

Winning Space Race with Data Science

Hadi Mohammadian
2023-05-15



Outline

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

Executive Summary

- We use data collection API to get SpaceX Data Sets
- And use exploratory data analysis to analyze data and visualize data
- Use Grid Search method to find the best Machine Learning Model to predict the classification of next landing

Introduction

SpaceX Falcon 9 rocket will launch with a cost of **62** million dollars; whereas other providers sometimes cost upward of **165** million dollars each.

So, if we can determine whether 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.

- Can we **predict** a new launch with success of the first stage landing according to the historical launch data?
- Can we tell what is the best choice for a successful launch?

Section 1

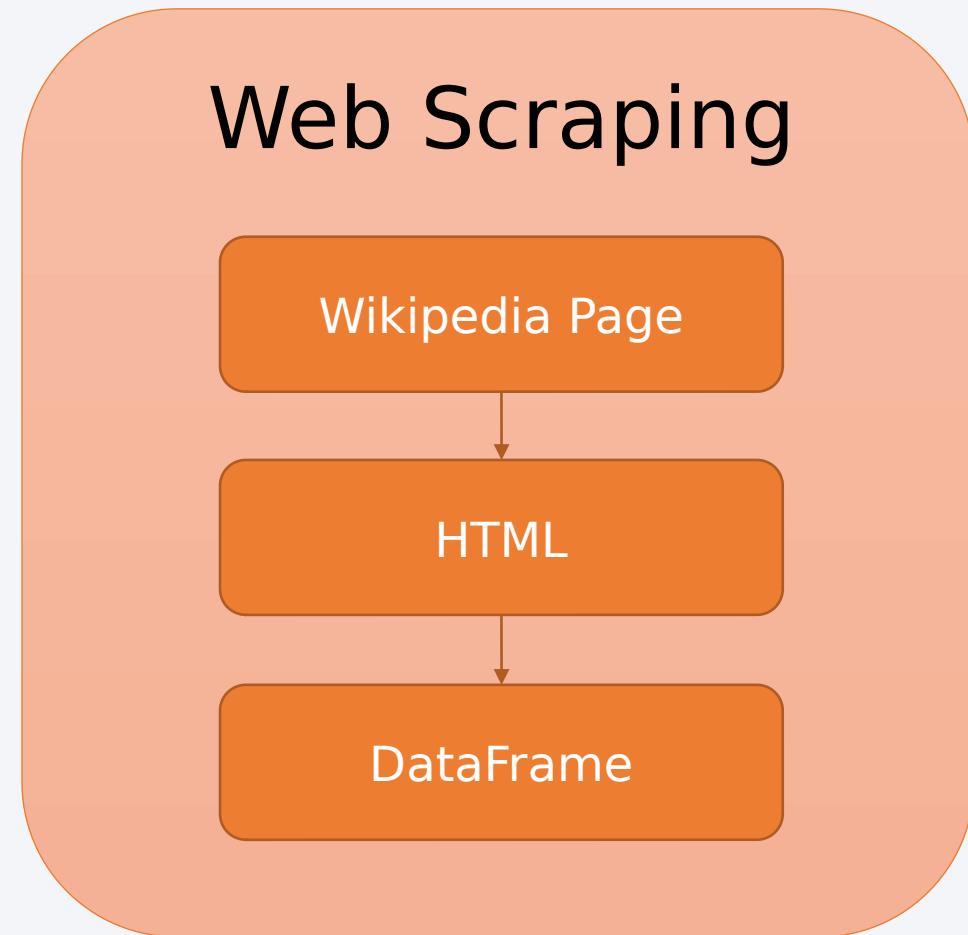
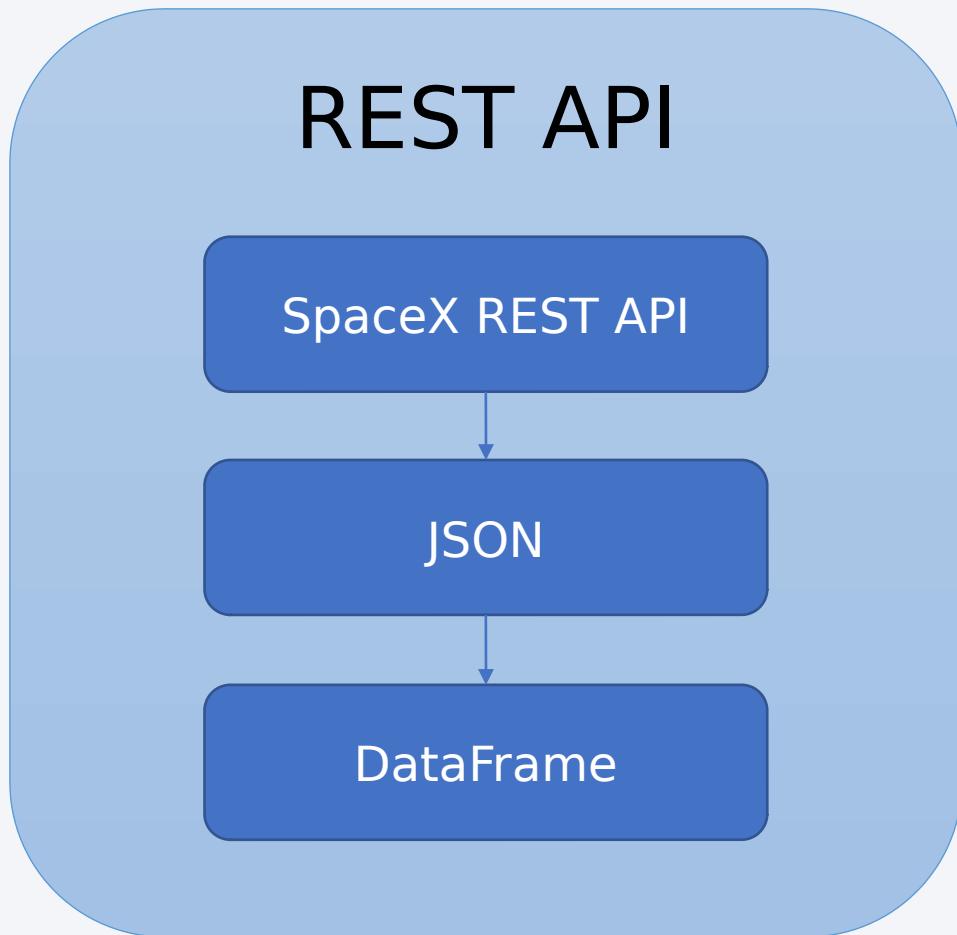
Methodology

Methodology

Executive Summary

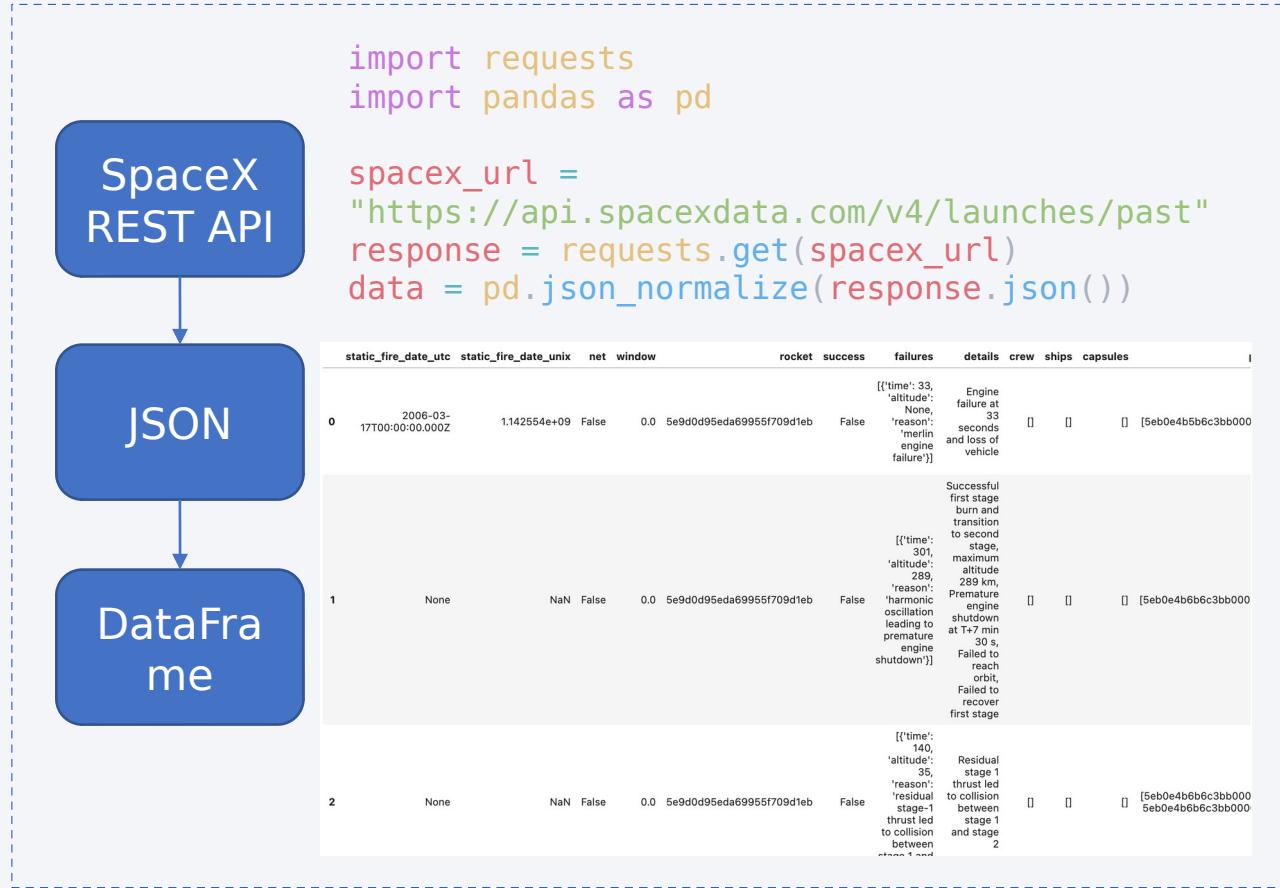
- Data collection methodology:
 - SpaceX REST API
 - Web Scraping (Wikipedia)
- Perform data wrangling
 - Generate landing Class from Outcome column
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive 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/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/1-jupyter-labs-spacex-data-collection-api.ipynb>



