IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Benedikt Reitberger
21.09.2022

Outline

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

Executive Summary

Summary of methodologies

- Data Collection through an API of SpaceX and targeted Web Scraping
- Data Wrangling
- Exploratory Analysis using SQL as well as Data Visualization
- Interactive Visual Analytics with Folium and Plotly Dash
- Machine Learning Prediction

Summary of all results

- Successful collection of data using both sources
- Identification of the best predictive features for the missile launches
- Several Machine Learning approaches in order to outline the most suitable factors for a successful launch

Introduction

Project background and context

In this capstone the task was to take on the role of a data scientist working for a new rocket company, Space Y. They aim to compete with SpaceX. The goal for this project is therefore to predict whether the Falcon 9 rocket of SpaceX will land successfully.

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.

Problems you want to find answers

What are the main factors for a successful landing?

What are the determining characteristics for SpaceX on reusing the first stage?

How does each variable/factor (i.e. launch sites) influence the success rate of the launches?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Web Scraping from Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches)
  - REST API of SpaceX (https://api.spacexdata.com/v4/rockets/)
- Perform data wrangling
  - Drop unnecessary columns
  - Create an “Outcome”-column which indicates whether a landing was successful or not
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Four distinct classification models were used to predict the launch outcomes based on a normalized training and test set

Data Collection

Data collection via two sources:

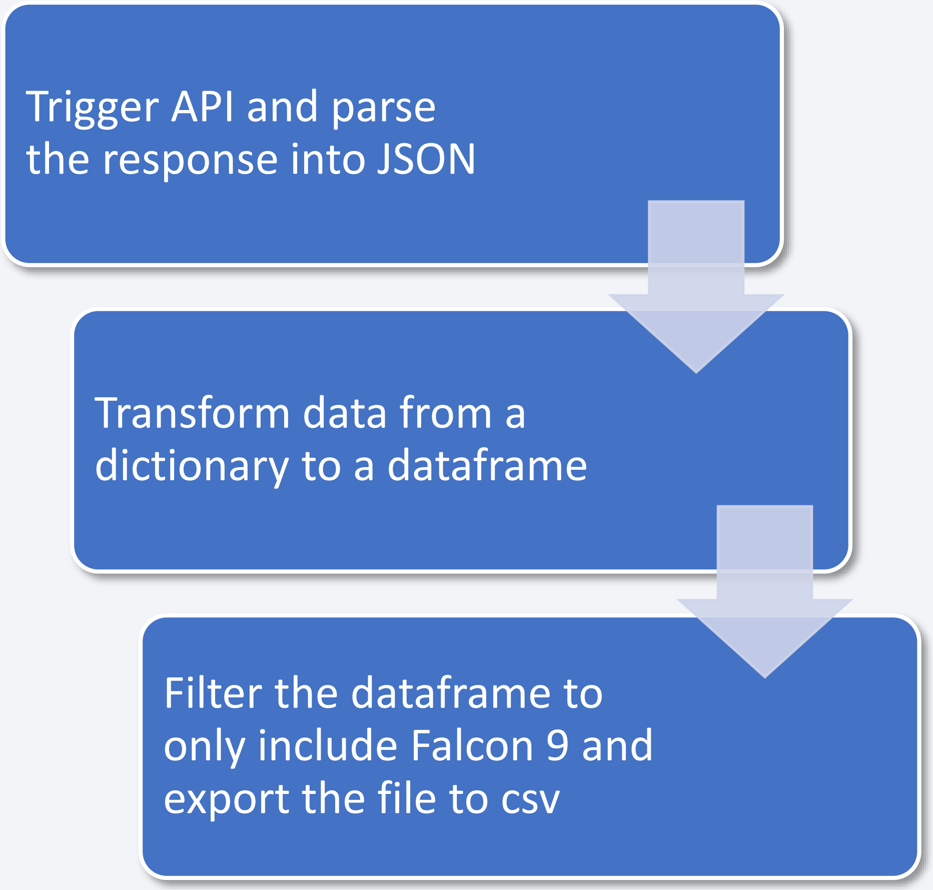
- Web Scraping from Wikipedia
  - https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches
  - Information about payload, launches and landings

- REST API of SpaceX
  - https://api.spacexdata.com/v4/rockets/
  - Information about payload, launches and the rockets

Data Collection – SpaceX API

- Relevant data was collected through the REST API of SpaceX according to the following scheme:

- GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/4e131a8d10f1a1c6ff64ca759782bb0a79990a70/Data%20Collection%20SpaceX%20API.ipynb



Trigger API and parse the response into JSON

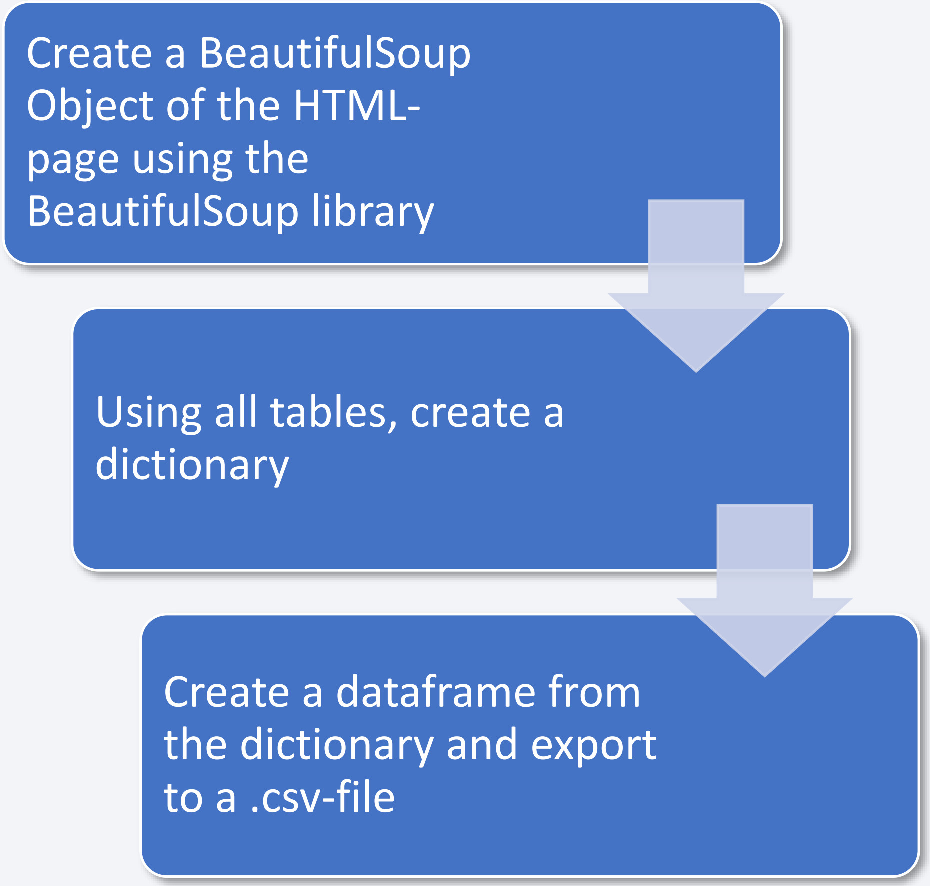
Transform data from a dictionary to a dataframe

Filter the dataframe to only include Falcon 9 and export the file to csv

Data Collection – Web Scraping

- Relevant data was collected through the Wikipedia page of the SpaceX launches

- GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/4e131a8d10f1a1c6ff64ca759782bb0a79990a70/Data%20Collection%20Web%20Scraping.ipynb



Create a BeautifulSoup Object of the HTML-page using the BeautifulSoup library

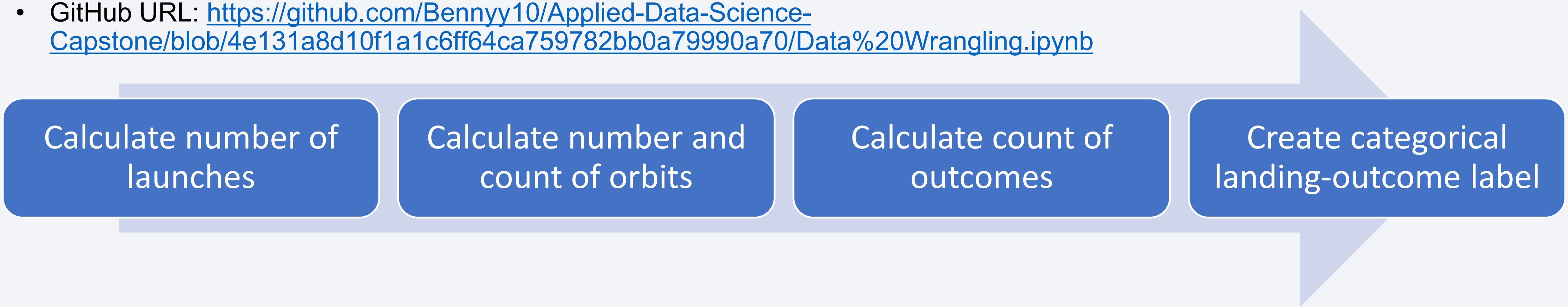
Using all tables, create a dictionary

Create a dataframe from the dictionary and export to a .csv-file

Data Wrangling

- Data Wrangling is the process of cleaning and unifying complex data for easy understanding and a following Exploratory Data Analysis
- Creating an outcome-column for the categorical variable of the landings success (1 = success, 0 = failure)
  - True ASDS, True RTLS, True Ocean = success
  - False ASDS, False RTLS, False Ocean = failure
  - None None, None ASDS = not defined

- GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/4e131a8d10f1a1c6ff64ca759782bb0a79990a70/Data%20Wrangling.ipynb



Calculate number of launches

Calculate number and count of orbits

Calculate count of outcomes

Create categorical landing-outcome label

EDA with Data Visualization

Used Plots for Data Visualization:

- **Line Graphs:** indicate trends and express an overall behavior of the data
- **Bar Graphs:** indicate the variation of the data and show the relationship of numerical and categorical data
- **Scatter Graphs:** indicate the correlation between different variables/measures

→The following relationships of the existing features were analyzed:

→Payload Mass ↔ Flight Number, Launch Site ↔ Flight Number, Launch Site ↔ Payload Mass, Orbit ↔ Flight Number, Payload ↔ Orbit

GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/51af266ad22b18349db0b14f3aede73d4f20cd56/EDA%20Visualization.ipynb

EDA with SQL

Performed SQL queries:

- 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 achieved
- 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. Use a subquery
- 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.
- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order

GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/834fa95ddf6711fe54fd7c7cd7a18332195d36a9/EDA%20with%20SQL%20.ipynb

Build an Interactive Map with Folium

In order to build an interactive Map, the Folium Maps library was used, centered around the NASA Johnson Space Center at Houston, Texas.

