IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Gilang Islamay
18 February 2024

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

- Project background and context
- Problems you want to find answers

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Through API and Web Scarping!
- Perform data wrangling
  - One Hot Encoding!
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Build, tune, and evaluate classification models!

Data Collection

Datasets were collected using two methods, which are

- SpaceX Api
- SpaceX Web Scraping using BeautifulSoup

Data Collection – SpaceX API

- We utilized the GET request with the SpaceX API to gather the data.
- https://github.com/AfterRain007/TFDS/blob/main/Finale/1%20jupyter-labs-spacex-data-collection-api.ipynb

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

[50]: response = requests.get(spacex_url)

[54]: # Use json_normalize meethod to convert the json result into a dataframe
      data = pd.json_normalize(response.json())

[55]: # Get the head of the dataframe
      data.head()
```

[55]:

	static_fire_date_utc	static_fire_date_unix	net	window	rocket
0	2006-03-17T00:00:00.000Z	1.142554e+09	False	0.0	5e9d0d95eda69955f709d1eb

Data Collection - Scraping

- We employed web scraping techniques to extract Falcon 9 launch records from a webpage using BeautifulSoup. After parsing the table, we are able to transformed the data into a pandas dataframe.
- https://github.com/AfterRain007/TFDS/blob/main/Finale/1%20jupyter-labs-webscraping.ipynb

```
[39]: df= pd.DataFrame({ key:pd.Series(value) for key, value in launch_dict.items() })
```

We can now export it to a **CSV** for the next section, but to make the answers consistent and in

Following labs will be using a provided dataset to make each lab independent.

```
[42]: df.to_csv('spacex_web_scraped.csv', index=False)
      df.head()
```

[42]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	
1	2	CCAFS	Dragon	0	LEO	.mw-parser-outpu ol,.mw-pa
2	3	CCAFS	Dragon	525 kg	LEO	NA
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	N
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	N

Data Wrangling

- Before going into Exploratory Data Anlysis, etc etc. First thing first we need to do some Data Wrangling! Let's Create a landing outcome label from Outcome column, where we set 0 to the bad outcome and 1 if it's otherwise. We can achieve this by doing simple list comprehension.

```
[17]: # landing_class = 0 if bad_outcome
      # landing_class = 1 otherwise
      landing_class = [0 if x in bad_outcomes else 1 for x in df['Outcome']]

