Rocketry - The New Space Age

IBM Developer

Arunava Patra - Aug 2024

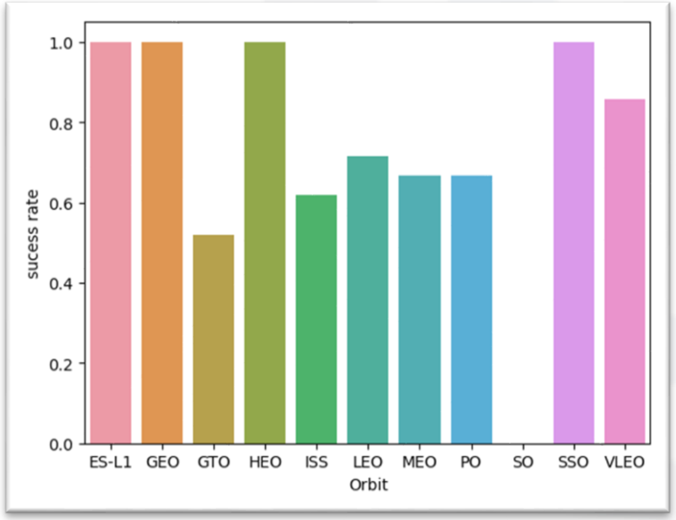
SKILLS NETWORK

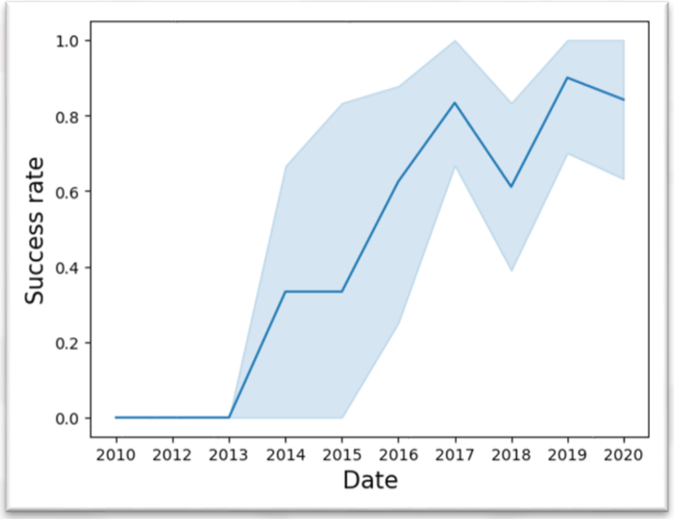
OUTLINE

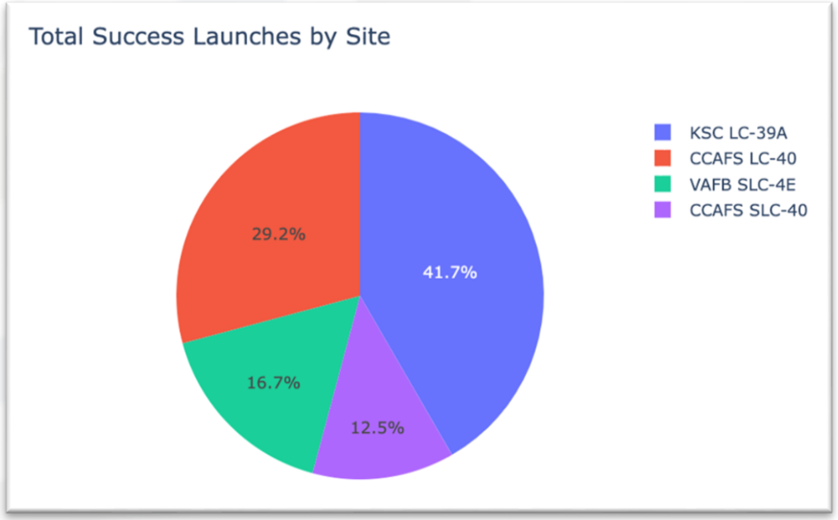
- Executive Summary
- Introduction
- Methodology
- Results
- Discussion
- Conclusion

EXECUTIVE SUMMARY

- Acquired SpaceX datasets through a data collection API and web scrapping.
- Employed exploratory data analysis techniques to analyse and visualise the acquired data.
- Utilising the Grid Search method to identify the most effective Machine Learning Model for predicting the classification of future landings.







INTRODUCTION

Given SpaceX's ability to reuse the first stage, the Falcon 9 rocket launches at a cost of **62** million dollars, significantly lower than the often-higher cost of up to **165** million dollars for rockets from other providers.

Therefore, by accurately predicting the successful landing of the first stage, we can estimate the cost of a launch. This predictive capability holds value for potential competitors bidding against SpaceX for rocket launches.

Key Questions:

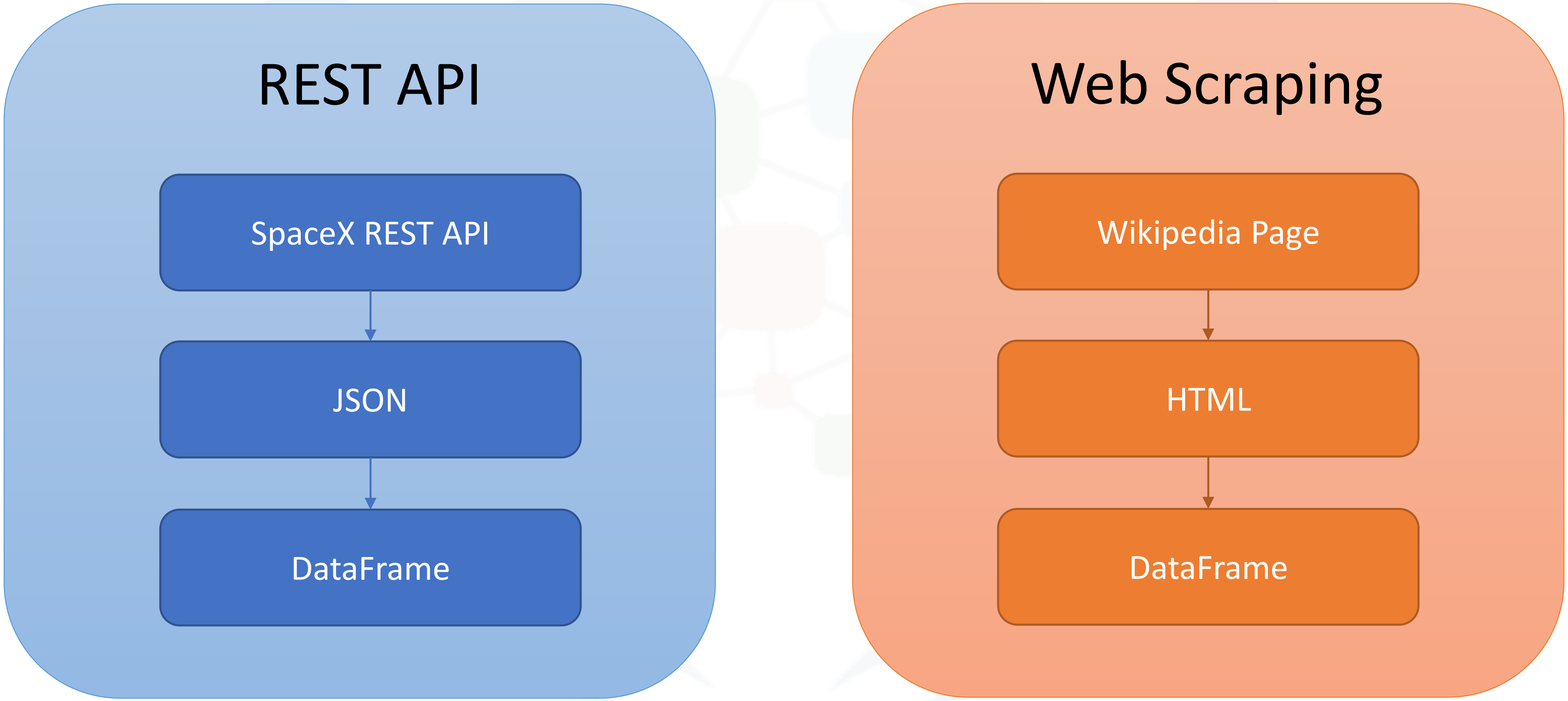
1. Can historical launch data help predict the success of a new launch based on the first stage's landing outcome?
2. Is there a discernible choice that maximises the likelihood of a successful launch based on historical data analysis?

METHODOLOGY

METHODOLOGY

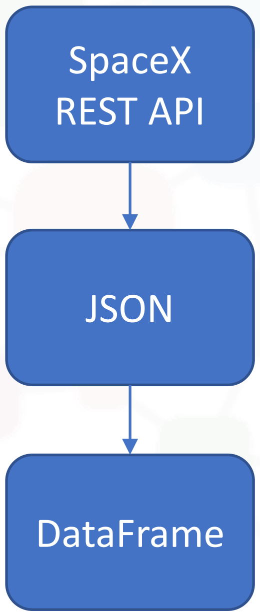
- Data collection methodology:
    - SpaceX REST API
    - Web Scraping (Wikipedia)
- Perform data wrangling:
    - Generate landing Class from Outcome column
- Perform exploratory data analysis (EDA) using visualization and SQL.
- Perform interactive visual analytics using Folium and Plotly Dash.
- Perform predictive analysis using classification models.
    - Using GridSearchCV to find best fit model

Data Collection



Data Collection – SpaceX API

- SpaceX API repository
  https://github.com/r-spacex/SpaceX-API
- Main Endpoint
  https://api.spacexdata.com/v4/launches/past
- My Notebook
  - https://github.com/arunava2508/SpacexCapstone




```python
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())
```

	static_fire_date_utc	static_fire_date_unix	net	window	rocket	success	failures	details	crew	ships	capsules	
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	[]	[]	[]	[5eb0e4b5b6c3bb000
1	None	NaN	False	0.0	5e9d0d95eda69955f709d1eb	False	[{'time': 301, 'altitude': 289, 'reason': 'harmonic oscillation leading to premature engine shutdown'}]	Successful first stage burn and transition to second stage, maximum altitude 289 km, Premature engine shutdown at T+7 min 30 s, Failed to reach orbit, Failed to recover first stage	[]	[]	[]	[5eb0e4b6b6c3bb000
2	None	NaN	False	0.0	5e9d0d95eda69955f709d1eb	False	[{'time': 140, 'altitude': 35, 'reason': 'residual stage-1 thrust led to collision between	Residual stage 1 thrust led to collision between stage 1 and stage 2	[]	[]	[]	[5eb0e4b6b6c3bb000 5eb0e4b6b6c3bb000

