Winning Space Race with Data Science

Rinu Jose
12/11/2025

Outline

Executive Summary

- Summary of methodologies
  - Data collection.
  - Data wrangling.
  - EDA with data visualization and sql.
  - Building an interactive map with folium.
  - Building a dashboard with plotly dash.
  - Predictive analysis (correlation).

- Summary of all results
  - Exploratory data analysis results.
  - Interactive analytics screenshots.
  - Predictive analytics results.

Introduction

- Problems you want to find answers
  - We predicted if the Falcon 9 first 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 first stage. Therefore, if we can determine if the first 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.

- Project background and context
  - What influences if the rocket will land successfully?

  The effect each relationship with certain rocket variables will impact 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
- Perform data wrangling
  - One hot encoding data fields for machine learning and dropping irrelevant columns
- 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

REST API

Space X

Json

DataFrame

WEB SCRAPPING

WIKI

HTML

DataFrame

Data Collection – SpaceX API

- import requests
  import pandas as pd
  spacex_url =
  "https://api.spacexdata.com/v4/launches/past"
  response = requests.get(spacex_url)
  data =
  pd.json_normalize(response.json())

- GitHub URL of the completed SpaceX API calls notebook: https://github.com/M02men311/data-science-capstone

Data Collection - Scraping

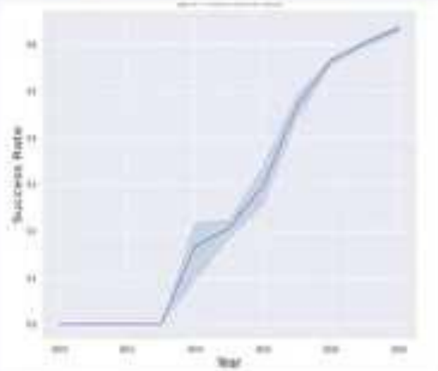
- import requests
  from bs4 import BeautifulSoup
  url =
  'https://en.wikipedia.org/wiki
  /List_of_Falcon_9_and_Falco
  n_Heavy_launches'
  response = requests.get(url)
  html_data=response.text
  soup=BeautifulSoup(html_data)
- https://en.wikipedia.org/wiki/
  List_of_Falcon_9_and_Falcon
  _Heavy_launches

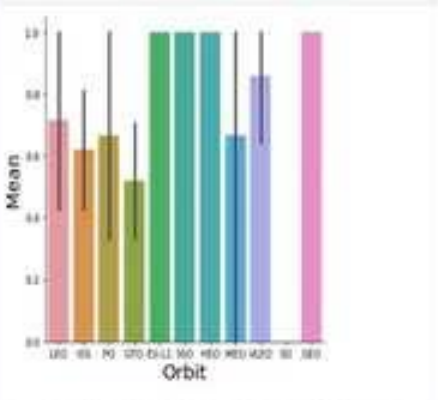
Data Wrangling

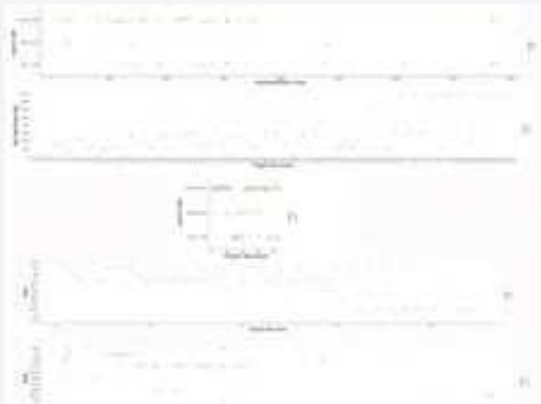
- After collecting the data we check the missing data ,and data types and do on of the following to clean the data :Replace the missing data with one-Using mean or so.Change data type of the data.Represent categorical data using integer or float dummy numbers -one hot encoding

