IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Karol Wilk-Juraszek
08.11.2025

Outline

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

Executive Summary

- Predictive models of various types (e.g. Logistic Regression, k-Nearest Neighbours) were used to try and find a way to predict a successful/unsuccessful landing based on past data
- While the prediction models are all imperfect, their accuracy stands at around 83%, which is not bad at all

Introduction

- Cosmic flights are enormously expensive and re-usable rockets provide an opportunity to combat this problem – but only if they do land successfully.
- In this report we provide methods used to find possible correlations and (hopefully) causations standing behind landing failures, so that we can find ways to minimize them. We also provide results of our investigation.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - We collected rocket launch data from SpaceX API, with usage of web scraping tools such as requests and BeautifulSoup libraries
- Perform data wrangling
  - We were interested only in Falcon 9 rocket launches, so we discarded all data related to other rockets. After that, any missing values were replaced with an average value of all non-corrupted rows.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection – SpaceX API

- Data was collected from SpaceX API via https://api.spacexdata.com/v4/launches/past.
- Thanks to requests library, we can decode the response content as a Json and turn it into a Pandas data frame.
- https://github.com/wilkjuraszekkarol/Final-presentation relevant notebook for this step is „(1) jupyter-labs-spacex-data-collection-api.ipynb”.

Data Collection - Scraping

- There isn't much to be talked about, first we have to take a list of Falcon 9 launches from Wikipedia, next use requests and BeautifulSoup libraries to extract an HTML table, next we extract all column and variable names by using find_all function. Finally, we can create a data frame by parsing the launch HTML tables.

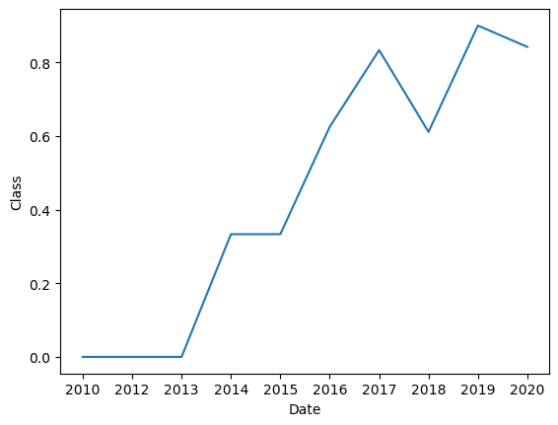
- https://github.com/wilkjuraszekkarol/Final-presentation You can find the relevant notebook here, under the name " jupyter-labs-webscraping.ipynb".

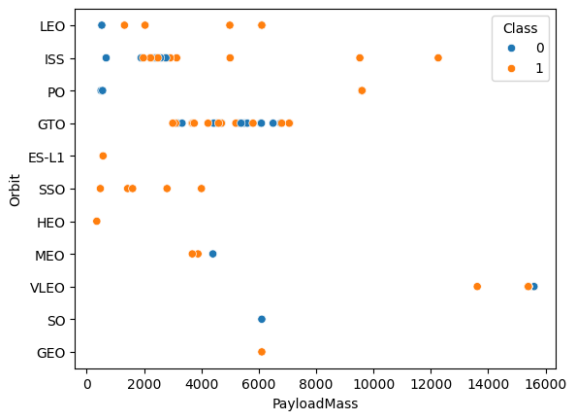
Data Wrangling

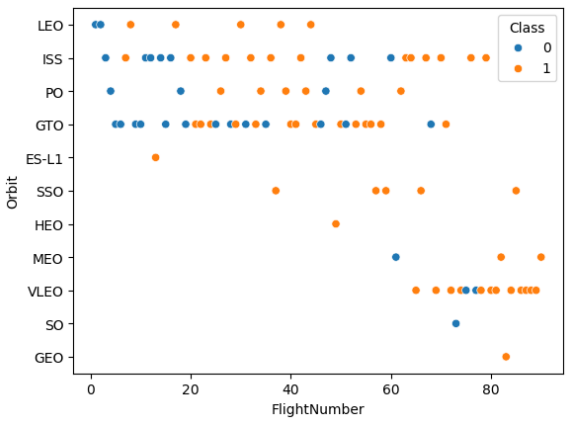
- Before any further analysis, a crucial thing to do is to clean the data. That's why we identified all empty rows – it turned out that LandingPad column had nearly 30% of all rows with NULL values.
- It was insightful to group launches by their target orbit, or by their landing method and its' outcomes.
- https://github.com/wilkjuraszekkarol/Final-presentation relevant notebook is under the name „(3) labs-jupyter-spacex-Data wrangling.ipynb".

EDA with Data Visualization

- We used scatter plots to wisualize relationships between mass of cargo taken by the rocket, launch site, and whether the landing was actually successful or not. In result, we found that in some cases practise does make perfect (i.e. after time landing success was getting closer to 100%), while in other there was still some work remaining.
- https://github.com/wilkjuraszekkarol/Final-presentation notebook numer 4







EDA with SQL

- One of the SQL queries was supposed to clean all the rows with NULL Date values
- Another delivered every unique launch site name
- We also got the total value of cargo mass delivered for NASA
- And an average value of cargo mass carried by each Falcon 9 flight
- At last, we got a total value of successful and unsuccessful landings
- https://github.com/wilkjuraszekkarol/Final-presentation notebook named „(5) jupyter-labs-eda-sql-coursera_sqllite.ipynb”

