



Winning Space Race with Data Science

DAVID ZAMUDIO TABARES
NOV 11TH 2025



Outline

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

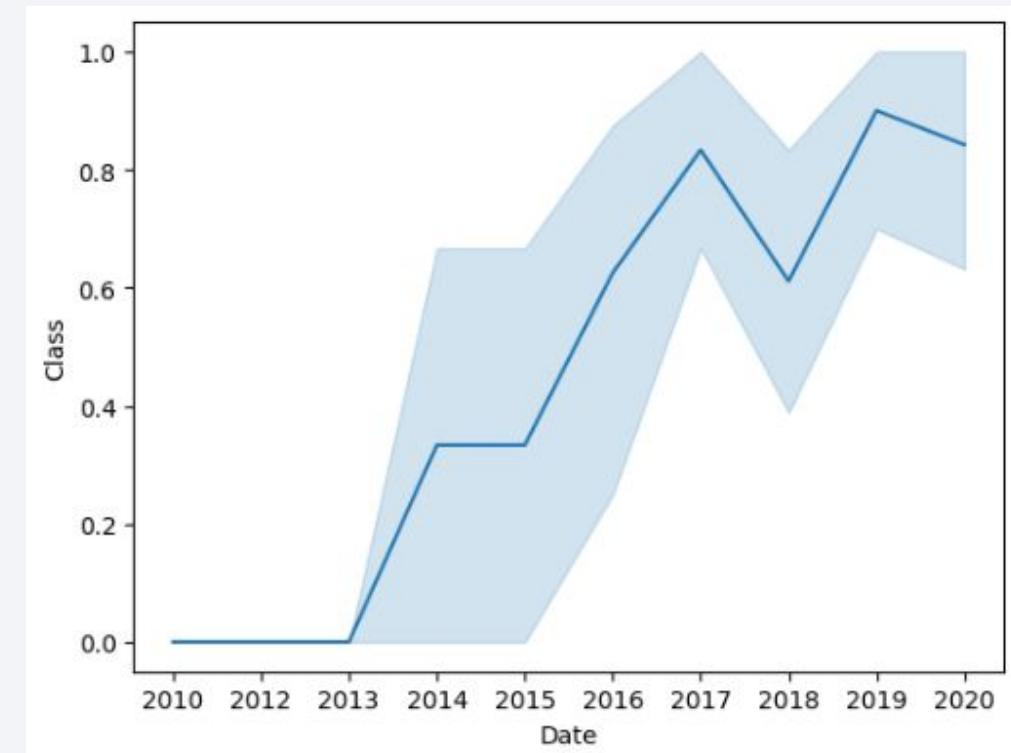
Executive Summary

- Summary of methodologies
 - In this analysis we have done a deep research on how to predict if the SPACE Y company will be able to successfully put a payload on orbit depending on multiple variables that the company is able to manipulate, such as payload, Launch site, rocket model, etc. We have done this using multiple methodologies and AI models and selected the best one, so we are able to better determine which results will be best to the company and what key areas we must focus in order to have better chances to reach our goal; reach the space and come back safely!



Executive Summary

- Summary of all results
 - In these cases, we have noted that our best models are able to predict whether the spacecraft will be able to successfully land back again or not, this at the end of the day will save the company costs by being able to reuse the same motor.. In our findings we noted that the company has been able to significantly improve the landing rate of the spaceship, and most of those successful landings had been made on the CCAFS SLC 40 landing pad and heavy payloads, which increases the benefit for the company with heavier payloads and better success rates.



Introduction

we will predict if the Falcon 9 first stage will land successfully. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore if we can determine if 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.



Section 1

Methodology

Methodology

Executive Summary

The Data was collected primarily from the space X API which included multiple features of relevant information such as:

- Payloads
- Launch Pad
- Orbit
- Booster Version
- Reused Count
- Longitude & Latitude
- Date
- Outcome

```
1 launch_dict= dict.fromkeys(column_names)
2
3 # Remove an irrelevant column
4 del launch_dict['Date and time ( )']
5
6 # Let's initial the launch_dict with each value to be an emp
7 launch_dict['Flight No.']= []
8 launch_dict['Launch site']= []
9 launch_dict['Payload']= []
10 launch_dict['Payload mass']= []
11 launch_dict['Orbit']= []
12 launch_dict['Customer']= []
13 launch_dict['Launch outcome']= []
14 # Added some new columns
15 launch_dict['Version Booster']= []
16 launch_dict['Booster landing']= []
17 launch_dict['Date']= []
18 launch_dict['Time']= []
```

This data will help us understand better how to predict the outcome of the mission.

Methodology

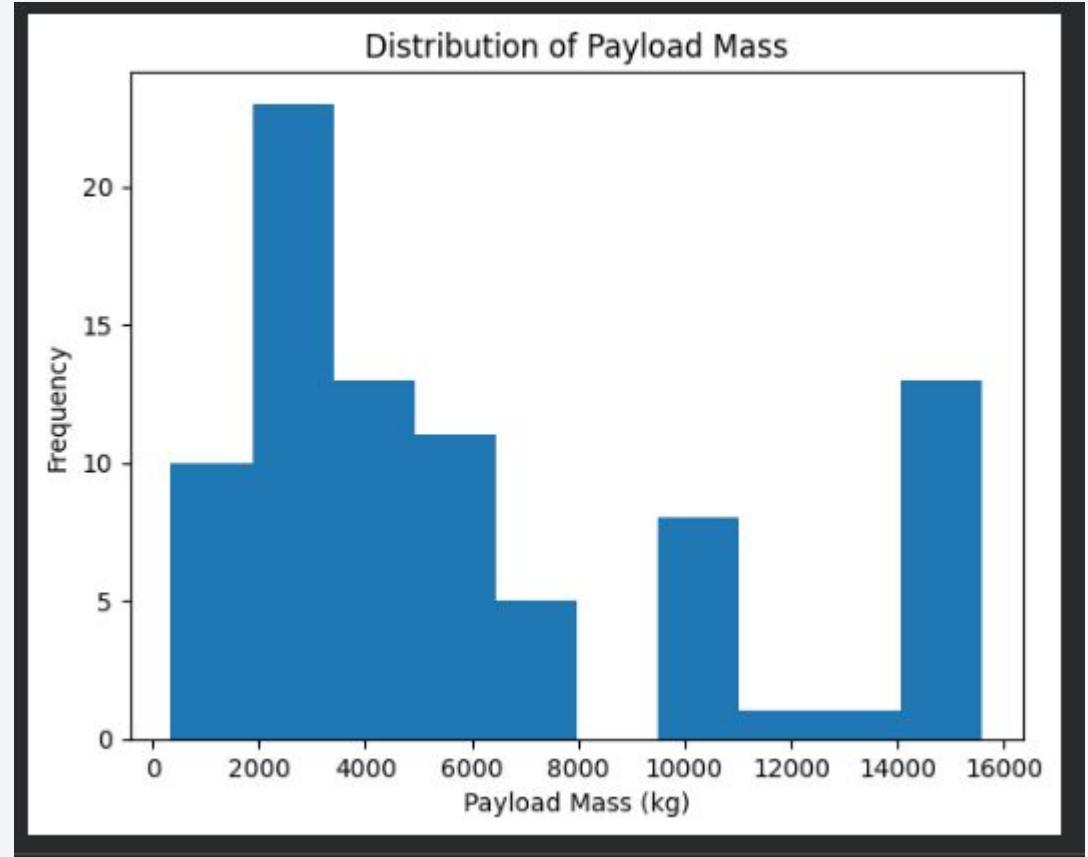
- Perform data wrangling
 - The data was processed using python functions that allowed us to be able to get relevant information after filtering it such as the booster version, launch site and many others, since the data from the API is large and not always ready to go we had to transform it... then used Pandas Dataframes for easier manipulation..

```
1 data.info()

<class 'pandas.core.frame.DataFrame'>
Index: 94 entries, 0 to 105
Data columns (total 8 columns):
 #   Column      Non-Null Count Dtype  
--- 
 0   payloads    94 non-null    object  
 1   cores       94 non-null    object  
 2   id          94 non-null    object  
 3   rocket      94 non-null    object  
 4   launchpad   94 non-null    object  
 5   flight_number 94 non-null    int64  
 6   date_utc    94 non-null    object  
 7   date        94 non-null    object  
dtypes: int64(1), object(7)
memory usage: 6.6+ KB
```

Methodology

- Perform data wrangling
 - we were interested only in the falcon 9 launches (since this is the current motor version being used for all launches) and also we had to do some feature engineering to our data set since there was some record missing the payload mass..



