IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Joel Levy
30 August 2022

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- The ability to understand whether a launch will be successful in a way we can reuse the first stage of the Falcon 9 missile is crucial.
- In the research we used data from the SpaceX API and scrapped data from the Wiki page of SpaceX, then wrangled the data and applied various visualizations to check for a relationship between attributes. Searched for a relation between success rate and the location of the launching site using Folium and Dash Dashboard.
- We explored several models and successfully built a machine learning model that achieves 83% accuracy in classifying successful launches.
- In these presentation we'll discuss the various methods used for collecting data, exploring it, hyperparameters optimization and selecting the ML model.

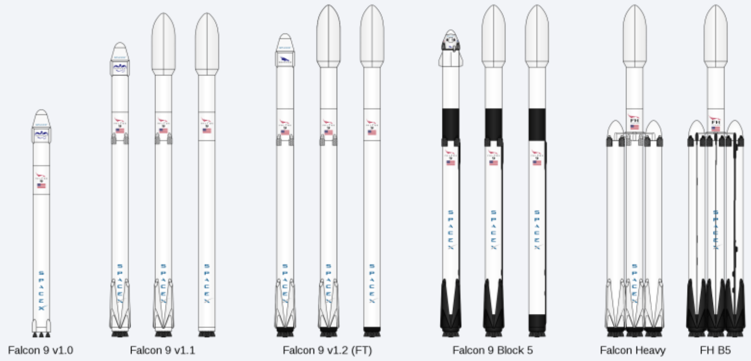
Introduction

Space X is probably the most successful commercial company, offering rocket launches to space. Its Falcon 9 rocket series launches costs only 62 million $, when other providers offer the same service in costs of 165 million $. One major reason for that is the Falcon 9's ability to reuse the rocket first stage, which is the most expansive one in the launch process. That is, of course, when the first stage of the missile succeeds in landing back on Earth.

To succeed, there are few supporting parameters that can be analyzed and shed light to increase the success rate and keep the costs low, such as the rocket payload mass, the orbit and launch site.

Till today, we can examine up to 90 different launches with varied specs. As we want to continue being the lead company in the field, we must keep the costs low, that is, bringing the first stage back in one piece.

In this research we will analyze the data, **suggest the most important attributes for a successful mission**, and **create a machine learning model to predict if a mission will be successful** bringing the first stage back before the launch is executed.



Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using several sources – SpaceX API Url and web scrapping from the SpaceX Wiki page
- Perform data wrangling
  - Data types were corrected, missing values handled, and an indicator column was added to easily identify successful launches.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models using GridCV to maximize the model's efficiency.
  - 4 models were tested – Logistic Regression, Support Vector Machine, Decision Tree Classifier and K-Nearest Neighbor

Data Collection

- Data sets were collected from several sources:
  - SpaceX API – ‘https://api.spacexdata.com/v4/’
    - Suffix ‘launches/past’ – for the past data about launches
    - Suffix ‘rockets’ – converting rockets IDs to usable data
    - Suffix ‘launchpad’ – converting launchpad IDs to usable data
    - Suffix ‘payloads’ – converting payloads IDs to usable data
  - Web scraping from the SpaceX Wiki page ‘https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches’ Using BeautifulSoup

Data Collection – SpaceX API

1. Request the SpaceX launch data using the GET request
2. Decoding the data and converting it to a DataFrame
3. Extracting more data from lists inside the data
4. Transforming IDs in the data to usable information
5. Filtering the needed data

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Using BeautifulSoup library to scrape data from the Wiki page.

1. Importing BeautifulSoup
2. Making a request.get to the Falcon9 Launch Wiki page from its URL
3. Creating a BeautifulSoup constructor
4. Extract all column/variable names from the HTML table header
5. Create a data frame by parsing the launch HTML tables

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/jupyter-labs-webscraping.ipynb

Data Wrangling

- Overall look on the data.
- Identifying and handling missing values.
- Correcting data formats.
- Making basic stats to 'understand the data'.
- Creating an indicator variable for the launching success.

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- The following charts were created:
  - Flight Nr. vs. Payload Mass – examining the success as the project progress.
  - Flight Nr. vs. Launch Site – digging deeper and examining the success in progress by sites.
  - Launch Site vs. Payload Mass – examining the influence of weight on the success by sites.
  - Orbit Type vs. Success Rate
  - Flight Nr. vs. Orbit Type

    together can show the most successful orbits.
  - Orbit Type vs. Payload – finding the best fit orbit for low-mid-high payload weight.
  - Success Rate trend over the Years – examining the learning curve of the project.

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/jupyter-labs-eda-dataviz.ipynb

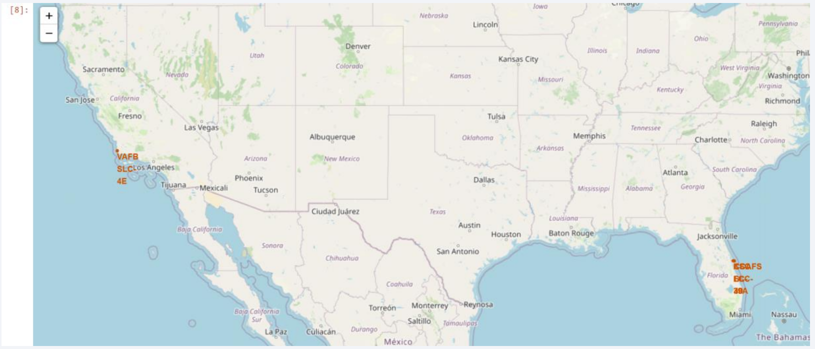
EDA with SQL

- The following queries were made to learn the data:
  - Unique launch sites in the space mission.
  - Displaying 5 launches from sites begin with 'CCA’.
  - Total payload mass carried by boosters launched by NASA (CRS).
  - Average payload mass carried by booster version F9 v1.1.
  - The date of the first successful landing in ground pad.
  - The names of the successful boosters in drone ship with payload between 4000-6000 Kg.
  - Total numbers of successful and failure mission outcomes.
  - The names of the Booster Versions which carried the max payload.
  - The failed landing in drone ship launches in 2015.
  - Ranked list of successful landings between 04-06-2010 and 20-03-2017.