Build an Interactive Map with Folium

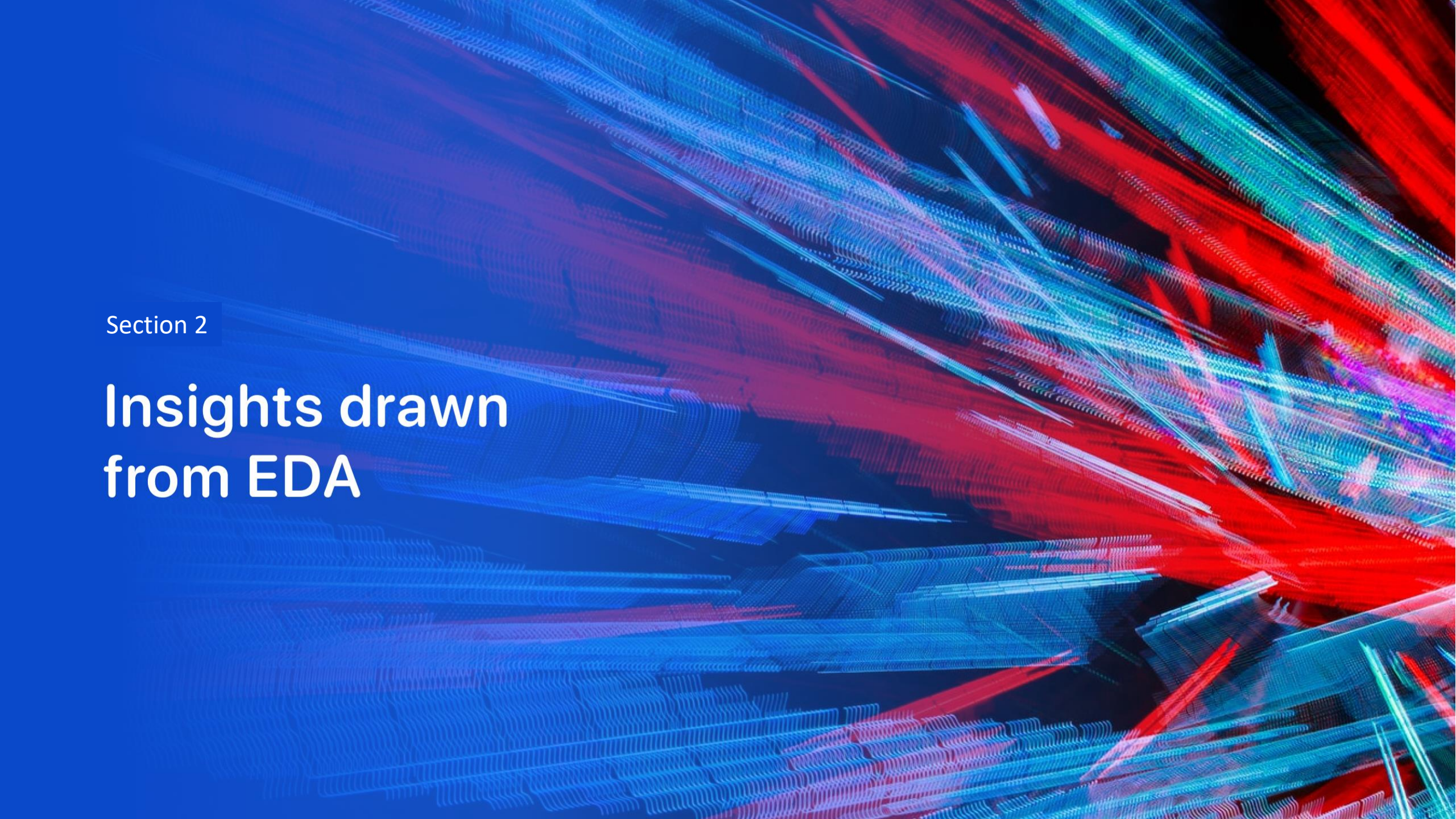
- Using Folium library we generated a US map and added markers depicting launch sites (for each launch), annotated whether it was a success or failure, and drew a line between launch site and nearest coastline.
- It was worth investigating whether one launchsite is safer than the other. In case of a positive answer to that question, it would be a massive acknowledgement.
- https://github.com/wilkjuraszekkarol/Final-presentation go to „(6) lab_jupyter_launch_site_location.ipynb" to find more info

Predictive Analysis (Classification)

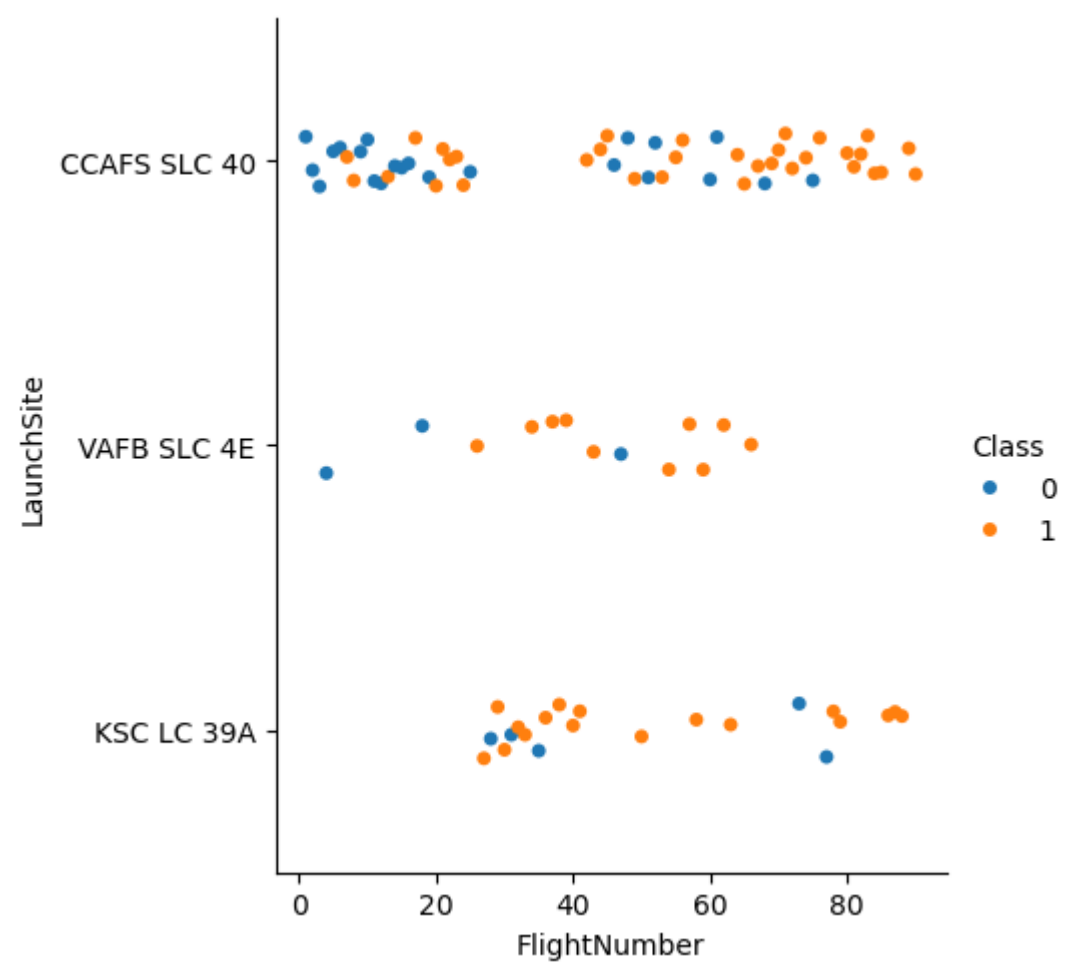
- Scikit-learn is a very powerful library that we put to use here.
- We took data and trained it using various method so we can create a model that will try to attempt whether an attempt will land successfully or not.
- Model types that we used are: Logistic Regression, Support Vector Machine, Decision Tree, and k-Nearest Neighbours
- https://github.com/wilkjuraszekkarol/Final-presentation Find the full results here in „(7) SpaceX_Machine Learning Prediction_Part_5.ipynb" notebook

Results

- The more rockets we launch, the more they actually land
- Different predictive models did not actually sprout different results, most likely due to scarce data