Methodology

- Perform data wrangling
 - Since the payload Mass distribution was normal and not skewed we went ahead and replaced those missing records with its mean.. having left 90 records of launches ready to be examined to find our answers.

```
1 data_falcon9.info()  
  
<class 'pandas.core.frame.DataFrame'>  
Index: 90 entries, 4 to 93  
Data columns (total 17 columns):  
 #   Column           Non-Null Count  Dtype     
---  --  
 0   FlightNumber    90 non-null      int64    
 1   Date             90 non-null      object    
 2   BoosterVersion   90 non-null      object    
 3   PayloadMass     90 non-null      float64   
 4   Orbit            90 non-null      object    
 5   LaunchSite       90 non-null      object    
 6   Outcome          90 non-null      object    
 7   Flights          90 non-null      int64    
 8   GridFins         90 non-null      bool     
 9   Reused           90 non-null      bool     
 10  Legs              90 non-null      bool     
 11  LandingPad       64 non-null      object    
 12  Block             90 non-null      float64   
 13  ReusedCount      90 non-null      int64    
 14  Serial            90 non-null      object    
 15  Longitude          90 non-null      float64   
 16  Latitude          90 non-null      float64  
dtypes: bool(3), float64(4), int64(3), object(7)  
memory usage: 12.9+ KB
```

Methodology

- We wanted to know some key features of our dataset using **SQL**, like
 - Launch sites names
 - Total mass
 - Average payload masses
 - First successful landing
 - Successful boosters between 4000 KG - 6000 KG
 - Number and type of mission outcomes
 - Booster versions that have carried the maximum payload mass
 - Information about boosters that have failed

Methodology

- Performed interactive visual analytics using Folium and Plotly Dash that will be shown with its results in following charts

Methodology

Perform predictive analysis using classification models

- We built our models using 4 main models for comparison
 - Logistic regression
 - Support Vector Machine (SVM)
 - Decision Tree Classification
 - K Neighbors classification

All these models (without getting into technical details) work in different ways to solve one question: Will the rocket land successfully?

Data Collection



- The Datasets were collected using the SPACE X API to get information about the orbit, payload,

Wikipedia Page

- Web Scraping from Wikipedia to be able to determine the landing outcome from the launches

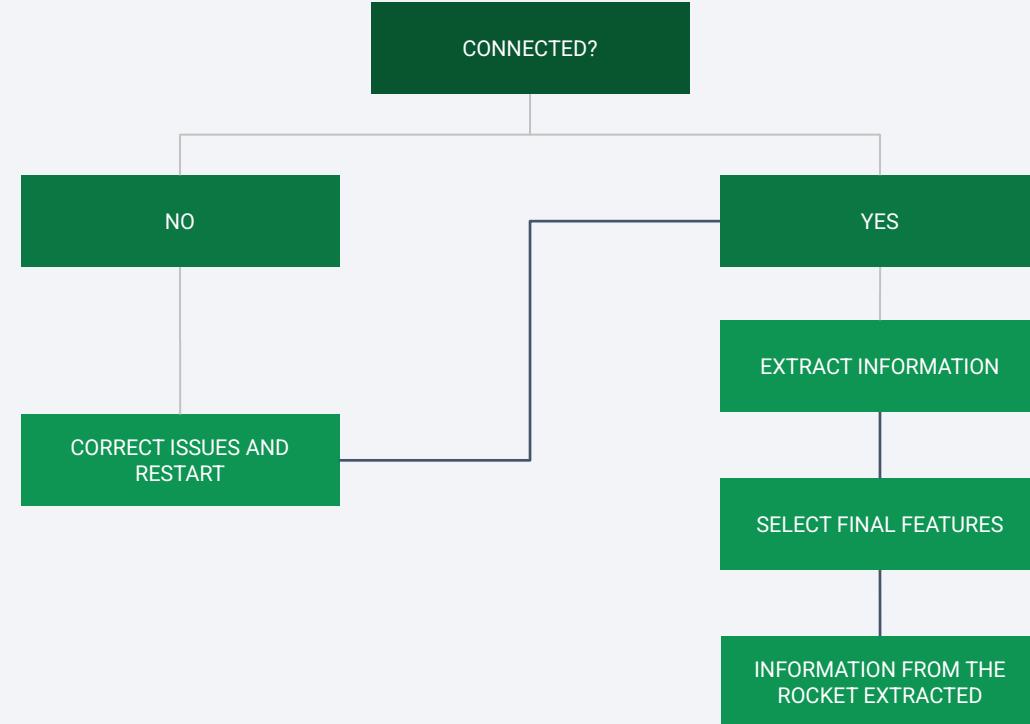
Data Wrangling

Joint both data frames to get both detailed information about the launch and then the landing outcome

Data Collection – SpaceX API

- We used the SPACEX API to extract information from the booster version, details about the rocket like payload, legs, launch site, landing pod, etc.

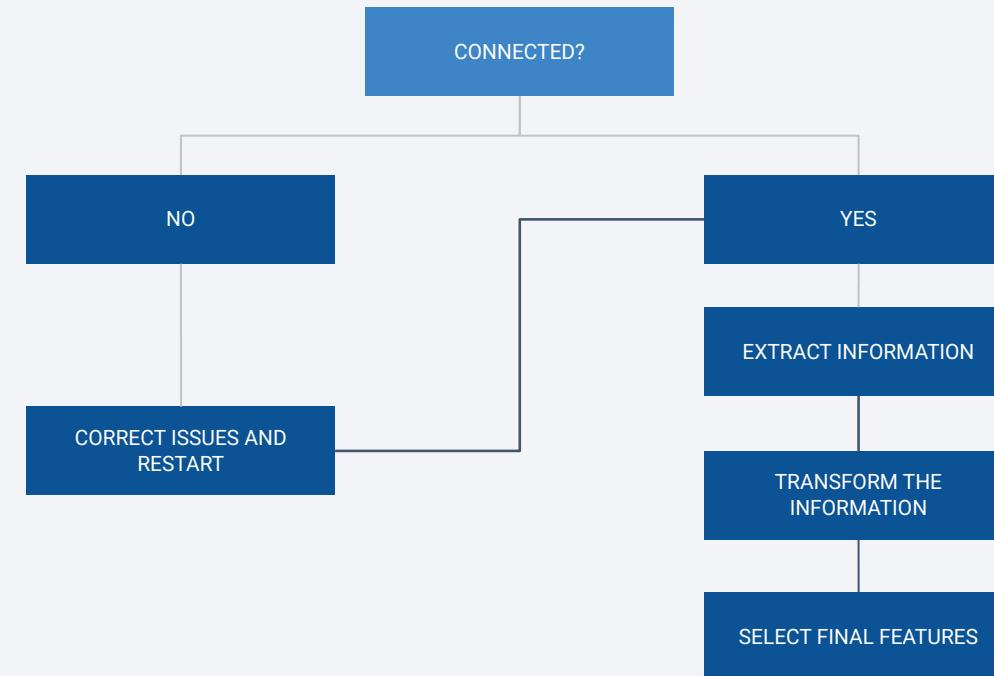
You will be able to see the document here: [LINK](#)



Data Collection - Scraping

- For the web Scraping we used the Wikipedia website were we extracted the information relevant to the launch, like its date, location, latitude, longitude and landing outcome, all information not present in the SPACEX API

You will be able to see the full doc here: [LINK](#)



Data Wrangling

- In this Data wrangling part we wanted to extract relevant information for the analysis, such as
 - What were the launch sites used for the rockets?
 - What orbits did the team used to launch its rockets to?
 - What were the possible outcomes of the rocket?

Getting this information we noted that there were multiple possible out comes, and we abstracted them down to 2; Failed or success

You will be able to see the full doc here: [LINK](#)



EDA with Data Visualization

We plotted a few but very important charts including:

- Payload Vs Flight Number: To see any trends in payload
- Launch Site Vs Flight Number: To see if there is any preference according to date or trend
- Payload Vs Launch Site: To see if there is any limitation of payload in the launch sites
- Orbit Vs Class: To see most successful orbits
- Flight Number Vs Orbit: To see any recent trends in orbit selection.
- Payload Mass Vs Orbit: To see any relation between Orbit and payload
- Date Vs Class: To see the trend for successful launches.

