IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Bo Madsen

June 6 2025

Outline

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

Executive Summary

In this project several steps has been carried out to

- Retrieve Data
- Shape Data
- Explore and Visualize Data
- Build predition models

All with the purpose of understanding and predicting how likely it is that it is possible to land the Phase One of the rocket which can be re-used for future launches and reduce the cost.

The final conclusion is that the developed prediction models are relativly accurate in predicting if the Phase One of a rocket used in a given launch can be landed successfully.

Introduction

SpaceX is a private space company which is the only company to have successfully landed part of the rocket launch material after payload of the rocket is successfully sent into orbit. SpaceX announce on their webpage that the cost of a Falcon 9 rocket launch is 62 million $ where competitors may charge up to 165 million $. Most of the savings is because SpaceX can land and re-use the First Phase of the rocket.

The objective with this data science project is therefore to aim to predict for a given rocket launch if the First Phase can be successfully landed based relevant known data about each launch. This prediction can be used to estimate the price of a launch.

Section 1

Methodology

Executive Summary

- Data collection methodology:
    - The data was collected by using Web Scraping of a Wikipedia page called List of Falcon 9 and Falcon Heavy launches and by using a Rest API to retrieve the data from a database
- Perform data wrangling
    - Since the data is formatted in different ways which are not usable for Machine Learning algorithms, the data had to be explored and formatted. E.g. the outcome of a landing was given in different text formats in the original data, but was transformed to either 1 for successful landing or 0 for failed lading
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - How to build, tune, evaluate classification models

Data Collection

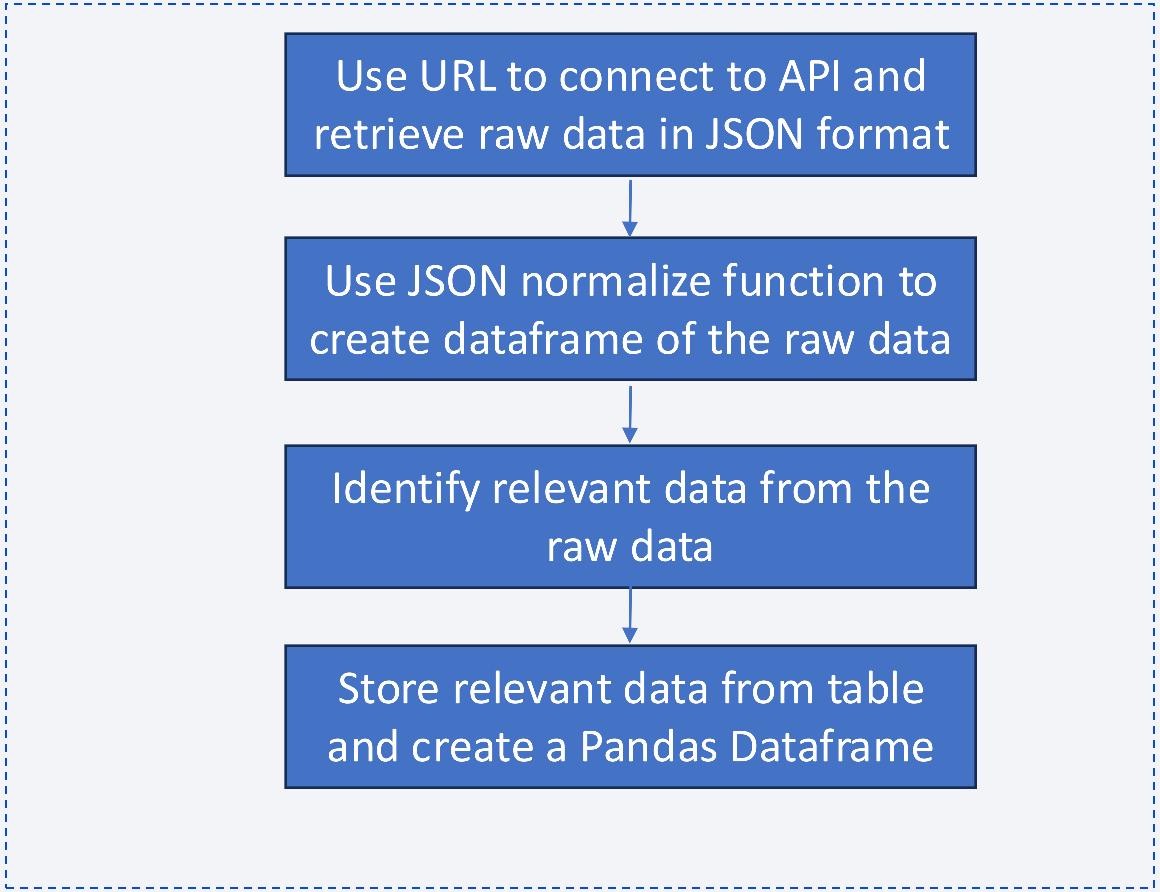
The data for the project was collected in two different ways (mainly to demonstrate the ability to use both methods)

1. Using a SpaceX Rest API to retrieve the data directly from a SpaceX database
2. Using Web Scraping of HTML table from a Wikipedia site with detailed SpaceX information about the rocket launches

Data Collection – SpaceX API

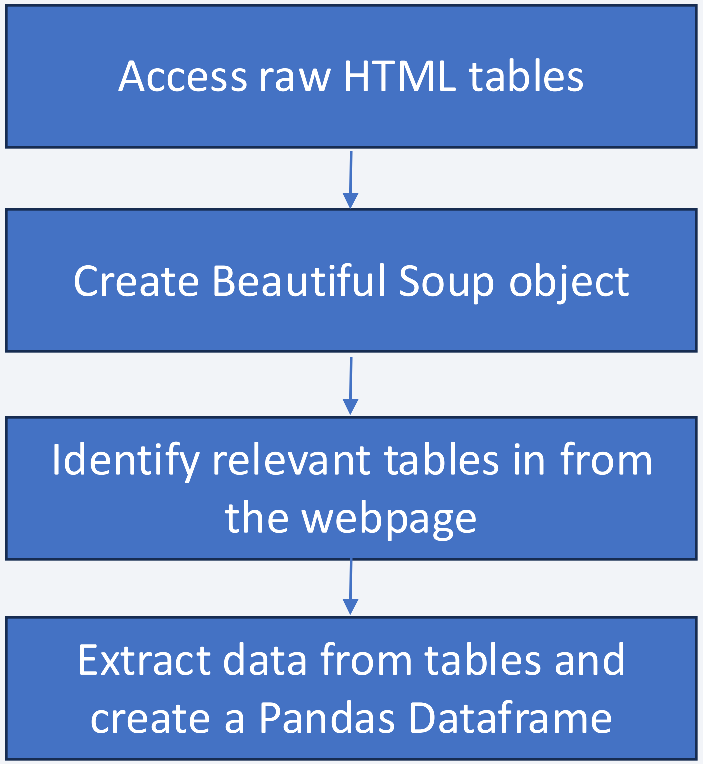
- The Rest API is used to retrieve a vast amount of raw data about the launches from SpaceX
- The launch data can be normalized with the Pandas json_normalise function and relevant data can be picked from the dataframe containing the raw data

Link to lab notebook: https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- The web scraping was performed using Beautify Soup which is a python package to parse HTML
- Once the HTML code is stored as a Beautiful Soup object it is possible to access specific tables and cells in them to retrieve data and store it in a Pandas dataframe for further processing