Data Collection – Web Scraping

- Wikipedia Falcon Page

  https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

- My Notebook

https://github.com/arunava2508/SpacexCapstone

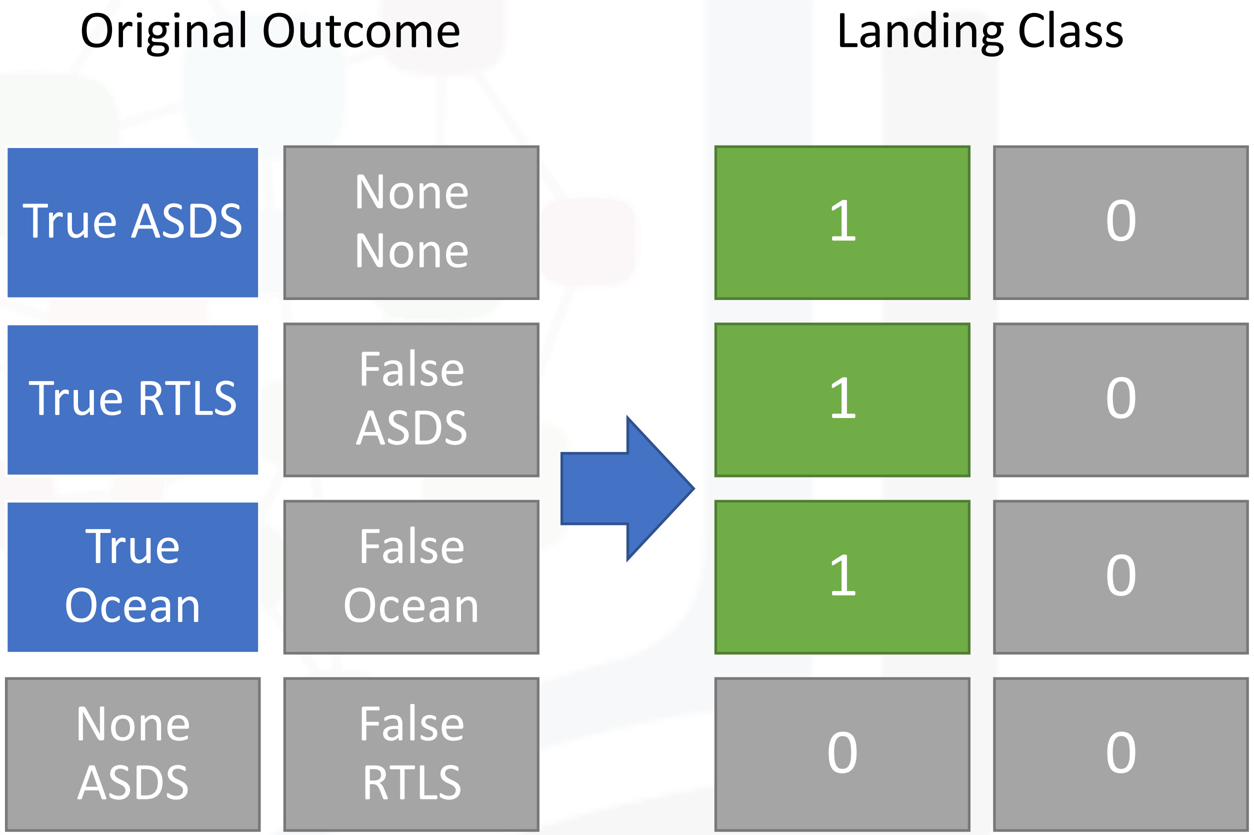

```python
import requests
from bs4 import BeautifulSoup
url =
'https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches'
response = requests.get(url)
html_data = response.text
soup = BeautifulSoup(html_data)
```

```html
<tr>
<th scope="col">Flight No.
</th>
<th scope="col">Date and<br/>time (<a href="/wiki/Coordinated_Universal_Time" title="Coordin
Time">UTC</a>)
</th>
<th scope="col"><a href="/wiki/List_of_Falcon_9_first-stage_boosters" title="List of Falcon
boosters">Version,<br/>Booster</a> <sup class="reference" id="cite_ref-booster_11-0"><a href
[b]</a></sup>
</th>
<th scope="col">Launch site
</th>
<th scope="col">Payload<sup class="reference" id="cite_ref-Dragon_12-0"><a href="#cite_note-
</th>
<th scope="col">Payload mass
</th>
<th scope="col">Orbit
</th>
<th scope="col">Customer
</th>
<th scope="col">Launch<br/>outcome
</th>
<th scope="col"><a href="/wiki/Falcon_9_first-stage_landing_tests" title="Falcon 9 first-sta
tests">Booster<br/>landing</a>
</th></tr>
```

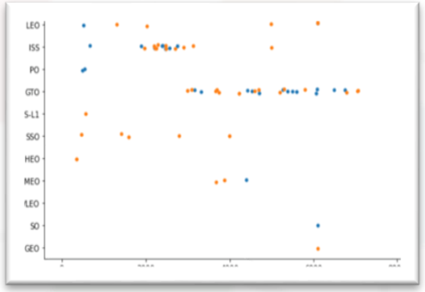
Data Wrangling

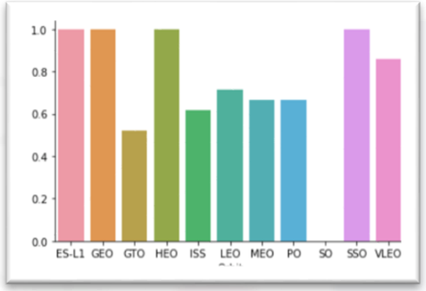
- My Notebook
  https://github.com/arunava2508/SpacexCapstone

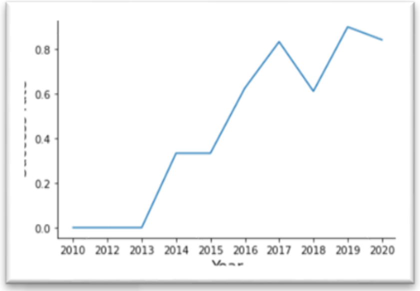
- Refine the raw data by creating a new landing classification column based on the original outcome labels. This column will serve as our target for predicting landing success, denoted as:
  - 1 for success
  - 0 for failure



EDA with Data Visualisation







- My Visualization Notebook
  - https://github.com/arunava2508/SpacexCapstone

Scatter Plot	Acquire relationship between variables, Flight Number vs. Orbit type Payload vs. Orbit type Flight Number vs. Payload Mass Flight Number vs. Launch Site
Bar Plot	Plot success rate of each orbit
Line Chart	Acquire yearly average launch success trend

EDA with SQL

- My SQL Notebook
  https://github.com/arunava2508/SpacexCapstone

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

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

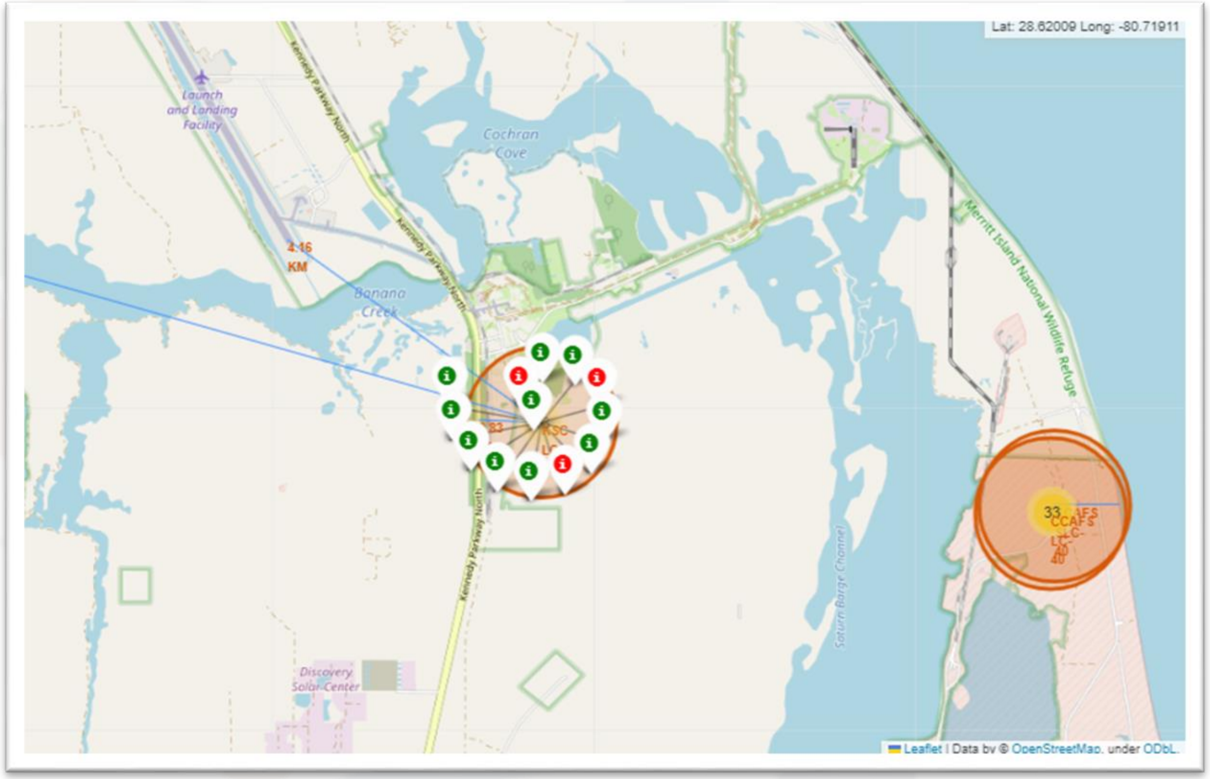
Landing_Outcome	landings
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

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

- ✓ Query the names of the **unique launch sites** in the space mission
- ✓ Query the names of the **booster_versions** which have carried the maximum payload mass.
- ✓ List the total number of **successful** and **failure** mission outcomes
- ✓ List the names of the boosters which have **success in drone ship** and have **payload mass** in some range
- ✓ Rank the count of successful **landing_outcomes** in date range in descending order.