Data Collection - Scraping

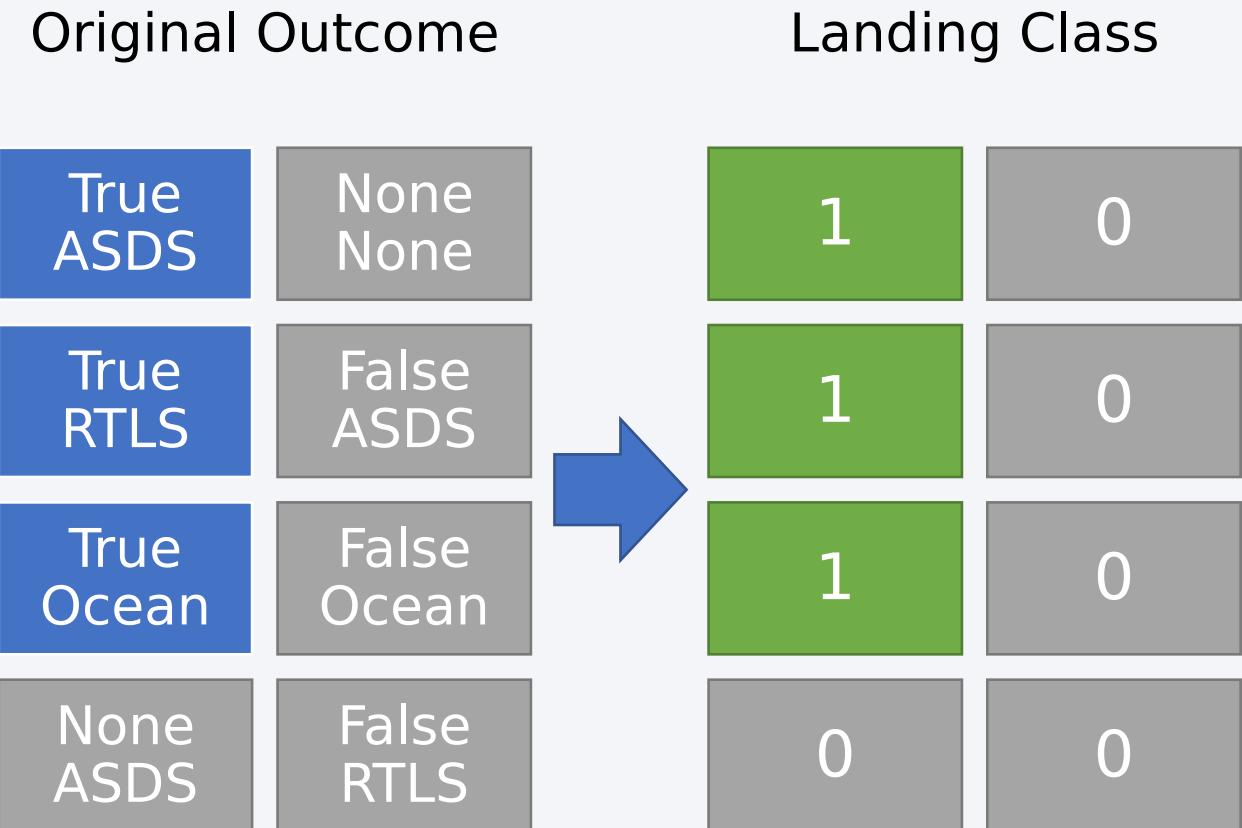
- Wikipedia Falcon Page
https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- My Notebook
 - <https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/2-jupyter-labs-webscraping.ipynb>



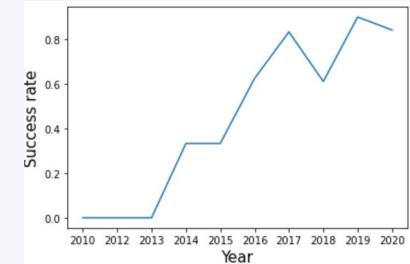
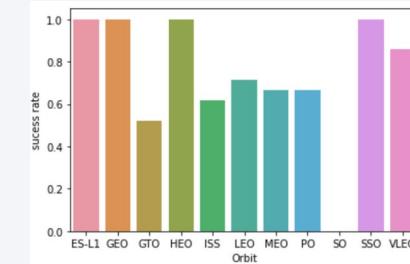
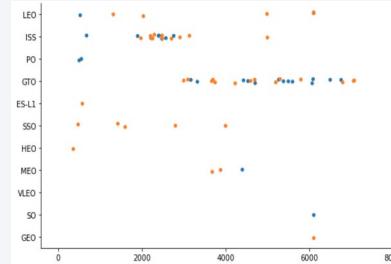
Data Wrangling

- **My Notebook**

- <https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/3-jupyter-spacex-data-wrangling.ipynb>
- Transform raw data to **useful** data. For example, convert original outcome labels into landing class that represent landing classification which will be our new landing prediction target.
 - 1 for success
 - 0 for failure



EDA with Data Visualization



My Visualization Notebook

<https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/5-jupyter-labs-eda-dataviz.ipynb>

