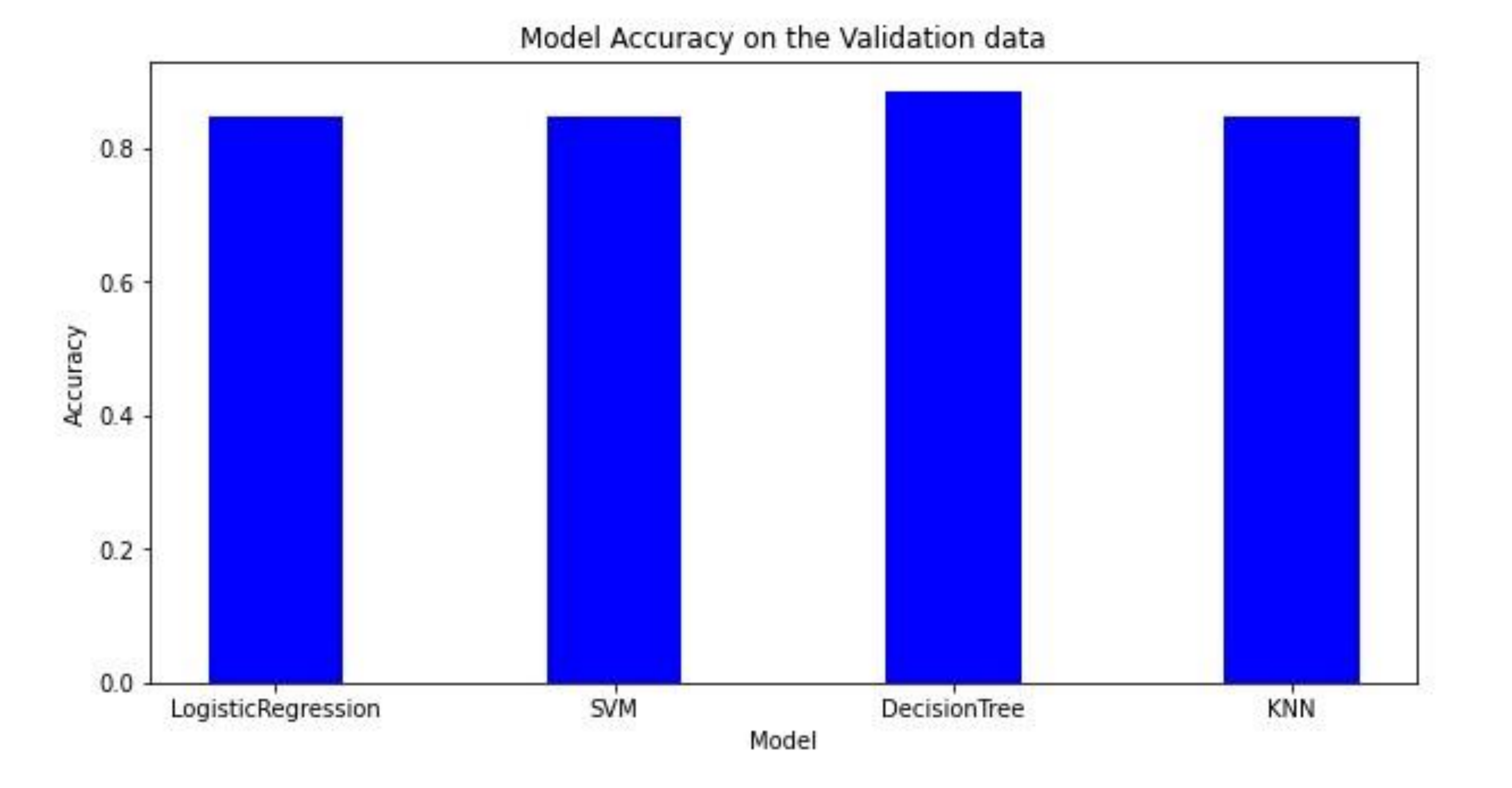


We can see that the success rate for low weighted payload (0~4000kg) is higher than heavy weighted payload (4000~10000kg)

Section 5

Predictive Analysis (Classification)

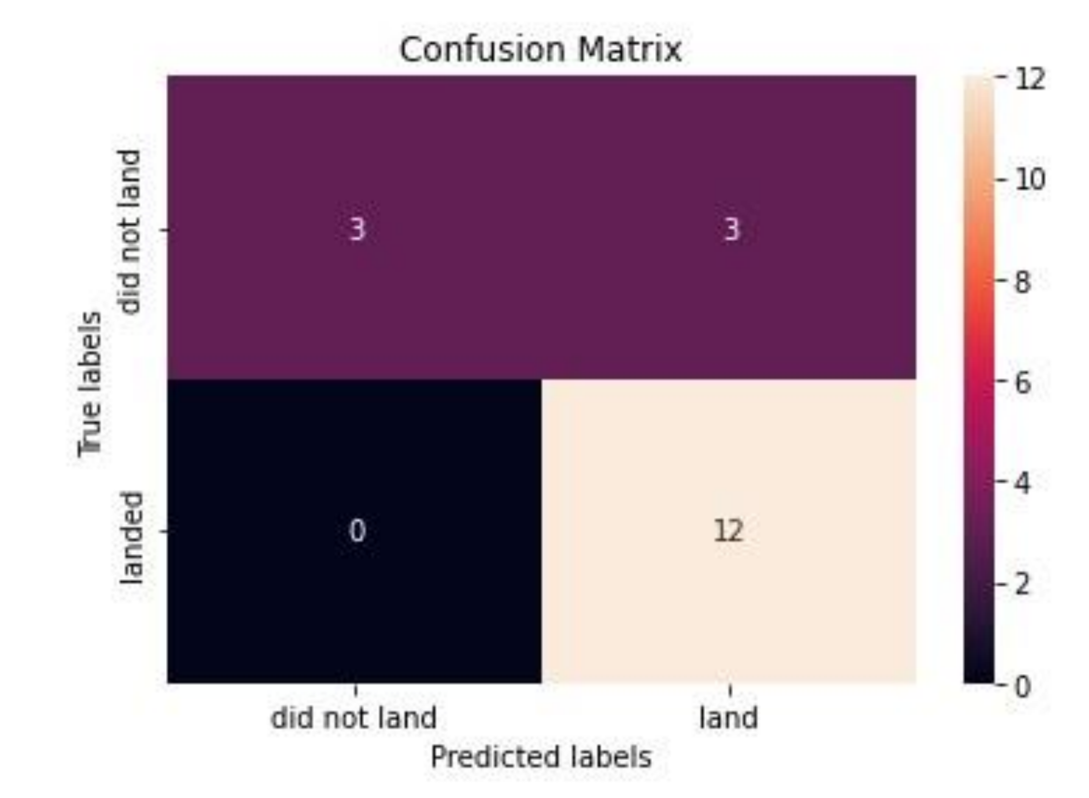
Classification Accuracy



We can observe that Decision Tree has the highest accuracy by the bar plot

Confusion Matrix

- we see that decision tree can distinguish between the different classes. The major problem is false positives (unsuccessful landing predicted as successful landing).



Conclusions

- Decision Tree is the best model for this dataset.
- Low weighted payloads performed better than heavy weighted payloads.
- Sucess rate since 2013 kept increasing till 2020 for SpaceX.
- KSC LC-39A had the most successful launches from all the sites.
- Orbit GEO, HEO, SSO, ES L1 has the highest success rate.

Thank you!