[19]: df.head(5)
```

[19]:

te	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
0-)4	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
2- 22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0

EDA with Data Visualization

Now, let's explore the data to better understand it's value. There's a couple chart that we plotted, which are

- Scatter Plot
- Line Plot
- Bar Plot

We use these chart to better understand the correlation between two variable!

https://github.com/AfterRain007/TFDS/blob/main/Finale/4%20jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- Not only with pandas, we can use SQL too in Jupyter Notebook! In this example we load the SpaceX dataset into a PostgreSQL database and gain some insight from the dataset!

https://github.com/AfterRain007/TFDS/blob/main/Finale/5%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- How does one build a map? Let alone in python? Well, let me introduce you to Folium!
- With Folium, we are capable of marking all the launch sites with markers, circles, and line! Where if there's a high success rate based on the data in one launch site, we are able to make the colors to the said markers to be green.

https://github.com/AfterRain007/TFDS/blob/main/Finale/6%20lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- We can also build an interactive dashboard in Jupyter Notebook using Plotly Dash. With this dashboard, you don’t need to change any line of code! So everyone can see the insight of the data intuitively.

https://github.com/AfterRain007/TFDS/blob/main/Finale/spacex_dash_app.ipynb

Predictive Analysis (Classification)

Now let’s do some Predictive Analysis!

- Load the data into a pandas dataframe to do some data wrangling
- Split the data into 2, which are training and testing data
- Train a couple of model using our training data with GridSearchCV as our hyperparameter tuning
- Evaluate each model to compare which one is the best

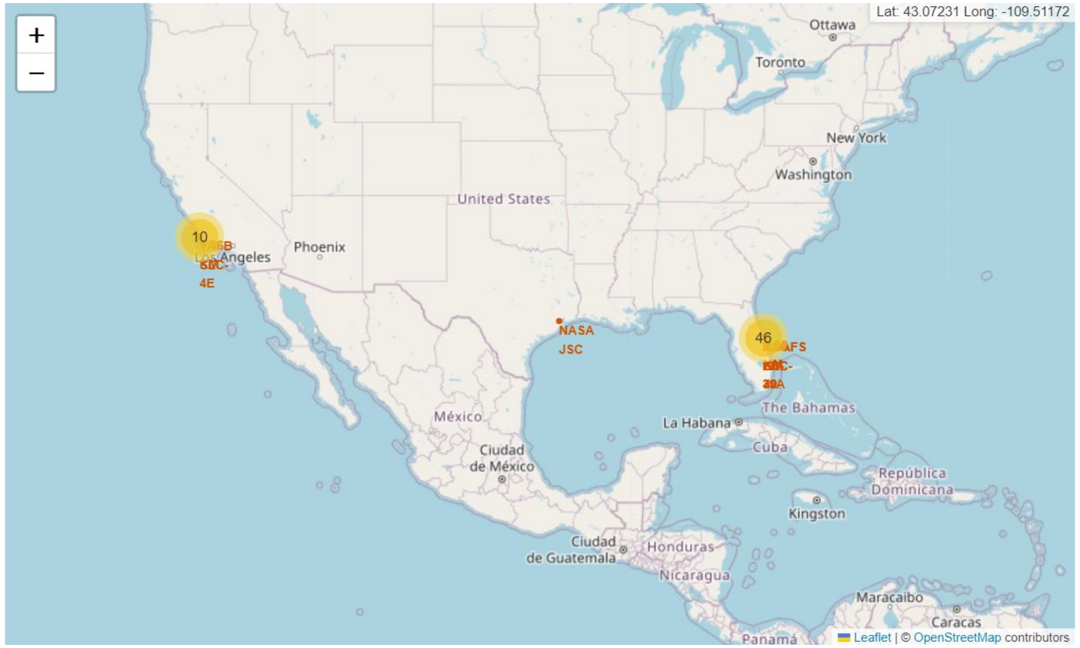