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

EDA with SQL

• My SQL Notebook

- https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/4-jupyter-labs-eda-sql-coursera_sqlite.ipynb

```
%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.

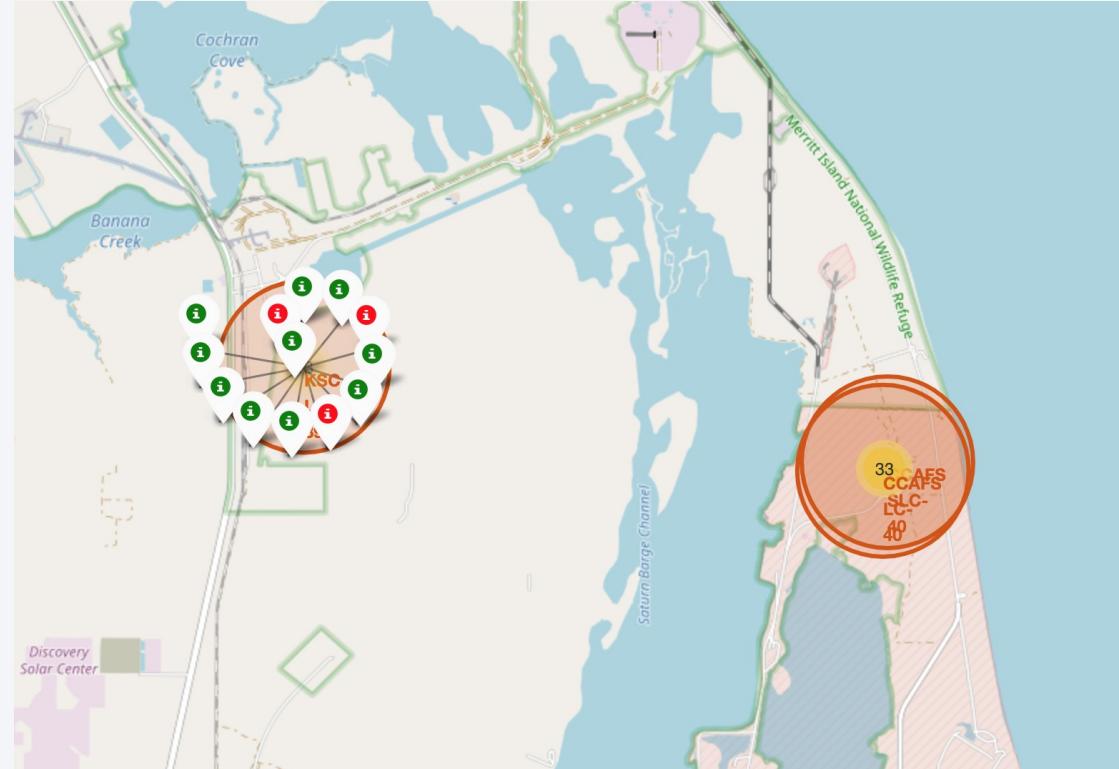
Launch_Site	Booster_Version	Landing_Outcome	landings
CCAFS LC-40	F9 FT B1022	Success	20
VAFB SLC-4E	F9 FT B1026	No attempt	10
KSC LC-39A	F9 FT B1021.2	Success (drone ship)	8
CCAFS SLC-40	F9 FT B1031.2	Success (ground pad)	6
		Failure (drone ship)	4
		Failure	3
		Controlled (ocean)	3
		Failure (parachute)	2
		No attempt	1

Build an 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

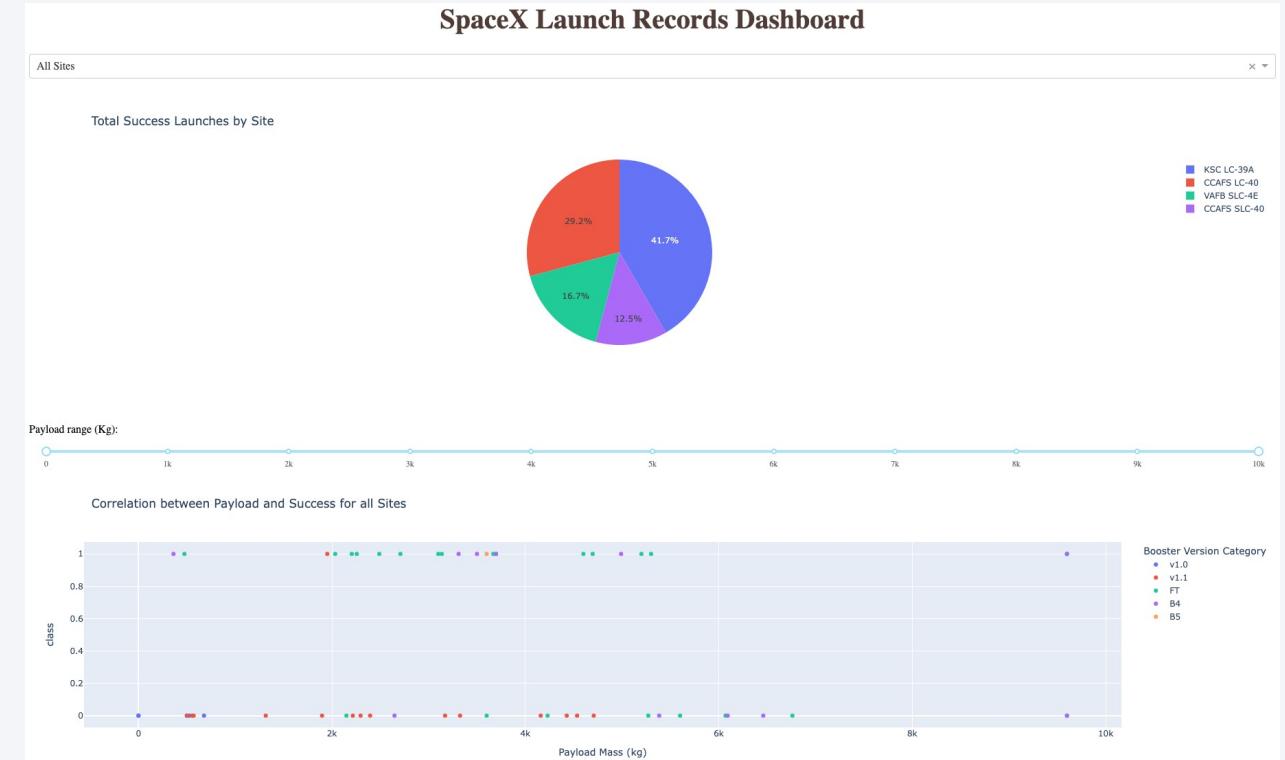


My Notebook

https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/6-jupyter_launch_site_location_folium.ipynb

Build a Dashboard with Plotly Dash

With a **Dropdown menu** and a **Pie Chart**, we can get success launches distribution by launch site
Additionally, with a **Range Slider** and a **Scatter Plot**, we can analyze the correlation between Payload and Success for different launch sites

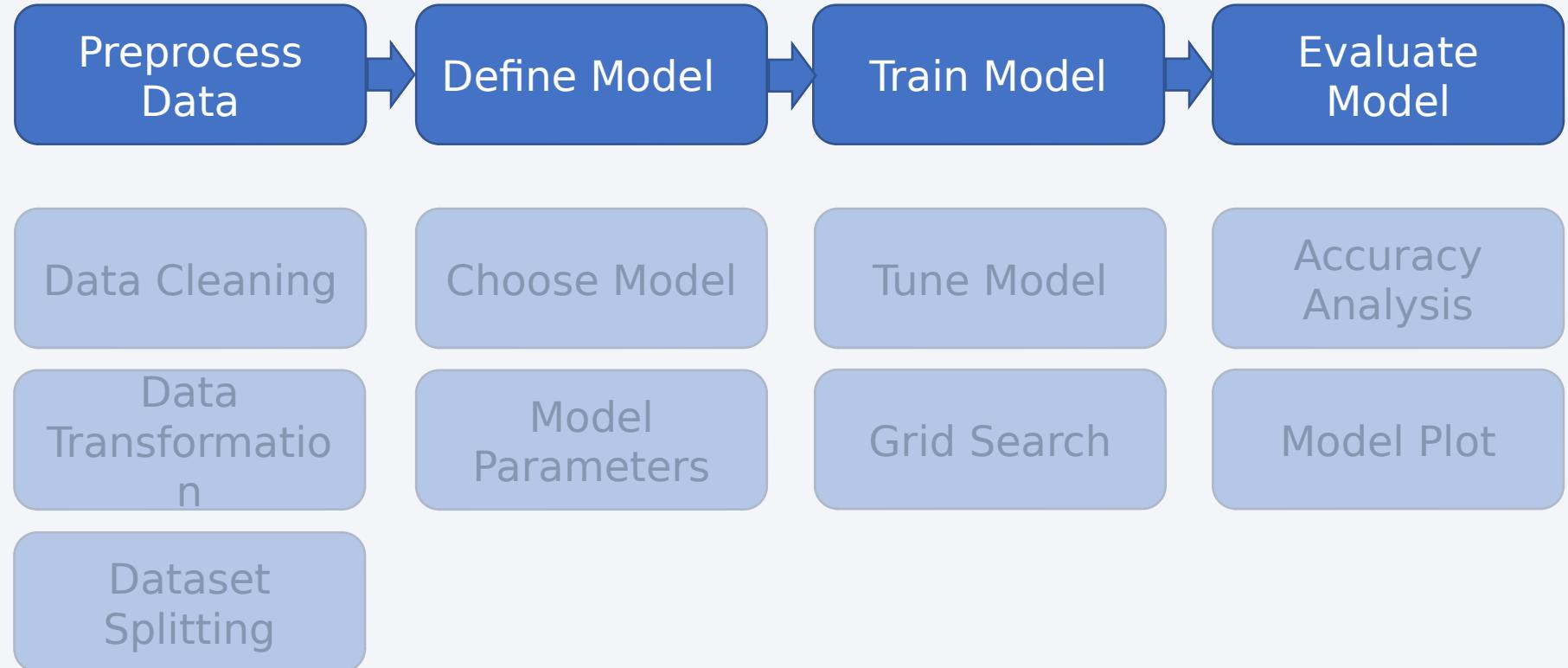


My Dashboard Python Code

https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

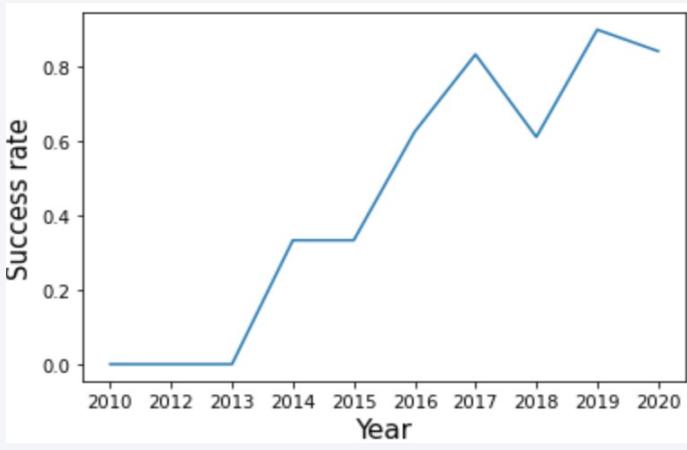
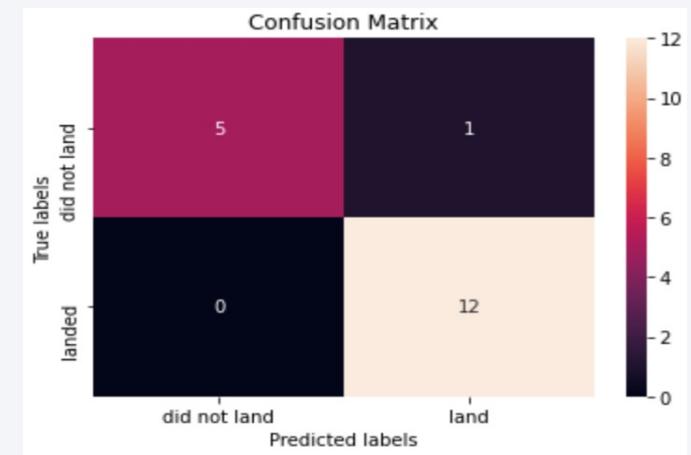
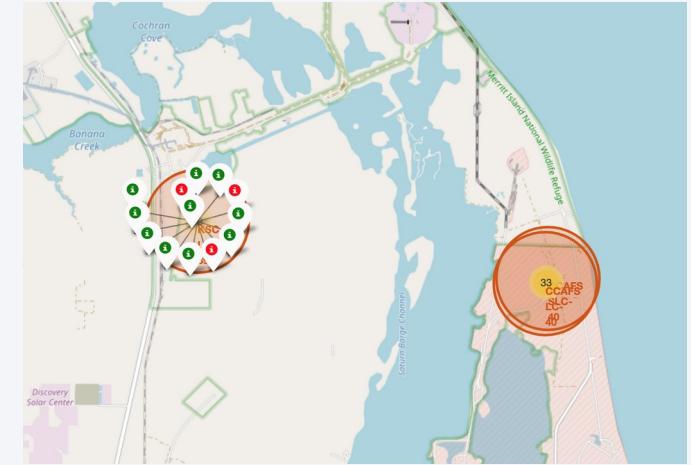
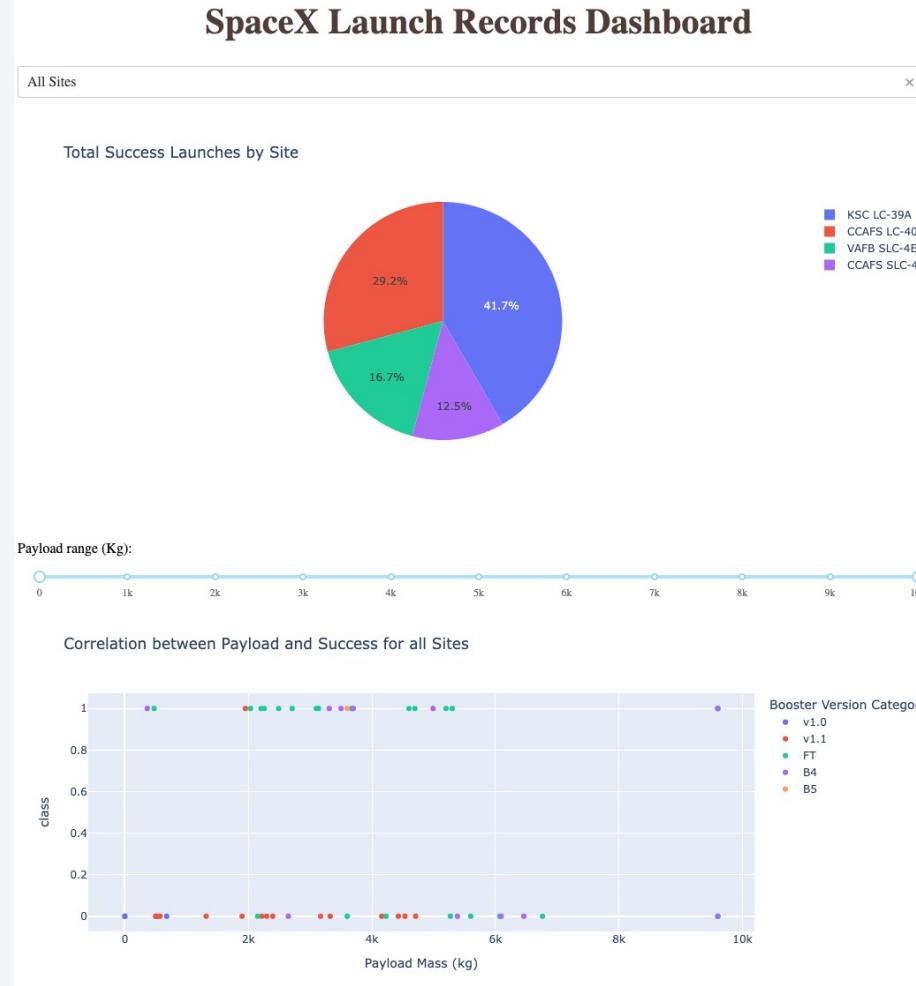
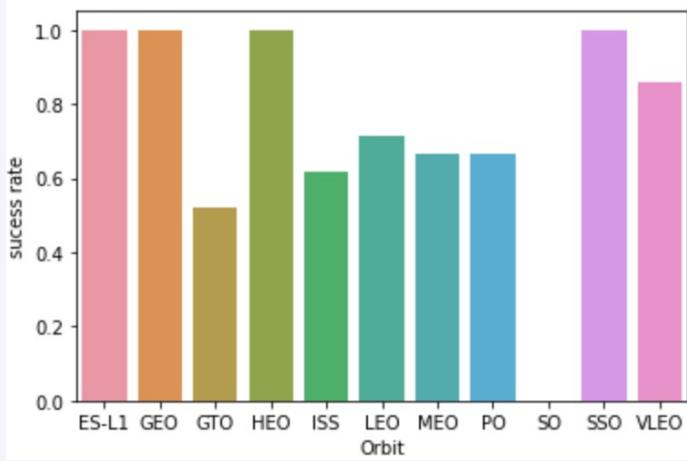
- Prepare** data
- Create** a column for the class
- Standardize** the data
- Split** into training data and test data
- Define** model and parameters
- Train** and **Grid Search** for best parameters
- Evaluation**

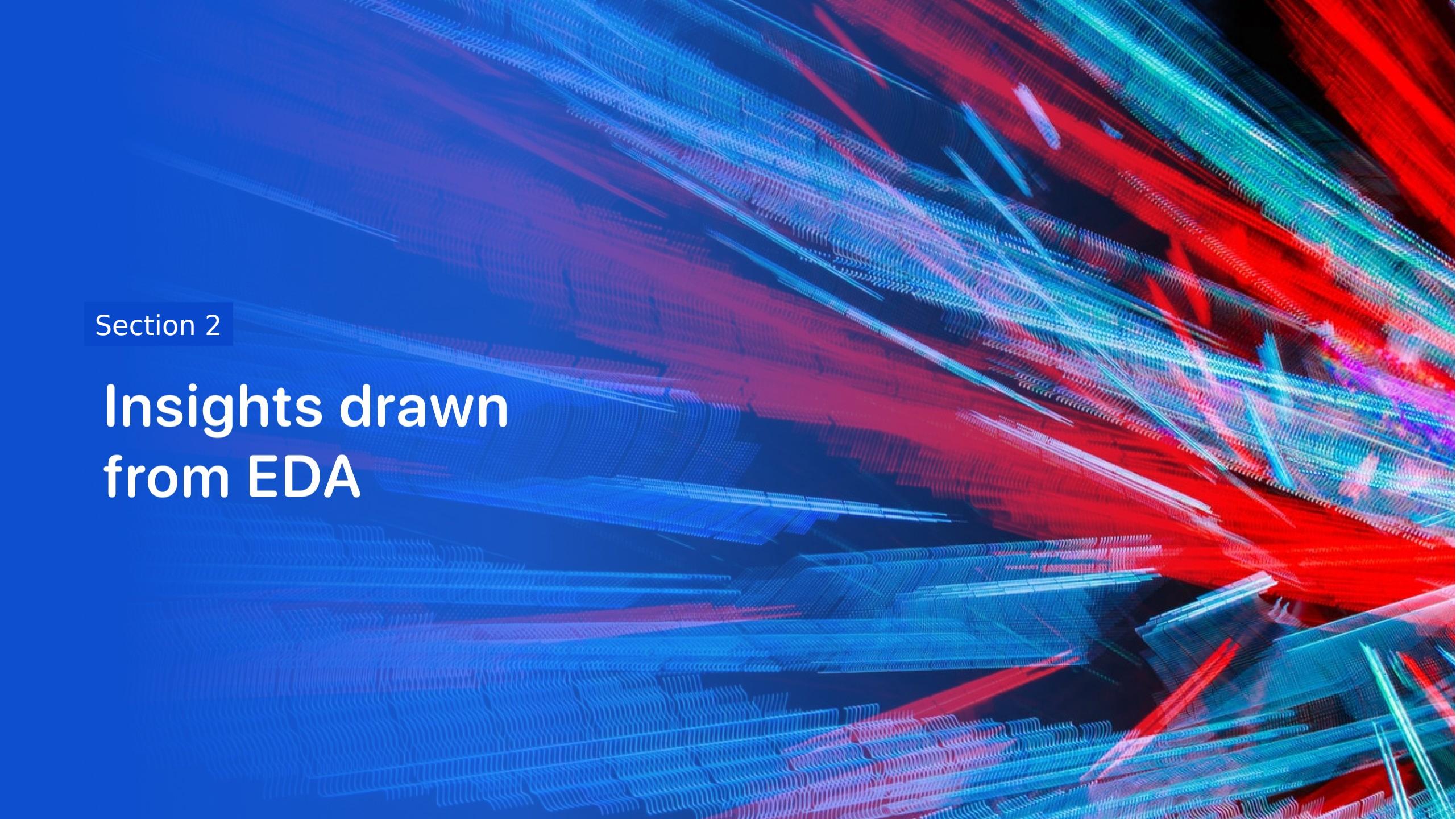


My Notebook

https://github.com/hadimohammadian/IBM-Applied-Data-Science-Capstone/blob/main/7-SpaceX_Machine_Learning_Prediction_Part_5.ipynb

Results



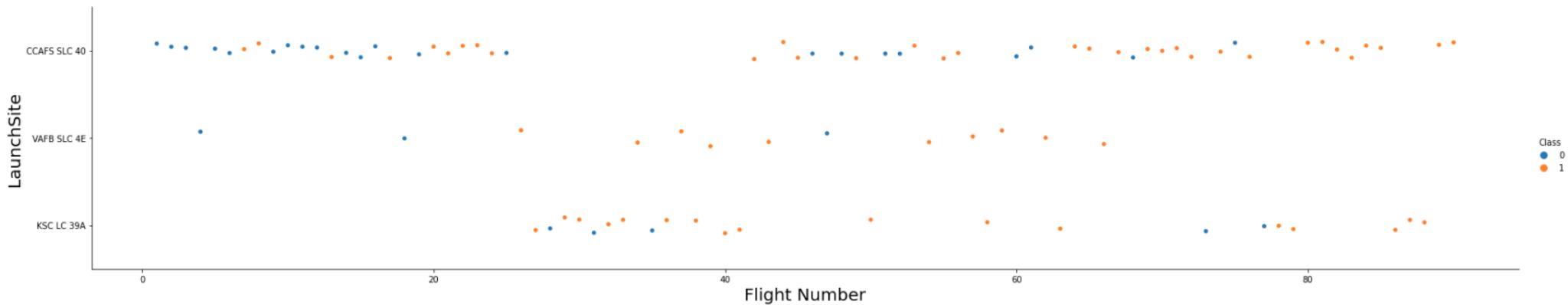
The background of the slide features a complex, abstract pattern of wavy, horizontal lines. These lines are primarily colored in shades of blue, red, and green, creating a sense of depth and motion. They are arranged in several layers, with some lines being more prominent than others. The overall effect is reminiscent of a digital or scientific visualization of data flow or signal processing.

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

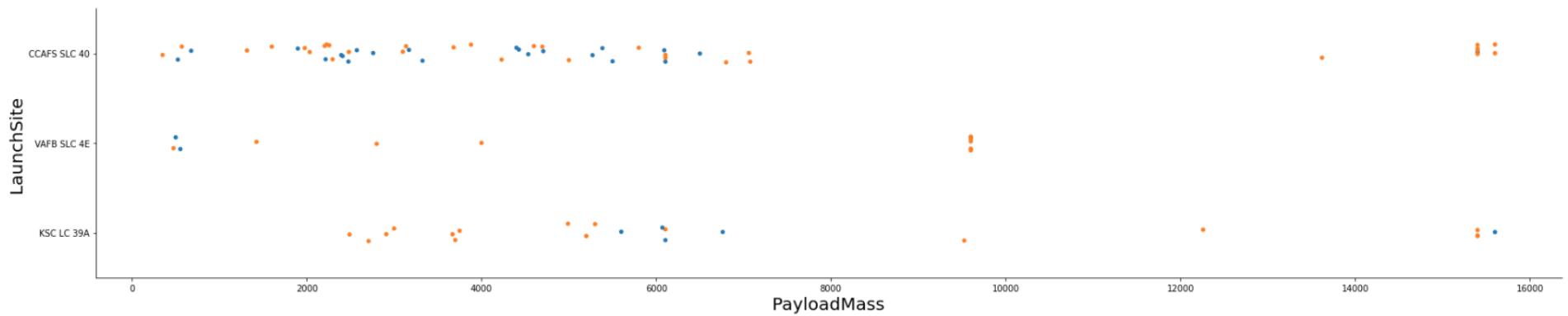