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- Launch sites indicators - created a Marker and Circle folium objects for each of the launches sites and added these objects to the map by using the add_child method.
- Success / Failure launches indicators - added an indicator column for the elements color (successful - green / unsuccessful – red). Added them using a Marker Cluster object.
- Distances to proximities – were calculated using a help function, then added Marker objects and PolyLine objects to the map.



GitHub URL: https://github.com/levy-joel/Space-Race/blob/e3c5b4b93f687db0b8dffae65f5c5a03e6b8b8f7/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- The Dashboard consists of 2 charts:
  - Pie chart – showing the launches success rate from different launching sites.
  - Scatter chart – showing the success rate vs. payload mass with separation to booster version categories.
- Both charts are customizable by launching site and a range of payload mass.

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/spacex_dash_app.ipynb

Predictive Analysis (Classification)

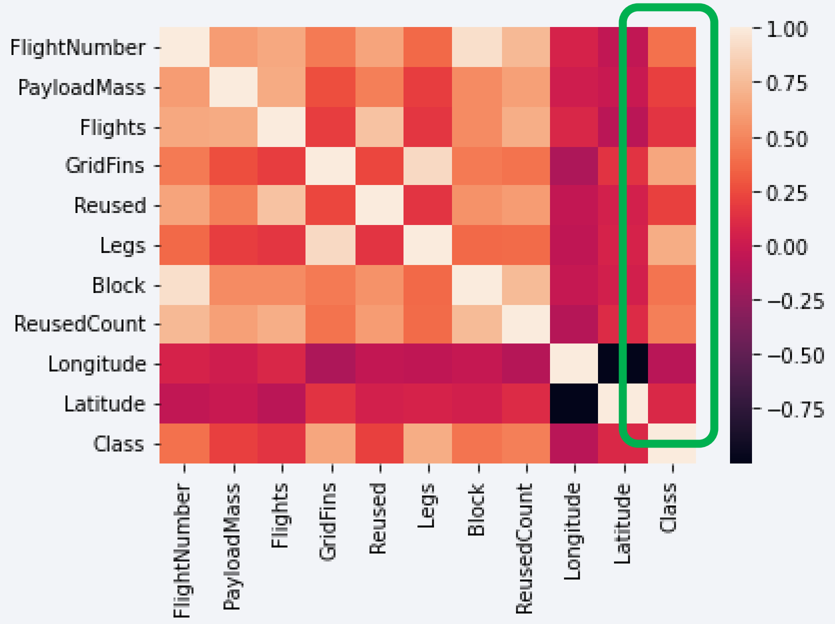
Dividing the data to independent variables (X) and target (Y) → Standardize the data → Splitting the data to Train and Test sets → Creating the model's and the GridCV's objects with possible parameters → Training the GridCV model and getting the best parameters → Making a prediction → Evaluating the model

- Used GridCV to apply 4 different models on the data:
  - Logistic Regression
  - Support Vector Machine
  - Decision Tree Classification
  - K-Nearest Neighbor
- Evaluating the models using Confusion Matrix, Accuracy Score and F1-Score.

GitHub URL: https://github.com/levy-joel/Space-Race/blob/49175eded0b851d9120dc8e589f7539ebc2ade61/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

- The most important attributes for a successful mission are:
    - By the data analysis – Launch Site, Payload Mass and Orbit Type
    - By using heatmap we can add – GridFins and Legs.
- We can see clearly that a positive learning curve exists.
- Based on the analysis, we can make a prediction, using Machine Learning Model, of 83% accuracy.