Interactive Map with Folium

- Add **Circles** for Launch sites and **Markers** for labels
- Add **MarkerCluster** for successful and failed launches
- Add **Lines** for calculate distance between launch sites and their proximities

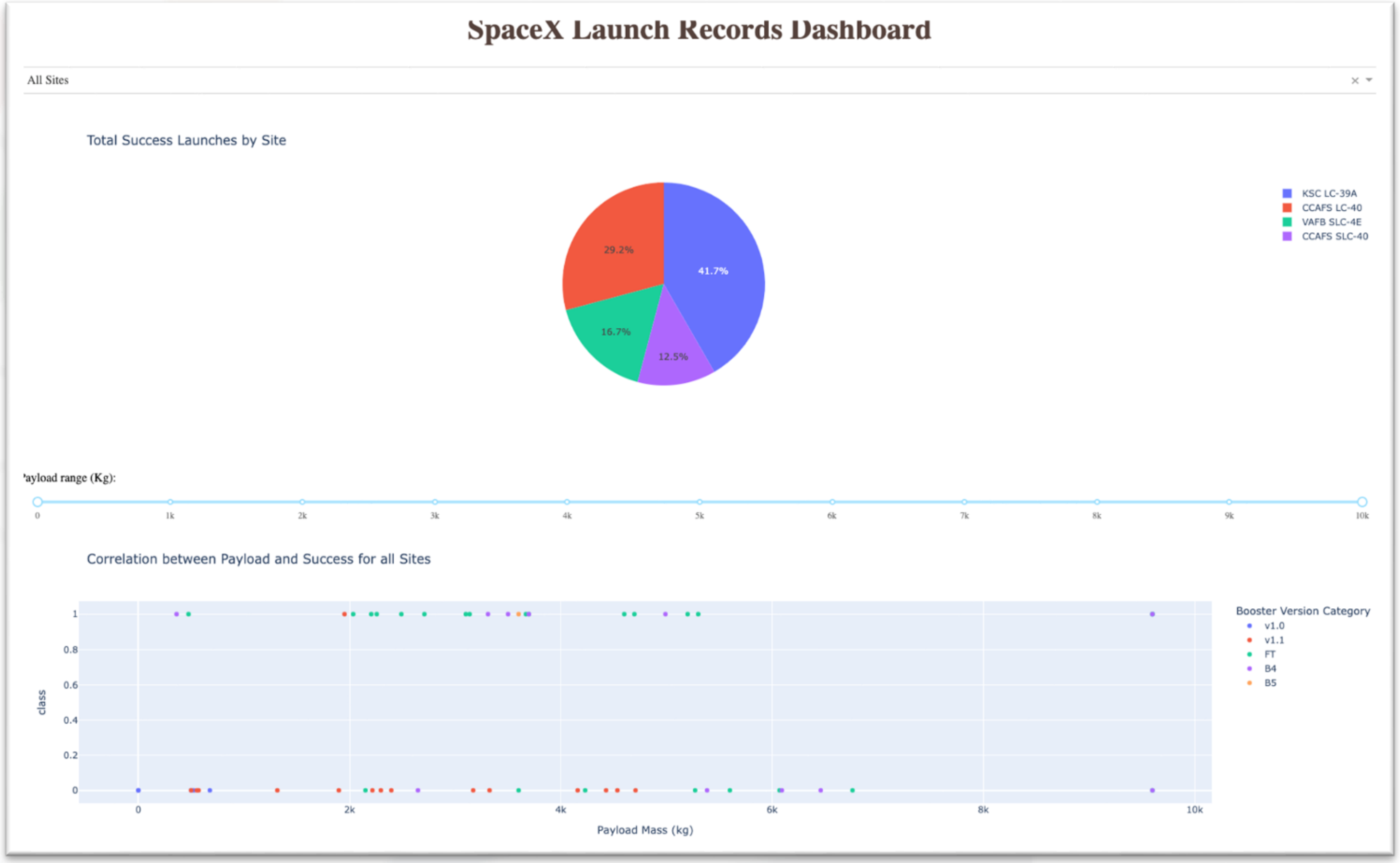


- My SQL Notebook
- https://github.com/arunava2508/SpacexCapstone

Build Dashboard with Plotly Dash

- **Dropdown menu + Pie Chart:** Visualises success launch distribution per launch site.
- **Range Slider + Scatter Plot:** Analyses Payload vs. Success correlation across diverse launch sites.

- My SQL Notebook
- https://github.com/arunava2508/SpacexCapstone



Predictive Analysis(Classification)

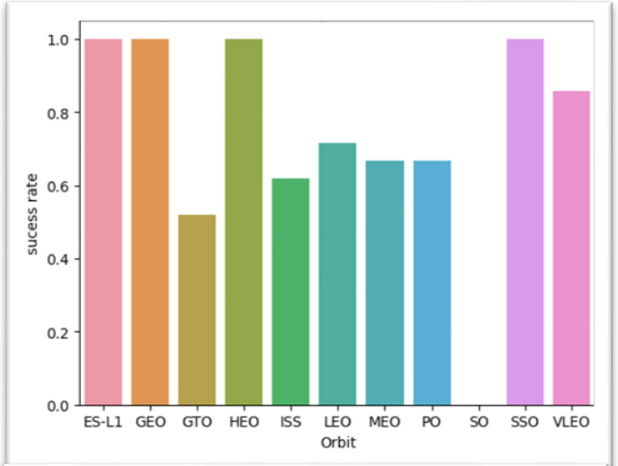
- **Prepare** data
- **Create** a column for the class
- **Standardize** the data
- **Split** into training data and test data
- **Define** model and parameters
- **Train** and **Grid Search** for best parameters
- **Evaluation**

Preprocess Data	Define Model	Train Model	Evaluate Model
Data Cleaning	Choose Model	Tune Model	Accuracy Analysis
Data Transformation	Model Parameters	Grid Search	Model Plot
Dataset Splitting			