```
[4] # Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number", fontsize=20)
plt.ylabel("LaunchSite", fontsize=20)
plt.show()
```



Explanation: We can see from the scatter plot that as flight number increases, there are more successful first stage landing. With small flight numbers, launches happens more in the site CCAFS SLC 40 and with much lower success rate. Although there are less launches in VAFB SLC 4E and KSC LC 39A, higher success rate can be seen in these two sites.

Payload vs. Launch Site

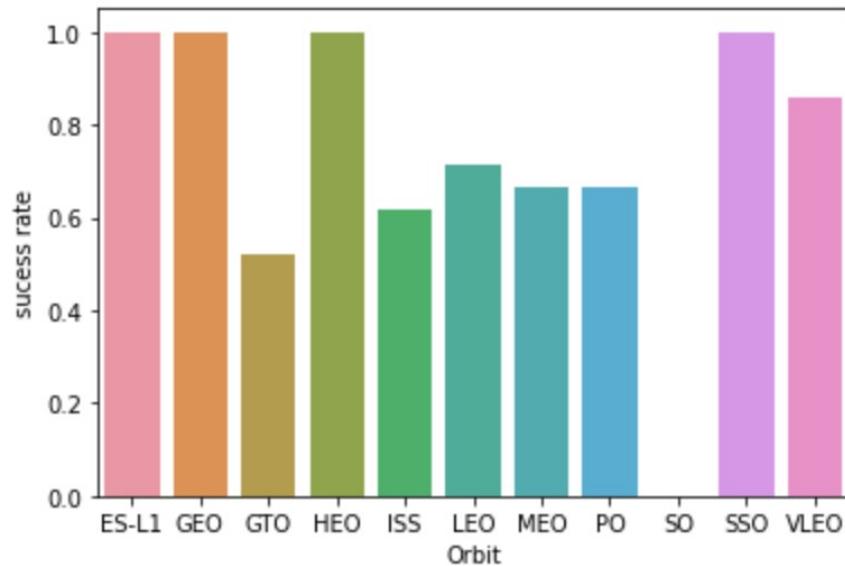
```
[5] # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, a  
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)  
plt.xlabel("PayloadMass", fontsize=20)  
plt.ylabel("LaunchSite", fontsize=20)  
plt.show()
```



Explanation: With higher Payload the success rate is much higher. And in KSC LC39A launchsite we can see much higher success rate with low Payload whereas this rate is mucher lower in CCAFS SLC 40 launchsite. Besides, there no rockets launched in VAFB-SLC for Payload greater than 10000. Furthermore, with Payload more than 9500, we can see very high success rate overall.

Success Rate vs. Orbit Type

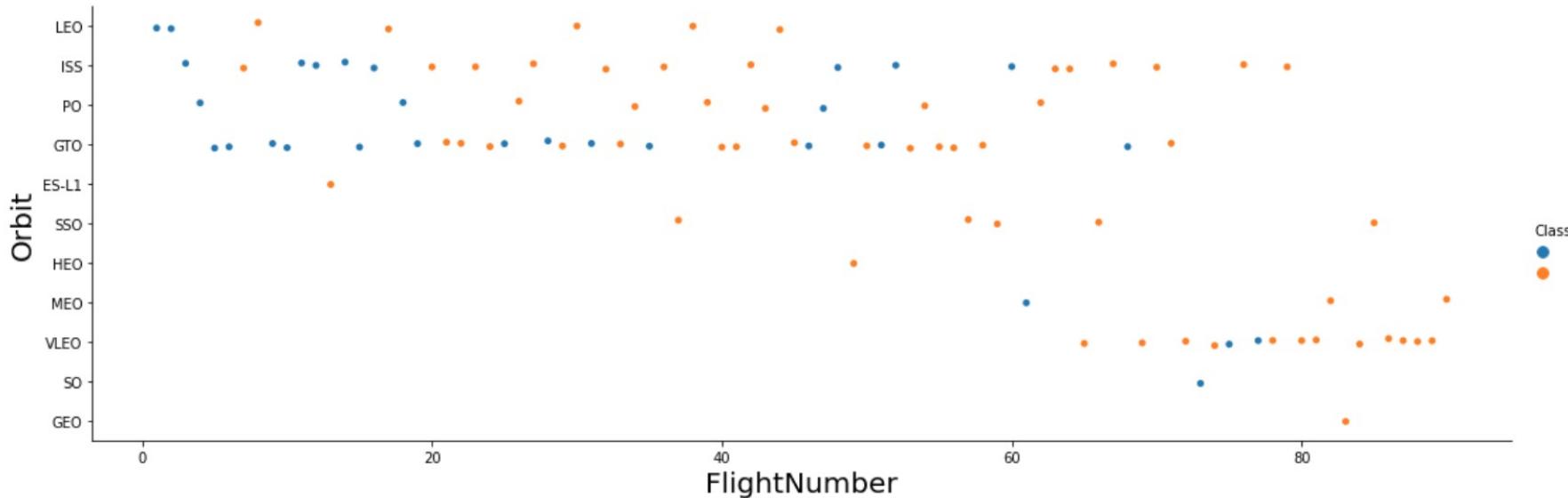
```
[ ] sns.barplot(y='Class', x='Orbit', data=df_success_rate)
plt.xlabel("Orbit", fontsize=10)
plt.ylabel("sucess rate", fontsize=10)
plt.show()
```



Explanation: From the Bar Plot we can see for Orbit type ES-L1, GEO, HEO, and SSO have the highest success rate, which is 100%. And we also find in SO orbit, the rate is zero.

Flight Number vs. Orbit Type

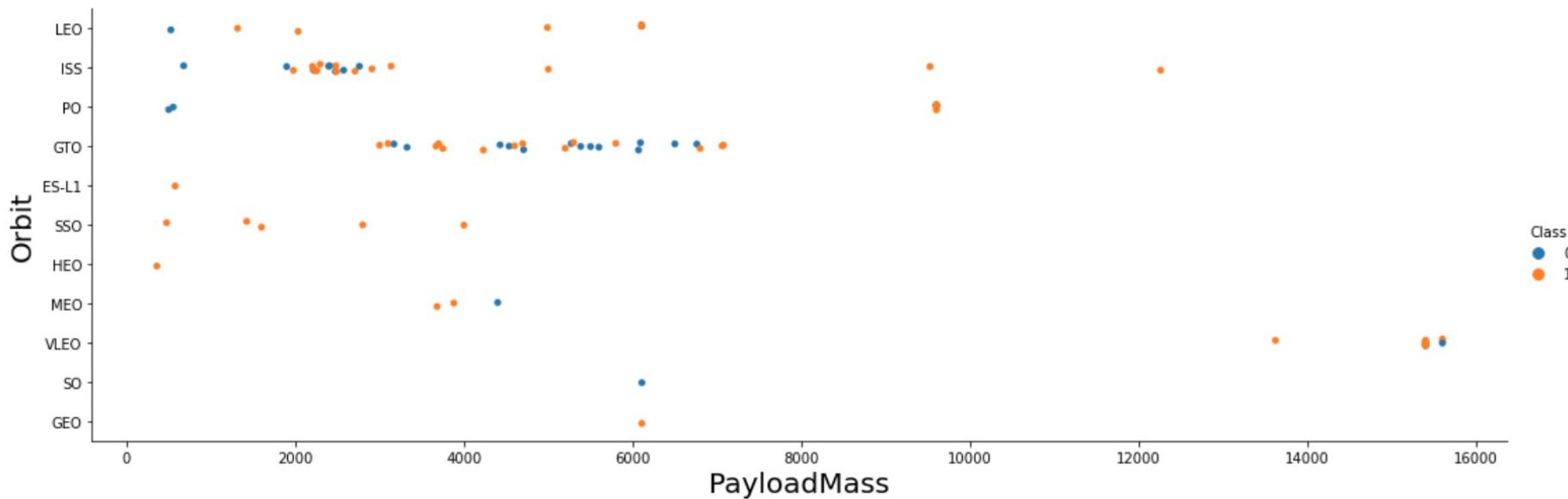
```
[9] # Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be  
sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df, aspect = 3)  
plt.xlabel("FlightNumber", fontsize=20)  
plt.ylabel("Orbit", fontsize=20)  
plt.show()
```



Explanation: In ES-L1, GEO, HEO, and SSO orbits, all launches are successful. There is a clear relationship between flight number and success rate in LEO orbit since as flightnumber increases, the success rate increases. In contrast, there is no such obvious relationship in GTO orbit.

Payload vs. Orbit Type

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

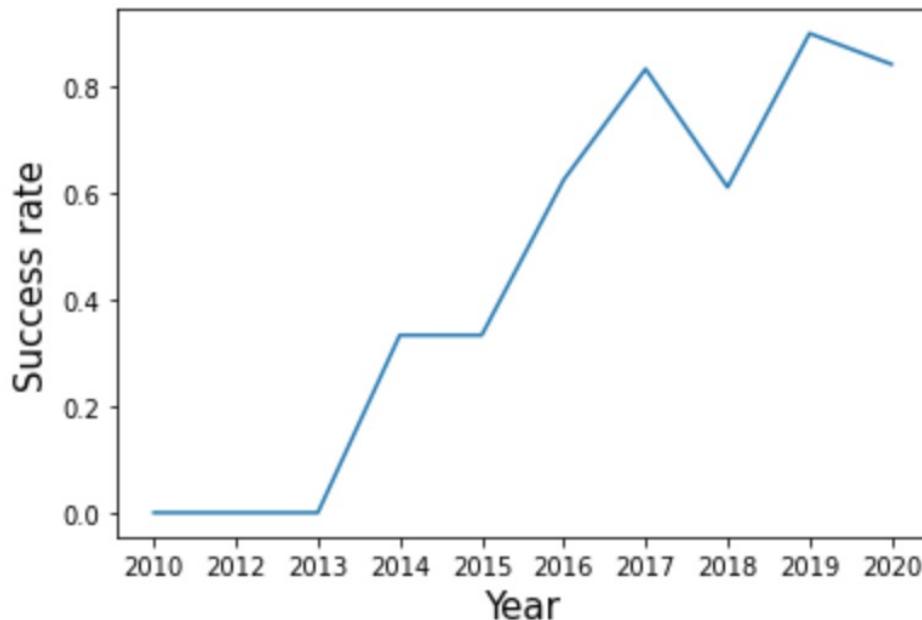


Explanation: With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.

However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

