IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Christian Fiedler
10.01.2022

# Outline

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

# Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - EDA with data visualization
  - EDA with SQL
  - Building an interactive map with Folium
  - Building a Dashboard with Plotly Dash
  - Predictive analysis (Classification)
- Summary of all results
  - Exploratory data analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

# Introduction

- Project background and context

  We predicted if the Falcon 9 1st stage will land successfully. 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 1st stage. Therefore, if we can determine if the 1st stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

- Problems you want to find answers

  - Influence parameter to a successful land of the rocket.
  - The effect each relationship with certain rocket variables will impact in determining the success rate of a successful landing.
  - What conditions does SpaceX have to achieve to get the best results and ensure the best rocket success landing rate.

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection methodology:

  SpaceX Rest API

  Web Scrapping from Wikipedia SpaceX entry

- Perform data wrangling

  One Hot Encoding data fields for Machine Learning and cleaning

- Perform exploratory data analysis using visualization and SQL

  Plotting : Scatter Graphs, Bar Graphs t(EDA) o show relationships between variables to show patterns of data

- Perform interactive visual analytics using Folium and Plotly Dash

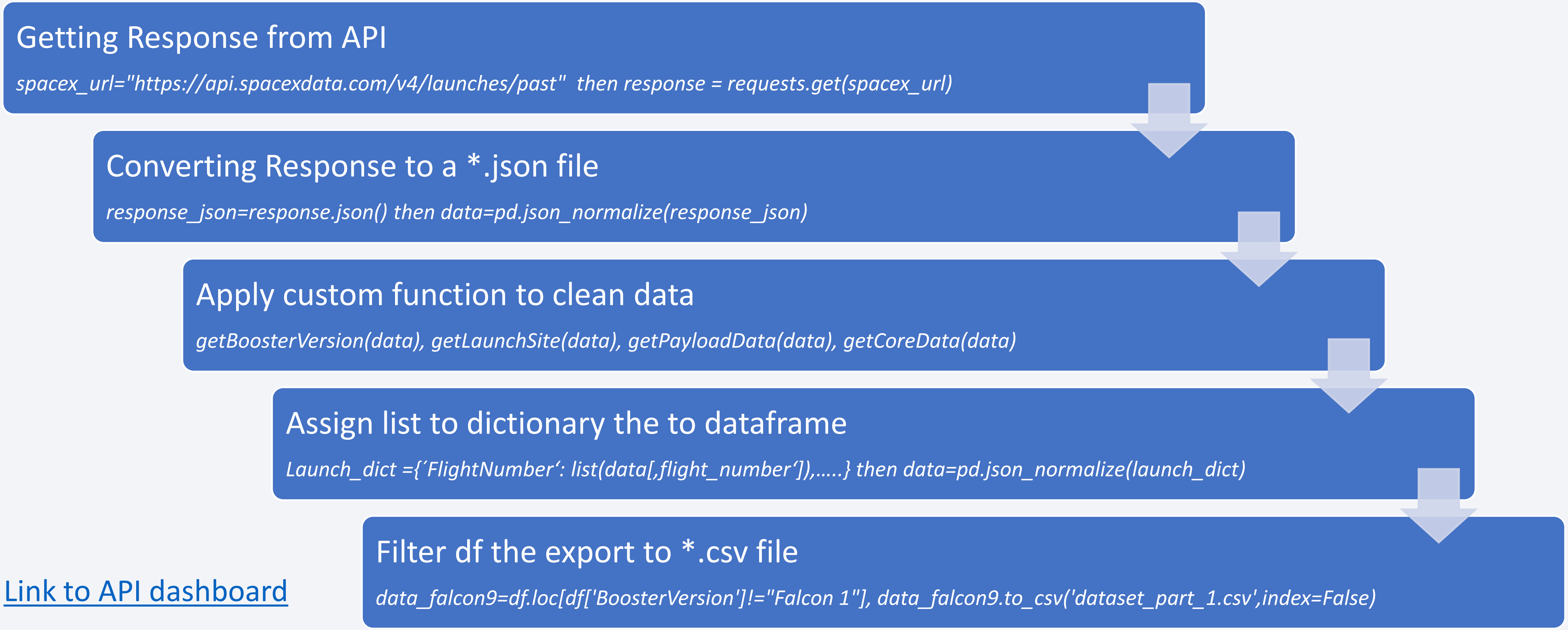
- Perform predictive analysis using classification models

  How to build, tune, evaluate classification models

# Data Collection from REST API and HTML

- The following datasets was collected by
  - SpaceX launch data that is gathered from the SpaceX REST API.
  - API will give us data about launches, payload delivered, launch specifications, landing specifications, and landing outcome
  - Our goal is to use this data to predict whether SpaceX will attempt to land the main booster or not.
  - SpaceX REST API endpoints or URL starts with api.spacexdata.com/v4/
  - Another popular data source for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"

# Data Collection – SpaceX API



Getting Response from API

*spacex_url="https://api.spacexdata.com/v4/launches/past"  then response = requests.get(spacex_url)*

Converting Response to a *.json file

*response_json=response.json() then data=pd.json_normalize(response_json)*

Apply custom function to clean data

*getBoosterVersion(data), getLaunchSite(data), getPayloadData(data), getCoreData(data)*

Assign list to dictionary the to dataframe

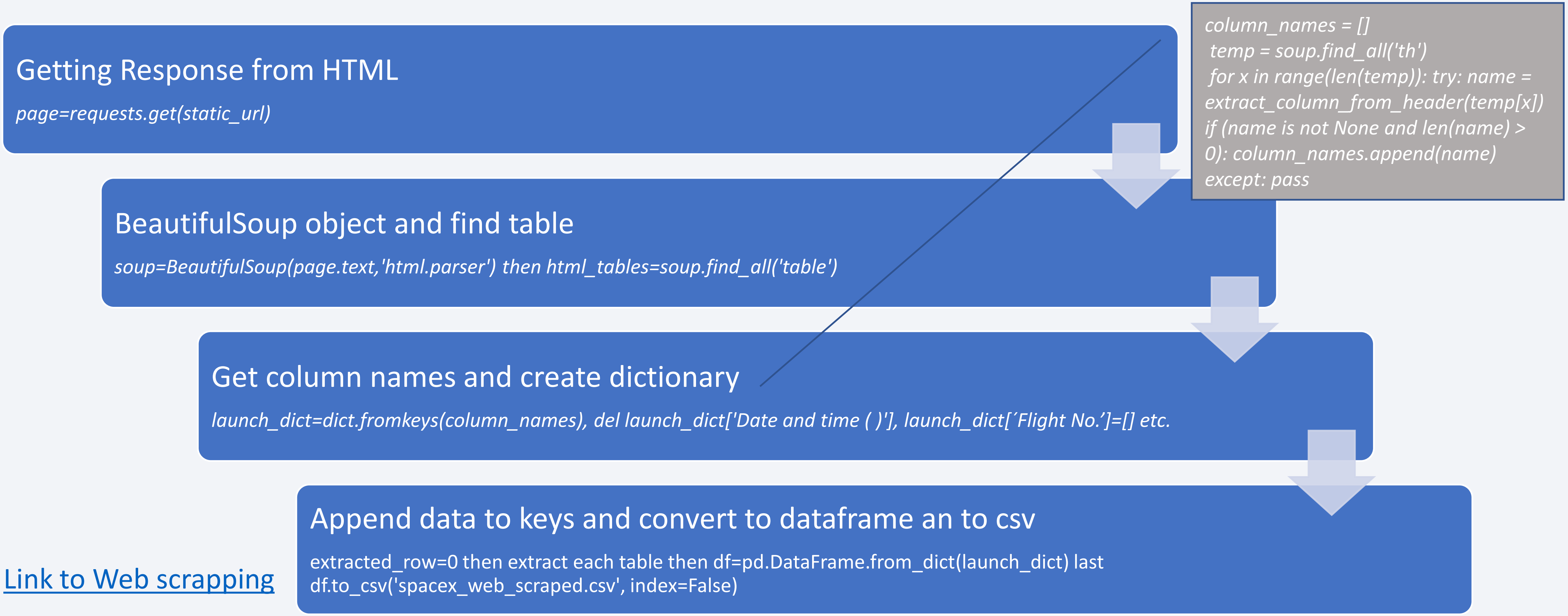
*Launch_dict ={´FlightNumber‘: list(data[,flight_number‘]),…..} then data=pd.json_normalize(launch_dict)*

