IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

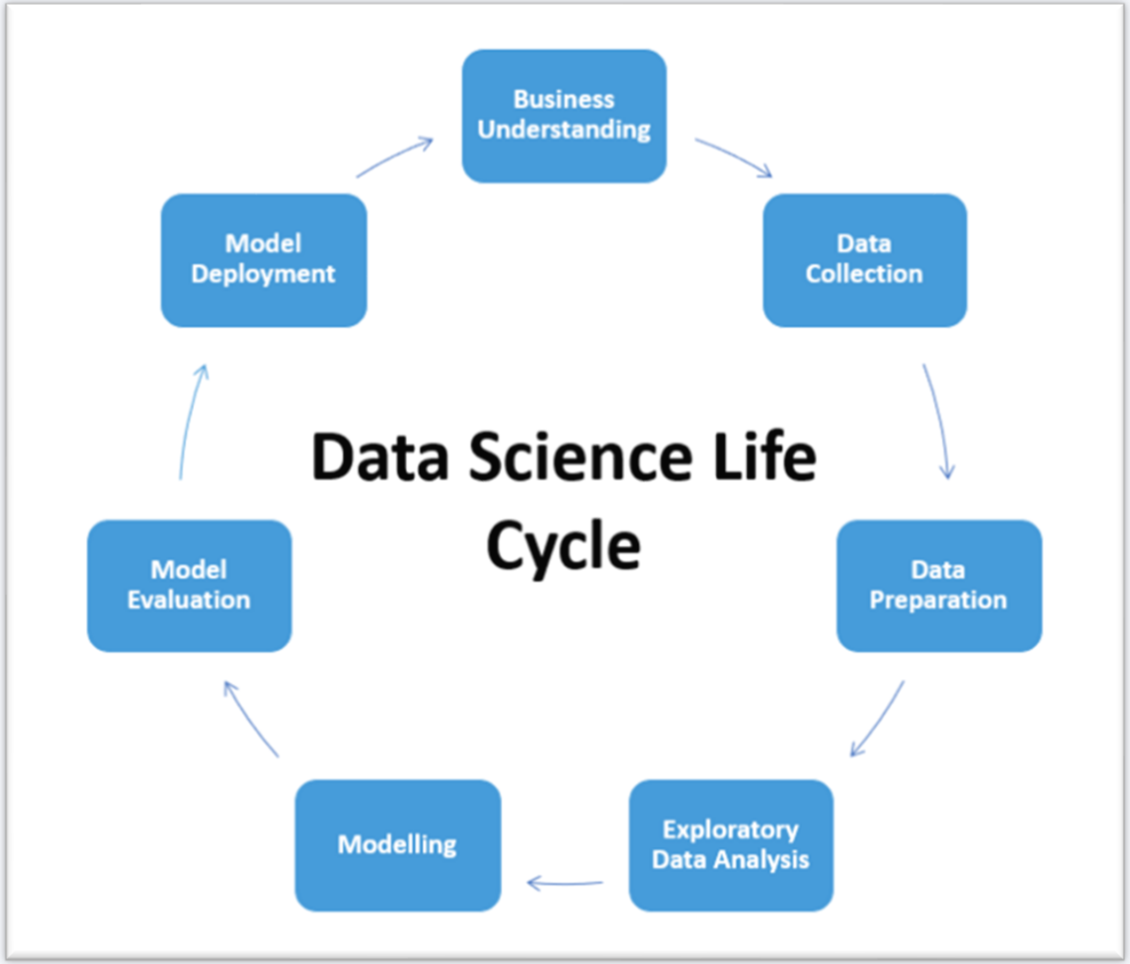
François-Paul Briand
April 20, 2023

Outline

• Executive Summary	p. 3
• Introduction	p.4
• Methodology	section 1, pp. 6 - 16
• Results	
○ EDA with Visualization	section 2A, pp. 19 - 24
○ EDA with SQL	section 2B, pp. 26 - 35
○ Proximities Analysis	section 3, pp. 37 - 40
○ Dashboard	section 4, pp. 42- 46
○ Predictive Analysis	section 5, pp. 48 - 49
• Conclusions	p. 50
• Appendices	pp. 51 - 57

Executive Summary

- This study was done with a methodology consistent with the steps of a data science study as outlined in the diagram.
- Summary of all results:
  - Data was collected and properly cleaned.
  - A set of quantitative and visuals tools was developed to study and refined the data set.
  - A Decision Tree model was trained to predict landing outcomes of future launches.
- All tools and model can be used again on any iteration of the data set.



Introduction

- Our new company wants to compete in the new commercial space market, in order to establish a winning strategy, the first step is to study the competition and analyze their successes and failures.
- SpaceX being one of the leading and most successful competitor on the market we chose to look at them first.

- 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. If we can determine if the first stage will land and therefore can be reused, we can determine the cost of a launch and compete effectively.

Section 1

Methodology

Methodology

Executive Summary

This page lists the steps followed in our study; steps further detailed in the rest of the section.

- Data collection:

  Data was collected from SpaceX API and Wikipedia web scraping

- Perform data wrangling:

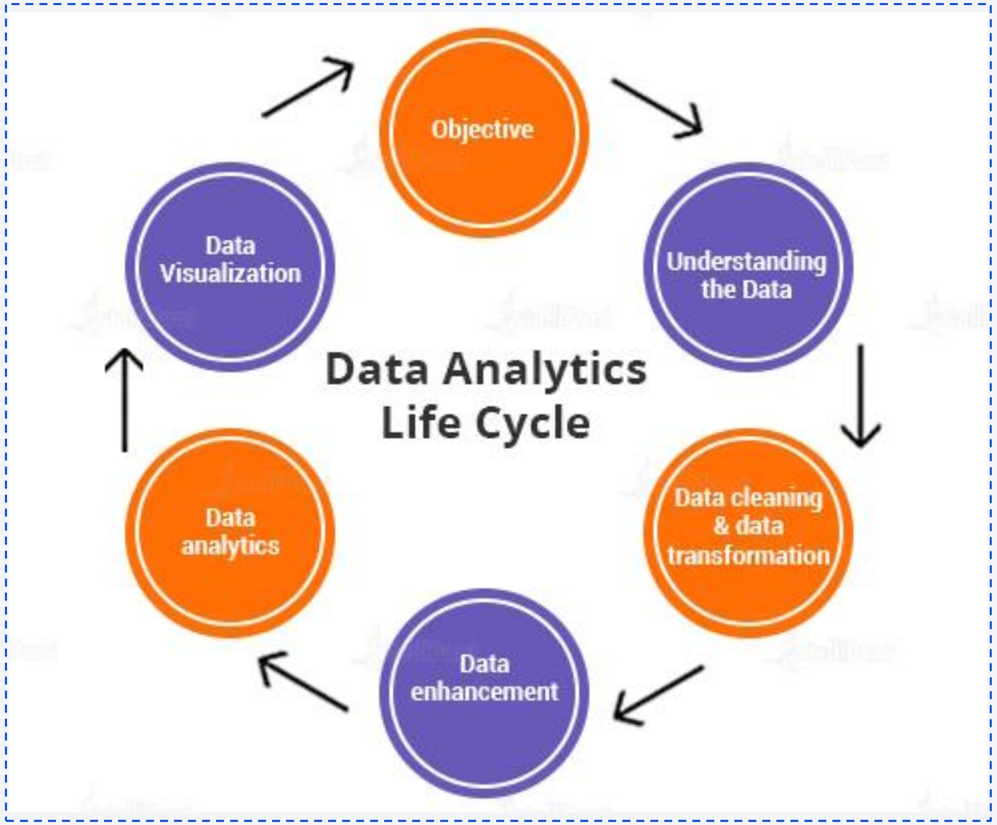
  Python language and Python libraries were used to process data, create tools and models

- 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 classification models were built, tuned, and evaluated

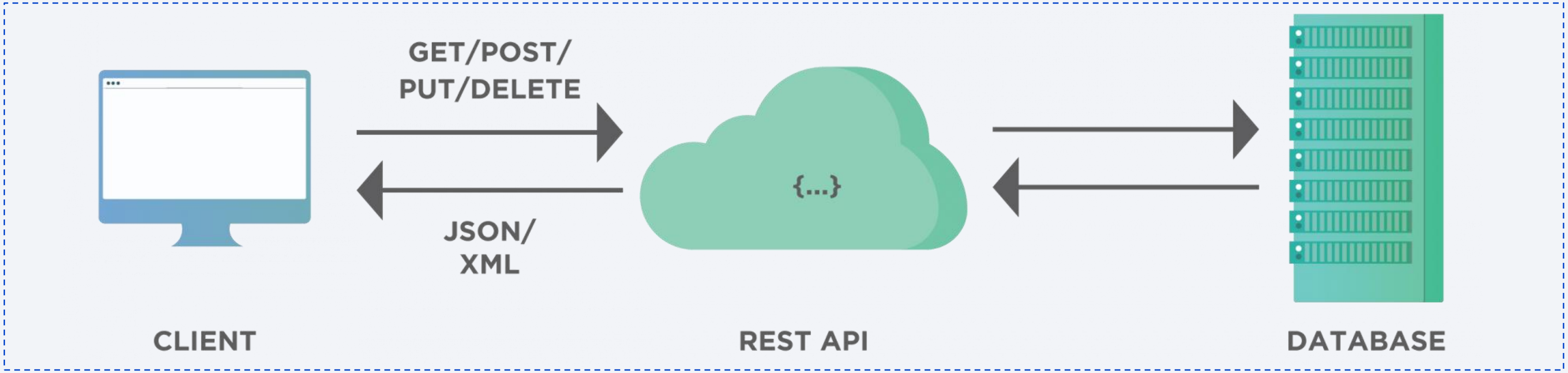
Data Collection (1) - Process

- Data collection is the process of gathering, measuring, and analyzing accurate data from a variety of relevant sources to find answers to research problems, answer questions, evaluate outcomes, and forecast trends and probabilities. This is illustrated on our diagram.
- The first step is data gathering, there are many different ways of gathering data but for this study we will collect data from two reliable sources on the internet:
  - SpaceX itself using their REST API
  - HTML pages from Wikimedia by web scraping



Data Collection (2) – SpaceX API

GitHub URL: [Data Collection with SpaceX API]()

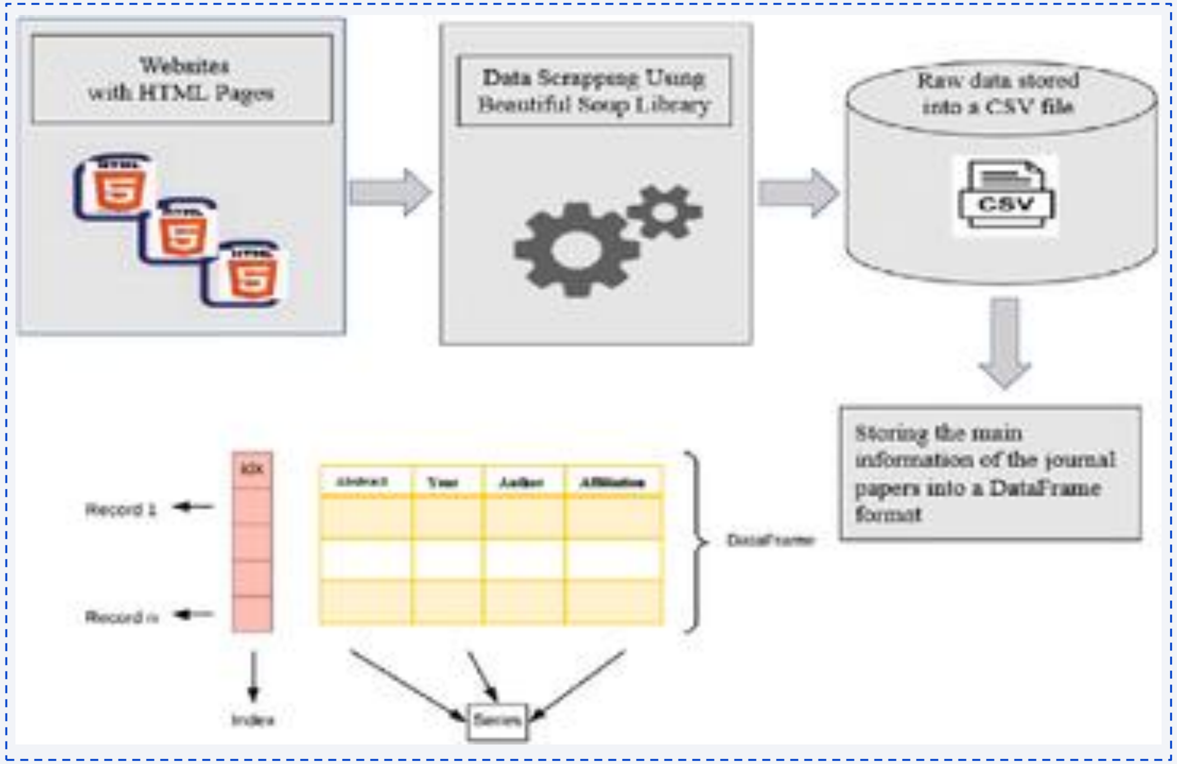


- The API is contacted with a GET request and response is obtained as JSON textual data. The data will be normalized and stored in a panda data frame.
- It is important to clean the data, if necessary multiple calls will be made to different APIs.

