



IBM Developer  
SKILLS NETWORK

# Winning Space Race with Data Science

PAVAN KUMAR GODUGU  
18-07-2023



# Outline

---

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

# Executive Summary

---

- Summary of methodologies

- Data collection
- Data wrangling
- Exploratory Data Analysis with Data Visualization
- Exploratory Data Analysis with SQL
- Building an interactive map with Folium
- Building a Dashboard with Plotly Dash
- Predictive analysis (Classification)

- Summary of all results

- Exploratory Data Analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

# Introduction

---

- Project background and context

SpaceX is the most successful company of the commercial space age, making space travel affordable. The company 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. Based on public information and machine learning models, we are going to predict if SpaceX will reuse the first stage.

- Problems you want to find answers

- How do variables such as payload mass, launch site, number of flights, and orbits affect the success of the first stage landing?
- Does the rate of successful landings increase over the years?
- What is the best algorithm that can be used for binary classification in this case?



Section 1

# Methodology

# Methodology

---

## Executive Summary

- Data collection methodology:
  - Using SpaceX Rest API
  - Using Web Scrapping from Wikipedia
- Perform data wrangling
  - Dealing with missing values
  - Using One Hot Encoding to prepare the data to a binary classification
  - Filtering the data
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, tuning and evaluation of classification models to ensure the best results

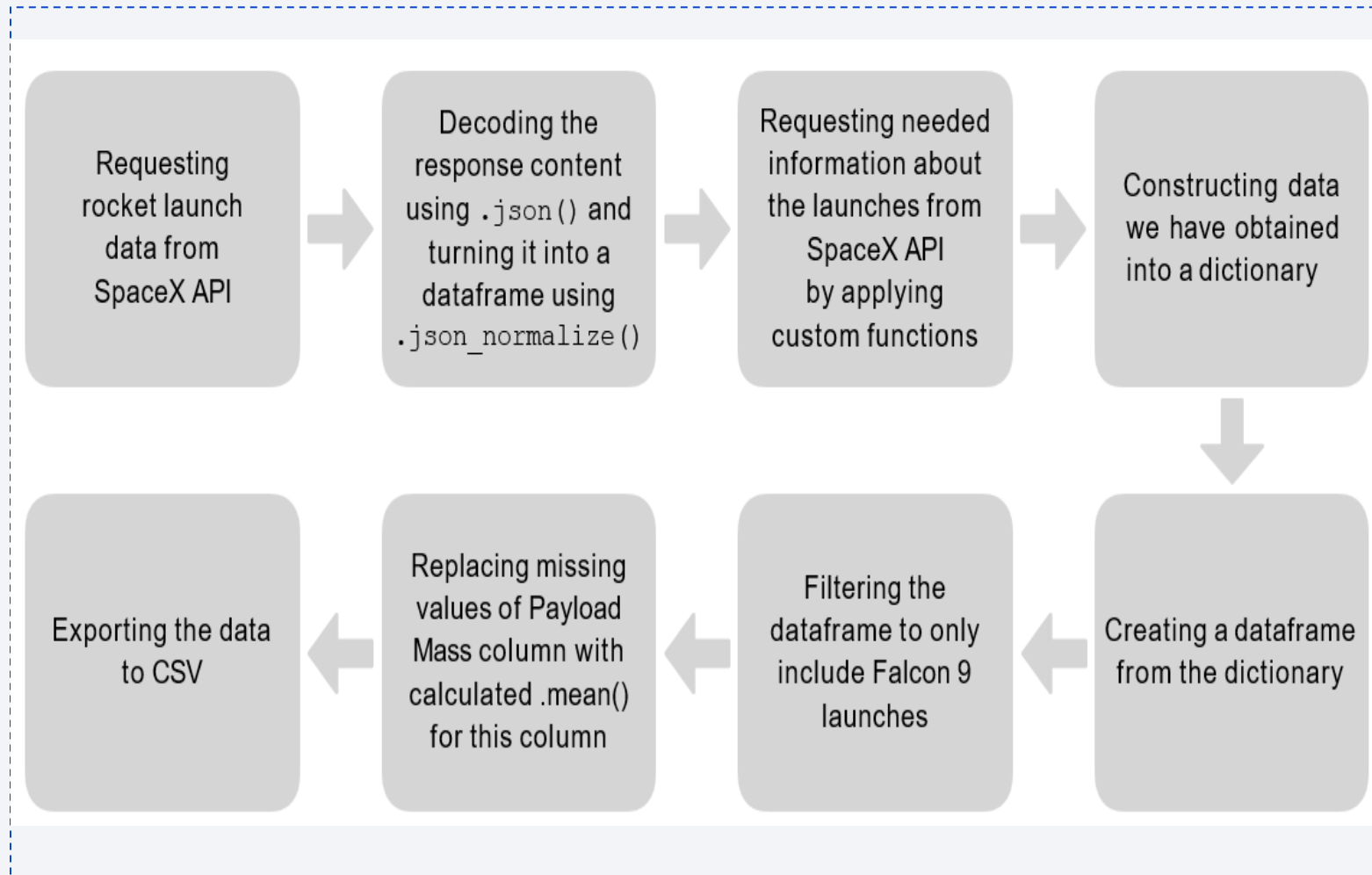
# Data Collection

---

- Data collection process involved a combination of API requests from SpaceX REST API and Web Scraping data from a table in SpaceX's Wikipedia entry.
  - We had to use both of these data collection methods in order to get complete information about the launches for a more detailed analysis.
- 
- Data Columns are obtained by using SpaceX REST API:
    - FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude
  - Data Columns are obtained by using Wikipedia Web Scraping:
    - Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