https://github.com/M02men311/data-science-capstone

EDA with Data Visualization







Scatter Plot	To get relationship between variables, e.g.: FlightNumber vs. Orbit type Payload vs. Orbit type FlightNumber vs. PayloadMass FlightNumber vs. Launch Site
Bar Plot	To plot success rate of each orbit
Line Chart	To get the yearly average launch success trend

https://github.com/M02men311/data-science-capstone

EDA with SQL

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'KSC'
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1 Listing the date where the successful landing outcome in drone ship was achieved.
- Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes Listing the names of the booster_versions which have carried the maximum payload mass.
- Listing the records which will display the month names, successful landing_outcomes in ground pad booster versions, launch_site for the months in year 2017 Ranking the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order

```
%sql select distinct Launch_Site
from SPACEXTBL
```

https://github.com/M02men311/data-science-capstone

Build an Interactive Map with Folium

- Folium makes it easy to visualize geo-spatial data in Python using interactive leaflet map. In this project, we use latitudes and longitudes of each launch site with a Circle Marker, Name Label and Cluster Markers for Successful and failed launches on each site
- Since spaceX launches come from different launch sites I displayed the information of failed and successful launches as a cluster on the map.Through zooming in and out you can observer the clusters of success launches and failed launches
- https://github.com/M02men311/data-science-capstone

Build a Dashboard with Plotly Dash

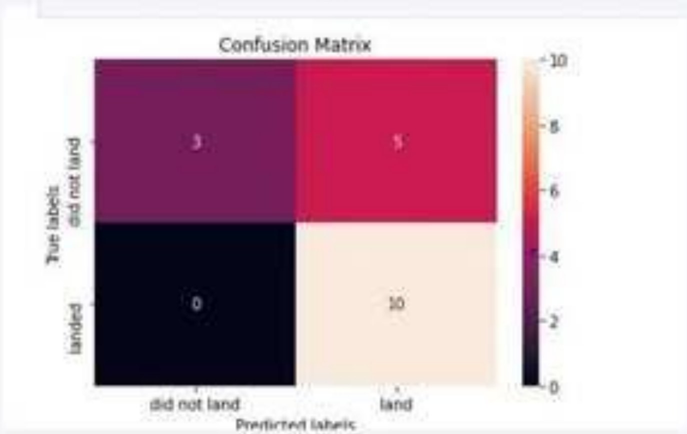
- Dash is a python framework created by plotlyfor creating interactive web applications written on the top of Flask, Plotly.Js and React.Js. In this project, we used IBM's Theia IDE platform to create interactive dashboard with Pie chart and Scatter plots.
- Plotly Dash is Python library that makes it easier to create a dashboard for us as Data Scientist . With a simple interactive dashboard one can change the inputs to see representation of values in graphs.
- https://github.com/M02men311/data-science-capstone

Predictive Analysis (Classification)

- Prepare data
- Create a column for the class
- Standardize the data
- Split into training data and test data
- Define model and parameters
- Train and Grid Search for best parameters
- Evaluation