Link to lab notebook:
https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

Once the data is retrieved there is a need to change the data to be able to use it for e.g. Machine Learning models

E.g. the outcome of each rocket launch which is what we are aiming to predict (as success or failure) is given with severeal different text labels like:

- True Ocean
- False Ocean
- True ASDS
- False ASDS

Outcome	Class
False Ocean	0
True Ocean	1
None None	0
None None	0
False Ocean	0
False ASDS	0
True ASDS	1

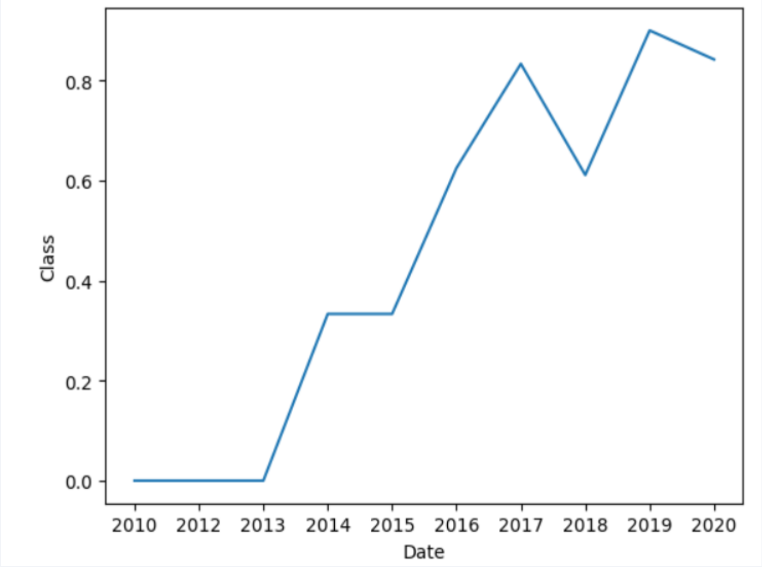
Each of the text labels has to be changed into a value 0 for failed landing or 1 for successfully landing to be used as the predicted class in a Machine Learning model

Link to lab notebook: https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/labs-jupyter-spacex-data%20wrangling_jupyterlite.ipynb

EDA with Data Visualization

During the exploratory data analysis there was created a number of different visualizations which helps to understand the rocket launch data and the relation to the landing success rate. There was e.g. created:

- Scatterplot to show the landing outcome based on payload and flight number
- Scatterplot to show the landing outcome based on launch site and flight number
- Lineplot to show the yearly landing success rate development



Link to lab notebook:
https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/edadataviz.ipynb

EDA with SQL

As a part of the project, the SpaceX data was loaded into a database and the following SQL queries was executed get more insight in the data:

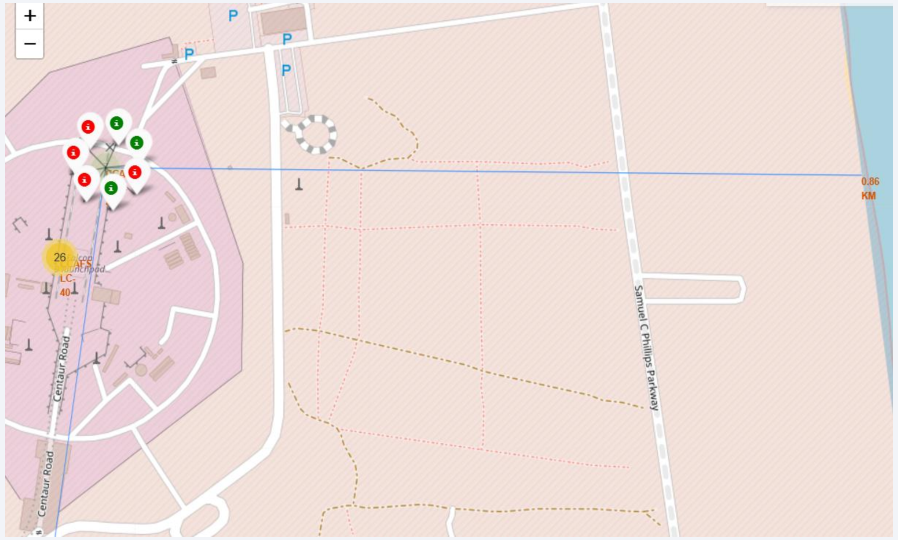
1. Display the unique list of launch sites and booster versions
2. Display 5 rows where the launch site starts with a specific string ("KSC")
3. Display the sum of the total payload from all the launches
4. Display the average sum of payloads with a specific booster version
5. List the rows where a successful landing was performed on a drone ship
6. List successful landings for a given landing site and payload within a range
7. List the total number of success and failed landings
8. Using a subquery to list successful landings with all booster version carrying max payload

Link to lab notebook:
https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/jupyter-labs-eda-sql-edx_sqllite.ipynb

Build an Interactive Map with Folium

For a visualization of the launch site on a Map, Folium was used for amongst others to:

- Indicate the locations of the launch site as circles
- Add the launches a clusters on the map – which makes it possible to click and investigate successful/failed launches for each site
- Indicate proximities as nearest railway or coast etc. to the launch sites and create a line to indicate the distance to the proximities.
- Use active mouse pointer coordination to calculate the distance to proximities



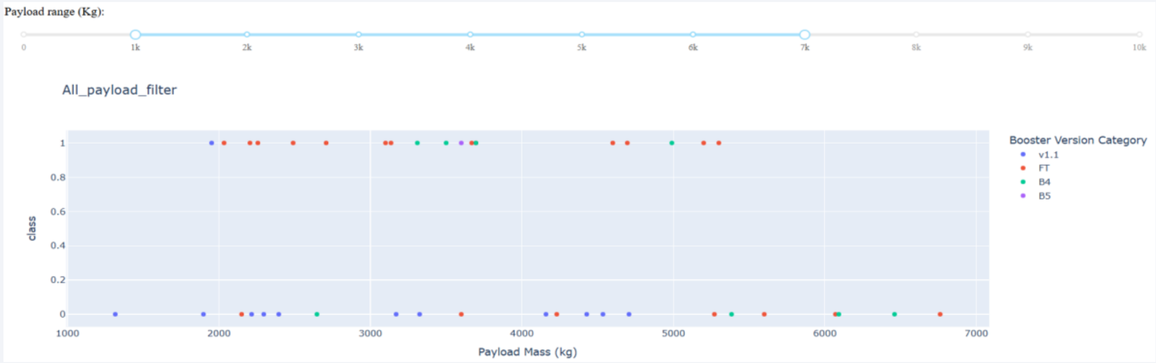
Link to lab notebook:
https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

An interactive dashboard was created with Plotly Dash to be able to gain useful insight in the data by filtering to e.g. specific launch sites or a given range of payload. The interactive dashboard is able to:

- Provide pie charts with success rate for selected launch sites
- Provide scatterplots to indicate correlation between payload and launch outcome for selected booster versions

Link to lab notebook:
https://github.com/madsenbo/IBM-Data-Science-
--Capstone-Project/blob/main/spacex-dash-app_final_2.py