Data Collection (3) – Web Scraping

GitHub URL: Data Collection by Web Scraping

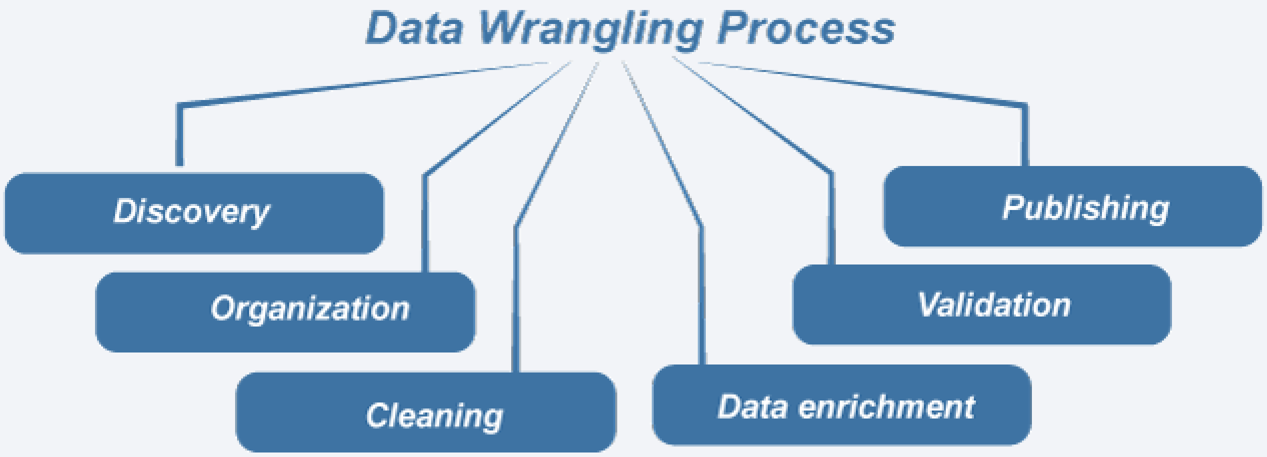
- Web scraping is the process of using automated processes to extract content and data from a website. Web scraping extracts underlying HTML code and, with it, data stored in a database.
  We will use the Python tool “Beautiful Soup”.
- Scraping the Wikipedia site:
  List of Falcon 9 and Falcon Heavy launches
- We then remove the HTML, parse the data and store it into a CSV file.


Websites with HTML Pages
Data Scrapping Using Beautiful Soup Library
Raw data stored into a CSV file
CSV
Storing the main information of the journal papers into a DataFrame format
Record 1
Record n
idx
Abstract
Year
Author
Affiliation
DataFrame
Index
Series

Data Wrangling

GitHub URL: Data Wrangling with Pandas and Numpy

- Data wrangling is the process of transforming raw data into a more processed shape by reorganizing, cleansing, and enriching it. Data wrangling entails processing data in various formats and analyses and combining them with another data set to produce meaningful insights.
- Some of the more important steps are:
  - Combining data sources for analysis.
  - Filling or removing data gaps.
  - Deleting unnecessary or irrelevant project data.
  - Identifying data outliers and explaining or deleting them to allow analysis.
- Among the benefits the data may be exported to any visual analytics platform to sort, analyze, and summarize the data (*see next page*).



- In our study the following steps are applied:
  - Identify and calculate the percentage of the missing values in each attribute
  - Identify which columns are numerical and categorical
  - Create a set of outcomes
  - Create a classification variable

EDA with Data Visualization

GitHub URL: [Exploratory Data Analysis with Visualization]

- In this study we will use scatter plots, bar charts, and line charts.
- These will allow the following (visual) analysis:
  - Potential correlation among various variables (attributes / features)
  - Outcome classification for a given variable
  - Trend over time for a given variable
- This step will help in determining which variables (features) we want to retain for the data set used to train predictive models later on.

EDA with SQL

GitHub URL: Exploratory Data Analysis with Structured Query Language

- Using structured Query Language (SQL) queries are performed on the data.
- Using SELECT queries and subqueries data is analyzed for:
  - Launch sites characteristics
  - Payload mass
  - Booster versions
  - Landing outcomes
- Queries are refined using the WHERE, LIKE, GROUP BY, ORDER BY, AND and other key words.

Note: for actual query code see Appendices : SQL Queries

Build an Interactive Map with Folium

GitHub URL: Interactive Analysis with Folium
nbviewer: Interactive Analysis with Folium
Colaboratory: Interactive Analysis with Folium

- Folium maps allows us to relate our data sets with geospatial features of interest such as location data, proximities, etc.
- The interactive nature of the maps (zooming, clicking) allows for both global and detailed study. This may allow us to include geographical features as an omportant part of the modeling.
- For this study we have located the launch sites and added markers and labels to record results data visually (such as landing outcomes).
- We also have marked (lines, distances) proximities of interest to geographical locations: cities, coastline, highways, railways, etc.

Build a Dashboard with Plotly Dash

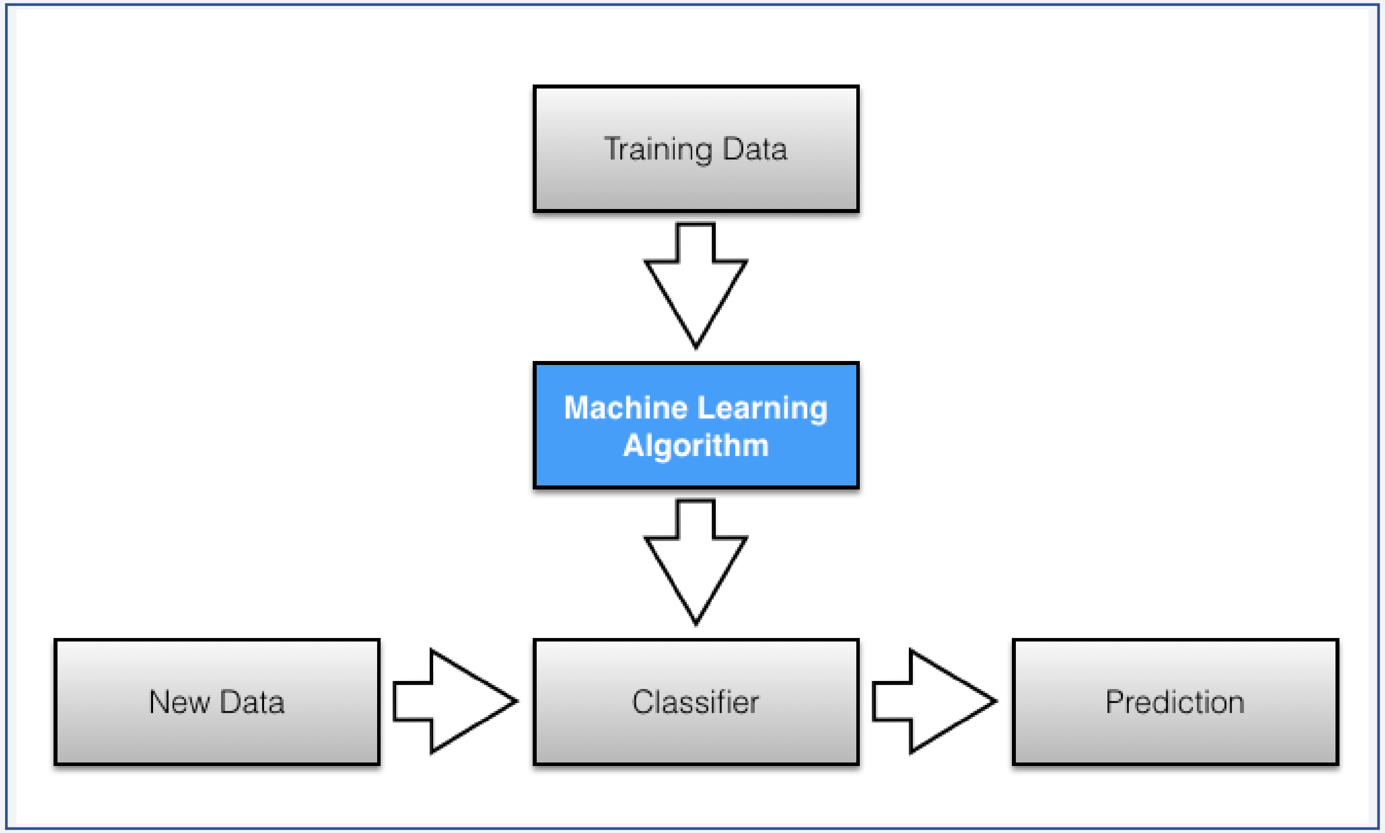
GitHub URL: [SpaceX Plotly Dashboard (Python Code)]
Colaboratory: [SpaceX Plotly Dashboard (Notebook)]

- In this study we have constructed a dashboard to be able to visually inspect important results of the data set as well as potential correlation between variables (features) of the data set which may orient our modeling choices.
- For this we have included both pie charts and scatter plots to present landing outcomes in relation to launch sites, payload mass and booster versions.
- We made it interactive with the addition of a dropdown selector for launch sites and a slider selector for payload mass.
- Change in selection is immediately reflected in the charts / plots by way of callback functions.