Results

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

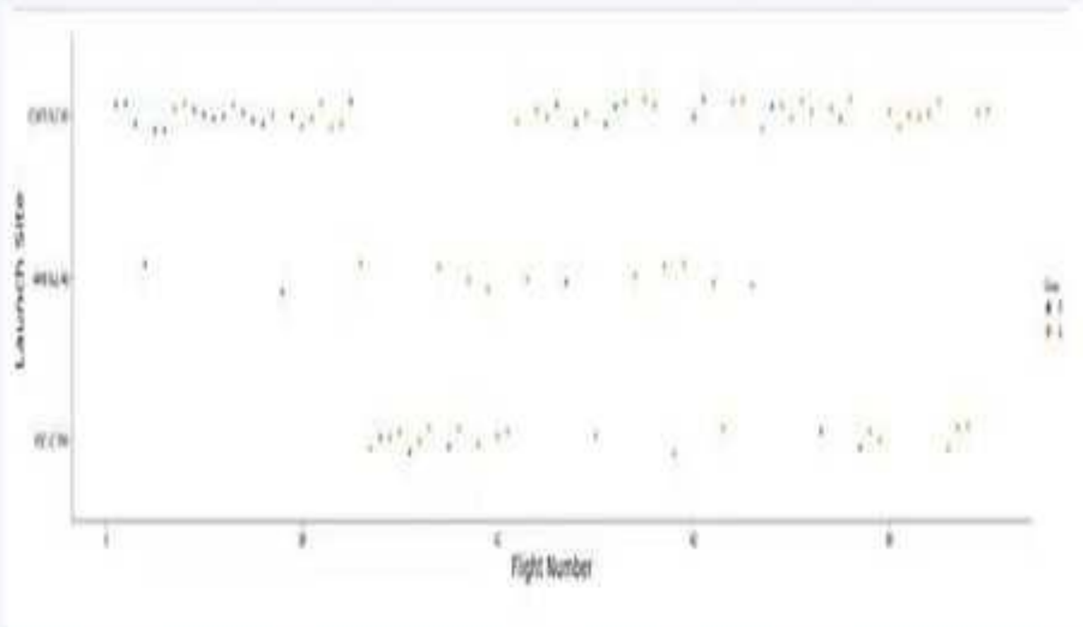


Section 2

Insights drawn from EDA

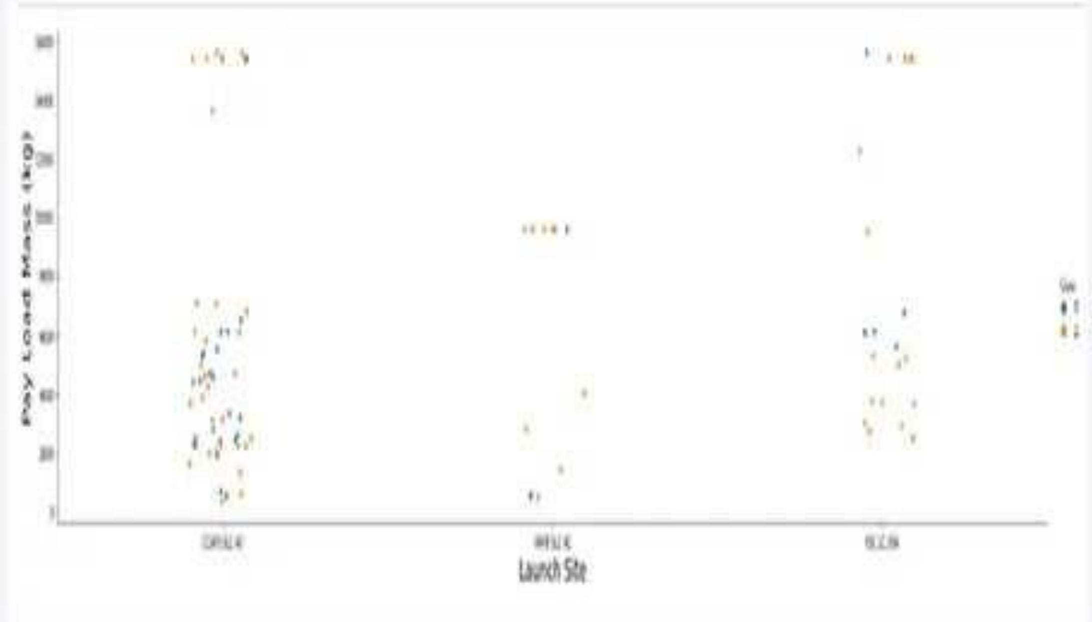
Flight Number vs. Launch Site

- Sns.catplot(y="launch site", x="flight number", hue=class , data=df , aspect=5)
- Plt.xlabel=("flight number", fontsize=25)
- Plt.ylabel=("launch site" , fontsize=25)
- Plt.show()