https://github.com/AfterRain007/TFDS/blob/main/Finale/8%20SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

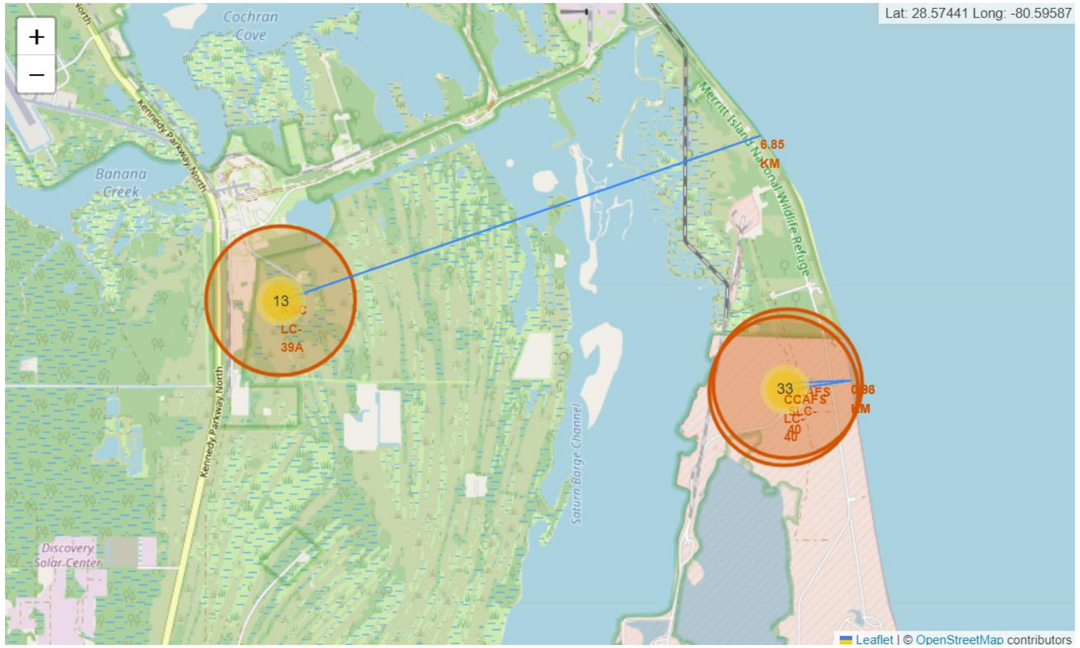
Results

- Exploratory data analysis results
  1. There are 4 launch site used by spaceX which are CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, and CCAFS SLC-40
  2. The total payload masss done by SpaceX are 107010 kg with an average of 2534 kg
  3. The first success landing outcome happened in 2015
  4. SpaceX have a 99% success rate.

Results

- Interactive analytics demo in screenshots





Results

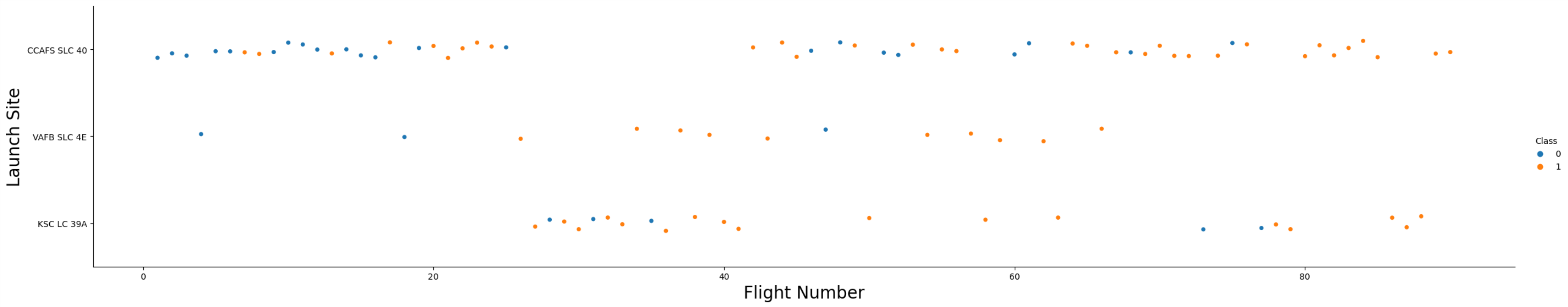
- Predictive analysis results

Comparing 4 model (logreg, SVM, Decision Tree, and KNN), it’s found that using model logreg is the best based on this dataset with an accuracy score of 94.44%.

Section 2

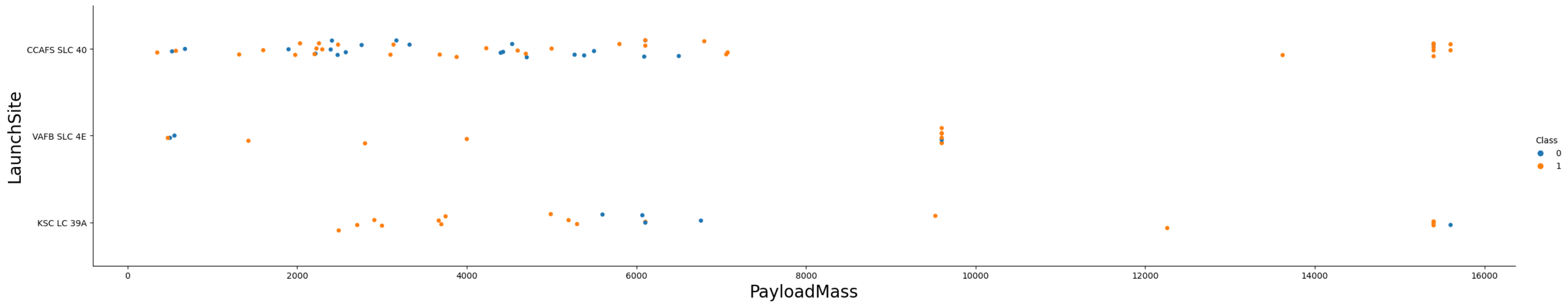
Insights drawn from EDA

Flight Number vs. Launch Site



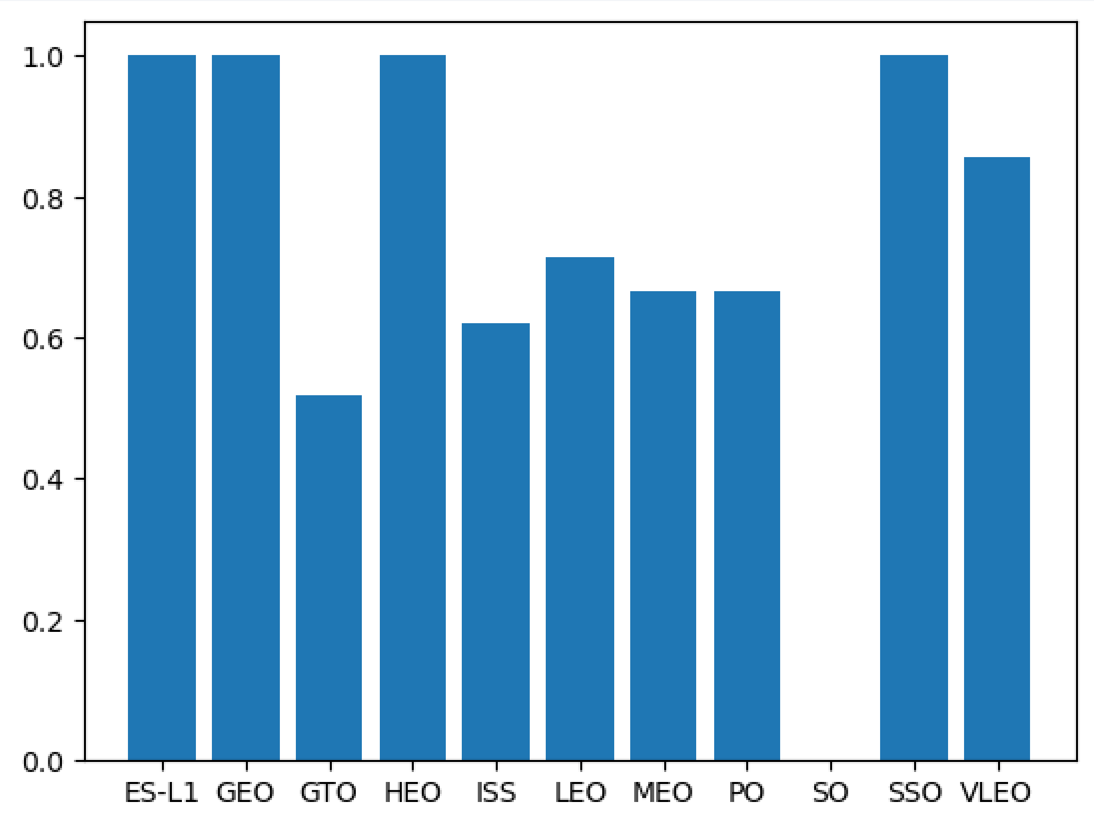
Based on the plot above, it is found that as the number of flights increased at a launch site, the success rate also increase.

Payload vs. Launch Site



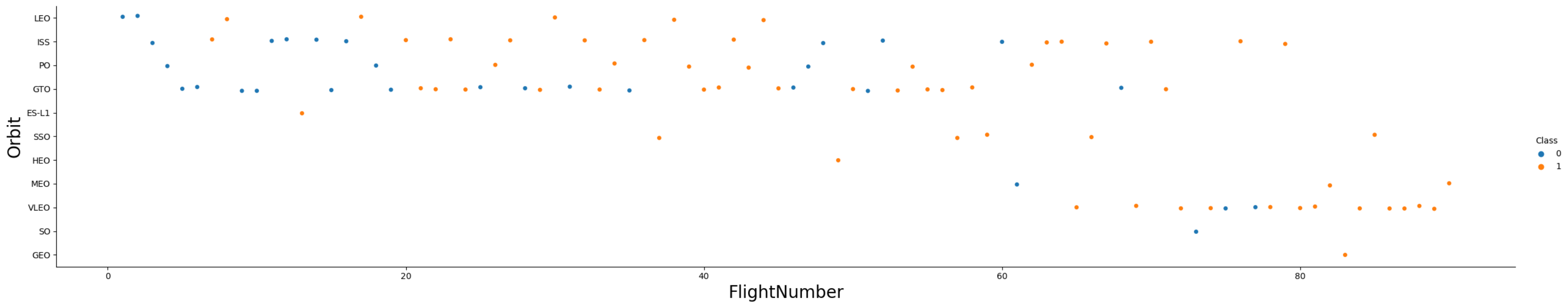
A correlation was observed indicating that a higher payload mass at launch site CCAFS SLC 40 corresponds to a higher success rate for the rocket.

Success Rate vs. Orbit Type



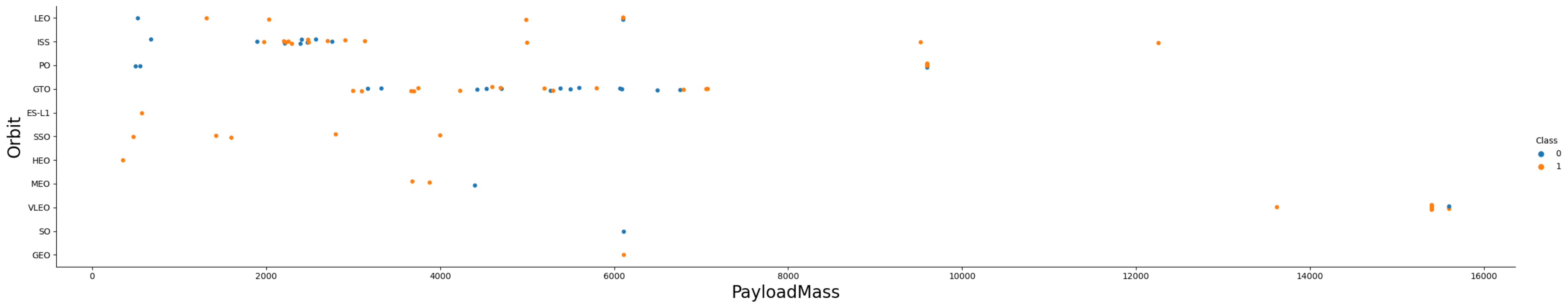
The plot highlights that missions to ES-L1, GEO, HEO, and SSO orbits exhibited the highest success rates.

Flight Number vs. Orbit Type



In the LEO orbit, success appears to be correlated with the number of flights, whereas in the GTO orbit, there is no discernible relationship between the number of flights and the orbit's success rate.

Payload vs. Orbit Type



It can be observed that with heavier payloads, successful landings are more common for PO, LEO, and ISS orbits.

Launch Success Yearly Trend

According to the plot, the success rate has steadily increased from 2013 to 2020.

All Launch Site Names

In [9]:
```
%%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