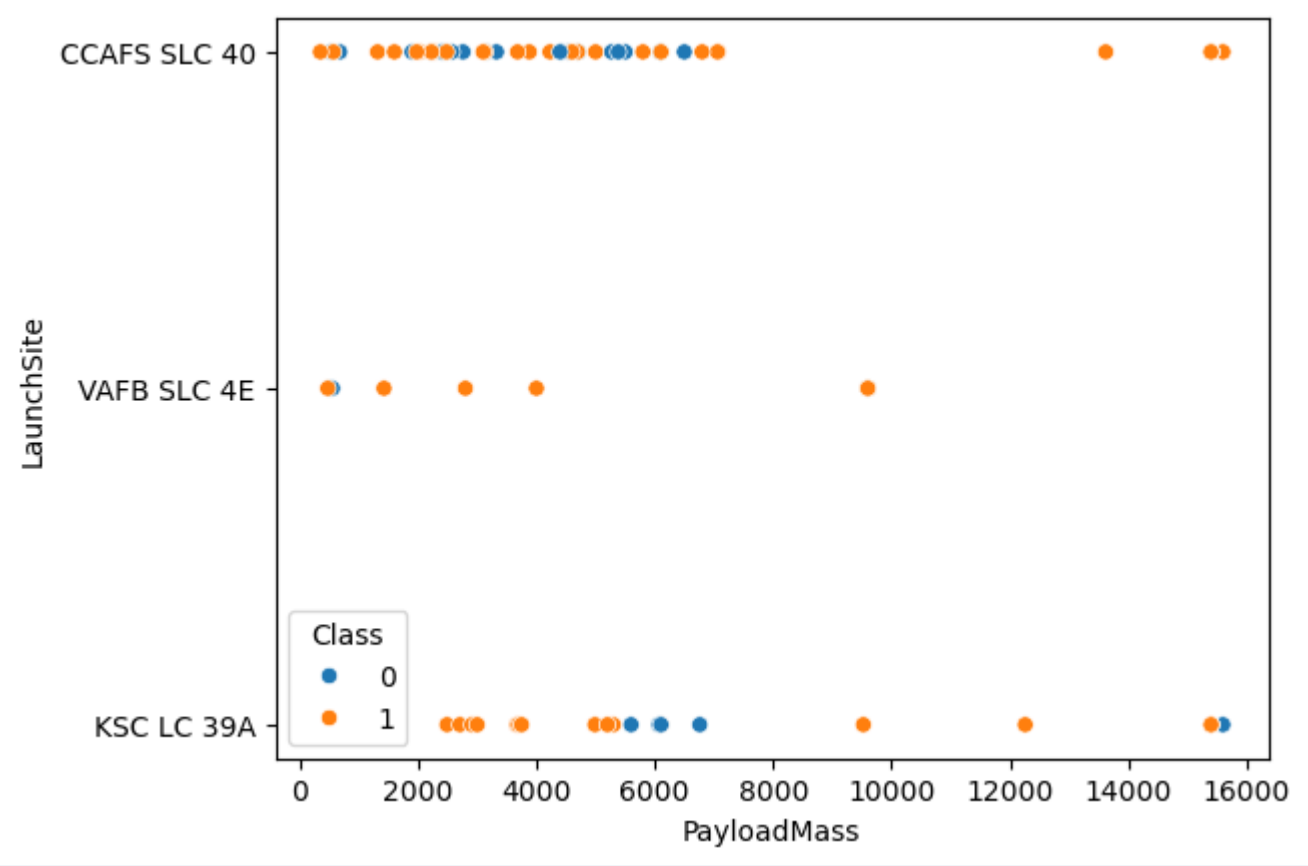

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



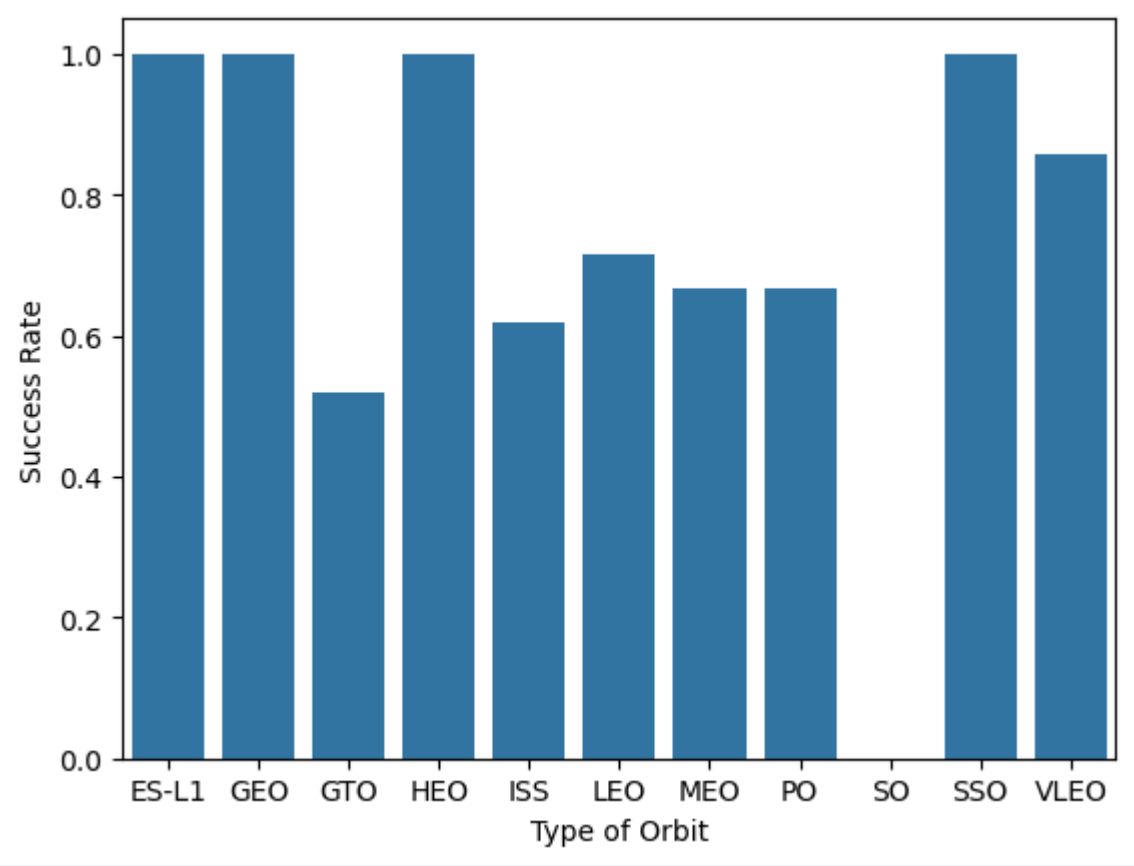
- „Class” value represents whether a landing was successful or not.
- 1 – Success, 0 – Failure
- CCAFS SLC 40 – Cape Canaveral (Florida)
- VAFB SLC 4E – Vandenberg SpaveX Launch Site (California)
- KSC LC 39A – Kennedy Space Center (Florida)

