



IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

João Coroa
21/06/2024



Outline


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

Executive Summary

This report will provide important insights about the success rate of the landing of the first stage of the Space X Falcon 9. Information will be requested through an HTTP method, organized accordingly to its outcome and then given to machine learning algorithms to try to predict the success rate.



Introduction

- Rocket launches are costly: up to 165 M\$ each 
- Space X revolutionizes the market by bringing down the cost to only 62 M\$

HOW? → Space X can reuse the first stage of Falcon 9!

- **Problem:**
 - ❖ How can we determine the cost of a launch?
- **Objective:**
 - ❖ Determine the success rate that the 1st stage of Falcon 9

Section 1

Methodology

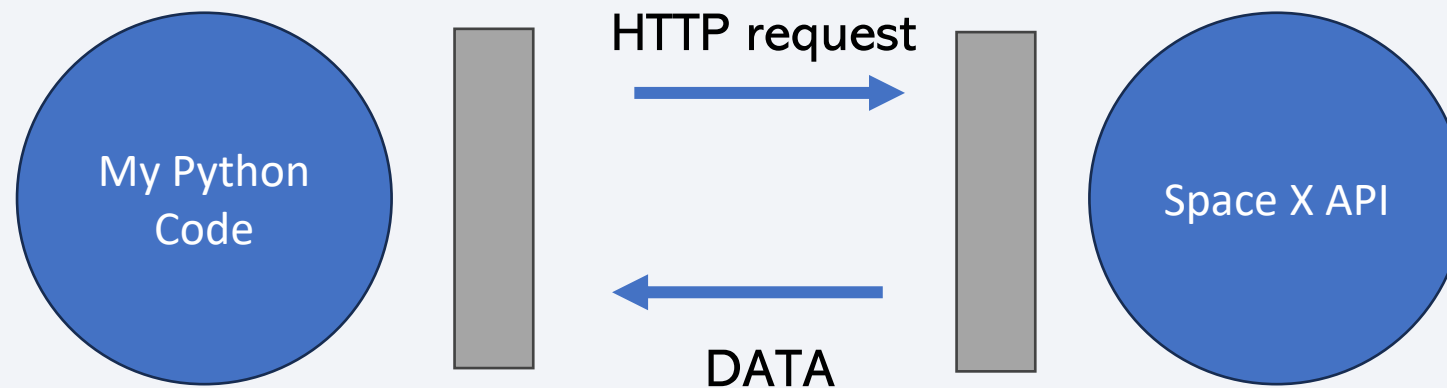
Methodology

Executive Summary

- Data collection methodology:
 - Data was collected through a HTTP Method.
- Data wrangling
 - Data was organized taking in account the landing outcomes (“Classes”)
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using classification models
 - How to build, tune, evaluate classification models

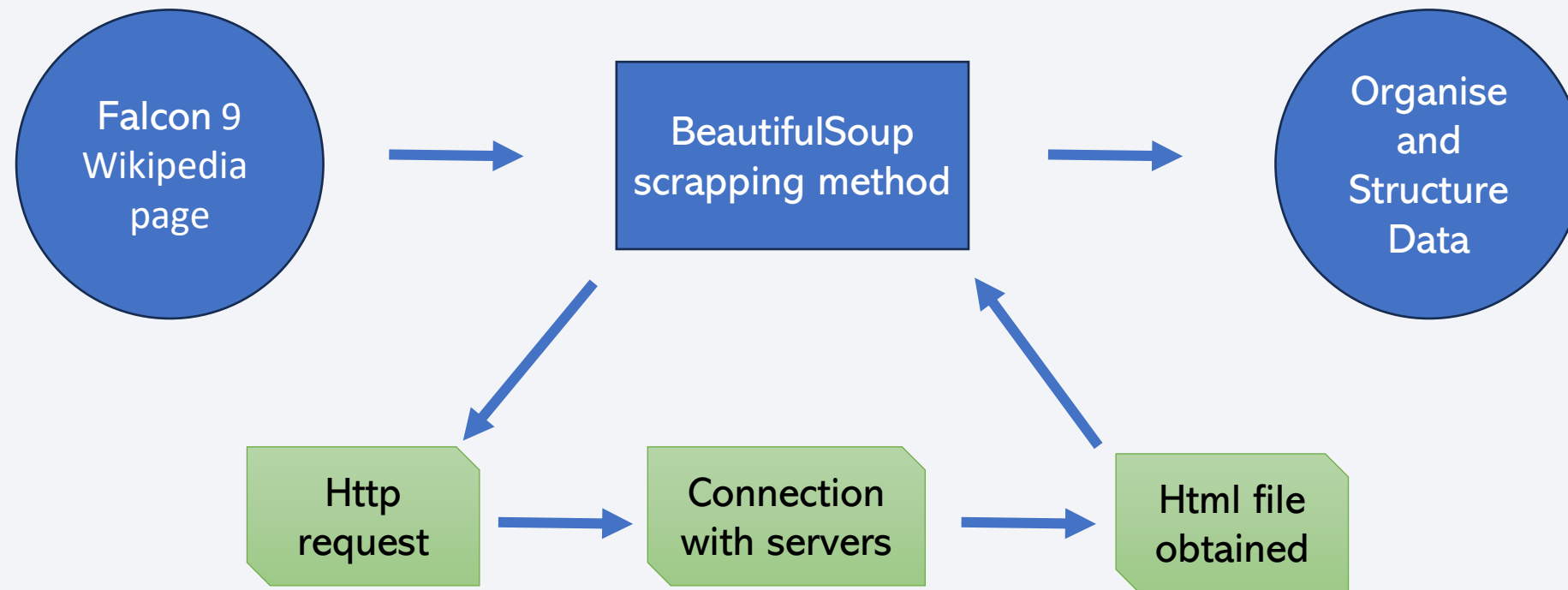
Data Collection – SpaceX API

- DataSets were collected through the library “requests” calling the “get” method, which is a HTTP method.