Predictive Analysis (Classification) – Part 1

GitHub URL: [Machine Learning Models for Classification]

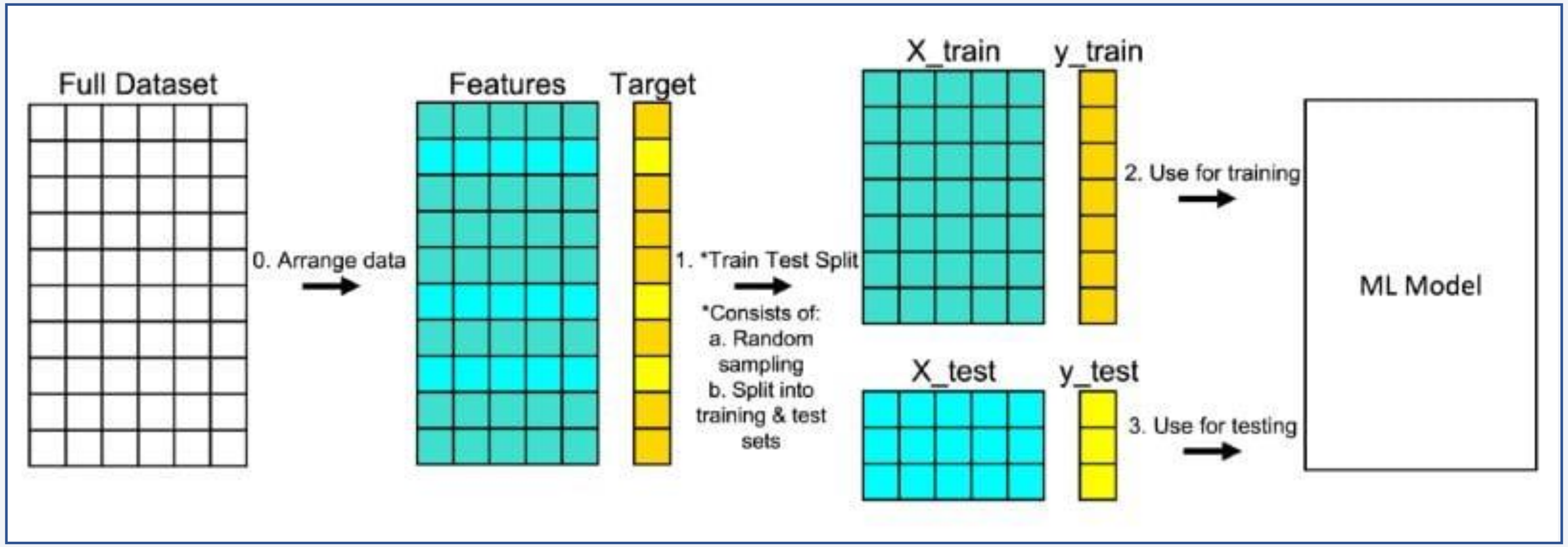
- For this study we want to predict a binary outcome (landing success / failure) based on historical data.
- Classification models fall under the branch of supervised machine learning models. They are very appropriate for binary outputs.
- 4 models will be trained, and their results compared: logistic regression, SVM (Support Vector Machine), Decision Tree and KNN (K-nearest Neighbors).
- The available data will be split in two sets, one for training the model and the other for testing (the so-called “New Data”) the results (predictions).



Predictive Analysis (Classification) – Part 2

GitHub URL: Machine Learning Models for Classification

- The first key step is to split the available data step in two parts: a set to train the models
- The second key step is to split the available data in two parts: a set to train the models and another to test each model’s predictions. The split is random but on a predefined percentage.



- Each model is then train with the training set and subsequently the test set is run through the model to get the predictions. These results are analyzed based on the known historical data to evaluate the model efficiency.
  The evaluation criteria are: the “accuracy” based on the train set, the “score” based on the test set results and a the “confusion matrix” which gives a visual evidence of the precision of the model by identifying things such as “false positives” and other conditions.

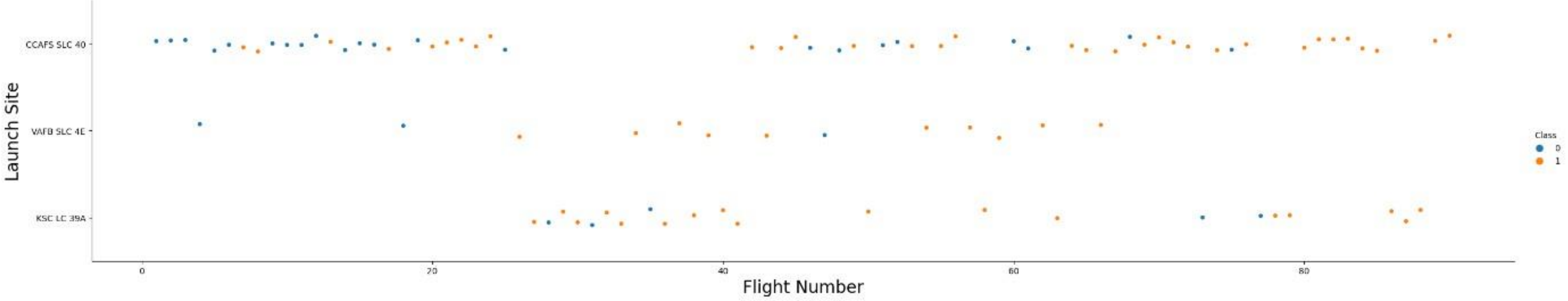
Results

• Exploratory data analysis results	section 2A, B
• Interactive analytics (screenshots)	section 3
• Interactive dashboard (screenshots)	section 4
• Predictive analysis results	section 5

Section 2A: Visualization

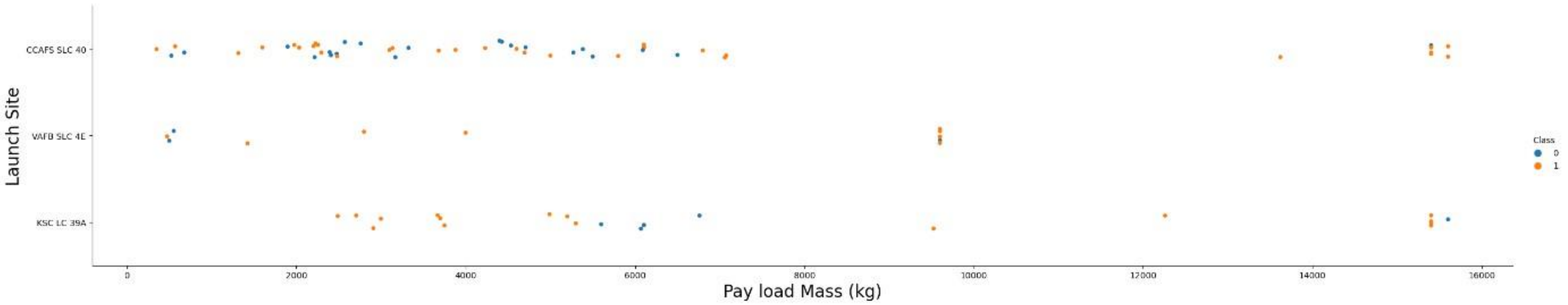
Insights drawn from EDA

Exploratory Data Analysis: Flight Number vs. Launch Site



- Class 0 is a failure; Class 1 is a success. Flight Number acts partially as a timeline as well as how many launches have been made.
- No clear correlation appears between the variables, even if as the number of launches increases the ratio of success seems to improves, yet failure may occur at any point.

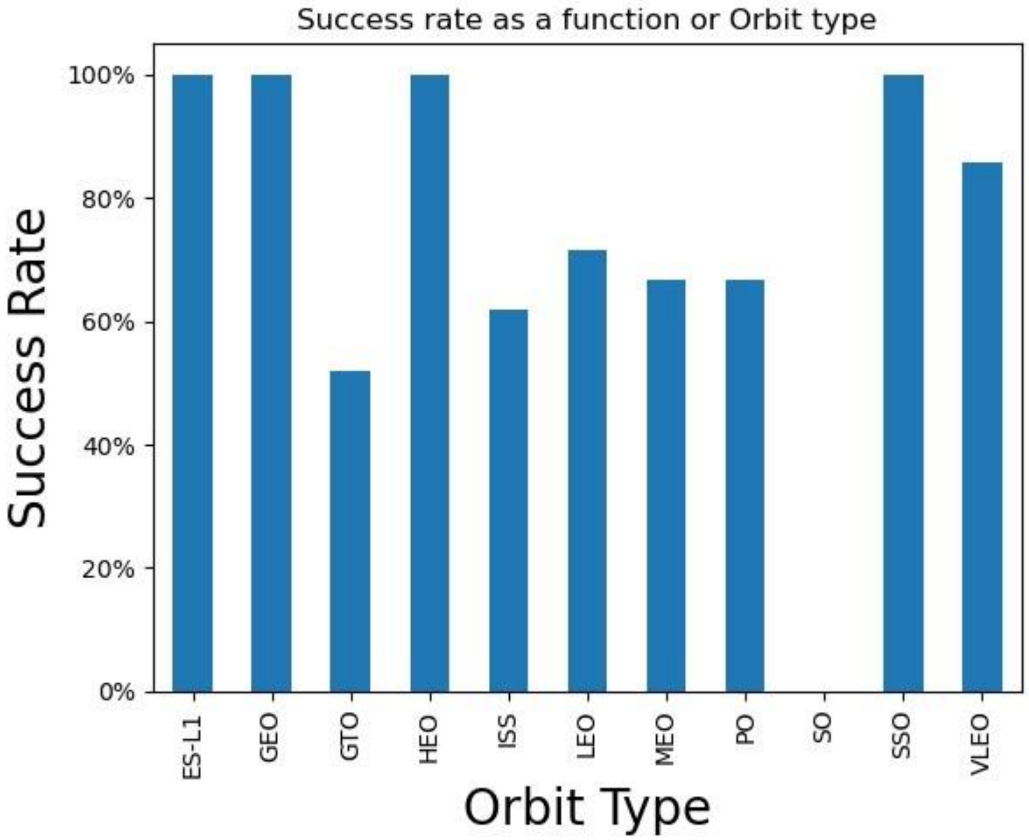
Exploratory Data Analysis: Payload Mass vs. Launch Site



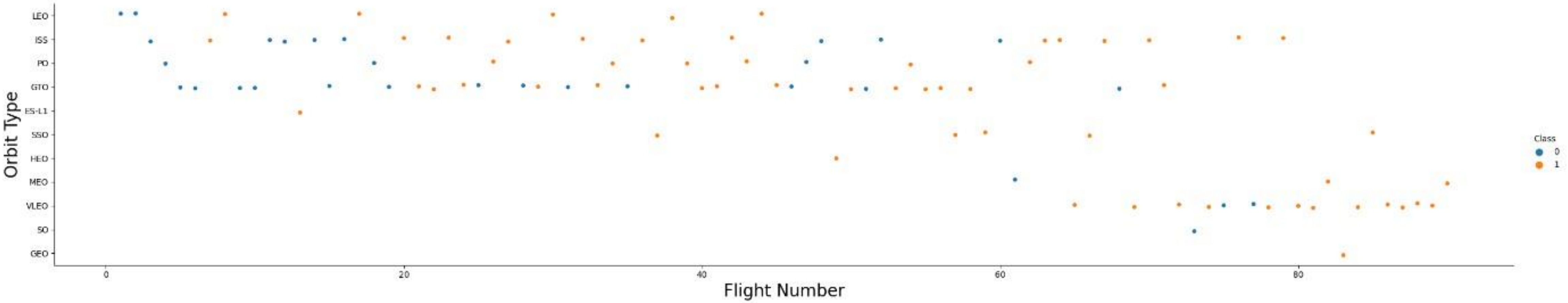
- Class 0 is a failure; Class 1 is a success. Most launches occur in the 1000 – 7000 kg range.
- No clear correlation appears between the variables, failure may occur at any payload range. It is to be noted that The VAFB site did not have launch with payload > 10 000 kg.

Exploratory Data Analysis: Success Rate vs. Orbit Type

- For Orbit Type details see Appendix (p 53).
- Note that ES-L1, GEO, HEO and SSO all have 100% success rate.
- Only GTO has a success rate nearing 50% only.

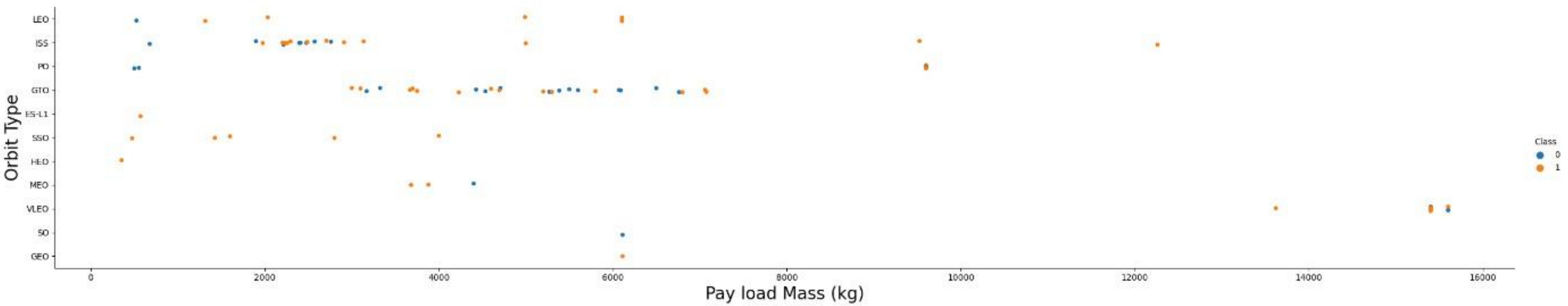


Exploratory Data Analysis: Flight Number vs. Orbit Type



- Class 0 is a failure; Class 1 is a success. Flight Number acts partially as a timeline as well as how many launches have been made.
- No clear correlation appears between the variables, failure may occur at anytime except for the LEO orbit which as been consistently successful as launches increase.