# Data Collection – SpaceX API

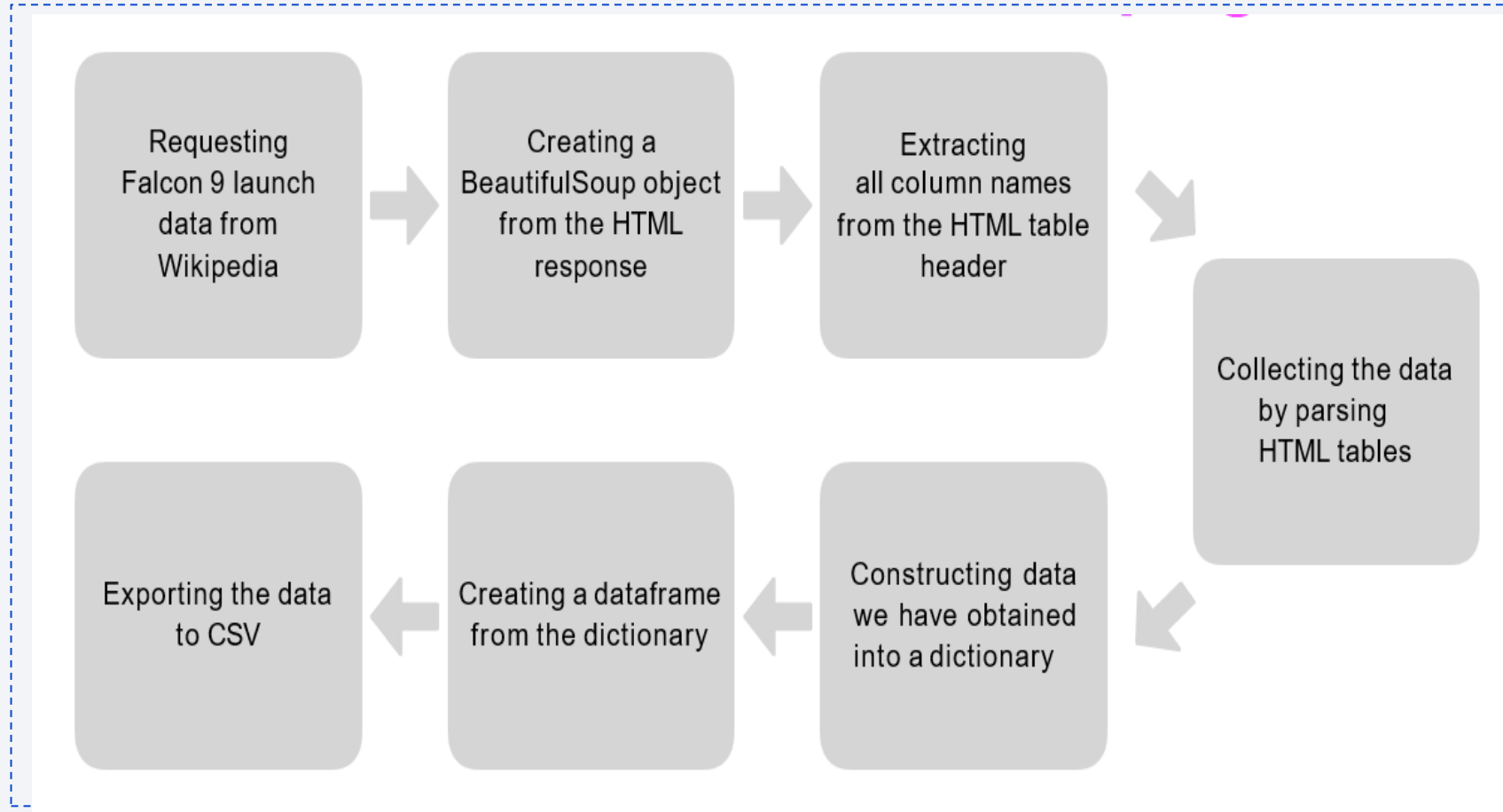
- We request data from spacex API then we store the data using json() library and try to export the file as a csv file.
- GitHub URL :
- [https://github.com/pavankumarmargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/spacex-data-collection-api.ipynb](https://github.com/pavankumarmargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/spacex-data-collection-api.ipynb)





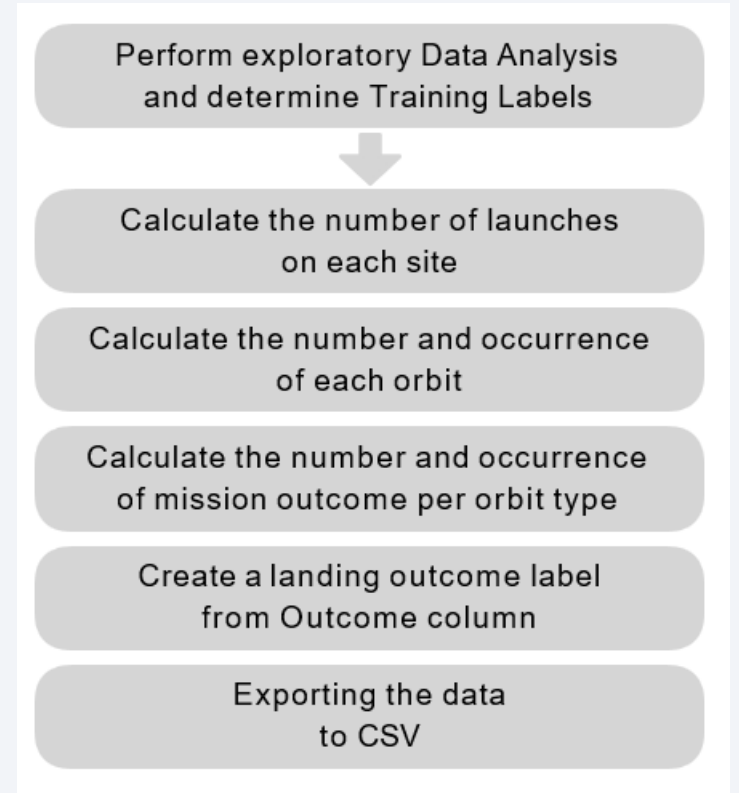
# Data Collection - Scraping

- Request launch data from wikipedia and collect the data by parsing the html responses using BeautifulSoup package.
- Github Link:
  - [https://github.com/pavankumarg o/ibmdatascience/blob/a6c91de274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/webscraping.ipynb](https://github.com/pavankumarg o/ibmdatascience/blob/a6c91de274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/webscraping.ipynb)



# Data Wrangling

- In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship.
- We mainly convert those outcomes into Training Labels with “1” means the booster successfully landed, “0” means it was unsuccessful.
- GithubLink:  
[https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/spacex-data\\_wrangling\\_jupyterlite.jupyterlite.ipynb](https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/spacex-data_wrangling_jupyterlite.jupyterlite.ipynb)



# EDA with Data Visualization

---

Charts were plotted between following pairs for better understanding:

Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit Type vs. Success Rate, Flight Number vs. Orbit Type, Payload Mass vs Orbit Type and Success Rate Yearly Trend

Scatter plots show the relationship between variables. If a relationship exists, they could be used in machine learning model.

Bar charts show comparisons among discrete categories. The goal is to show the relationship between the specific categories being compared and a measured value.

Line charts show trends in data over time (time series).