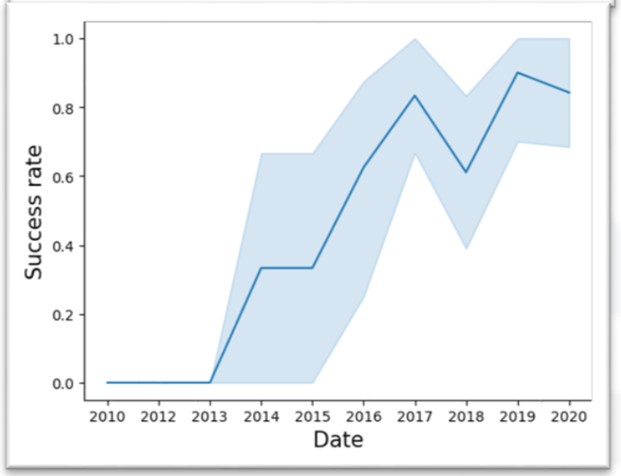
My Notebook

https://github.com/arunava2508/SpacexCapstone

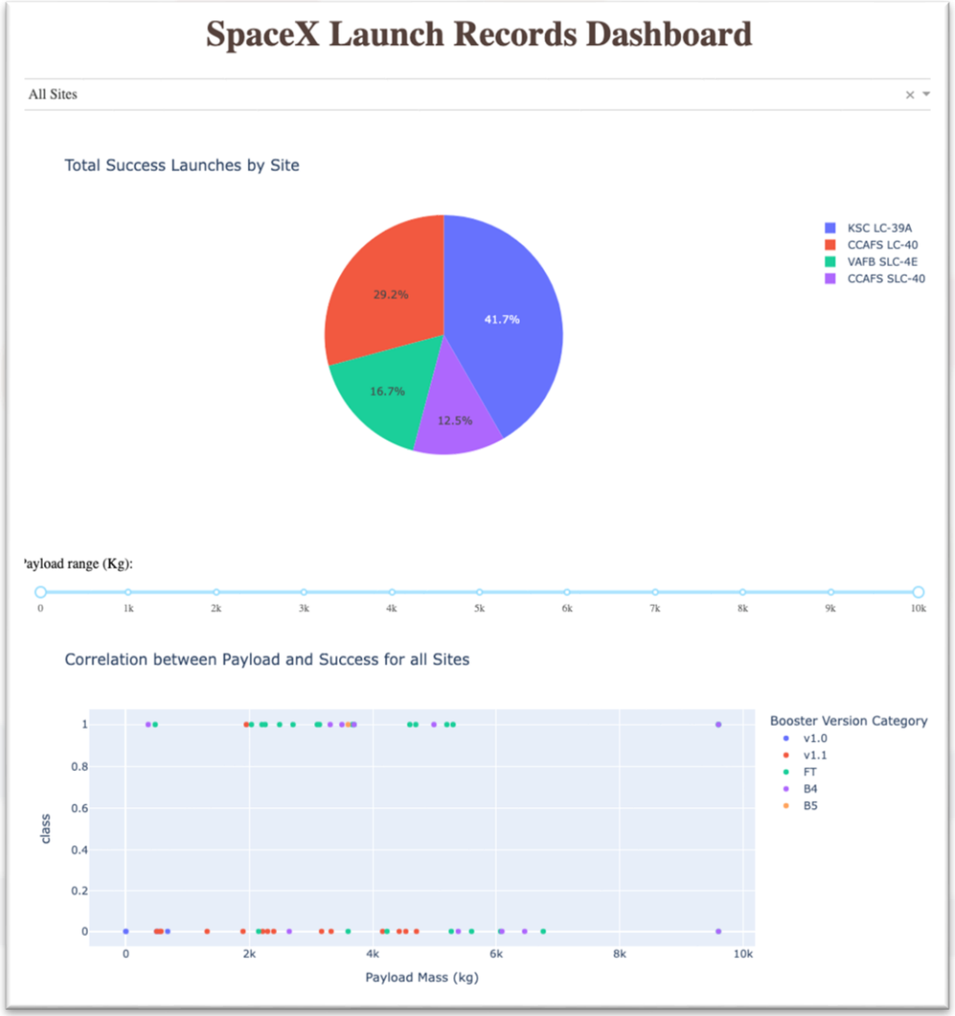
RESULTS



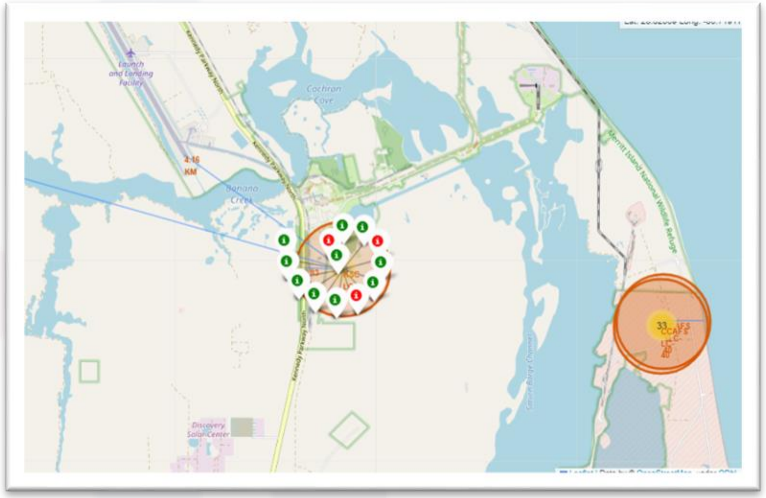
sucess rate
Orbit
ES-L1
GEO
GTO
HEO
ISS
LEO
MEO
PO
SO
SSO
VLEO

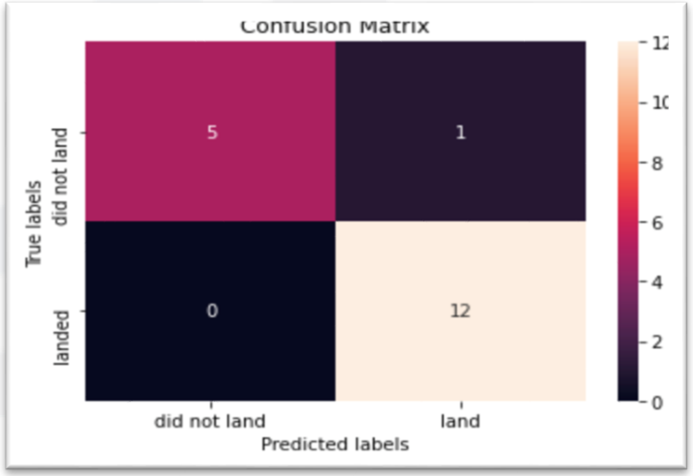


Success rate
Date



SpaceX Launch Records Dashboard
All Sites
Total Success Launches by Site
KSC LC-39A
CCAFS LC-40
VAFB SLC-4E
CCAFS SLC-40
41.7%
29.2%
16.7%
12.5%
Payload range (Kg):
Correlation between Payload and Success for all Sites
Booster Version Category
v1.0
v1.1
FT
B4
B5
class
Payload Mass (kg)



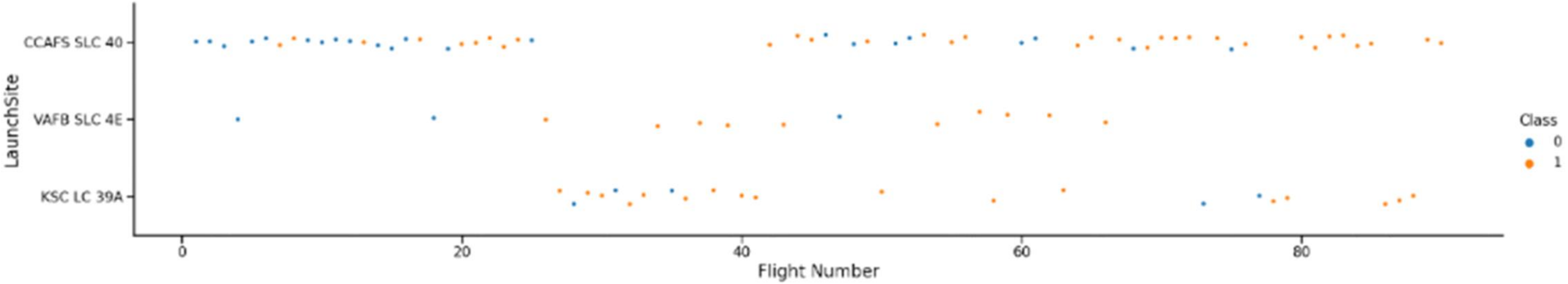


Confusion Matrix
True labels
did not land
landed
Predicted labels
did not land
land
5
1
0
12