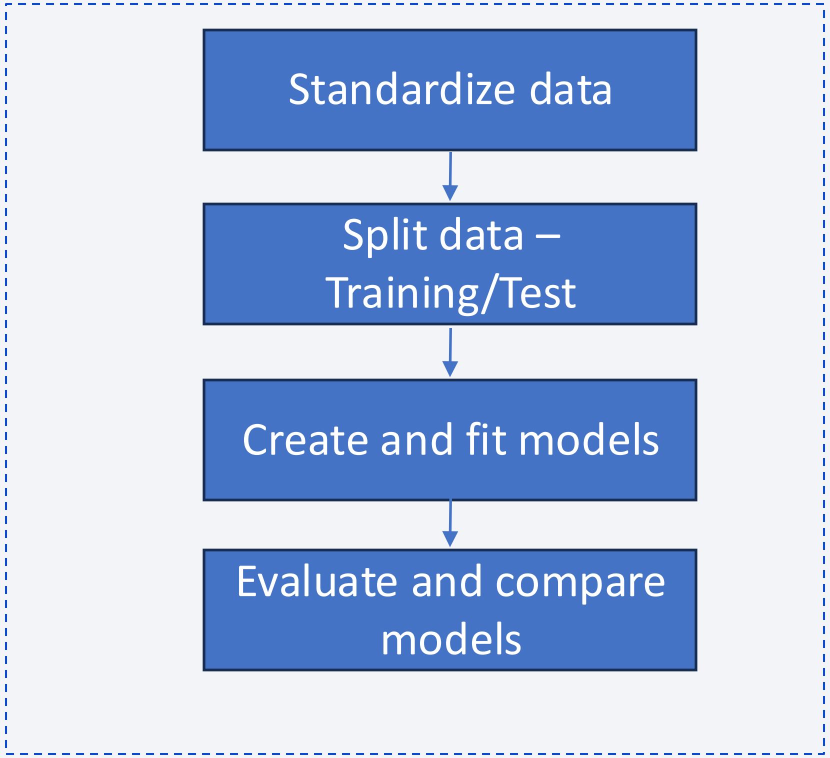


Predictive Analysis (Classification)

A predictive model was build with the aim to be able to predict the outcome of future launches and whether it is expected for each launch that the first phase of the rocket can land successfully and be re-used for other launches

To build the model the following steps where necessary:

1. Transform and standardize the data (both variables and outcome)
2. Split the data into training and test data
3. Create several different models and fit them
    - Logistic regression model
    - Support Vector Machine (SVM) model
    - Decision Tree Classifier model
    - K Nearest Neighbour model
4. Evaluate the model performance of each model and compare them



Link to lab notebook:
https://github.com/madsenbo/IBM-Data-Science---Capstone-Project/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

- The results drawn from the exploratory data analysis will be presented under section 2 "Insights drawn from EDA"
- The results from the interactive analytics demo is presented in Section 3 "Launch Site Proximities Analysis" and Section 4 "Build a Dashboard with Plotly Dash"
- The results of the predictive analysis is presented in Section 5 "Predictive Analysis (Classification)"

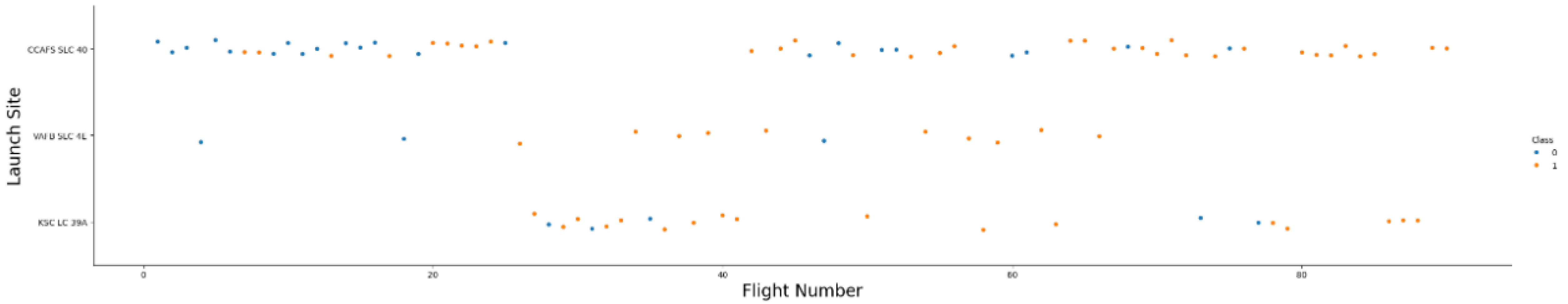
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

The below scatterplot shows each launch based on its incremental flight number split into the different launch sites. The launch is indicated with orange color if the launch outcome was successful and blue if it failed.

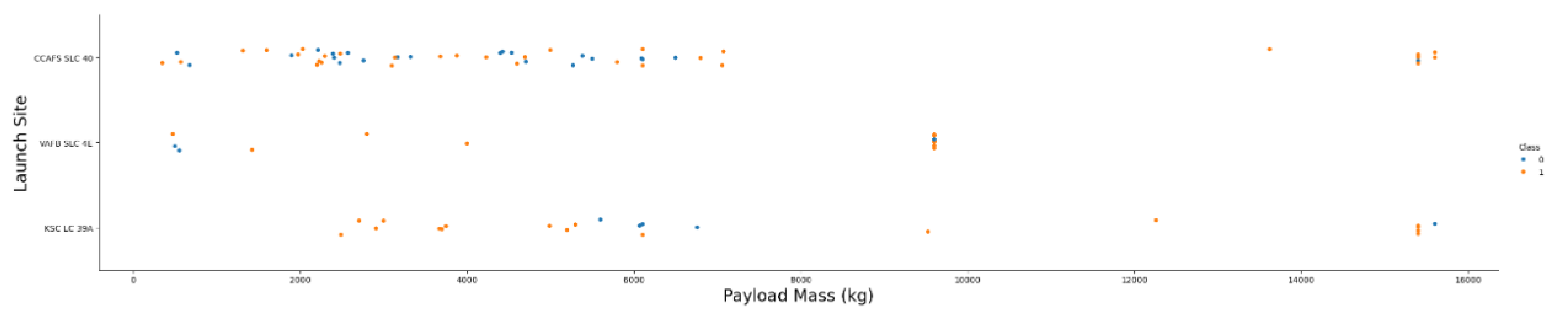
I can be seen quite clearly that the success rate for each launch site is better the more launches has been carried out on the given launch site



Payload vs. Launch Site

The below scatterplot shows each launch based on its payload split into the different launch sites. The launch is indicated with orange color if the launch outcome was successful and blue if it failed.

It shows that most launches are carried out with a payload below 7000 kg, but also that the launches with higher payload have a rather good success rate