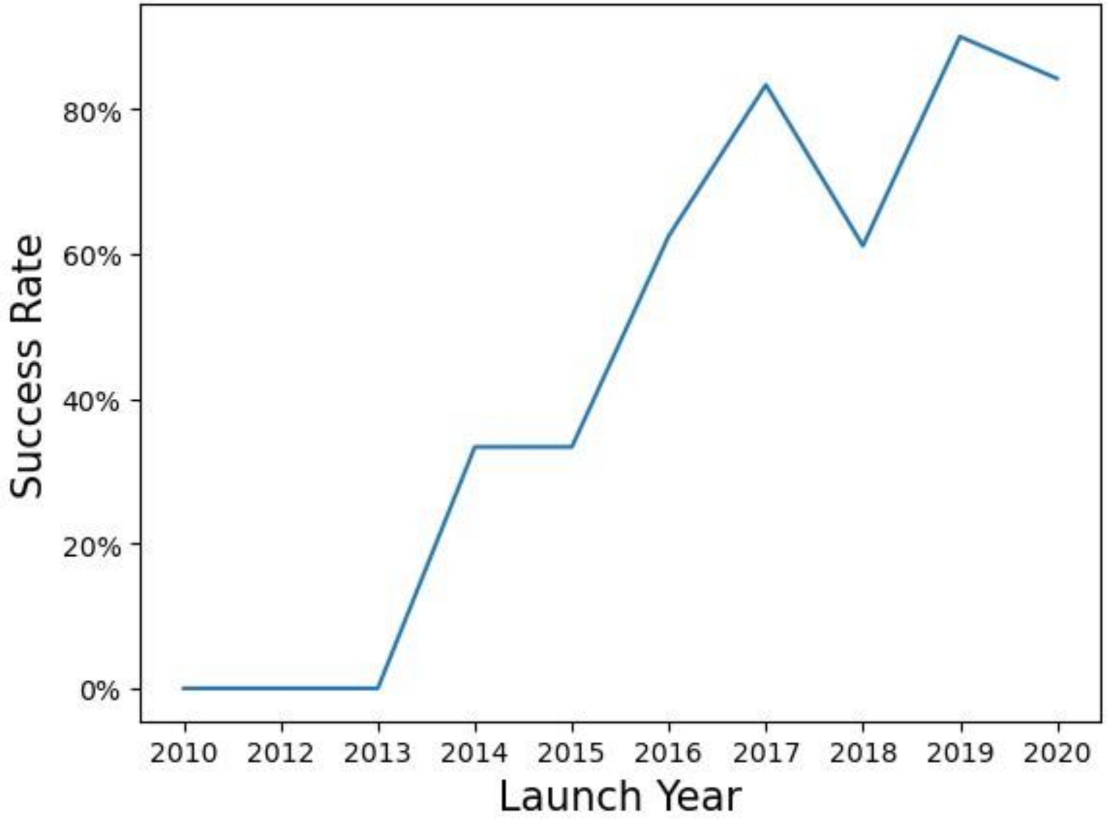
Exploratory Data Analysis: Payload Mass vs. Orbit Type



- Class 0 is a failure; Class 1 is a success. Most launches occur in the 1000 – 7000 kg range.
- Heavy payload > 10 000 kg tend to be successful independently of Orbit Type. In the most used range GTO and ISS show a random success pattern, but with ISS improving as payload increases. Polar, SSO and LEO appear consistently successful after a few initial failures.

Exploratory Data Analysis: Launch Success Yearly Trend

- It took 3 years to get to a successful launch.
- Since 2013 the general trend has been upwards, with the exception of 2018.
- Since 2017 a plateau between 80 and a 100% has been reached.



Section 2B: Structured Query Language

Insights drawn from EDA

All Launch Site Names

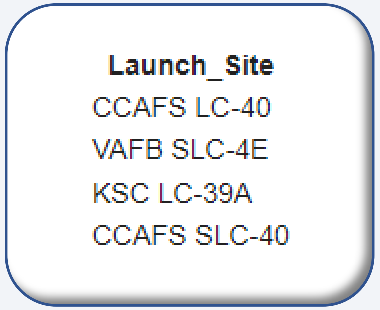
Comments:

- Acronyms:
  CC = Cape Canaveral
  V = Vanderberg
  KSC = Kennedy Space Center

  AF(B,S) = Air Force (Base, Station)
  (S)LC = (Space) Launch Complex
- Vanderberg is on the US west coast (California) the other 3, very close together, on Florida's Atlantic coast.
- In 2020 most names with "Air" Force were changed to "Space" Force e.g. CCAFS is now CCSFS.

Note: for actual query code see Appendices : SQL Queries (1)

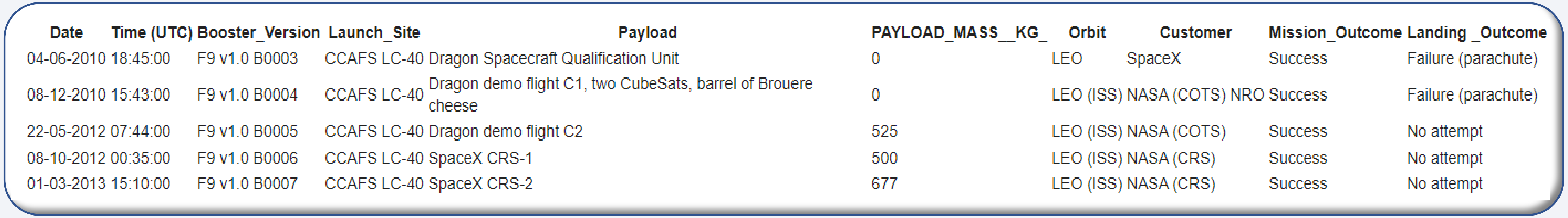
Query Results



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

Launch Site Names Begin with 'CCA'

Query Results



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

Comments:

- First 5 records where launch sites begin with `CCA`
- A useful query to select (extract) all sites with a given geographical location.

Note: for actual query code see Appendices : SQL Queries (1)

Total Payload Mass

Comments:

- Calculate the total payload carried by boosters from a given client (in this case: NASA) for the period under study.
- The payload being a key factor in a launch, this query allows evaluation of the work done for a given client.

Query Results

45 596 kg

Note: for actual query code see Appendices : SQL Queries (1)

Average Payload Mass

Comments:

- Calculate the average payload carried by a given booster (in this case: F9 v1.1) for the period under study.
- The payload being a key factor in a launch, this query allows evaluation of the work done for a given booster type.

Query Results

2 928.4 kg

Note: for actual query code see Appendices : SQL Queries (1)

First Successful Ground Landing Date

Comments:

- The time period covered starts in 2010.
- This type of query allows to see the time necessary to achieve the first successful landing on a given type of launch (in this case: Ground Pad).

Query Results

22 December 2015

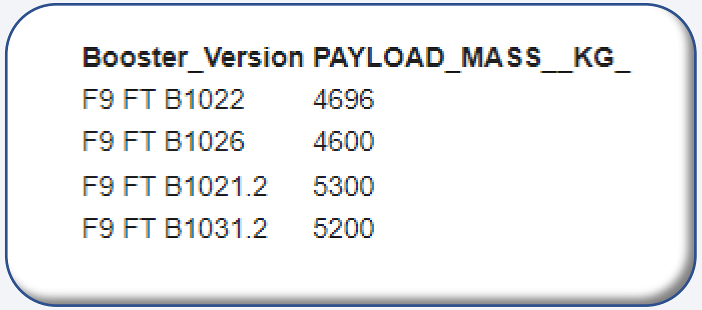
Note: for actual query code see Appendices : SQL Queries (1)

Successful Drone Ship Landing with Payload between 4000 and 6000 kg

Comments:

- This payload range is a good indicator as it covers a significant number of launches.
- This type of query allows to construct a success history with multiple variables involved.

Query Results



Booster_Version	PAYLOAD_MASS__KG_
F9 FT B1022	4696
F9 FT B1026	4600
F9 FT B1021.2	5300
F9 FT B1031.2	5200

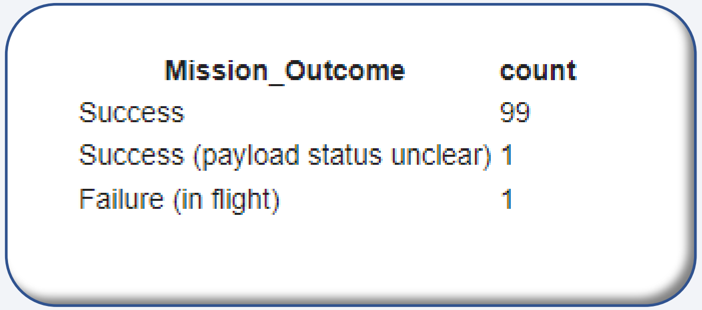
Note: for actual query code see Appendices : SQL Queries (2)

Total Number of Successful and Failure Mission Outcomes

Comments:

- A clear distinction must be made between mission outcomes and landing outcomes (the focus of much of this study).
- The query clearly shows that other the time period the missions were greatly successful.
- This result can also allow for refining the landing outcome results by attributing failure at the proper level.

Query Results



Mission_Outcome	count
Success	99
Success (payload status unclear)	1
Failure (in flight)	1

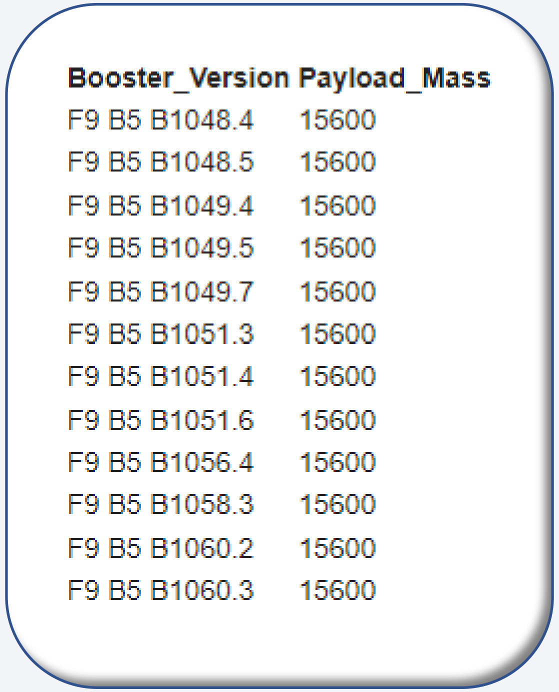
Note: for actual query code see Appendices : SQL Queries (2)

Boosters Carried Maximum Payload

Comments:

- This query shows which distinct booster version carried the maximum payload mass.
- This also allows to detect what the maximum payload was for these boosters; the result indicate that the max payload is always the same, which could indicate the upper limit of booster capacity.
- It could also help track issues in boosters' history of payload requests.