```
[14] sns.lineplot(y='Class', x='Year', data=df_year_success)
    plt.xlabel("Year", fontsize=15)
    plt.ylabel("Success rate", fontsize=15)
    plt.show()
```



Explanation: you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

Four Launch Sites:

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

```
%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
```

- 1** in western coast
- VAFB SLC-4E
- 3** in eastern coast
- KSC LC-39A
 - CCAFS SLC-40
 - CCAFS LC-40



Launch Site Names Begin with 'CCA'

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

```
%sql select * from SPACEXTBL 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 _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

Explanation: these 5 launches happened in LEO orbit, and four of them were from customer NASA.

Total Payload Mass

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

* sqlite:///my_data1.db
Done.

sum(PAYLOAD_MASS__KG_)
99980
```

Explanation: 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
```

Explanation: the 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
```

Explanation: the first successful landing outcome on ground pad is **01-05-2017**.

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

```
%%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

Explanation:

- 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

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

```
%%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

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
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

```
%%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

The background of the slide is a photograph taken from space at night. It shows the curvature of the Earth against the dark void of space. City lights are visible as numerous small white and yellow dots, primarily concentrated in coastal and urban areas. In the upper right corner, there are bright green and yellow bands of light, characteristic of the aurora borealis or aurora australis. The overall atmosphere is dark and mysterious.

