IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Jorlina Panoti>
<03/18/2023>

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

❑Methodologies

- Data were collected by Space X API (get request) and Wikipedia (Web scraping)
- Eda Analysis is done using SQL, Visualization using Pandas, Matplotlib, Interactive visual Analytics(Folium and Plotly dashboard).
- Machine Learning for building models.

❑Summary of Results

- The success rate of the Falcon 9 rocket launches is associated with the launch sites, payload mass, orbit type and year of launch.
- Visualization were great for decision making
- We gained the optimal model using the decision tree.

Introduction

❑Project background and context

- The purpose is to predict if the Falcon 9 rocket first stage will land successfully
- The Falcon 9 rocket launches is estimated to be about 62 million dollars which is cheaper than other providers due to SpaceX ability to reuse the first stage
- We will determine if the first stage will land, and this will help to determine the cost of the launches.

❑Problems you want to find answers

- Will the rocket land successfully?
- What factors are associated with the success rate of Falcon 9 rocket launches?
- What is the accuracy of successful landing?

Methodology

Executive Summary

Data collection methodology:

- Rest API (`https://api.spacexdata.com/v4/launches/past`)
- Web scraping

Perform data wrangling

- The data was transformed

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

SpaceX launch data is gathered from an API -The SpaceX REST.(https://api.spacexdata.com/v4/launches/past)

- *I collected data with an API Call*

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

- *Converted to a data frame with JSON*

	static_fire_date_utc	static_fire_date_unix	net	window	rocket	success	failures	details	crew	ships	capsules	payload
0	2006-03-17T00:00:00.000Z	1.142554e+09	False	0.0	5e9d0d95eda69955f709d1eb	False	[{'time': 33, 'altitude': None, 'reason': 'merlin engine failure'}]	Engine failure at 33 seconds and loss of vehicle	[]	[]	[]	[5eb0e4b5b6c3bb0006eeb
							[{'time':	Successful first stage burn and transition to second				

- *Updating rows and columns*

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Long
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin1A	167.7
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2A	167.7
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2C	167.7
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin3C	167.7
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.5

- *We kept only Falcon 9 Launches*

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	8	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
6	10	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.577366	28.561857
7	11	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.610829	34.632093
8	12	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-80.577366	28.561857

- *Convert the data to a CSV file*(data_falcon9.to_csv('dataset_part_1.csv', index=False))

Data Collection – SpaceX API

- The GitHub URL of the completed SpaceX API calls notebook

(Applied-Data-Science-Capstone/data_collection api.ipynb at main · Jorlina1/Applied-Data-Science-Capstone (github.com))

Data Collection - Scraping

Created a BeautifulSoup object from the HTML response

Extracted all column/variable names from the HTML table header

Created a data frame by parsing the launch HTML tables

1.Created an empty dictionary

```
launch_dict= dict.fromkeys(column_names)
```

2.Filled up the launch_dict with launch records extracted from table rows.

3.Converted it to a Pandas dataframe

4..Export it to a CSV file

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

[Applied-Data-Science-Capstone/webscraping.ipynb at main · Jorlina1/Applied-Data-Science-Capstone (github.com)]

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text)
```

```
print(column_names)
```

```
['Flight No.', 'Date and time ( )', 'Launch site', 'Payload', 'Payload mass', 'Orbit', 'Customer', 'Launch outcome']
```

```
df=pd.DataFrame(launch_dict)
```

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10
...	...	...	...	...	...	...	...	...	...	...	...
116	117	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1051.10	Success	9 May 2021	06:42
117	118	KSC	Starlink	~14,000 kg	LEO	SpaceX	Success\n	F9 B5B1058.8	Success	15 May 2021	22:56
118	119	CCSFS	Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1063.2	Success	26 May 2021	18:59
119	120	KSC	SpaceX CRS-22	3,328 kg	LEO	NASA	Success\n	F9 B5B1067.1	Success	3 June 2021	17:29
120	121	CCSFS	SXM-8	7,000 kg	GTO	Sirius XM	Success\n	F9 B5	Success	6 June 2021	04:26

Data Wrangling

- Identified numerical or categorical columns and calculated the percentage of the missing values.

```
FlightNumber        int64
Date               object
BoosterVersion     object
PayloadMass       float64
Orbit              object
LaunchSite         object
Outcome            object
Flights             int64
GridFins             bool
Reused               bool
Legs                 bool
LandingPad         object
Block             float64
ReusedCount         int64
Serial             object
Longitude         float64
Latitude          float64
dtype: object
```

```
df.isnull().sum()/df.count()*100
```

```
FlightNumber       0.000
Date               0.000
BoosterVersion     0.000
PayloadMass        0.000
Orbit              0.000
LaunchSite         0.000
Outcome            0.000
Flights            0.000
GridFins           0.000
Reused             0.000
Legs               0.000
LandingPad        40.625
Block              0.000
ReusedCount        0.000
Serial             0.000
Longitude          0.000
Latitude           0.000
dtype: float64
```

Data Wrangling

- The number of launches on each site
- The number and occurrence of each orbit
- The number and occurrence of mission outcome per orbit type