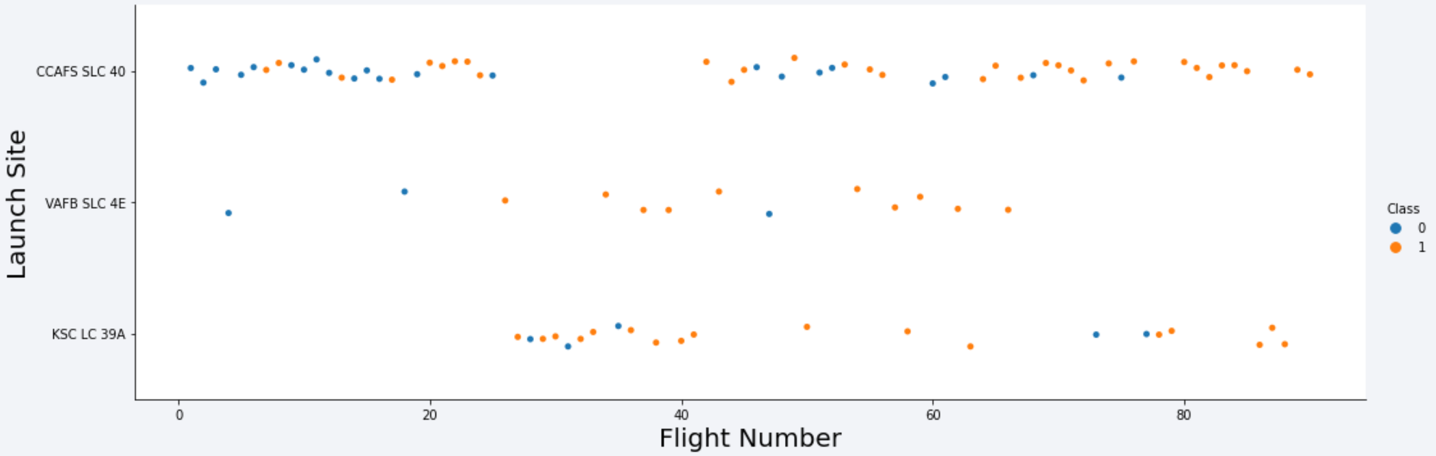


Section 2

Insights drawn from EDA

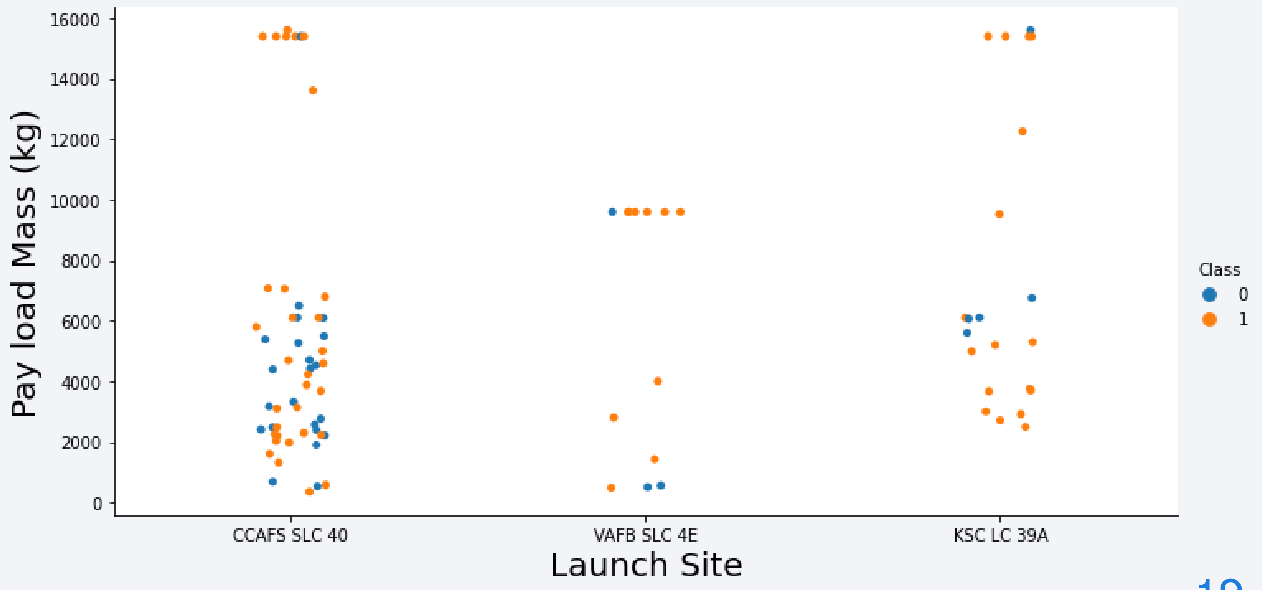
Flight Number vs. Launch Site

- The CCAFS SLC 40 was and remained the main launches site over time. We can also see that the number of successful launches from it increased significantly.
- The VAFB SLC 4E was abended with time, although most of the launches from it were successful.
- The KSC LC 39A has very good rates of successful launches, and it serves as a secondary launch site.



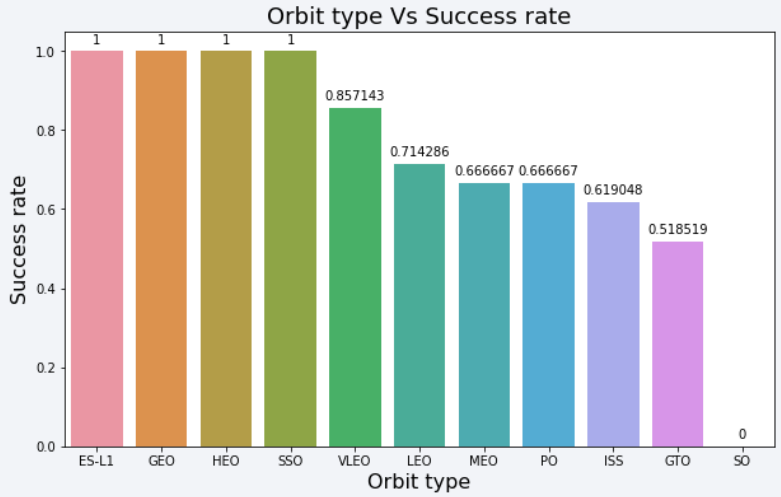
Payload vs. Launch Site

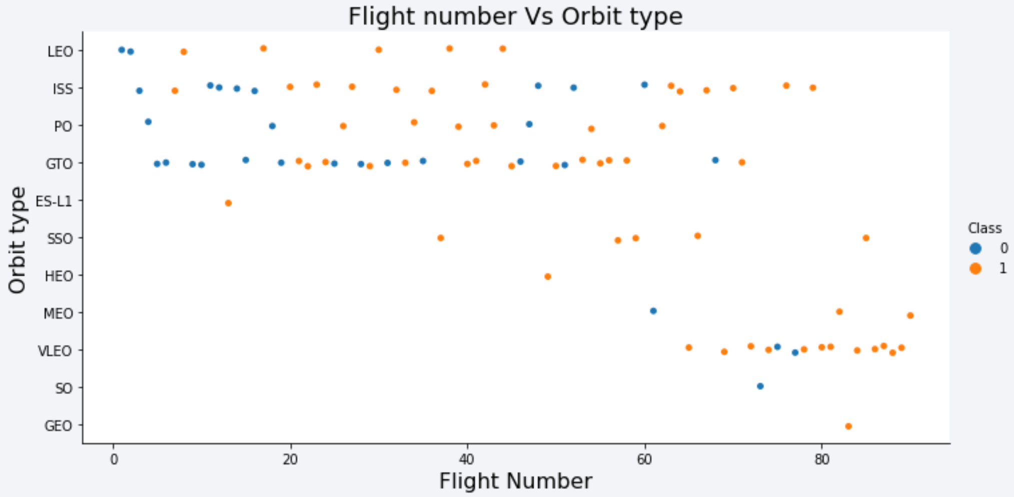
- When dealing with heavy payloads the CCAFS SLC 40 and KSC LC 39A best fit.
- In the mid-range payloads of 10000Kg VAFB SLC 4E best fit. It's not suitable for the high-range.
- In the low-range payloads beneath 5000Kg KSC LC 39A best fit. It's not suitable for the mid-range.
- Heavy load launches are less ambiguous and have higher success rate.