Filter df the export to *.csv file

*data_falcon9=df.loc[df['BoosterVersion']!="Falcon 1"], data_falcon9.to_csv('dataset_part_1.csv',index=False)*

[Link to API dashboard]()

# Data Collection – Web Scraping



```
column_names = []
 temp = soup.find_all('th')
 for x in range(len(temp)): try: name =
extract_column_from_header(temp[x])
if (name is not None and len(name) >
0): column_names.append(name)
except: pass
```

## Getting Response from HTML

*page=requests.get(static_url)*

## BeautifulSoup object and find table

*soup=BeautifulSoup(page.text,'html.parser') then html_tables=soup.find_all('table')*

## Get column names and create dictionary

*launch_dict=dict.fromkeys(column_names), del launch_dict['Date and time ( )'], launch_dict[´Flight No.’]=[] etc.*

## Append data to keys and convert to dataframe an to csv

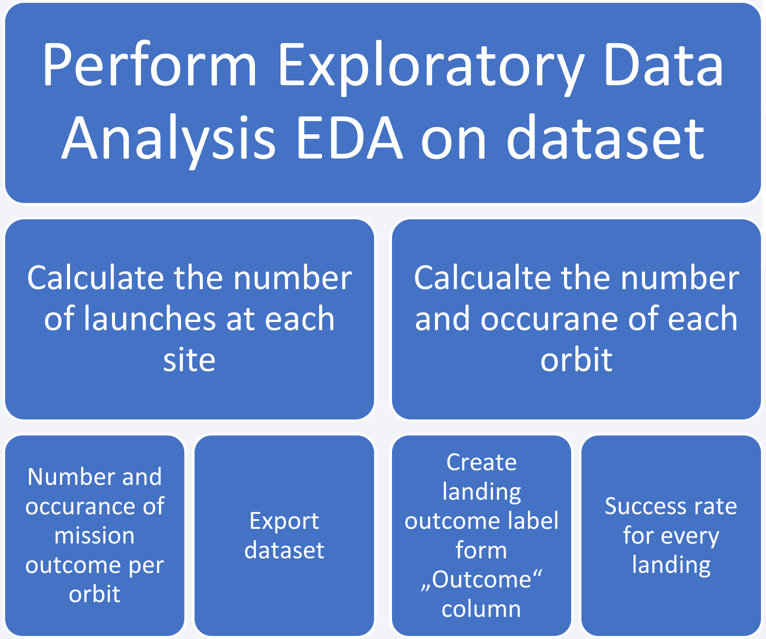
extracted_row=0 then extract each table then df=pd.DataFrame.from_dict(launch_dict) last df.to_csv('spacex_web_scraped.csv', index=False)

[Link to Web scrapping]

# Data Wrangling

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship. We mainly convert those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.

[Link to Data Wrangling notebook]



Perform Exploratory Data Analysis EDA on dataset

Calculate the number of launches at each site

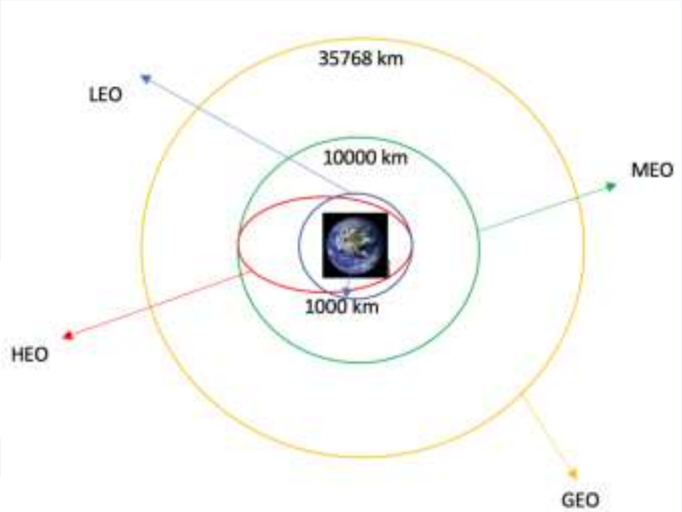
Calcualte the number and occurance of each orbit

Number and occurance of mission outcome per orbit

Export dataset

Create landing outcome label form „Outcome“ column

Success rate for every landing



# EDA with Data Visualization

## Scatter Graphs being drawn:

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload vs. Launch Site
- Orbit vs. Flight Number
- Payload vs. Orbit Type
- Orbit vs. Payload Mass

Scatter plots show how much one variable is affected by another. The relationship between two variables is the correlation.

## Bar Graph being drawn:

Mean vs. Orbit

A bar diagram makes it easy to compare sets of data between different groups at a glance. The graph represents categories on one axis and a discrete value in the other. The goal is to show the relationship between the two axes.

## Line Graph being drawn:

Success Rate vs. Year

Line graphs are useful in that they show data variables and trends very clearly and can help to make predictions about the results of data not yet recorded

Link to Exploring Data

# EDA with SQL

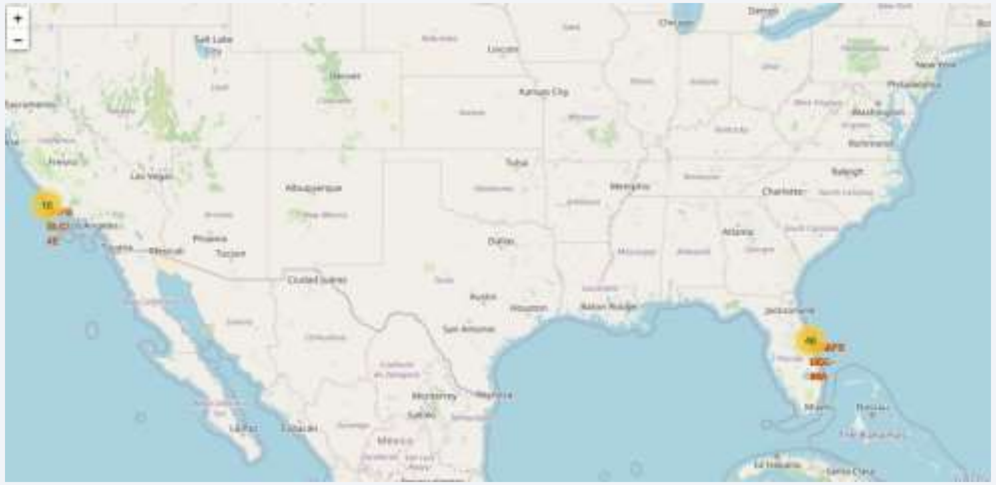
For example of some questions we were asked about the data we needed information about. Which we are using SQL queries to get the answers in the dataset :

- Displaying the names of the unique launch sites, 5 records 'KSC' launch sites, total payload mass carried by boosters, average payload mass carried by booster version F9 v1.1
- Listing the date where the successful landing, names of boosters which have success in ground pad and have payload mass >4000 kg but < 6000 kg, total number of successful and failure mission outcomes
- Ranking the count of successful landing outcomes between the date 2010-06-04 and 2017-03-20 in descending order.
- Link to SQL notebook

# Build an Interactive Map with Folium

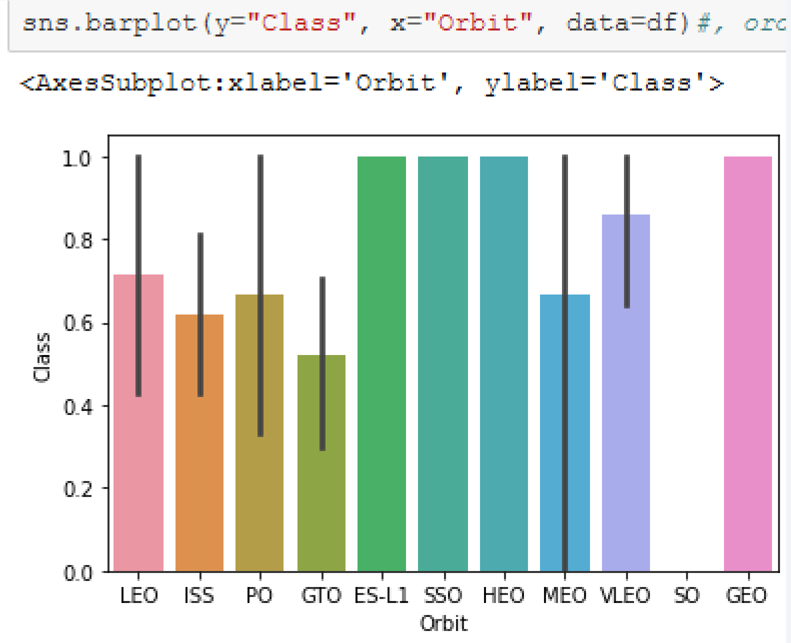
The interactive map is created by the following Folium objects:

- Circle Marker: for launch site with coordinates and label name
- MarkerCluster: if mission succeed with green or failed in red
- Add_child: mouse position to see position for calculate distance to e.g. railways, coastline, highways, city and use folium.Marker and folium.PolyLine
- [Link to interactive map and code]()



# Build a Dashboard with Plotly Dash

- Graphs
  - Bar Chart: displays e.g. the success rate “class” 0 no success, 1 success of each orbit mission
  - Pie Chart showing the total launches by a certain site/all sites, display relative proportions of multiple classes of data, size of the circle can be made proportional to the total quantity it represents.
  - Scatter Graph showing the relationship with Outcome and Payload Mass (Kg) for the different Booster Versions
    - It shows the relationship between two variables.
    - It is the best method to show you a non-linear pattern.
    - The range of data flow, i.e. maximum and minimum value, can be determined.
    - Observation and reading are straightforward
- [Link to graphics and code]()



```
sns.barplot(y="Class", x="Orbit", data=df)#, orc
```

```
<AxesSubplot:xlabel='Orbit', ylabel='Class'>
```

# Predictive Analysis (Classification)

**BUILDING MODEL**

- Load our dataset into NumPy and Pandas
- Transform Data
- Split our data into training and test data sets
- Check how many test samples we have
- Decide which type of machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit our datasets into the GridSearchCV objects and train our dataset

**EVALUATING MODEL**

- Check accuracy for each model
- Get tuned hyperparameters for each type of algorithms
- Plot Confusion Matrix

**IMPROVING MODEL**

- Feature Engineering
- Algorithm Tuning

**FINDING THE BEST PERFORMING CLASSIFICATION MODEL**

The model with the best accuracy score wins the best performing model
In the notebook there is a dictionary of algorithms with scores at the bottom of the notebook.

Link to Predictive Analysis notebook

# Results

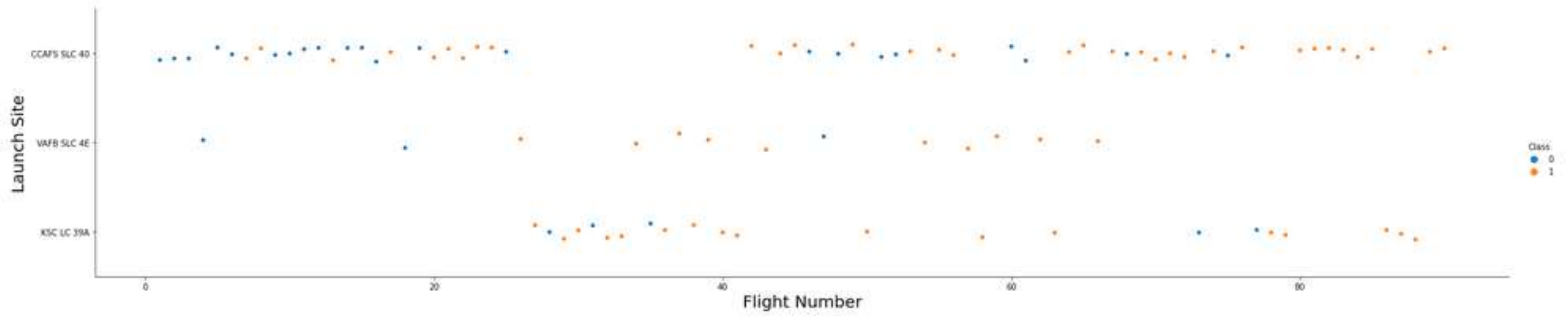
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

# Insights drawn from EDA

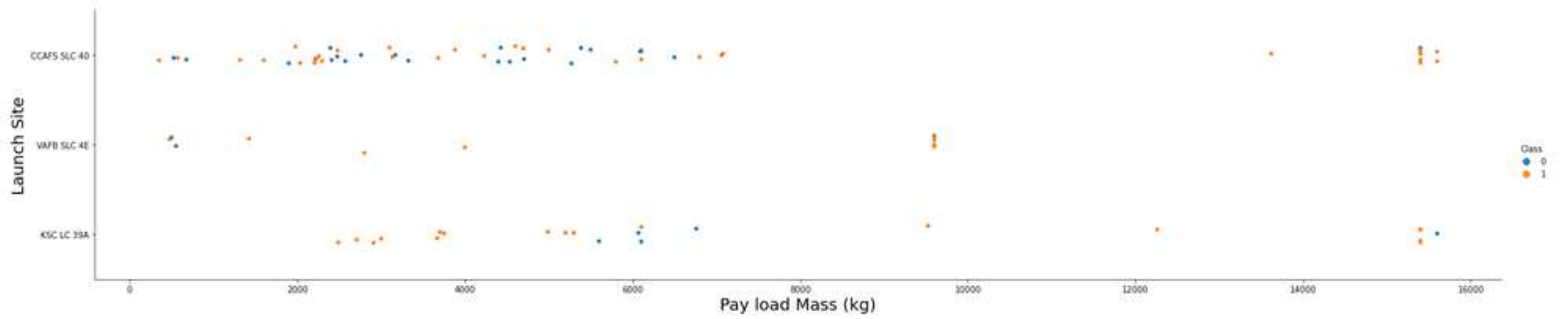
# Flight Number vs. Launch Site

- You can the the relation between flight numbers and the success rate of each launch site. More flights = higher success



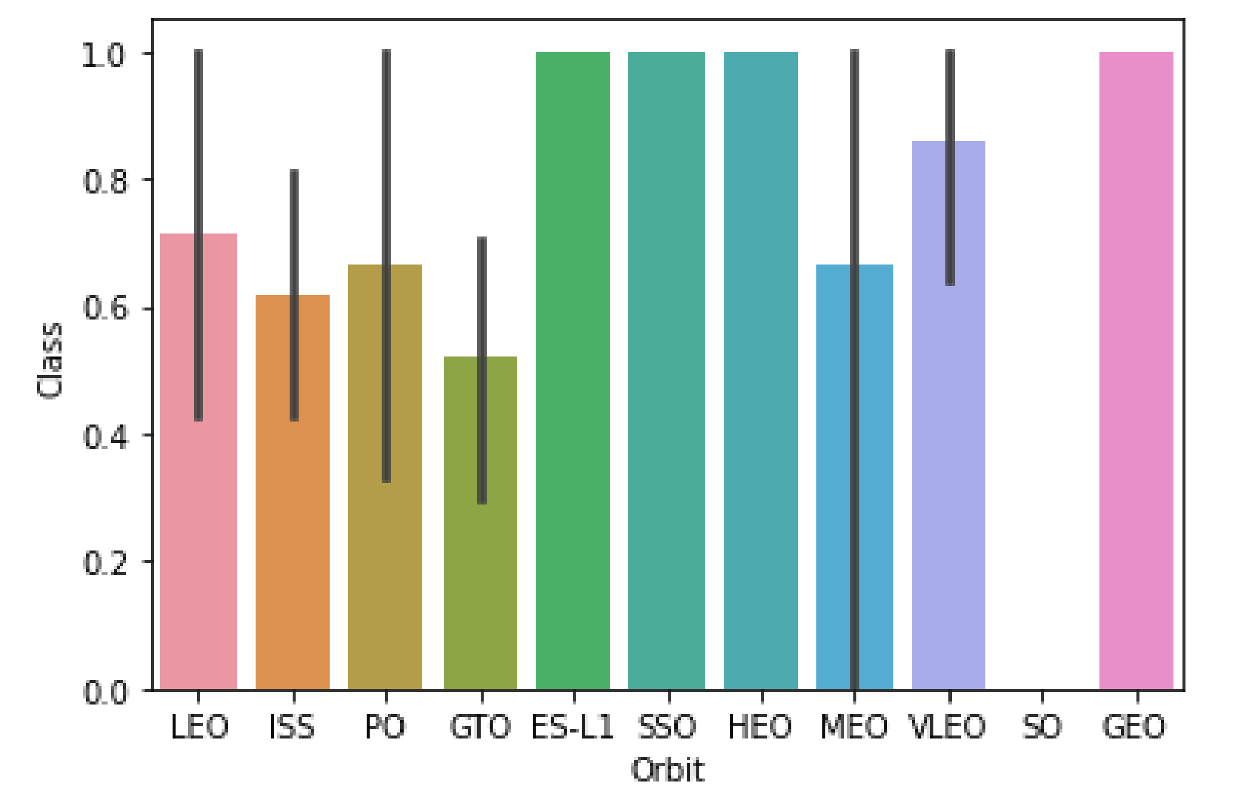
# Payload vs. Launch Site

- The greater the payload mass for Launch Site CCAFS SLC 40 the higher the success rate for the Rocket.
  There is not quite a clear pattern to be found using this visualization to make a decision if the Launch Site is dependent on Pay Load Mass for a success launch.