Success Rate vs. Orbit Type

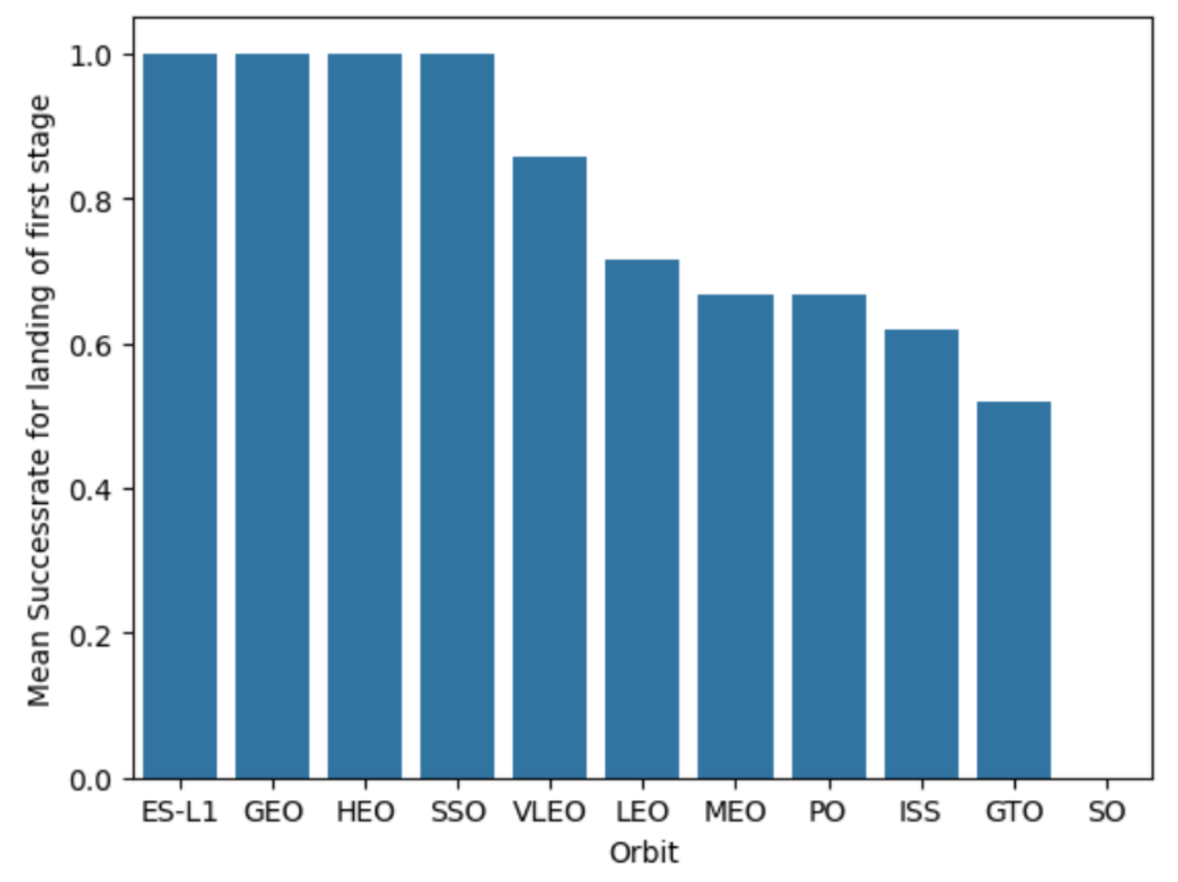
The bar chart shows the success rate for each orbit type of the launches

It shows that fore 4 orbit types

- ES-L1
- GEO
- HEO
- SSO

the success rate is 100%

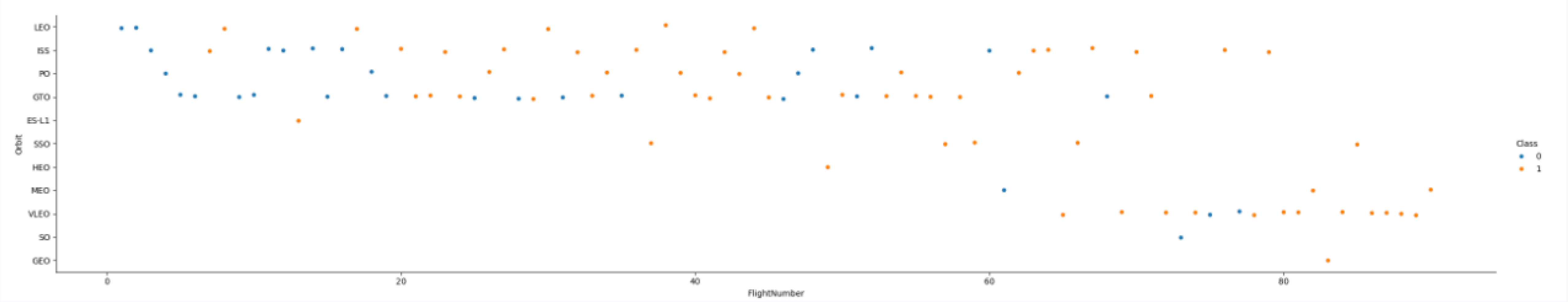
And for almost all orbit type the success rate is higher than 50 %



Flight Number vs. Orbit Type

The below scatterplot shows each launch based on its Flight number split into the different Orbit types. The launch is indicated with orange color if the launch outcome was successful and blue if it failed.

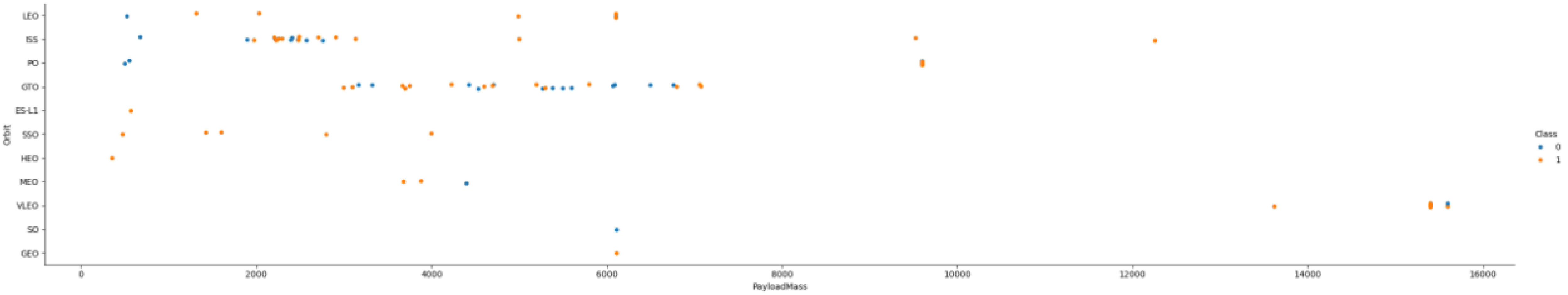
It shows that far most of the launches are for the Orbit types – LEO, ISS, PO and GTO. There is also a small visible tendency that the first launches with a given Orbit type has a higher risk of failing



Payload vs. Orbit Type

The below scatterplot shows each launch based on its Payload split into the different Orbit types. The launch is indicated with orange color if the launch outcome was successful and blue if it failed.