```
df['LaunchSite'].value_counts()
```

```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

```
df['Orbit'].value_counts()
```

```
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
ES-L1     1
HEO       1
SO        1
GEO       1
Name: Orbit, dtype: int64
```

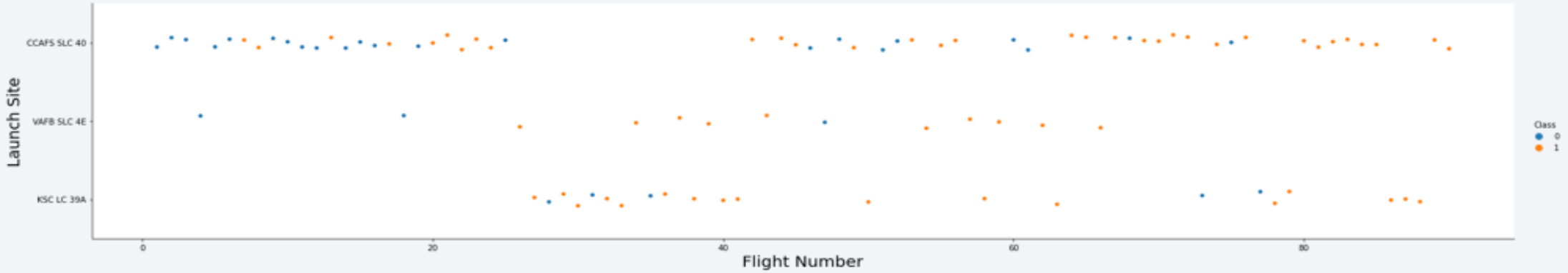
```
landing_outcomes=df['Outcome'].value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: Outcome, dtype: int64
```

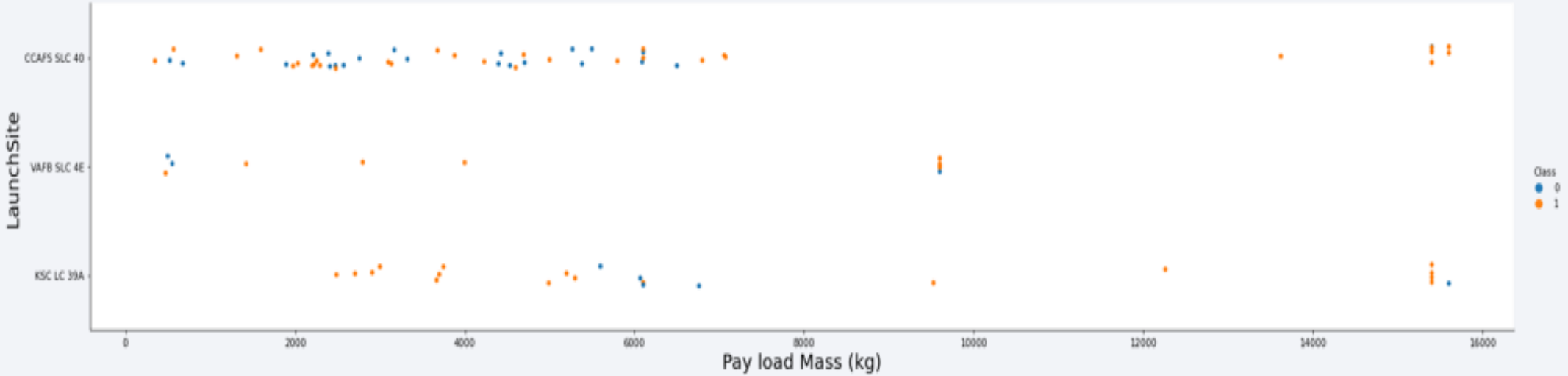
- [Applied-Data-Science-Capstone/data_wrangling_eda.ipynb at main · Jorlina1/Applied-Data-Science-Capstone (github.com)]

EDA with Data Visualization

- The increase of Flight Number comes with the increase of success rate in launch sites

