IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Alireza Ramezani
03/28/2024

Outline

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

Executive Summary

- Summary of methodologies
  - Our strategy involved utilizing both API integration and web scraping methods to gather data. After obtaining the data, we utilized various Python techniques to thoroughly process and clean it. We then used SQL queries to extract relevant information from the refined dataset. Early insights were obtained through systematic data visualization and trend analysis. To complete our analytical framework, we utilized supervised machine learning models to predict the success of landing events.
- Summary of all results
  - By conducting thorough data analysis, we uncovered clear patterns and correlations among variables that directly impact the success of landing events. Using these insights, we built and trained a predictive model that showed significant ability to accurately predict the likelihood of a successful landing event. Importantly, the model achieved an impressive accuracy rate of 83%, highlighting its effectiveness in delivering dependable predictions within this field.

Introduction

- SpaceX's dedication to reusable rockets has notably decreased the expenses associated with space travel by concentrating on retrieving the initial phase of the rocket. Preserving and reusing costly components from this phase is crucial for reducing costs directly. Evaluating the success rate of these retrieval events provides a significant metric for assessing efficiency and cost-effectiveness in SpaceX's innovative approach. This project focuses on predicting the success of the first phase retrieval event, providing predictive insights to improve decision-making within the space industry

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
    - Space X REST API
    - Web Scraping
- Perform data wrangling
    - Dropping everything but Falcon 9 data
    - Missing Payload Mass values replaced by mean
    - Making outcomes into binary categorical data
- Perform exploratory data analysis (EDA) using visualization and SQL
    - Scatter plots and Histograms
    - Maps and dashboards
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Logistics Regression, SVM, Decision Trees, KNN

Data Collection

- Data was first collected using SpaceX API (a RESTful API) by making a get request to theSpaceX API. This was done by first defining a series helper functions that would help in the use of the API to extract information using identification numbers in the launch data and then requesting rocket launch data from the SpaceX API url.
- SpaceX launch data was requested and parsed using the GET request and then decoded the response content as aJson result which was then converted into a Pandas data frame.
- performed web scraping to collect Falcon 9 historical launch records from Wikipedia page

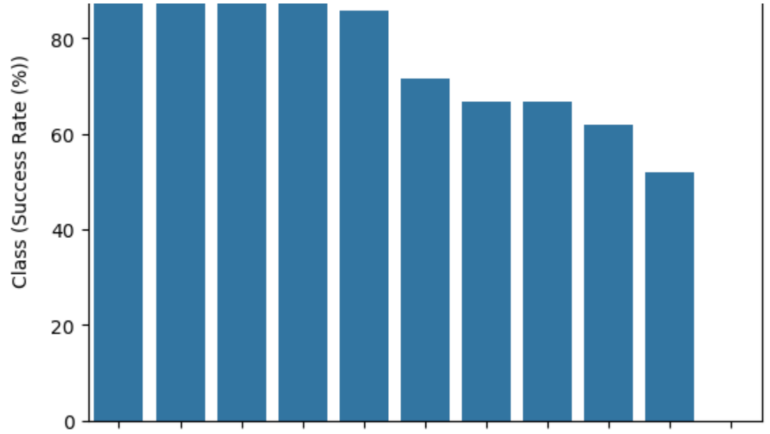
Data Collection – SpaceX API

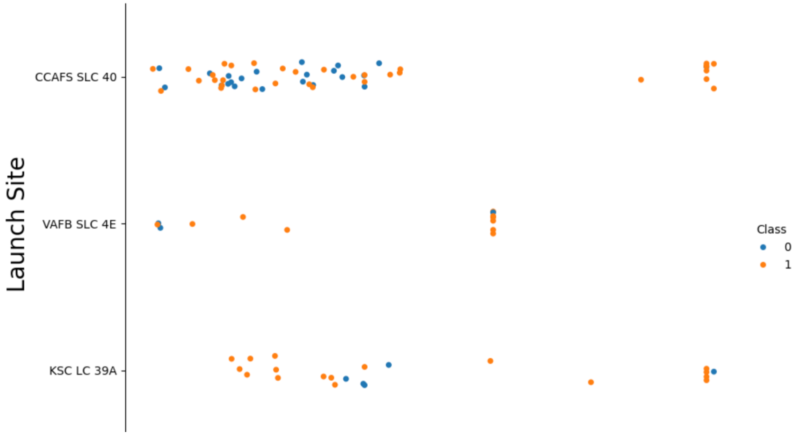
- Define auxiliary function to parse the data
- Retrieve data from the REST API using the method GET
- Parse the data with the previously built auxiliary functions
- Store the data in PANDAS DataFrame