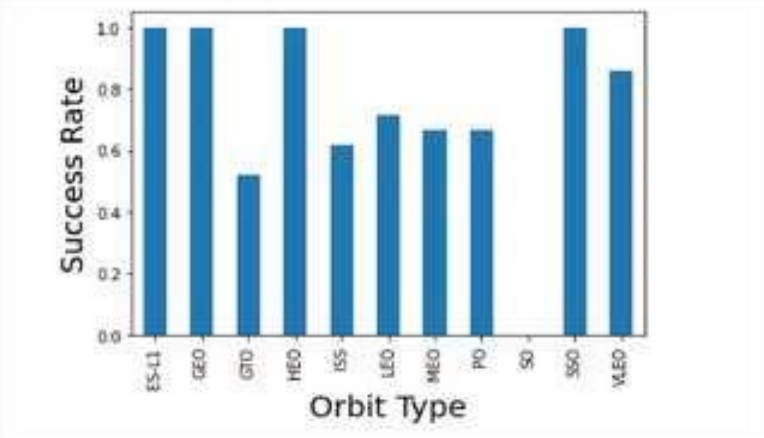
Payload vs. Launch Site

- Sns.catplot(x="launch site", y="pay load mass <kg>", hue=class , data=df , aspect=5)
- Plt.ylabel=("pay load mass <kg>", fontsize=25)
- Plt.xlabel=("launch site" , fontsize=25)
- Plt.show()



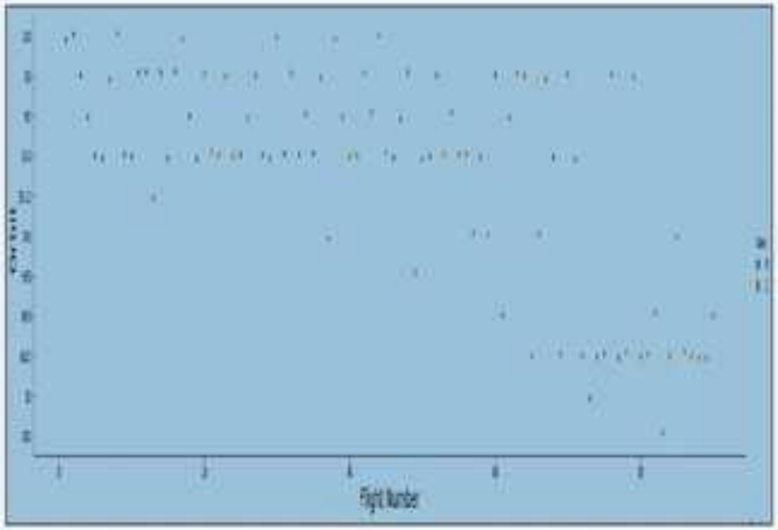
Success Rate vs. Orbit Type

- Sns.catplot(x="orbit type", y="success rate", data=df_success_rate)
- Plt.xlabel=("success rate", fontsize=15)
- Plt.ylabel=("orbit type" , fontsize=15)
- Plt.show()