Payload vs. Launch Site



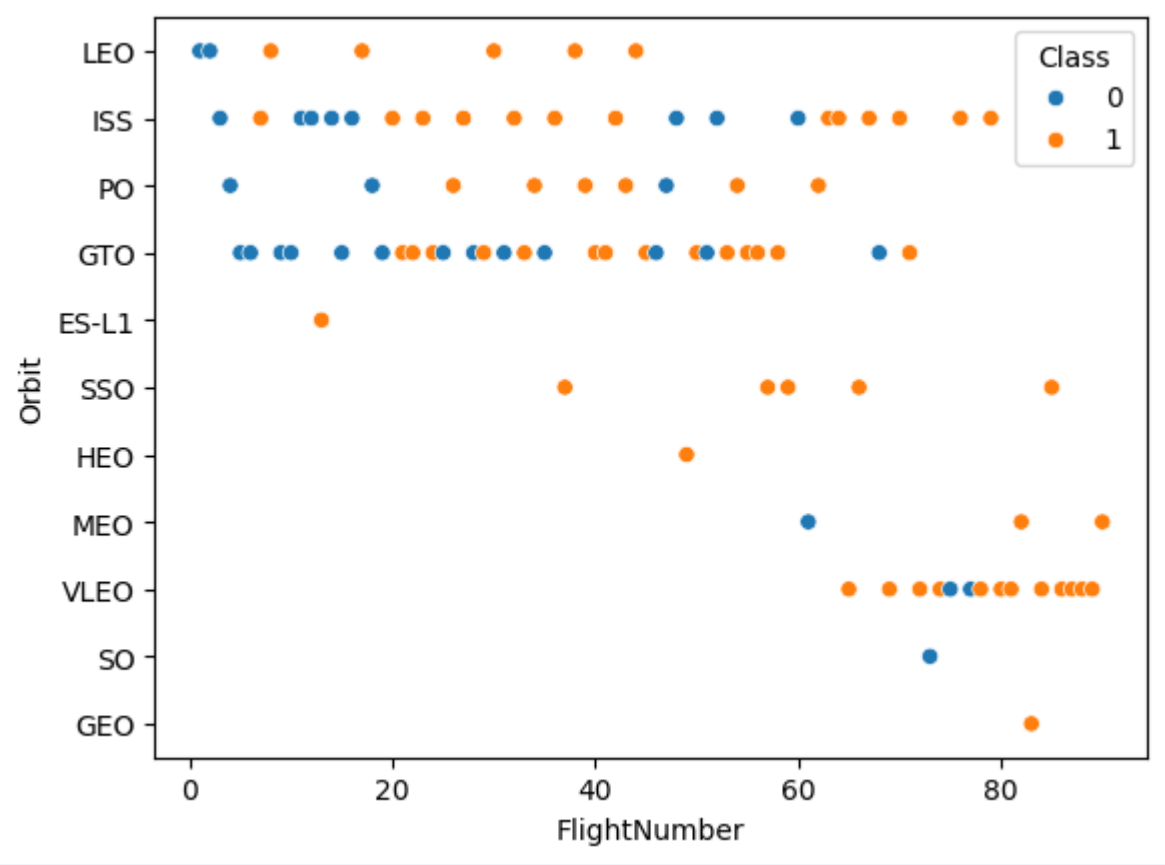
Payload mass is represented in kilograms. Everything else is the same as on the previous slide.

Success Rate vs. Orbit Type



- Success rate: 1 = 100%, 0.4 = 40%, etc.
- Every bar represents a different orbit. SO is a sun-synchronous orbit, which seems to always result in a failure – definitely worth investigating.

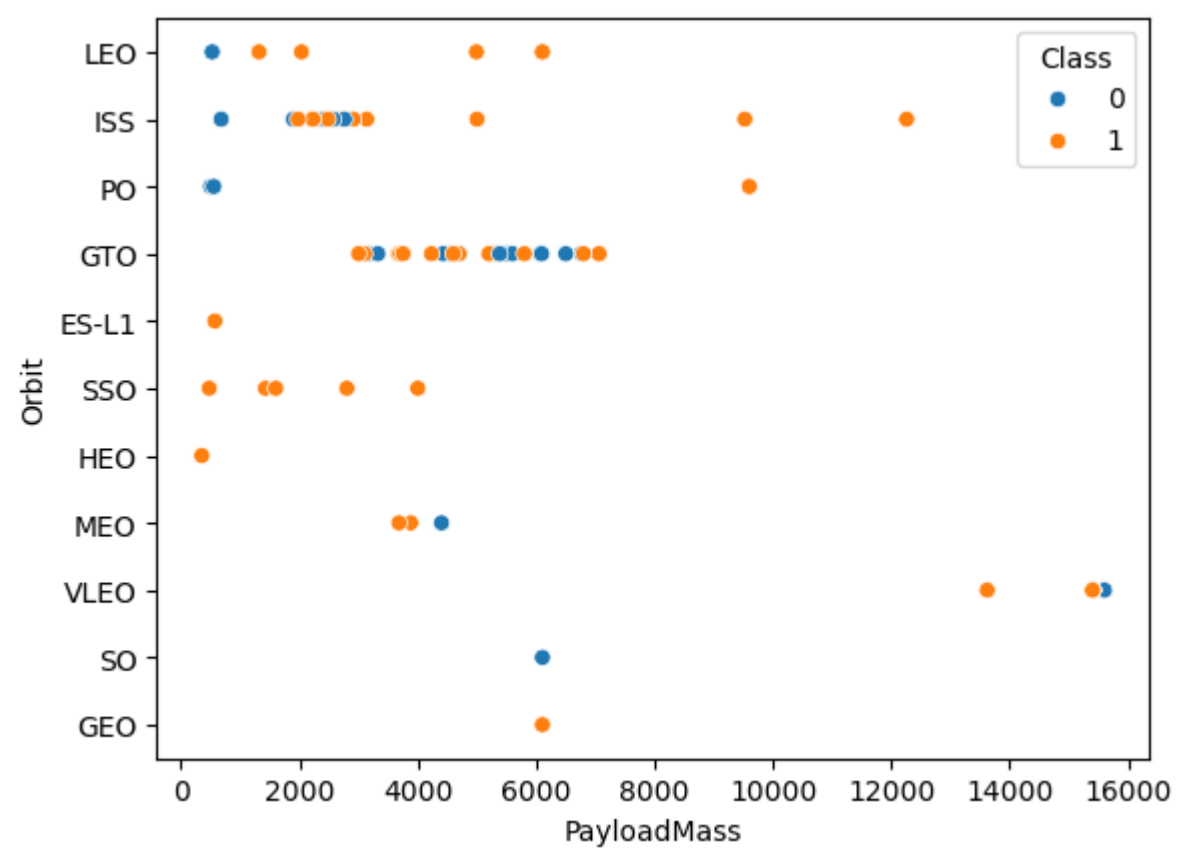
Flight Number vs. Orbit Type



Seems that there was only one attempt of launching a rocket onto an SO orbit, so it isn't surprising to have a 0% success rate. Same with ES-L1, HEO, and GEO orbits.