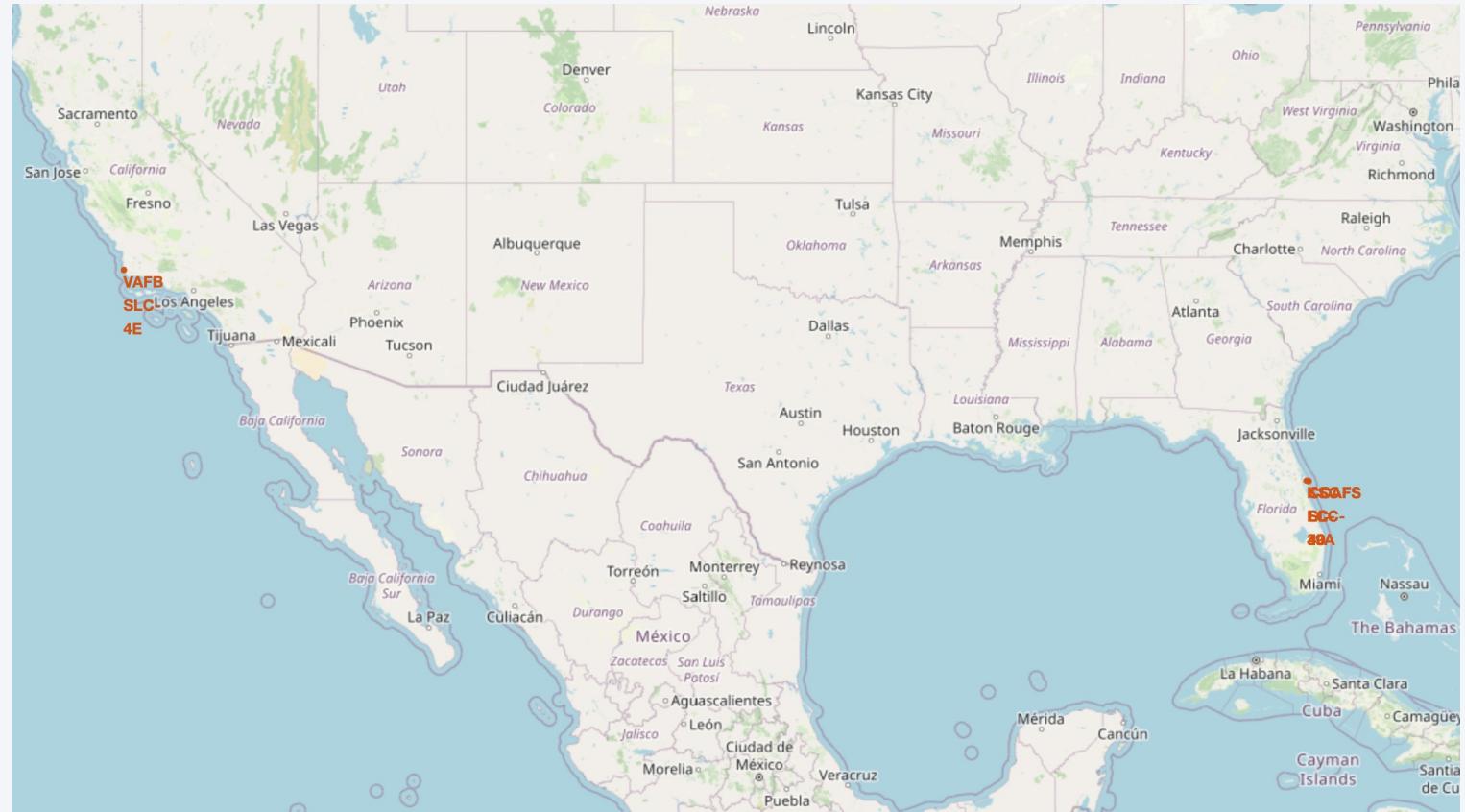
Section 3

Launch Sites Proximities Analysis

Locations of Launch Sites on Maps

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

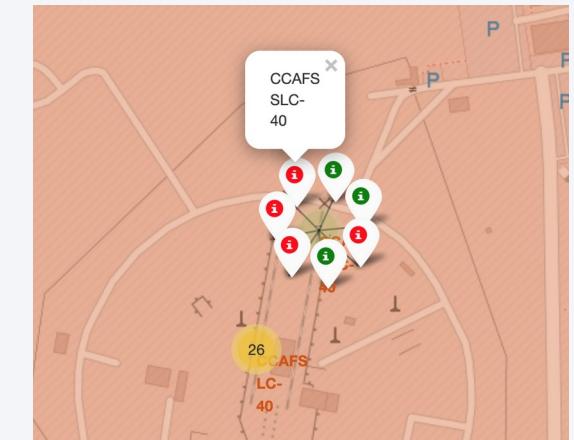
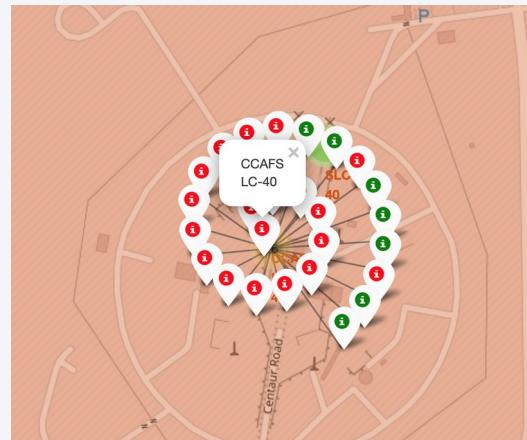
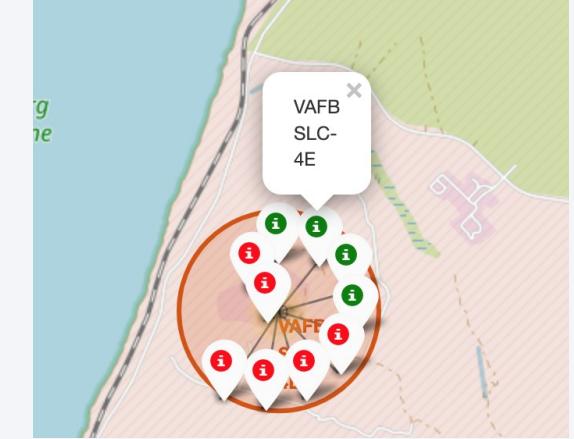
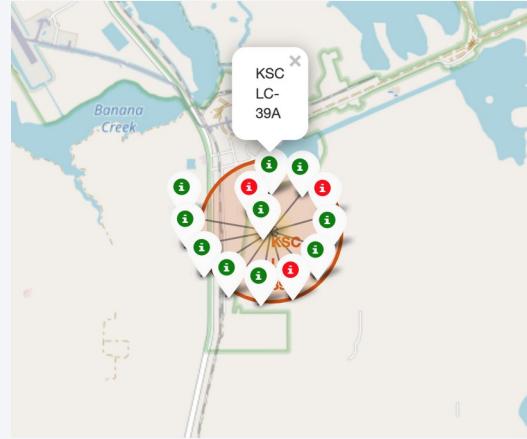
Launch Site	Lat	Long
CCAFS LC-40	28.56230197	-80.57735648
CCAFS SLC-40	28.56319718	-80.57682003
KSC LC-39A	28.57325457	-80.64689529
VAFB SLC-4E	34.63283416	-120.6107455



Display Launch Outcome by Color

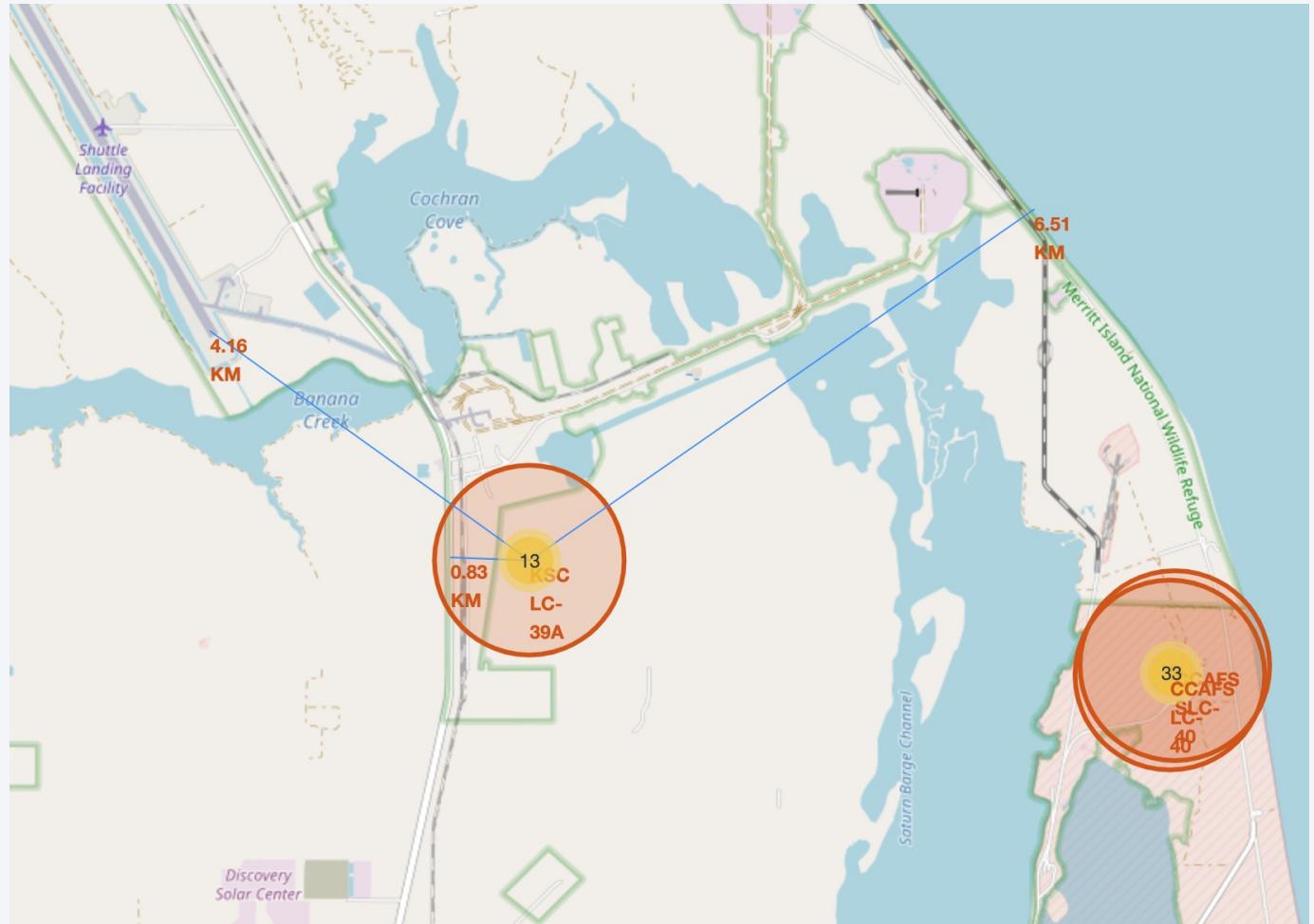
From the color labels, we can easily see