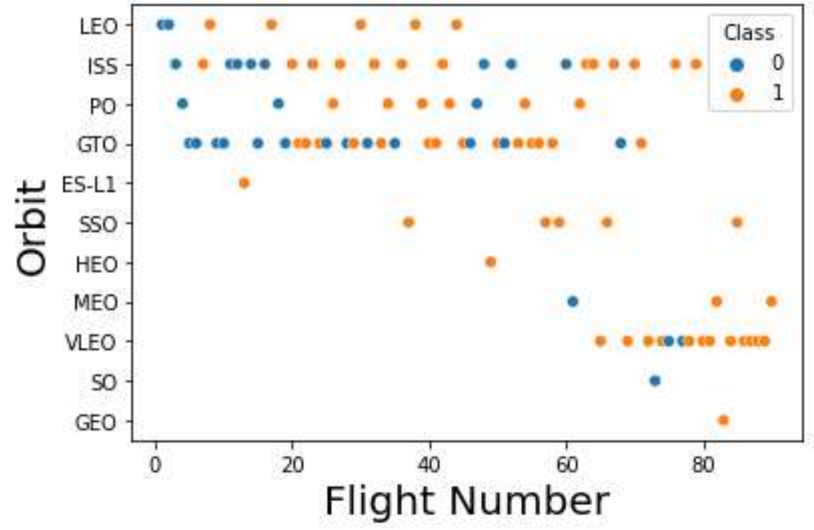
# Success Rate vs. Orbit Type

- Class 1.0 means 100 % success rate, here Orbit:
  - GEO
  - HEO
  - SSO
  - ES-L1



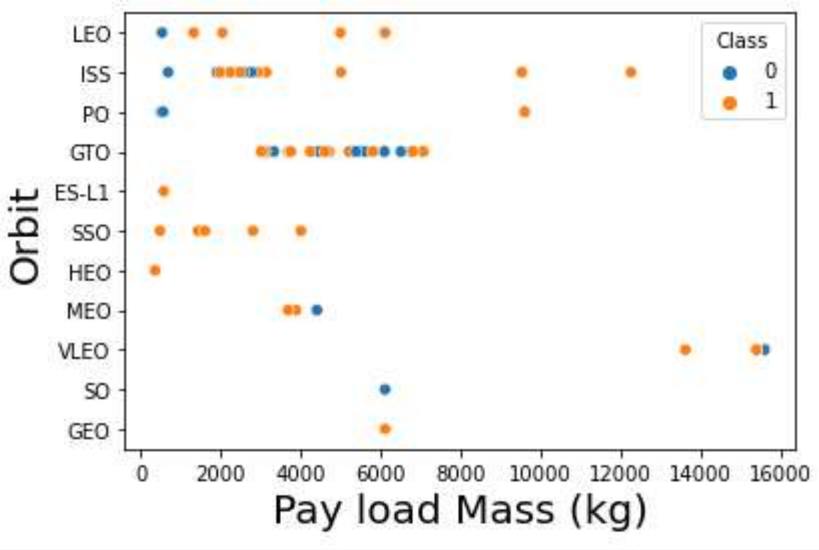
# Flight Number vs. Orbit Type

- LEO shows increasing success to increasing flight numbers
- For GTO there is not relationship between the variabels



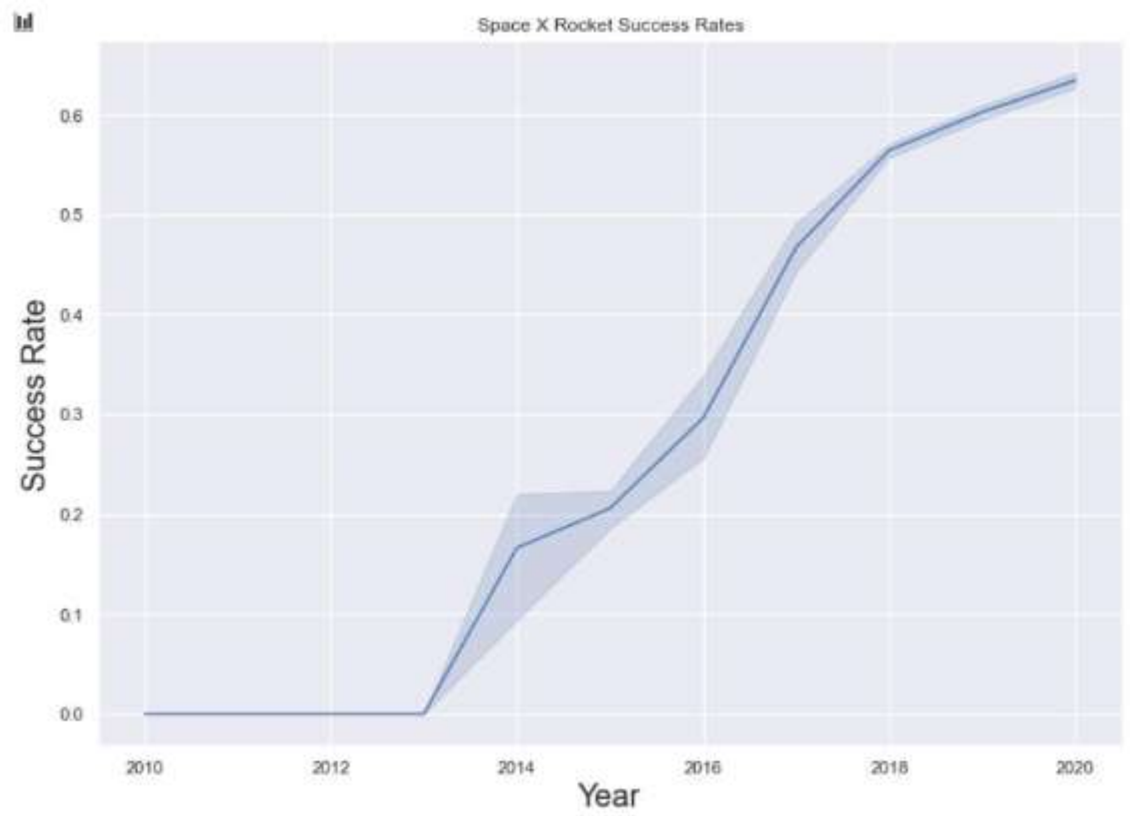
# Payload vs. Orbit Type

- In general high pay load mass has a negative impact for the success rate
- Exceptions are GTO and ISS