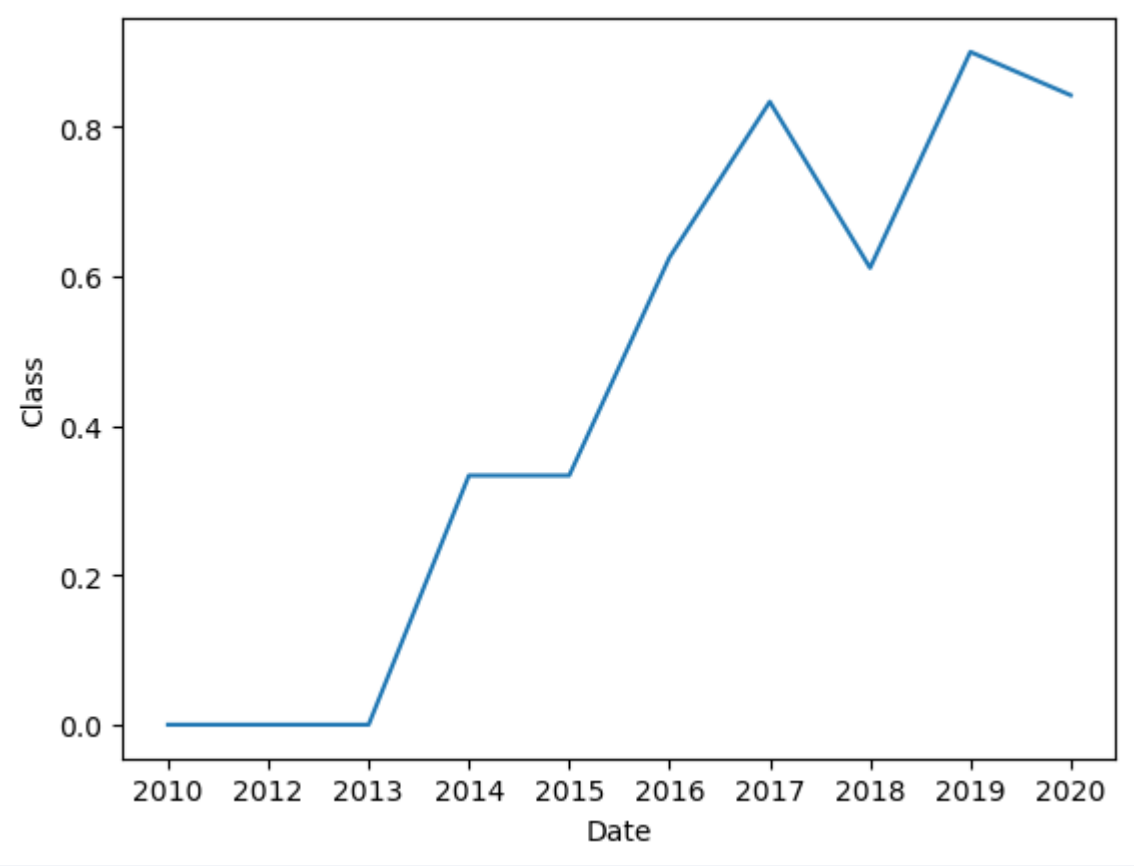
We can clearly see that while initially there were a lot of problems, as time progresses these have been largely resolved.

Payload vs. Orbit Type



There are visible patterns at ISS and GTO orbit – these are the International Space Station's orbit and Geostationary Transfer Orbit.

Launch Success Yearly Trend



As previously mentioned, it's clear that as time progresses, the success rate went from 0% to getting closer to 100% over just a few years.

All Launch Site Names

- CCAFS LC-40 (Cape Canaveral)
- VAFB SLC-4E
- KSC LC-39A
- CCAFS SLC-40 (Cape Canaveral, launch pad designated for SpaceX)
- The query was %sql SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE;
- It's self-explanatory, it just asks for all unique values inside the Launch_Site column

Launch Site Names Begin with 'CCA'

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- %sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE "CCA%" LIMIT 5;
- Similar to the previous query, except now we are only searching for results that Begin with „CCA” – hence the „LIKE „CCA%”” part
- „LIMIT 5” gets us only the first five results of this query, ordered by Date, not random by any means

Total Payload Mass

- Total payload carried by boosters from NASA is 107010 kg
- This result was achieved by a %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer LIKE "%NASA%"; query
- It takes a sum of all elements in a Payload_Mass_Kg column, where the customer is NASA

Average Payload Mass by F9 v1.1

- It's 2534.67 kg
- %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version LIKE "%F9 v1.1%";
- It's a very similar query to the one previously mentioned, except this time we take an average value instead of a sum of all of them

First Successful Ground Landing Date

- It was on 4th of June, 2010
- %sql SELECT MIN(Date) FROM SPACEXTABLE WHERE Mission_Outcome = "Success„;
- It takes the earliest date of a successful landing. Pretty self-explanatory.

Successful Drone Ship Landing with Payload between 4000 and 6000

- Booster names:

- %sql SELECT DISTINCT(Booster_Version) FROM (SELECT * FROM SPACEXTABLE WHERE Mission_Outcome = "Success") WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
- The query take all unique values of booster names which have successfully landed at least once, where the Payload mass was between 4000 and 6000 kg (and landed at least once with such payload)

Booster_Version
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1014
F9 v1.1 B1016
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1030
F9 FT B1021.2
F9 FT B1032.1
F9 B4 B1040.1
F9 FT B1031.2
F9 FT B1032.2
F9 B4 B1040.2
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5 B1046.3
F9 B5 B1048.3
F9 B5 B1051.2
F9 B5B1060.1
F9 B5 B1058.2
F9 B5B1062.1

Total Number of Successful and Failure Mission Outcomes

- 101 successes, and also 101 failures
- %sql SELECT COUNT(Mission_Outcome = "Success"),COUNT(Mission_Outcome - "Failure") FROM SPACEXTABLE;

Boosters Carried Maximum Payload

- List of the names of the booster which have carried the maximum payload mass:
- %sql SELECT DISTINCT(Booster_Version) FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
- It takes unique booster names which have taken the maximum allowed payload mass at least once.