- Github link:  
[https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/eda-dataviz.ipynb.jupyterlite.ipynb](https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/eda-dataviz.ipynb.jupyterlite.ipynb)

# EDA with SQL

---

- Performed SQL queries:
  - Displaying the names of the unique launch sites in the space mission
  - Displaying 5 records where launch sites begin with the string 'CCA'
  - Displaying the total payload mass carried by boosters launched by NASA (CRS)
  - Displaying average payload mass carried by booster version F9 v1.1
  - Listing the date when the first successful landing outcome in ground pad was achieved
  - Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - Listing the total number of successful and failure mission outcomes
  - Listing the names of the booster versions which have carried the maximum payload mass
  - Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015
  - Ranking the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 in descending order

# Build an Interactive Map with Folium

---

- Markers of all Launch Sites:
  - Added Marker with Circle, Popup Label and Text Label of NASA Johnson Space Center using its latitude and longitude coordinates as a start location.
  - Added Markers with Circle, Popup Label and Text Label of all Launch Sites using their latitude and longitude coordinates to show their geographical locations and proximity to Equator and coasts.
- Coloured Markers of the launch outcomes for each Launch Site:
  - Added coloured Markers of success (Green) and failed (Red) launches using Marker Cluster to identify which launch sites have relatively high success rates.
- Distances between a Launch Site to its proximities:
  - Added coloured Lines to show distances between the Launch Site KSC LC-39A (as an example) and its proximities like Railway, Highway, Coastline and Closest City.
  - Github Link:  
[https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/launch\\_site\\_location.jupyterlite.ipynb](https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/launch_site_location.jupyterlite.ipynb)

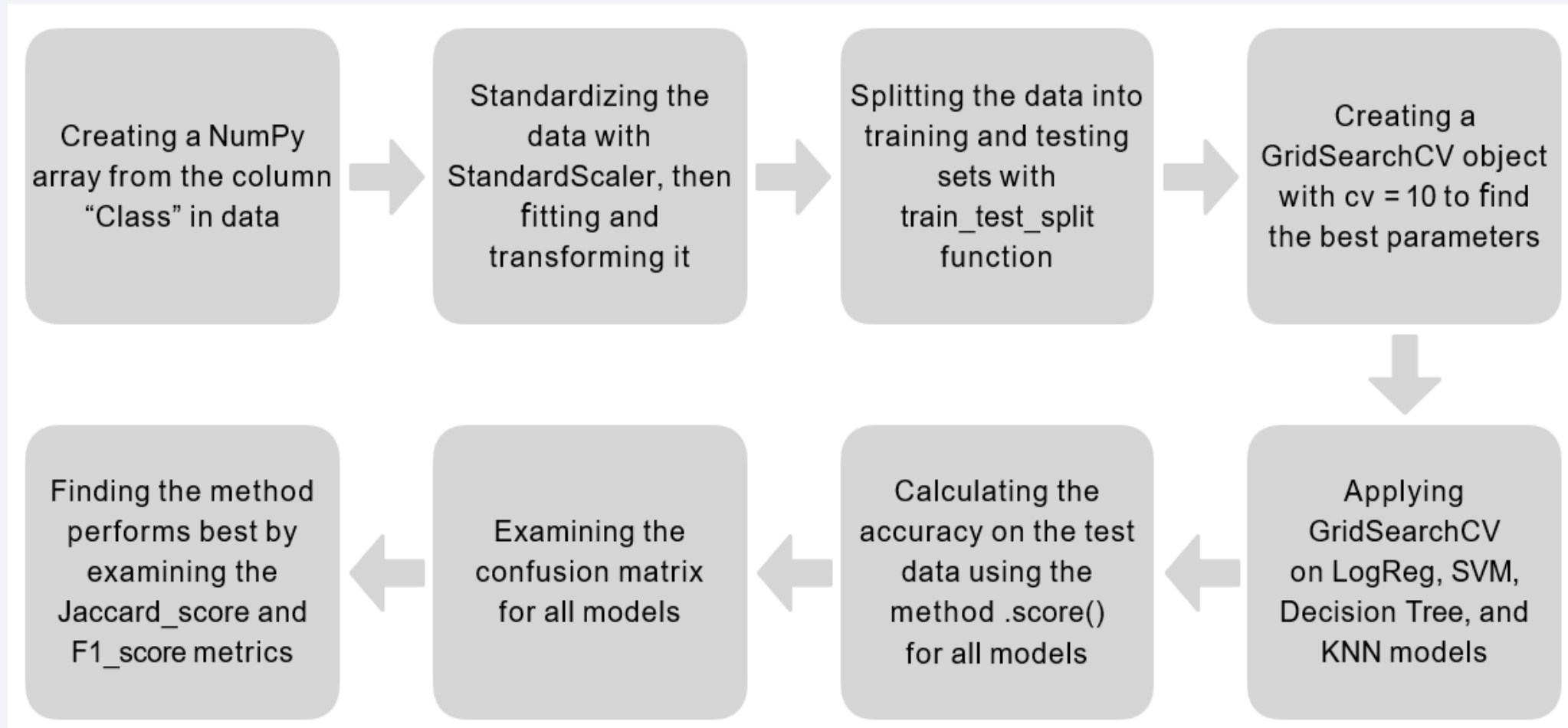


# Build a Dashboard with Plotly Dash

---

- Launch Sites Dropdown List:
  - Added a dropdown list to enable Launch Site selection.
- Pie Chart showing Success Launches (All Sites/Certain Site):
  - Added a pie chart to show the total successful launches count for all sites and the Success vs. Failed counts for the site, if a specific Launch Site was selected.
- Slider of Payload Mass Range:
  - Added a slider to select Payload range.
- Scatter Chart of Payload Mass vs. Success Rate for the different Booster Versions:
  - Added a scatter chart to show the correlation between Payload and Launch Success.
- GithubURL:[https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/spacex\\_dash\\_app.py](https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/spacex_dash_app.py)

# Predictive Analysis (Classification)



Github URL:

[https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied\\_data\\_science\\_capstone/SpaceX\\_Machine\\_Learning\\_Prediction\\_Part\\_5.jupyterlite.ipynb](https://github.com/pavankumargo/ibmdatascience/blob/a6c91dee274f35959418dd075c437bd291ab905d/Applied_data_science_capstone/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb)

# Results

---

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



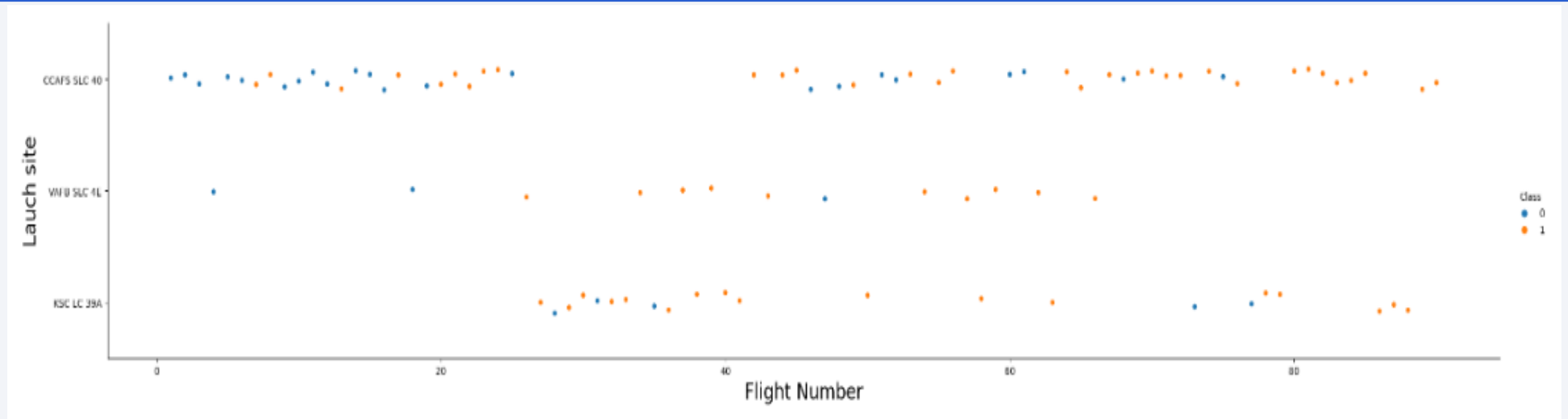
The background of the slide is an abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks in shades of red and cyan. A faint, light blue grid pattern is also visible, particularly in the lower-left quadrant. The overall effect is dynamic and technological.