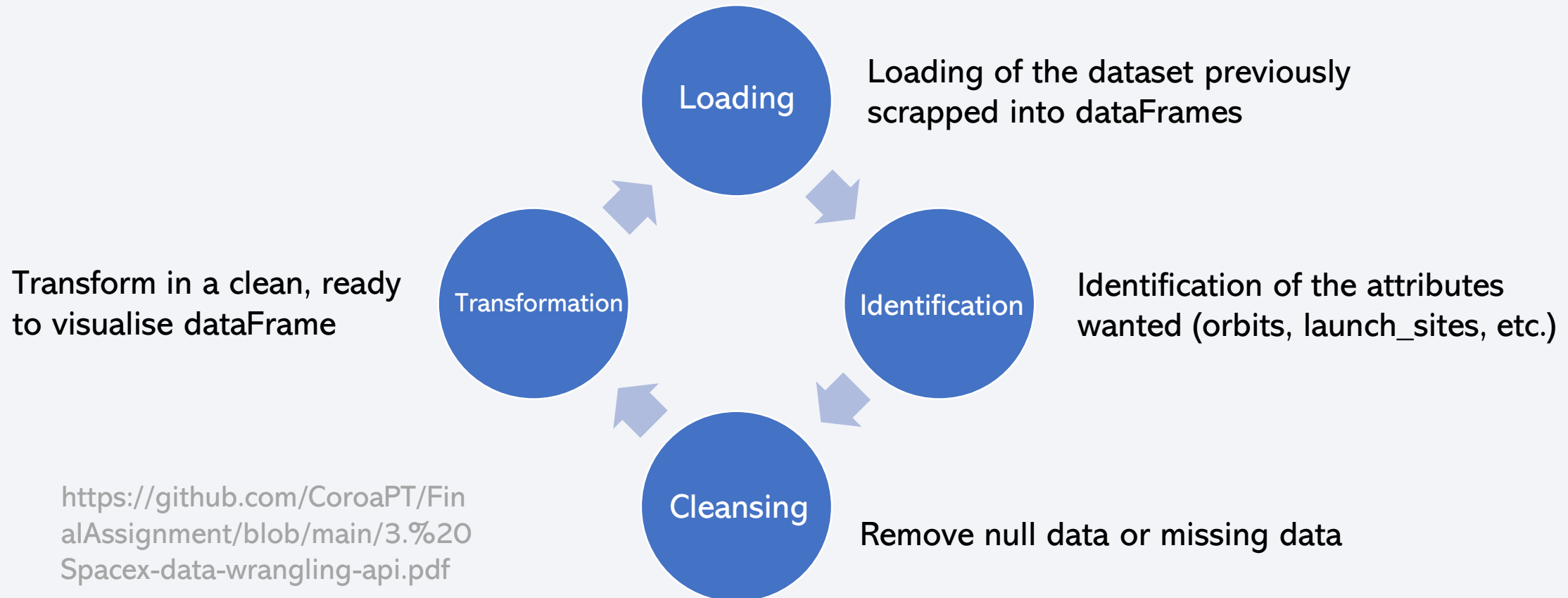
Data Collection – Scraping

- DataSets were collected through Webscrapping, by calling the method "BeautifulSoup" and retrieving information from a website.



Data Wrangling

- Data was organised around the success rate of landing, and transformed into classes where "1" means successful landing and "0" means unsuccessful.



EDA with Data Visualization

- Some examples of Data Visualization offer valuable insights:



BAR PLOT

It shows the relationship between a numeric and a categorical variable



LINE PLOTS

To track changes over short and long periods of time.

EDA with SQL

- The following SQL queries were performed:
 - `SELECT` distinct(feature) `FROM` SpaceXTable
 - `SELECT` feature `FROM` SpaceXTable `WHERE` feature `LIKE` "" `LIMIT` x
 - `SELECT` sum(feature).. ; `SELECT` avg(feature)... ; `SELECT` min(feature)...
 - `SELECT` count(feature) with `GROUPBY` etc.

For more examples, please check:

<https://github.com/CoroaPT/FinalAssignment/blob/main/5.%20Spacex-sqlite-api.pdf>

Build an Interactive Map with Folium

- For an easy visualisation of the mapping of the launches, some features were included:
 - **Circle** – to highlight an area of interest, i.e. the launching site;
 - **Marker** - to add a text label on a specific launching site;
 - **Cluster** – to group the amount of launches (successful and unsuccessful) in each launch site;
 - **Mouse Position** – to know in real-time while visualizing the map, the latitude and longitude coordinates;
 - **Real Distance** – to know the distance between launch sites and various points of interest (e.g., coast, highway, train station, etc.)
 - **Lines** - to draw a line over the calculated distance in "Real Distance"

Build a Dashboard with Plotly Dash

- For an even more interactive visualisation, a dash app can be built:
 1. Select which launch site the user is interested.
 2. The graphs update automatically with the information stored in the dataframe!

Graphs include:



Pie charts

To quickly evaluate the total successful launches per site



Scatter charts

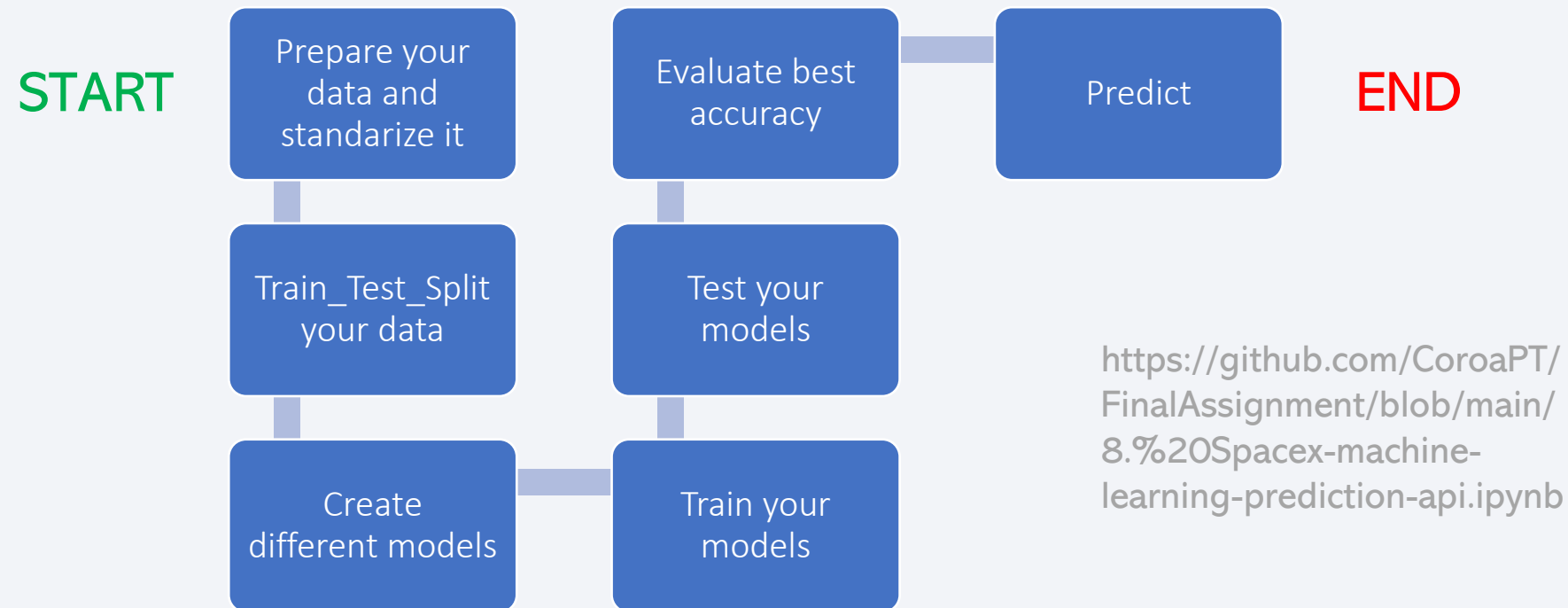
To quickly evaluate the correlation between two variables