You will be able to see the full documentation [here](#)

EDA with SQL

We went ahead with a SQL method analyze the dataset with the main parameters:

- Retrieve the main table
 - Check the unique names for the launch sites
 - Check the total mass carried by all booster versions
 - Check the average mass carried by all booster versions
 - Check that unique results for each Landing outcome
 - Check the first success Landing
 - Count each of the results of the landing outcomes
 - Check for all the names of the boosters version
 - Check for the failures in 2015
 - ETC
- You will be able to check the full SQL analysis [here](#)

Build an Interactive Map with Folium

- We have created maps in Folium using markers to locate each side location, using circles to group every location and how many launches were made into that location, and lastly we have some lines that assisted at the moment of calculating the distance to those launch sites two points of interest.
- You will be able to check the full document for Folium [here](#)



Build a Dashboard with Plotly Dash

We have provided a selector where you can select to see each site successful launches or to see all successful launches by Launch Site

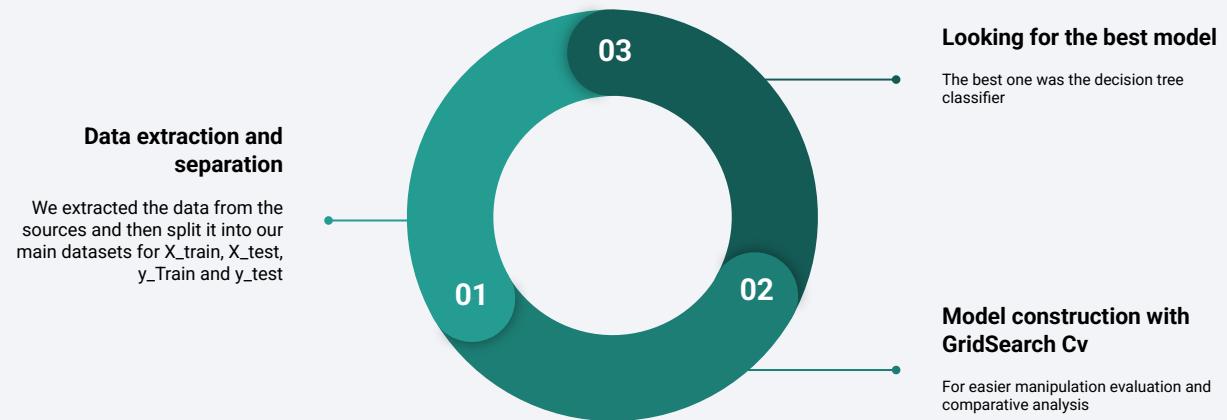
And also we have provided a way to select the payload Mass and see the successful lunches by the site and the Payload Mass

This interactions are provided so that this stakeholders are able to quickly see relevant information for quick decision making

- You will be able to check the full document for the app [here](#)

Predictive Analysis (Classification)

- At first we Define our X and Y data sets using pandas and then we divided the data set into extreme X test while trying and why test then we could logistic regression using a CV grid search that way we will be able to ever like the model and look for the best parameters as well as the score and the confusion Matrix within the very same for the support Vector machine classifier Then the decision tree and lastly we did it with the KN Neighbors. With all of this information we compare the models with their respective scores and at last the best model was the tree classifier
- You will be able to check the full document [here](#)

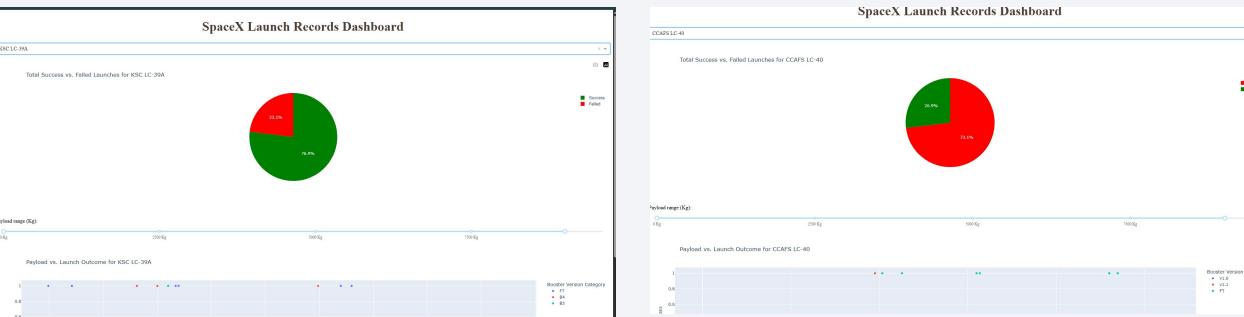


Results

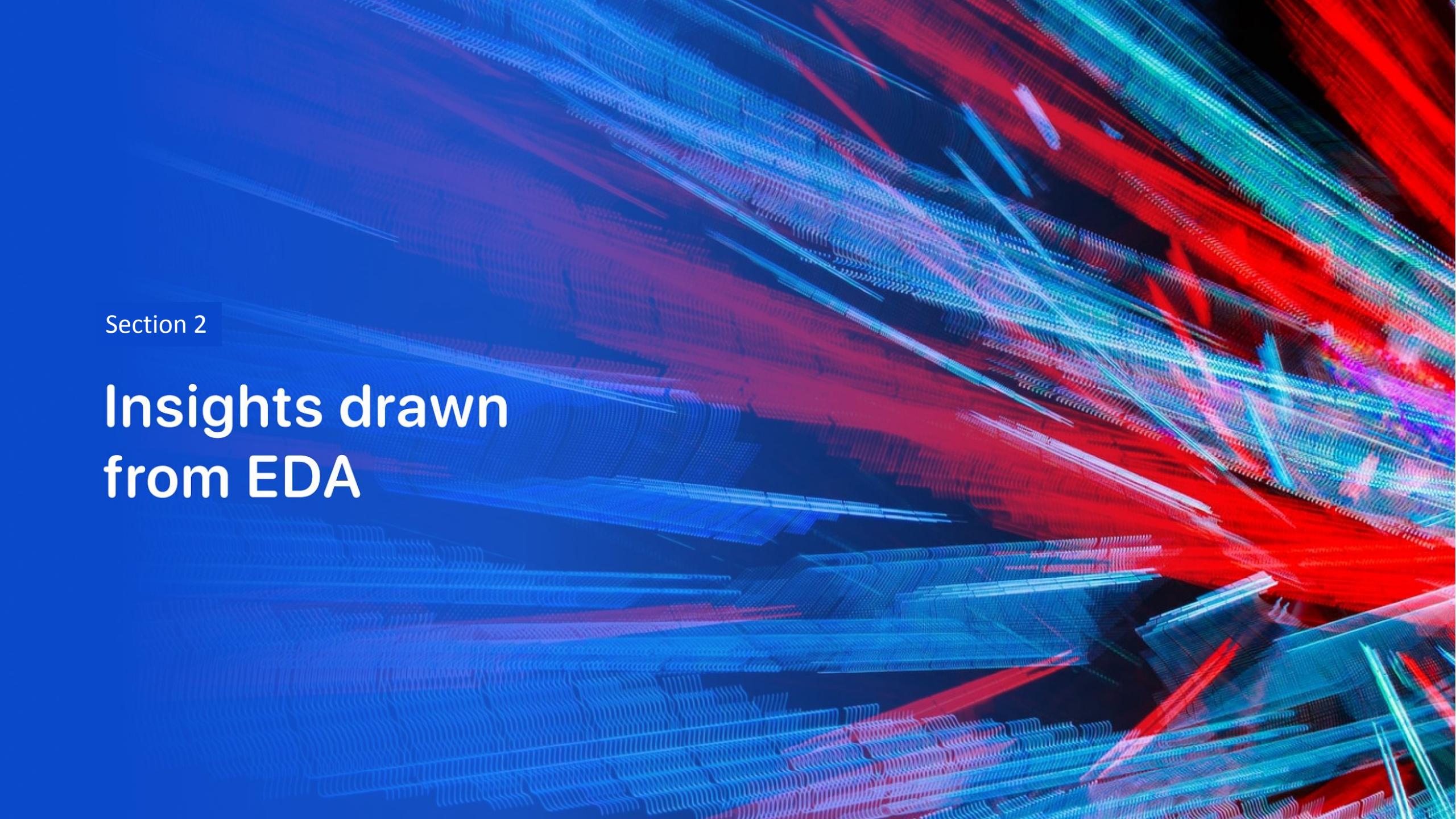
- The outcome of the missions was mostly success full