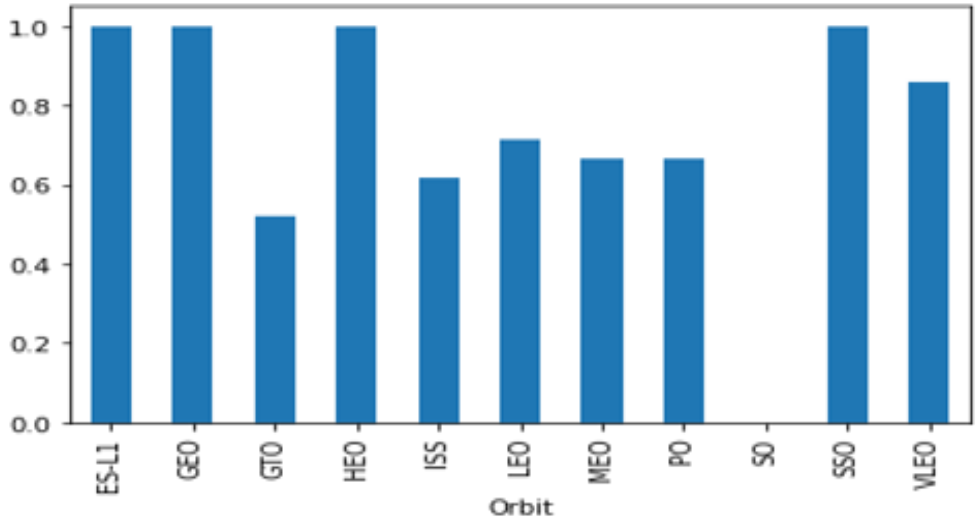


- The increase of Pay load Mass comes with the increase of success rate in Launch Sites

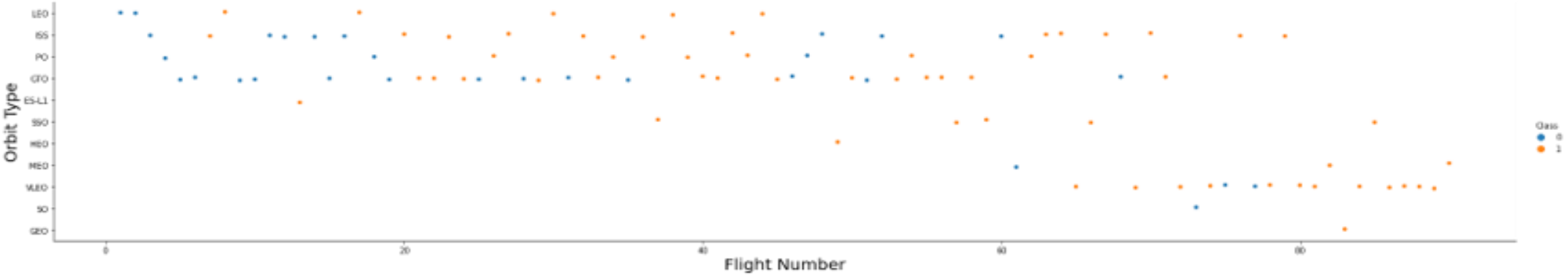


EDA with Data Visualization

- The relationship between success rate of each orbit type.(SO has no success rate and ESL-1,GEO,HEO and SSO has a success rate of 100%)