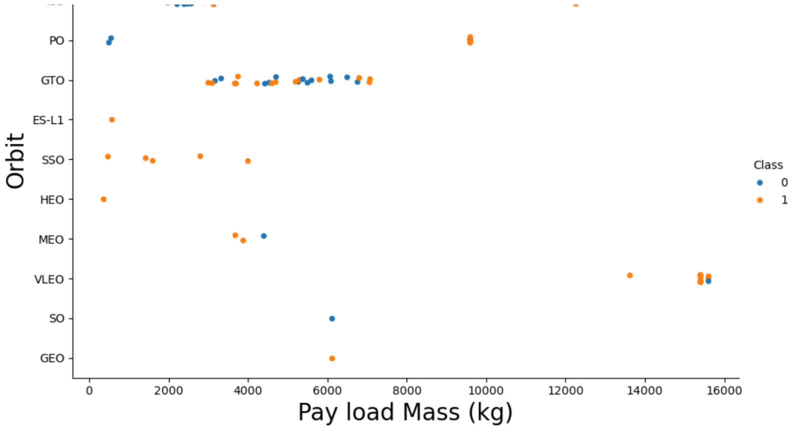
Data Wrangling

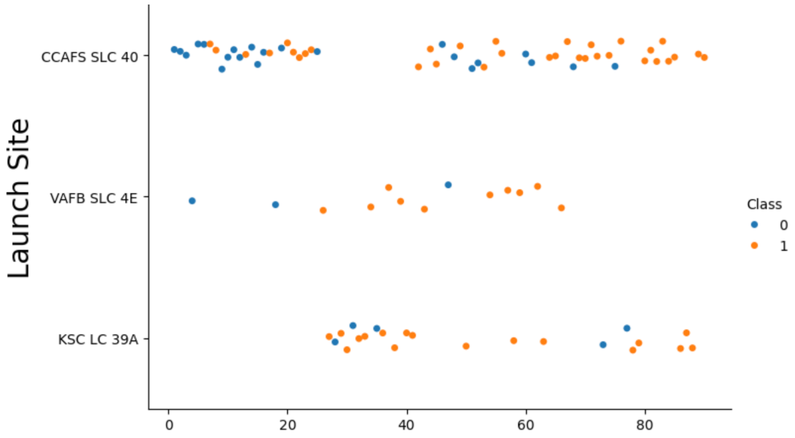
- Describe how data were processed
- You need to present your data wrangling process using key phrases and flowcharts
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose

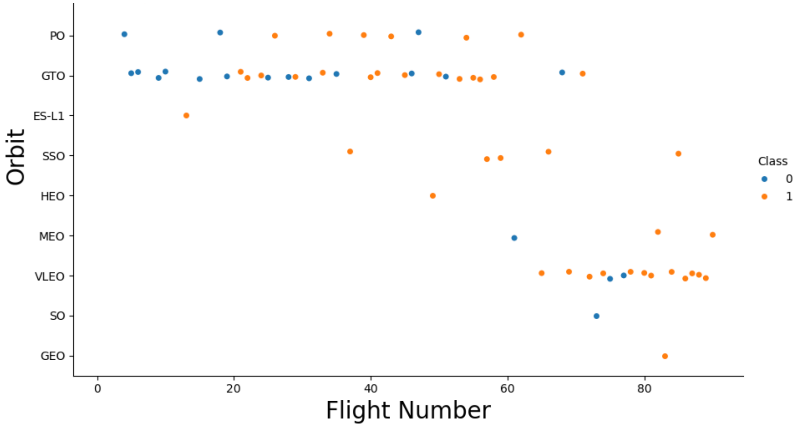
EDA with Data Visualization











EDA with SQL

- Displaying the names of the launch sites.
- Displaying 5 records where launch sites begin with the string ‘CCA’.
- Displaying the total payload mass carried by booster launched by NASA (CRS).
- Displaying the average payload mass carried by booster version F9 v1.1.
- Listing the date when the first successful landing outcome in ground pad was achieved.
- Listing the names of the boosters which have success in drone ship 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.