There are 4 Launch Site,

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

Launch Site Names Begin with 'CCA'

```
%%sql

SELECT *
FROM SPACEXTABLE
WHERE Launch_Site LIKE ("CCA%")
LIMIT 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_(
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (par
2010-12-08	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 (par

The query above is to showcase five records where the launch sites start with "CCA".

Total Payload Mass

```
%%sql

SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Customer LIKE ("%NASA%")
```

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

SUM(PAYLOAD_MASS__KG_)
107010

It is found that the sum of mass in payload are 107.010 kg

Average Payload Mass by F9 v1.1

- With an average payload mass of 2534.66 KG

```
%%sql

SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Booster_Version LIKE ("%F9 v1.1%")
```

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

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

```
%%sql

SELECT Date
FROM SPACEXTABLE
WHERE Date = (SELECT MIN(date) FROM SPACEXTABLE WHERE Landing_Outcome LIKE ("Success (ground pad)"))
```

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

Date
2015-12-22

With the first Successful ground landing on 22nd of December 2015

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql

SELECT DISTINCT Booster_Version
FROM SPACEXTABLE
WHERE Landing_Outcome LIKE ("%drone ship%")
AND Mission_Outcome LIKE ("Success")
AND PAYLOAD_MASS__KG_ < 6000 AND PAYLOAD_MASS__KG_ > 4000
```

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

Booster_Version
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

- It is found that there are 5 Successful Drone Ship Landing with Payload between 4000 and 6000. Which are:

  1. F9 FT B1020
  2. F9 FT B1022
  3. F9 FT B1026
  4. F9 FT B1021.2
  5. F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- The total number of Successful Mission Outcome Exceed the Failed one with the ratio of 100:1

```
%%sql

SELECT DISTINCT Mission_Outcome, COUNT(Mission_Outcome)
FROM SPACEXTABLE
WHERE Mission_Outcome LIKE ("%Success%")
UNION
SELECT DISTINCT Mission_Outcome, COUNT(Mission_Outcome)
FROM SPACEXTABLE
WHERE Mission_Outcome LIKE ("%Failure%")
```

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

Mission_Outcome	COUNT(Mission_Outcome)
Failure (in flight)	1
Success	100

Boosters Carried Maximum Payload

```
%%sql

SELECT Booster_Version, PAYLOAD_MASS__KG_
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE ORDER BY PAYLOAD_MASS__KG_ DE
LIMIT 1
```

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

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600

- Booster with the heaviest payload fell to F9 B5 B1048.4 with payload of 15600

2015 Launch Records

```
%%sql

SELECT  substr(Date, 6, 2) as Month, Landing_Outcome, Booster_Version, Launch_Site
FROM SPACEXTABLE
WHERE substr(Date, 0, 5) == "2015"
AND Landing_Outcome LIKE ("%Failure%drone ship%")
```

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

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

There are 2 failure in the year of 2015. And it was carried by F9 v1.1 B1012 and F9 v1.1 B10915 Booster, with both booster launched in the Launch Site of CCAFS LC-40

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

```
%%sql

SELECT DISTINCT Landing_Outcome, COUNT(Landing_Outcome) as Counts
FROM SPACEXTABLE
WHERE Date > "2010-06-04"
AND Date < "2017-03-20"
GROUP BY Landing_Outcome
ORDER BY Counts DESC
```