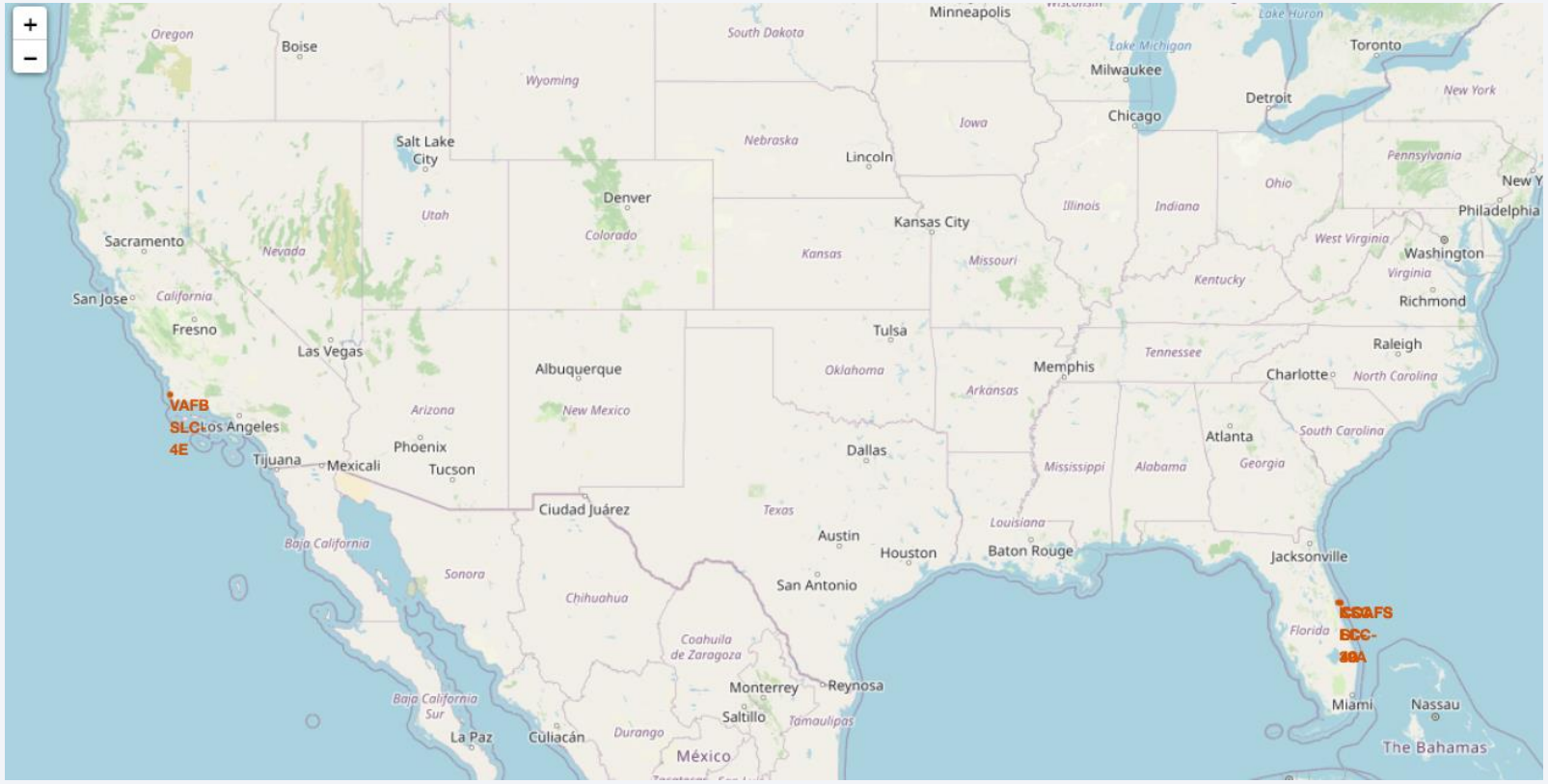
Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

Section 3

Launch Sites Proximities Analysis

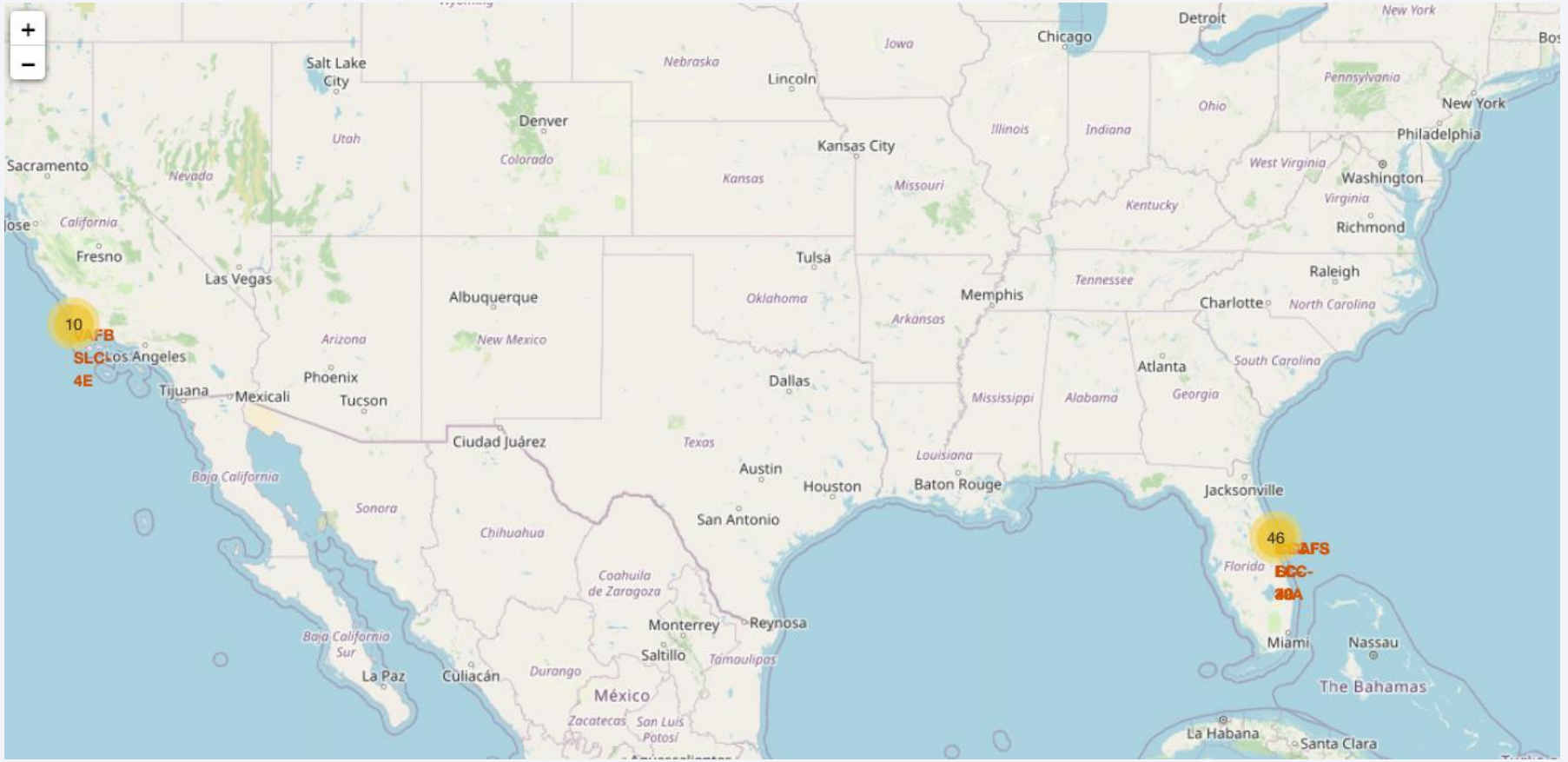
Folium Map of all Launch Sites

This screenshot shows all physical locations of launch sites included in the database.