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) as "Payload Mass Kgs", Customer, Booster_Version FROM 'SPACEXTBL' WHERE Booste
```

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

Payload Mass Kgs	Customer	Booster_Version
2534.6666666666665	MDA	F9 v1.1 B1003

```
%sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTBL;
```

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

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) as "Total Payload Mass(Kgs)", Customer FROM 'SPACEXTBL' WHERE Customer = 'NASA
```

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

Total Payload Mass(Kgs)	Customer
45596	NASA (CRS)

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

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landin _Outcom
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failu (parachut
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	Failu (parachut
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attem
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attem
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attem

```
%sql SELECT DISTINCT Booster_Version, Payload FROM SPACEXTBL WHERE "Landing _Outcome" = "Success (drone ship)" A
```

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

Booster_Version	Payload
F9 FT B1022	JCSAT-14
F9 FT B1026	JCSAT-16
F9 FT B1021.2	SES-10
F9 FT B1031.2	SES-11 / EchoStar 105

```
%sql SELECT MIN(DATE) FROM 'SPACEXTBL' WHERE "Landing _Outcome" = "Success (ground pad)";
```

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

MIN(DATE)
01-05-2017

```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") as Total FROM SPACEXTBL GROUP BY "Mission_Outcome";
```

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

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

```
%sql SELECT "Booster_Version",Payload, "PAYLOAD_MASS__KG_" FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX
```

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

Booster_Version	Payload	PAYLOAD_MASS__KG_
F9 B5 B1048.4	Starlink 1 v1.0, SpaceX CRS-19	15600
F9 B5 B1049.4	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600
F9 B5 B1051.3	Starlink 3 v1.0, Starlink 4 v1.0	15600
F9 B5 B1056.4	Starlink 4 v1.0, SpaceX CRS-20	15600
F9 B5 B1048.5	Starlink 5 v1.0, Starlink 6 v1.0	15600
F9 B5 B1051.4	Starlink 6 v1.0, Crew Dragon Demo-2	15600
F9 B5 B1049.5	Starlink 7 v1.0, Starlink 8 v1.0	15600
F9 B5 B1060.2	Starlink 11 v1.0, Starlink 12 v1.0	15600
F9 B5 B1058.3	Starlink 12 v1.0, Starlink 13 v1.0	15600
F9 B5 B1051.6	Starlink 13 v1.0, Starlink 14 v1.0	15600
F9 B5 B1060.3	Starlink 14 v1.0, GPS III-04	15600
F9 B5 B1049.7	Starlink 15 v1.0, SpaceX CRS-21	15600

```
%sql SELECT substr(Date,7,4), substr(Date, 4, 2),"Booster_Version", "Launch_Site", Payload, "PAYLOAD_MASS__KG_",
```

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

substr(Date,7,4)	substr(Date, 4, 2)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Mission_Outcome	Landing _Outcome
2015	01	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	Success	Failure (drone ship)
2015	04	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898	Success	Failure (drone ship)

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

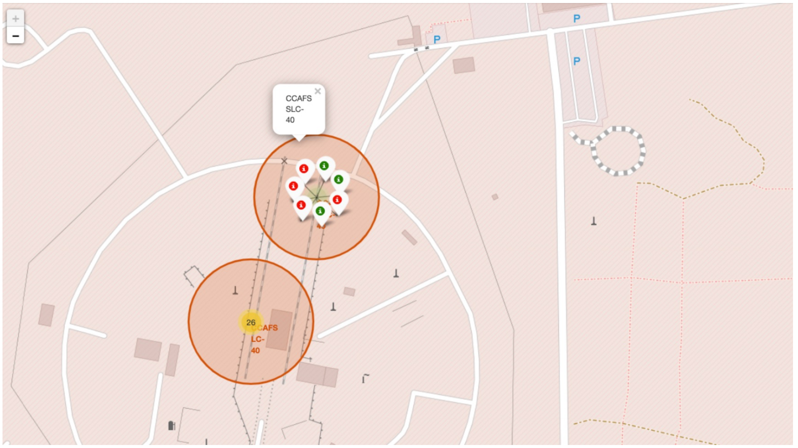
```
* sqlite:///my_data1.db
Done.
```

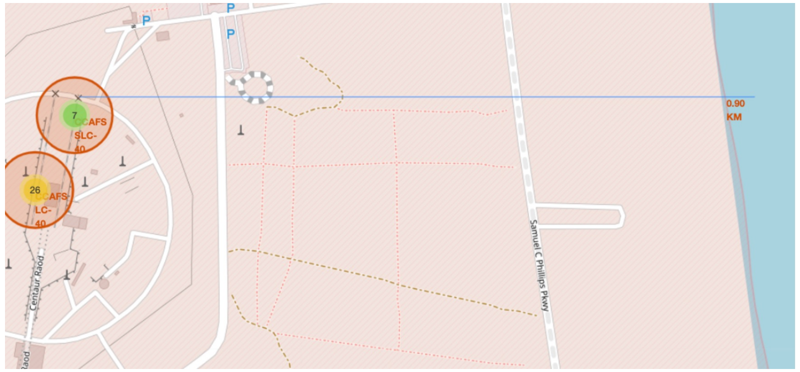
Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