Side Bars

To interactively change data in the graphs

Predictive Analysis (Classification)

- To predict a successful launch rate, the following criteria was considered:



Results

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

The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of red and cyan. A faint, light blue grid pattern is also visible, particularly in the lower half of the image. The overall effect is dynamic and technological.

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

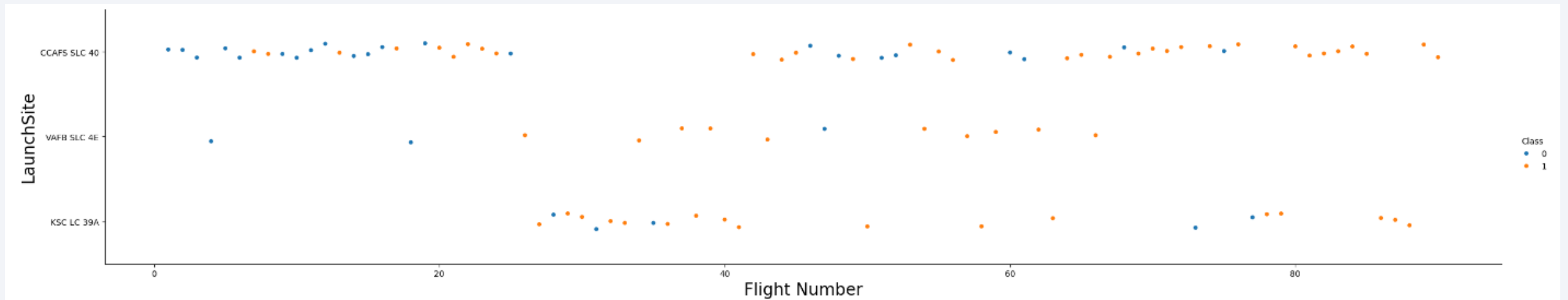


Figure 1 - Flight number Vs. Launch sites

Observations

- For a higher number of flights, the success rate increases (class = 1);
- CCAFS-SLC 40 seems to be a preferable launch site.

Payload vs. Launch Site

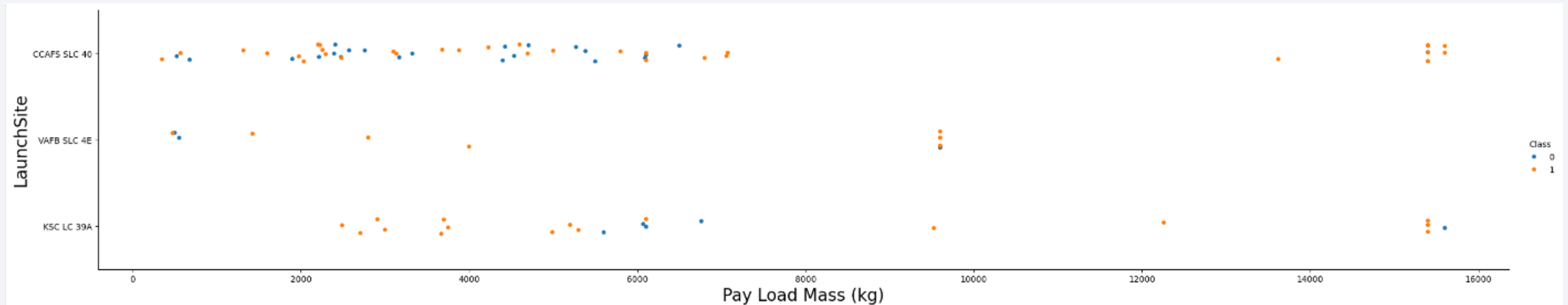


Figure 2 - Pay Load Mass Vs. Launch sites

Observations

- Lighter pay loads are mostly launched from CCAFS-SLC;
- Heavier pay loads are only launched from CCAFS-SLC and KSC-LC, with a better success rate for the former;
- VAFB-SLC doesn't do launches for masses superior to 10,000 kg.

Success Rate vs. Orbit Type

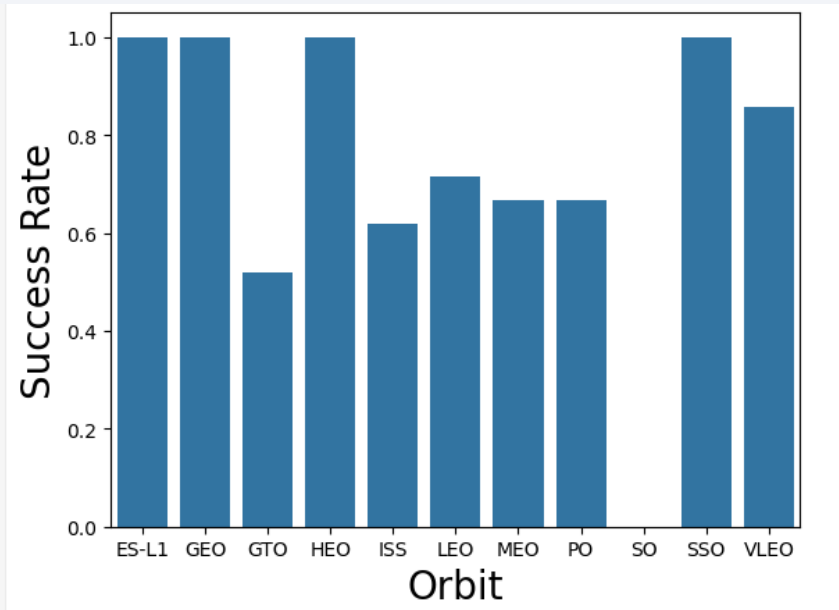


Figure 3 - Orbits and their success rate

Observations

- When launching to SO orbit, either the first stage landing has no success or there is no data;
- The best orbits for a success 1st stage landing are ES-L1, GEO, HEO and SSO;
- The worst orbit for a success 1st stage landing is GTO.