Query Results



Booster_Version	Payload_Mass
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

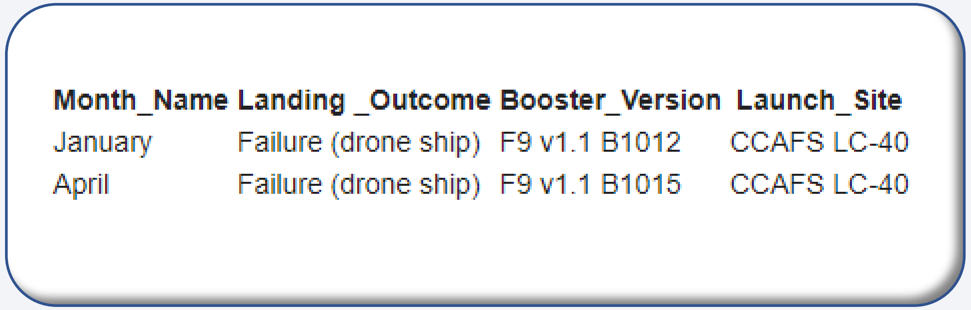
Note: for actual query code see Appendices : SQL Queries (2)

2015 Launch Records

Comments:

- The query lists the failed landing outcomes in drone ship, their booster versions, and launch site names for the year 2015.
- From the results we can see that this type of multi variable query can be useful in correlating certain variables with a given result. Here we can see that both failures were for the same booster version and launch site.

Query Results



Month_Name	Landing _Outcome	Booster_Version	Launch_Site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

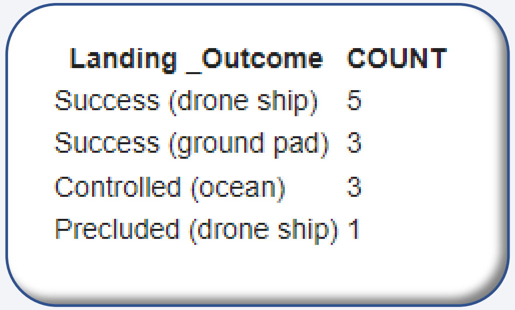
Note: for actual query code see Appendices : SQL Queries (3)

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

Comments:

- This query allows to compare the number of successes for different types of landing.
- From the results Drone Ship has the most successes. Such a study could lead to the determination of the best type of landing for a given mission.

Query Results



Landing _Outcome	COUNT
Success (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Precluded (drone ship)	1

Note: for actual query code see Appendices : SQL Queries (3)

Section 3

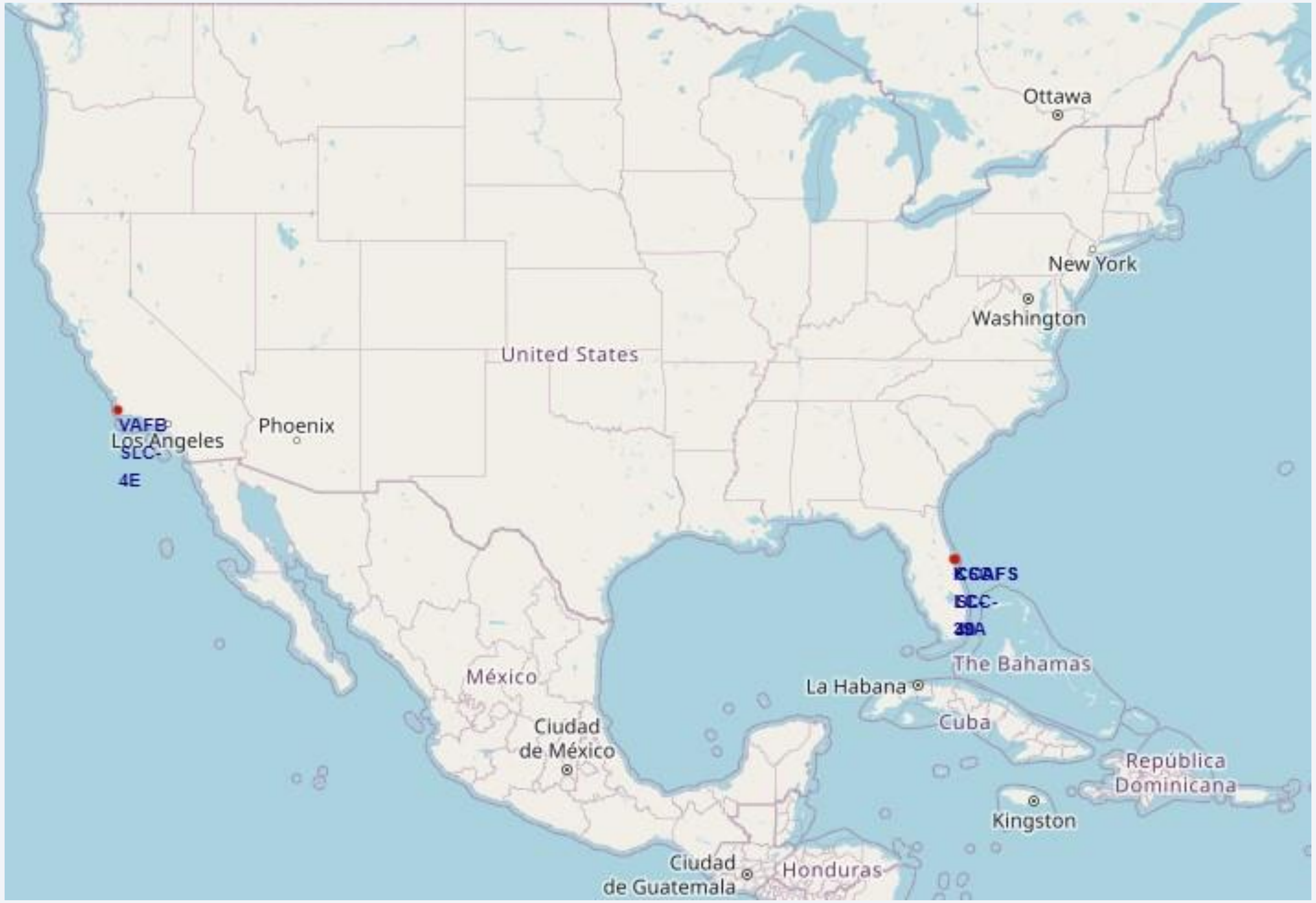
Launch Sites Proximities Analysis

Interactive Launch Site Study with Folium: Locations

- This screenshot is a global view of the location of all launch sites.

- The 3 Florida sites are in close proximity to one another hence the overlapping markers.

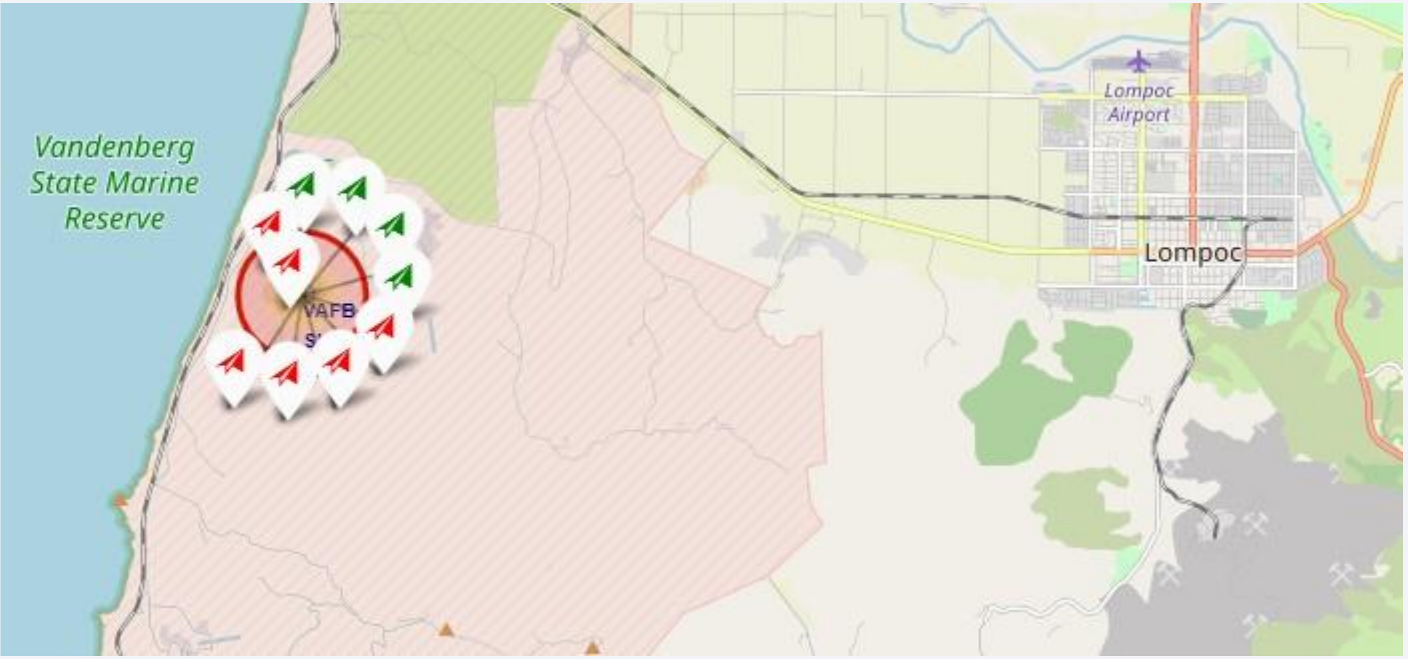
- This map shows that the main characteristic of the site locations is to be on the coastline.



Note: the interactive map has zoom functionality, this is just a screenshot.

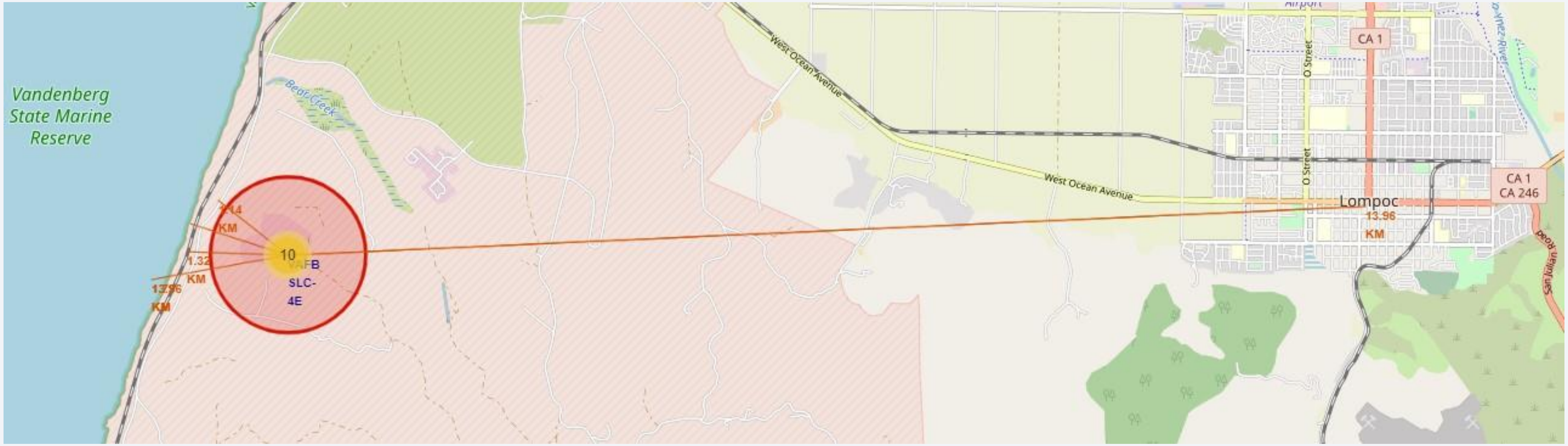
Interactive Launch Site Study with Folium: Launches

- This screenshot is a zoom on the Vanderberg AFS site.
- The markers visualize the launch successes and failure for the site (for the period under study)
- Green indicates a successful outcome and red a failure.



Note: the interactive map has zoom functionality, this is just a screenshot.