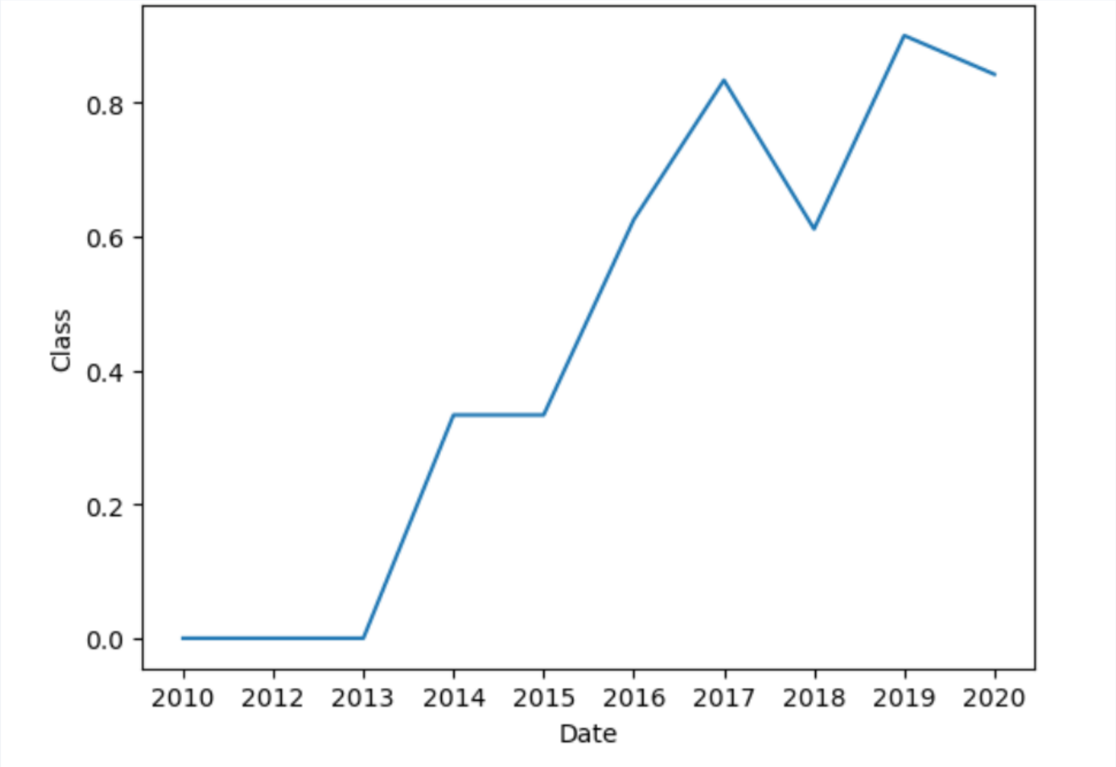
It shows that payload categories for the higher payloads are used for specific Orbits. It also shows a very high success rate for higher payloads for some Orbits



Launch Success Yearly Trend

The line chart shows the yearly average success rate for landing phase 1 of the rockets

It clearly indicates that the success rate is increasing almost every year from 2013 to 2020 and in 2019 and 2020 the success rate is above 80%



All Launch Site Names

Below is given the result of an SQL query finding all the unique launch sites by using the distinct(Launch_Site) function

Out[26]:

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

Launch Site Names Begin with 'KSC'

Below is given the first 5 rows including "KSC" in the launch site name by using the Launch_Site like "%KSC%" function

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Missio
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	

Total Payload Mass

Below is given the total payload carried by boosters from NASA in an SQL query by using the sum(PAYLOAD_MASS__KG_) function

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

Below is given the average payload mass carried by booster version F9 v1.1 from an SQL query using the avg(PAYLOAD_MASS__KG_) function

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Drone Ship Date

Below is given the date of the first successful landing outcome on drone ship from an SQL query using the min(Date) function and the where Landing_Outcome = "Success (drone ship)" clause

Out[41]:

min(Date)
2016-04-08

Successful Drone Ship Landing with Payload between 4000 and 6000

- Belov is given the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 from a query using the where clause to filter both on Landing_Outcome and Payload_Mass

Booster_Version
F9 FT B1032.1
F9 B4 B1040.1
F9 B4 B1043.1

Total Number of Successful and Failure Mission Outcomes

- By using the COUNT(date) function in an SQL query it was found that only 2 out of the 90 launches didn't list as Mission_outcome scucess

88 Successful

2 failed

Boosters Carried Maximum Payload

To the right is given the names of the booster which have carried the maximum payload mass from an SQL query using the where function with the subquery PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_)

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

2017 Launch Records

Below is given the records which will display the month, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017 from an SQL query using the where clause on a substring of the data and on landing outcome

substr(Date,6,2)	Landing_Outcome	Booster_Version	Launch_Site
02	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
05	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
06	Success (ground pad)	F9 FT B1035.1	KSC LC-39A
08	Success (ground pad)	F9 B4 B1039.1	KSC LC-39A
09	Success (ground pad)	F9 B4 B1040.1	KSC LC-39A
12	Success (ground pad)	F9 FT B1035.2	CCAFS SLC-40

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

To the right is given the rank of the count of landing outcomes between the date 2010-06-04 and 2017-03-20, in descending order. This is created using an SQL query and the GROUP BY Landing_Outcome ORDER BY count_of_landing_outcomes DESC function

Landing_Outcome	count_of_landing_outcomes
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

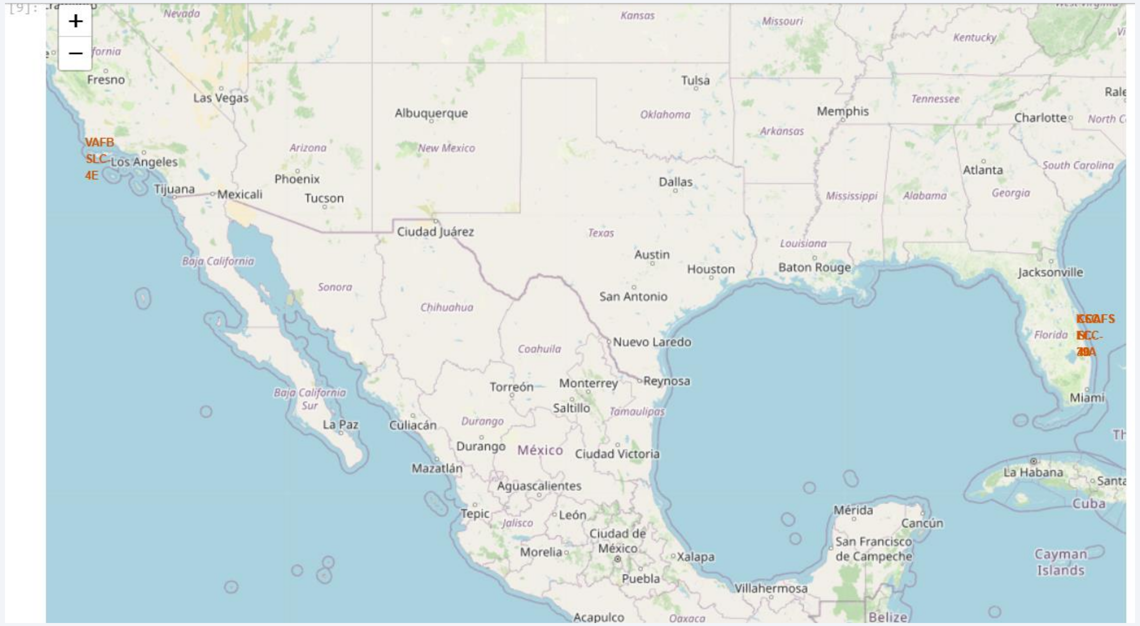
Section 3

Launch Sites Proximities Analysis

Folium Interactive Map With Launch Site

To the right is a screenshot of the interactive map with the locations of the launch sites