- Using the latitude and longtitude coordinates, **circles** highlight each launch site with labels indicating the specific launch site names
- **Markers** refer to the landing that either resulted in success or in failure.
- With the help of the Haversine's formula the distance between different launch sites to key infrastructure is shown through plot **lines**
- GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/5d953321ac8893733bb1ec08823a85b911b7294b/Launch%20Sites%20Location%20Analysis%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

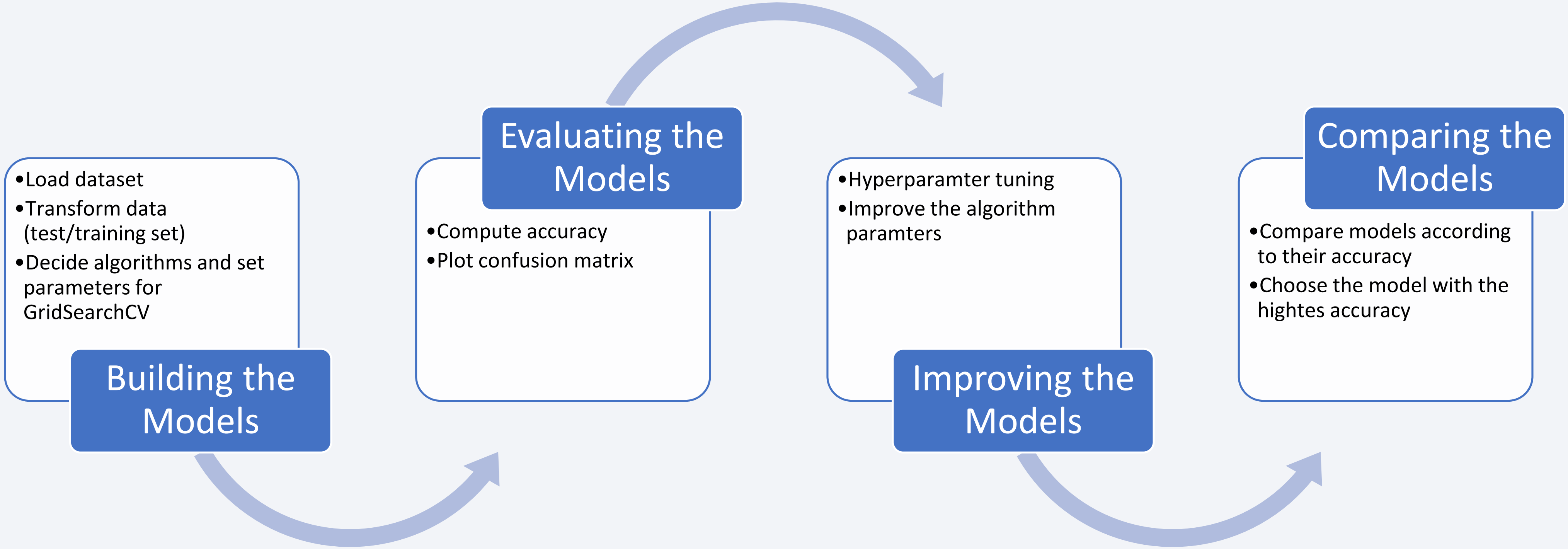
A Dashboard was created in order to analyze the relation between launch sites and payloads. The goals was to identify the best place to launch regarding the payloads.

The following components were used to display the content:

- Dropdowns (comfortable maneuvering of the different launch sites for the user)
- Pie Charts (concise overview of the total percentage of success and failure)
- Rangesliders (comprehensible adjustments of payloads of interest)
- Scatter chart (clear distinction of the relationship of the variables involved)

GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/ba998512e985fc621ad30588fd3014b0b55980dc/Dash%20python.txt

Predictive Analysis (Classification)



Building the Models

- Load dataset
- Transform data (test/training set)
- Decide algorithms and set parameters for GridSearchCV

Evaluating the Models

- Compute accuracy
- Plot confusion matrix

Improving the Models

- Hyperparamter tuning
- Improve the algorithm paramters

Comparing the Models

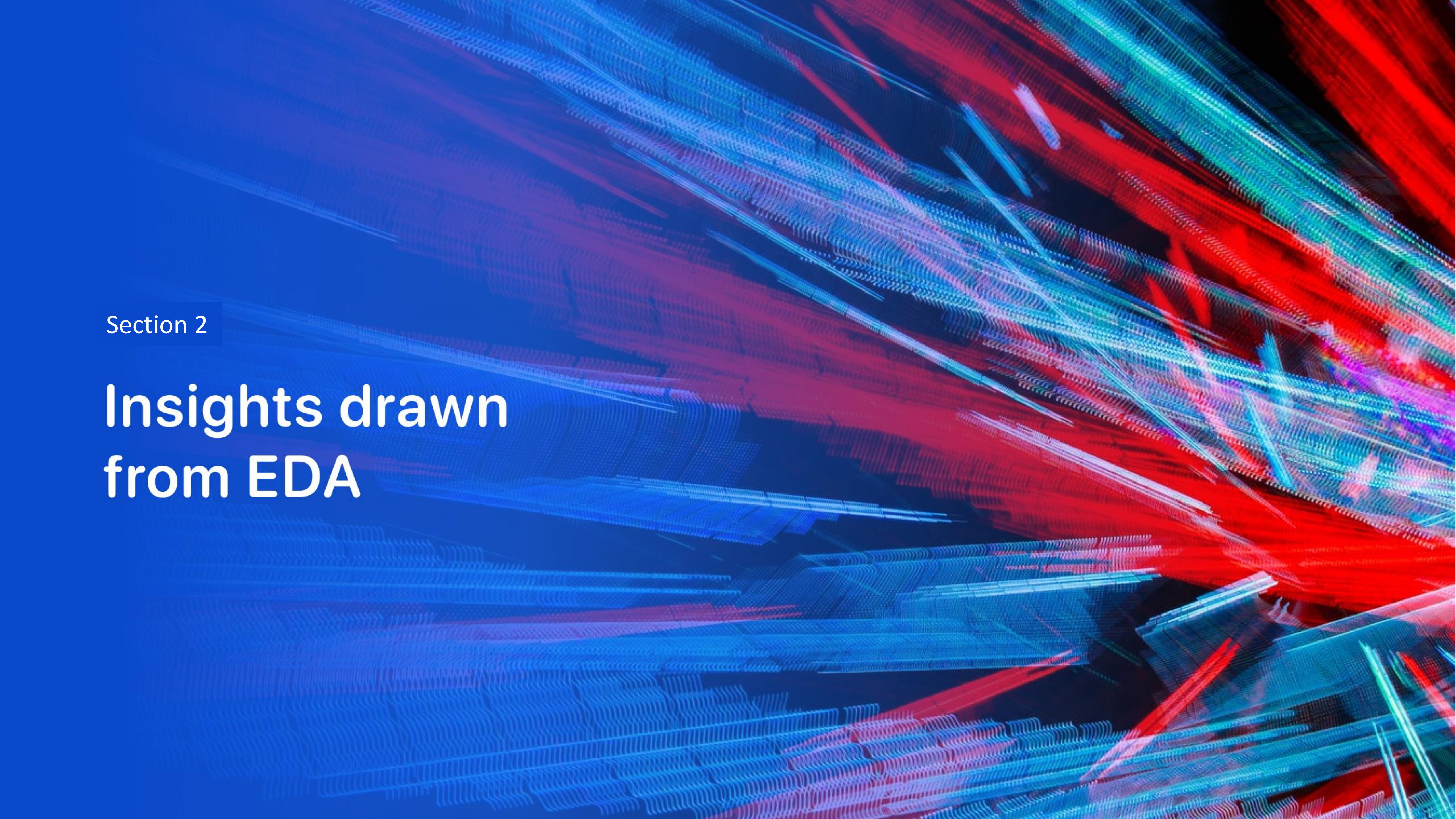
- Compare models according to their accuracy
- Choose the model with the hightes accuracy

- GitHub URL: https://github.com/Bennyy10/Applied-Data-Science-Capstone/blob/5f3a650759fb3dbf2d4c27be4d8a99b1cf981a8c/Machine%20Learning%20Predictions%20.ipynb

Results

The following chapters will provide insights about the following points:

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



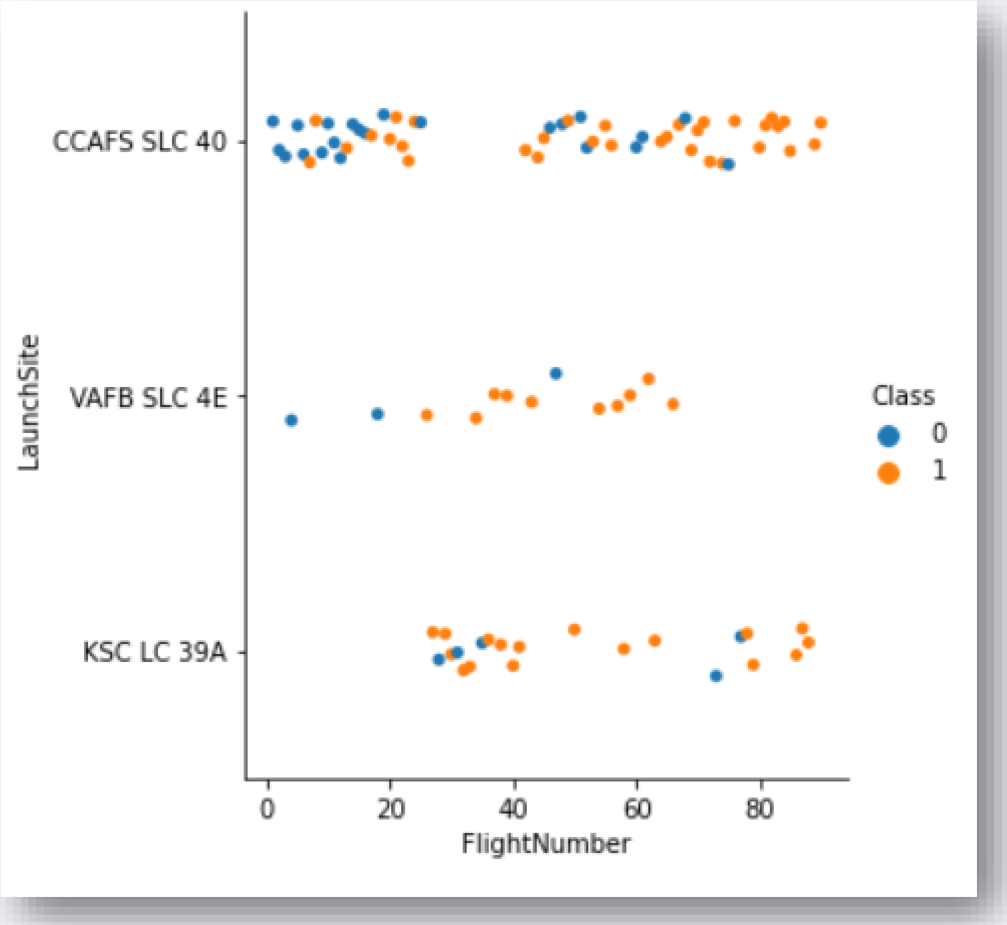
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

- The scatter plot indicates that there is a positive correlation between the number of flights and the success rate