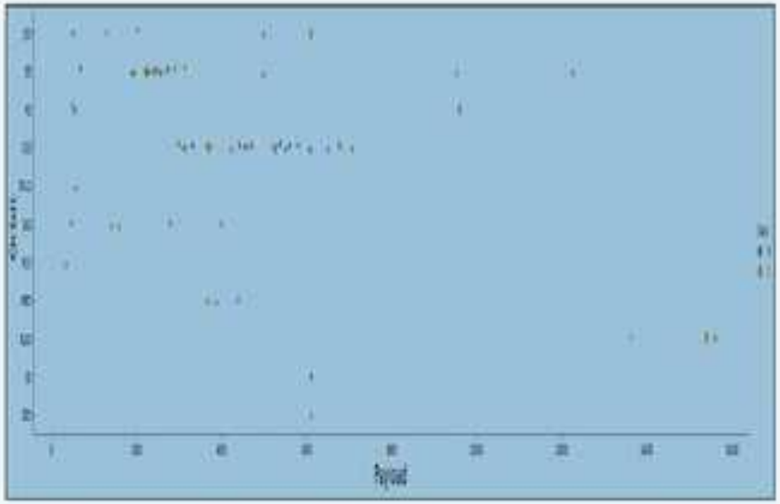
Flight Number vs. Orbit Type

- Sns.catplot(x="flight number", y="orbit", hue=class , data=df , aspect=3)
- Plt.xlabel=("orbit", fontsize=25)
- Plt.ylabel=("flight number" , fontsize=25)
- Plt.show()



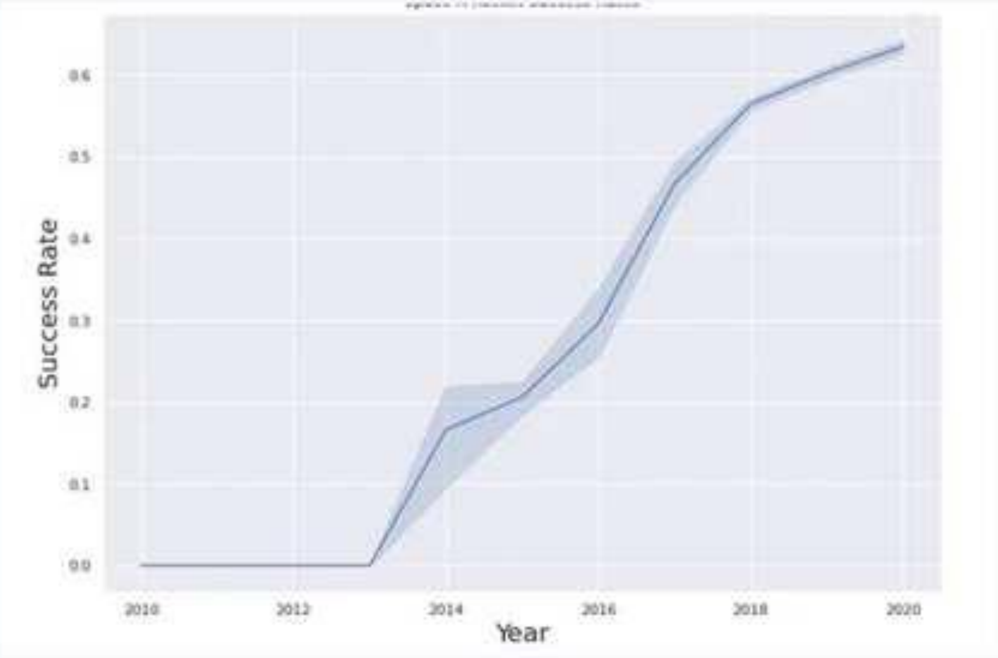
Payload vs. Orbit Type

- Sns.catplot(x="payload", y="orbit", hue=class , data=df , aspect=3)
- Plt.xlabel=("payload", fontsize=20)
- Plt.ylabel=("orbit" , fontsize=20)
- Plt.show()



Launch Success Yearly Trend

- Sns.catplot(x="year", y="success rate", data=df_year_success)
- Plt.xlabel=("year", fontsize=15)
- Plt.ylabel=("success rate" , fontsize=15)
- Plt.show()



All Launch Site Names

- %sql select distinct launch_site from spacextable ;
- Launch_site:
  - CCAFS LC-40
  - CCAFS SLC-40
  - KSC LC-39
  - VAFB

Launch Site Names Begin with 'CCA'