INSIGHT DRAWN FROM EDA

Flight Number VS Launch Site

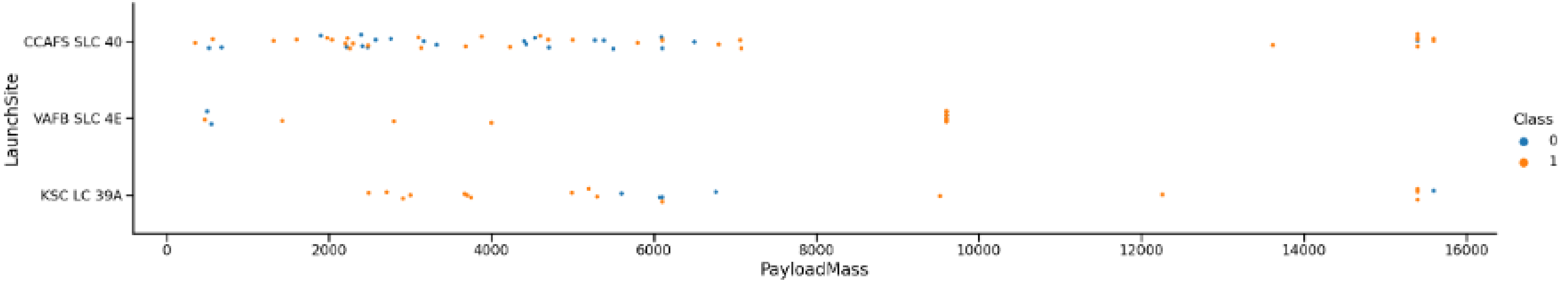
In [39]:
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("LaunchSite",fontsize=20)
plt.show()
```



Explanation: The scatter plot demonstrates a correlation between the flight number and successful first stage landings. As the flight number increases, there is a noticeable increase in successful landings. Initially, launches are more frequent at the CCAFS SLC 40 site with a comparatively lower success rate. However, fewer launches occur at the VAFB SLC 4E and KSC LC 39A sites, where a higher success rate is observed.

Payload VS Launch Site

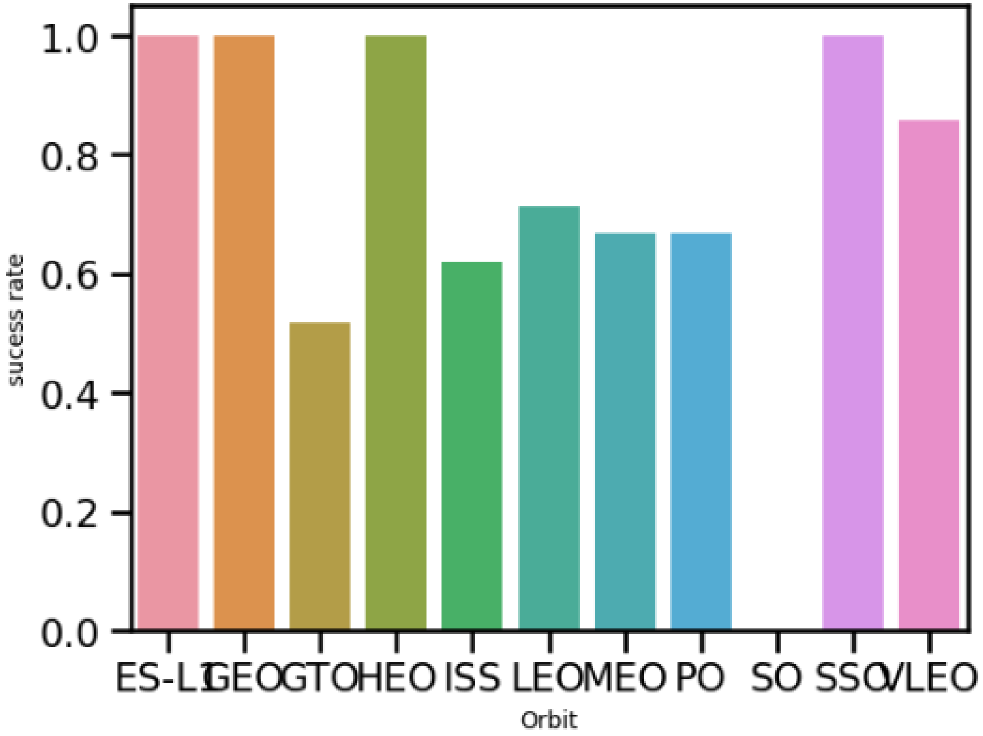
In [40]:
```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("PayloadMass",fontsize=20)
plt.ylabel("LaunchSite",fontsize=20)
plt.show()
```



Explanation: The success rate significantly increases with a higher payload. Specifically, at the KSC LC39A launch site, there's a notably higher success rate associated with lower payloads compared to a substantially lower success rate at the CCAFS SLC 40 launch site. Additionally, there have been no rockets launched at the VAFB-SLC site for payloads exceeding 10000. Furthermore, there is an observed very high success rate overall for payloads greater than 9500.

Success Rate VS Orbit Type

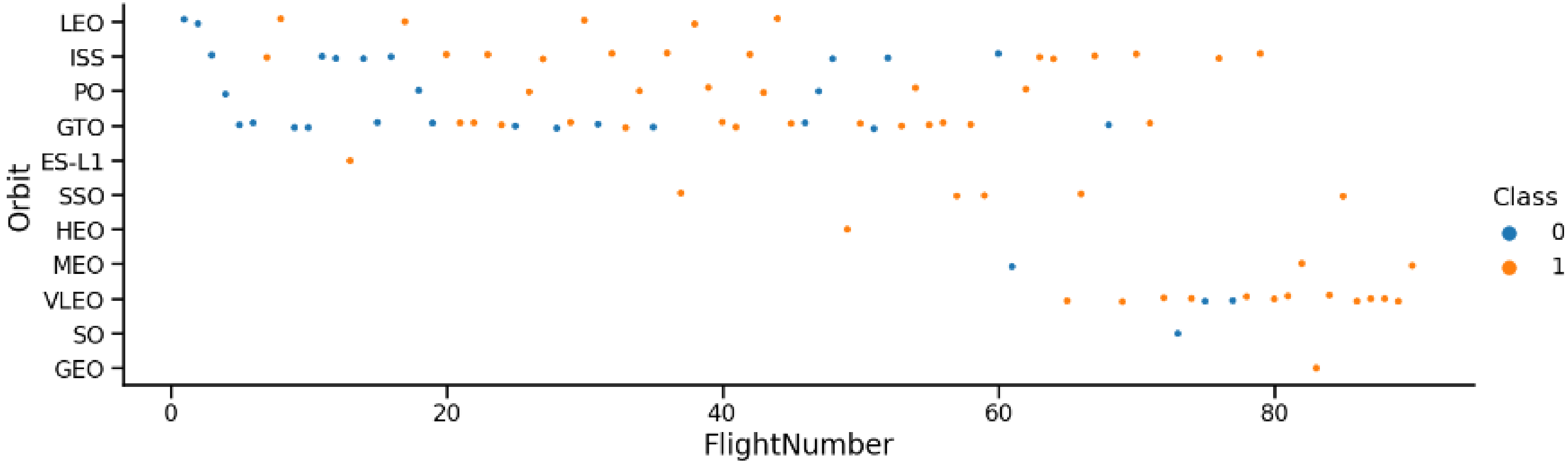
In [43]:
```
sns.barplot(y='Class', x='Orbit', data=df_success_rate)
plt.xlabel("Orbit",fontsize=10)
plt.ylabel("sucess rate",fontsize=10)
plt.show()
```



Explanation: The bar plot clearly indicates that Orbit types ES-L1, GEO, HEO, and SSO exhibit the highest success rate, all reaching 100%. Conversely, within the SO orbit type, the success rate is recorded as zero.

Flight Number VS Orbit Type

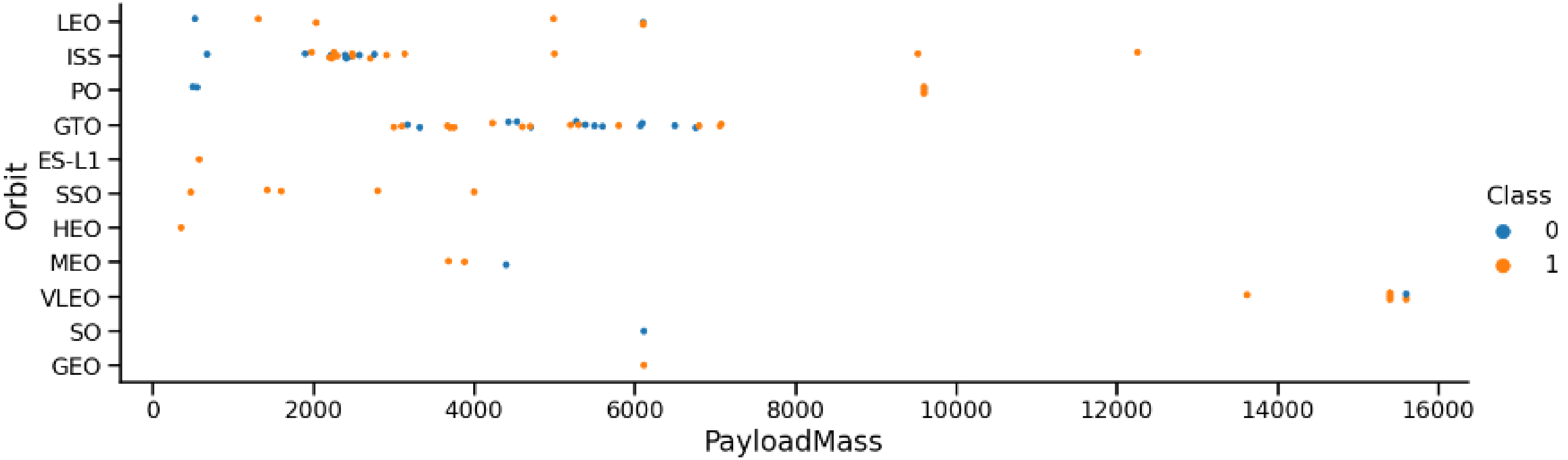
In [44]:
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df, aspect = 3)
plt.xlabel("FlightNumber",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



Explanation: In the ES-L1, GEO, HEO, and SSO orbits, every launch has resulted in a successful mission. Notably, there is a discernible correlation between the flight number and success rate in the LEO orbit, where an increase in flight number coincides with a rise in the success rate. However, this clear relationship is absent in the GTO orbit, where no obvious pattern between flight number and success rate can be observed.

Payload VS Orbit Type

In [45]:
```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
sns.catplot(y="Orbit", x="PayloadMass", hue="Class", data=df, aspect = 3)
plt.xlabel("PayloadMass",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



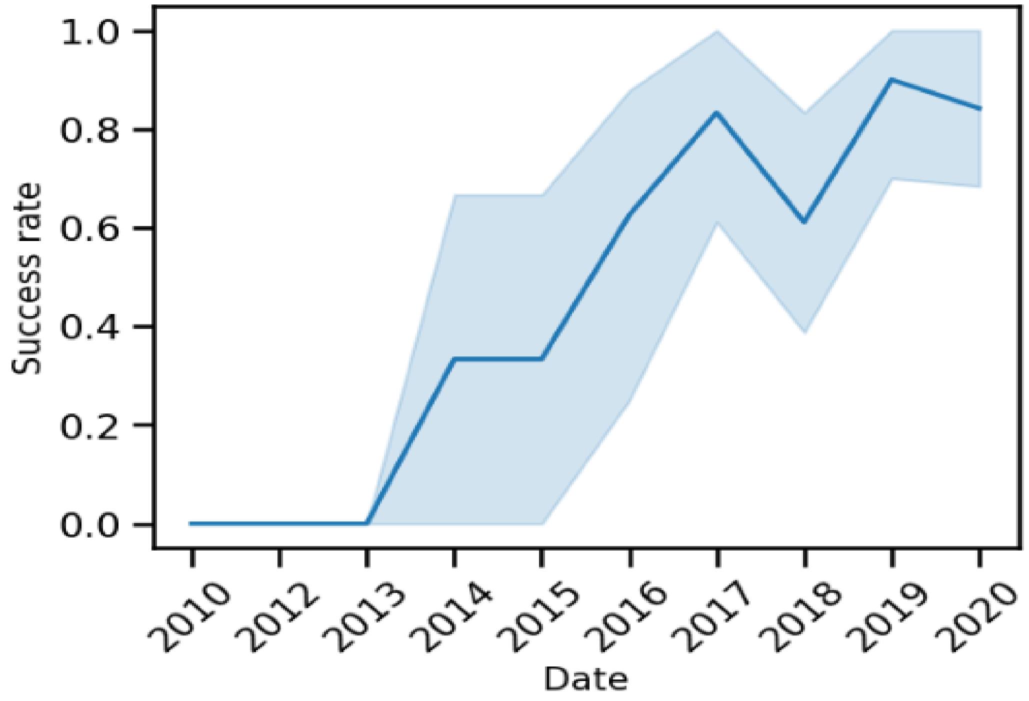
Explanation: For missions carrying heavy payloads, there is a higher frequency of successful or positive landings observed in the Polar, LEO, and ISS orbits.

IBM Developer

SKILLS NETWORK

Launch Success Yearly Trend

```
In [55]: # Plot a line chart with x axis to be the extracted year and y axis to be the success rate
sns.lineplot(y='Class', x='Date', data=df)
plt.xlabel("Date",fontsize=15)
plt.ylabel("Success rate",fontsize=15)
plt.xticks(rotation=45)
plt.show()
```



Explanation: The success rate, observed from 2013 through 2020, shows a consistent upward trend, steadily increasing over this period.