Interactive Launch Site Study with Folium: Proximities (1)

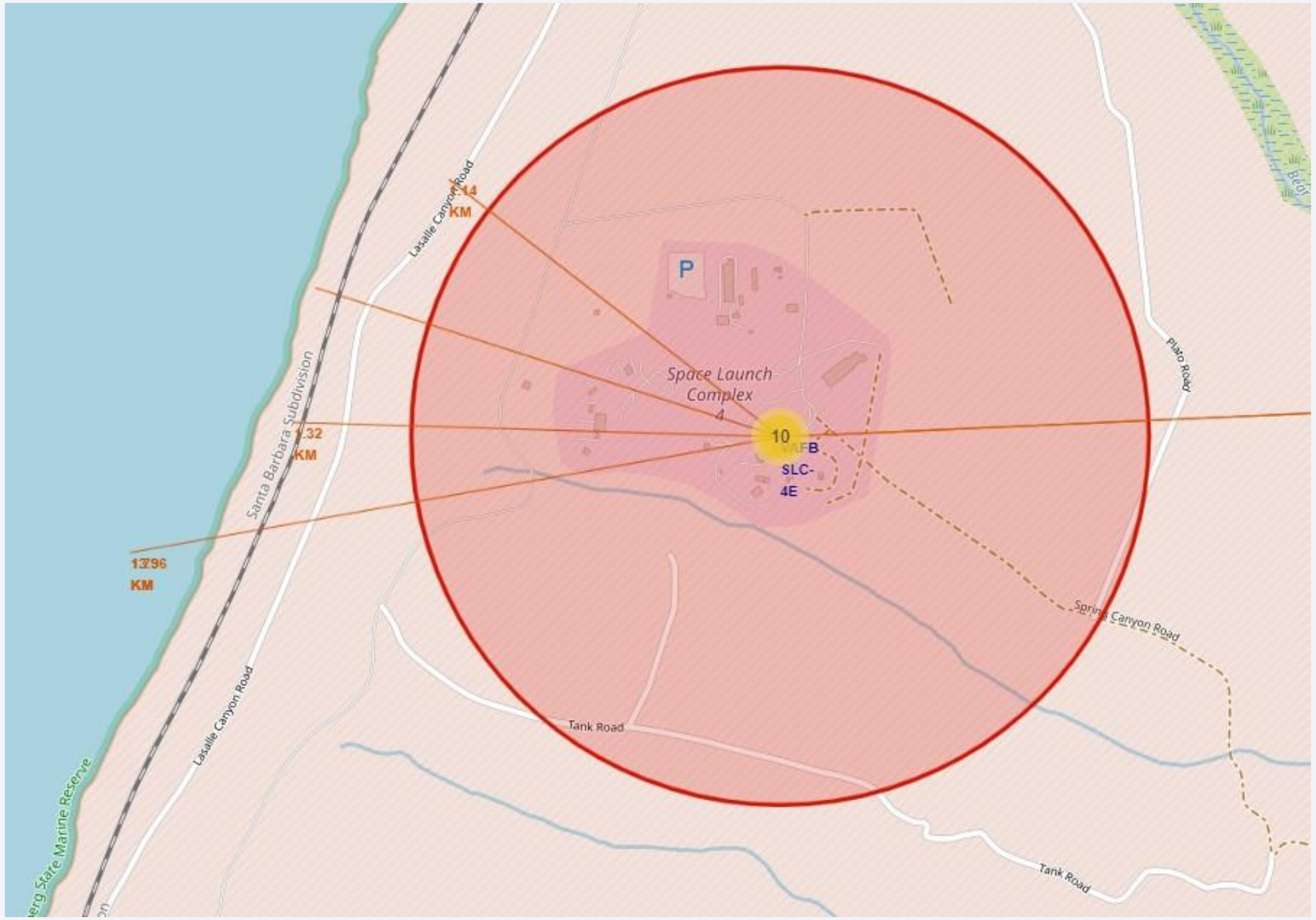


- This screenshot is a zoom on the Vanderberg AFS site and its local environment.
- The main point is to show the proximity (with distance) to the nearest city (Lompoc). Sites are usually at a relative distance from towns.
- Other relevant proximities are shown on the next map.

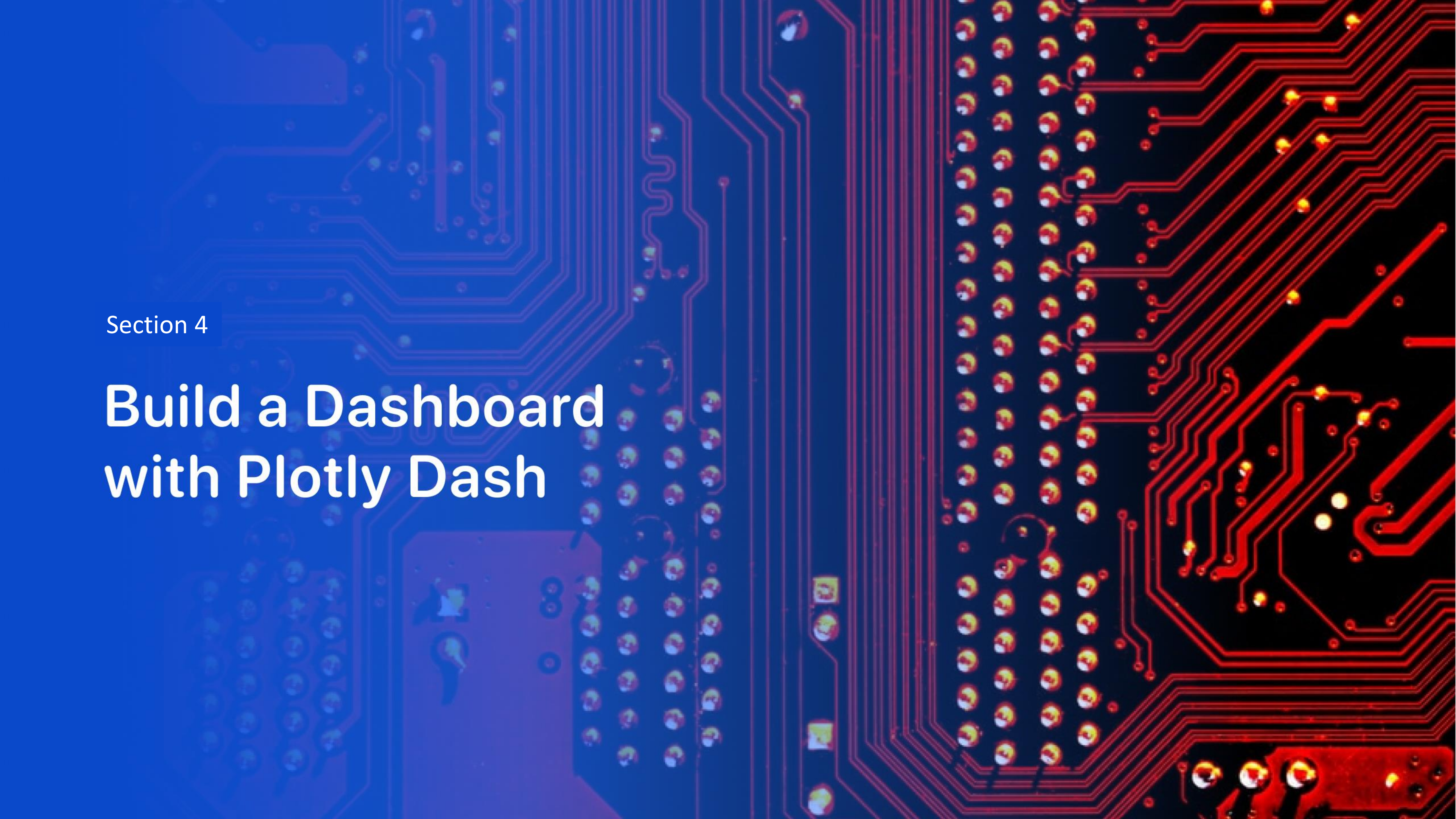
Note: the interactive map has zoom functionality, this is just a screenshot.

Interactive Launch Site Study with Folium: Proximities (2)

- This screenshot is a zoom on the previous map.
- The red lines visualize the distances between the site and the most important features surrounding it such has rail lines and highways, as well as the coastline.
- Distances to the features are shown in Kms, sites are usually in close proximity to these features.



Note: the interactive map has zoom functionality, this is just a screenshot.

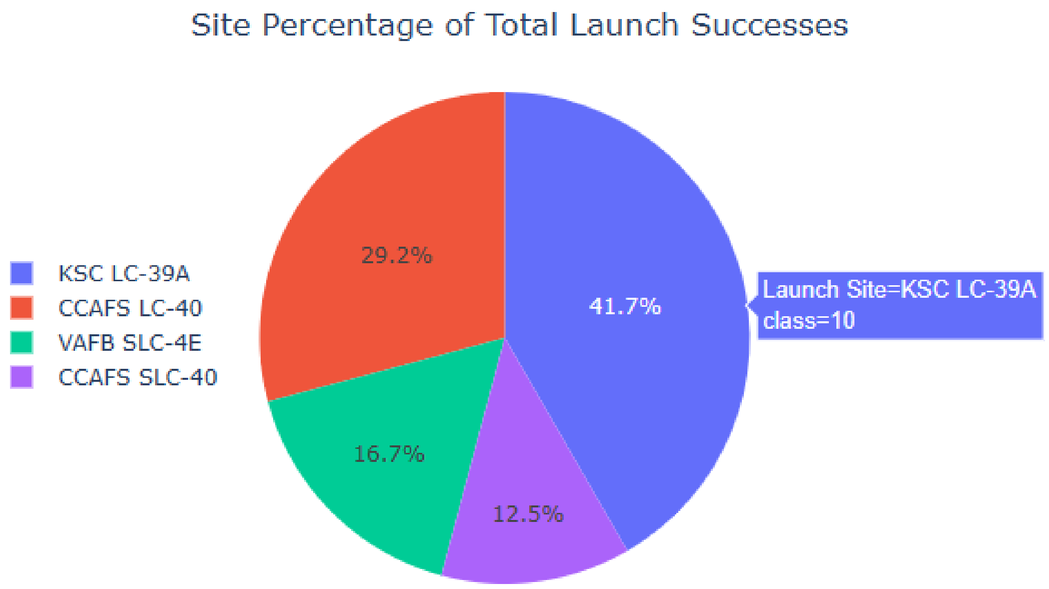


Section 4

Build a Dashboard with Plotly Dash

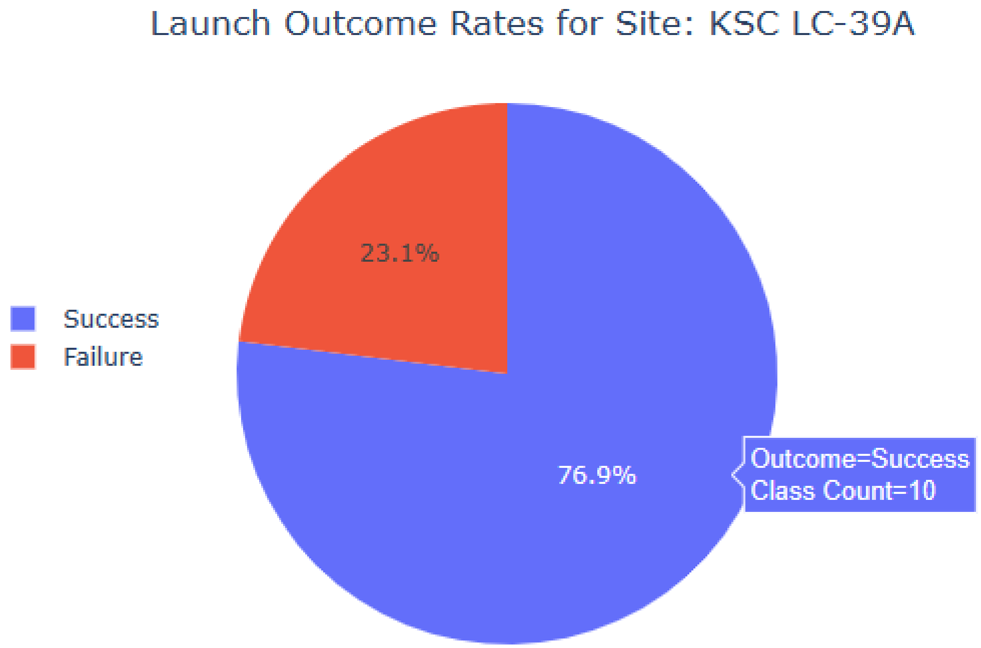
Dashboard Analysis: Launch Successes – All Sites

- Given the total number of launch successes the pie chart shows what share of these successes each site is responsible for.
- The chart shows that the KSC site is leading in that department, however the total of both CCAF sites (which are very close together) is equivalent.
- Also important is the fact that all sites did not get the same number of launches which affects the comparison.