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



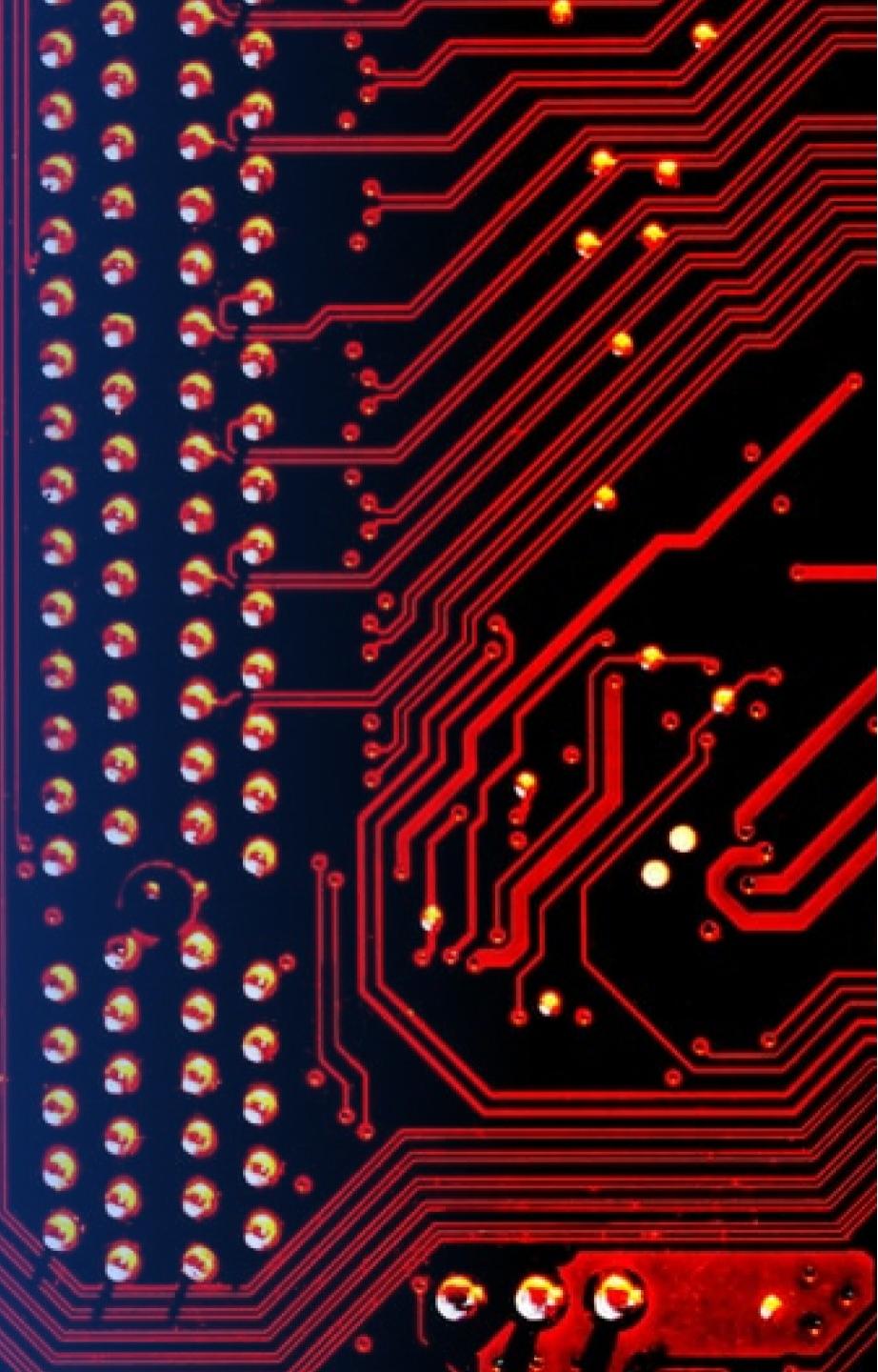
Show Distance to Proximities

- ❖ 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.



Section 4

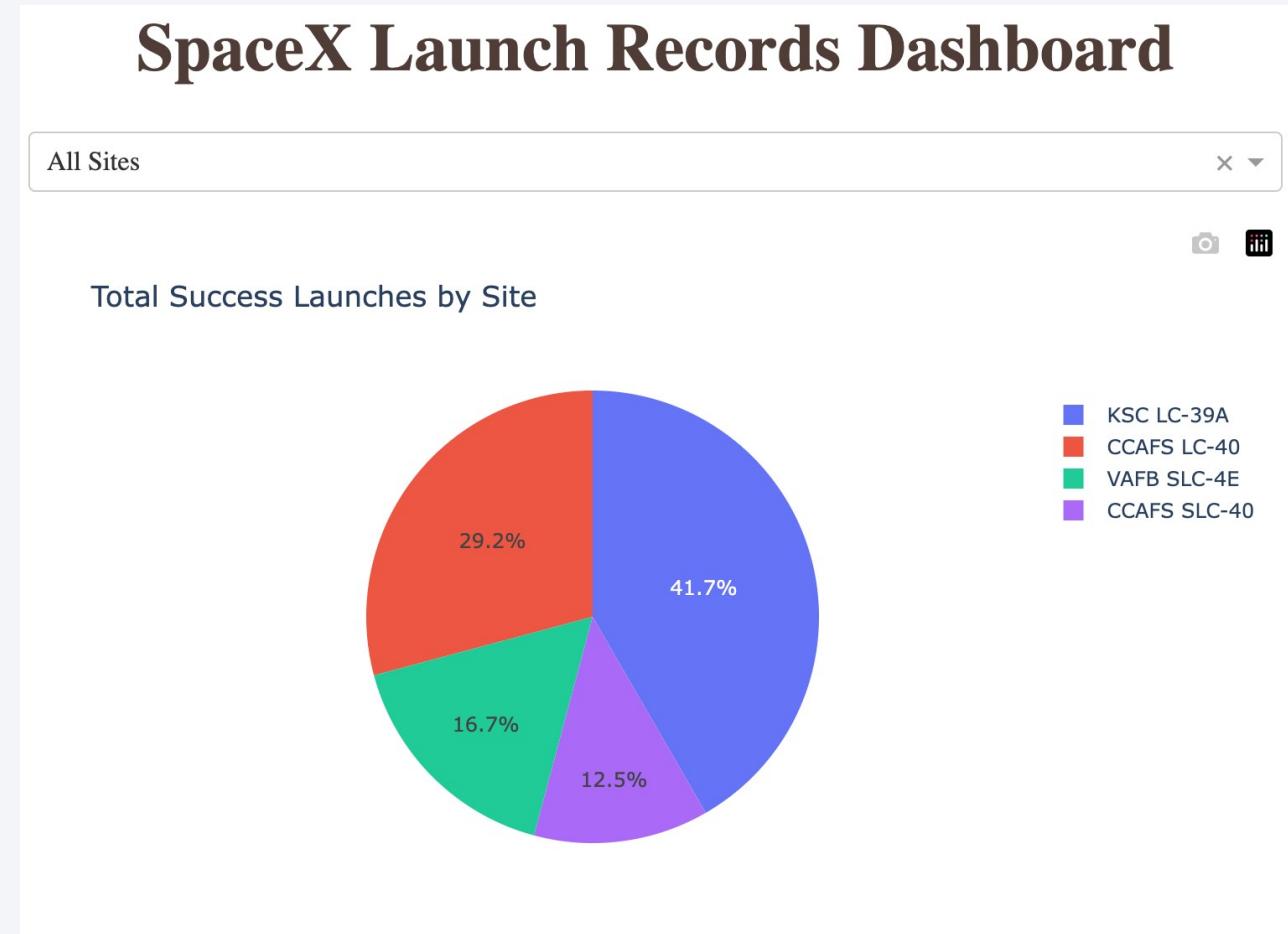
Build a Dashboard with Plotly Dash



Total Success Launches for All Sites

Total Success Launches for All Sites is

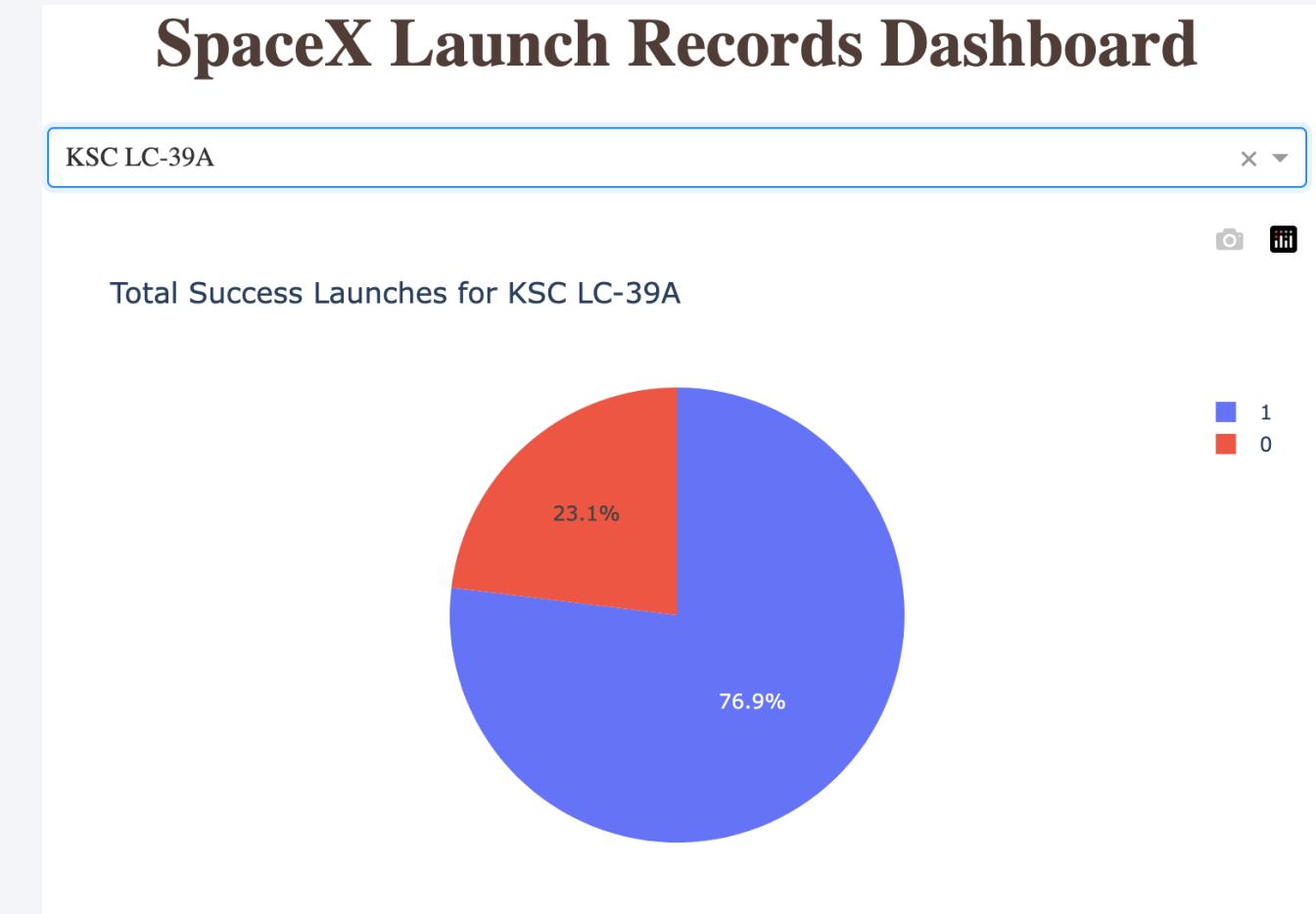
- CCAFS LC-40: 29.2%
- VAFB SLC-4E: 16.7%
- KSC LC-39A: 41.7%
- CCAFS SLC-40: 12.5%