# Launch Success Yearly Trend

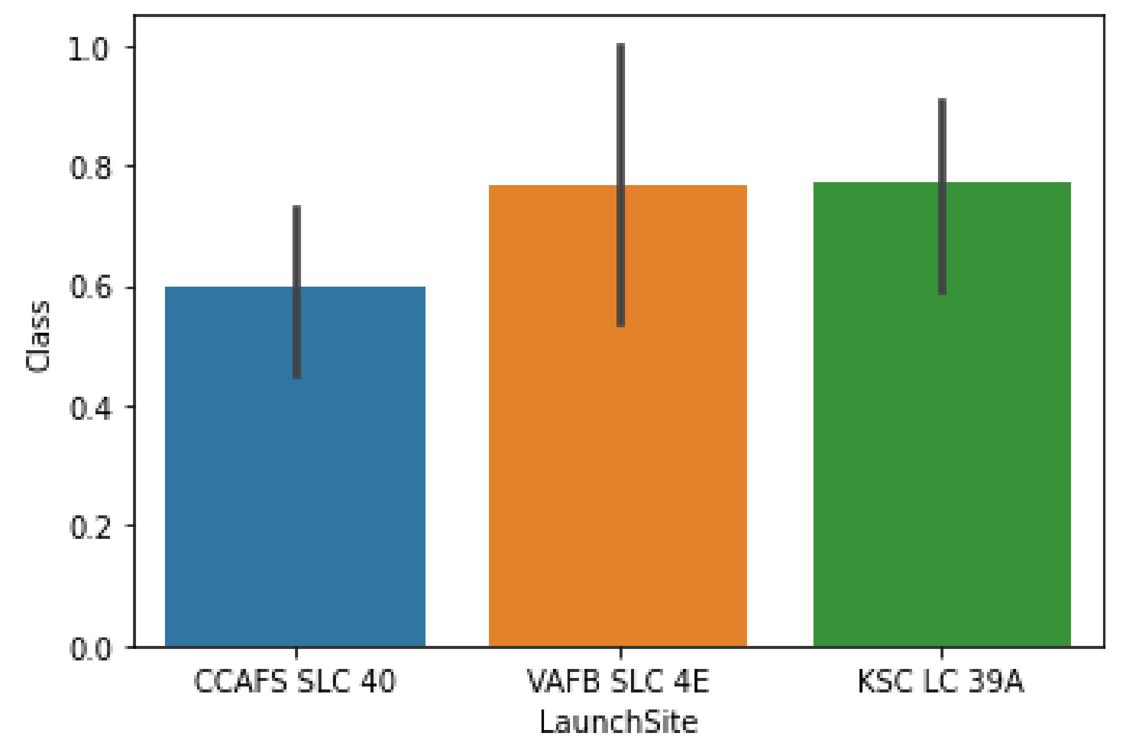
- The chart shows a strong increasing successrate from 2013 on



# All Launch Site Names

In this Bar Chart you can see the different launch sites with the success rate of the missions.

- Space Launch Complex 40 (SLC-40) Cape Canaveral Space Force Station
- Space Launch Complex 4E (VAFB SLC-4E) of Vandenberg Space Force Base in California
- Launch Complex 39A (KSC LC-39A) nearby Kennedy Space Centers



# Launch Site Names Begin with 'CCA'

- *Select TOP 5 * from tblSpaceX WHERE Launch_site LIKE ‘CCA%’*

→ creates an output of all Launch sites which starts with CCA

# Total Payload Mass

- *select SUM(PAYLOAD_MASS_KG_) TotalPayloadMass from tblSpaceX*

→Use the SUM function to the column PAYLOAD_MASS_KG

# Average Payload Mass by F9 v1.1

- *select AVG(PAYLOAD_MASS_KG_) AveragePayloadMass from tblSpaceX*

*where Booster_Version = 'F9 v1.1‘*

*→ Use AVG function to the column PAYLOAD_MASS_KG_ with WHERE function to column Booster_Version wih “F9 v1.1”*

# First Successful Ground Landing Date

- *select MIN(Date) SLO from tblSpaceX*

→ *Use MIN function to the column Date*

# Successful Drone Ship Landing with Payload from 4t to 6t

- *Select Booster_Version from tblSpaceX where Landing_Outcome = 'Success (ground pad)’ AND Payload_MASS_KG_ > 4000 AND Payload_MASS_KG_ < 6000*
- Select Boodster_Version with WHERE function to filter on column Landing_Outcome Success (drone ship) and use filter conditions with column Payload_Mass_KG_

# Total Number of Successful and Failure Mission Outcomes

- *SELECT(SELECT Count(Mission_Outcome) from tblSpaceX where Mission_Outcome LIKE '%Success%’ as Successful_Mission_Outcomes,*
- *(SELECT Count(Mission_Outcome) from tblSpaceX where Mission_Outcome LIKE '%Failure%’ as Failure_Mission_Outcomes*

→Two queries in one table with function Count and Like to count the succeeded and fails mission, word success or failure is in the data set

# Boosters Carried Maximum Payload

- SELECT DISTINCT Booster_Version, MAX(PAYLOAD_MASS_KG_) AS [Maximum Payload Mass] FROM tblSpaceX GROUP BY Booster_Version ORDER BY [Maximum Payload Mass] DESC

→ Using the word DISTINCT in the query means that it will only show Unique values in the Booster_Version column from tblSpaceX GROUP BY puts the list in order set to a certain condition. DESC means its arranging the dataset into descending order

# 2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- SELECT DATENAME(month, DATEADD(month,MONTH(CONVERT(date, Date, 105)), 0) - 1) AS Month, Booster_Version, Launch_Site, Landing_Outcome FROM tblSpaceX WHERE (Landing_Outcome LIKE '%Failure (drone ship)') AND (YEAR(CONVERT(date, Date, 105)) = '2015')

→ Creates table for each month 2015 with function DATENAME, DATEADD, CONVERT then filtered with WHERE function the columns Landing_Outcome  and YEAR

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

- *SELECT COUNT(Landing_Outcome) AS [Cnt_Landing_Outcome] FROM tblSpaceX WHERE (Landing_Outcome LIKE '%Failure (drone ship)‘ ) OR WHERE (Landing_Outcome LIKE '%Success (ground pad)’) AND (Date > '04-06-2010’) AND (Date < '20-03-2017') ORDER BY [Cnt_Landing_Outcome]  DESC*

→Use function GROUP BY to put the list in order set to a certain condition.

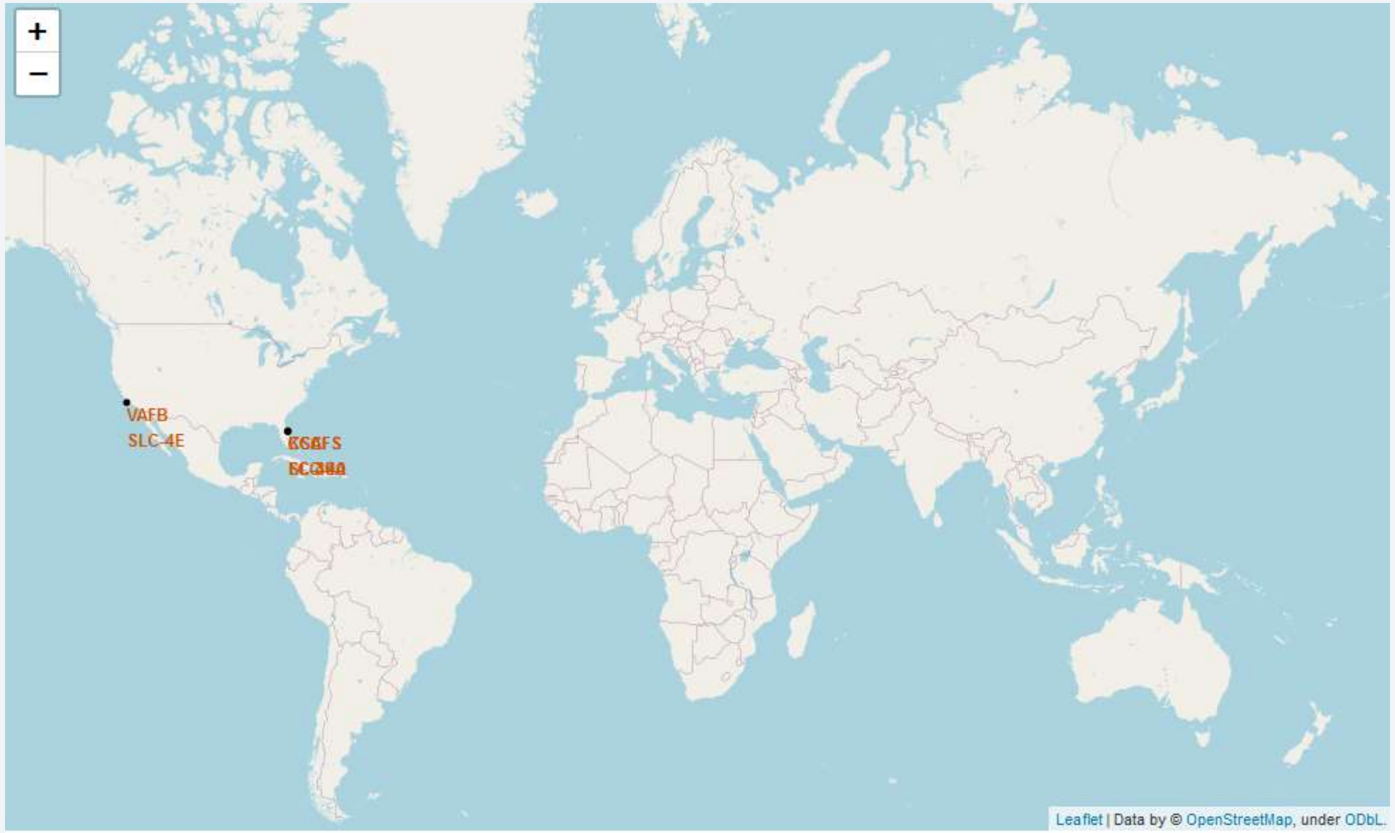
→Use function DESC to arrange the dataset into descending order