Orbit Type vs. Success Rate and Flight Number

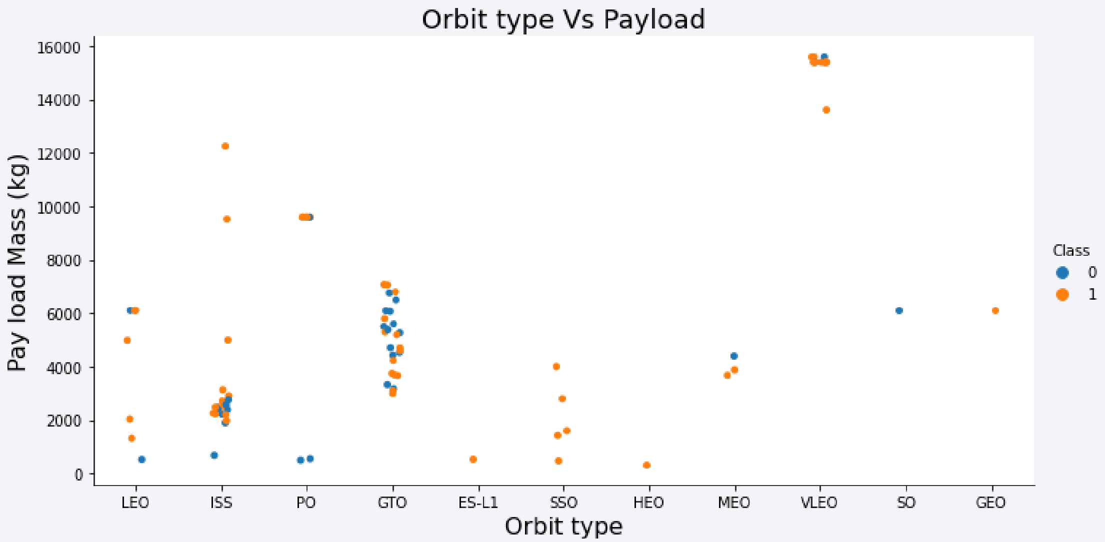
- The most successful launch is in the VLEO orbit type (All the Orbit types with the highest success rate (1) had been done only few times).
- It appears that there is a relationship between the experience in an Orbit type to the success in that Orbit but including the first period (~20) launches in the success rate makes it much lower.





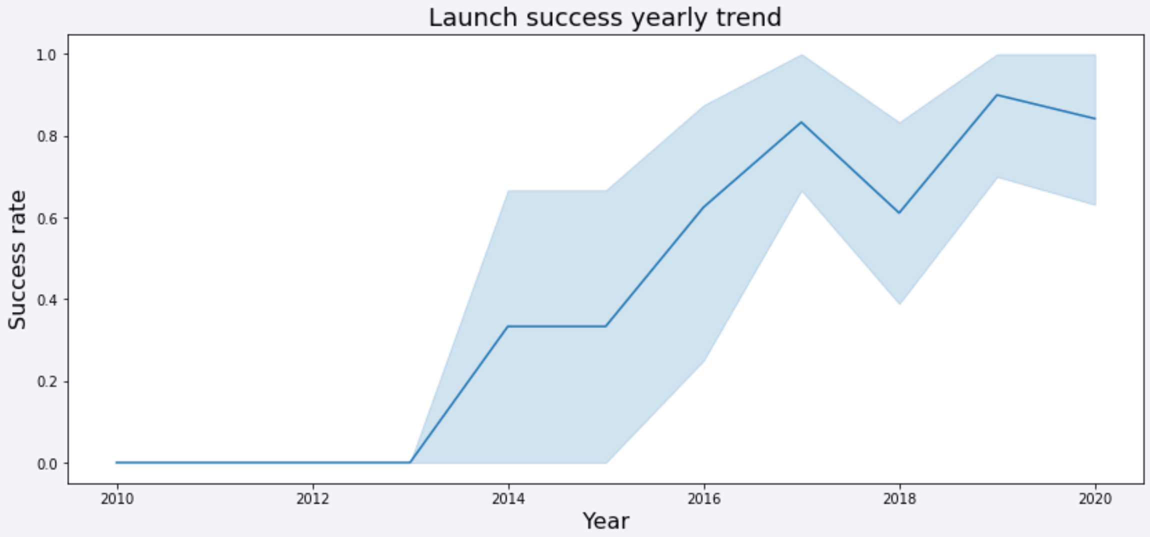
Payload vs. Orbit Type

- In general, heavy load launches are less ambiguous and have higher success.
- The ISS, PO and VLEO orbits are best for high-range payloads.
- The LEO orbit is best for mid-range payloads.
- The SSO orbit is best for low-range
- The GTO orbit looks very risky, but as we can see in the last slide it has improved with time.



Launch Success Yearly Trend

- Since 2013, there has been as upward and continuous trend in the success rate (except for two drop backs in 2018 and 2020).
- Since the mid of 2015 there's a higher chance for a launch to succeed (over 50%).



All Launch Site Names

- Using DISTINCT to get unique values.

Task 1

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

[22]:
```
%%sql query_task1 <<
SELECT DISTINCT Launch_Site
FROM SPACEXTBL
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task1
```

[23]:
```
query_task1
```

[23]:

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

Launch Site Names Begin with 'CCA'

- Using LIKE to find the Launch sites begin with ‘CCA’.
- Using LIMIT to fetch only 5 records.

Task 2

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

[32]:
```
%%sql query_task2 <<
SELECT *
FROM SPACEXTBL
WHERE Launch_Site LIKE 'CCA%'
LIMIT 5
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task2
```

[33]:
```
query_task2
```

[33]:

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

- Using WHERE to filter the query to NASA's launches.
- Using SUM function.