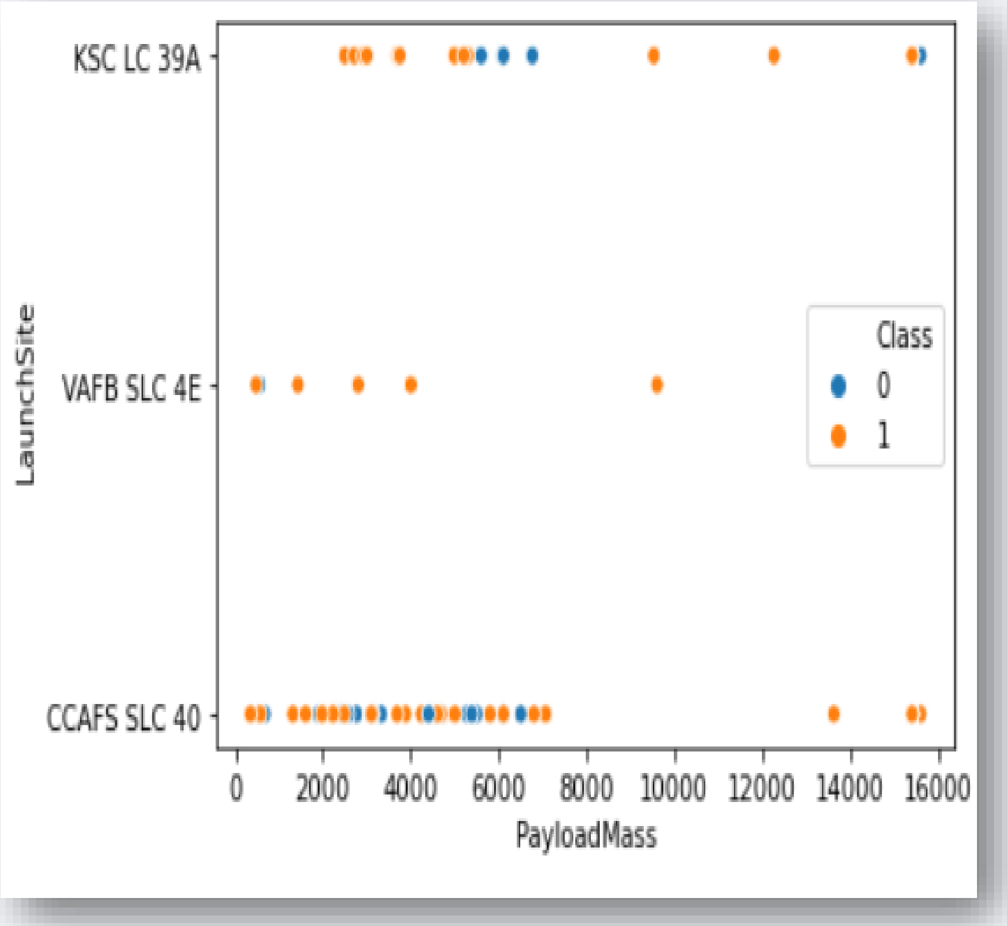
- The Launch Site CCAFS SLC 40 shows the most volatile results



Payload vs. Launch Site

- There is an indication that an increase in Payload Mass leads to a higher success rate of the launches

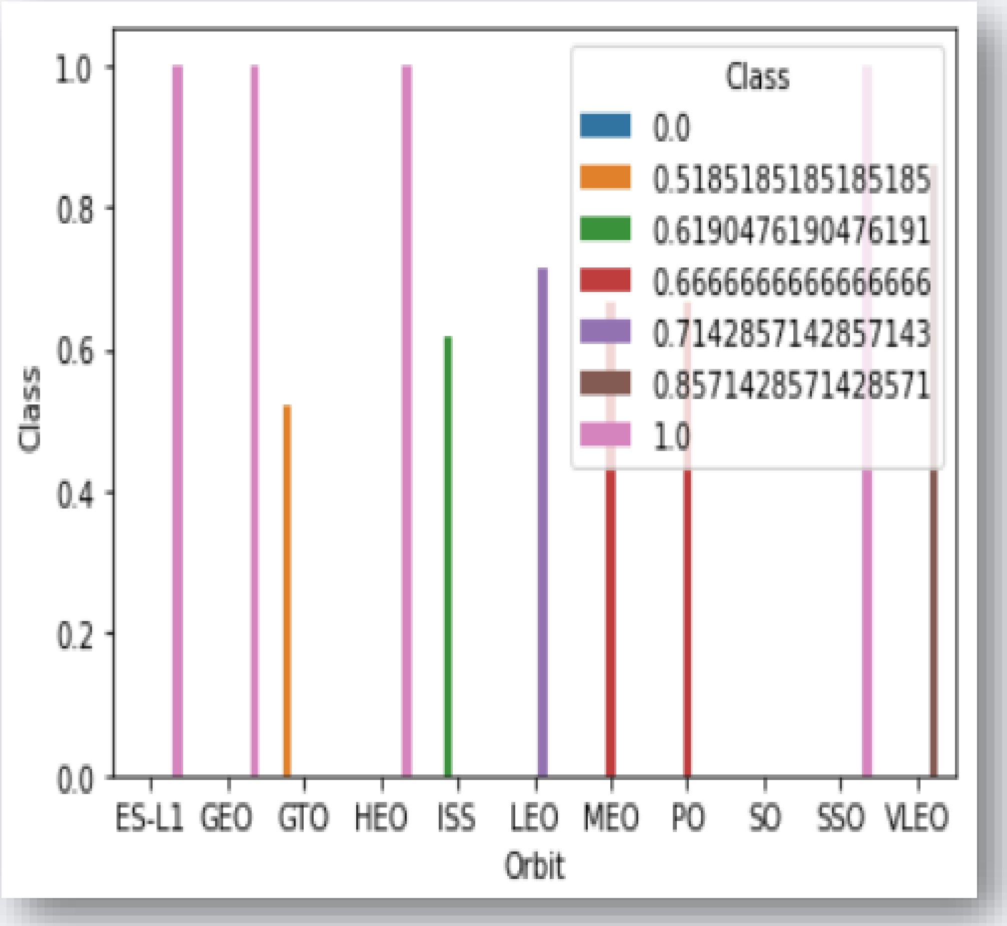
- But there is no sign that the Launch Site is a dependent factor in this comparison



Success Rate vs. Orbit Type

- The bar chart clearly shows the success rate for the different Orbit Types

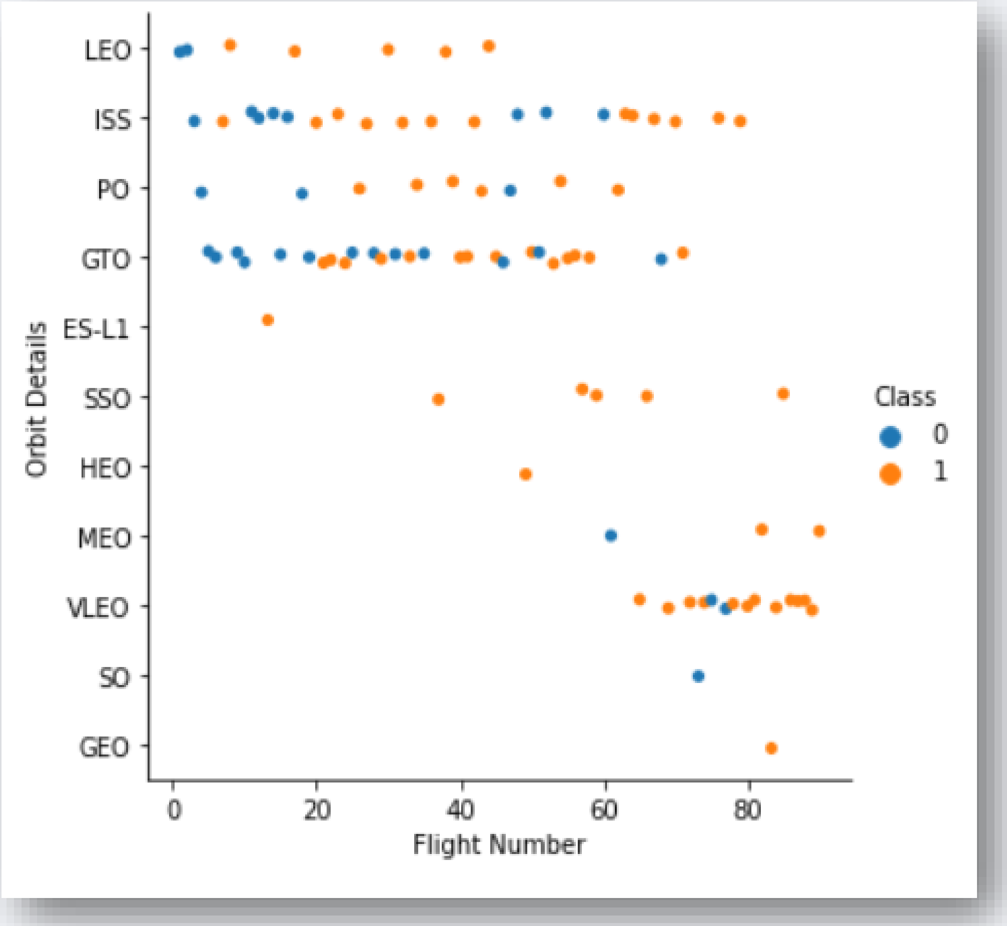
- It can be seen that ES-L1, GEO, HEO, and SSO have the highest success rate



Flight Number vs. Orbit Type

- With the exception of the GTO Orbit Type a clear positive correlation is shown between the number of flights and the Orbit Type

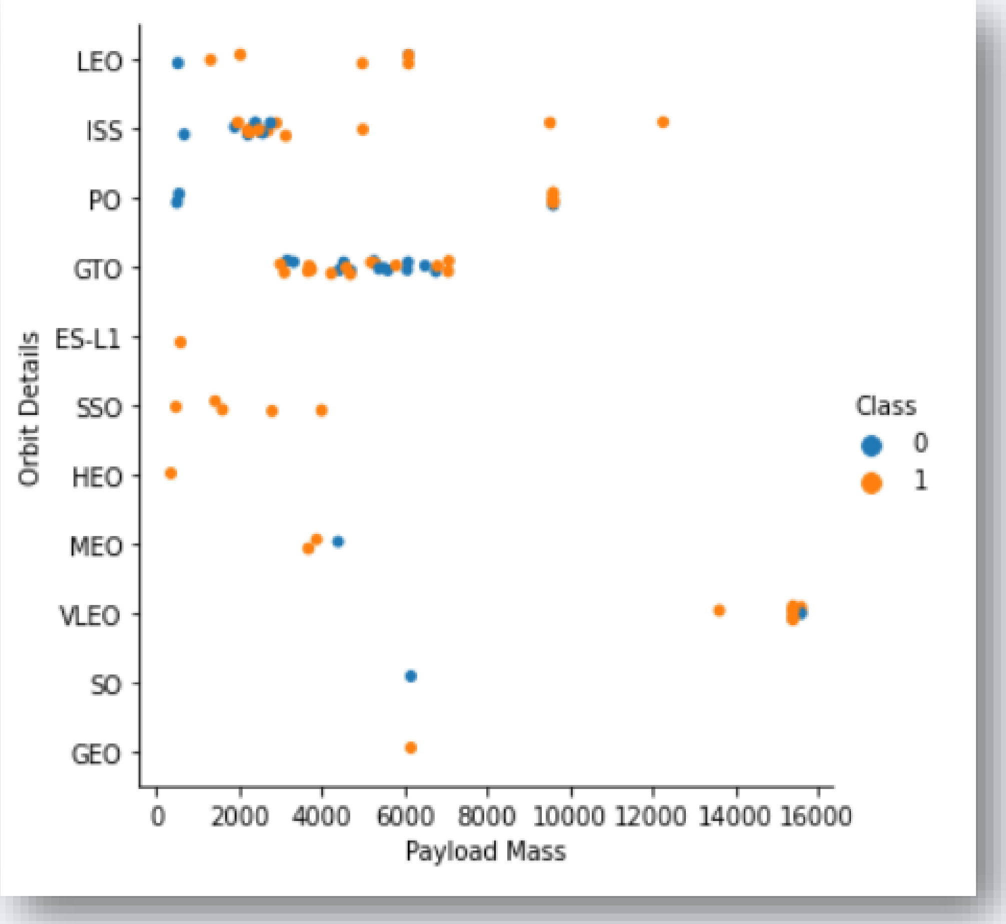
- Due to the low number available data points, a clear statements for the Orbit Types ES-L1, HEO and GEO cannot be made