Section 4

# Launch Sites Proximities Analysis

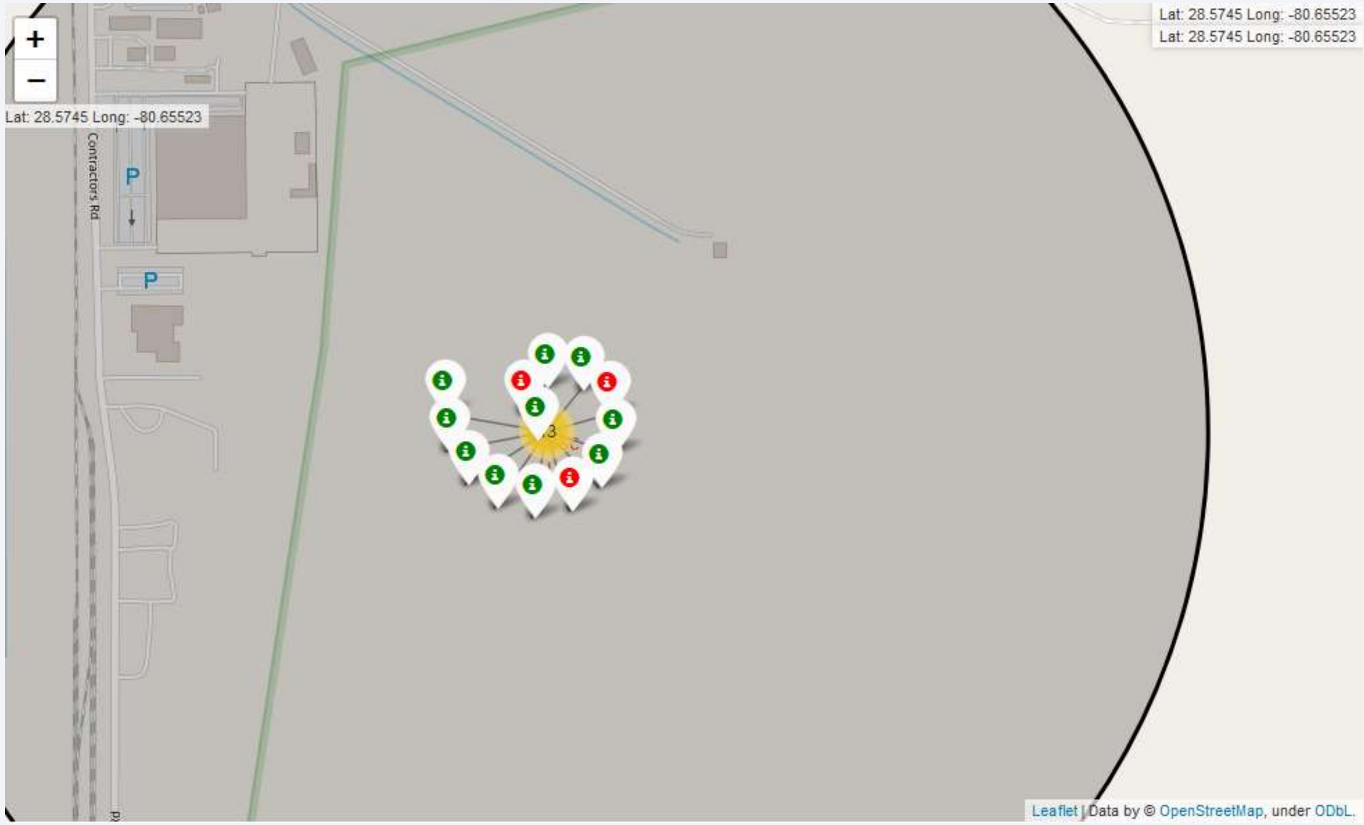
# Map with Folium SpaceX launch site

- SpaceX launch sites are located at the coasts of Florida and California USA.



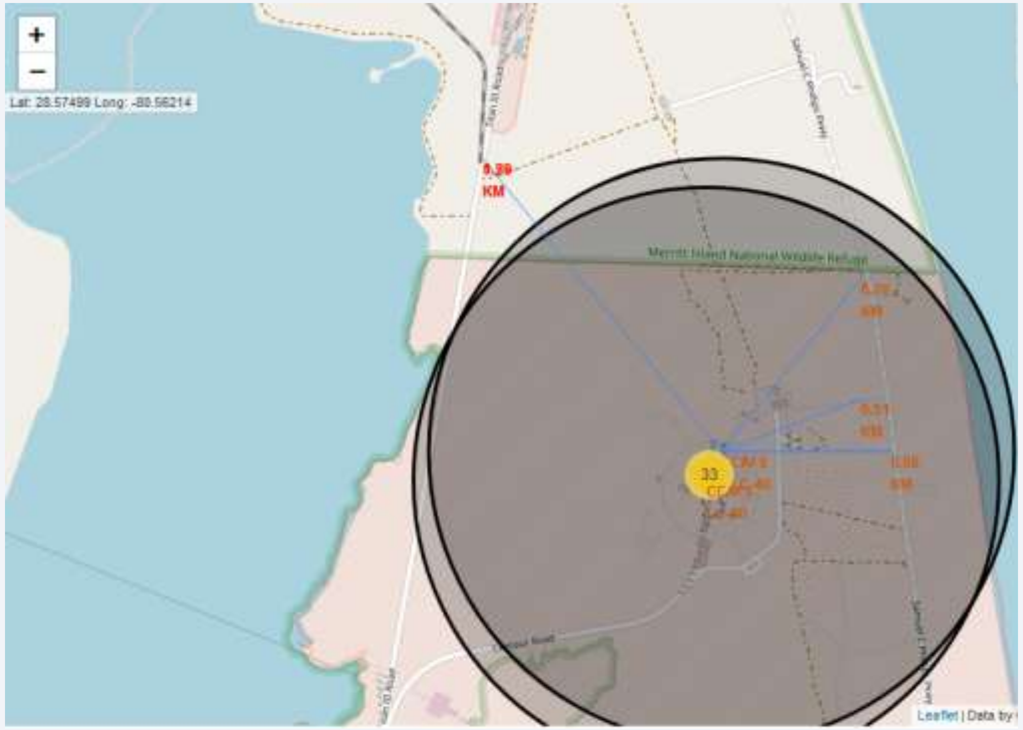
# Successful launches for launch site KSC