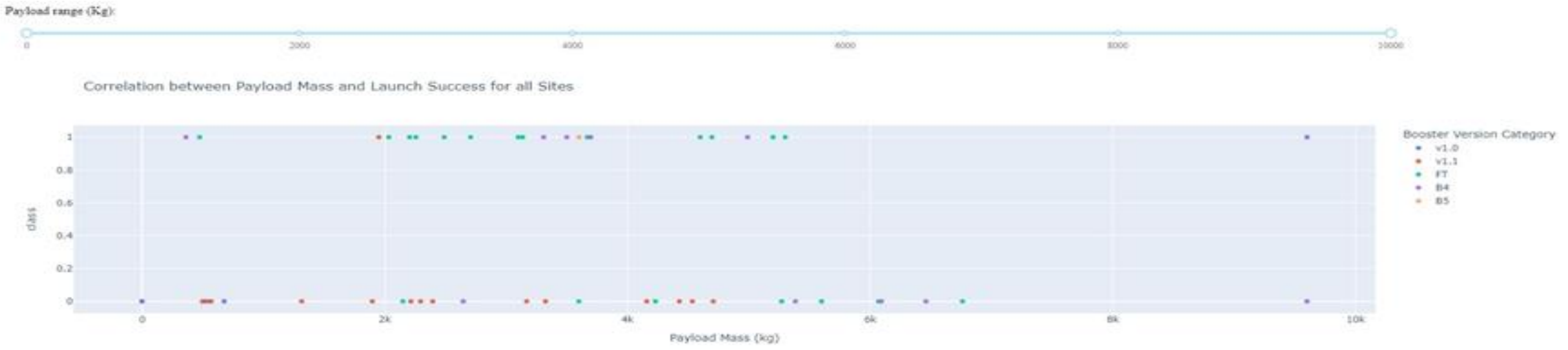


Dashboard Analysis:
Launch Successes – Site KSC LC-39A

- The KSC site had the best share of the successes, this pie chart actually details the success rate of this site.
- It shows that the site had roughly a 3 out 4 success rate (landing outcome) for the period under study.
- This study can be done for the other sites to actually give a better comparison of each site.

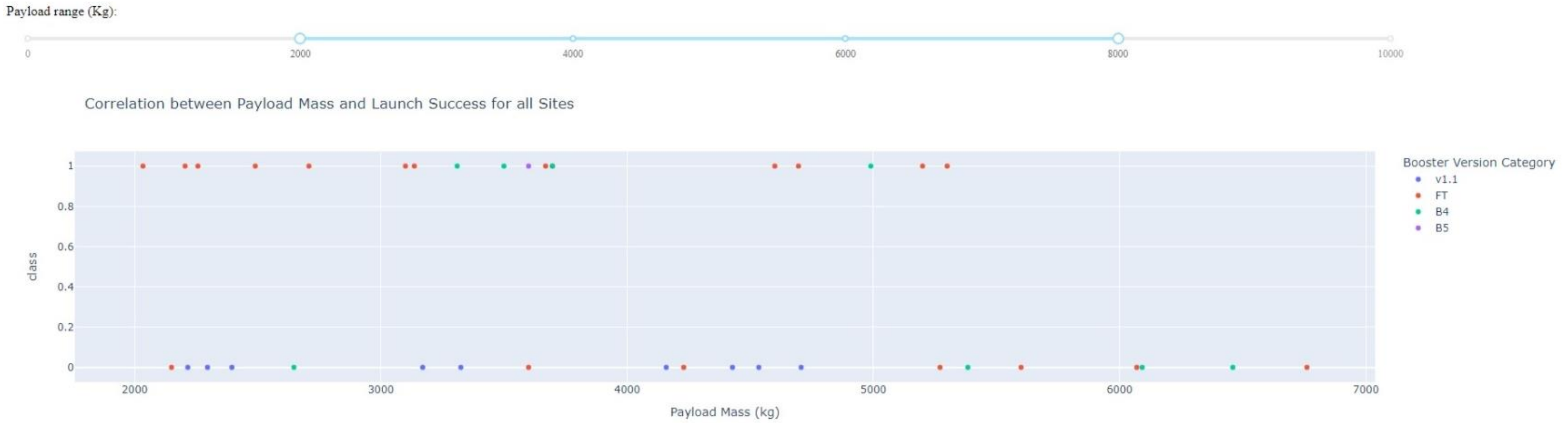


Dashboard Analysis:
Payload Mass Correlation – All Sites, Full Range



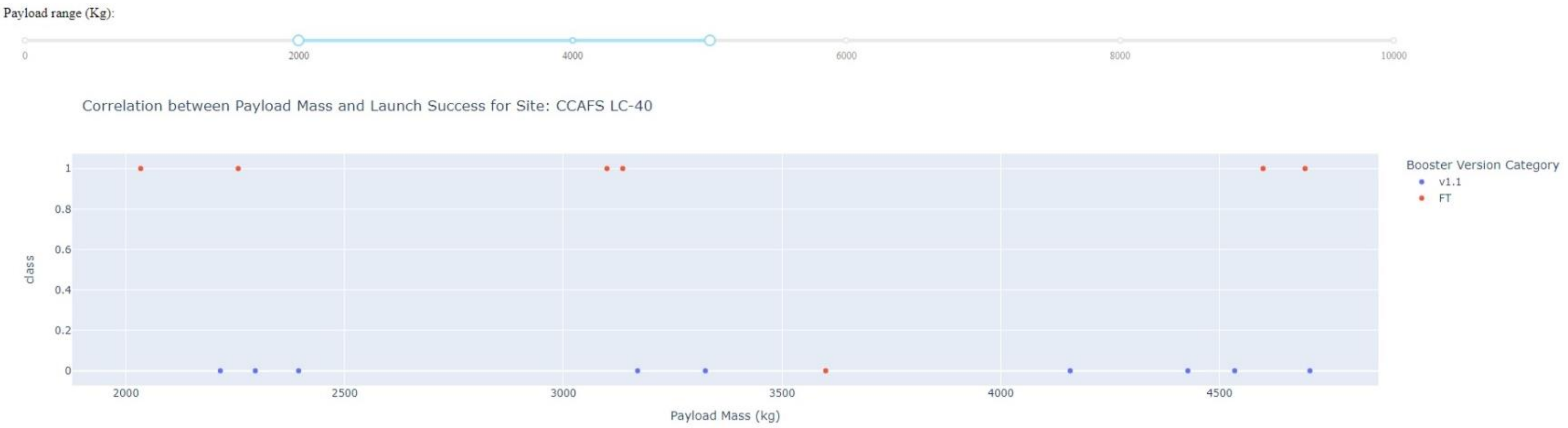
- It can be seen that most of the launches occur in the 2000 – 6000 kg range, with most launches above that range ending in landing failures, with FT being the most successful booster while v1.0 and v1.1 are mostly unsuccessful.
- The 2000 – 4000 kg is the densest and with the highest rate of success. Boosters FT, B4 and B5 are generally successful.

Dashboard Analysis:
Payload Mass Correlation – All Sites, Restricted Range


Payload range (Kg):

- In this restricted range (eliminating extremes) it can be seen that payload above the 5000 kg result mostly in failures.
- It can also be observed that Booster Version v1.1 is unsuccessful and that version B5 has only be used once.

Dashboard Analysis:
Payload Mass Correlation – Site CCAFS LC-40, Restricted Range



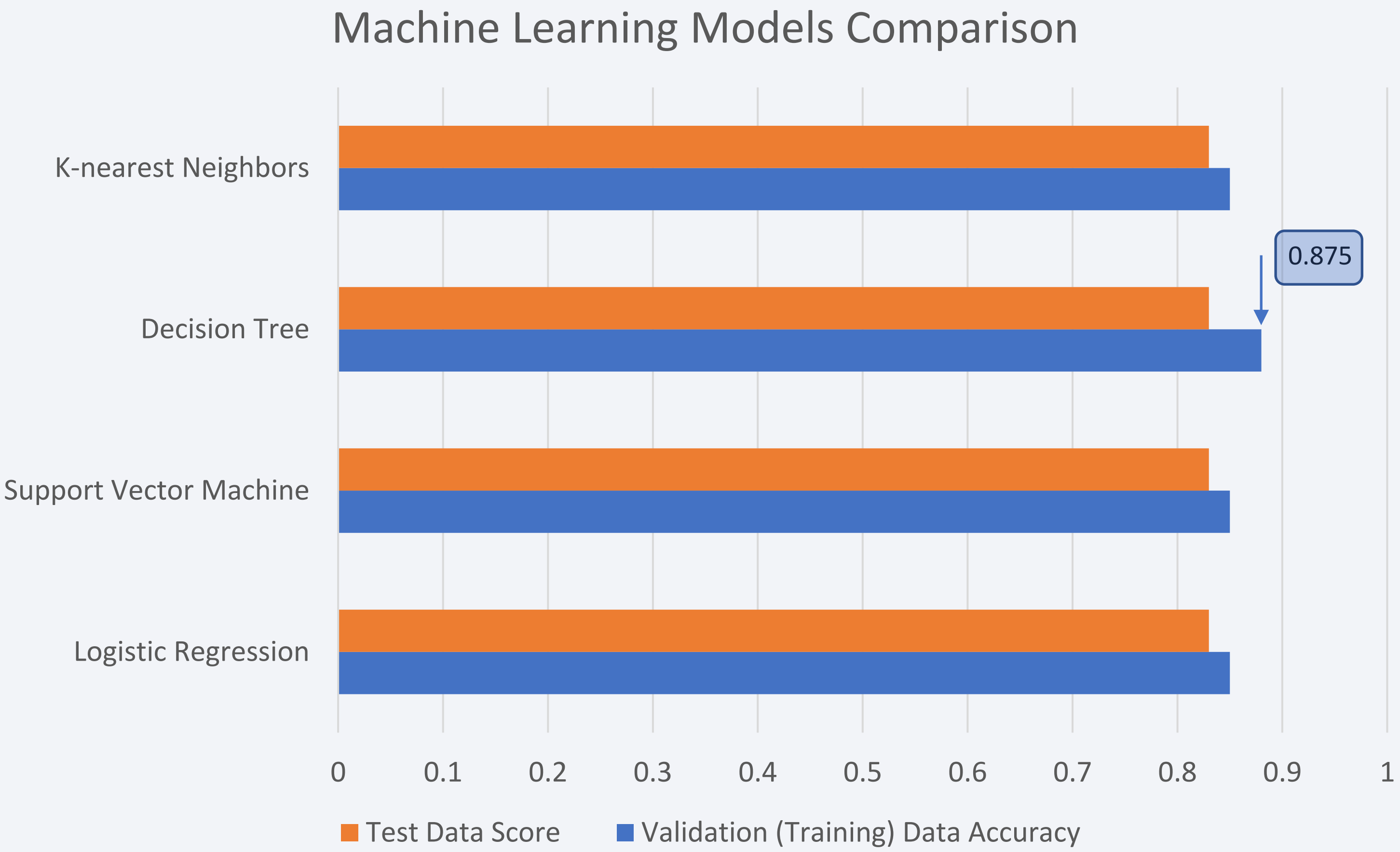
- The range is set at 2000 – 5000 kg with the selector for site CCAFS LC-40.
- For these parameters it is clear that the success driving factor is in fact the Booster Version with V1.1 having a 100% failure while FT is almost perfect.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