Flight Number vs. Orbit Type

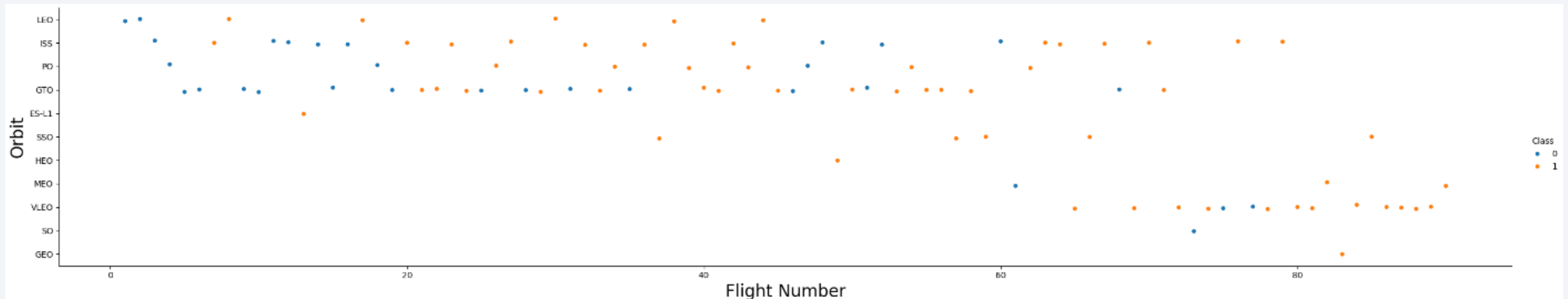


Figure 4 – Flight Number Vs. Orbits

Observations

- The success rate starts to increase in LEO and VLEO with the flight number;
- At later number of flights, another orbits are added;
- GTO doesn't have any relationship with the flight number.

Payload vs. Orbit Type

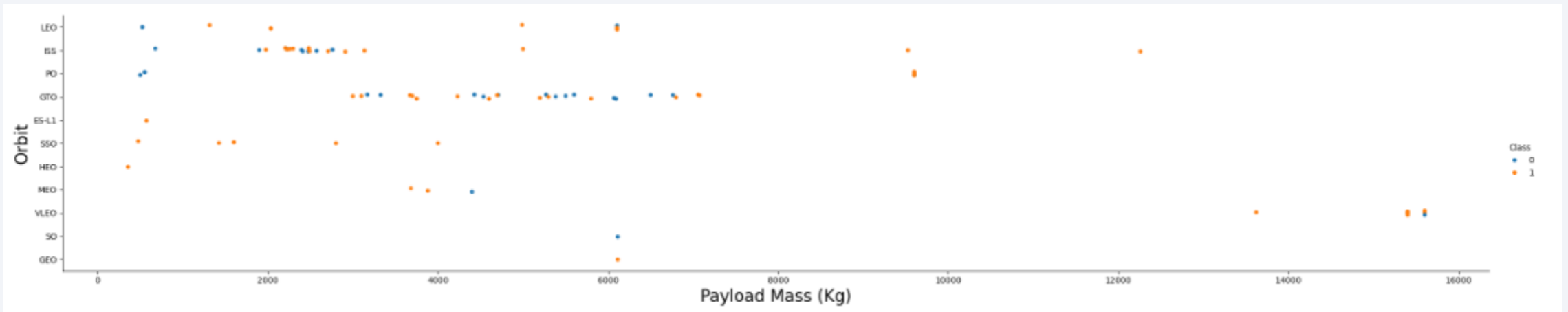


Figure 5 – Payload Mass Vs. Orbits

Observations

- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS;
- GTO hard to analyse as both positive landing rate and negative landing (unsuccessful mission) exist for close payload masses.

Launch Success Yearly Trend

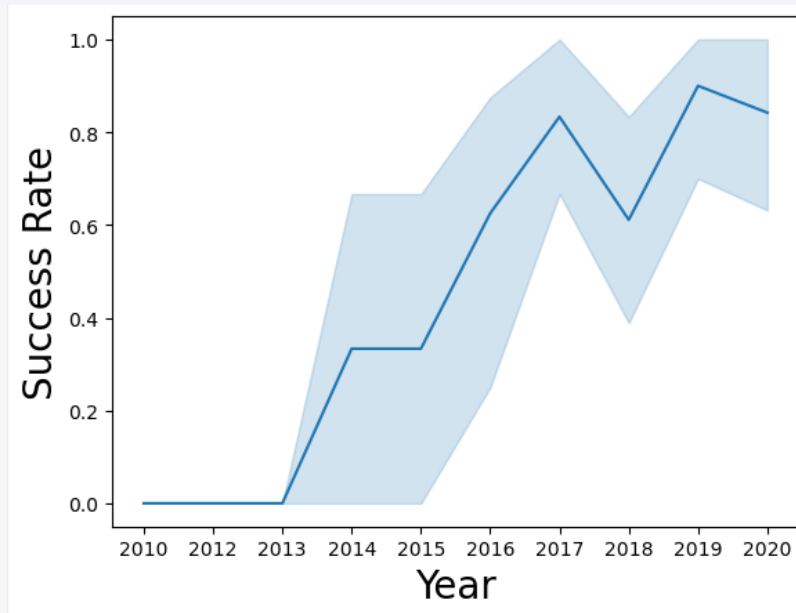


Figure 6 – Success Rate of the 1st stage landing with time

Observations

- With time, success rate has been increasing, meaning that technology has been improving;
- It started in 2013 and the data goes until 2020.