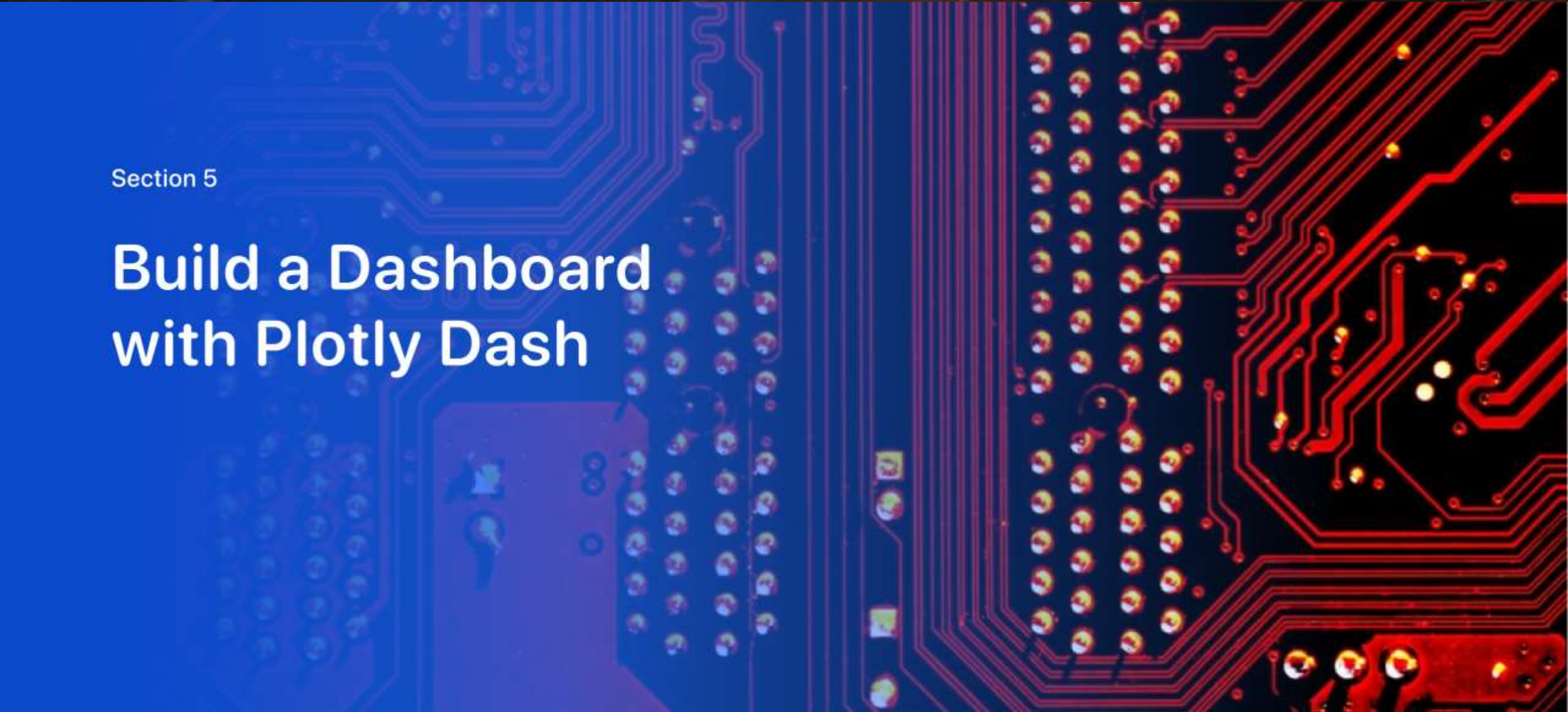
- E.g. KSC LC-39A green marker = successful launch, red marker = failed launch


+
−
Lat: 28.5745 Long: -80.65523
Lat: 28.5745 Long: -80.65523
Lat: 28.5745 Long: -80.65523
Contractors Rd
P
P
13
Leaflet | Data by © OpenStreetMap, under ODbL.

# Launch site CCAFS LC-40 infrastructure

- Due to minimize the risk the launch site CCAFS LC-40 is located close to the coast no big cities are nearby but it has a good infrastructure like railways and highways nearby.


+
−
Lat: 28.57499 Long: -80.56214
9.29
KM
Merritt Island National Wildlife Refuge
33
Leaflet | Data by

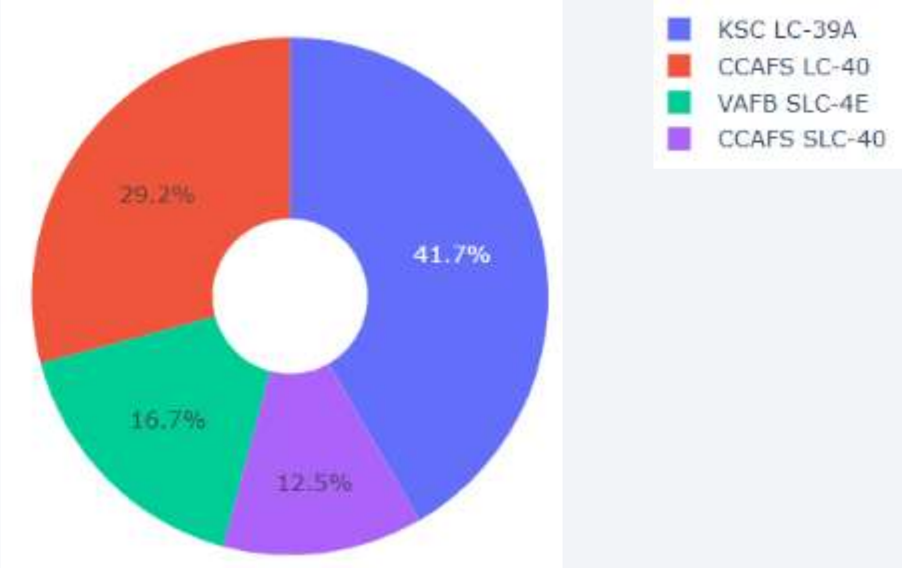


Section 5

# Build a Dashboard with Plotly Dash

# Successful launches

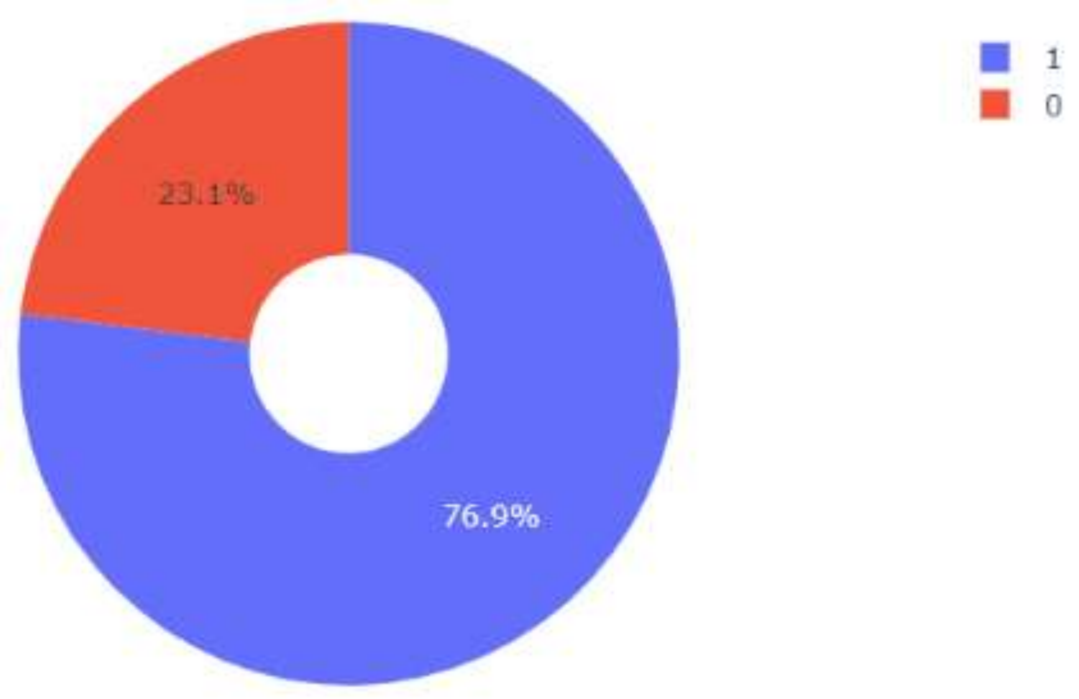
Successful launches by launch sites



most successful launch site: KSC

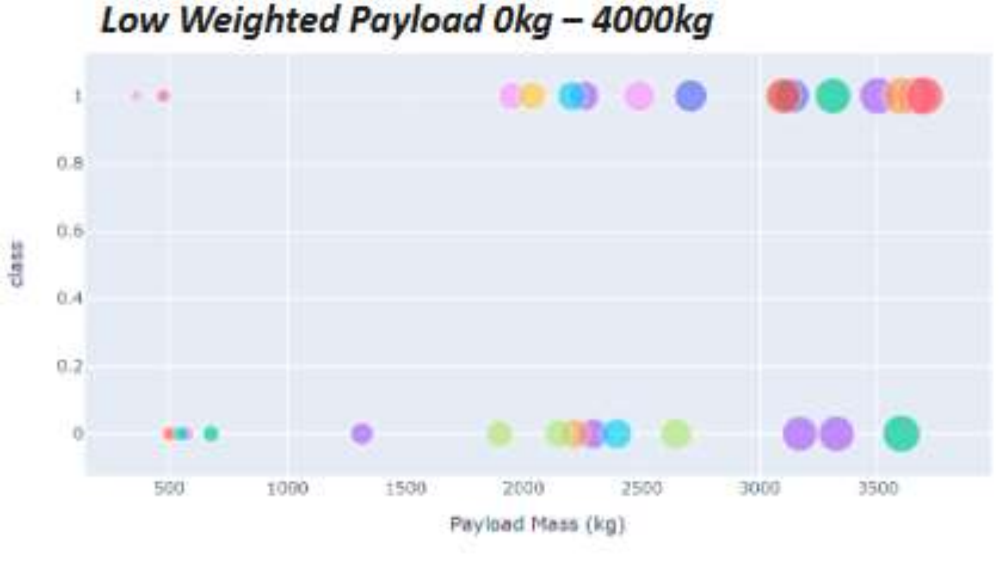
# Launch site with the highest success ratio