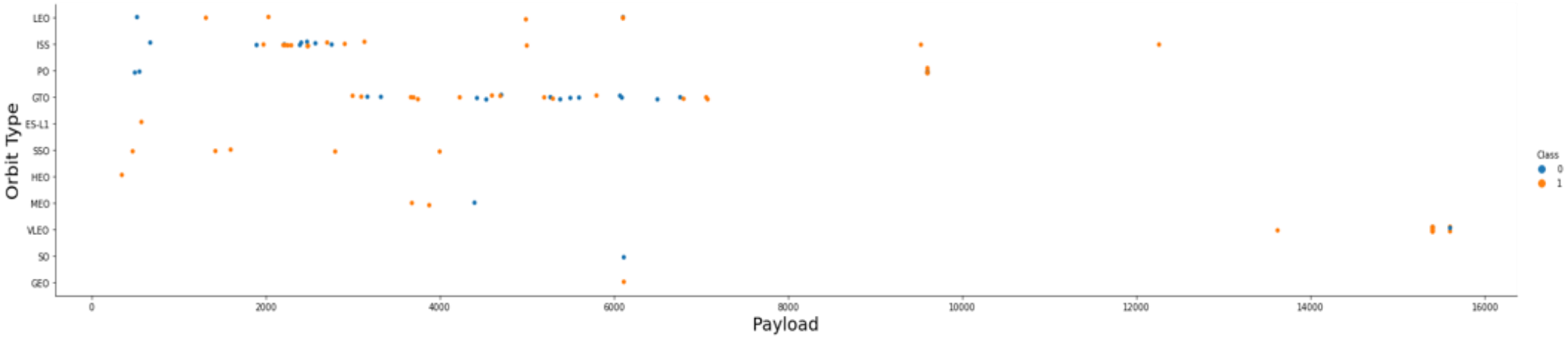


- No relationship between Flight Number and Orbit type

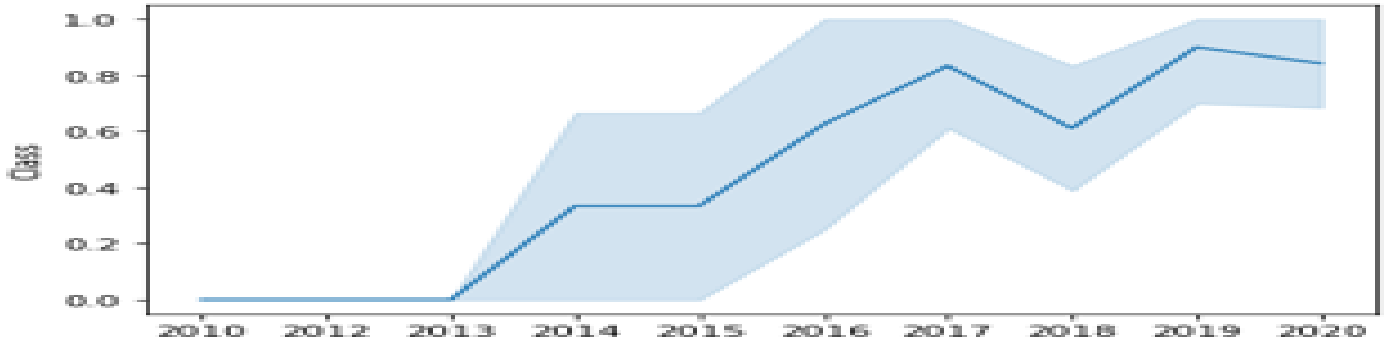


EDA with Data Visualization

- The relationship between Payload and Orbit type



- The launch success yearly trend (the success rate since 2013 kept increasing till 2020)



EDA with SQL

- The names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'

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

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

- The total payload mass carried by boosters launched by NASA (CRS)

sum
45596

- Average payload mass carried by booster version F9 v1.1

Average
2534.6666666666665

EDA with SQL

- The date of the first successful landing outcome in ground
- Total number of successful and failure mission outcomes

Mission_Outcome	Count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