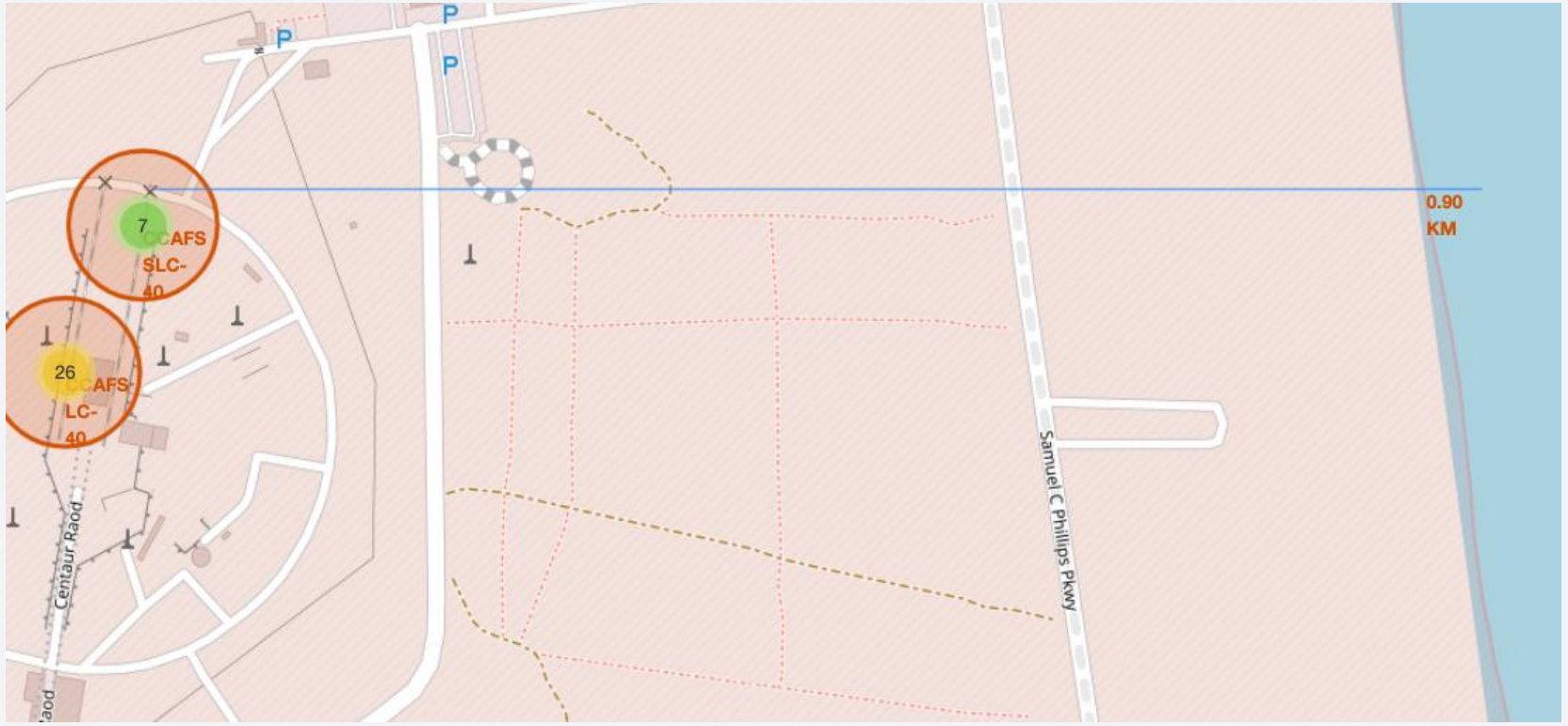
Folium Map of every single launch

This map here shows location of every single launch. After zooming in, it's annotated whether it was a success or not.



Folium Map showing distance of each launch from the nearest coastline

Pretty self-explanatory. You can find an example just below.