All Launch Site Names

Four Launch Sites:

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

```
In [30]: %%sql
select distinct Launch_Site from SPACEXTBL

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

Out[30]:

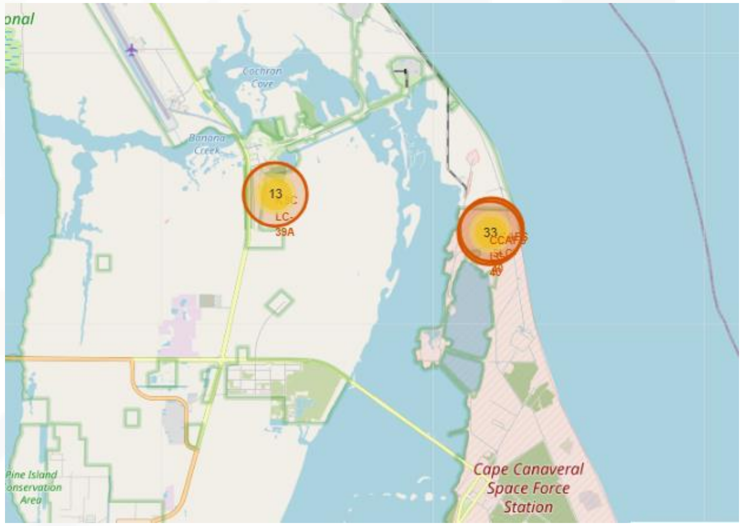
Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

1 in western coast

- VAFB SLC-4E

3 in eastern coast

- KSC LC-39A
- CCAFS SLC-40
- CCAFS LC-40



Launch Sites Names Begin with CAA

Display 5 records where launch sites begin with the string 'CCA'

In [31]:
```
%%sql
select * from SPACEXTBL
where Launch_Site like 'CCA%' LIMIT 5
```

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

Out[31]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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 (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

Display the total payload mass carried by boosters launched by NASA (CRS)

In [32]:
```
%%sql
Select sum(PAYLOAD_MASS__KG_) from SPACEXTBL
where Customer=='NASA (CRS)'
```

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

Out[32]:

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass carried by F9V1.1

Display average payload mass carried by booster version F9 v1.1

```
In [33]: %%sql
         select avg(PAYLOAD_MASS__KG_) from SPACEXTBL
         where Booster_Version like 'F9 v1.1%'

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

Out[33]:

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

First Saucerful Ground Pad Landing

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

In [34]:
```
%%sql
select min(Date) from SPACEXTBL
where "Landing_Outcome" = "Success (ground pad)"
```

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

Out[34]:

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

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

In [35]:
```
%%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.
```

Out[35]:

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

Total Number of Successful and Failure Mission Outcomes

Explanation:

- the total number of **successful** mission outcomes is **100**

- the total number of **failure** mission outcomes is **1**

```
In [37]: %%sql
select count(*) from SPACEXTBL
where "Mission_Outcome" like "Success%"

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

Out[37]:

count(*)
100

```
In [38]: %%sql
select count(*) from SPACEXTBL
where "Mission_Outcome" like "Failure%"

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

Out[38]:

count(*)
1

Boosters That Carried Maximum Payload

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

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

In [39]:
```
%%sql

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

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

Out[39]:

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

2015 Launch Records

```
In [40]: %%sql
select substr(Date,6,2) as Month, Booster_Version, Launch_Site from SPACEXTBL
where substr(Date,0,5)='2015' and "Landing_Outcome" = "Failure (drone ship)"

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

Out[40]:

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

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

Rank the count of landing outcomes between the date 2010-06-04 and 2017-03-20:

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

In [41]:

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

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

Out[41]:

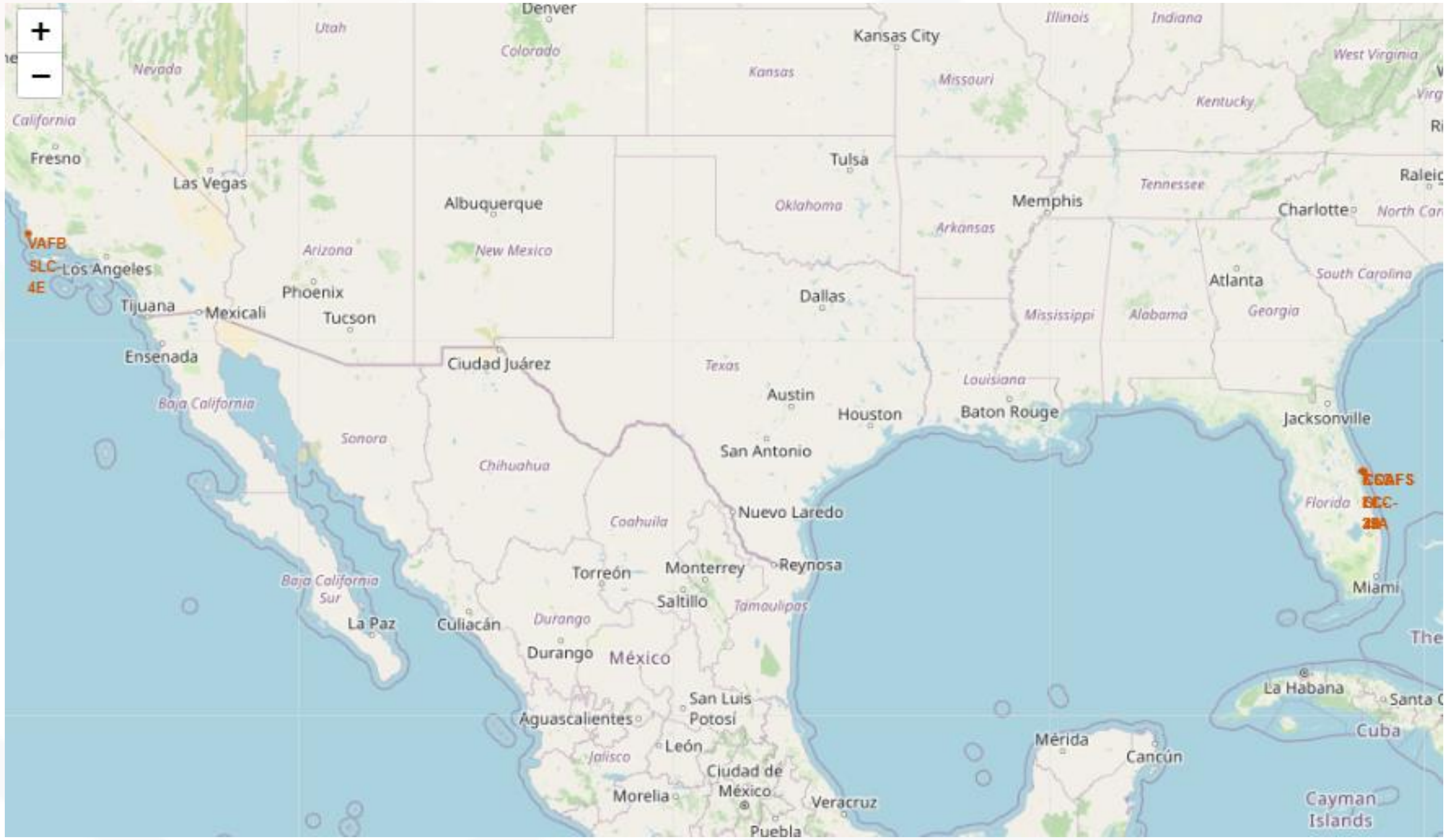
Landing_Outcome	landings
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

LAUNCH SITES PROXIMITY ANALYSIS

Locations of Launch Sites on Maps

- Three in the east
- One in the west
- All in the south

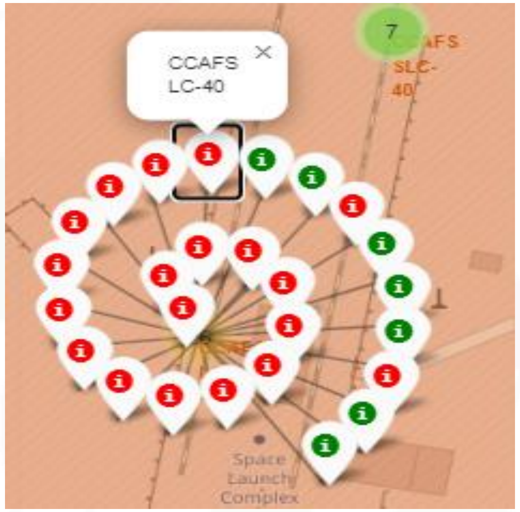
Launch Site	Lat	Long
CCAFS LC-40	28.56230197	-80.57735648
CCAFS SLC-40	28.56319718	-80.57682003
KSC LC-39A	28.57325457	-80.64689529
VAFB SLC-4E	34.63283416	-120.6107455

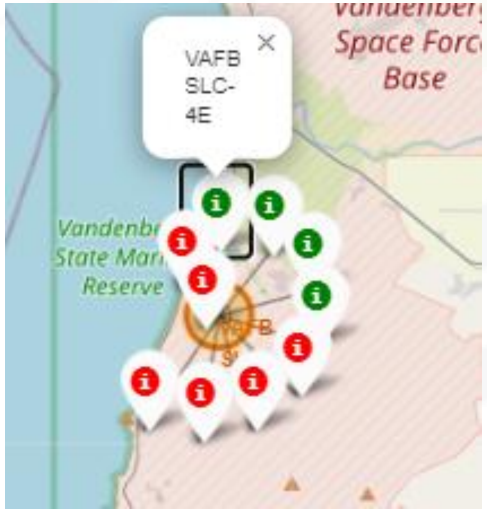


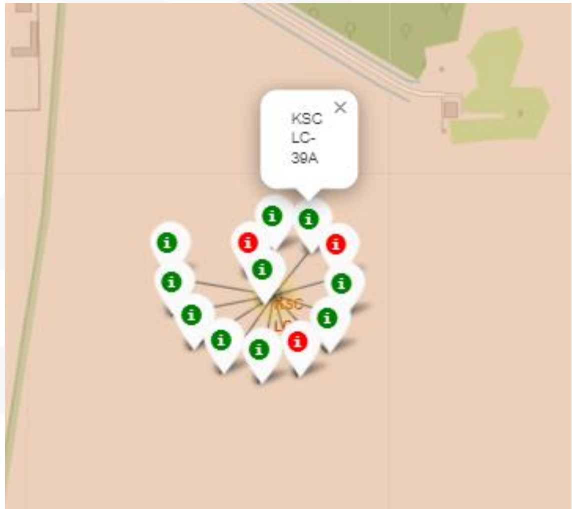
Display Launch Outcome by Color

From the color labels, we can easily see

- ➢ KSC LC-39A has a rather higher success rate
- ➢ Whereas CCAFS LC-40 and CCAFS SLC-40 have much lower rate