Section 2

# Insights drawn from EDA



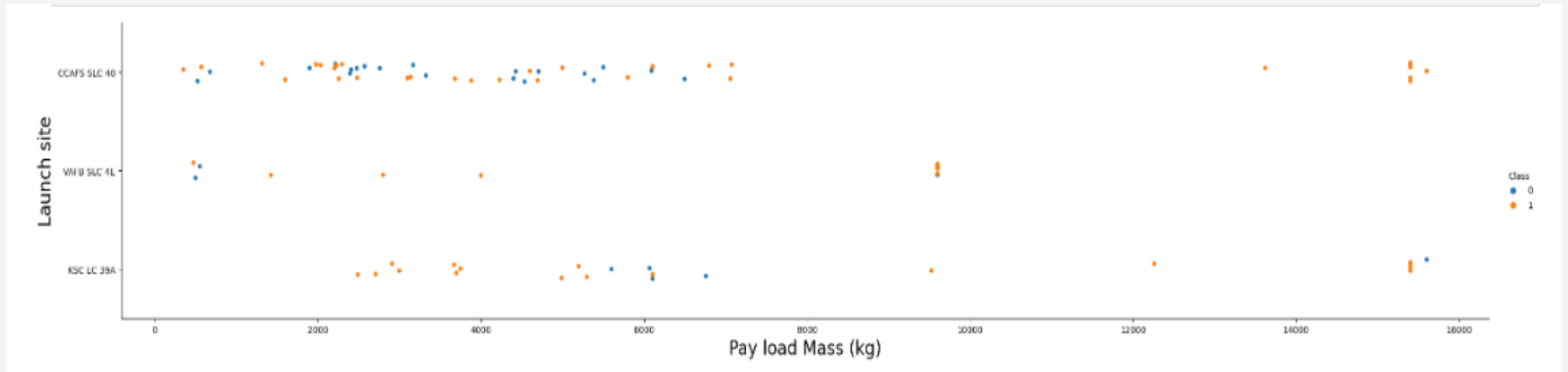
# Flight Number vs. Launch Site



- Explanation:
  - The earliest flights all failed while the latest flights all succeeded.
  - The CCAFS SLC 40 launch site has about a half of all launches.
  - VAFB SLC 4E and KSC LC 39A have higher success rates.
  - It can be assumed that each new launch has a higher rate of success.



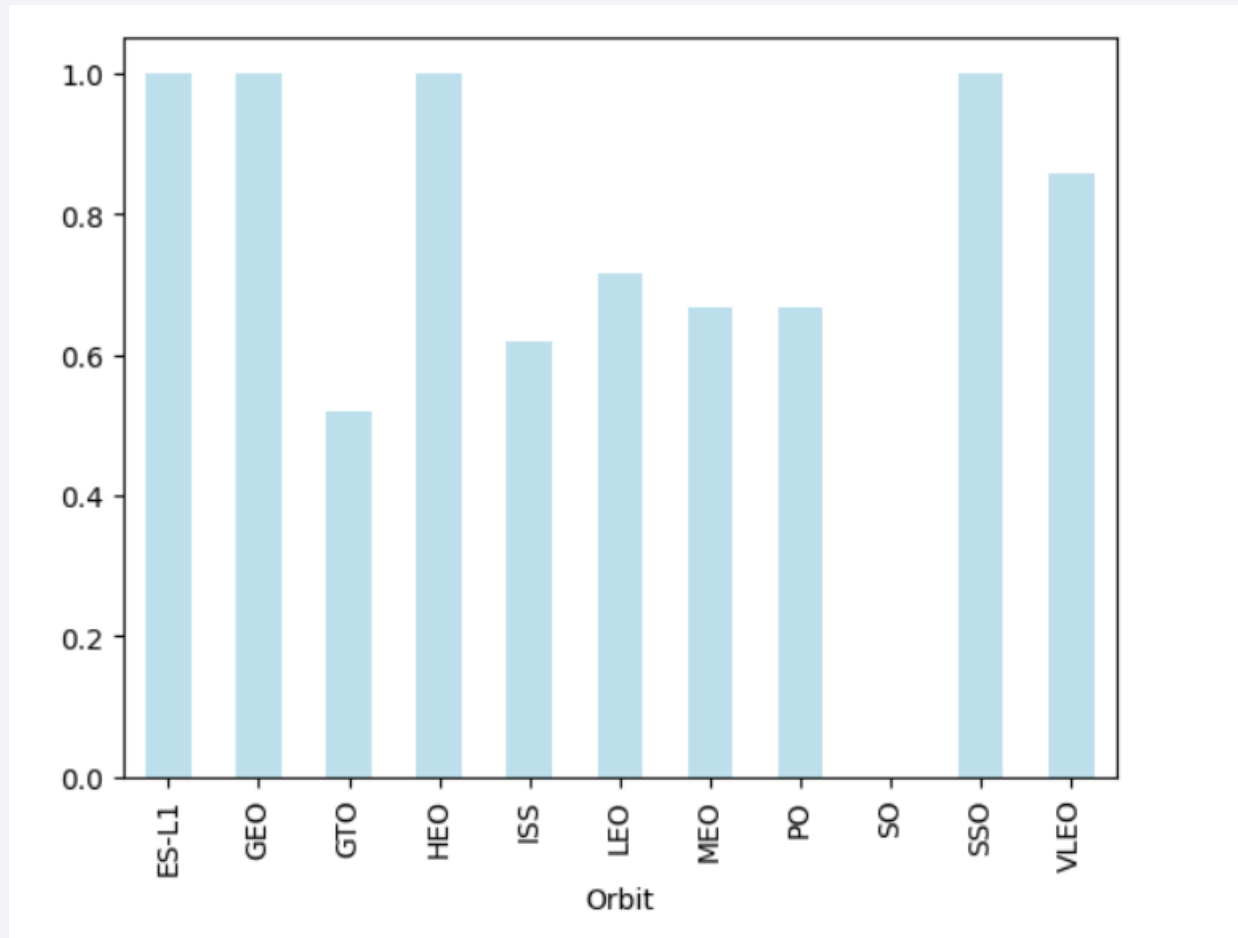
# Payload vs. Launch Site



- Explanation:
  - For every launch site the higher the payload mass, the higher the success rate.
  - Most of the launches with payload mass over 7000 kg were successful.
  - KSC LC 39A has a 100% success rate for payload mass under 5500 kg too