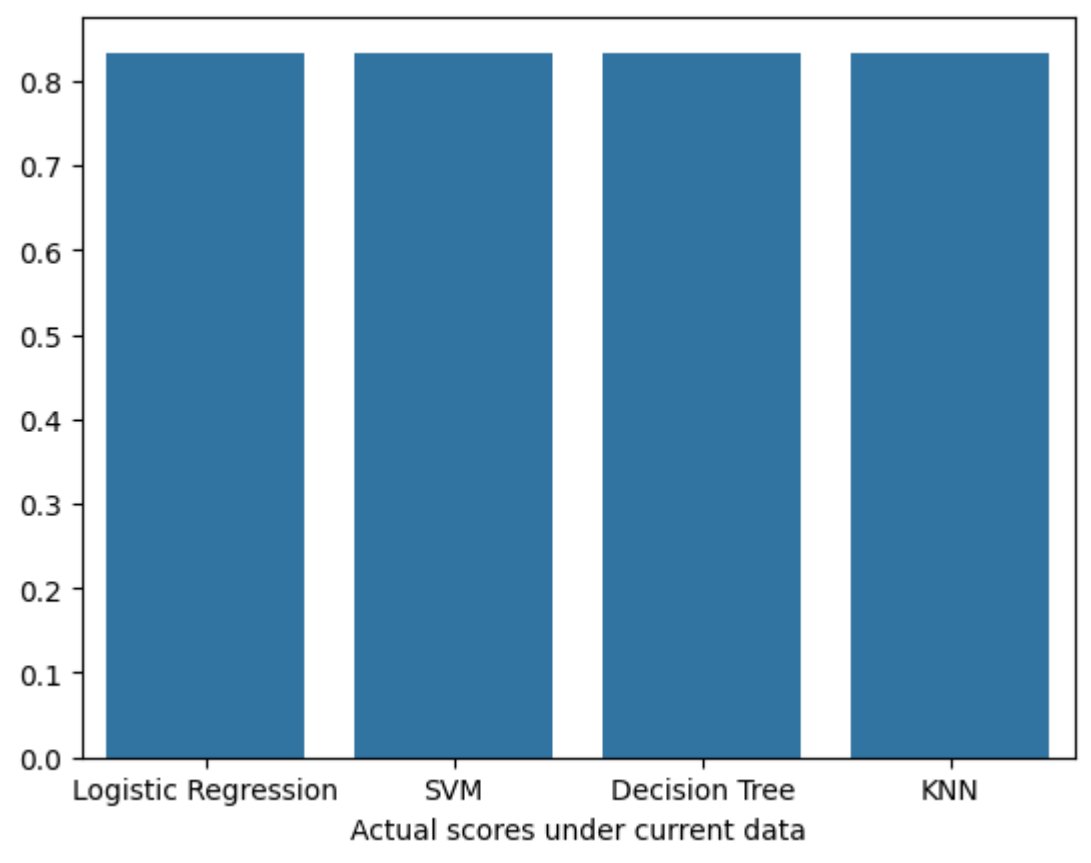


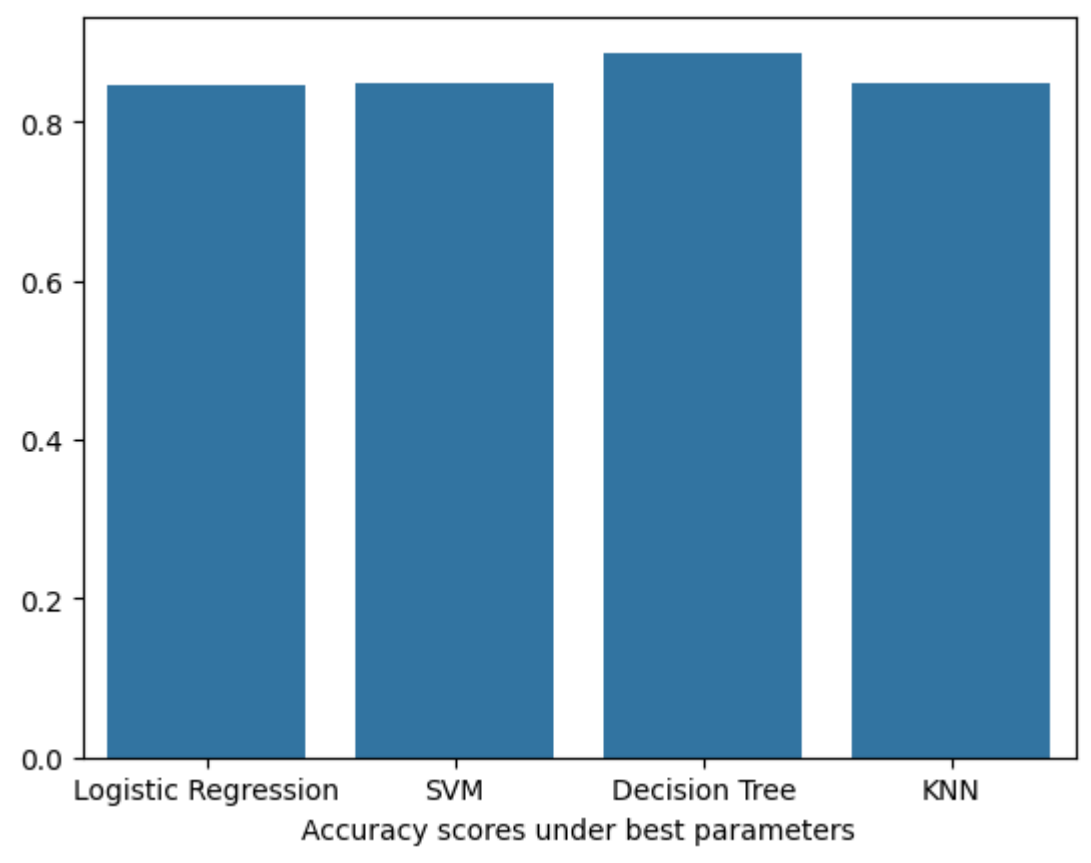
Section 5

Predictive Analysis (Classification)

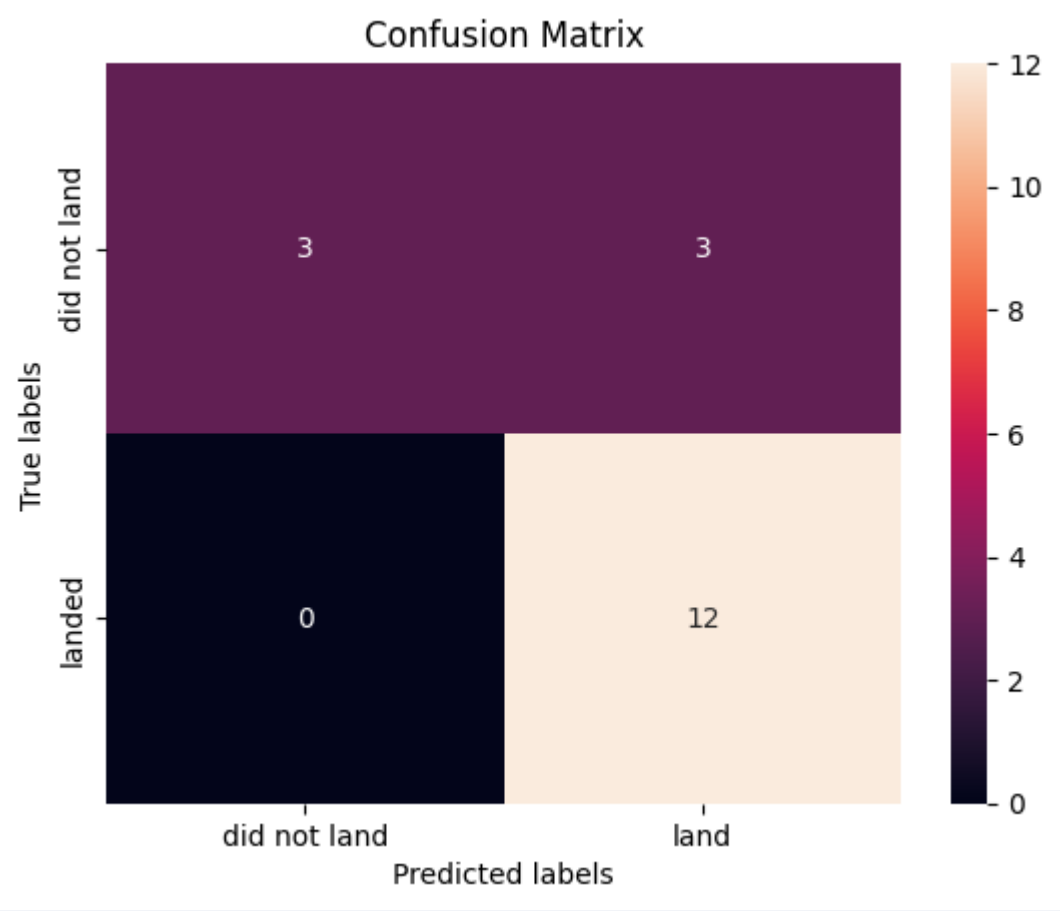
Classification Accuracy

Decision Tree has the highest accuracy score under the best parameters, but the data is scarce enough that it did not actually make any difference here.





Confusion Matrix



- This is confusion matrix of the best model in theory, in this case the Decision Tree. However, it looks the same for every other model (likely reason = data scarcity).

Conclusions

- Practise really does make perfect
- Data scarcity makes building models very difficult – a sample size = 18 is simply not large enough
- There does not seem to be a correlation between launch site and success/failure
- It's likely that more patterns will emerge when more data becomes available

Thank you!