- The booster_versions which have carried the maximum payload mass

Date
01-03-2013

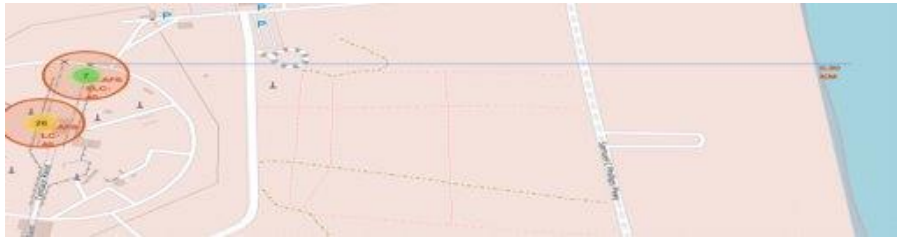
Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

Build an Interactive Map with Folium

- All launch sites on a map
- folium.Circle and folium.Marker to create markers and circles above markers.
- Mark the success/failed launches for each site on the map
- *folium.PolyLine* object
  to draw PolyLine between a launch site to the selected coastline point
- Launch sites are in close proximity to railways, to highways and to coastline.
- Launch sites have significant distance from cities.
- Closest distance of launch sites is to the city of Tutsville at about 0.9km.

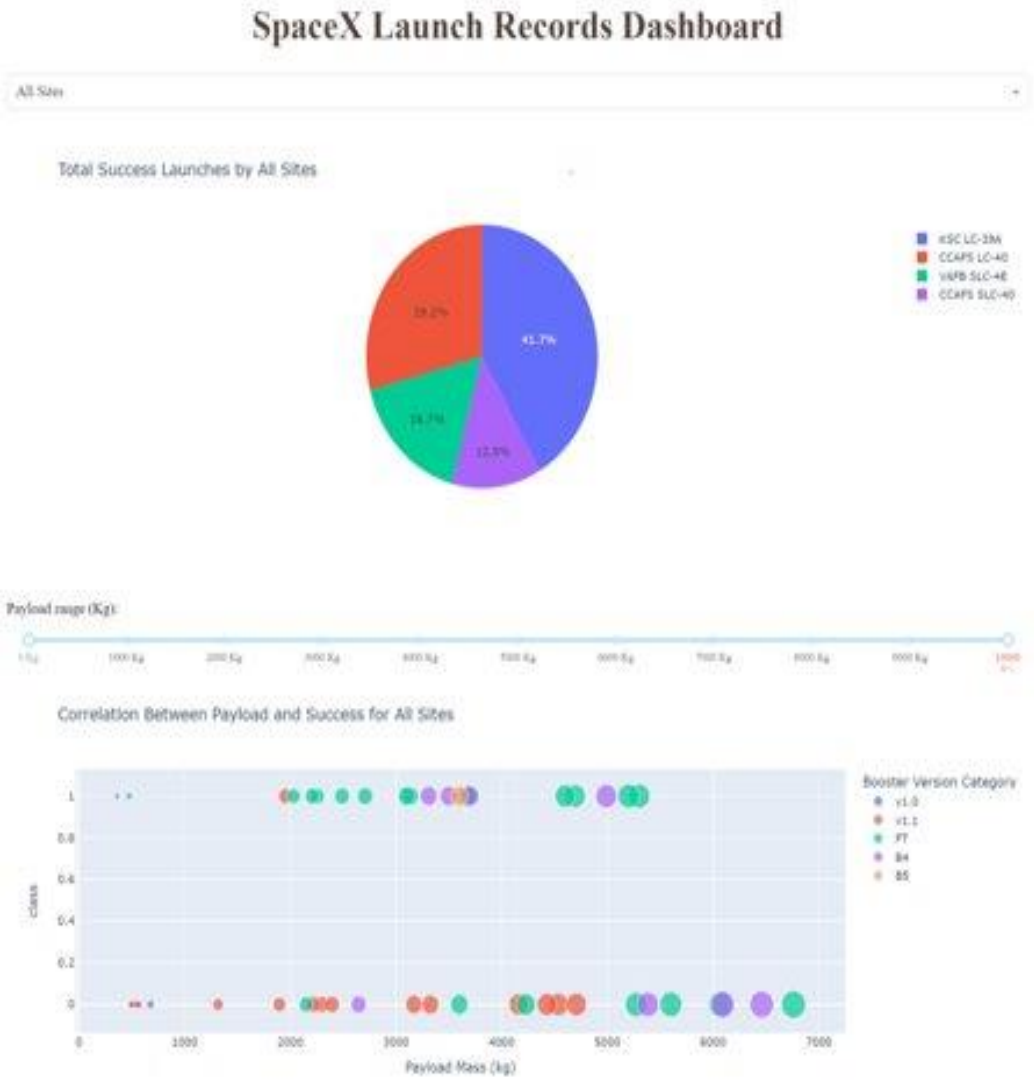
	Launch Site	Lat	Long
0	CCAFS LC-40	28.562302	-80.577356
1	CCAFS SLC-40	28.563197	-80.576820
2	KSC LC-39A	28.573255	-80.646895
3	VAFB SLC-4E	34.632834	-120.610745