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

- Interactive analytics demo in screenshots



- The prediction was best suited using the decision tree classifier model

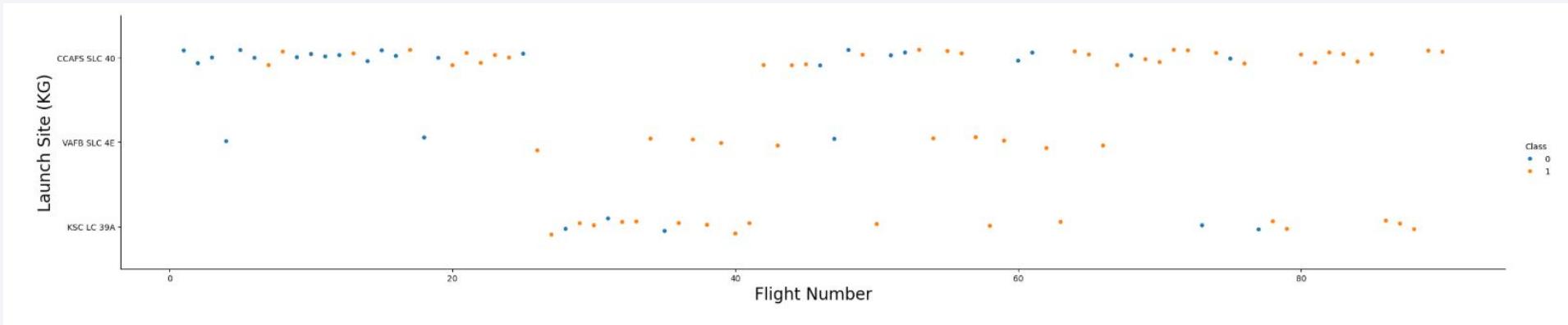
The background of the slide features a complex, abstract pattern of glowing lines. These lines are primarily blue and red, creating a sense of depth and motion. They appear to be composed of numerous small, glowing particles or dots, giving them a textured, almost liquid-like appearance. The lines converge and diverge, forming various shapes and directions across the dark, solid-colored background.

Section 2

Insights drawn from EDA

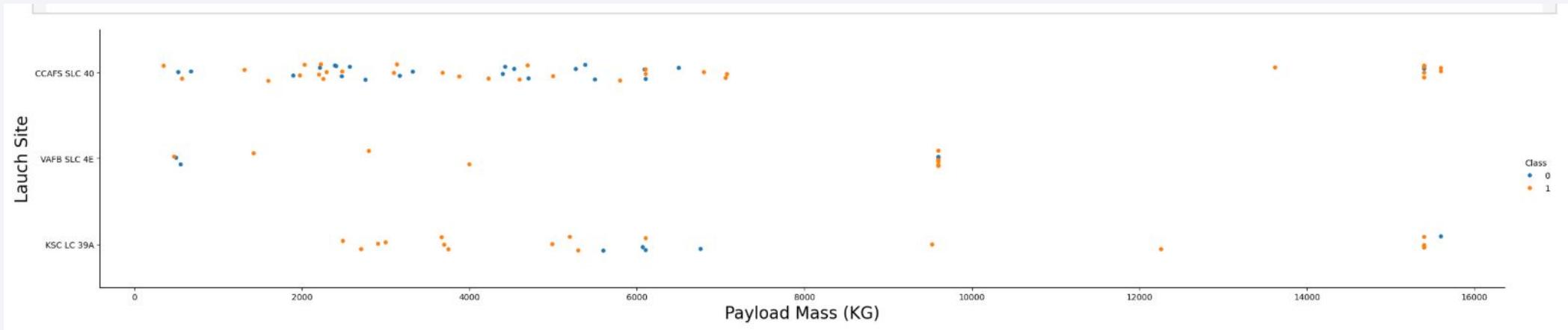
Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site



- Here we can see that the launch site SLC 40 is a must used lately and also it has the higher rate of successful Landings shown in Orange
- Also note that that same lunch site was the first one used and the lunch site SLC 4E was used most rarely

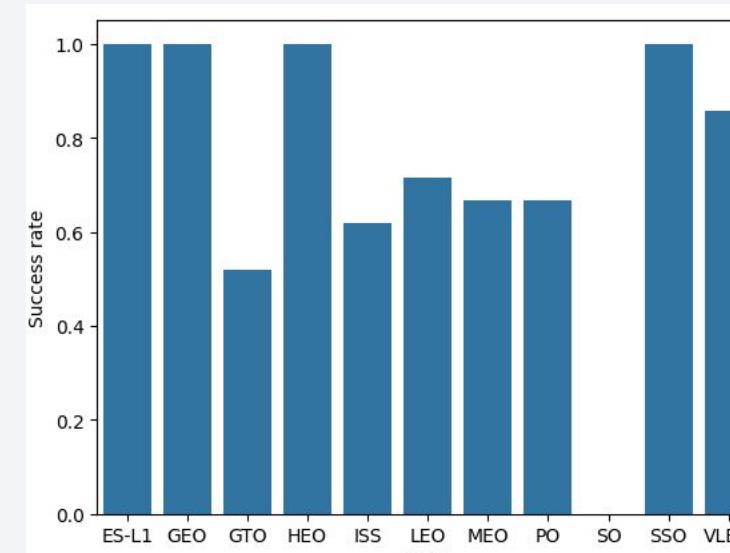
Payload vs. Launch Site



- Note that as previously stated the most frequent launch sites used for this lunches were the first one and the third one the middle one is not as used as the other two and also the success rate has increasingly being way better then on previous lunches also from a certain point upwards the Palomas is restricted and it is very same payload Mass for each launch not being as customizable by the customers as the less payload lunches

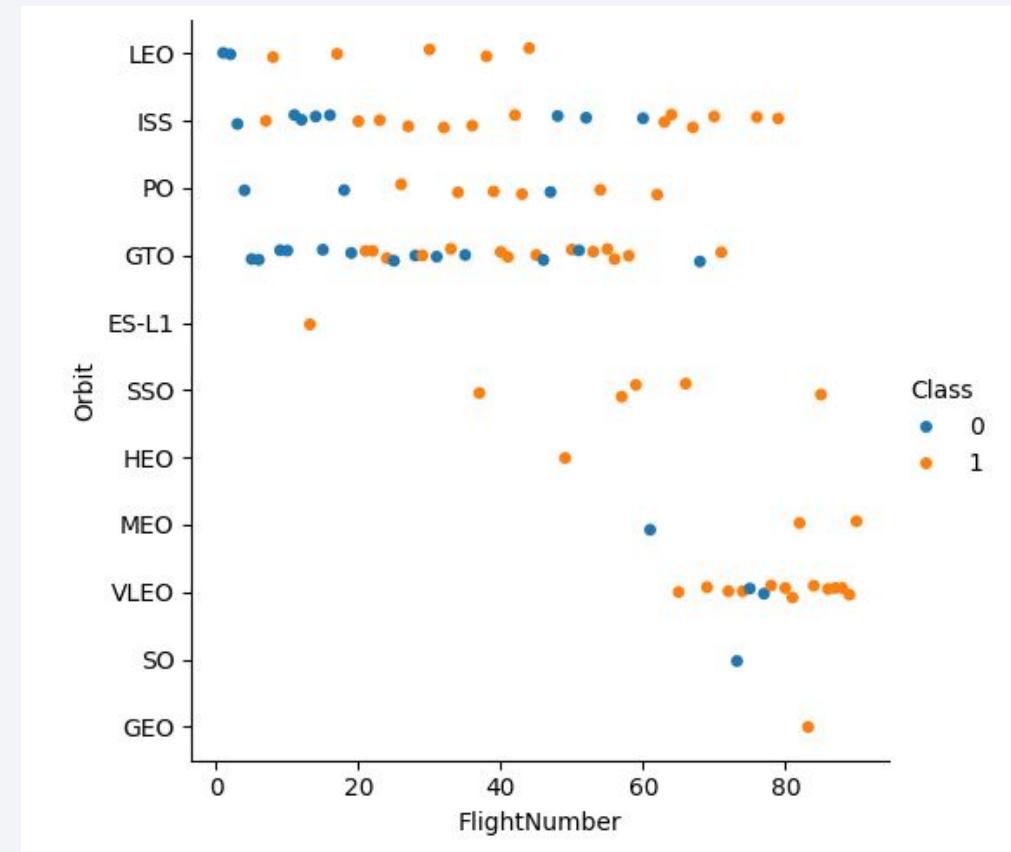
Success Rate vs. Orbit Type

The success rate for each of the launches has been always over the 50% except for the SO orbit which have never been successful and the better successful rates are those for the orbits ES-L1 the Geo also HEO and lastly the SSO orbit