All Launch Site Names

```
%sql SELECT distinct(Launch_Site) FROM SPACEXTABLE
```

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

Done.

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

Observations

- By choosing "**distinct**", we are selecting the unique code for the launch_site

Launch Site Names Begin with 'CCA'

```
%sql SELECT Launch_Site FROM SPACEXTABLE where Launch_Site LIKE 'CCA%' LIMIT 5
```

* sqlite:///my_data1.db

Done.

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Observations

- The important here is the search "LIKE" and the "LIMIT" to only 5 samples.

Total Payload Mass

```
%sql SELECT sum(PAYLOAD_MASS__KG_) AS 'Total payload mass carried by boosters la
```

Observations

- We use "sum" to collect all values from the column payload.
- We use "as" to define the query result

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

```
Done.
```

Total payload mass carried by boosters launched by NASA CRS (KG)

45596

Average Payload Mass by F9 v1.1

```
%sql SELECT avg(PAYLOAD_MASS__KG_) AS 'Average payload m
```

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

```
Done.
```

Average payload mass carried by boosters F9 v1.1 (KG)

2928.4

Observations

- Now we "avg" to collect the average of all values from the column payload.
- We use "as" to define the query result

First Successful Ground Landing Date

```
%sql SELECT min(Date) FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Success%'
```

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

min(Date)

2015-12-22

Observations

- Now we "min" to collect the smaller of all values from the column date.
- We use "LIKE" to match the search with the landing outcome that only has "success".

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT Booster_Version FROM SPACESTA
```

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

```
Done.
```

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Observations

- We select first the booster version that has success as outcome.
- Then we select the payload "**between**" 4000 and 6000.

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT count(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome LIKE  
* sqlite:///my_data1.db  
Done.  
  
count(Mission_Outcome)  
101
```

Observations

- Here I am only showing the success rate.
- To see the failing missions, we would have to change the "LIKE" argument or include another query.

Boosters Carried Maximum Payload

```
[15]: %sql SELECT DISTINCT(Booster_Version) FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ =  
      * sqlite:///my_data1.db  
Done.
```

```
[15]: 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

Observations

- Select "**distinct**" to groupby booster version.
- Use "**max**" function to look for max payload and then search the booster that carried it.

2015 Launch Records

```
%sql SELECT substr(Date,6,2) AS 'Month', Landing_Outcome, I
```

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

Done.

roa/Downloads/jupyter-labs-eda-sql-coursera_sqlite.html

jupyter-labs-eda-sql-coursera_sqlite

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

Observations

- "substr" was used to subtract month and year from the date column and replace it with values that SQLite accepts.
- We wanted to list some information...
- The query returned the information shown on the result

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

```
: %%sql SELECT Landing_Outcome, count (Landing_Outcome) as Landing_Outcome_Values
      GROUP BY Landing_Outcome ORDER BY count(Landing_Outcome) DESC
```

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