Payload vs. Orbit Type

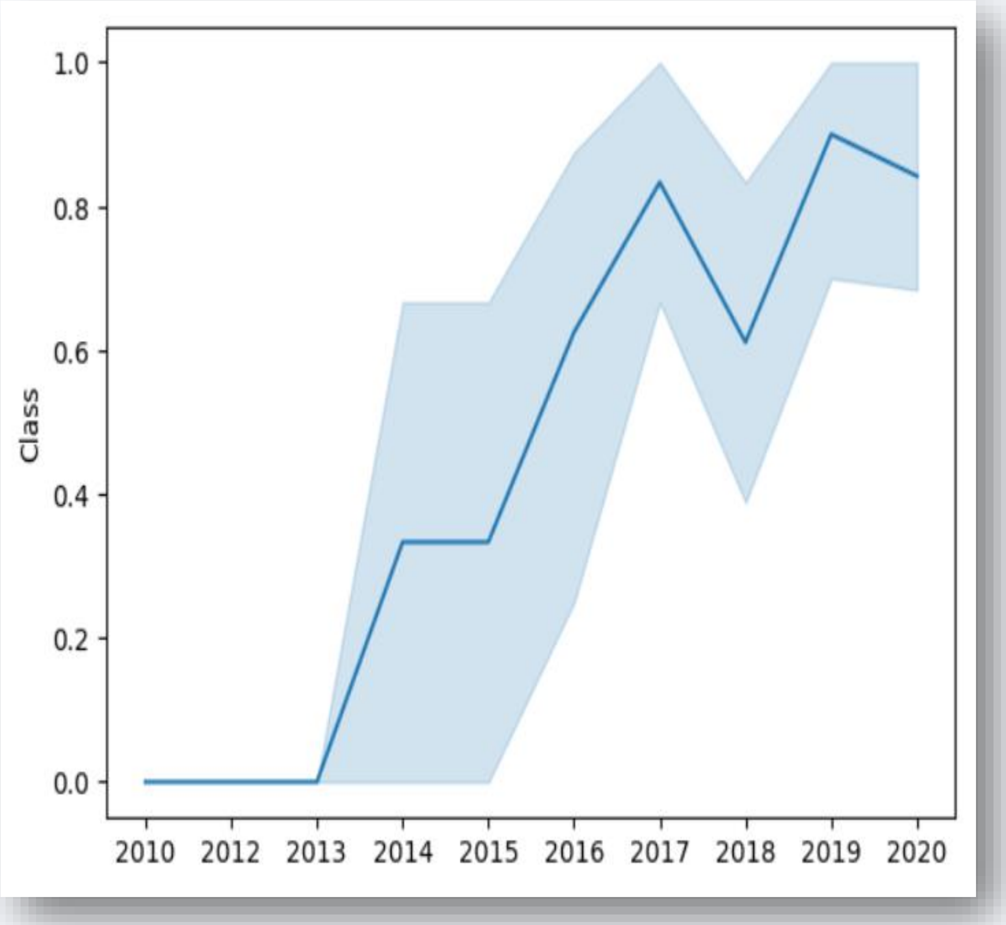
- The scatter plot shows a mostly positive correlation between Orbit Type and Payload Mass (exceptions are MEO, VLEO and SO)

- For a representative conclusion more data is needed



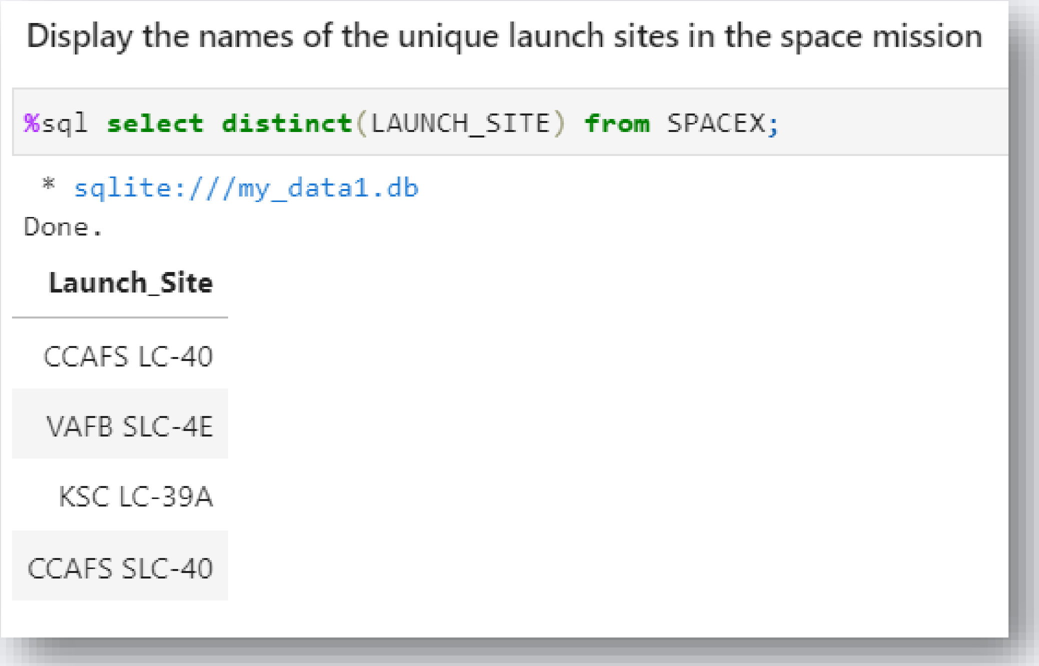
Launch Success Yearly Trend

- Since 2013 there is a steady positive trend of the success rate of the SpaceX rocket until 2020



All Launch Site Names

By using the “distinct” keyword the unique launch sites could be extracted



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

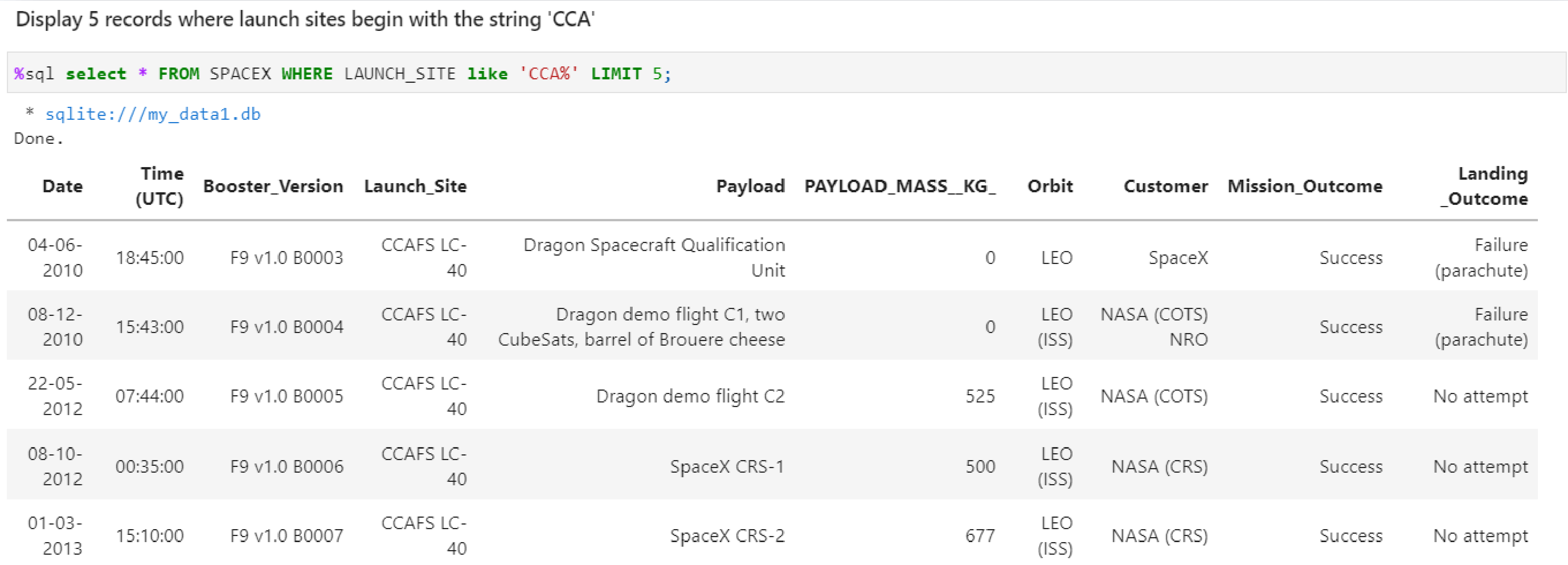
```
%sql select distinct(LAUNCH_SITE) from SPACEX;
```

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

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

Launch Site Names Begin with 'CCA'

- By using a “where”-statement a limitation to Launch Sites with the name “CCA” becomes possible