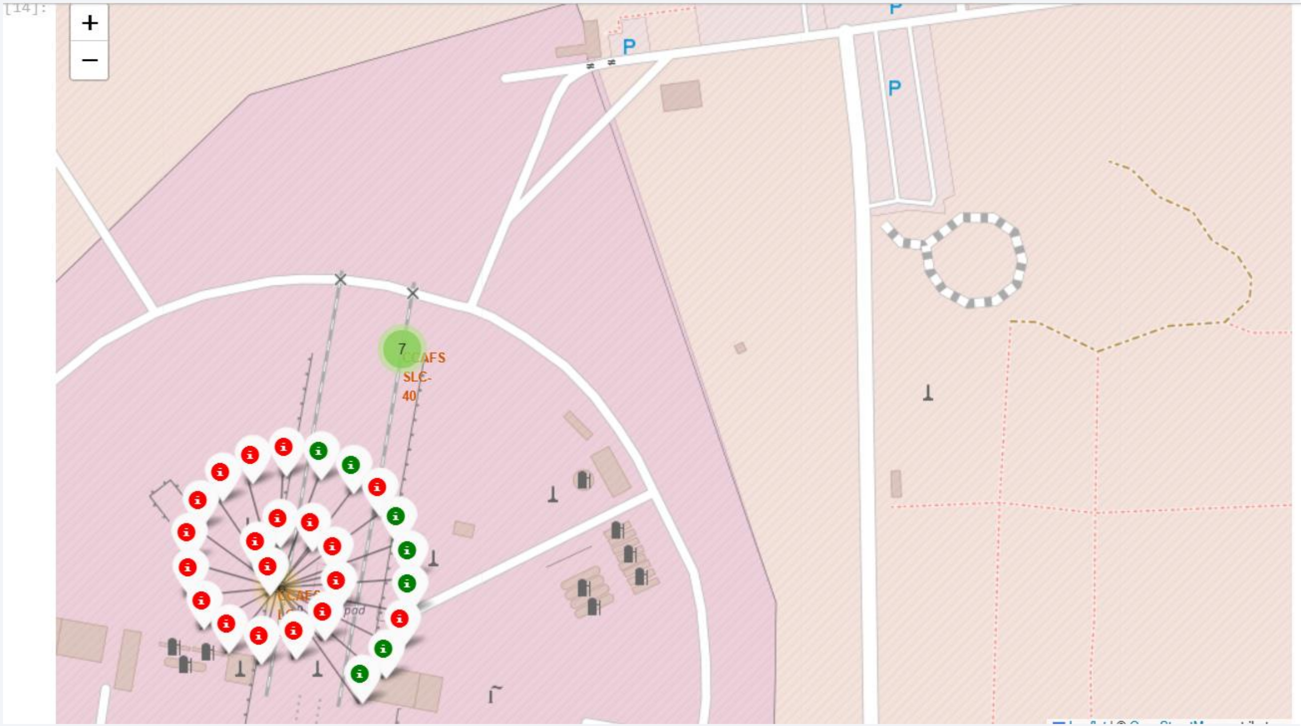
It is seen on the map that the launch sites are located in California and Floria and are all located close to the coast



Folium Interactive With Landing Outcome

To the right is given a screenshot of the interactive map which show the mission outcome for each launch site when clicking on it

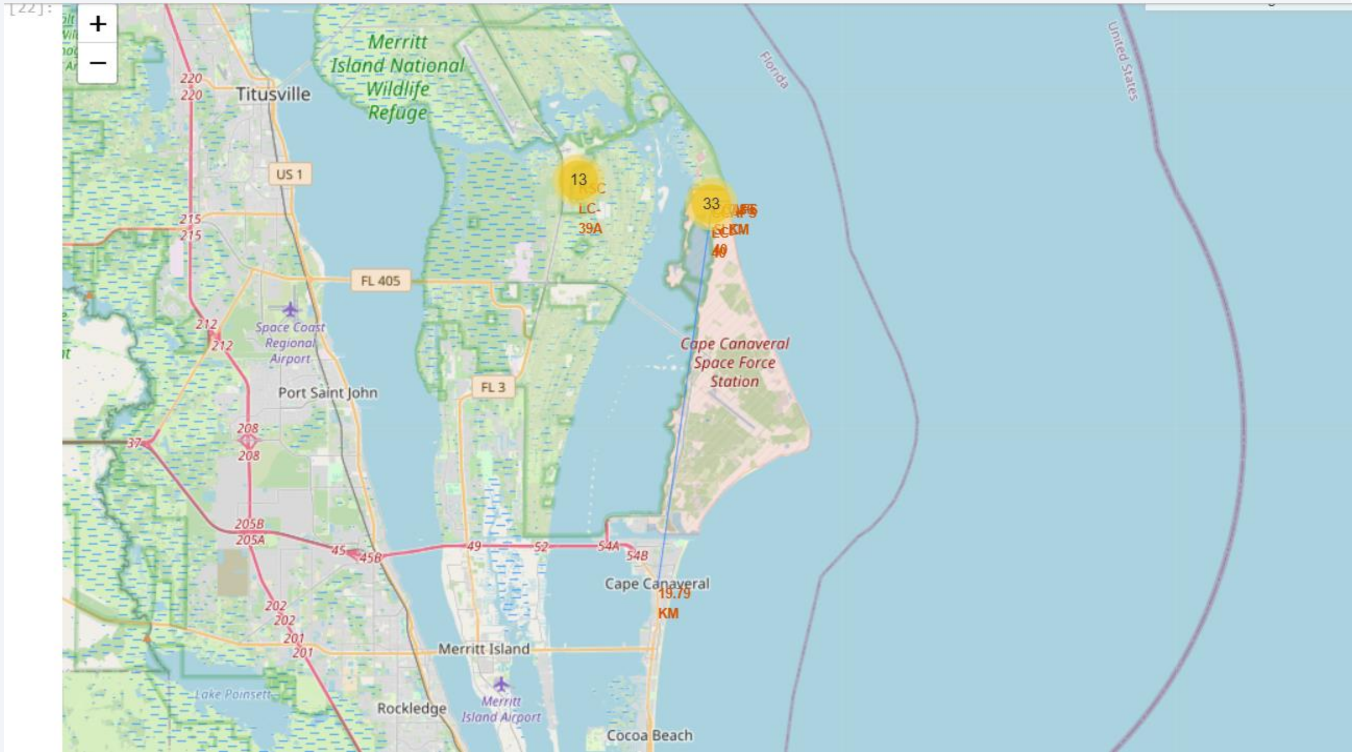
In the screenshot is given that 7 of the Landing outcomes was failed for the CCAFS-LC-40 launch site



Folium Interactive Map With Distances to Proximities

To the right is given a screenshot of an interactive map with indication of relevant proximities to a launch site

Most clearly visible is the line and the distance to the neares City from CCAFS-LC-40 the launch site 19.79 km