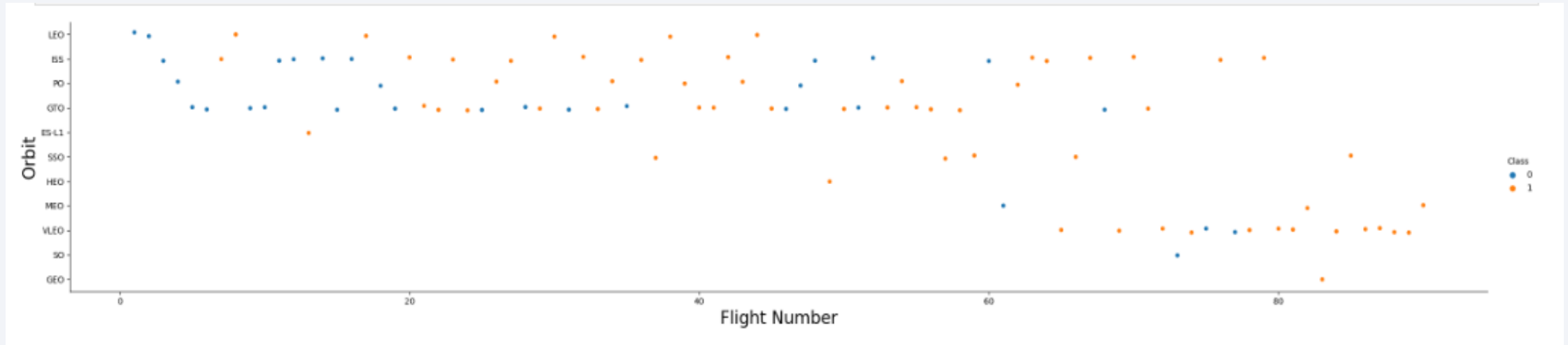
# Success Rate vs. Orbit Type

---



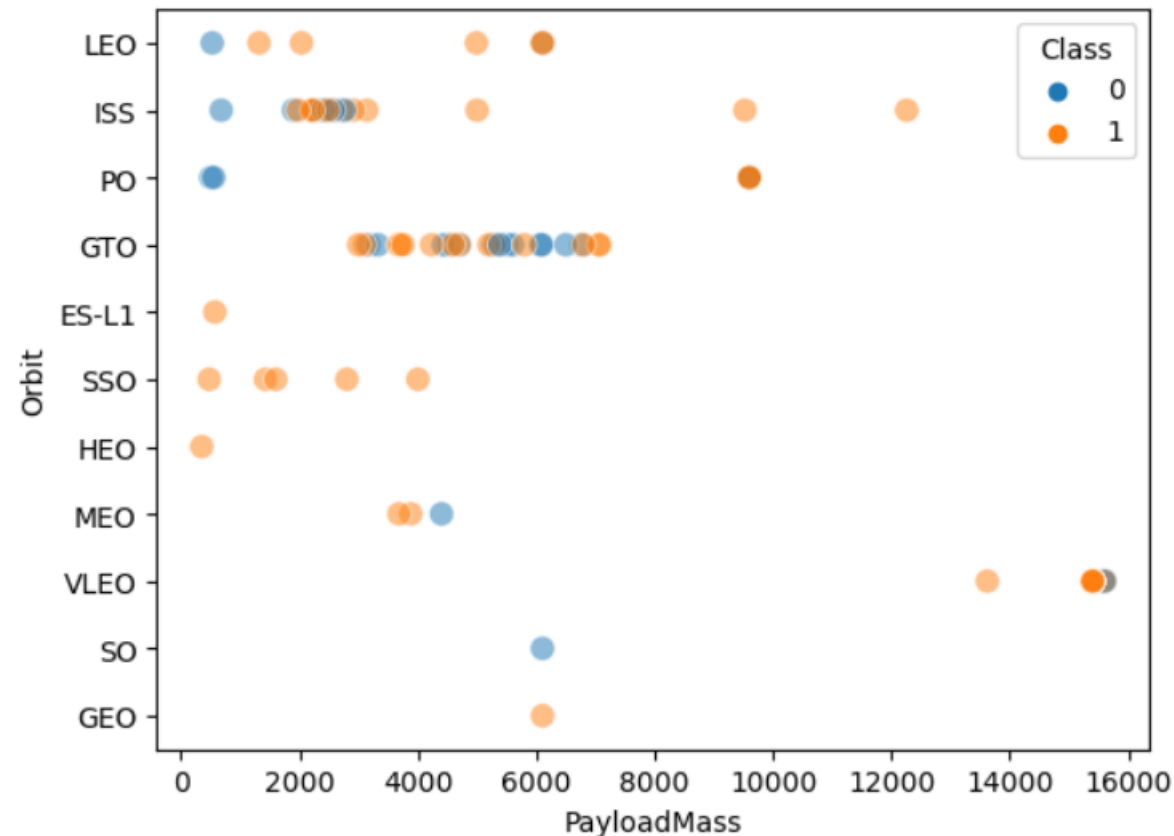
- Explanation
- Orbits with 100% success rate:
  - ES-L1, GEO, HEO, SSO
- Orbits with 0% success rate:
  - SO
- Orbits with success rate between 50% and 85%:
  - GTO, ISS, LEO, MEO, PO