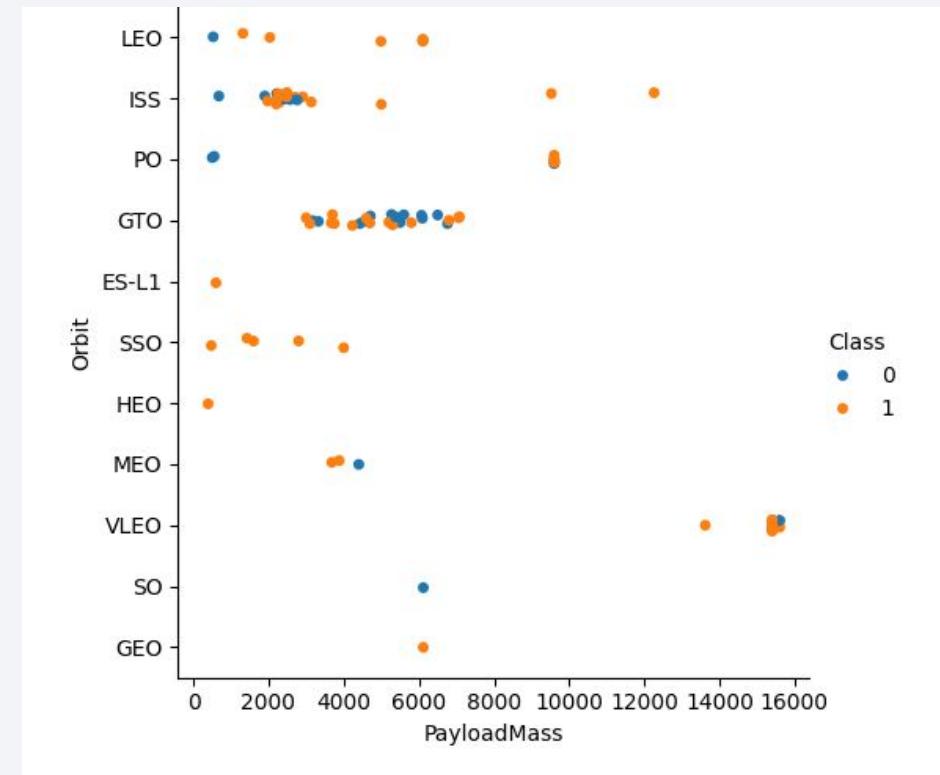
Flight Number vs. Orbit Type

Here we can see that the most successful orbit kind of launch is a VLEO orbit and also the less use orbit is the Geo orbit



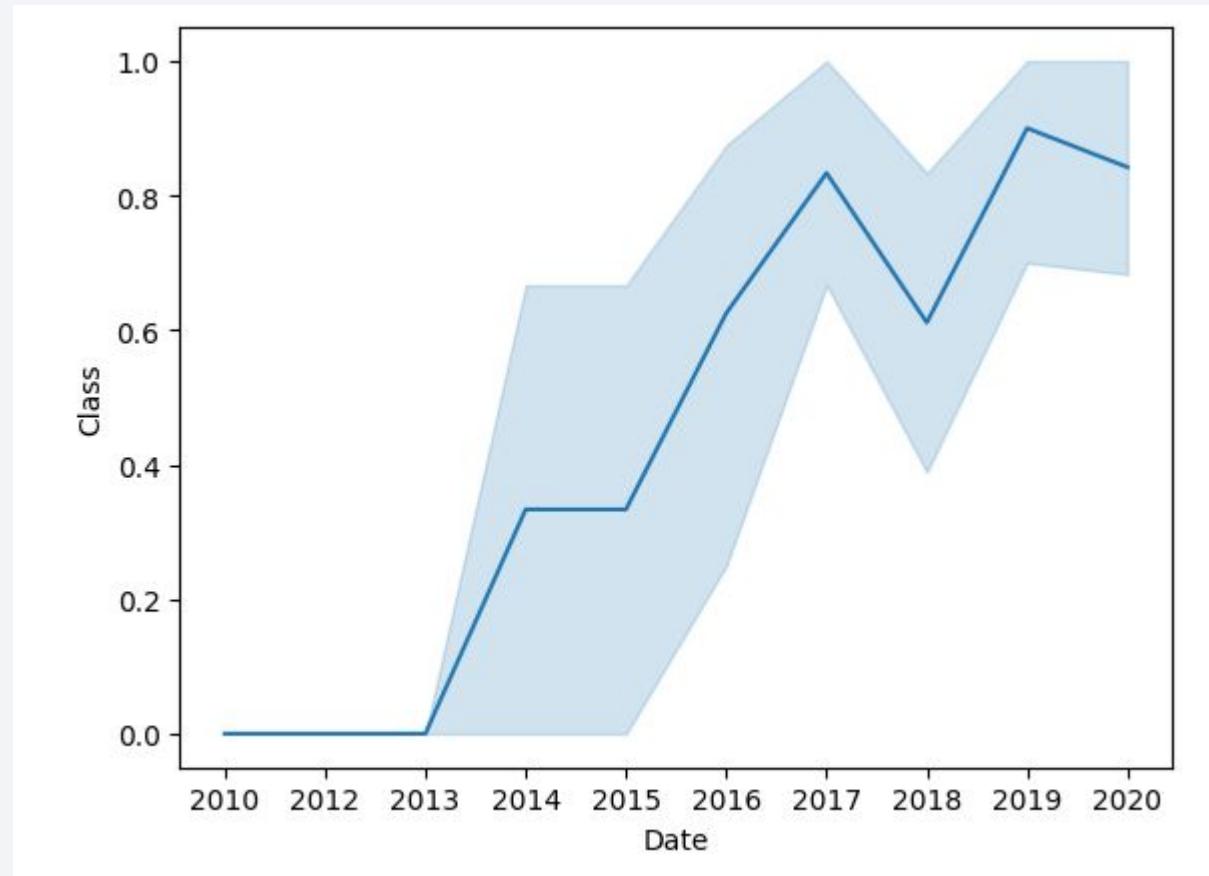
Payload vs. Orbit Type

This is the pale of mass versus the orbit and what we can see is that the payload masses for the orbit VLEO is much greater than the other ones also note that the SSO orbit has the lower payload mass and the higher success rate with 100% And also that orbits like GTO have a middle kind of success rates and also a kind of a average payload Mass



Launch Success Yearly Trend

The yearly trend for successfully launches show that between 2010 and 2013 there was no successful lunches at all and then the successful rate has been increasingly being better over the years with a little drop-down in 2018 and then came back up in 2019 meaning that the launches have been way better since they start of the project



All Launch Site Names

These are the launch sites unique names:

```
%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
```

Launch Site Names Begin with 'CCA'

These are the first five records with the large side name is starting with CCA meaning that those booster versions and launches were made in that exact same launch pad

```
[8]: %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
	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success
	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
	2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success
	2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success
	2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success

```
[9]: %sql SELECT SUM (PAYLOAD_MASS_KG_) AS Total_Mass FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'
```

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

Total Payload Mass

```
%sql SELECT SUM (PAYLOAD_MASS__KG_) AS Total_Mass FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'  
* sqlite:///my_data1.db  
Done.  
Total_Mass  
45596
```

The total payload carried by the NASA customer is 45,596 kilos
this means that the NASA customer is a very very lucrative one for
the company

Average Payload Mass by F9 v1.1

```
] : %sql SELECT AVG (PAYLOAD_MASS__KG_) AS Average_Mass_BB FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%' ;  
* sqlite:///my_data1.db  
Done.  
] : Average_Mass_BB  
2534.666666666665
```

The average payload carried by the falcon 9v1.1 is 2,534 kilos for this booster specifically

First Successful Ground Landing Date

The first successful Landing outcome on a grandpa was on December 22nd 2015

```
: %sql SELECT MIN (Date) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'  
* sqlite:///my_data1.db  
Done.  
: MIN (Date)  
-----  
2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

This is the list of boosters that have successfully landed on the Drone ship

```
3]: %sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)'

* sqlite:///my_data1.db
Done.

3]: Booster_Version

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

Total Number of Successful and Failure Mission Outcomes

- The total number of successful missions is 100 flights with just one failure mid-flight

```
: %sql SELECT Mission_Outcome, COUNT(*) AS Total FROM SPACEXTABLE 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
```

Boosters Carried Maximum Payload

This is the list of the booster versions that have carried the max amount of Payload Mass ever by SpaceX

%sql SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS_KG_ = (SELECT MAX(PAYLOAD_MASS_KG_) FROM SPACEXTABLE);
* 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

This is the list of failed Landing outcomes in drone ship and their respective booster versions for the year 2015

```
16]: %sql SELECT substr(Date, 6, 2) AS Month_of_2015 , Booster_Version, Launch_Site F
      * sqlite:///my_data1.db
Done.