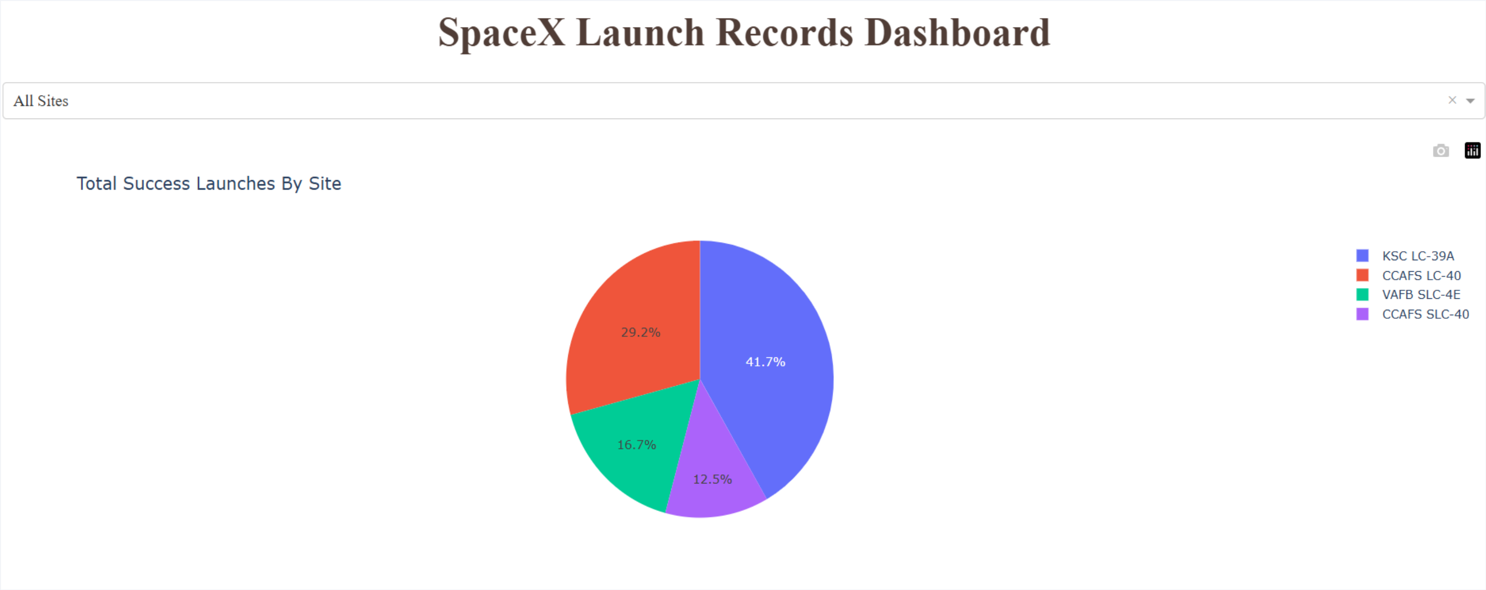
Section 4

Build a Dashboard with Plotly Dash

Dashboard - Pie Chart Success Rate - All Launch Sites

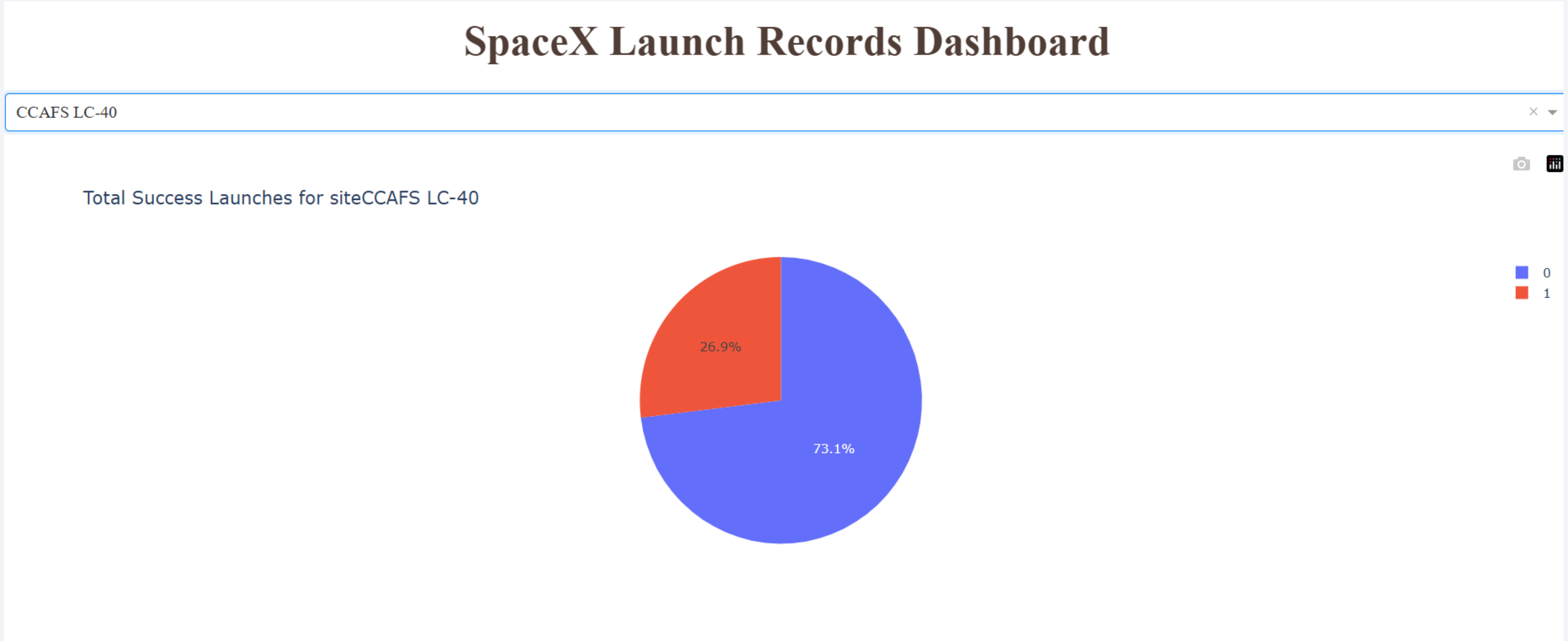
The Pie Chart shows the proportion of successful launches between all the launch sites because.

It can be seen that e.g. site CCAFS-LC-40 is contributing with 29,2% of successful outcomes



Dashboard - Pie Chart Success Rate - Specific Launch Sites

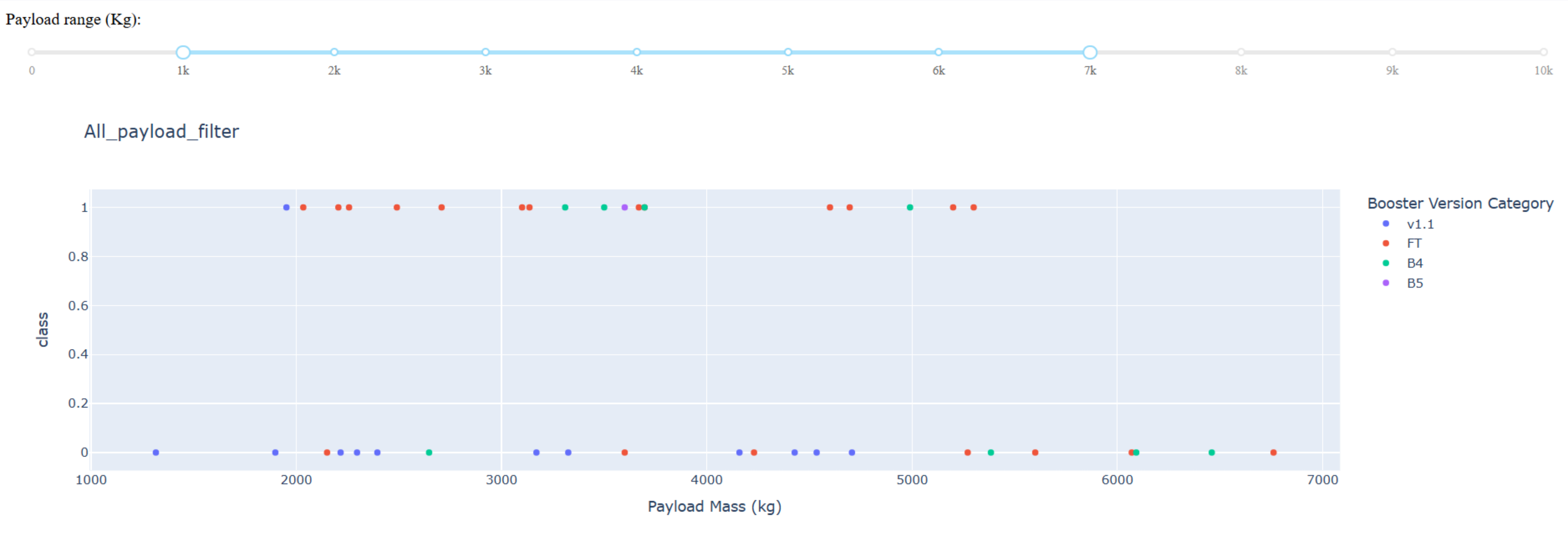
In the Pie Chart below on a specific launch site is shown (CCAFS-LC-40) and it can be seen that the success rate for this launch site is 26.9 %