# Flight Number vs. Orbit Type



- Explanation:
  - In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

# Payload vs. Orbit Type

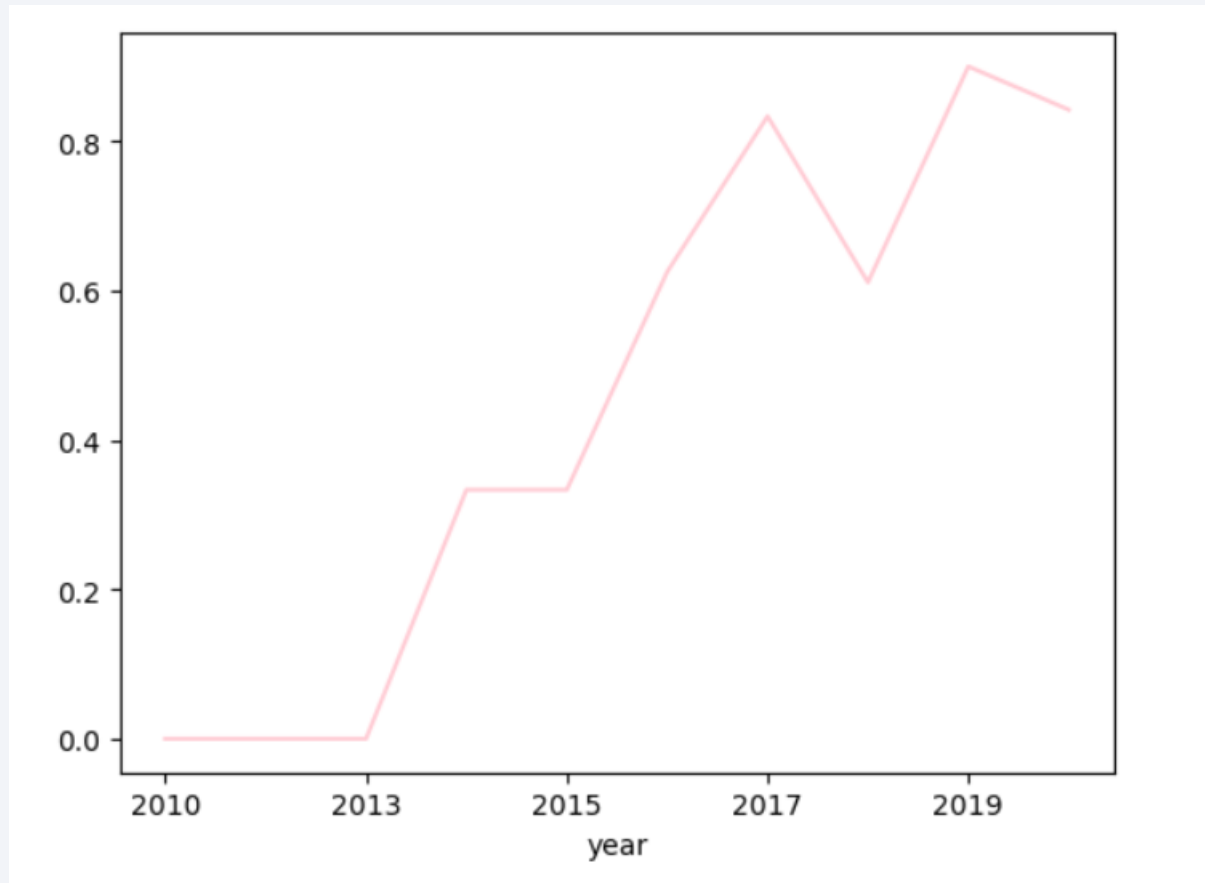


- Explanation:

- Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.

# Launch Success Yearly Trend

---



- Explanation

The success rate since 2013 kept increasing till 2020.



# All Launch Site Names

---

```
In [8]: %sql select distinct(Launch_Site) from SPACEXTBL
```

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

```
Out[8]: Launch_Site
```

```
CCAFS LC-40
```

```
VAFB SLC-4E
```

```
KSC LC-39A
```

```
CCAFS SLC-40
```

- Explanation:
  - Displaying the names of the unique launch sites in the space mission.

# Launch Site Names Begin with 'CCA'

```
In [9]: %sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5
```

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

```
Out[9]:
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS_KG	Orbit	Customer	Mission_Outcome	Landing_Outc
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

# Total Payload Mass

---

```
In [10]: %sql select sum(PAYLOAD_MASS_KG_) from SPACEXTBL where customer LIKE 'NASA (CRS)'  
* sqlite:///my_data1.db  
Done.  
Out[10]: sum(PAYLOAD_MASS_KG_)  
45596.0
```

- Explanation:
  - Displaying the total payload mass carried by boosters launched by NASA(CRS).

# 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.
```

<u>avg(PAYLOAD_MASS_KG_)</u>
2534.6666666666665

- Explanation:
  - Displaying average payload mass carried by booster version F9 v1.1.

# First Successful Ground Landing Date

---

```
%sql select min(Date) from SPACEXTBL where Landing_Outcome='Success'
```

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

```
Done.
```

<b>min(Date)</b>
------------------

01/07/2020
------------

- Explanation:
  - Listing the date when the first successful landing outcome in ground pad was achieved.



# Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select Booster_Version from SPACEXTBL where Landing_Outcome='Success' and PAYLOAD_MASS__KG_ between 4000 and 6000
```

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

**Booster\_Version**

F9 B5 B1046.2

F9 B5 B1047.2

F9 B5 B1046.3

F9 B5 B1048.3

F9 B5 B1051.2

F9 B5B1060.1

F9 B5 B1058.2

F9 B5B1062.1

- Explanation:

- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.

# Total Number of Successful and Failure Mission Outcomes

---

```
%sql select count(*) as success from SPACEXTBL where Landing_Outcome like 'Success%';
```

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

```
Done.
```

success
---------

61
----

```
%sql select count(*) as failure from SPACEXTBL where Landing_Outcome not like 'Success%';
```

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

```
one.
```

failure
---------

40
----

- Explanation:
  - Listing the total number of successful and failure mission outcomes.

# Boosters Carried Maximum Payload

---

```
%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

- Explanation:
  - Listing the names of the booster versions which have carried the maximum payload mass.

# 2015 Launch Records

---

```
%sql select Booster_Version, Launch_Site, substr(Date, 4, 2) as month from SPACEXTBL where Landing_Outcome='Failure (drone st
```

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

```
Done.
```

Booster_Version	Launch_Site	month
F9 v1.1 B1012	CCAFS LC-40	10
F9 v1.1 B1015	CCAFS LC-40	04

- Explanation:

- Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015.

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

```
%sql select Landing_Outcome,count(Landing_Outcome) as count from SPACEXTBL where Date between '04/06/2010 ' and '20/03/2017'
```

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

Landing_Outcome	count
Success	20
No attempt	9
Success (drone ship)	8
Success (ground pad)	7
Failure (drone ship)	3
Failure	3
Failure (parachute)	2
Controlled (ocean)	2
No attempt	1

- Explanation:
  - Ranking the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 in descending order.

A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The background is a deep blue gradient.