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

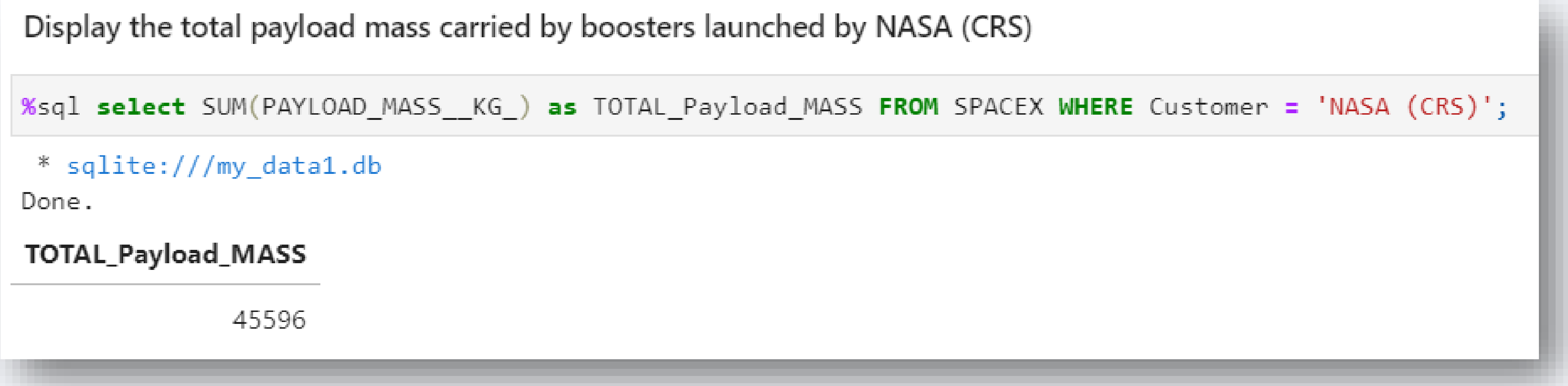
```
%sql select * FROM SPACEX WHERE LAUNCH_SITE like 'CCA%' LIMIT 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Through the aggregate function SUM() the total Payload Mass can be calculated where the customer is “NASE (CRS)”



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

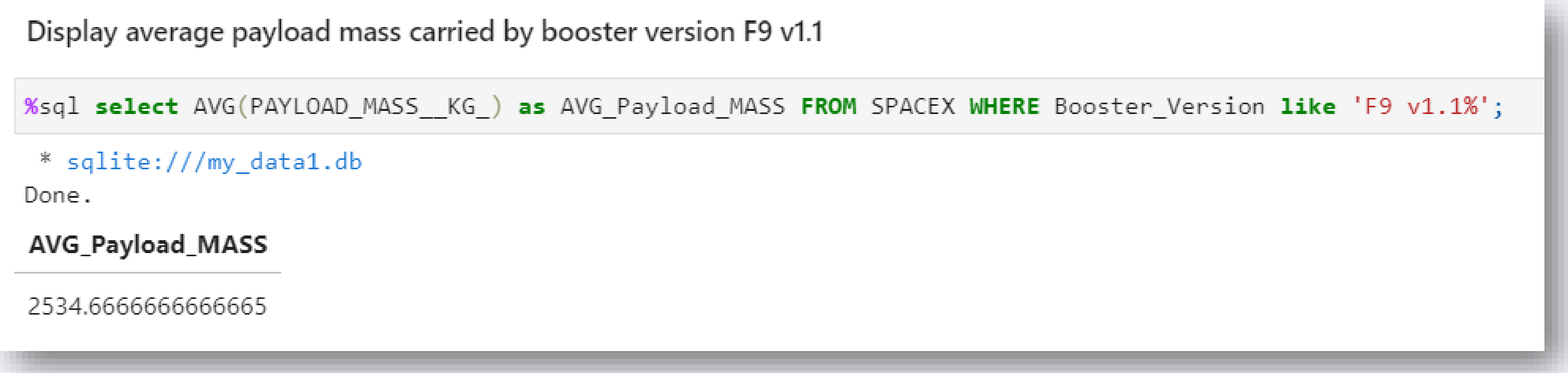
```
%sql select SUM(PAYLOAD_MASS__KG_) as TOTAL_Payload_MASS FROM SPACEX WHERE Customer = 'NASA (CRS)';
```

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

TOTAL_Payload_MASS
45596

Average Payload Mass by F9 v1.1

- With the help of the aggregate function AVG() the average Payload Mass can be calculated. Combined with the distinction of the Booster Version through the “like” keyword, the following result is shown:



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

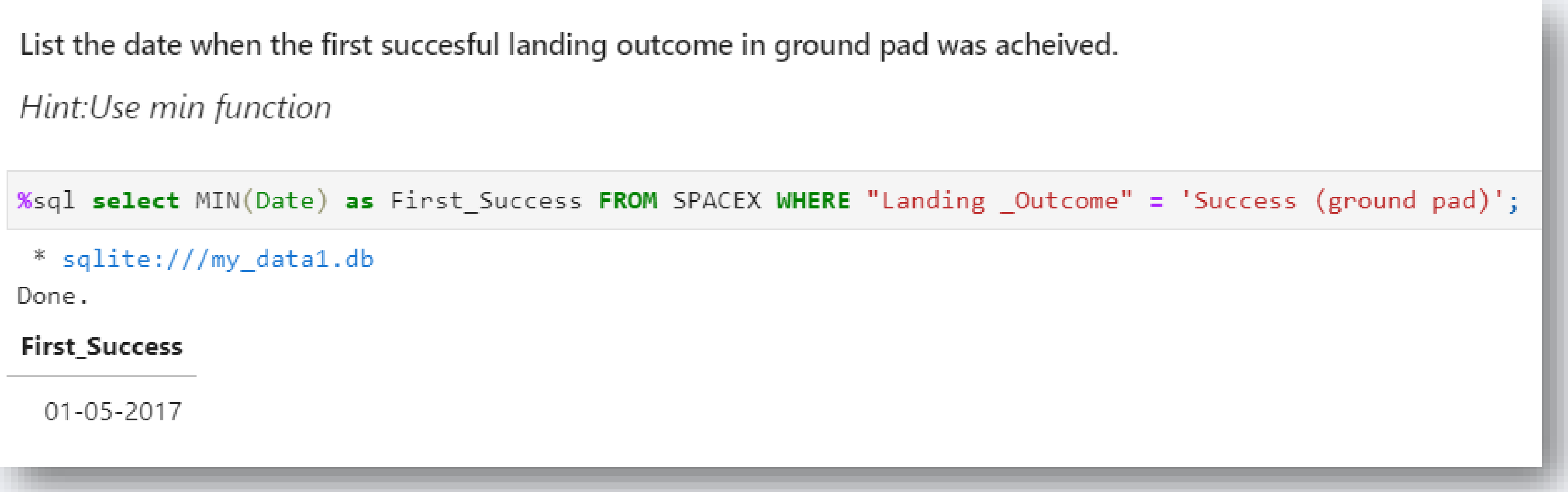
```
%sql select AVG(PAYLOAD_MASS__KG_) as AVG_Payload_MASS FROM SPACEX WHERE Booster_Version like 'F9 v1.1%';
```

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

AVG_Payload_MASS
2534.6666666666665

First Successful Ground Landing Date

- Through the use of the MIN() function, the oldest successful landing is selected. The option for success is regulated in the where-clause.



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

Hint:Use min function

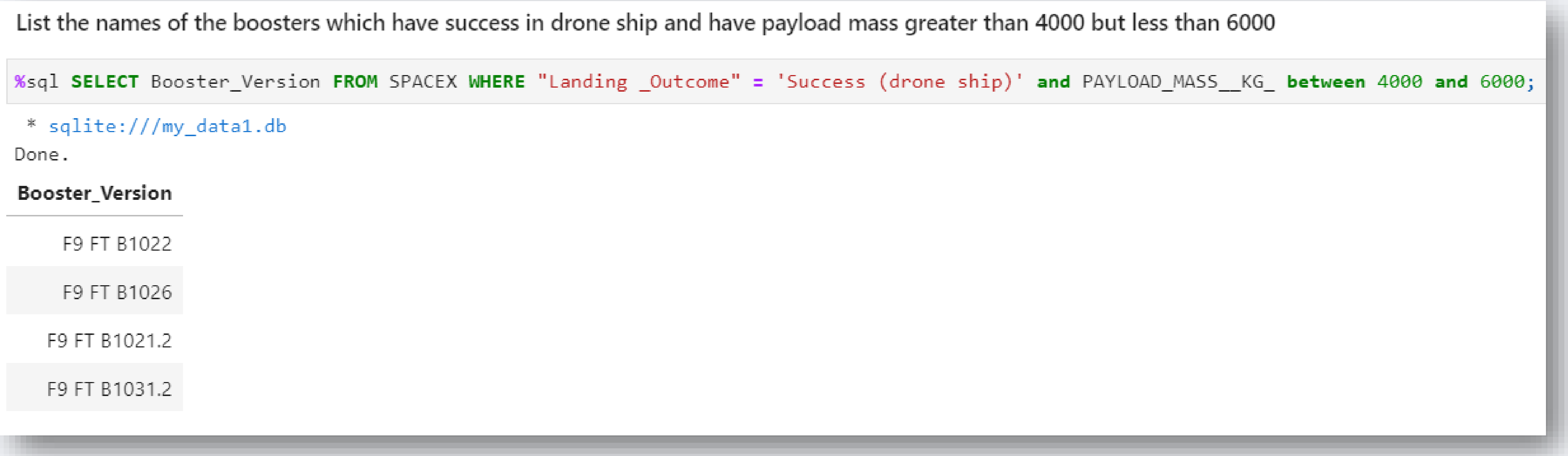
```
%sql select MIN(Date) as First_Success FROM SPACEX WHERE "Landing _Outcome" = 'Success (ground pad)';
```

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

First_Success
01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- Multiple conditions are ensured within the where-statement through the connection of the “and”-keyword.



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

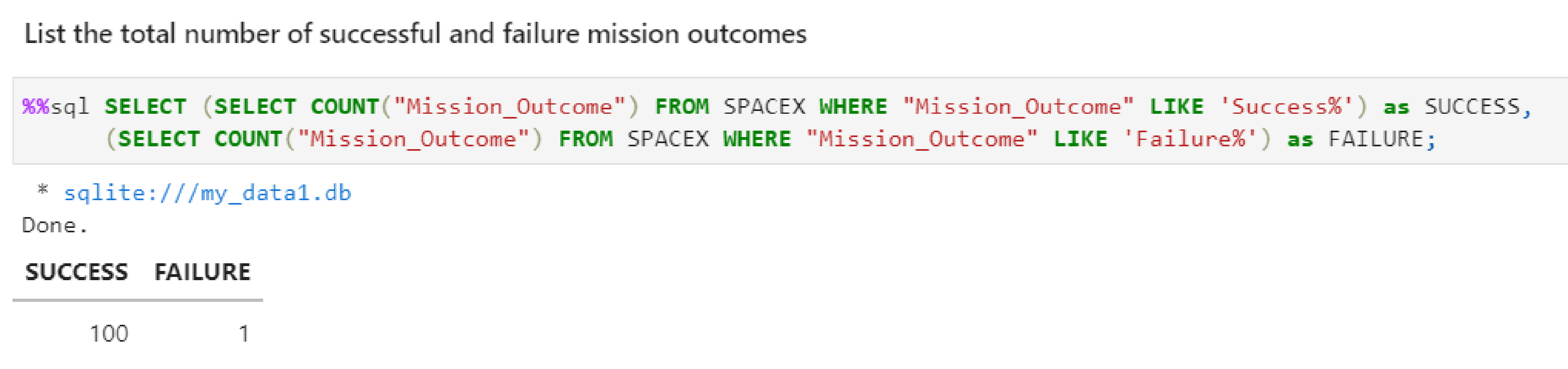
```
%sql SELECT Booster_Version FROM SPACEX WHERE "Landing _Outcome" = 'Success (drone ship)' and PAYLOAD_MASS__KG_ between 4000 and 6000;
```

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

- The first SELECT-statement counts the number of successful missions. The second query counts the number of the failed missions. The “LIKE”-keyword is responsible for the distinction of the outcomes.



List the total number of successful and failure mission outcomes

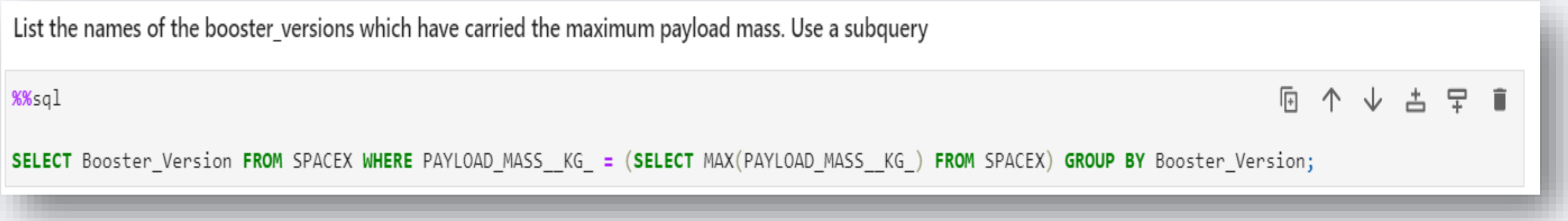
```
%%sql SELECT (SELECT COUNT("Mission_Outcome") FROM SPACEX WHERE "Mission_Outcome" LIKE 'Success%') as SUCCESS,
      (SELECT COUNT("Mission_Outcome") FROM SPACEX WHERE "Mission_Outcome" LIKE 'Failure%') as FAILURE;
```

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

SUCCESS	FAILURE
100	1

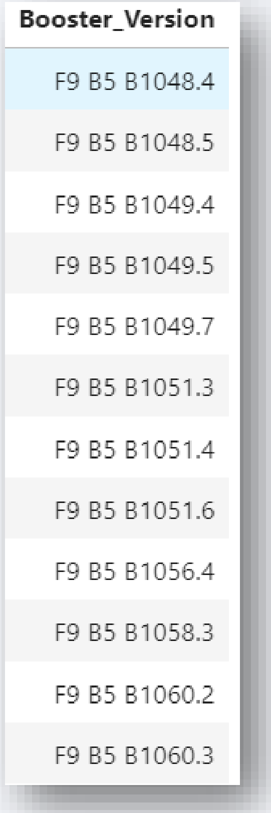
Boosters Carried Maximum Payload

- Using a nested subquery within the where-clause it is possible to only return the maximum of the payload mass.



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

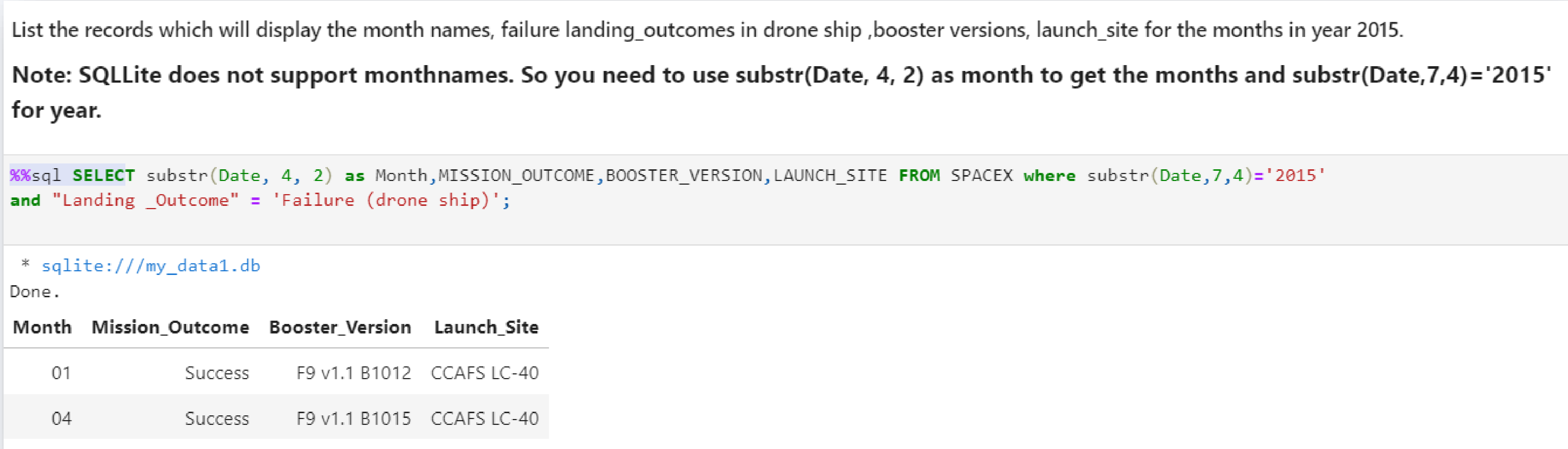
```
%%sql