Dashboard – Scatterplot With Range Slider on Payload

Below is shown a screenshot of a scatterplot of all launch sites where the Payolad range is selected using a slider on top of the screenshot

It can be seen that in the payload range 1000 – 7000 kg the booster version FT was contributing with the most successful landings



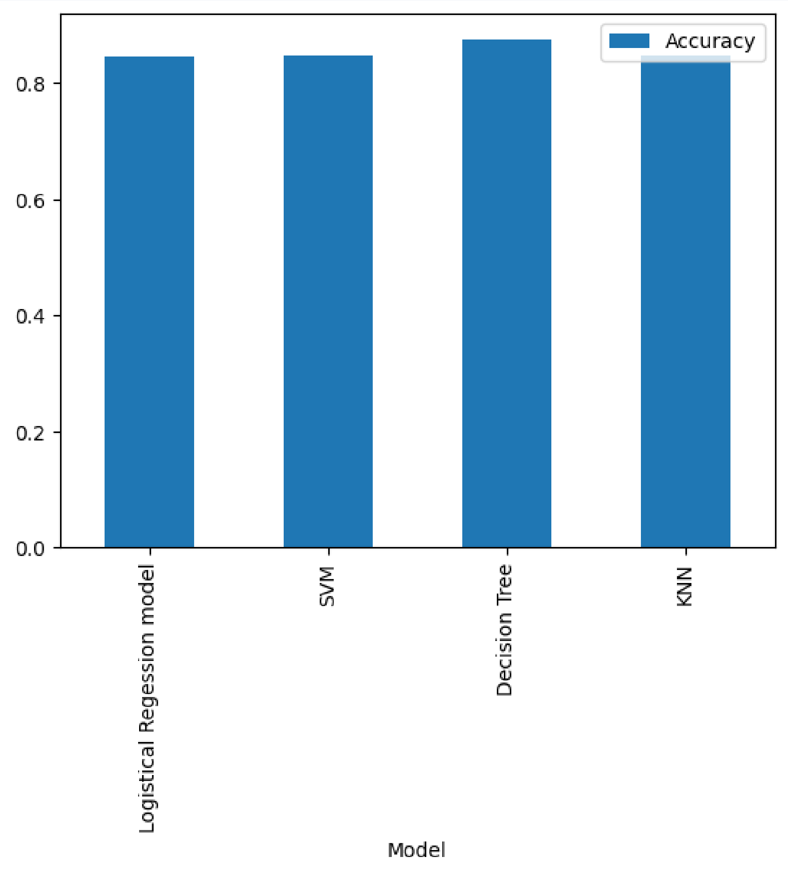
Section 5

Predictive Analysis (Classification)

Classification Accuracy

To the right is given a bar chart of the model accuracy of the 4 models used for comparison in the Prediction Notebook

As it can be seen the Decition Tree model has the highest accuracy with 0.875

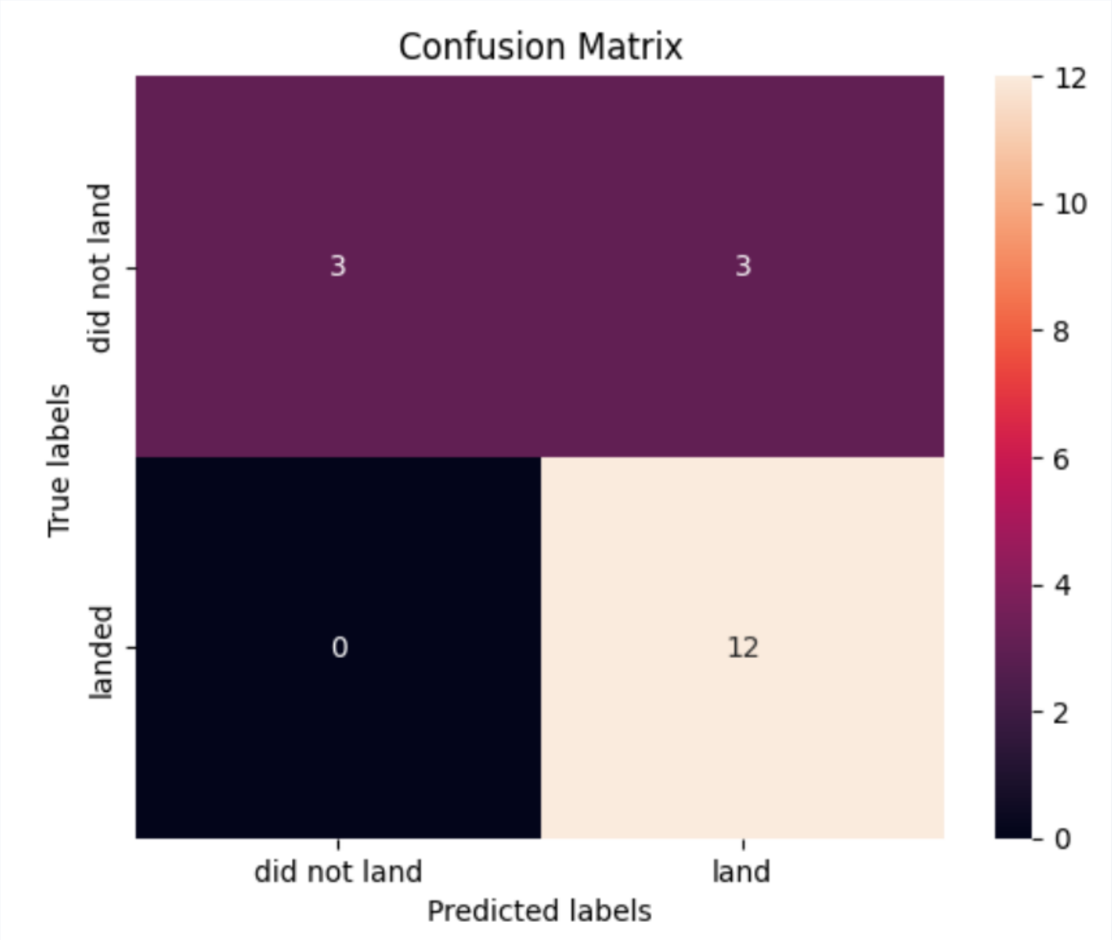


Confusion Matrix

To the right is given a confusion matrix showing how well the decition tree model performed when predicting the outcome of the test data set

As it is seen the model predicted correctly 12 outcome as landed and correctly on 3 outcomes where the phase 1 didn't land

However in 3 occations the model predicted that the phase 1 would land when it in reality did not land (False Positive)



Conclusions

- The 4 classification models where all relativly good in predicting the outcome of a phase 1 landing with an accuracy of 83% on the test data
- On the test data the models performed equally well, but meassured on the training data the Decision Tree model had the best accuracy
- It was necessary to use a Grid Serach Cross Validation on the models because the amount of data was a bit limited

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!