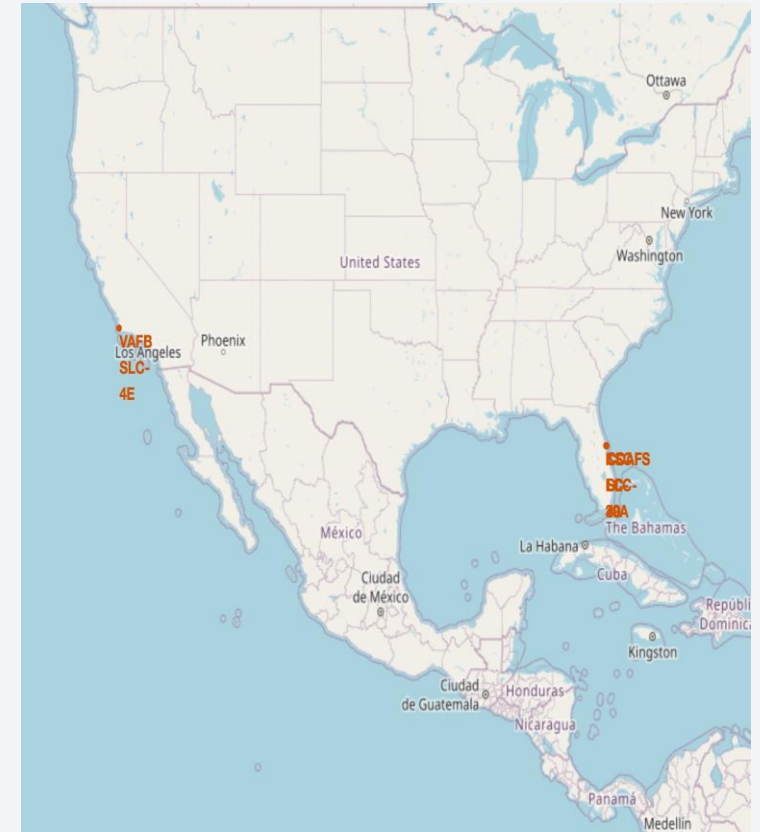
Section 3

# Launch Sites Proximities Analysis

# <Folium Map Screenshot 1>

---

- Explanation:
  - Most of Launch sites are in proximity to the Equator line. The land is moving faster at the equator than any other place on the surface of the Earth. Anything on the surface of the Earth at the equator is already moving at 1670 km/hour. If a ship is launched from the equator it goes up into space, and it is also moving around the Earth at the same speed it was moving before launching. This is because of inertia. This speed will help the spacecraft keep up a good enough speed to stay in orbit.
  - All launch sites are in very close proximity to the coast, while launching rockets towards the ocean it minimises the risk of having any debris dropping or exploding near people.





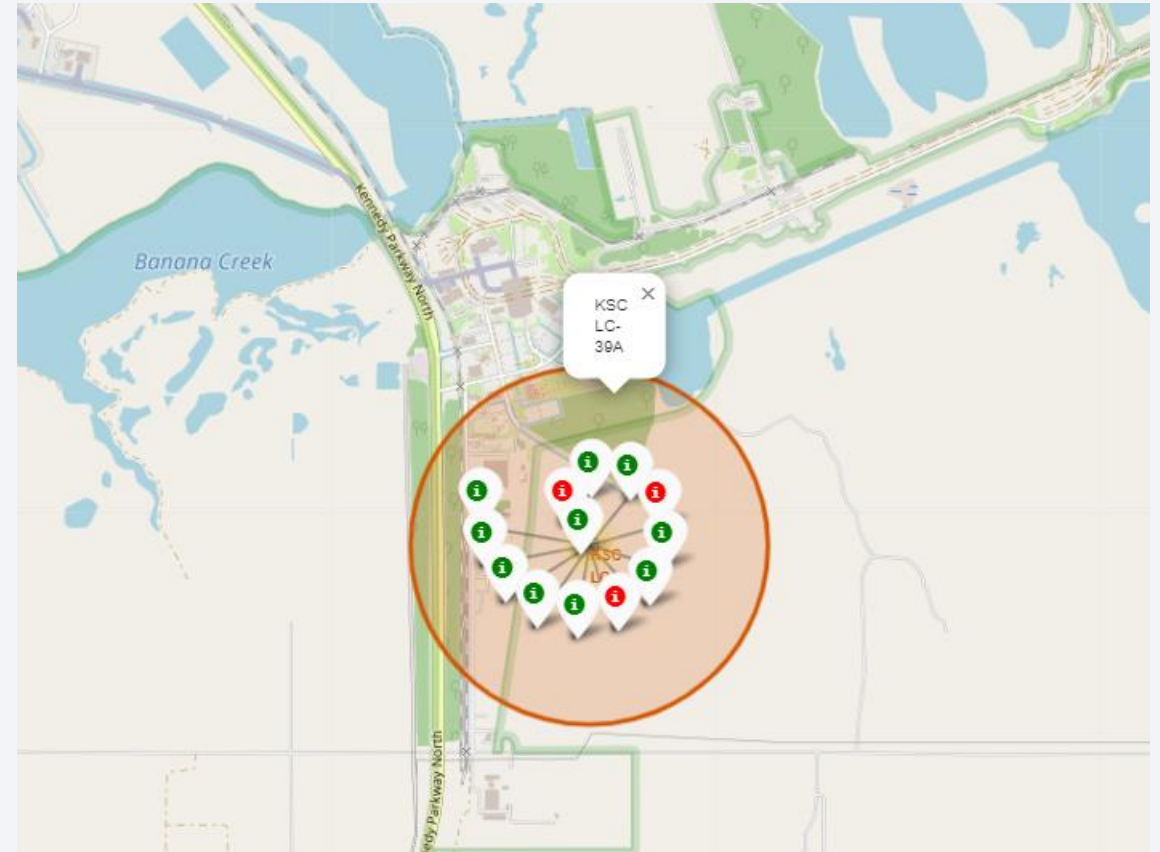
## <Folium Map Screenshot 2>

- Explanation

From the colour-labeled markers we should be able to easily identify which launch sites have relatively high success rates.

- Green Marker = Successful Launch
- Red Marker = Failed Launch

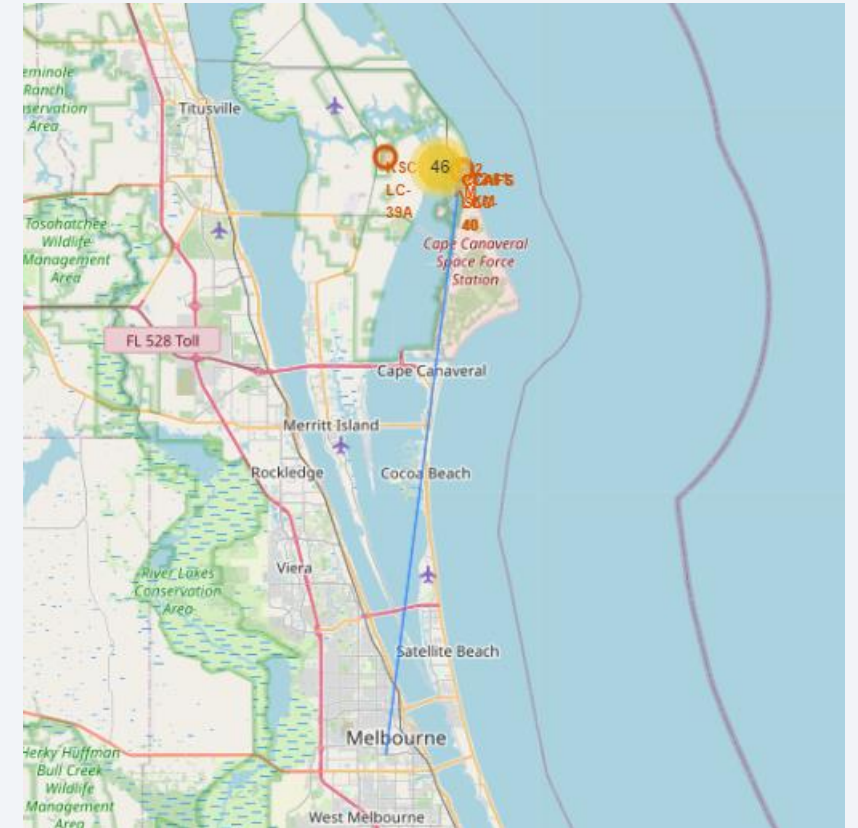
- Launch Site KSC LC-39A has a very high Success Rate.