```
Done.
```

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

Observations

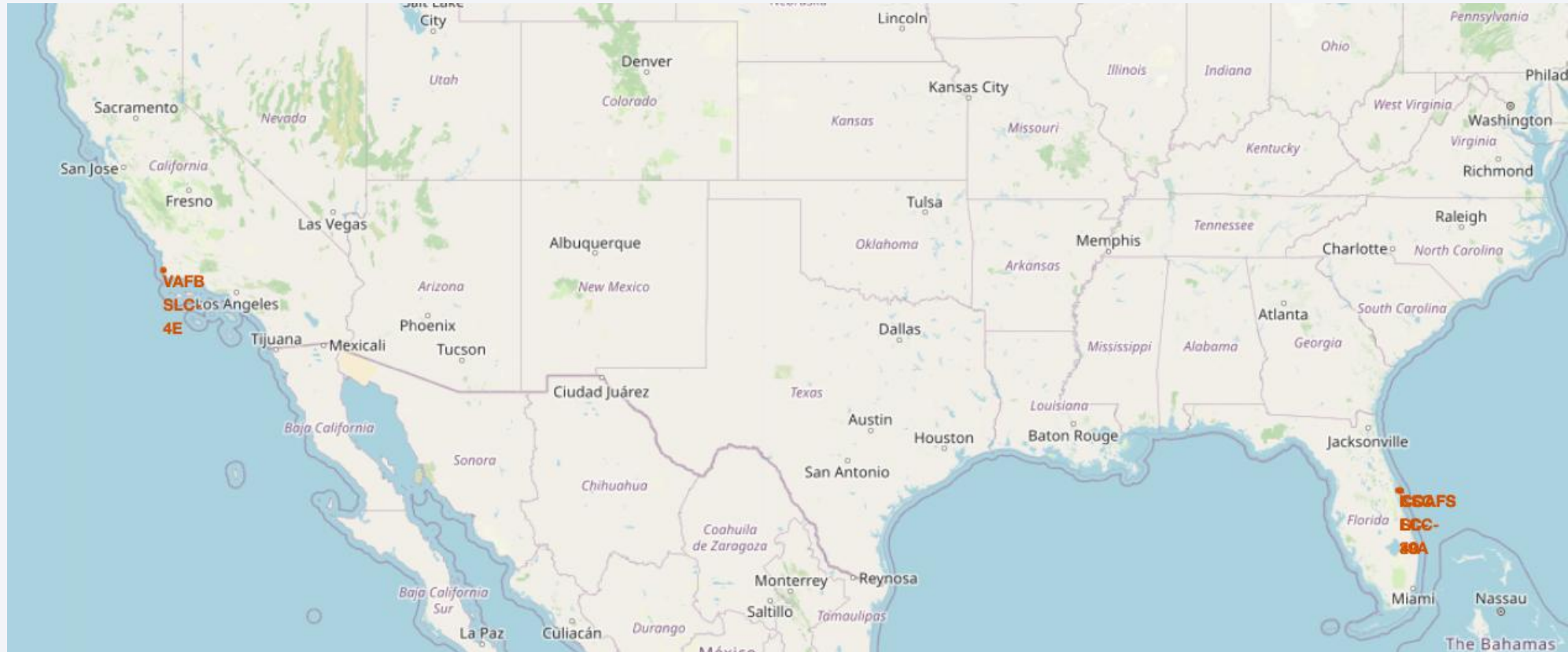
- Several SQL commands explained before were used
- "Order by desc" was used to rank them per number of events.

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The image is a composite of a dark blue sky and a view of the Earth's surface, which is covered in a dense network of city lights and clouds. The lights are concentrated in the lower right portion of the image, while the upper left shows a clear blue sky.

Section 3

Launch Sites Proximities Analysis

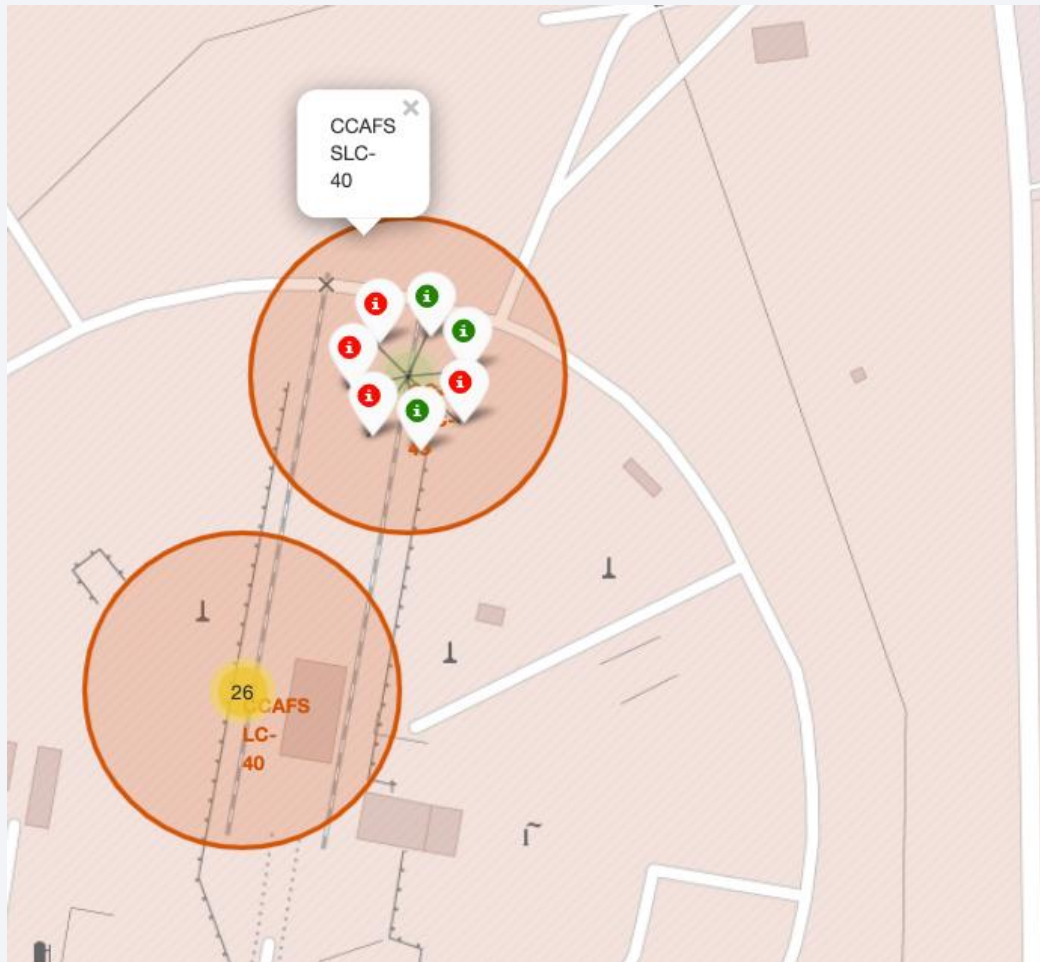
Mapping with Folium - Launch site locations



Observations

- Launch sites are distributed at the extremities of the country: east (Florida) and west (California).

Mapping with Folium - To and To Not Success



Observations

- Zoomed in area in one of the launch sites;
- **Informative greens** show successful launches
- **Informative red** show unsuccessful launches

Mapping with Folium - where is it?