16]: Month_of_2015 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

This is the rank of Landing outcomes between the date June 4th 2010 and 2017 20th of April in descending order

```
: COUNT(*) AS OutcomeCount FROM SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND '2017-04-20'  
* sqlite:///my_data1.db  
Done.  
:
```

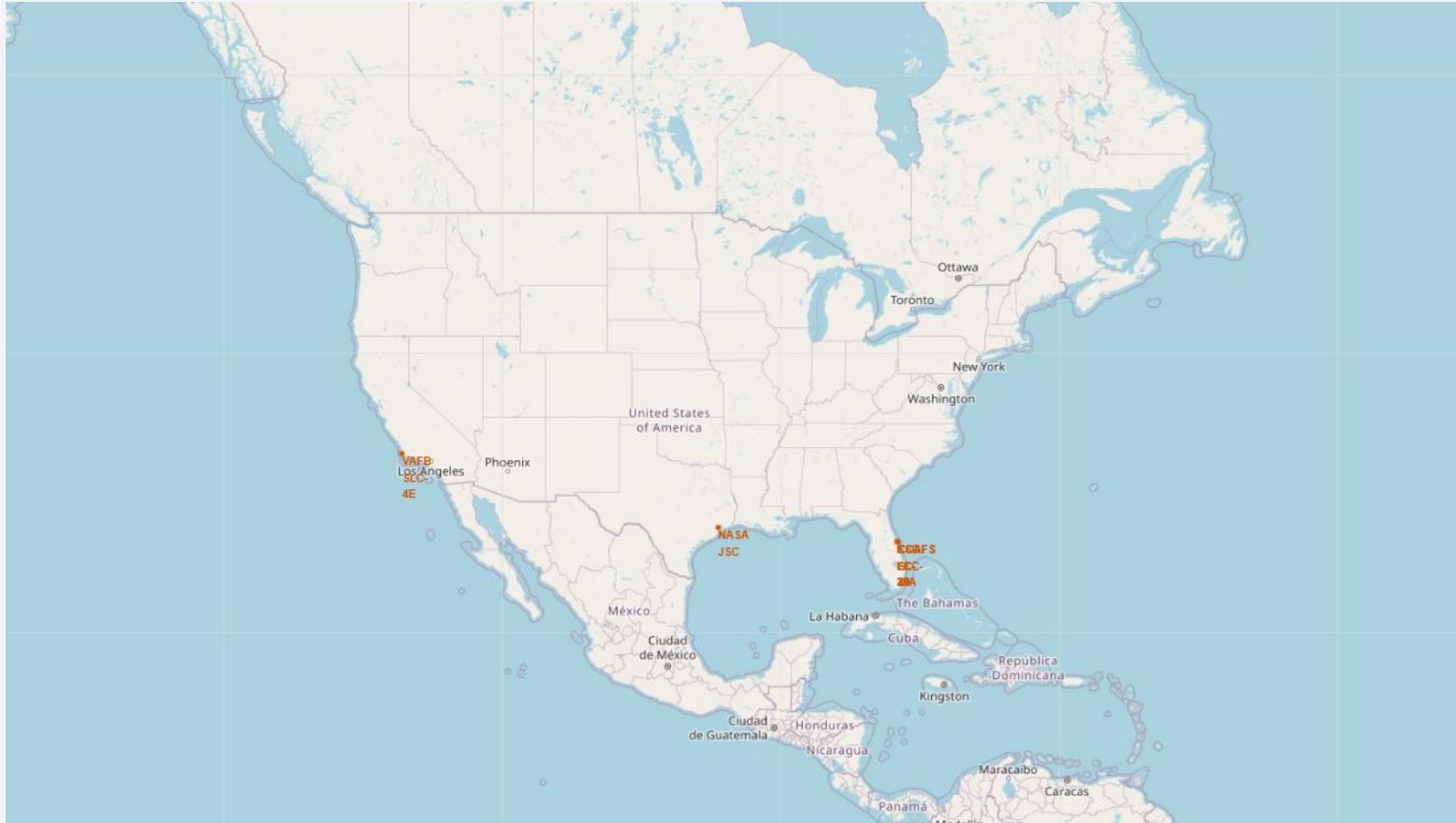
Landing_Outcome	OutcomeCount
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

The background of the slide is a photograph taken from space at night. It shows the curvature of the Earth's horizon against a dark blue sky. City lights are visible as small white dots, with larger clusters of lights indicating major urban areas. In the upper right corner, there is a faint, greenish glow of the aurora borealis or a similar atmospheric phenomenon.

Section 3

Launch Sites Proximities Analysis

Location of all launch sites including NASA's



as you can see from this map the location sites for the launching sites are near to the Sea and also as low as they can be on the equator and next to the principal cities meaning that for the mission is very important to be next to the equator next to the sea in next to a big urban area that can provide services and Logistics for the project

Count and outcome per launchsite



In this image you will be able to see that there are multiple markers on the left showing the successful (green) and failure (Red) launches and also on the right of the image you will be able to see the yellow marker with 30 that means there is another lunch site there with 30 lunches

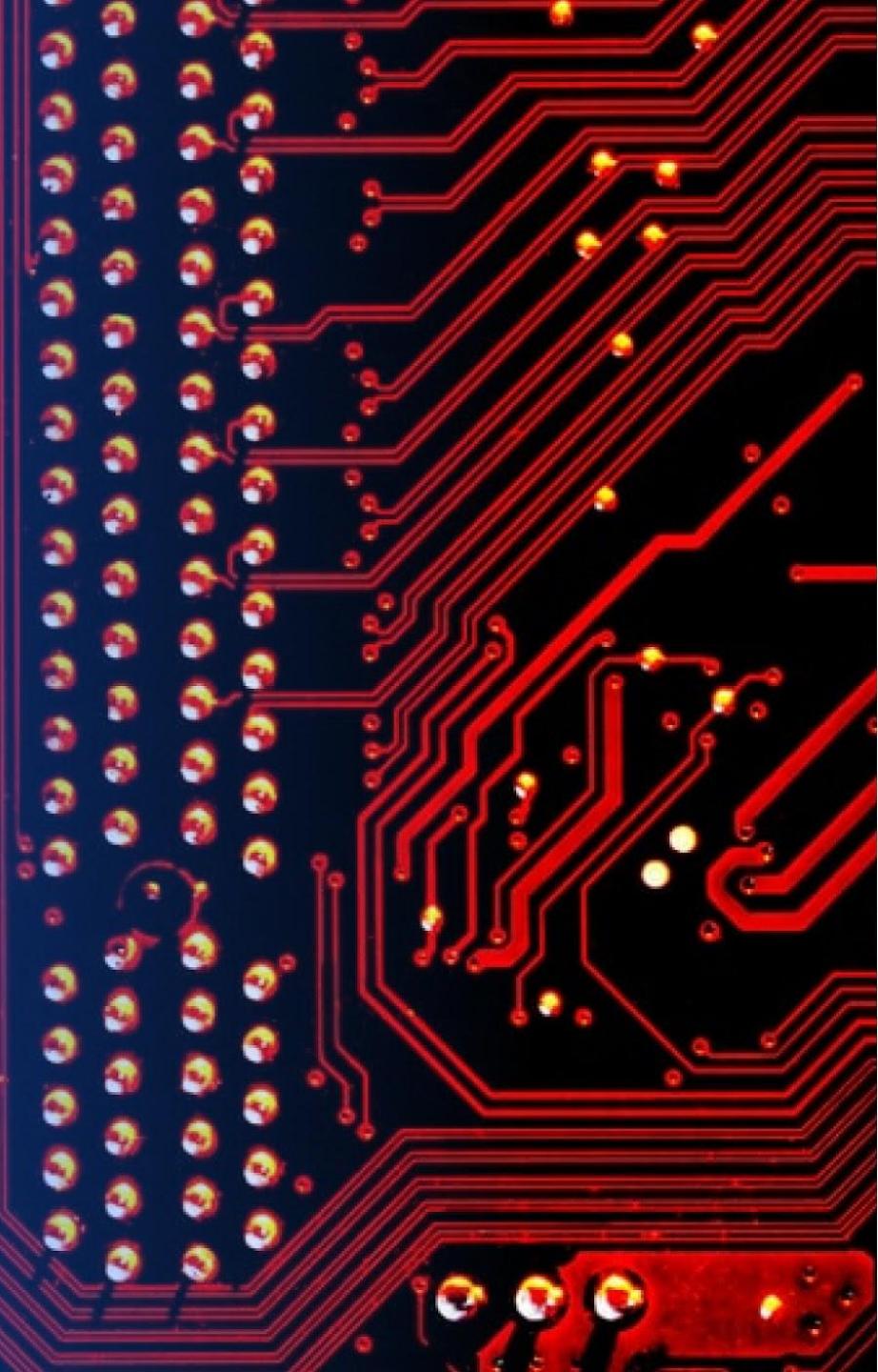
Proximity to highway from launch site



You'll be able to also see the blue line connected to the launch site to the highway that is very close to the highway so that it will be able to serve as and if a question site and also be able to serve as a logistics entrance for the rocket and for the whole project