SELECT Booster_Version FROM SPACEX WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEX) GROUP BY Booster_Version;
```



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

2015 Launch Records

- There are only two failed landing outcomes in 2015 for the drone ship.
- The condition for the date as well as failures is defined within the where-clause



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, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

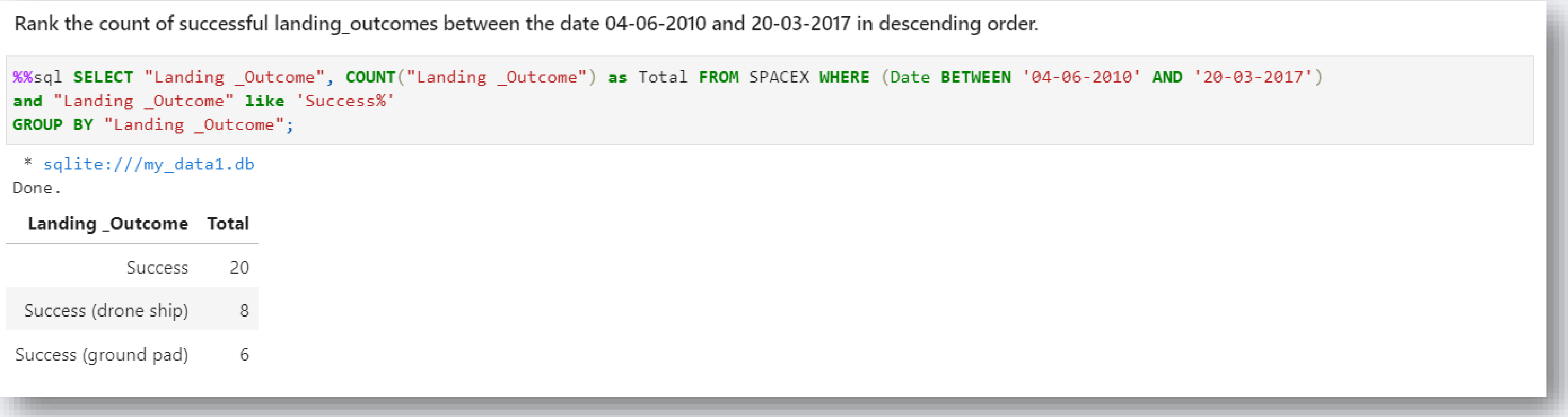
```
%%sql SELECT substr(Date, 4, 2) as Month,MISSION_OUTCOME,BOOSTER_VERSION,LAUNCH_SITE FROM SPACEX where substr(Date,7,4)='2015'
and "Landing _Outcome" = 'Failure (drone ship)';
```

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

Month	Mission_Outcome	Booster_Version	Launch_Site
01	Success	F9 v1.1 B1012	CCAFS LC-40
04	Success	F9 v1.1 B1015	CCAFS LC-40

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

- The query returns the specific landing outcomes as well as the number of their occurrence between 04.06.2010 and 20.03.2017.
- The order is by default in descending order. Otherwise an ORDER BY COUNT() DESC clause would have been necessary



Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