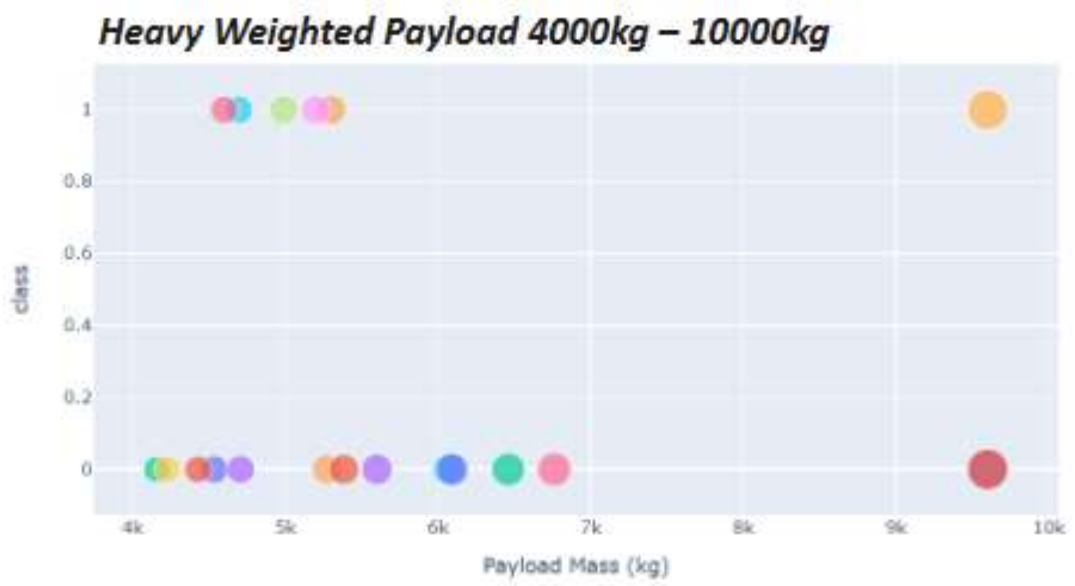
- KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate



# Influence of payload to success of mission

- This dashboard shows the influence of payload to the success of the mission. Here you can see that the class=success rate is higher for lower payload.



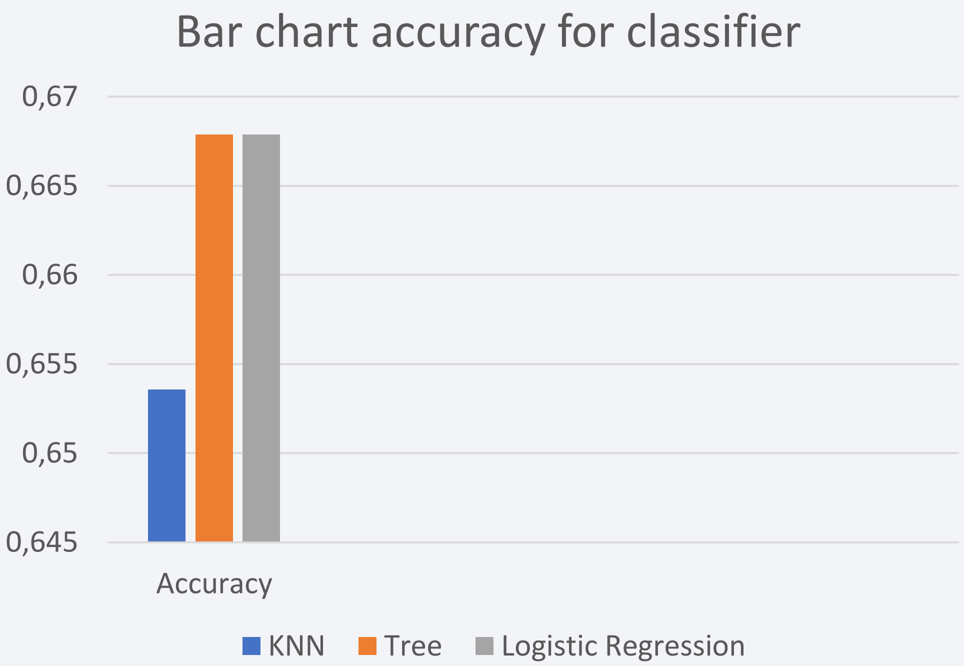


Section 6

# Predictive Analysis (Classification)

# Classification Accuracy

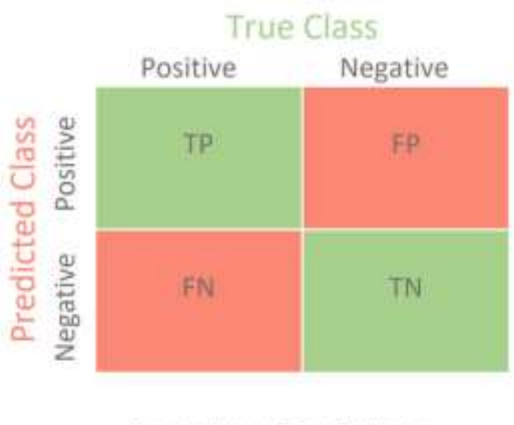
- The best classifier is the Tree classifier only by decimal in front of logistic regression.
- With the hyperparameter tuning the accuracy is at 83.3%.



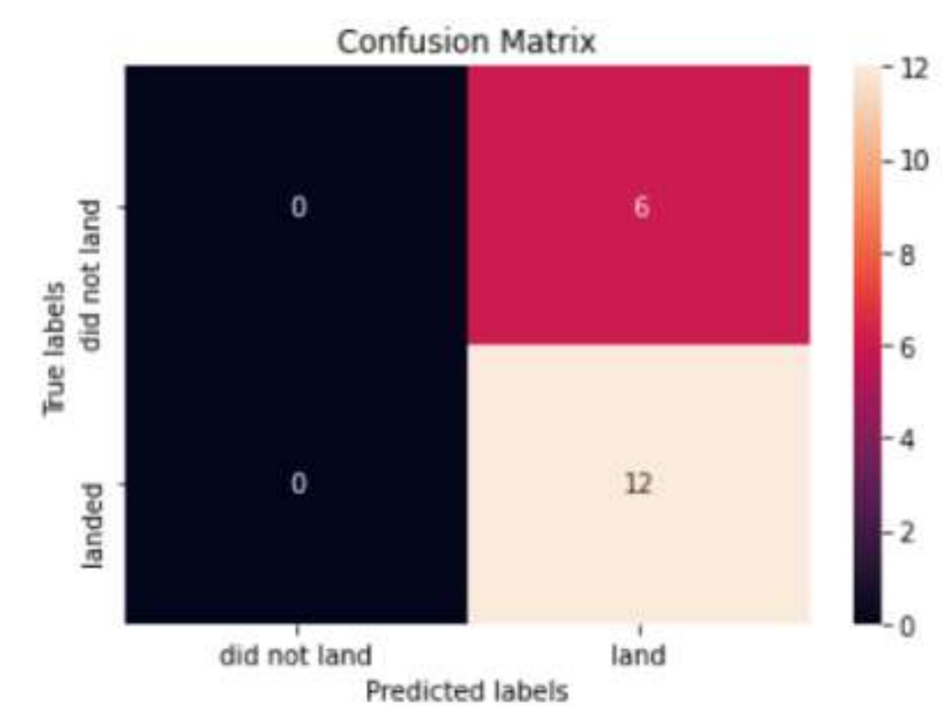
Bar chart accuracy for classifier

# Confusion Matrix for Tree Classifier

- The tree classifier can distinguish between different classes. The true labels and the predicted labels should be consistent. The false positive issue here could be solved by a look at the test data if they are enough land and didn’t land data.
- Here the principle:



Confusion Matrix for Binary Classification



# Conclusions

- The Tree Classifier Algorithm is the best for Machine Learning for this dataset
- Low weighted payloads perform better than the heavier payloads
- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches
- We can see that KSC LC-39A had the most successful launches from all the sites
- Orbit GEO,HEO,SSO,ES-L1 has the best success rate

# Appendix

- No relevant appendix, all for complete this tasks was given to me

Thank you!