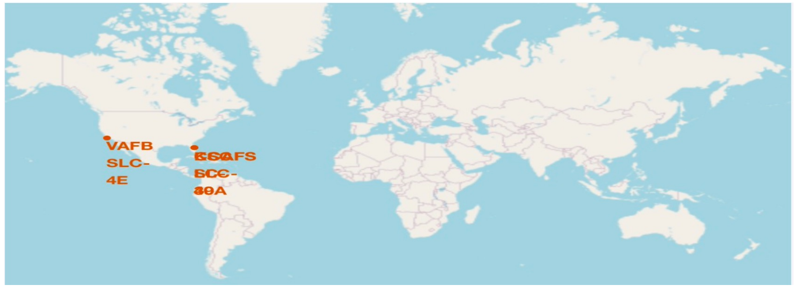
```
%sql SELECT * FROM SPACEXTBL WHERE "Landing _Outcome" LIKE 'Success%' AND (Date BETWEEN '04-06-2010' AND '20-03-2
```

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

Build an Interactive Map with Folium

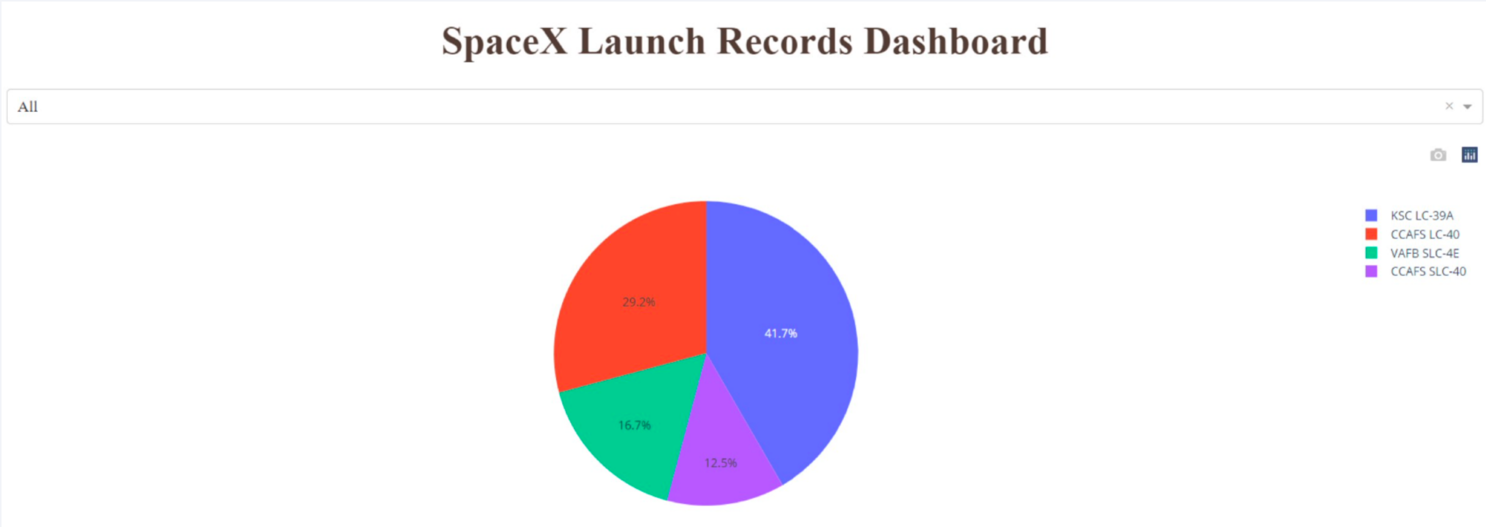


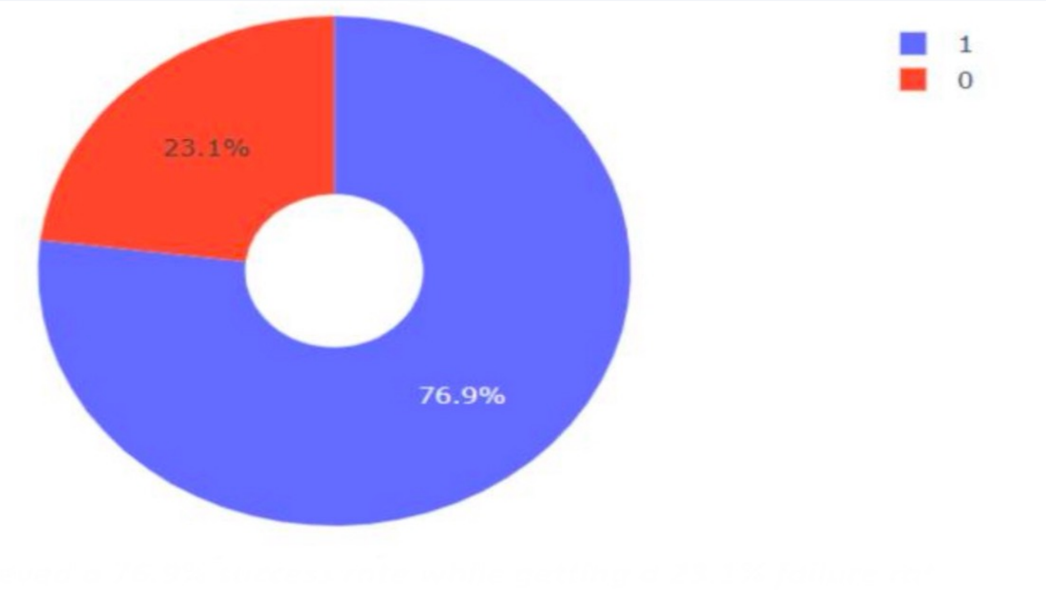




Build a Dashboard with Plotly Dash

- We built an interactive dashboard with Plotly dash.
  We plotted pie charts showing the total launches by a certain sites.
- We plotted scatter graph showing the relationship with Outcome and PayloadMass (Kg) for the different booster version.





Predictive Analysis (Classification)

- We loaded the data using numpy and pandas, transformed the data, split our data into training and testing sets.