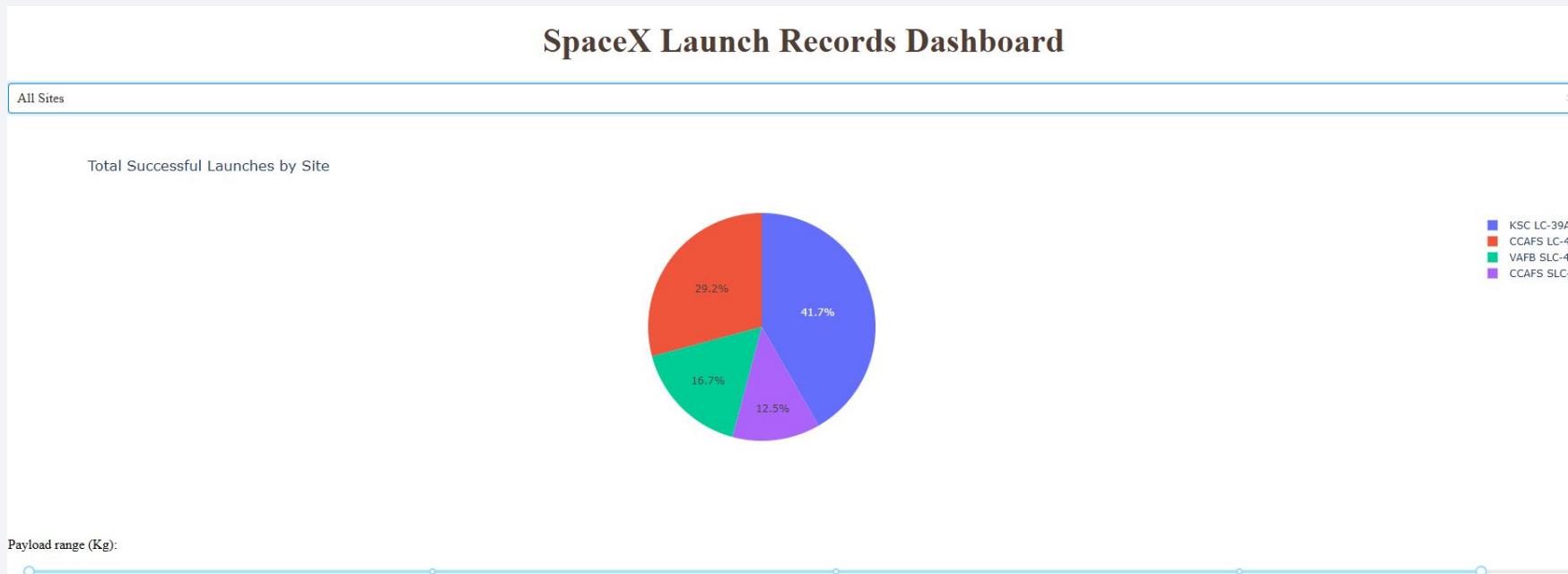
Section 4

Build a Dashboard with Plotly Dash



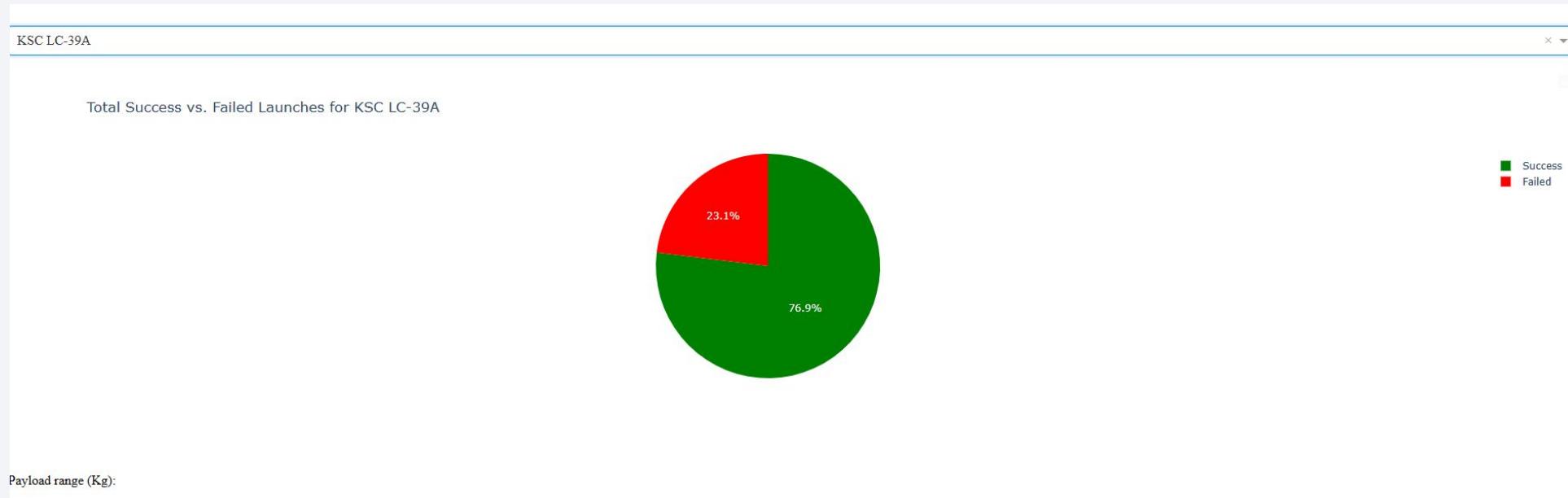
Success rates by launching site

Here you will be able to see the successful launch for all Sites this is relevant to check what launching sites have the best rate of successful launches and check which ones must be treated or improved to improve their success rates

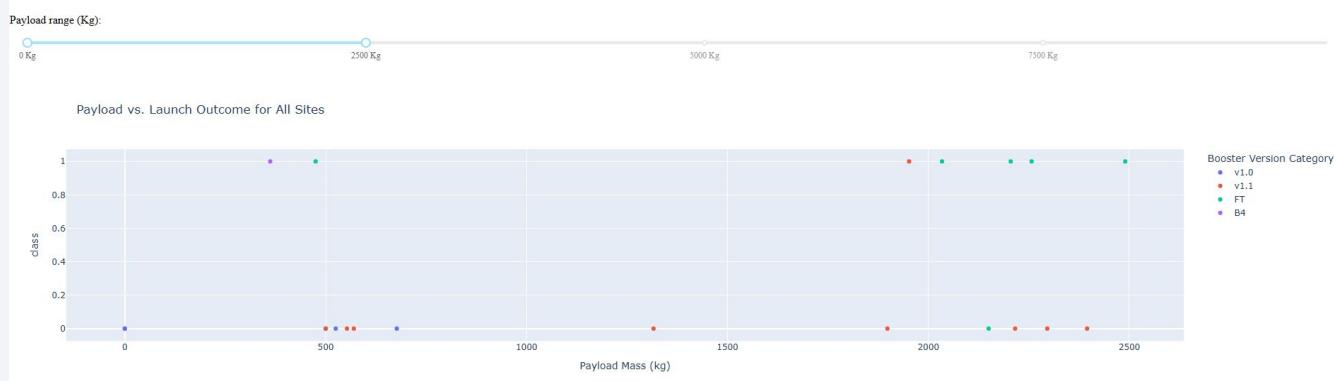


Success rate for the most successful launching site

Here you'll be able to see the success rate for the most successful launch site which is the KSC LC-39a this is important since we can see with detail what is the success rate for this launching site