Success Ratio for KSC LC-39A

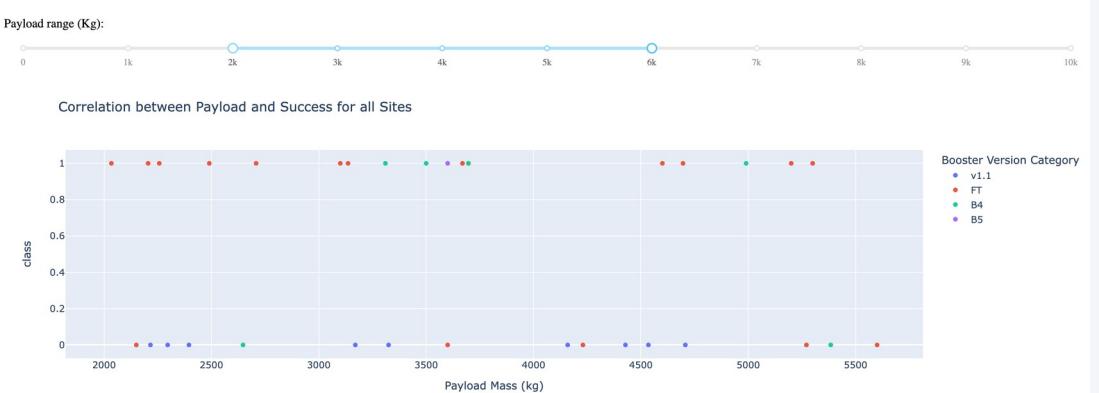
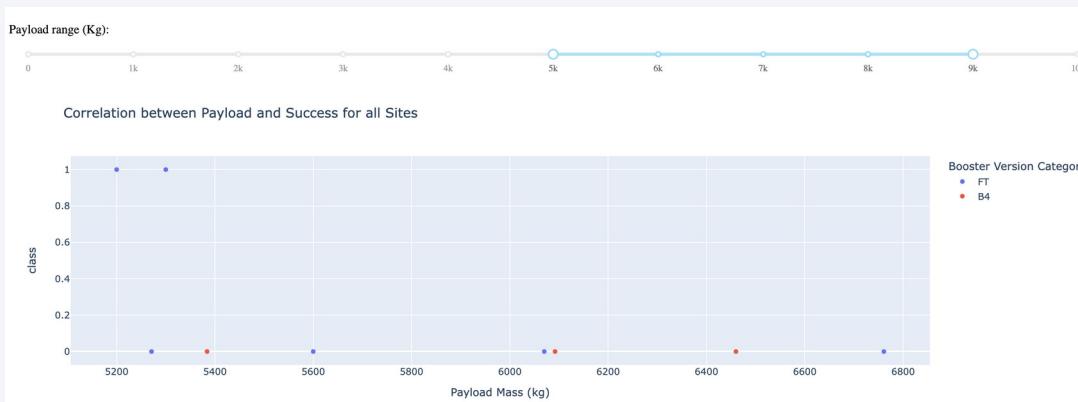
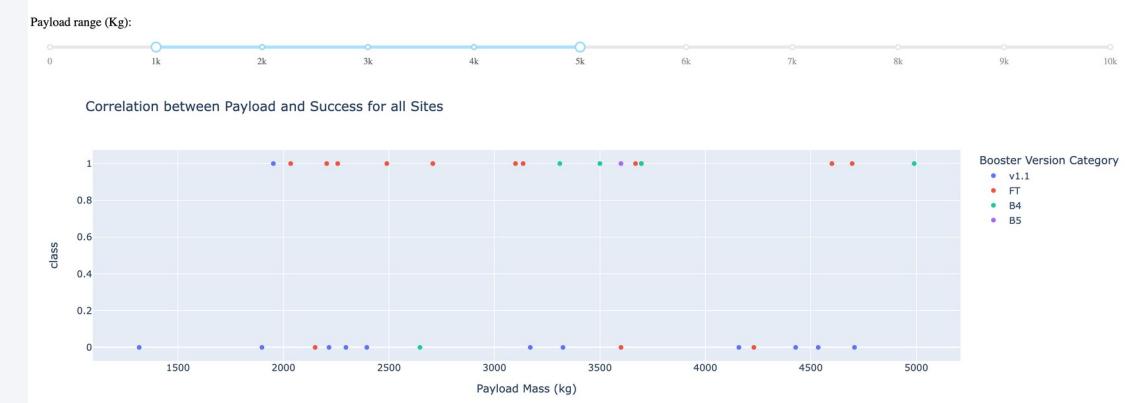
The launch site with highest launch success ratio is **KSC LC-39A**.

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.

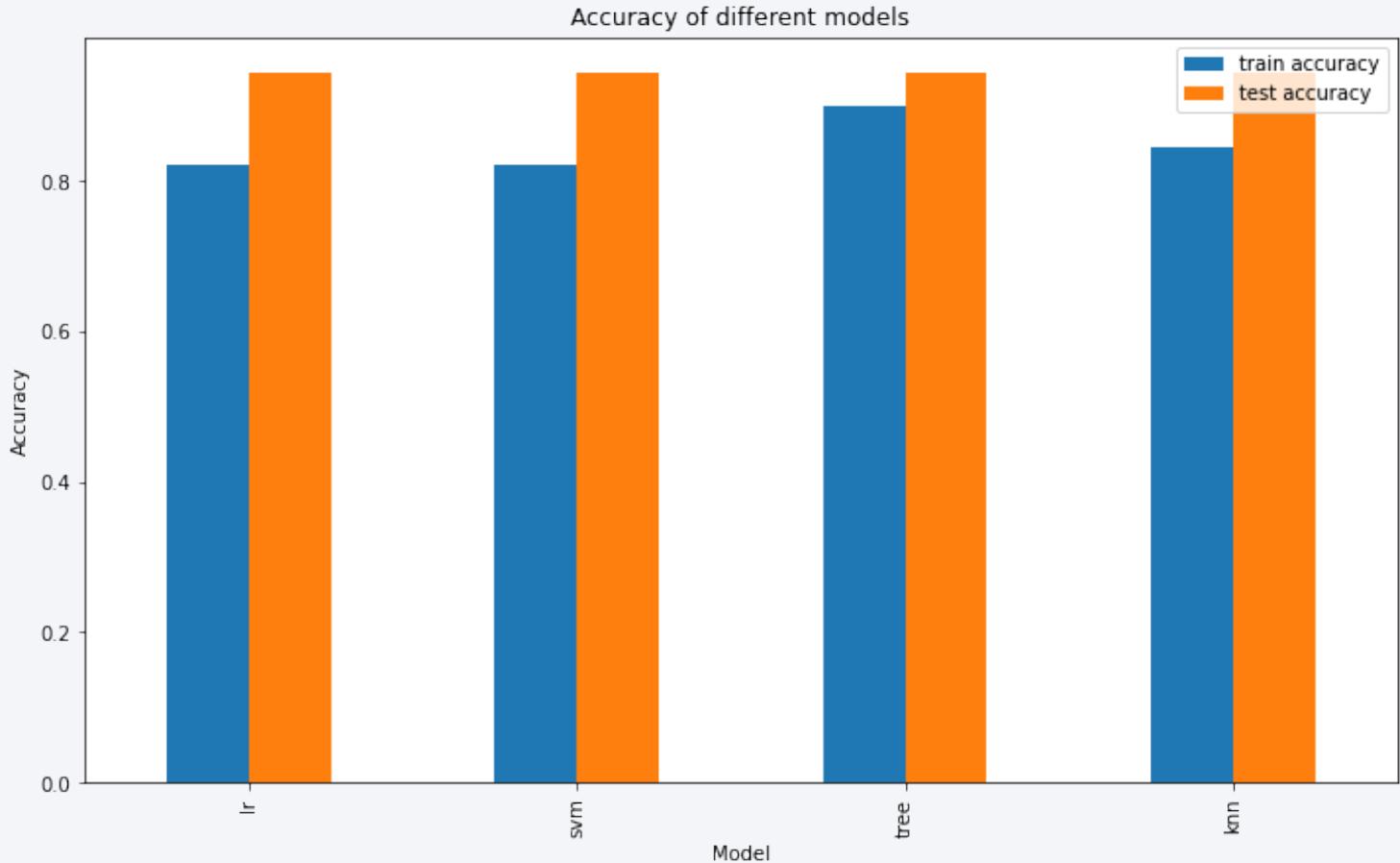


Section 5

Predictive Analysis (Classification)

Classification Accuracy

- **Decision Tree model** has the highest classification accuracy
- training accuracy 0.83
testing accuracy 0.94
- Parameter: {'criterion': 'gini', 'max_depth': 8, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 10, 'splitter': 'random'}



Confusion Matrix

- Decision Tree model can distinguish between the different classes.
- The major problem is **false positives**.



Conclusions

The dataset has 90 rows of data, with 83 columns. With 80/20 split, we have 72 rows of training data and 18 rows of testing data.

And enhanced by GridSearchCV, we trained four models which have all best performance on test data set.

Of these models, we can choose Decision Tree as our best model for predicting landing outcome of rocket.

By the decision tree, we might have some problem with false positives which probably will impact our estimation of next bid for rocket launch.

Pitfalls in Model Training

- ❑ The dataset only has 90 rows of data, but with 83 columns.
- ❑ With 80/20 split, we only have 72 records of training data.
- ❑ We have **more features than samples**! In this case, training model will lead to some unwanted results, such as **overfitting**.
- ❑ And we only have 18 test samples. Too few to find out problems.

	FlightNumber	PayloadMass	Flights	Block	ReusedCount	Orbit_ES-L1
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

Pitfalls in Model Training (continued)

- ❑ **How to handle this issue?**
- ❑ Get more data, or add regularization, or dimension reduction
- ❑ **May have a try:** since in the EDA we have found some **correlation** between some variables, maybe we can just get rid of some unimportant columns.
- ❑ Try **PCA** to reduce dimension.

	FlightNumber	PayloadMass	Flights	Block	ReusedCount	Orbit_ES-L1
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

Appendix

Share Links:

[This Assignment GitHub repository](#)

Other Reference Links:

[Test accuracy higher than training. How to interpret?](#)

[More features than observations](#)

Thank you!