- Tune different hyperparameters using GridSearchCV.
- We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.

Results

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

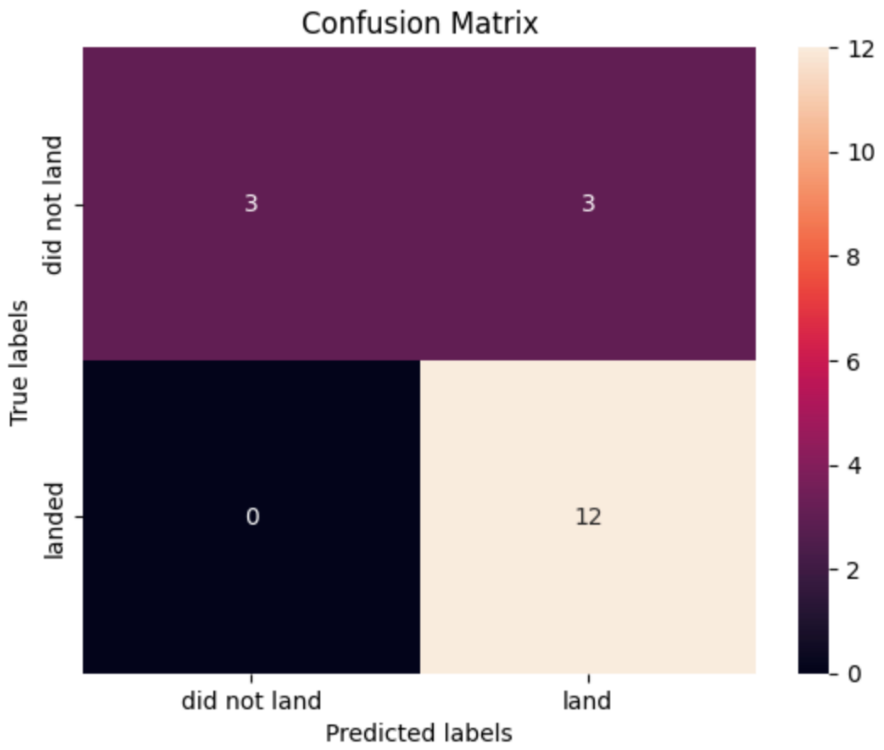
Section 5

Predictive Analysis (Classification)

Classification Accuracy

- We loaded the data using numpy and pandas, transformed the data, split our data into training and testing sets.
- Tune different hyperparameters using GridSearchCV.
- We used accuracy as the metric for our model, improved the model using feature engineering and algorithm tuning.

Confusion Matrix




```
print("accuracy: ",tree_cv.score(X_test,Y_test))

accuracy:  0.8333333333333334
```

Conclusions

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate started to increase in 2013 till 2020.

Thank you!