Display Launch Outcome by Color

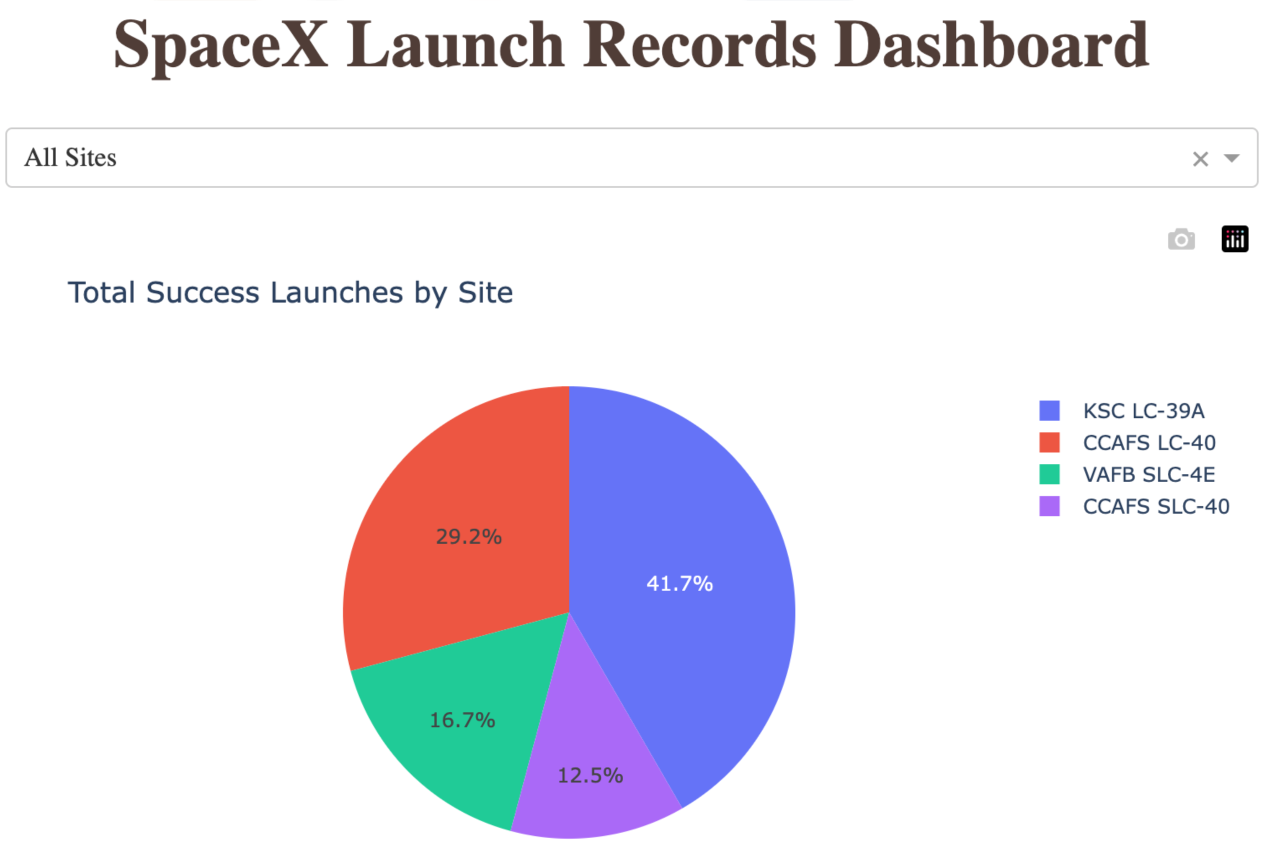
- The distance from KSC LC-39A to the nearest shuttle landing facility is about 4.16 km.
- The distance from KSC LC-39A to the nearest highway is less than 1 km.
- The distance from KSC LC-39A to the coastline is around 6.5 km.
- The distance from KSC LC-39A to the nearest city is around 16 km.

BUILD DASHBOARD WITH PLOTLY DASH

Total Success Launches for All Sites

Total Success Launches for All Sites is

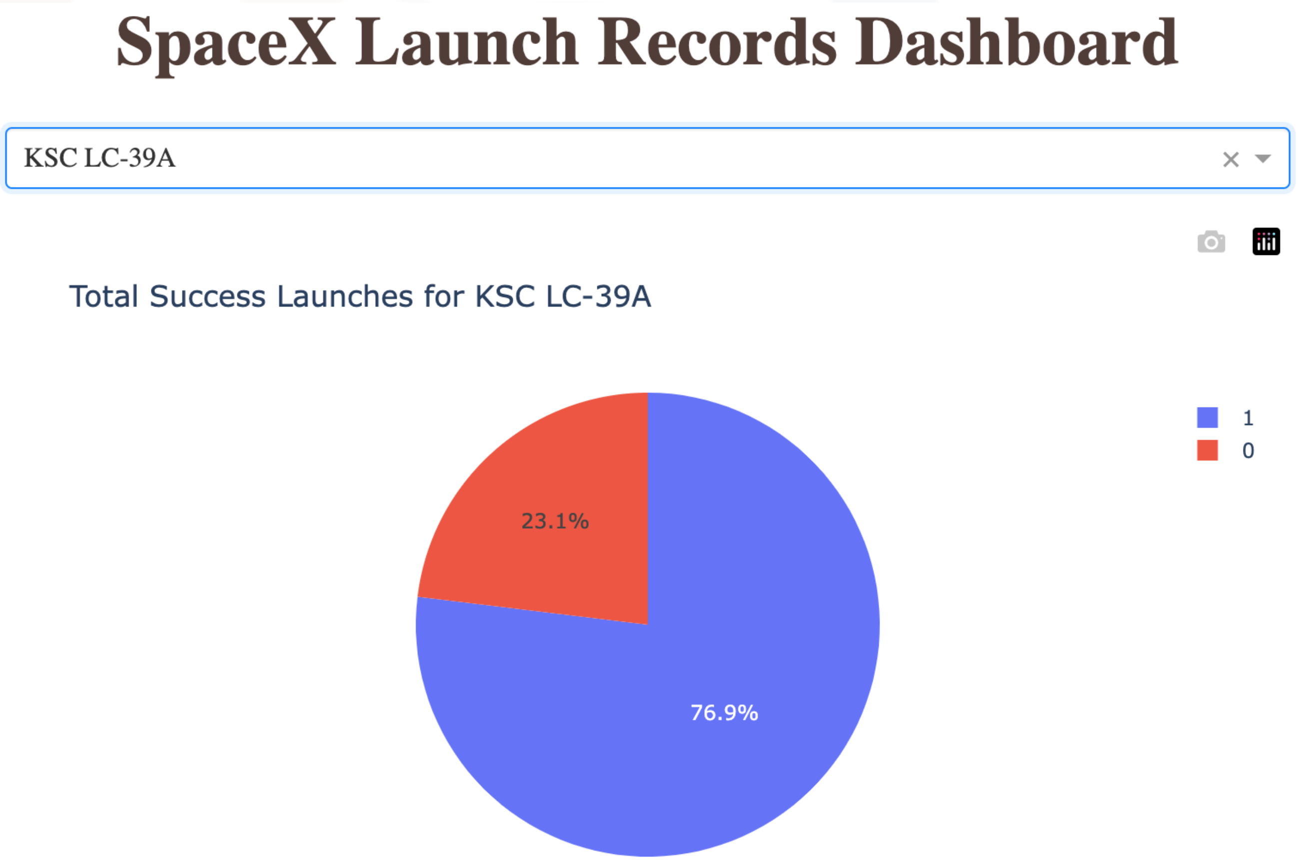
- CCAFS LC-40: 29.2%
- VAFB SLC-4E: 16.7%
- KSC LC-39A: 41.7%
- CCAFS SLC-40: 12.5%



Success Ratio for KSC LC-39A

The launch site with highest launch success ratio is **KSC LC-39A**.

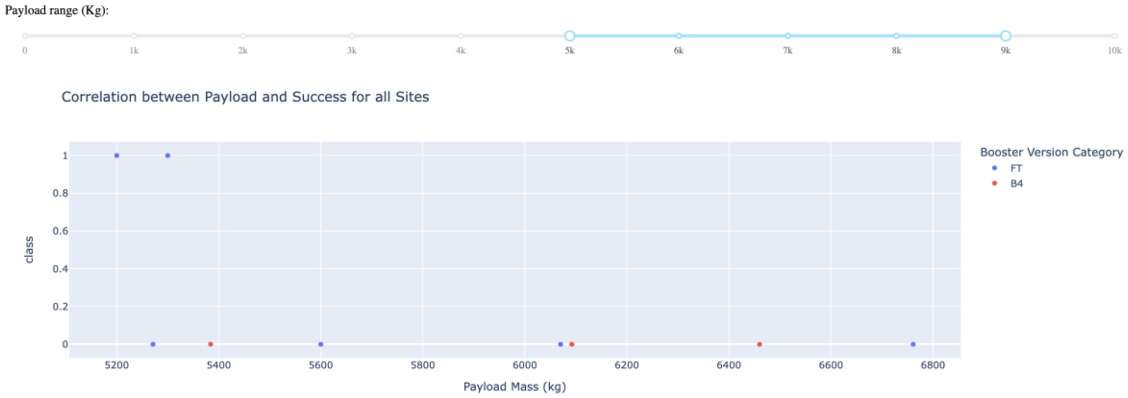
It has a success rate of **76.9%**.



Correlation Between Payload and Success

- Payload range in **[3000, 4000]** has the largest success rate.

- Booster version of **FT** has the largest success rate.