	Launch Site	Lat	Long	class	marker_color
46	KSC LC-39A	28.573255	-80.646895	1	green
47	KSC LC-39A	28.573255	-80.646895	1	green
48	KSC LC-39A	28.573255	-80.646895	1	green
49	CCAFS SLC-40	28.563197	-80.576820	1	green
50	CCAFS SLC-40	28.563197	-80.576820	1	green
51	CCAFS SLC-40	28.563197	-80.576820	0	red
52	CCAFS SLC-40	28.563197	-80.576820	0	red
53	CCAFS SLC-40	28.563197	-80.576820	0	red
54	CCAFS SLC-40	28.563197	-80.576820	1	green
55	CCAFS SLC-40	28.563197	-80.576820	0	red

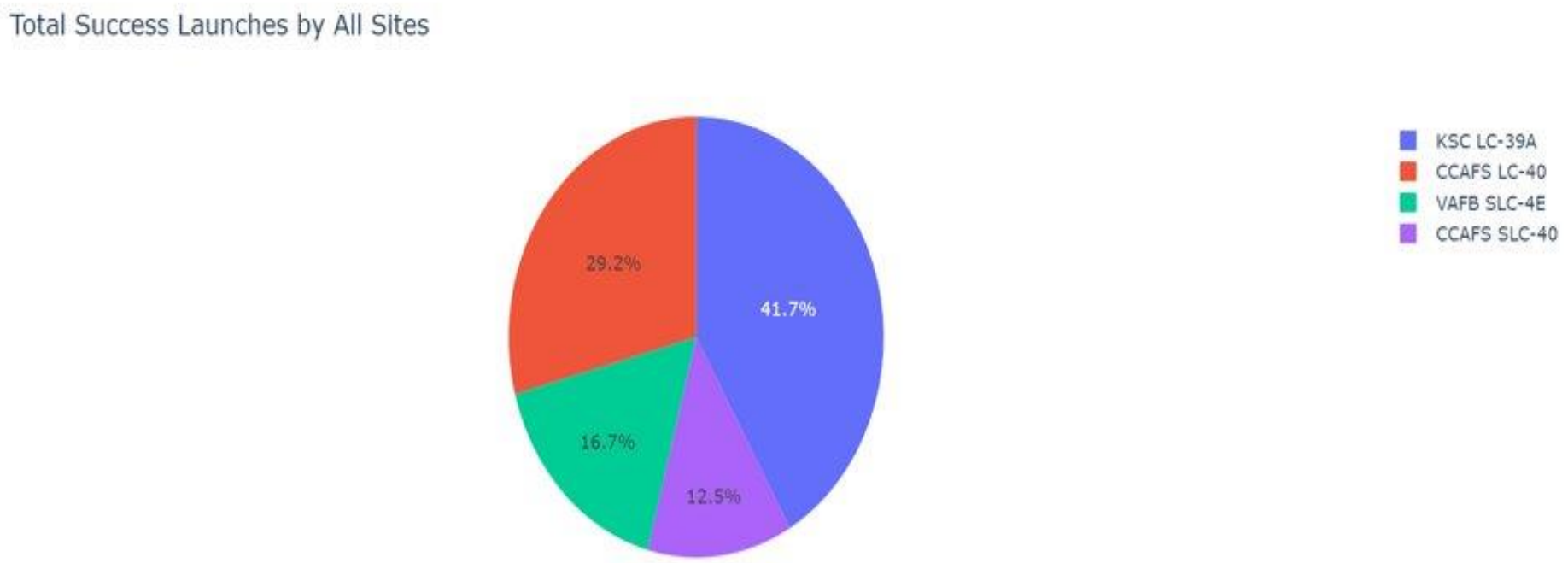


Dashboard with Plotly Dash

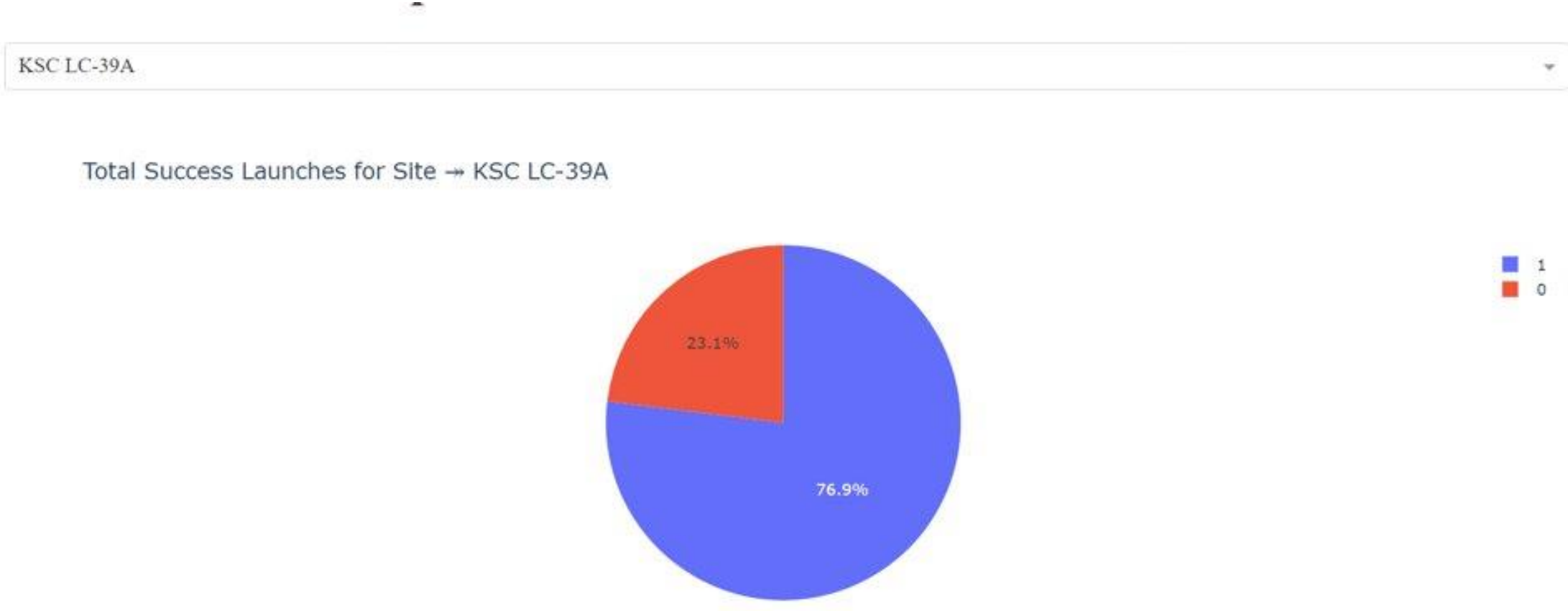
- Added a launch site drop-down input component
- Added a callback function to render the required success outcome pie chart
- Added a range slider to choose payload
- Added callback function to render the payload-outcome scatter plot