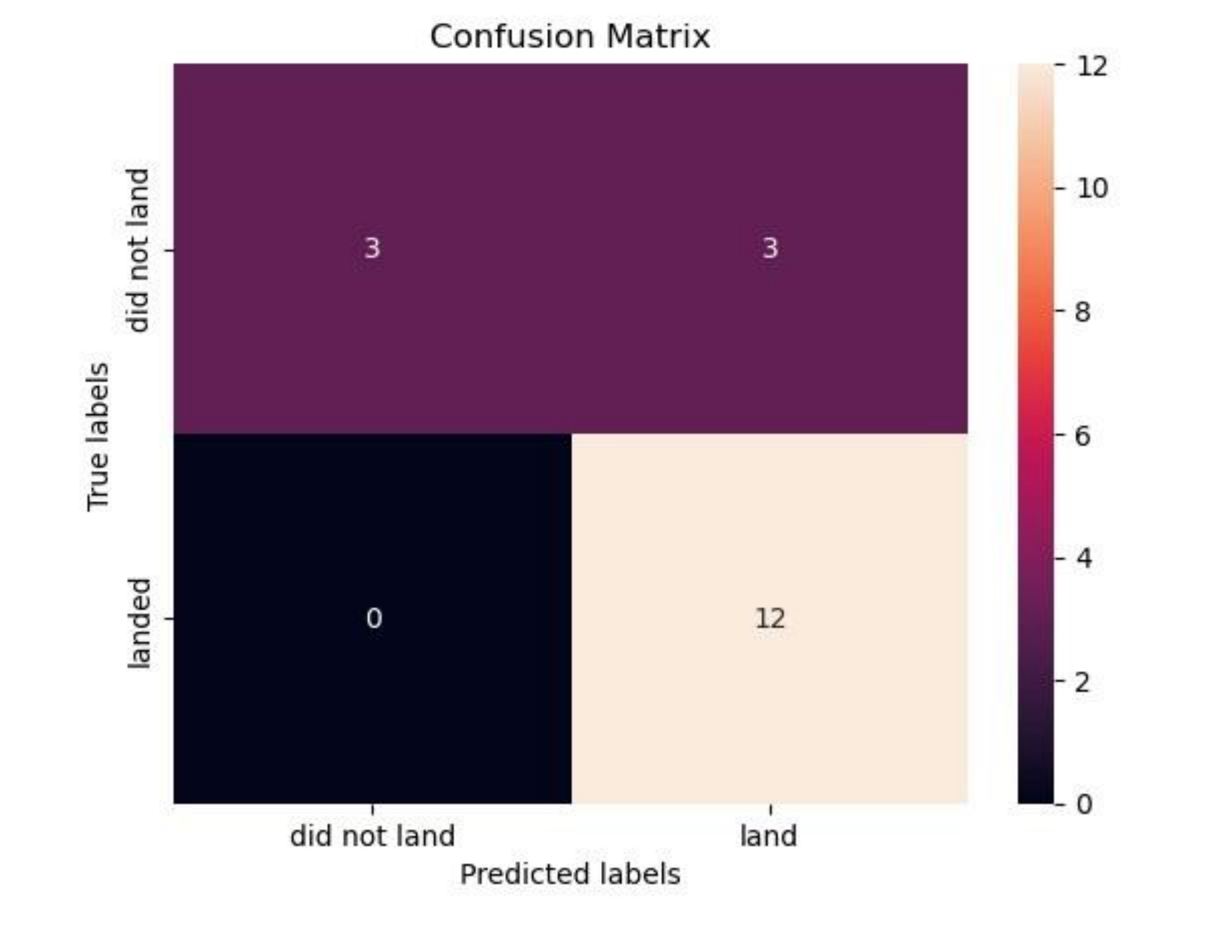
- All 4 models have the same Test Data score which prevents deciding on a “best” model.
- However, the Decision Tree model has a slightly higher accuracy, therefore it should be the one to use in this case.



Note: see Appendix Classification for further comments.

Confusion Matrix

- Showing the confusion matrix for the Decision Tree model.
- The matrix show that while the model is very accurate for successful outcomes it is not so for the “did not land” outcome where the model registers 3 “false positive” outcomes.
- Consequently, the model’s precision (TP / TP + FP) is only 80%.



Note: see Appendix Classification for further comments.

Conclusions

- SpaceX launch data can be collected
- Significant features can be determined
- Models can be trained
- Landing outcomes can be predicted with good accuracy
- As more data becomes available the process should be iterated to refine the models

Appendices

- Data Sets
- Orbit Types
- SQL Queries
- Classification

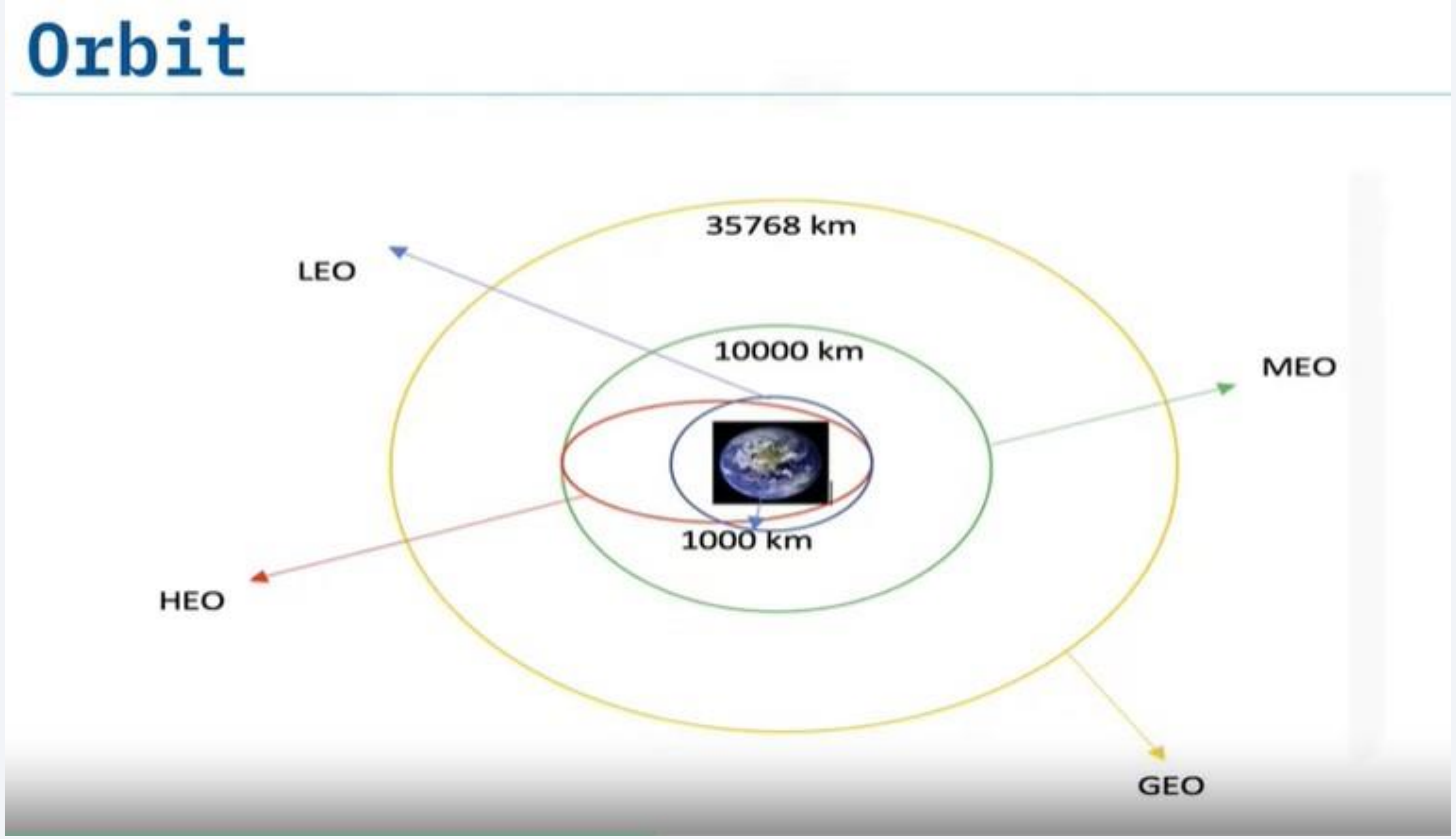
Appendix: Data Sets

GitHub URL: All Project Data Sets

- The linked repository contains all data sets created or used at various stages of the project.
- This allows to recreate the results when necessary or to compare inputs when more data becomes available.

Appendix: Orbit Types

- LEO: Low Earth Orbit
- HEO: Highly Elliptical Orbit, good for communications
- MEO: Medium Earth Orbit, contains the semi-synchronous orbit used by GPS
- GEO: Geostationary Orbit, many uses



Appendix: SQL Queries (1)

Slide xx Query: Listing of Distinct Launch Sites

```
%sql SELECT DISTINCT "Launch_Site" FROM "SPACEXTBL"
```

Slide xx Query: First 5 Launches for Sites with Names starting by “CCA”

```
%sql SELECT * FROM "SPACEXTBL" WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5
```

Slide xx Query: Total Payload Mass for NASA (CRS)

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS Sum FROM "SPACEXTBL" WHERE "Customer" = "NASA (CRS)"
```

Slide xx Query: Average Payload Mass for Booster Version F9 v1.1

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") AS Avg FROM "SPACEXTBL" WHERE "Booster_Version" = "F9 v1.1"
```

Slide xx Query: First Successful “Ground Pad” Landing Outcome

```
%%sql SELECT "Date", MIN(substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2)) AS SuccDate FROM "SPACEXTBL"
 WHERE "Landing _Outcome" = "Success (ground pad)"
```

Appendix: SQL Queries (2)

Slide xx Query: Successful Boosters for Payload Mass 4000 – 6000 kg

```
%%sql SELECT "Booster_Version", "PAYLOAD_MASS__KG_" FROM SPACEXTBL WHERE "Landing _Outcome" = 'Success (drone ship)’
 AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000
```

Slide xx Query: Total Successes vs. Failures

```
%sql SELECT "Mission_Outcome", COUNT(*) as count FROM SPACEXTBL GROUP BY TRIM("Mission_Outcome") ORDER BY count DESC
```

Slide xx Query: Booster Version with Maximum Payload Mass

```
%%sql  SELECT "Booster_Version", "PAYLOAD_MASS__KG_" AS Payload_Mass FROM SPACEXTBL
 WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_")
 FROM SPACEXTBL) ORDER BY "Booster_Version"
```

Appendix: SQL Queries (3)

Slide xx Query: Landing Failures 2015 (Drone Ship)

```
%%sql

SELECT case substr(Date,4,2)

when '01' then 'January' when '02' then 'Febuary' when '03' then 'March' when '04' then 'April' when '05' then 'May’

when '06' then 'June' when '07' then 'July' when '08' then 'August' when '09' then 'September' when '10' then 'October’

when '11' then 'November' when '12' then 'December' else '' end

AS Month_Name, "Landing _Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTBL

WHERE (substr(Date,7,4) = '2015')

AND ("Landing _Outcome" = "Failure (drone ship)")
```

Slide xx Query: Landing Outcomes

```
%%sql

SELECT "Landing _Outcome",COUNT("Landing _Outcome") AS COUNT FROM SPACEXTBL

WHERE (substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2) BETWEEN '20100604' AND '20170320')

AND ("Landing _Outcome" NOT LIKE "Failure%") AND ("Landing _Outcome" NOT LIKE "Uncontrolled%")

AND ("Landing _Outcome" NOT LIKE "%attempt%")

GROUP BY "Landing _Outcome" ORDER BY count("Landing _Outcome") DESC
```

Appendix: Classification

- The data of 90 launches was split randomly in a training sample of 72 and test sample of 18.
- On this fairly small sample the scores and confusion matrices for all models were identical (see the classification section for results), it might, therefore be advisable to revise the results when a bigger data sample becomes available.

Thank you!