Task 3

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

```
[40]: %%sql query_task3 <<
SELECT Customer, SUM(PAYLOAD_MASS__KG_) AS 'Total payload mass'
FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)'
GROUP BY Customer
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task3
```

```
[41]: query_task3
```

[41]:

Customer	Total payload mass
NASA (CRS)	45596

Average Payload Mass by F9 v1.1

- Using WHERE to filter the query to ‘F9 v1.1’ booster version.
- Using AVG function.

Task 4

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

[46]:
```
%%sql query_task4 <<
SELECT Booster_Version, AVG(PAYLOAD_MASS__KG_) AS 'Average payload mass'
FROM SPACEXTBL
WHERE Booster_Version = 'F9 v1.1'
GROUP BY Booster_Version
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task4
```

[47]:
```
query_task4
```

[47]:

Booster_Version	Average payload mass
F9 v1.1	2928.4

First Successful Ground Landing Date

- Using WHERE to filter the query to Successful Ground pad landings.
- Using MIN function.

Task 5

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

Hint:Use min function

```
[58]: %%sql query_task5 <<
SELECT "Landing _Outcome", MIN(Date) AS Date
FROM SPACEXTBL
WHERE "Landing _Outcome" = 'Success (ground pad)'
GROUP BY "Landing _Outcome"
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task5
```

```
[59]: query_task5
```

[59]:

Landing _Outcome	Date
Success (ground pad)	01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using WHERE and BETWEEN … AND … to filter the outcomes.

Task 6

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

[74]:
```
%%sql query_task6 <<
SELECT Booster_Version, "Landing _Outcome", PAYLOAD_MASS__KG_
FROM SPACEXTBL
WHERE "Landing _Outcome" = 'Success (drone ship)' AND
    PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
ORDER BY Booster_Version
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task6
```

[75]:
```
query_task6
```

[75]:

Booster_Version	Landing _Outcome	PAYLOAD_MASS__KG_
F9 FT B1021.2	Success (drone ship)	5300
F9 FT B1031.2	Success (drone ship)	5200
F9 FT B1022	Success (drone ship)	4696
F9 FT B1026	Success (drone ship)	4600

Total Number of Successful and Failure Mission Outcomes

- Using IIF and LIKE to set overcome different types of success and failure.

Task 7

List the total number of successful and failure mission outcomes

```
[86]: %%sql query_task7 <<
SELECT IIF(Mission_Outcome LIKE '%Success%','Success','Failure') AS Mission_Outcome, COUNT(*) AS Num_of_missions
FROM SPACEXTBL
GROUP BY IIF(Mission_Outcome LIKE '%Success%','Success','Failure')
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task7
```

```
[87]: query_task7
```

[87]:

Mission_Outcome	Num_of_missions
Failure	1
Success	100

Boosters Carried Maximum Payload

- Using a subquery to find the MAX value of payload mass.
- Using DISTINCT to get unique values of the Booster Version that fits the weight.

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

[91]:
```
%%sql query_task8 <<
SELECT DISTINCT Booster_Version
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
ORDER BY Booster_Version
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task8
```

[92]:
```
query_task8
```

[92]:

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

- Using SUBSTR to find the Month and the Year (because SQLite doesn’t support MONTHNAME() function

Task 9

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

Note: SQLLite does not support monthnames. So you need to use substr(Date, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

[146]:
```
%%sql query_task9 <<
SELECT SUBSTR(Date,7,4) AS Year, SUBSTR(Date,4,2) AS Month, "Landing _Outcome", Booster_Version, Launch_Site
FROM SPACEXTBL
WHERE SUBSTR(Date,7,4) = '2015' AND
    "Landing _Outcome" = 'Failure (drone ship)'
```

```
 * sqlite:///my_data1.db
Done.
Returning data to local variable query_task9
```

[147]:
```
query_task9
```

[147]:

Year	Month	Landing _Outcome	Booster_Version	Launch_Site
2015	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015	04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Using RANK OVER ‘Landing_Outcome’.
- Using WHERE and BETWEEN … AND … and LIKE to filter the outcomes.
- Using ORDER BY and DESC for descending order of the list by ‘Date’.

Task 10

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

```
[167]: %%sql query_task10 <<
SELECT Date, "Landing _Outcome", RANK() OVER(PARTITION BY "Landing _Outcome" ORDER BY Date DESC) AS Rank
FROM SPACEXTBL
WHERE Date BETWEEN '04-06-2010' AND '20-03-2017' AND
       "Landing _Outcome" LIKE '%Success%'
```

```
[168]: query_task10
```

[168]:

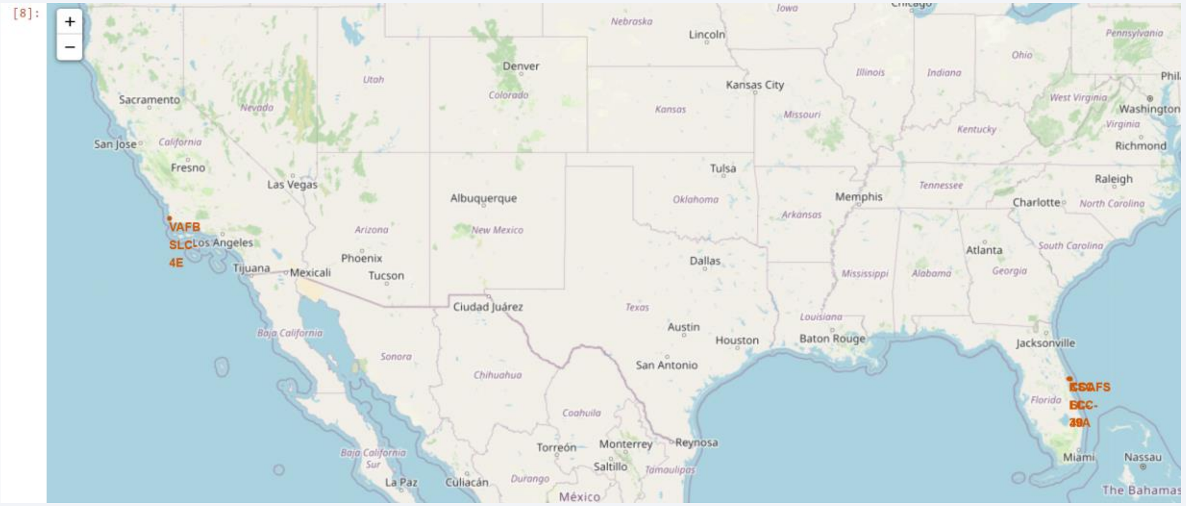
Date	Landing _Outcome	Rank
18-10-2020	Success	1
18-08-2020	Success	2
17-12-2019	Success	3
16-11-2020	Success	4
15-11-2018	Success	5
13-06-2020	Success	6
12-06-2019	Success	7
11-11-2019	Success	8
11-01-2019	Success	9
10-09-2018	Success	10
08-10-2018	Success	11
07-08-2020	Success	12
07-08-2018	Success	13
07-03-2020	Success	14
07-01-2020	Success	15
06-12-2020	Success	16
06-10-2020	Success	17
05-12-2019	Success	18
05-11-2020	Success	19
04-06-2020	Success	20
18-04-2018	Success (drone ship)	1
14-08-2016	Success (drone ship)	2
14-01-2017	Success (drone ship)	3
11-10-2017	Success (drone ship)	4
11-05-2018	Success (drone ship)	5
09-10-2017	Success (drone ship)	6
08-04-2016	Success (drone ship)	7
06-05-2016	Success (drone ship)	8
19-02-2017	Success (ground pad)	1
18-07-2016	Success (ground pad)	2
15-12-2017	Success (ground pad)	3
14-08-2017	Success (ground pad)	4
08-01-2018	Success (ground pad)	5
07-09-2017	Success (ground pad)	6

Section 3

Launch Sites Proximities Analysis

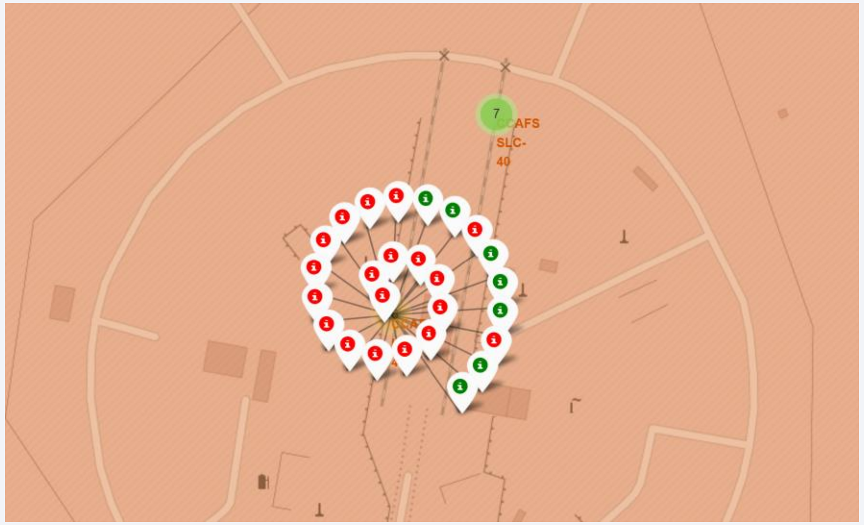
Launch Sites

- We can clearly see that there are 3 launch sites on the east coast as opposed to only one on the west coast.
- All launch sites situated very close to the see, probably because of two major reasons:
  - Getting as far as possible from populated areas.
  - There is a need for large amounts of water for cooling down the machinery.



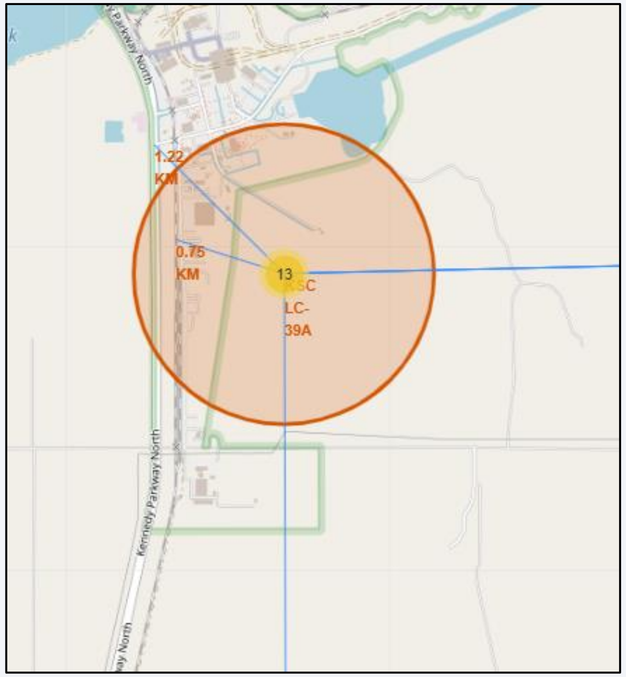
Success / Failed Launches on Site

- Visualizing the launches on each site enables to understand right away the success rate of the site.



KSC LC-39A Launch Site Proximities

- The launch site is very close to transportation ways such as a Highway (only 1.22 Km) and a Railroad (only 0.75 Km).
- The nearest city is Melbourne (52 Km).