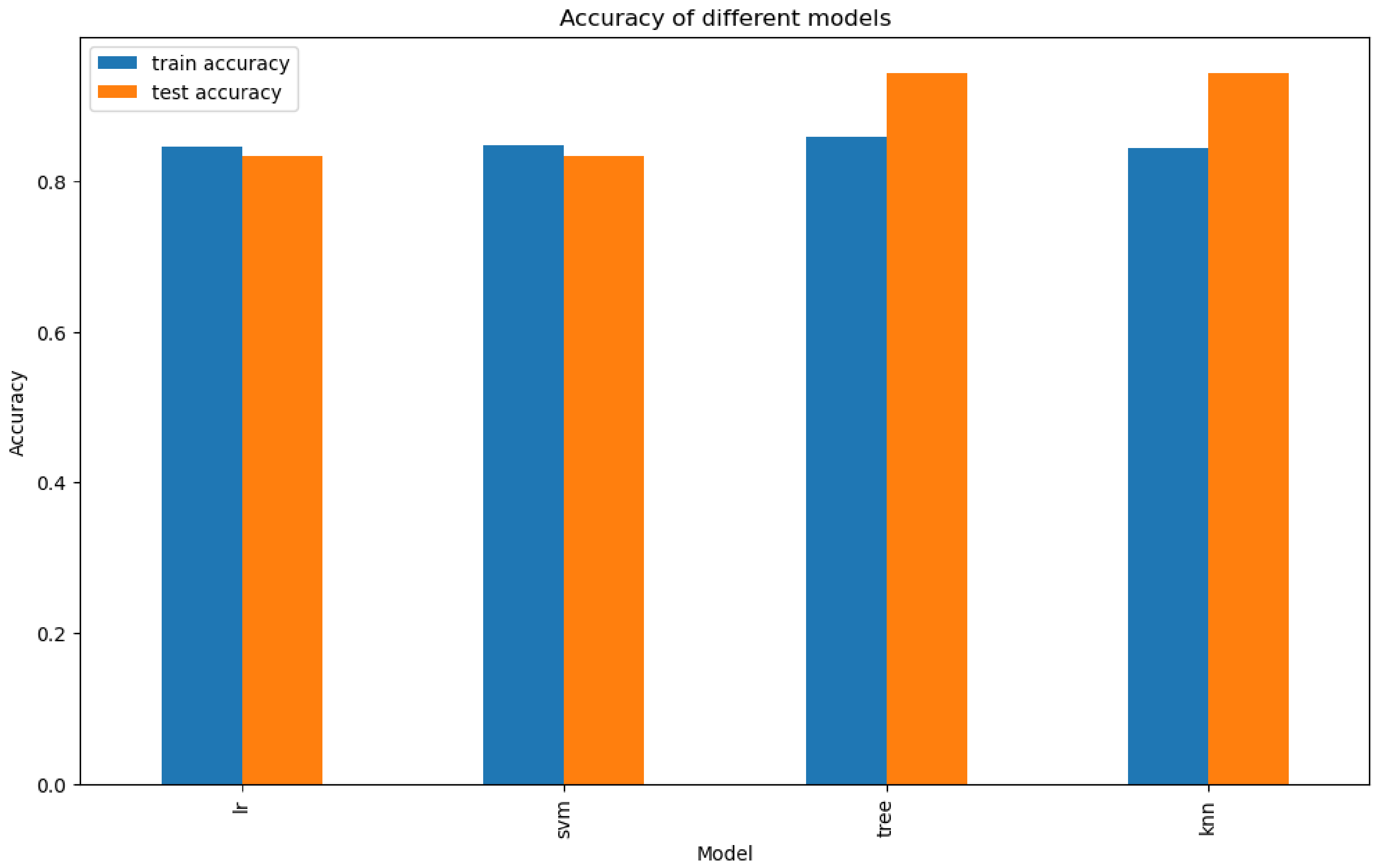
IBM Developer

SKILLS NETWORK

PREDICTIVE ANALYSIS CLASSIFICATION

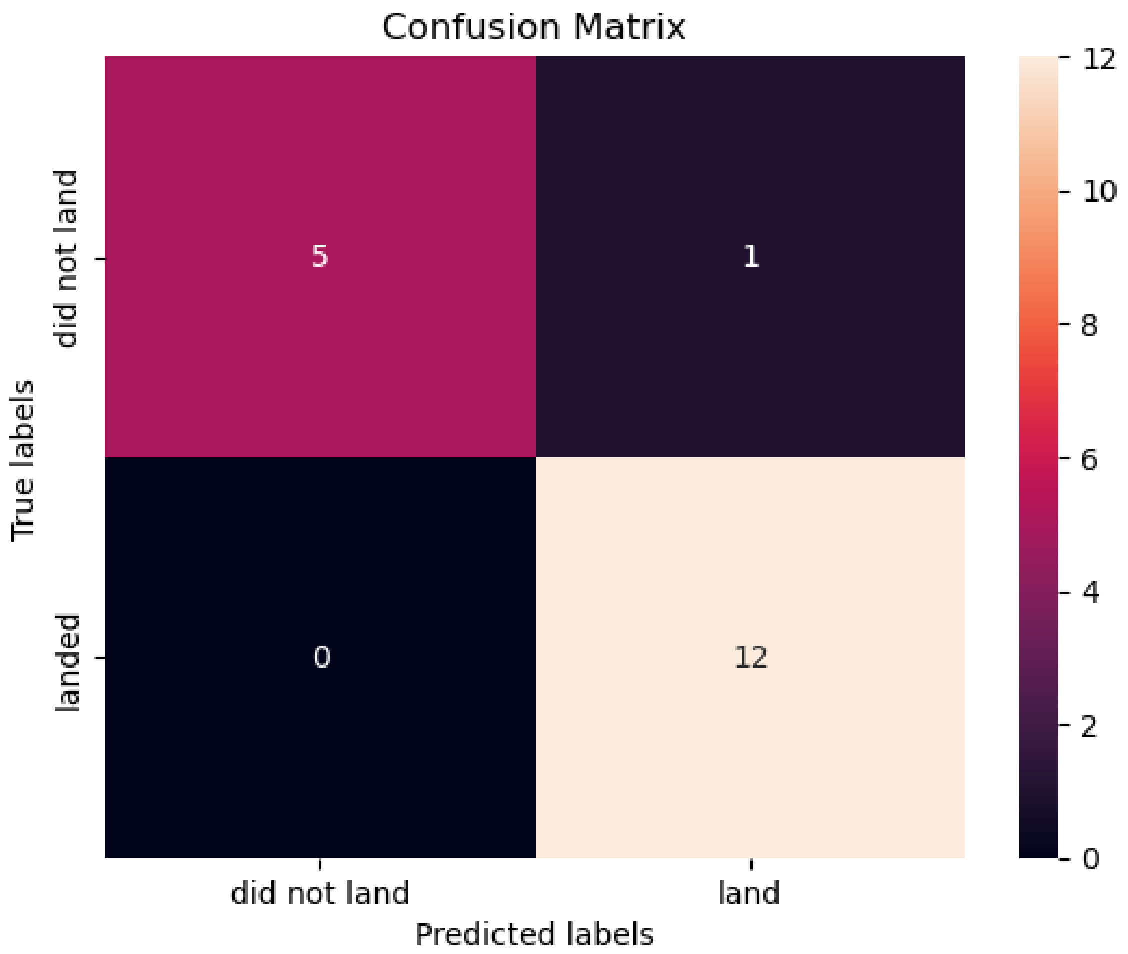
Classification Accuracy

- ❑ **Decision Tree model** has the highest classification accuracy
- ❑ training accuracy 0.86 and **testing accuracy 0. 0.94**
- ❑ Parameter: {'criterion': ‘entropy', 'max_depth’: 10, 'max_features': ‘sqrt', 'min_samples_leaf’: 4, 'min_samples_split’: 5, 'splitter': 'random'}



Confusion Matrix

- ❑ Decision Tree model can distinguish between the different classes.
- ❑ The major problem is **False Positives**.



Conclusions

- The dataset comprises **90 rows** and **83 columns**. Through an 80/20 split, we allocated 72 rows for training and 18 for testing purposes.
- Employing **GridSearchCV**, we trained four models, all exhibiting optimal performance on the test dataset.
- From these models, our top choice for predicting rocket landing outcomes is the **Decision Tree model**.
- However, utilising the decision tree may introduce concerns regarding **false positives**, potentially impacting our accuracy in estimating future bids for rocket launches.

Challenges in Model Training

- The dataset comprises 90 rows but contains 83 columns.
- With an 80/20 split, our training set consists of only 72 records for training.
- The imbalance of **more features than samples** raises concerns about **overfitting** during model training.
- Additionally, we only have 18 test samples. Too few to find out problems.

	FlightNumber	PayloadMass	Flights	Block	ReusedCount	Orbit_ES-L1	O
0	1.0	6104.959412	1.0	1.0	0.0	0.0	
1	2.0	525.000000	1.0	1.0	0.0	0.0	
2	3.0	677.000000	1.0	1.0	0.0	0.0	
3	4.0	500.000000	1.0	1.0	0.0	0.0	
4	5.0	3170.000000	1.0	1.0	0.0	0.0	
...	...	...	...	...	...	...	
85	86.0	15400.000000	2.0	5.0	2.0	0.0	
86	87.0	15400.000000	3.0	5.0	2.0	0.0	
87	88.0	15400.000000	6.0	5.0	5.0	0.0	
88	89.0	15400.000000	3.0	5.0	2.0	0.0	
89	90.0	3681.000000	1.0	5.0	0.0	0.0	

90 rows × 83 columns

Challenges in Model Training

- How to handle this issue?

- Obtain additional data, apply regularization, or employ dimension reduction techniques.

- **Consider** removing irrelevant columns based on identified **correlations** during EDA.

- Experiment with **PCA** to reduce dimensions efficiently.

	FlightNumber	PayloadMass	Flights	Block	ReusedCount	Orbit_ES-L1	O
0	1.0	6104.959412	1.0	1.0	0.0	0.0	
1	2.0	525.000000	1.0	1.0	0.0	0.0	
2	3.0	677.000000	1.0	1.0	0.0	0.0	
3	4.0	500.000000	1.0	1.0	0.0	0.0	
4	5.0	3170.000000	1.0	1.0	0.0	0.0	
...	...	...	...	...	...	...	
85	86.0	15400.000000	2.0	5.0	2.0	0.0	
86	87.0	15400.000000	3.0	5.0	2.0	0.0	
87	88.0	15400.000000	6.0	5.0	5.0	0.0	
88	89.0	15400.000000	3.0	5.0	2.0	0.0	
89	90.0	3681.000000	1.0	5.0	0.0	0.0	

90 rows × 83 columns

THANK YOU!