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

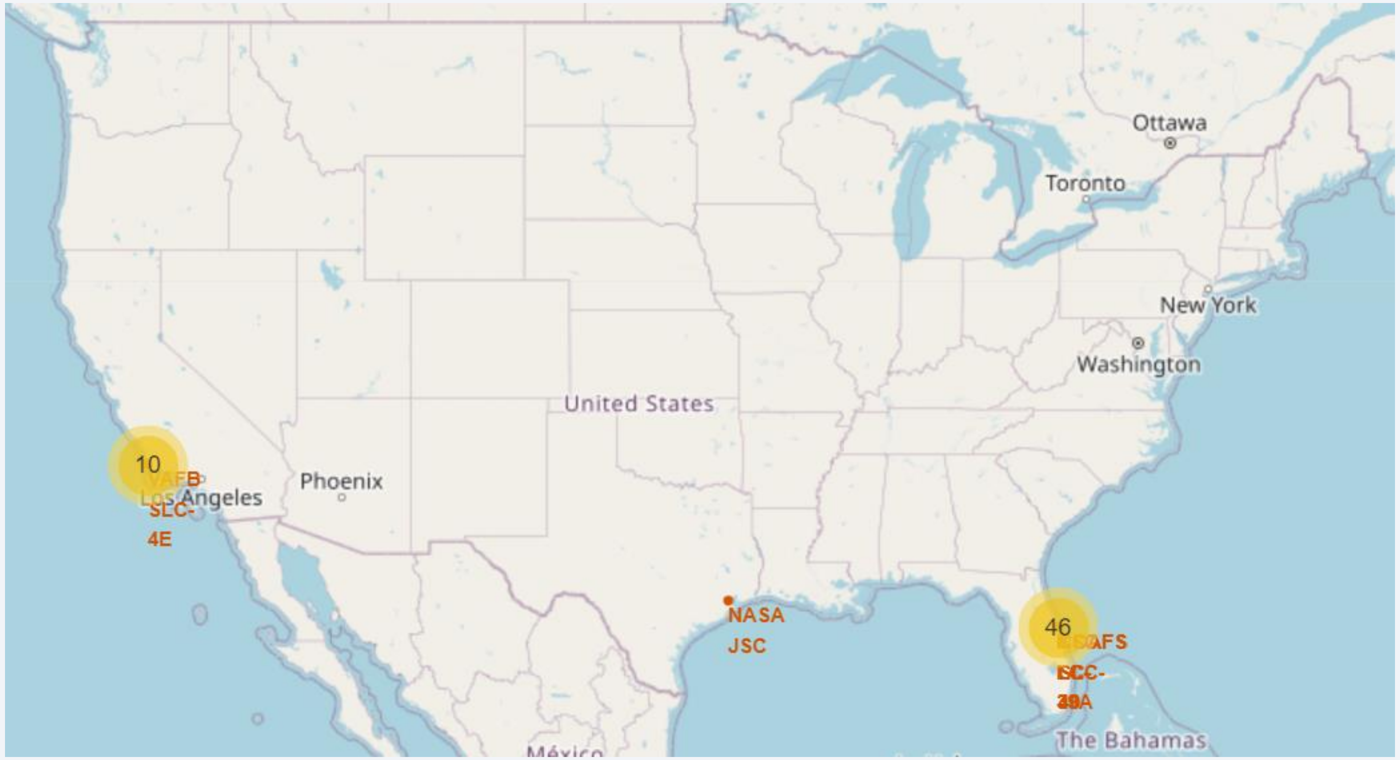
Landing_Outcome	Counts
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

Between 2010-06-04 and 2017-03-20 it is found there are 10 counts of no attempt, followed with success on drone ship tied with failure on drone ship.