Dashboard with Plotly Dash



The Highest Launch Success Rate

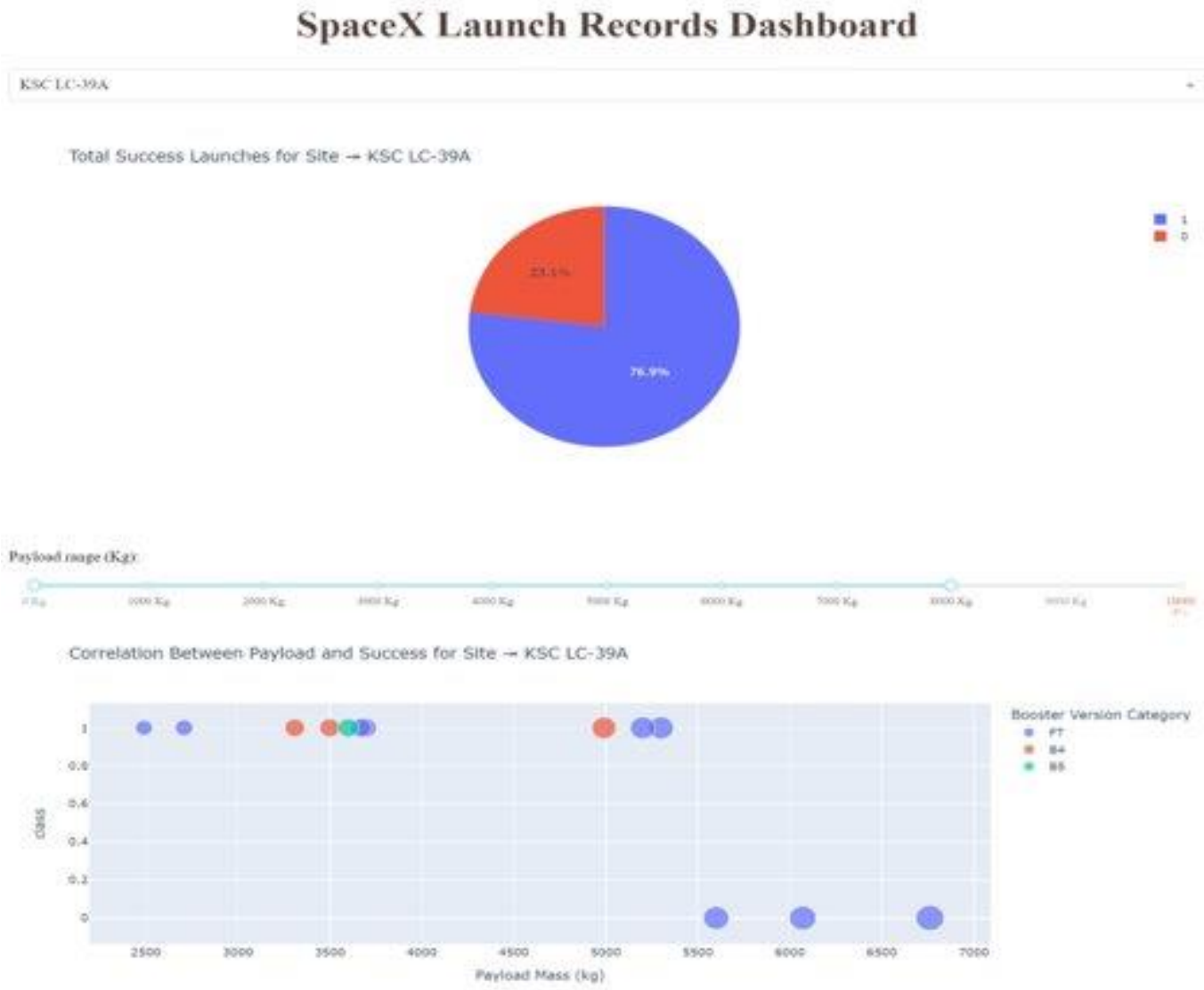


Payload Mass range for successful Launches

Payload range for unsuccessful lauches

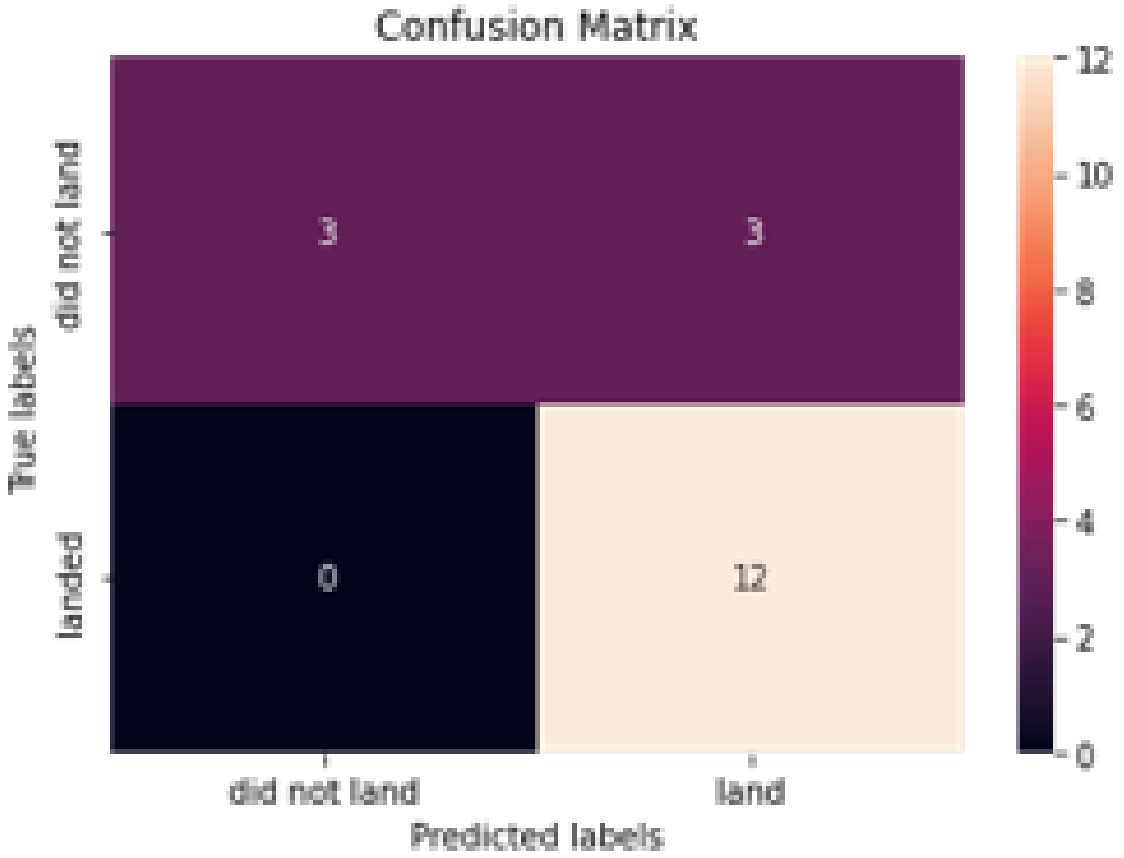
Payload Mass range

- The highest success rate is from 0-5000
- The lowest success rate is from 6000-10000
- FT booster version has the highest score



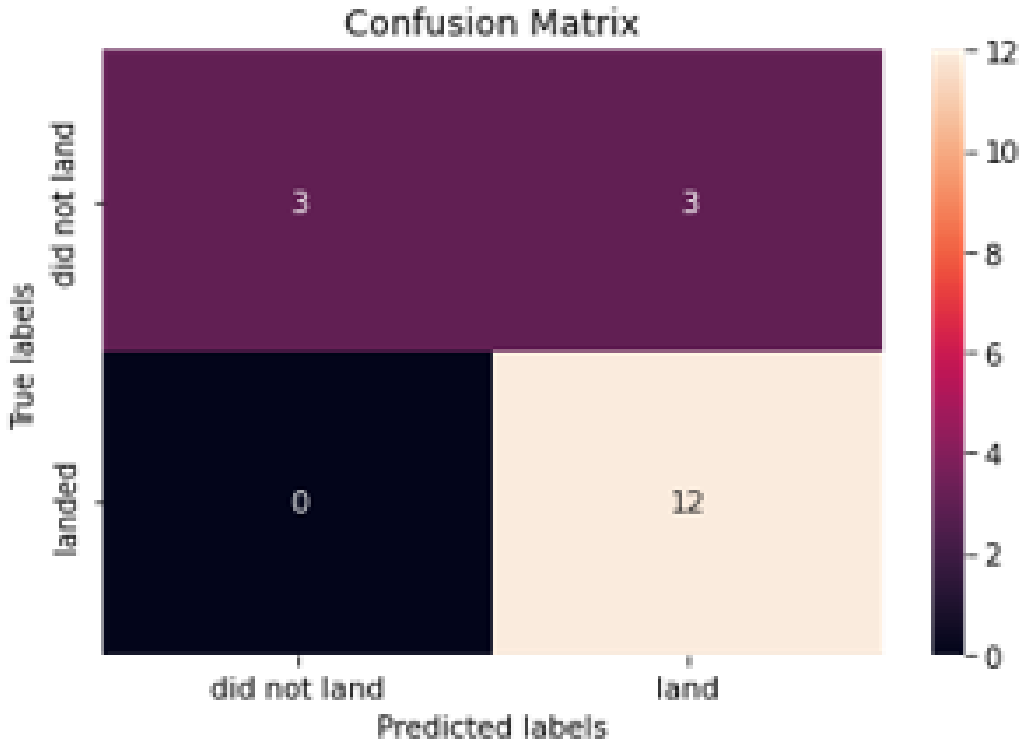
Predictive Analysis (Classification)

- Used method to_numpy() to create an array from column Class.
- We standardized data
- We split the data into training and testing data using the function train_test_split
- We only have 18 test samples
- Created a logistic regression and a GridSearchCV object
- The accuracy on the test data using the method score: 0.8333



Predictive Analysis

- Created a decision tree classifier object then create a `GridSearchCV` object `tree_cv`
- The accuracy of tree_cv on the test data using the method `score is 0.9444444`
- The method performs best is Decision Tree



]:

	ML Method	Accuracy Score(%)
0	Support Vector Machine	83.333333
1	Logistic Regression	83.333333
2	K Nearest Neighbour	83.333333
3	Decision Tree	94.444444

Conclusions

- The site with the highest score of landing successfully is KSC LC-39A
- The highest success rate is for payload mass=[0-5000]
- The rate of successful launch is correlated with flight number and payload mass.
- The method performs best is Decision Tree with accuracy of 94%

Apendix

- You can find all my notebook to this link:

Applied-Data-Science-Capstone/data_wrangling_eda.ipynb at main · Jorlina1/Applied-Data-Science-Capstone (github.com)

Thank you!