Observations

- Calculated distance (in a straight line) from the launch site and the coast.

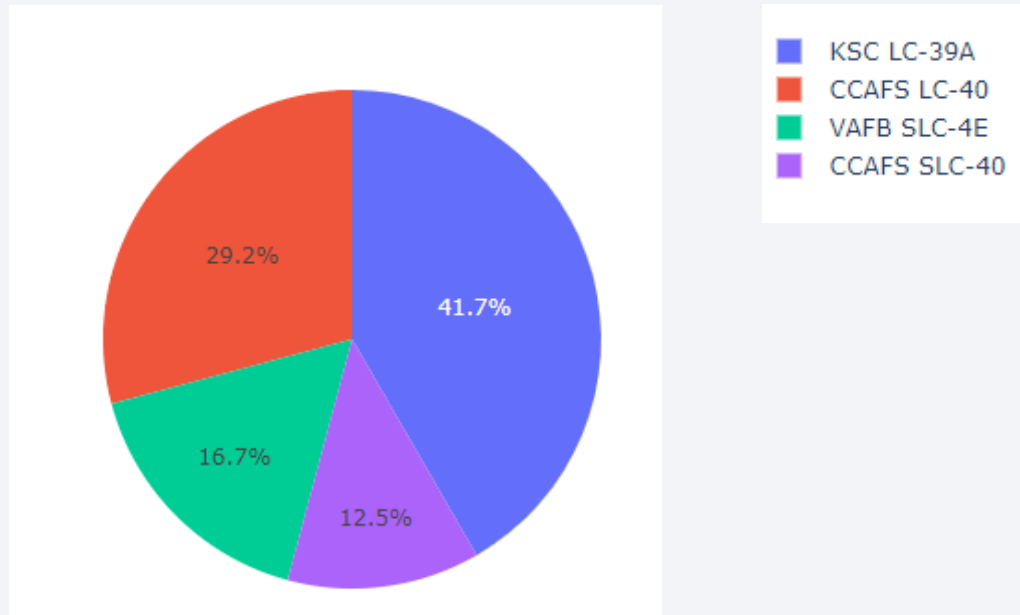


Section 4

Build a Dashboard with Plotly Dash

Pie chart with Dash App – All Sites

Total Success Launches by Site

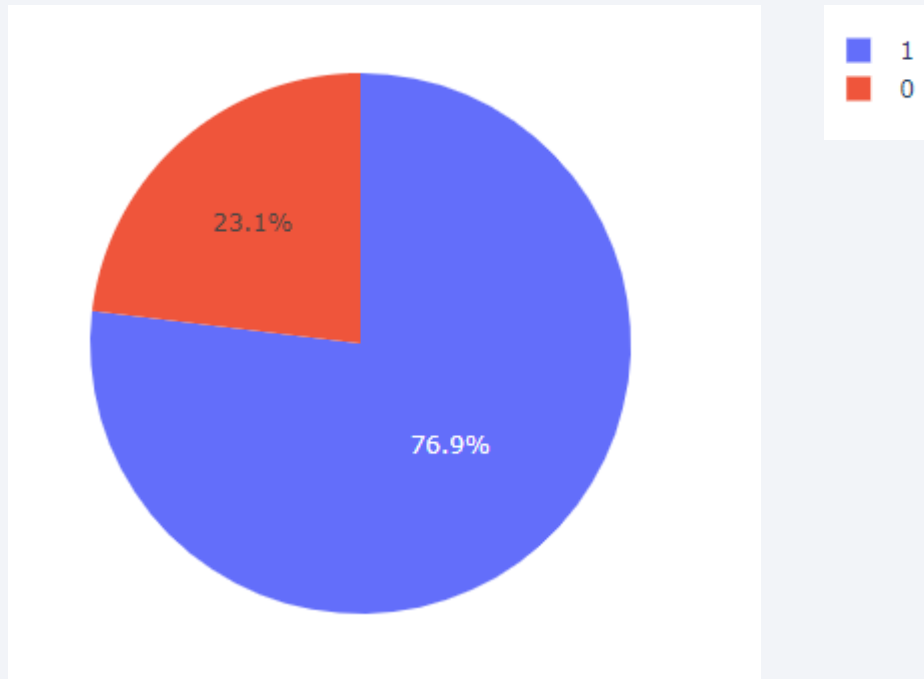


Observations

- This graph depicts really well how each site compares to each other regarding the total launches;
- KSC-LC has almost 50% off all the success landings, **more than double** of VAFB-SLC and CCAFS-SLC.

Pie chart with Dash App – The most successful site

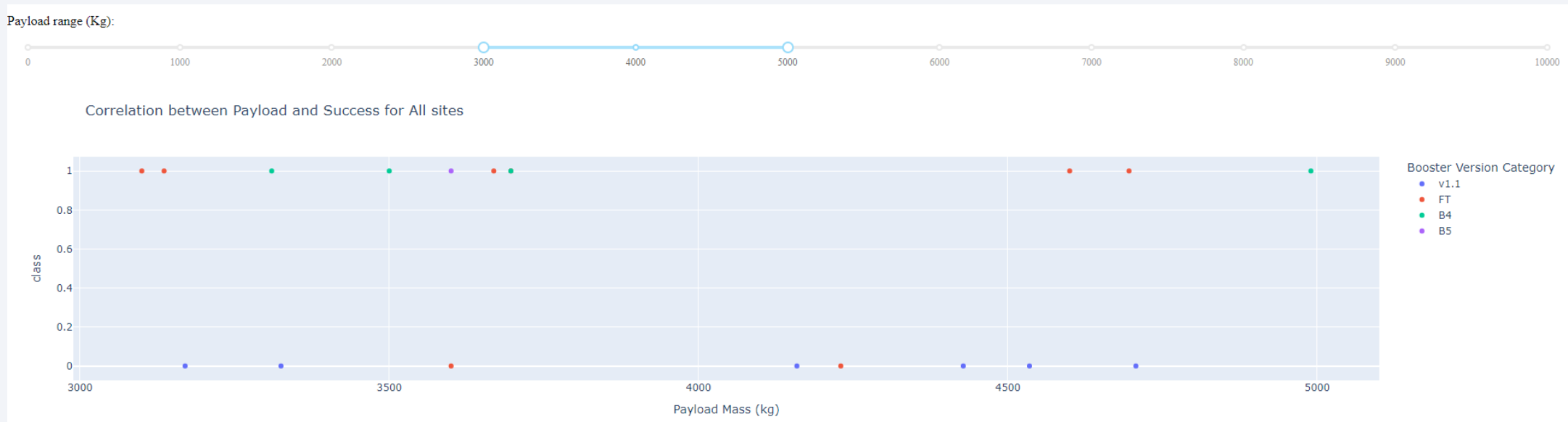
Total Success Launches for site KSC - LC



Observations

- KSC-LC has the higher success rate of all the sites;
- This correlates with the previous graph since this site had also the highest number of success rate.

Scatter Plot with Dash App – Flexible range



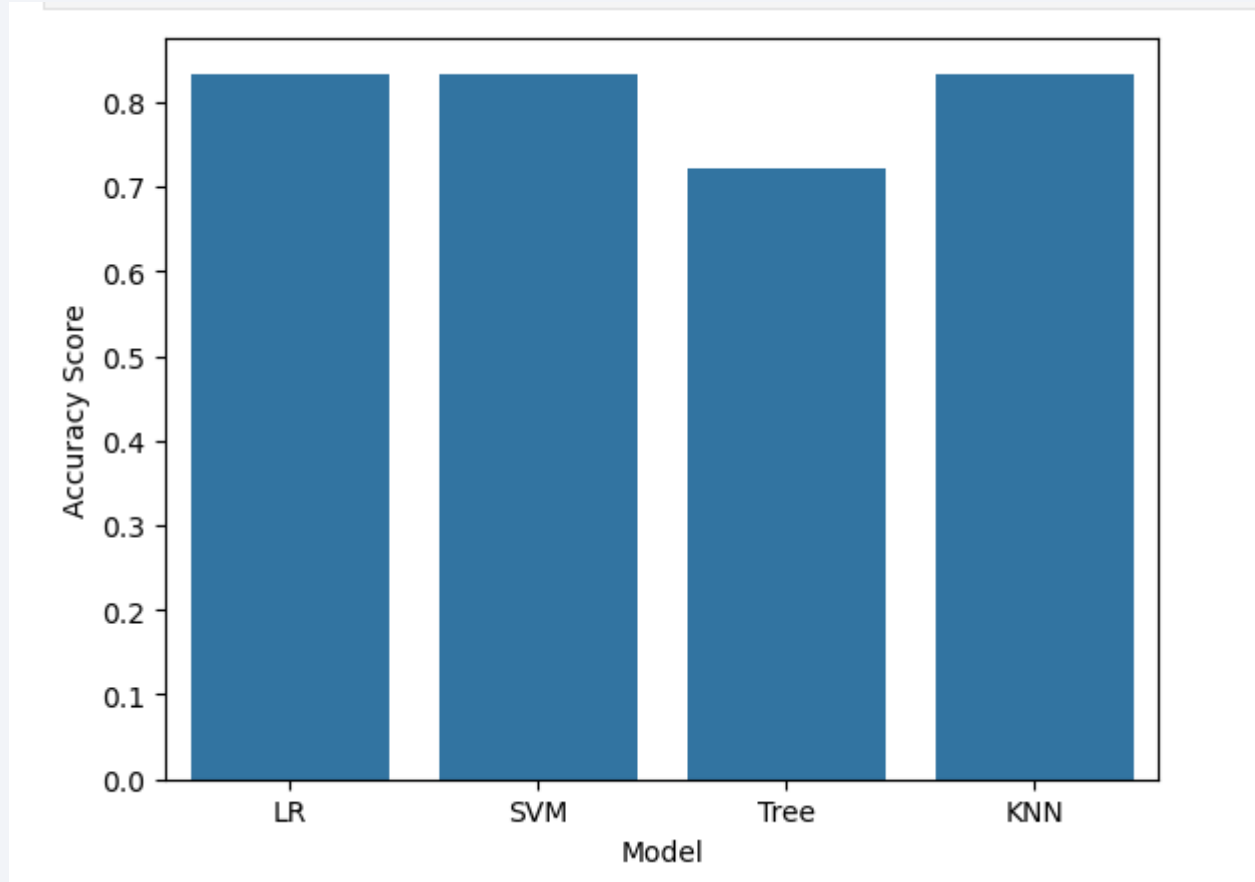
Observations

- The bar allows for the x axis to be interchangeable, with a range from 0 to 1000 kg on the payload.