Section 3

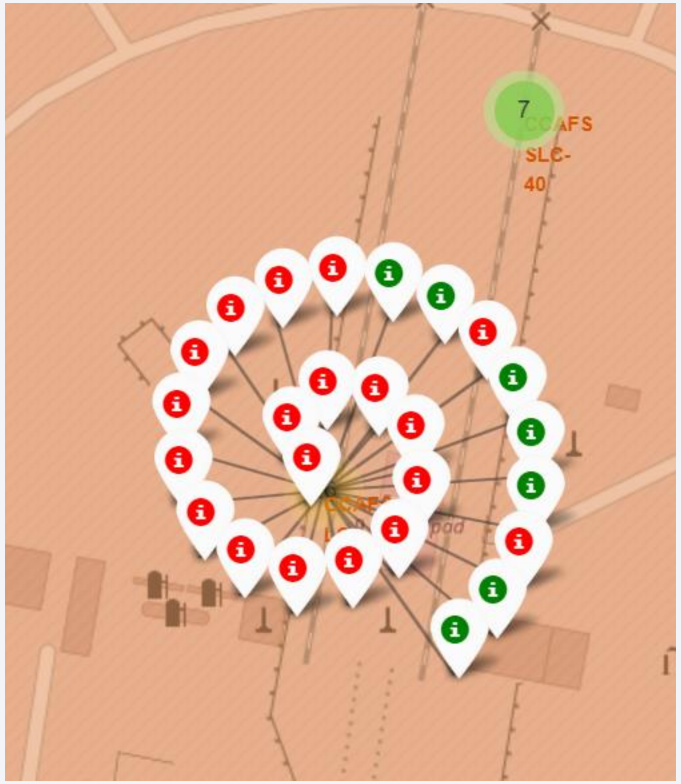
Launch Sites Proximities Analysis

<Launch Site>



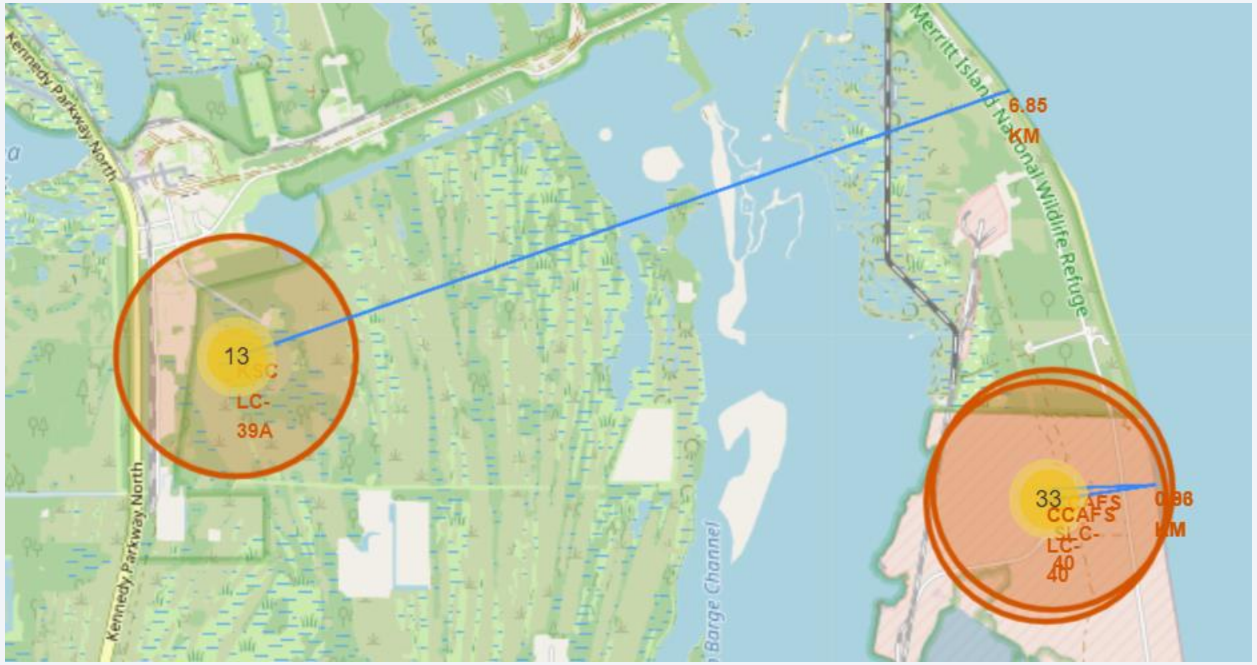
Using Folium we are able to put marker and circle to the launch site, with addition of sum of launch has been done on each launch site

<Successful and Failure Marker>



We can also see how many lunch has been a success or failure on each launch site

<Distance>

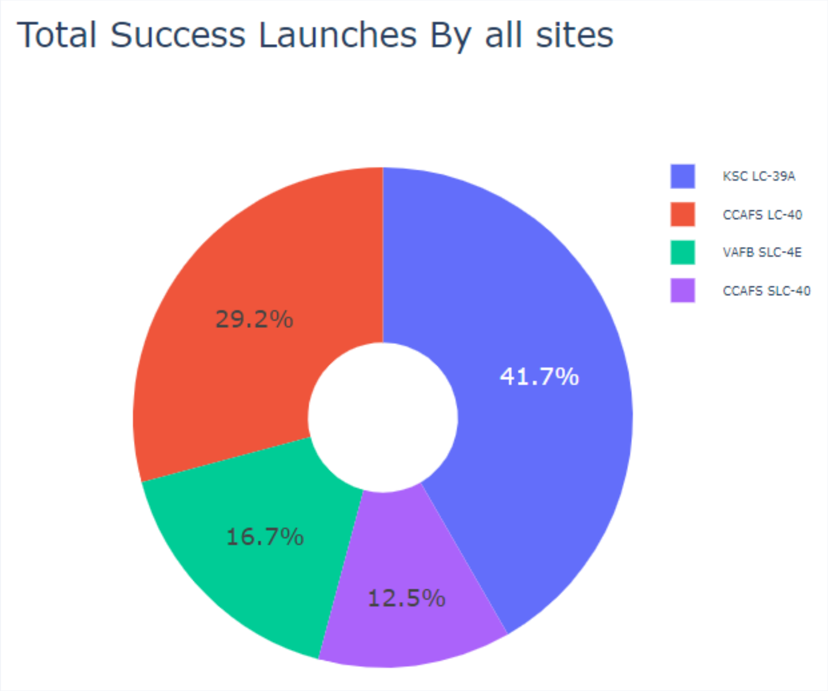


Using folium, we could also put a line to see how far is the distance from one coordinate to another, here an example of launch site to the nearest coastline