```
%%sql SELECT "Landing _Outcome", COUNT("Landing _Outcome") as Total FROM SPACEX WHERE (Date BETWEEN '04-06-2010' AND '20-03-2017')
and "Landing _Outcome" like 'Success%'
GROUP BY "Landing _Outcome";
```

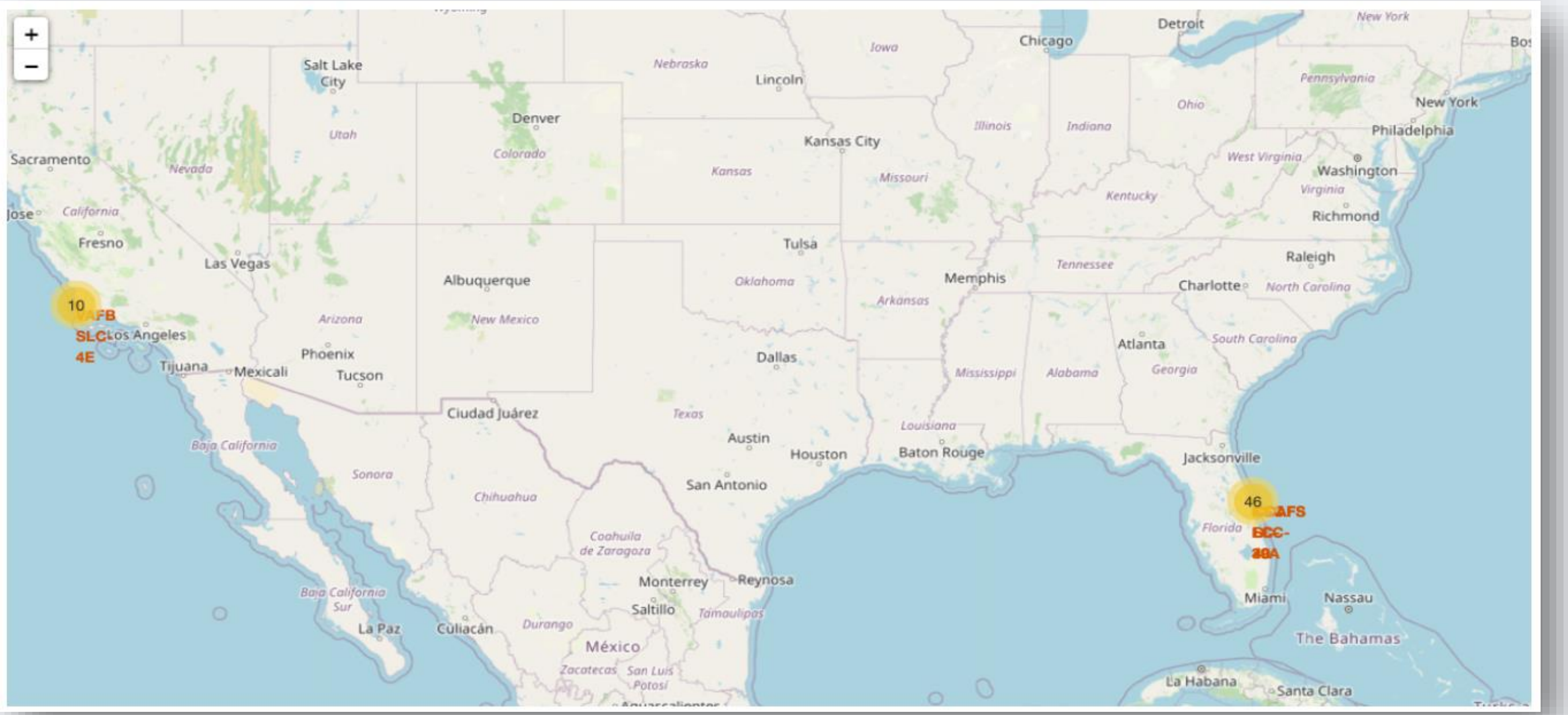
```
 * sqlite:///my_data1.db
Done.
```

Landing _Outcome	Total
Success	20
Success (drone ship)	8
Success (ground pad)	6

Section 3

Launch Sites Proximities Analysis

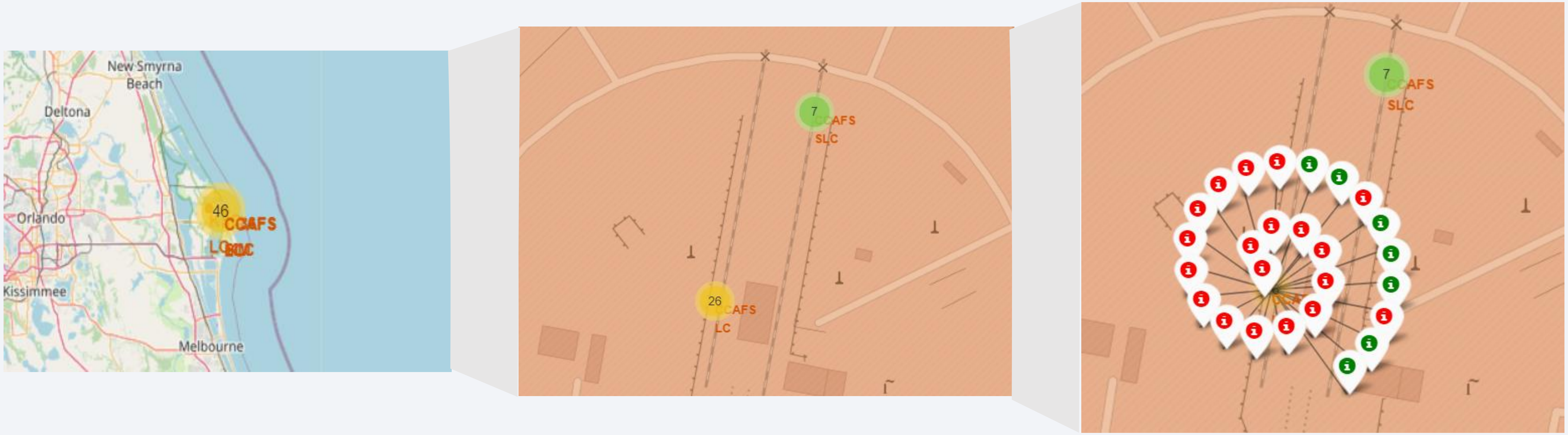
Overview of all launch sites



→ The available launch sites are all in the US, either on the west- or the east coast directly at the sea.

Colored markers representing the launch outcomes

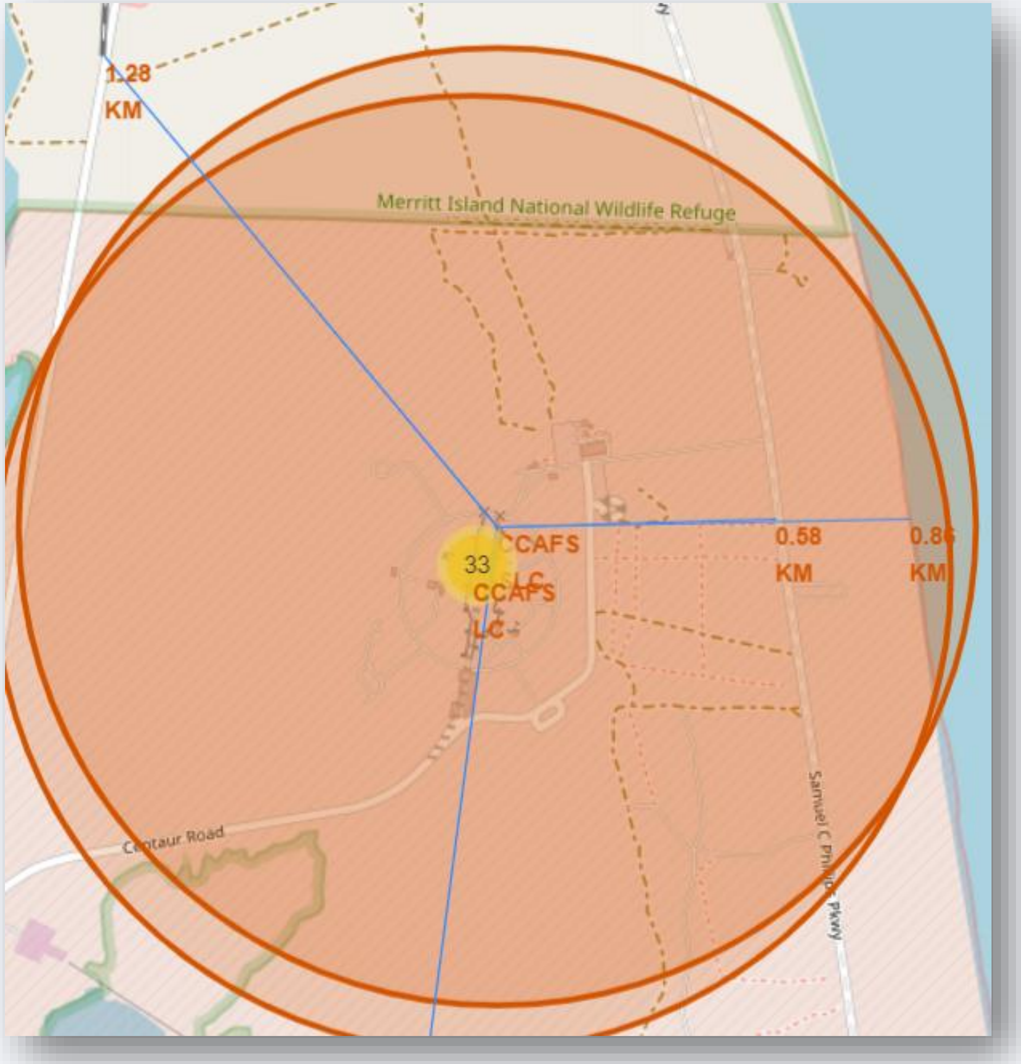
Example: Launch Site CAFS LC



Green markers indicate successful launches

Red markers indicate launches that ended in failure

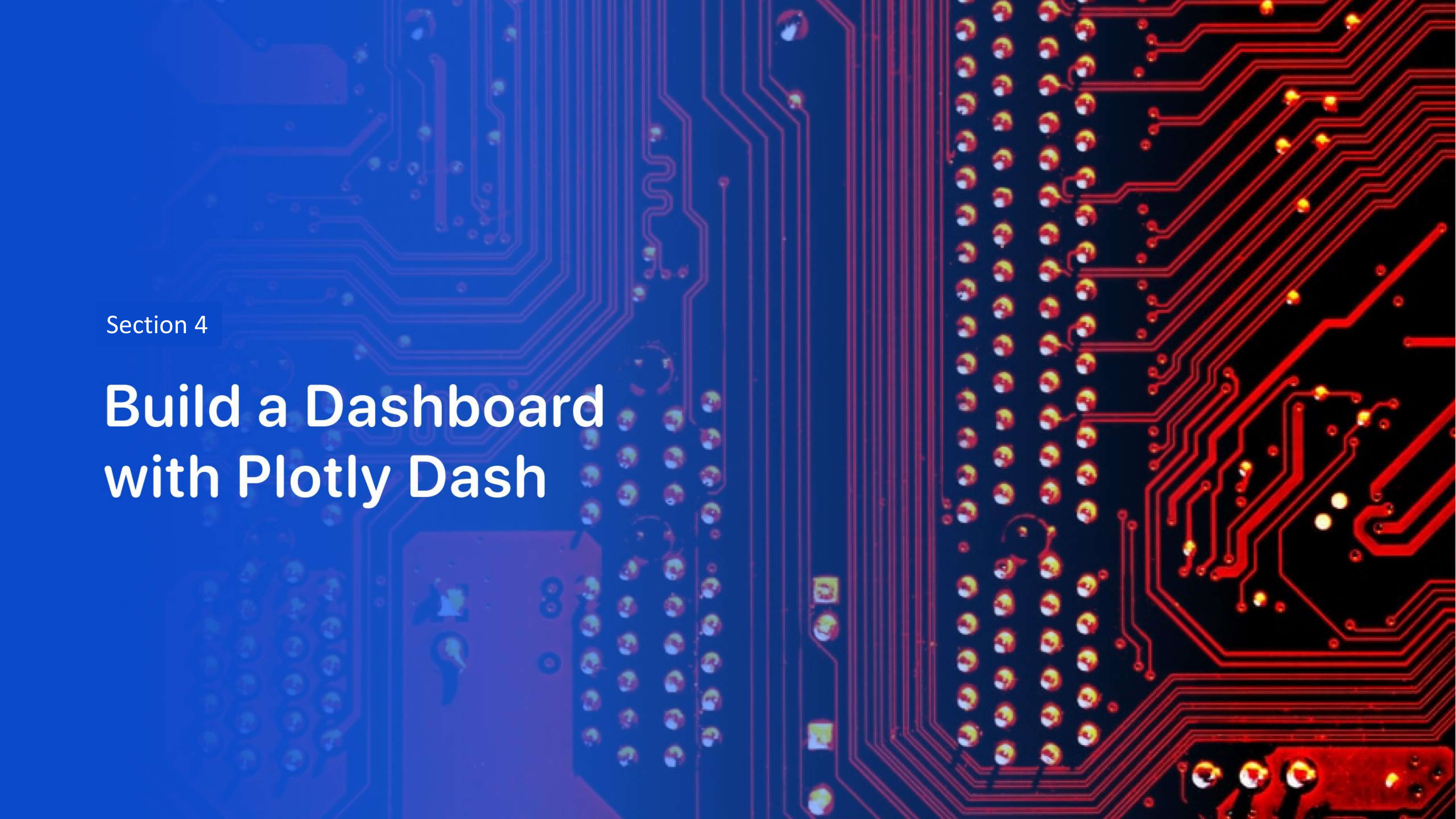
Launch site distance to infrastructure/landmarks



Distances to relevant infrastructure are drawn using a blue line

(example: Launch site CCAFS SLC-40)

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

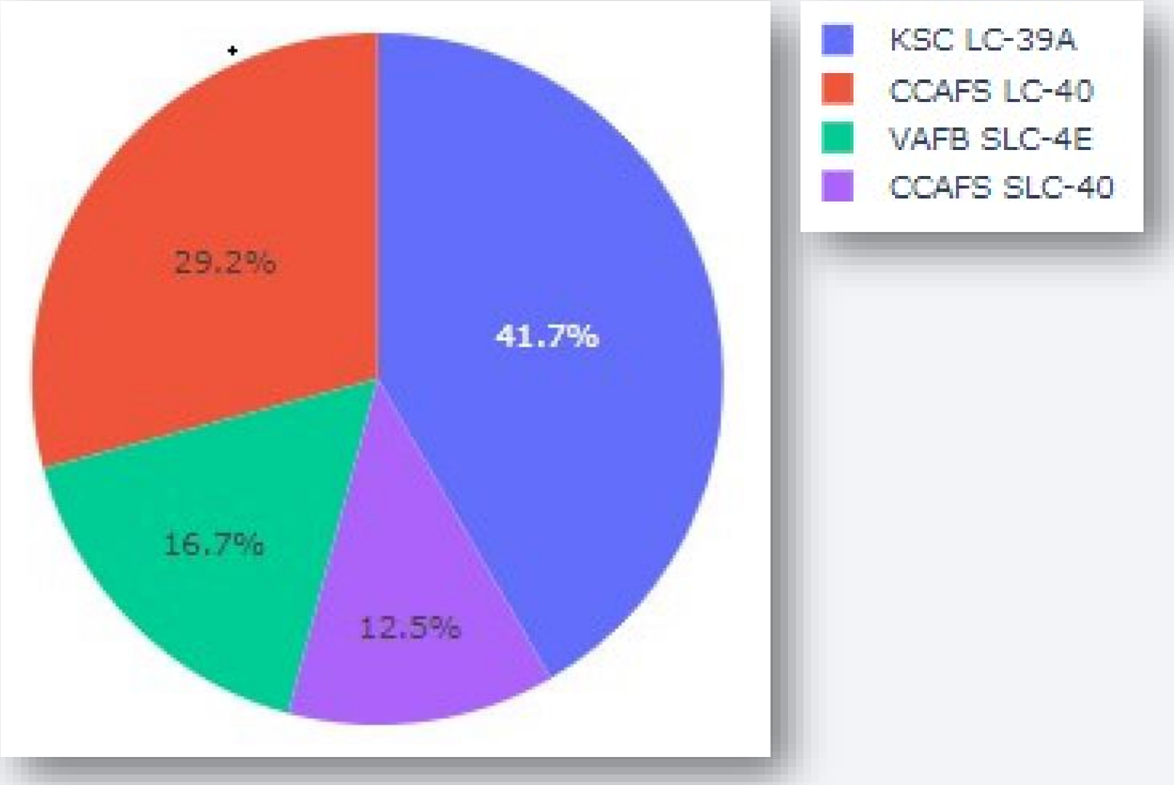


Section 4

Build a Dashboard with Plotly Dash

Total successful launches by Launch site

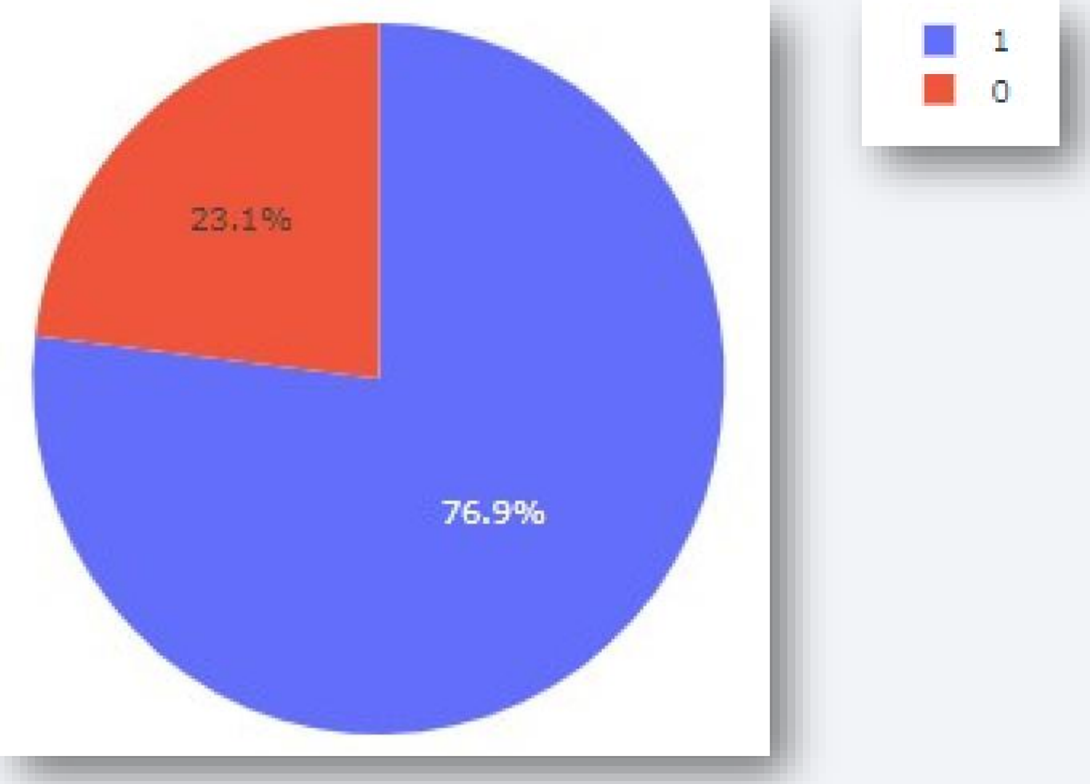
- Launch site KSC LC-39A had the highest amount of successful launches
- At Launch site CCAFS SLC-40 the lowest number of successful launches was recorded



Analysis of success rate of KSC LC-39A

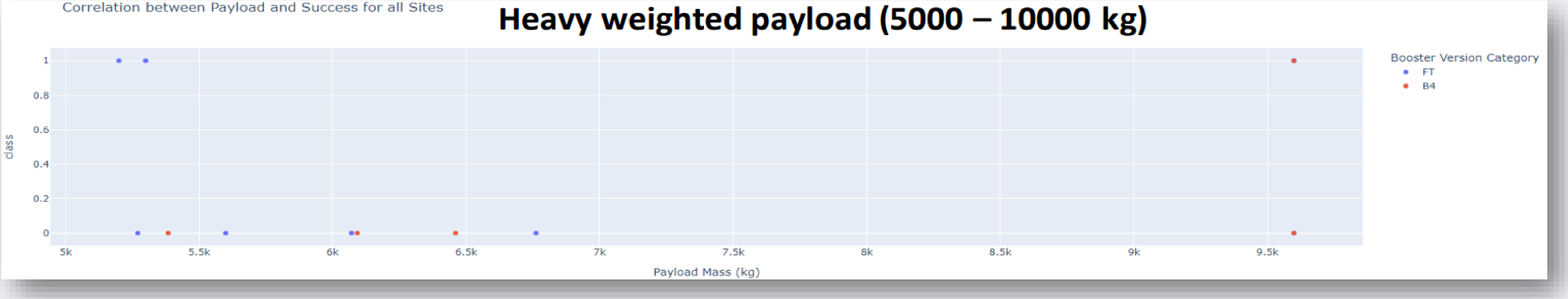
Launch site KSC LC-39A was able to achieve a 76.9% success rate for the rocket launches.

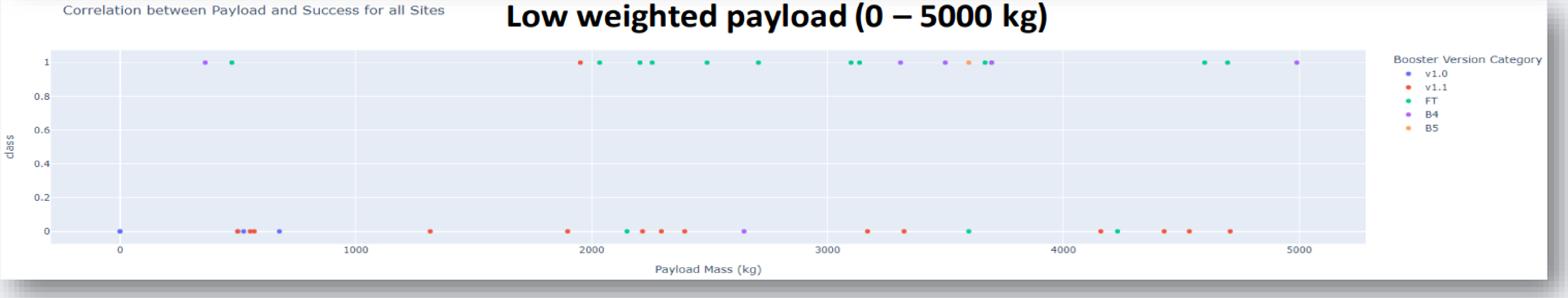
Accordingly the failure rate was 23.1%.



Relationship of Launch Outcome to Payload Mass

Through the dashboard can be deducted that low weighted payloads have a higher success rate.


Heavy weighted payload (5000 – 10000 kg)