- %sql select * from spacextbl where launch_site like 'CCA%' limit 5

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

Total Payload Mass

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where Customer like 'NASA%'
```

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

sum(PAYLOAD_MASS__KG_)
99980

The total payload carried by boosters from NASA is 99980

Average Payload Mass by F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version like 'F9 v1.1%'
```

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

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

average payload mass carried by booster version F9 v1.1 is
2534.67

First Successful Ground Landing Date

```
%sql select min(Date) from SPACEXTBL where "Landing _Outcome" = "Success (ground pad)"
```

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

min(Date)
01-05-2017

first successful landing outcome on ground pad is 01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000.

```
%%sql

select Booster_Version from SPACEXTBL
where "Landing _Outcome" = "Success (drone ship)"
    and PAYLOAD_MASS__KG_>4000
    and PAYLOAD_MASS__KG_ < 6000
```

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

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

Total Number of Successful and Failure Mission Outcomes

- the total number of successful mission outcomes is 100
- the total number of failure mission outcomes is 1

```
[10] %%sql

     select count(*) from SPACEXTBL
     where "Mission_Outcome" like "Success%"
```

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

count(*)
100

```
[11] %%sql

     select count(*) from SPACEXTBL
     where "Mission_Outcome" like "Failure%"
```

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

count(*)
1

Boosters Carried Maximum Payload

```
%%sql

select Booster_Version from SPACEXTBL
where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

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

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

The Names of the booster which have carried the maximum payload mass

2015 Launch Records

```
[14] %%sql

     select substr(Date, 4, 2) as Month, Booster_Version, Launch_Site from SPACEXTBL
     where substr(Date,7,4)='2015' and "Landing _Outcome" = "Failure (drone ship)"
```

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

Month	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

Failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015:

Month	**booster_version**	**launch_site**
1	F9 v1.1 B1012	CCAFS LC-40
4	F9 v1.1 B1015	CCAFS LC-40

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

Landing_outcomes	Landings
Success	20
No attempt	10
Success(drone ship)	8
Success(ground pad)	6
Failure(drone ship)	4
Controlled(ocean)	3
Failure	3
Failure(parachute)	2
No attempt	1