Section 4

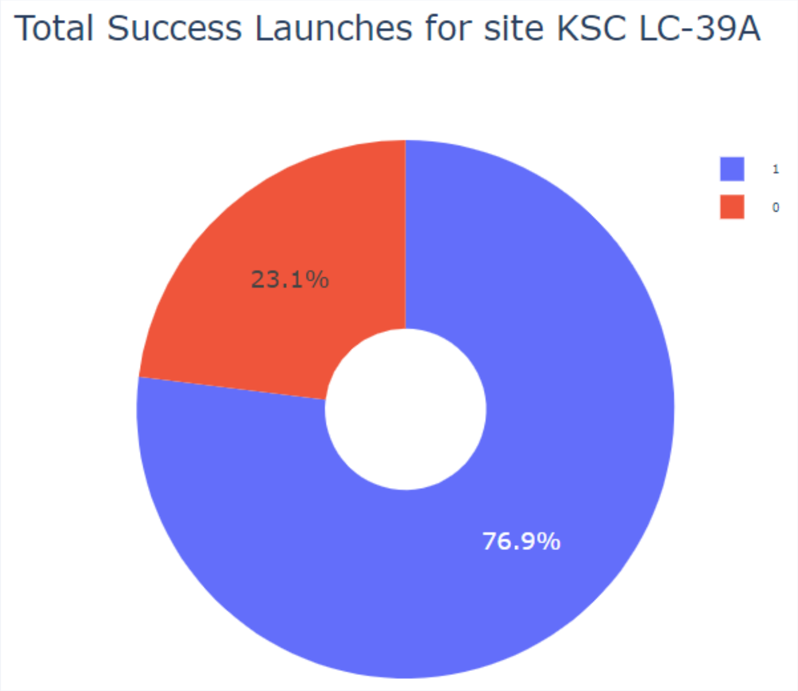
Build a Dashboard with Plotly Dash

<Pie Chart Success Percentage by Each Launch Site>



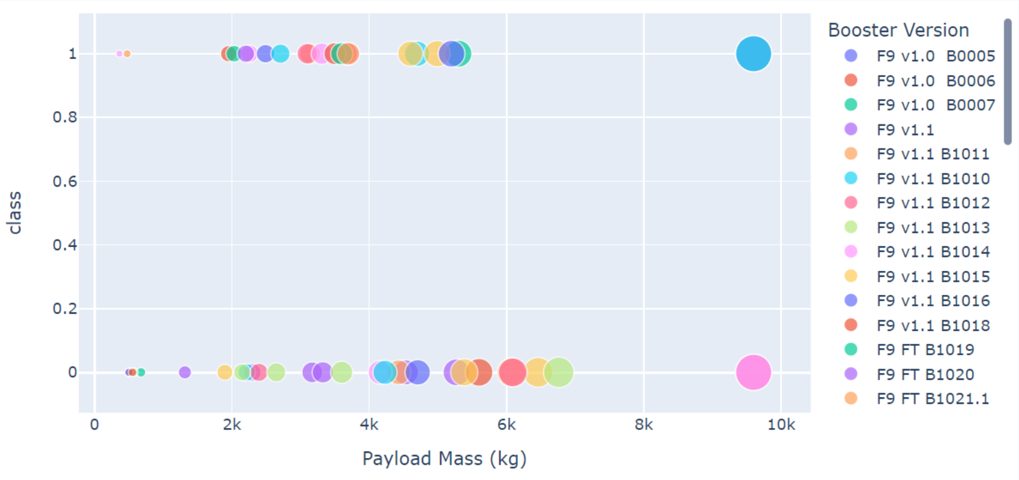
From the pie chart we can conclude that KSC LC-39A have the highest percentage of Success

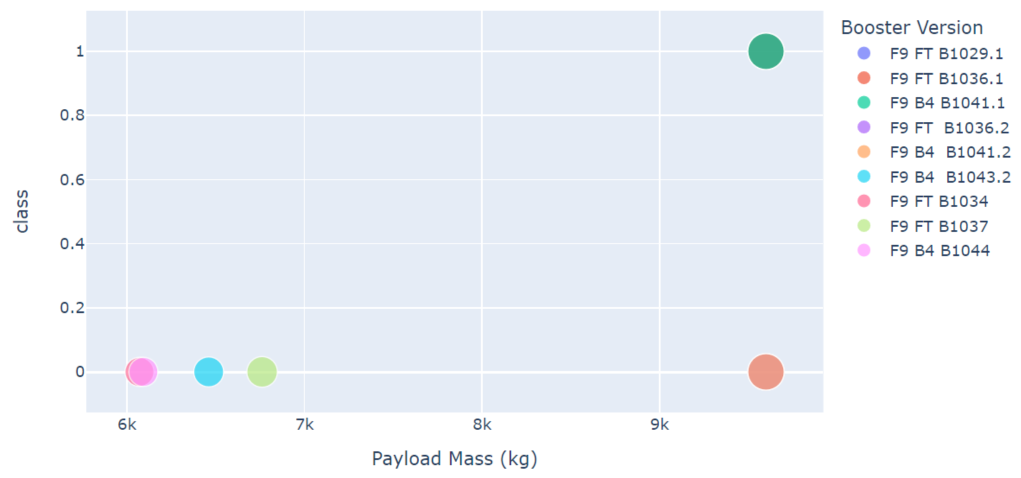
<Piechart for the Launch Site with Highest Launch Success Ratio>