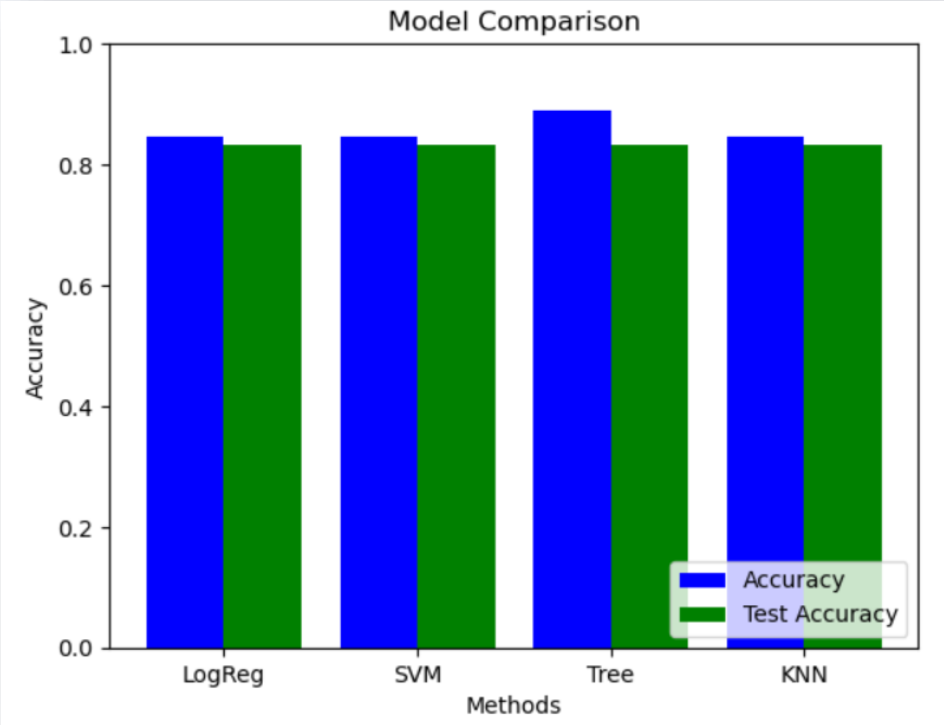

Low weighted payload (0 – 5000 kg)

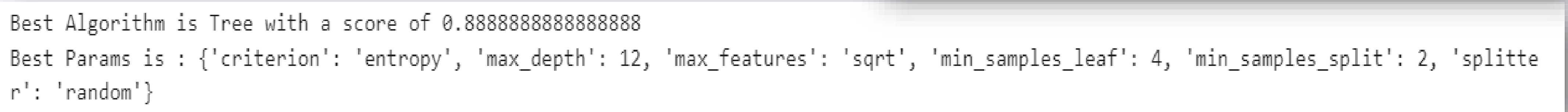
Section 5

Predictive Analysis (Classification)

Classification Accuracy

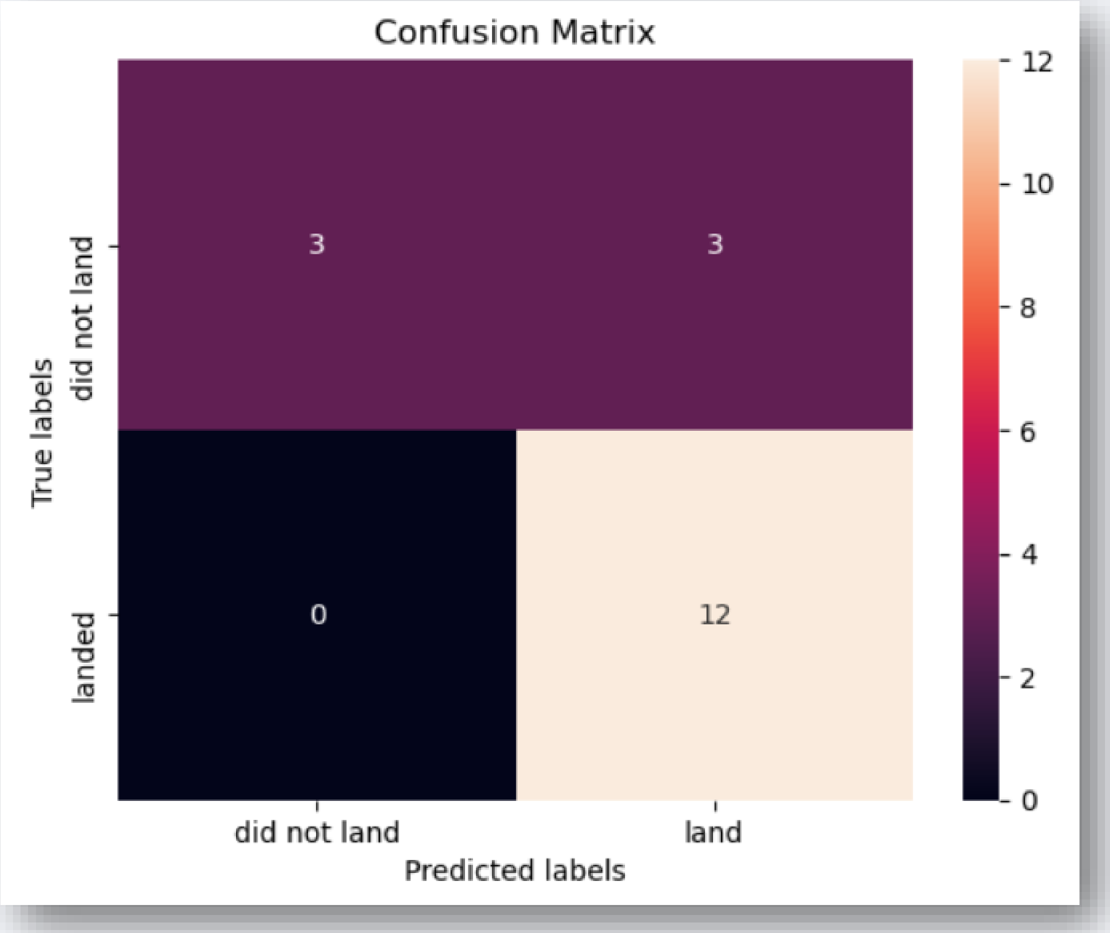
- Four different Machine Learning Models were tested:
  - Logarithmic regression
  - Support Vector Machines
  - Decision Tree Classification
  - k-Nearest Neighbors
- The best performing model is the **Decision Tree Classification**





```
Best Algorithm is Tree with a score of 0.8888888888888888
Best Params is : {'criterion': 'entropy', 'max_depth': 12, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 2, 'splitte
r': 'random'}
```

Confusion Matrix of the Decision Tree Classification



- The confusion matrix proves that the classifier can reliably distinguish between the different inputs
- The main issue are the remaining false positives (unsuccessful launches marked as successful ones)

Conclusions

- Since 2013 there is a positive trend for the number and percentage of successful rocket launches. Thereby the successes are dependent on sever factors such as the payload mass, launch orbits, launch sites and the number of previous launches. The increasing success rate also indicates a corresponding learning effect from prior launches.
- The Launch orbits with the highest success rates are HEO, SSO, ES-L1 and GEO with SSO being the most probable one.
- Not exclusively, but in summary low weighted payload masses lead to a higher success rate when compared to heavier weighted payload masses.
- According to the dataset, the Launch site with the highest success rate is KSC-LC39A.
- Out of the four tested Machine Learning Models, the Decision Tree Algorithm was selected as the best alternative due to the comparatively high accuracy statistic.

Thank you!