```
%%sql

select "Landing _Outcome",
    count("Landing _Outcome") as landings
from SPACEXTBL
where Date >= "04-06-2010" and Date <= "20-03-2017"
group by "Landing _Outcome"
order by landings desc
```

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

Landing _Outcome	landings
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Controlled (ocean)	3
Failure	3
Failure (parachute)	2
No attempt	1

Section 3

Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

Launch Site	Lat	Long
CCAFS LC-40	28.56230197	-80.57735648
CCAFS SLC-40	28.56319718	-80.57682003
KSC LC-39A	28.57325457	-80.64689529
VAFB SLC-4E	34.63283416	-120.6107455

<Folium Map Screenshot 2>

- We can see that CCAFS SLC-40 has low rate

<Folium Map Screenshot 3>

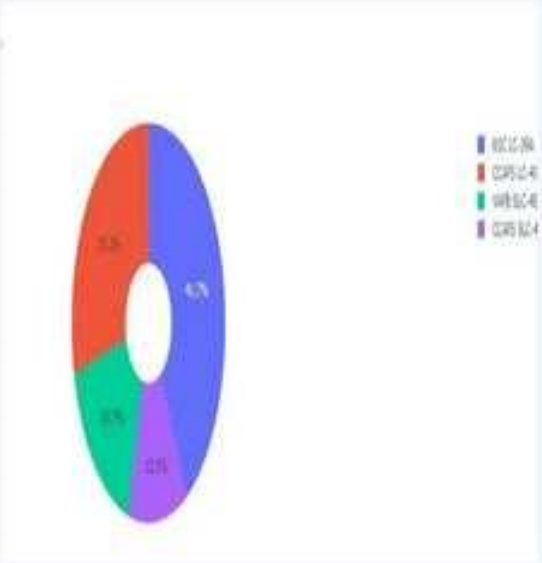
- launch sites are less than 2km from railways
- launch sites are less than 2km from highways
- launch sites are less than 5km from coastline
- It keeps 15 km away from the city

Section 4

Build a Dashboard with Plotly Dash

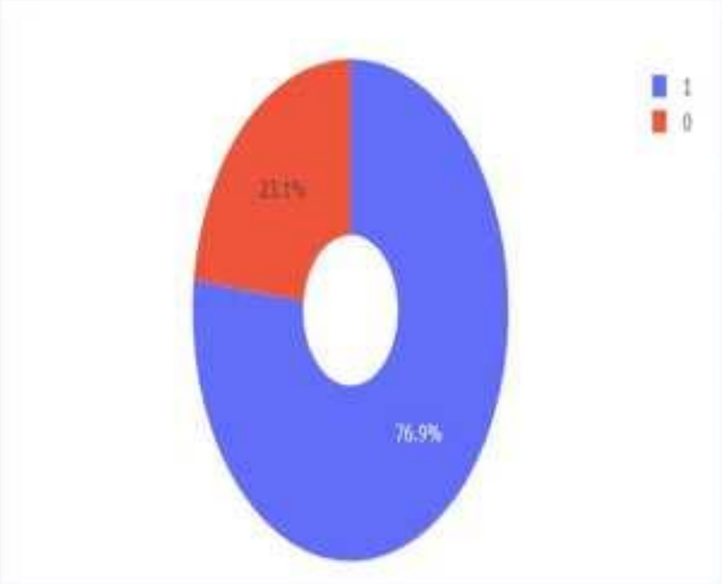
<Dashboard Screenshot 1>

- KSC LC-39A has the most successful launches