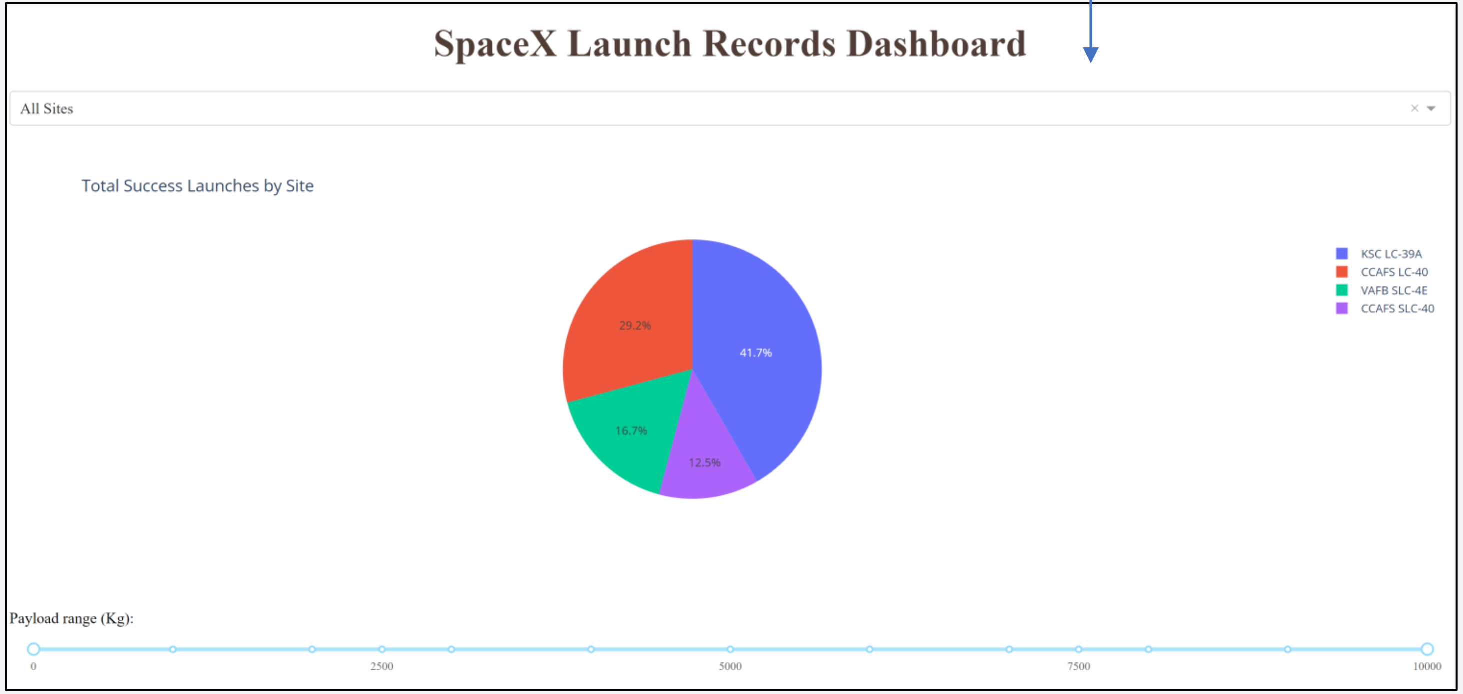
Section 4

Build a Dashboard with Plotly Dash

Dashboard - Launch Success - Key Elements

- We can filter by Launch Site and by Payload mass.
- All Sites look - general view of which launch site has the highest launch success rate.
- Specific Site look - site's success rate.

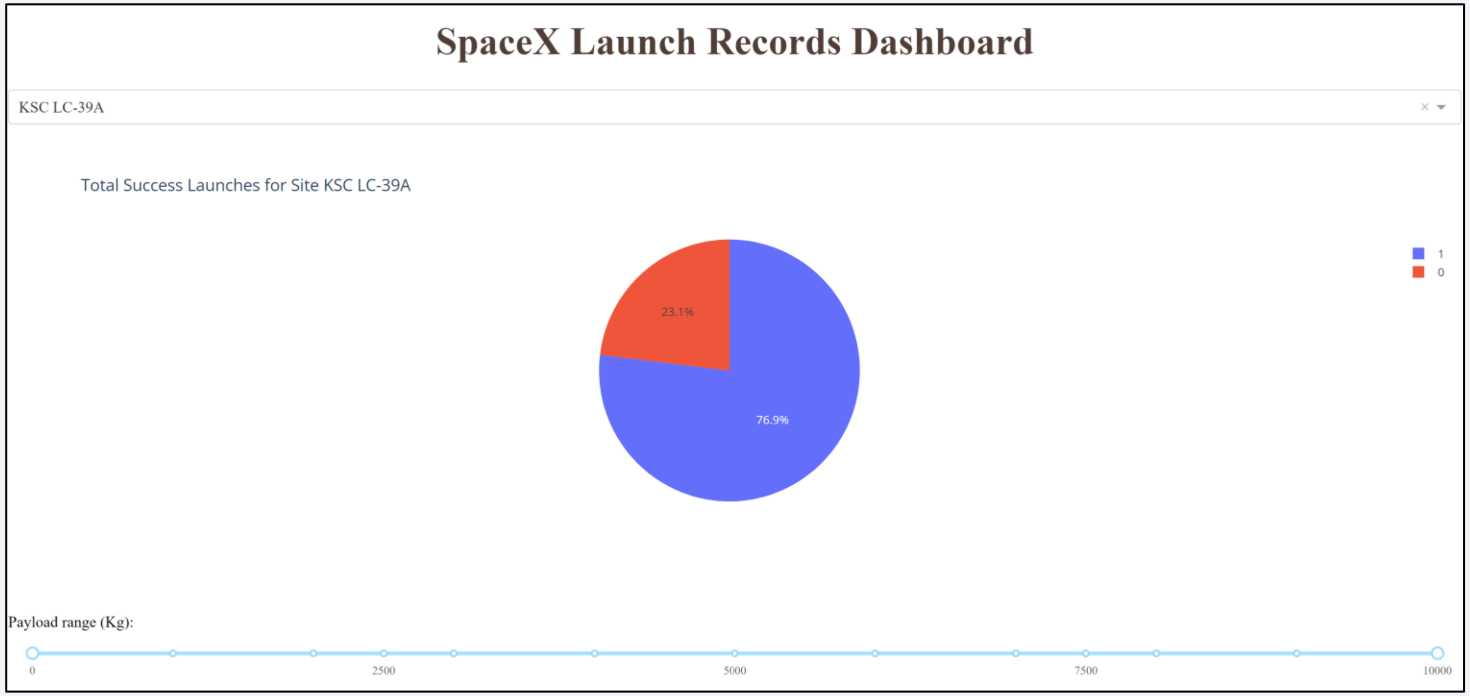
The upper dropdown selector allows us to choose the wanted launch site



The lower slider allows to filter the wanted payload mass

Dashboard - Highest Launch Success Site

- 41.7% of the successful launches were made from the KSC LC-39A site.
- 76.9% Successful launches and only 23.1% failures.