From the pie chart we can conclude that KSC LC-39A have success rate of 76.9%

<Payload Vs. Launch Outcome Scatter Plot>





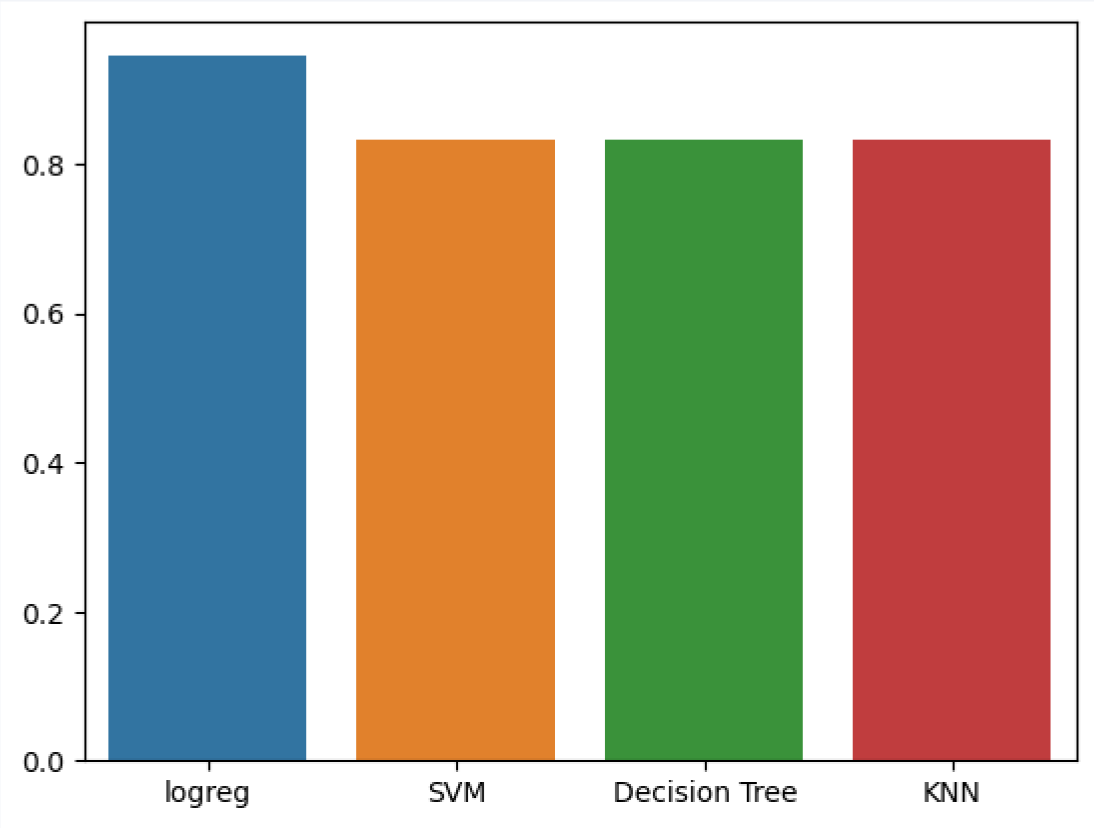
From the scatter plot we can can conclude that the more heavy the payload, the lower chances of success

Section 5

Predictive Analysis (Classification)

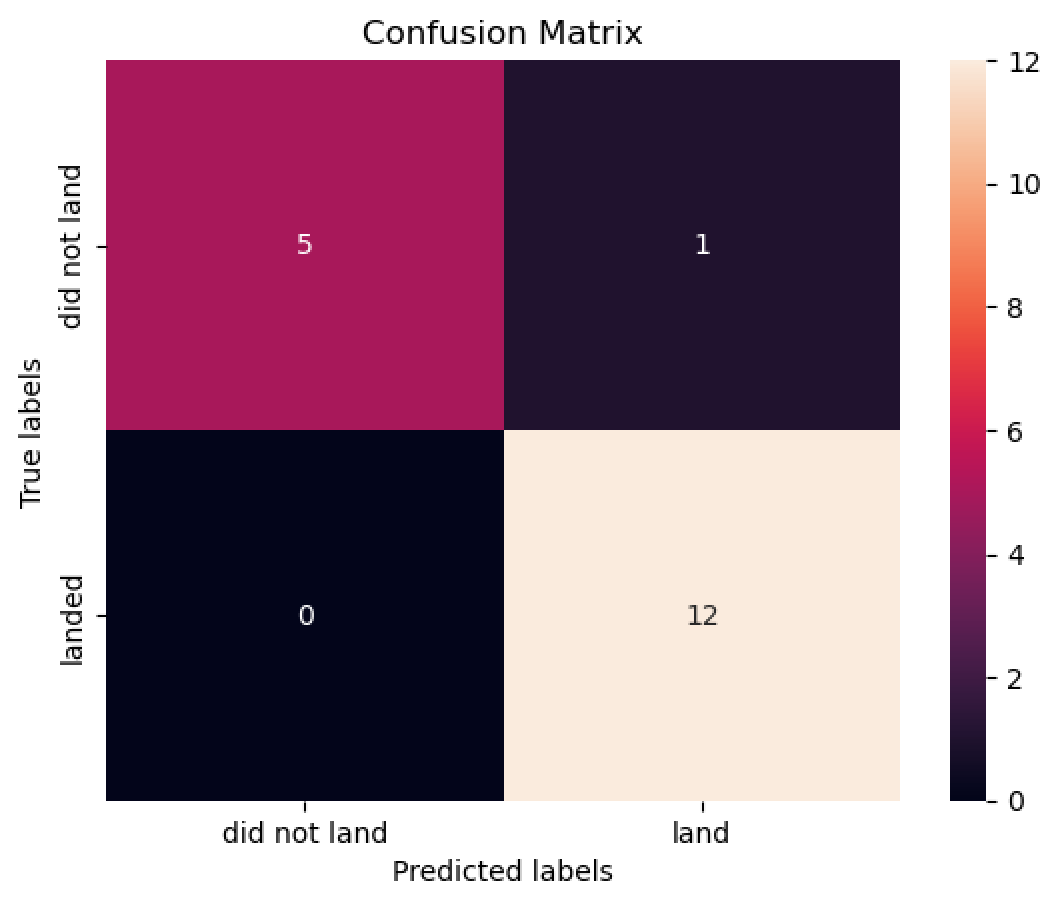
Classification Accuracy

As we can see, Classification Accuracy for the model logreg score higher than any other model. So we can conclude logreg is the best model



Confusion Matrix

With the LogReg model Confusion matrix almost scoring a perfect score. With only one False Positive, while the other data point scoring according the test data.



Conclusions

In summary, the following conclusions can be drawn:

- There is a positive correlation between the number of flights at a launch site and the success rate.
- The launch success rate has shown a continuous increase from 2013 to 2020. With a slight hiccup in 2018.
- Orbits ES-L1, GEO, HEO, SSO, and VLEO demonstrated the highest success rates.
- KSC LC-39A stands out as the launch site with the highest number of successful launches.
- Among the machine learning algorithms evaluated, the LogReg model performed the best for this task.

Thank you!