Success rate by booster version and Payload mass



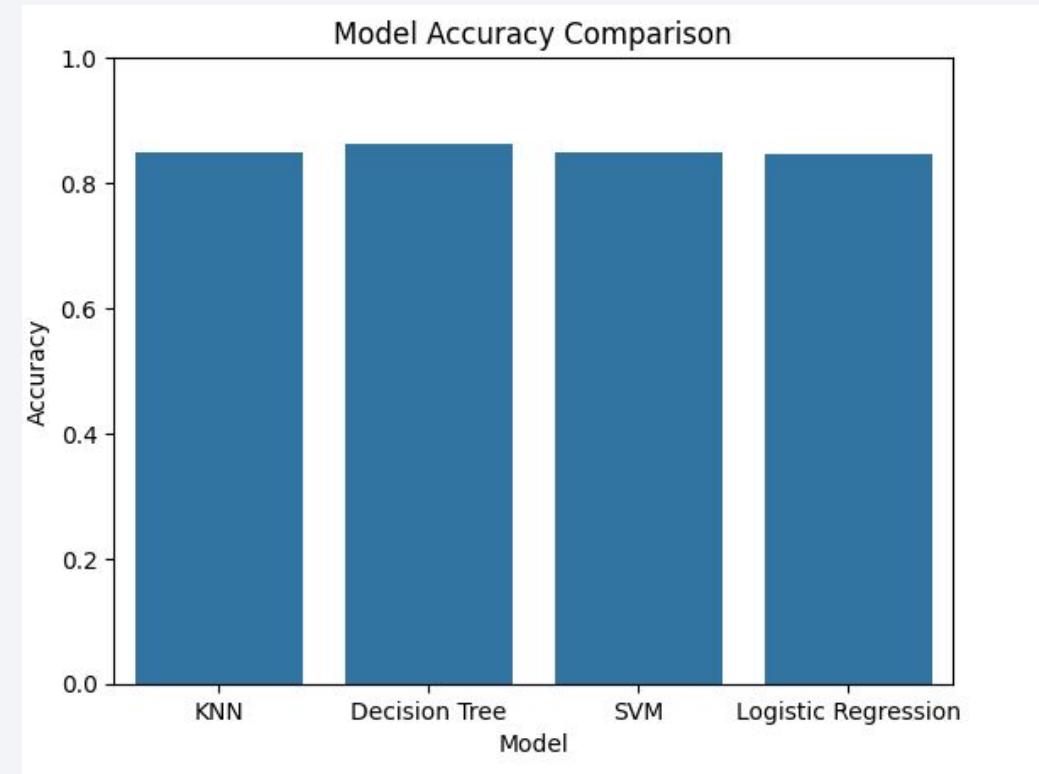
In this to a screenshots you will be able to see that for the lunches with payload Mass between 0 and 2500 kilos the most successful booster version is the Ft version alternatively between 5,000 kilos and 7,500 kilos they success rates drops and the Ft booster version is still the most successful one

Section 5

Predictive Analysis (Classification)

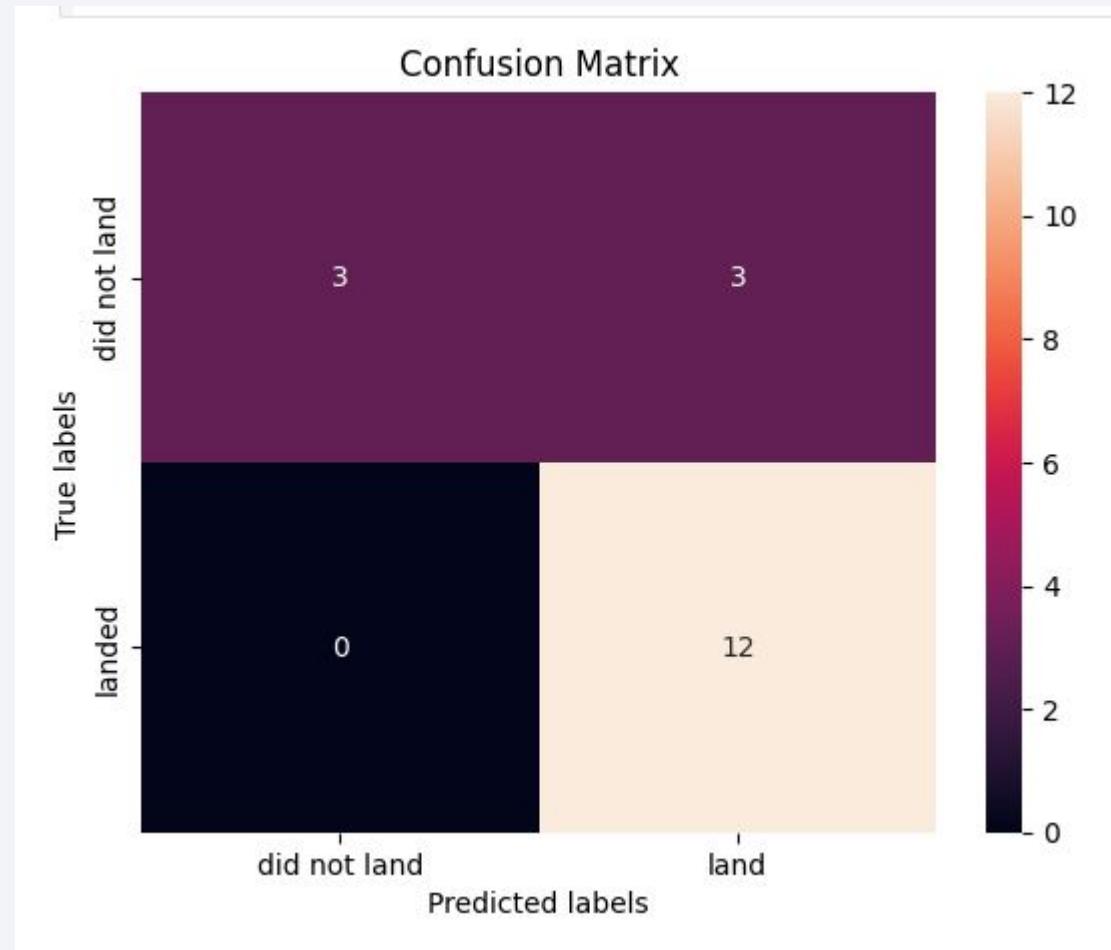
Classification Accuracy

The best model after our comparison is the decision tree. This comparison was made measuring the scores of each individual model and each model was made in a way that the best parameters for each model were selected using Gridresearch CV. And each model has only one task to predict if the rocket will land successfully or not.



Confusion Matrix

- Here's a confusion Matrix for the decision tree classifier, the Confucian Matrix explains whether the model did a correct prediction or not so in the left of the axis we have the true labels meaning the actual outcome of the mission and on the x axis we have the predicted labels meaning the prediction of our model this means that the model did correctly predicted three of the failures and correctly predicted 12 of the successes but in this model as well as the other ones the model did not correctly predict some failures which might be costly for the company believing that a lunch will be successful when it will be not



Conclusions

- The model has an accuracy around 87% of the time meaning that 87 out of 100 launches will be able to successfully predict whether or not it will last successfully for further Improvement of this model it is necessary to get more information and more lunches and over time the model will improve even better
- It is important to note also that there are some lunch insights that have not as much of a success rate as others so those ones should be considered for either an improvement or a decommission based on the company decisions
- About the payload masses and customers and booster versions the company might need to decide whether or not some payload masses with lunch insights should be considered to be related one with another so that have your payload masses hence increasingly costs should be paired

Appendix

- Note that the records only include about 100 launches which is a low amount of data available for analysis

```
] : x.head(100)
```

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

90 rows × 83 columns

Appendix

- Raw results for our models:

Comparing best models

```
n [ ]: bar_data = [knn_cv.best_score_, tree_cv.best_score_, svm_cv.best_score_, logreg_cv.best_score_]
bar_data
ut[ ]: [np.float64(0.8482142857142858),
         np.float64(0.8625),
         np.float64(0.8482142857142856),
         np.float64(0.8464285714285713)]
```

Appendix

- Possible outcomes of the mission:

Landing_Outcome
Failure (parachute)
No attempt
Uncontrolled (ocean)
Controlled (ocean)
Failure (drone ship)
Precluded (drone ship)
Success (ground pad)
Success (drone ship)
Success
Failure
No attempt

Landing_Outcome	OutcomeCount
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

Thank you!