- It is important the choice of the booster depending on the payload!

Section 5

Predictive Analysis (Classification)

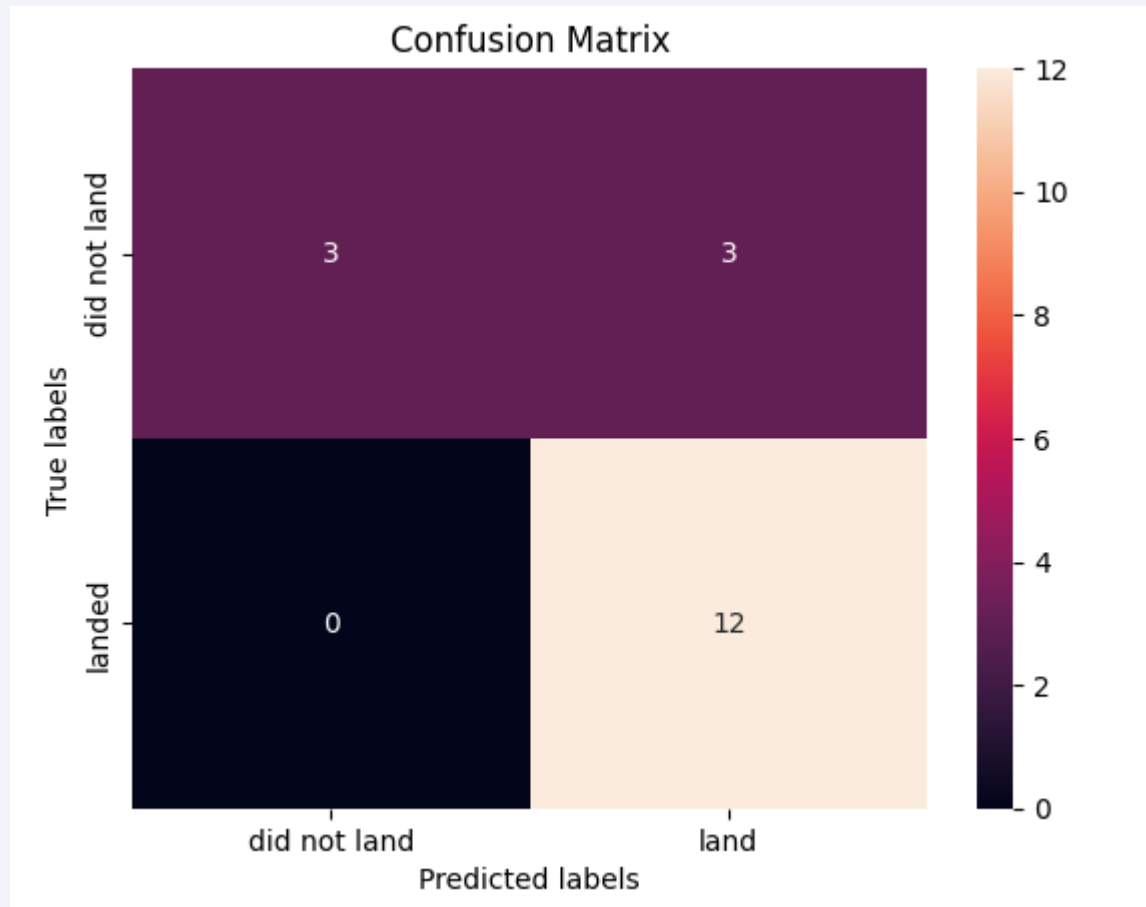
Classification Accuracy



Observations

- **Tree Decision** is the worst model to predict a successful landing;
- **LR, SVM and KNN** are equivalent as predictors
- They have an accuracy score of **83.3%**

Confusion Matrix



Observations

- This confusion matrix belongs to either **LR**, **SVM** or **KNN**;
- The problem of these models are the **False Positives**, i.e. the model predicts a land incorrectly (3).

Conclusions

- 1st stage landing successful rate has been increasing with number of flights
- **Orbit, payload masses and booster version** are important parameters to consider for predicting the succesful rate
- To predict a future launch, **LR, SVM or KNN** seems to be the best predictor
- We can predict a successful landing of the 1st stage with an accuracy of **83.3%**
- However, the confusion matrix shows a **major obstacle with False Positives detection**

Appendix

- None

Thank you!