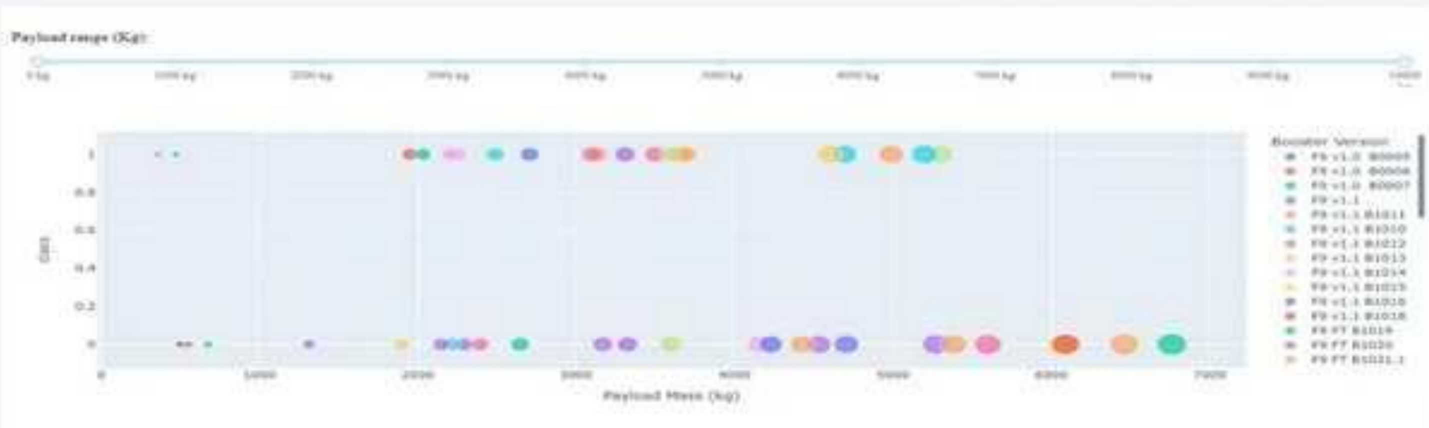
<Dashboard Screenshot 2>

- KSC LC-39A has 76.9% success rate
- KSC LC-39A has 23.1% failure rate



<Dashboard Screenshot 3>

- We can see that range from 2000kg to 4000kg payload mass has the most successful rate
- Also booster version ft has more successful rate

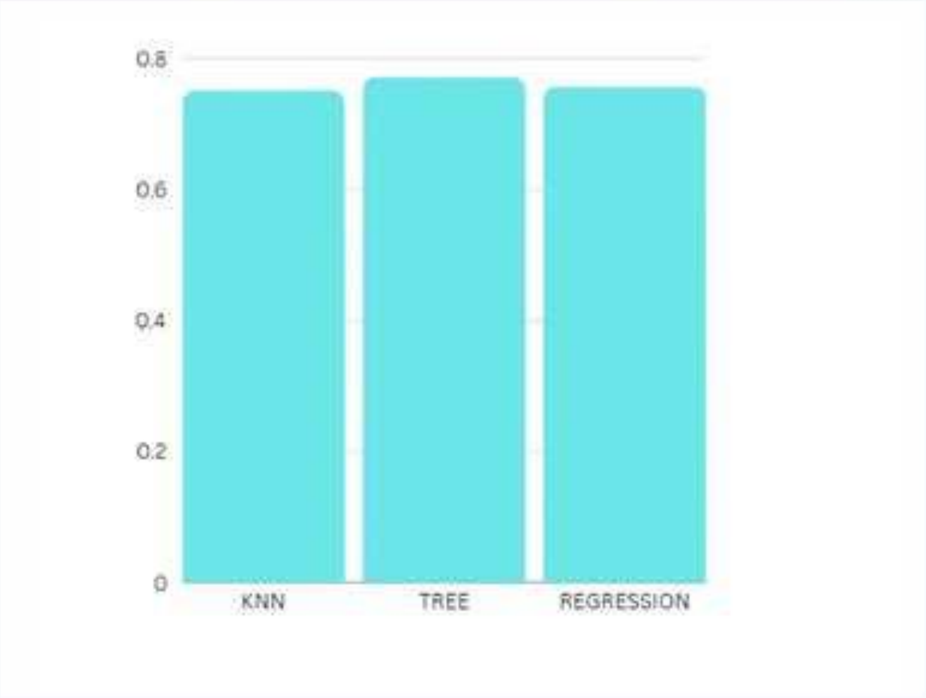


Section 5

Predictive Analysis (Classification)

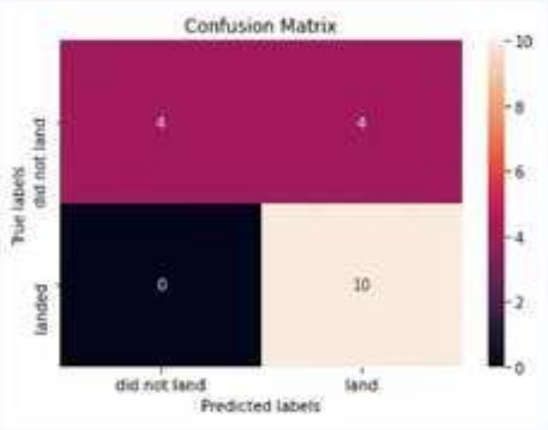
Classification Accuracy

- Decision tree model has the highest accuracy



Confusion Matrix

- After analyzing the models' scores and I get the most accurate model possible. and found that the decision tree classifier model has the lowest bias confusion matrix, the best score, and the accuracy.



Conclusions

- The KNN algorithm is the best model in machine learning for this dataset
- Launch site KSC LC-39A has the highest success rate
- Orbit GEO,HEO & ES-L1,SS have highest success rate
- Launches that have low payload has more success rate
- Success rate is increasing directly proportional with time

Appendix

- Canva tool
- Smart art tool
- Jupyter lite & python (pyodide)
- Labs.cognitive.ai


Thank you!