Dashboard - Payload vs. Launch Outcome (All Sites)

- Looking at the medium level Payload (3000-6000 Kg) for all sites.
- The data is inconclusive regarding which booster version is best for these payloads.

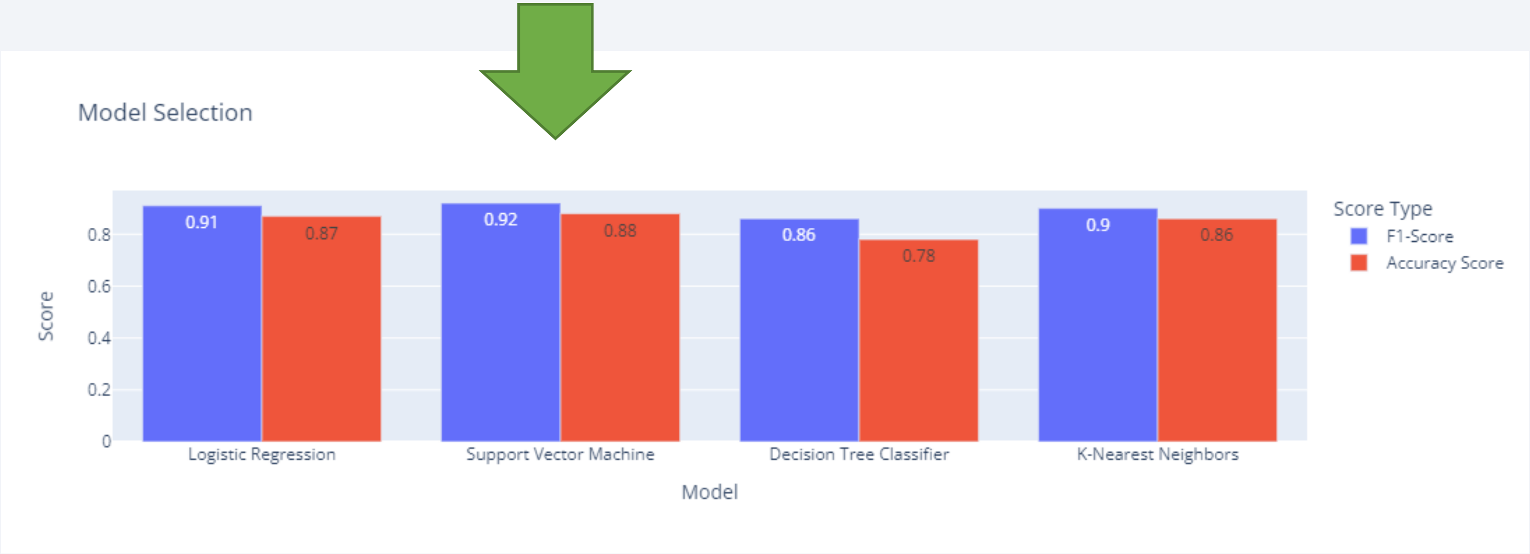
By clicking on an object in the legend we can filter the wanted booster version

Section 5

Predictive Analysis (Classification)

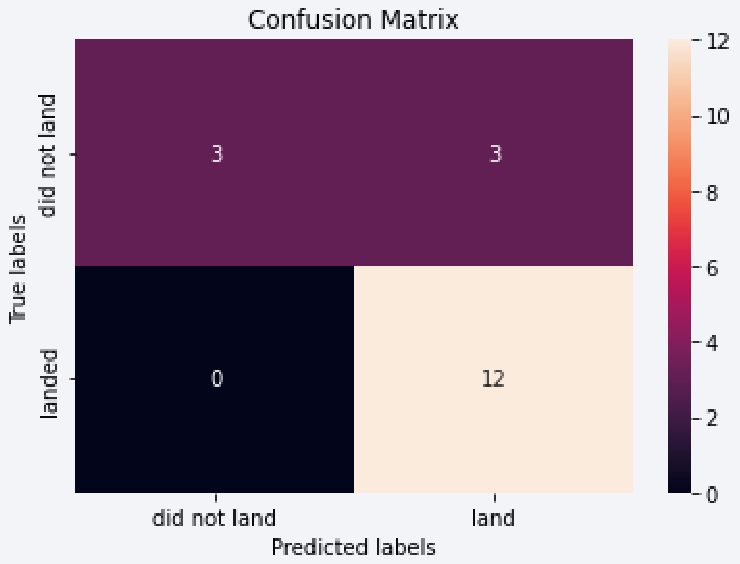
Classification Accuracy

- The models were tested with both the F1-Score and Accuracy Score.
- SVM model was ranked the highest on both.



Confusion Matrix – SVM model

- SVM can distinguish between the different classes with accuracy level of 83%.
- It has still an error level of 17% as we can see in the False-Positive.



Conclusions

- The recommended combinations between Launch Site - Payload - Orbit Type

Launch Site	High Payload (> 10000)	Mid Payload (5000 – 10000)	Low Payload (< 5000)
CCAFS SLC 40	1. VLEO 2. ISS 3. PO	Not Recommended	Not Recommended
VAFB SLC 4E	Not Recommended	1. LEO 2. GTO	Less Recommended
KSC LC 39A	1. VLEO 2. ISS 3. PO	Not Recommended	1. SSO

- The KSC LC-39A site is the most successful launch site with a rate of 76.9%.
- Heavy load launches are less ambiguous and have higher success rate.

Thank you!