# <Folium Map Screenshot 3>

- Explanation:
  - From the visual analysis of the launch site CCAFS SLC-40 we can clearly see that it is:
    - relative close to railway (15.23 km)
    - relative close to highway (20.28 km)
    - relative close to coastline (14.99 km)
  - Also the launch site CCAFS SLC-40 is relative close to its closest city Titusville (16.32 km).
  - Failed rocket with its high speed can cover distances like 15-20 km in few seconds. It could be potentially dangerous to populated areas.



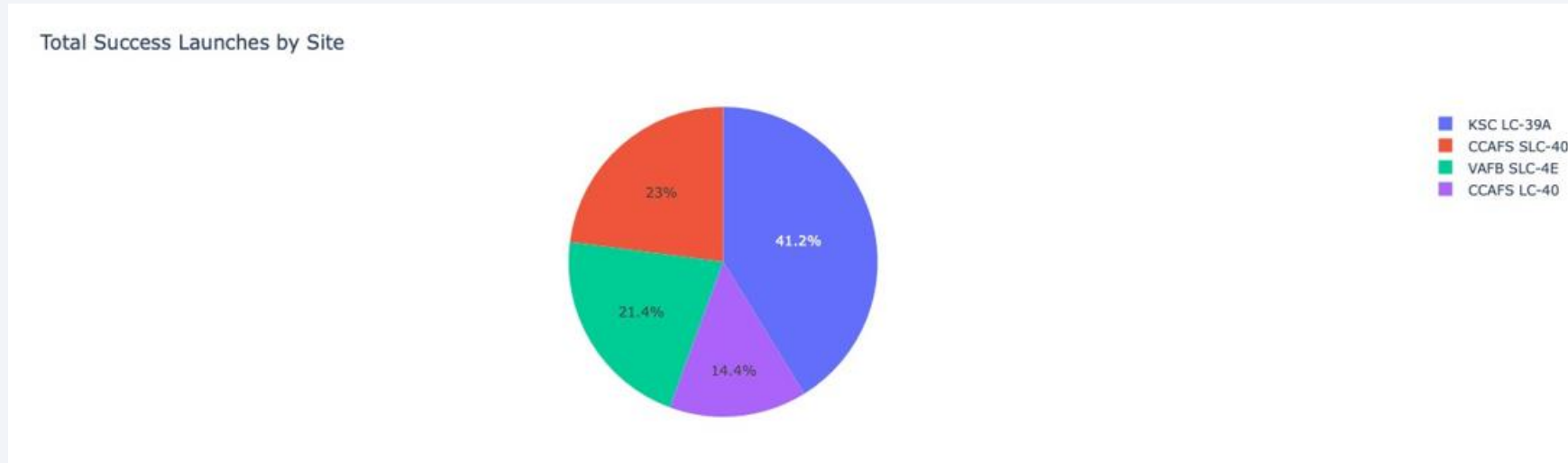


Section 4

# Build a Dashboard with Plotly Dash

# <Dashboard Screenshot 1>

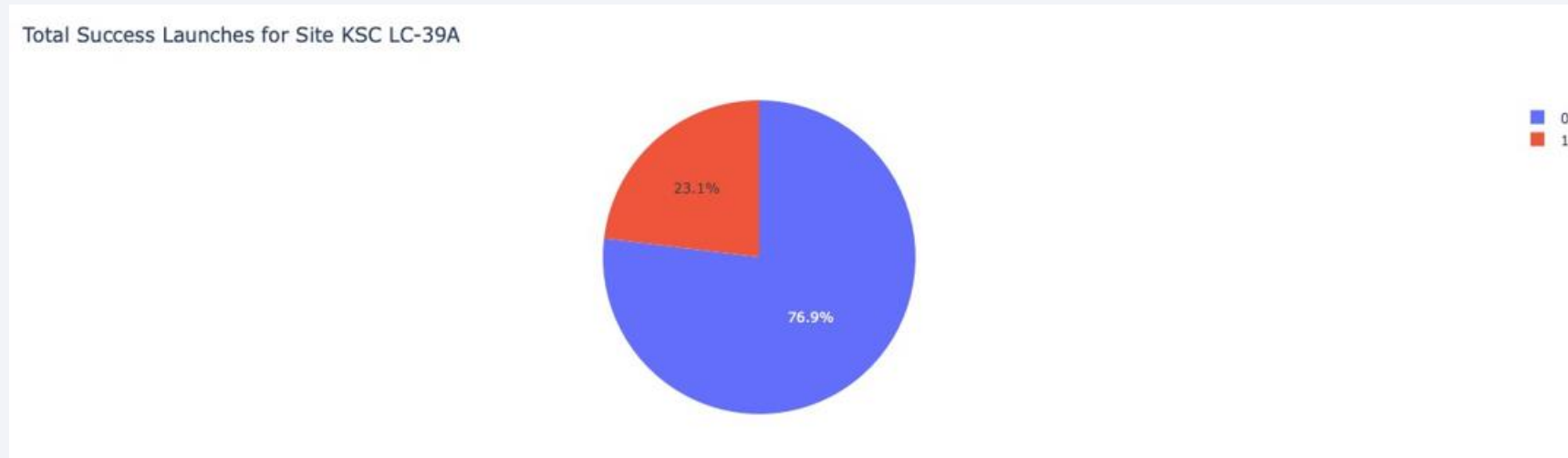
---



- Explanation:
  - The chart clearly shows that from all the sites, KSC LC-39A has the most successful launches.

## <Dashboard Screenshot 2>

---



- Explanation:
  - KSC LC-39A has the highest launch success rate (76.9%) with 10 successful and only 3 failed landings.



# <Dashboard Screenshot 3>

- Explanation

The charts show that payloads between 2000 and 5500 kg have the highest success rate.



Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

---

- Explanation:
  - Based on the scores of the Test Set, we can not confirm which method performs best.
  - Same Test Set scores may be due to the small test sample size (18 samples). Therefore, we tested all methods based on the whole Dataset.
  - The scores of the whole Dataset confirm that the best model is the Decision Tree Model. This model has not only higher scores, but also the highest accuracy.

Scores and Accuracy of the Test Set

	LogReg	SVM	Tree	KNN
<b>Jaccard_Score</b>	0.800000	0.800000	0.800000	0.800000
<b>F1_Score</b>	0.888889	0.888889	0.888889	0.888889
<b>Accuracy</b>	0.833333	0.833333	0.833333	0.833333

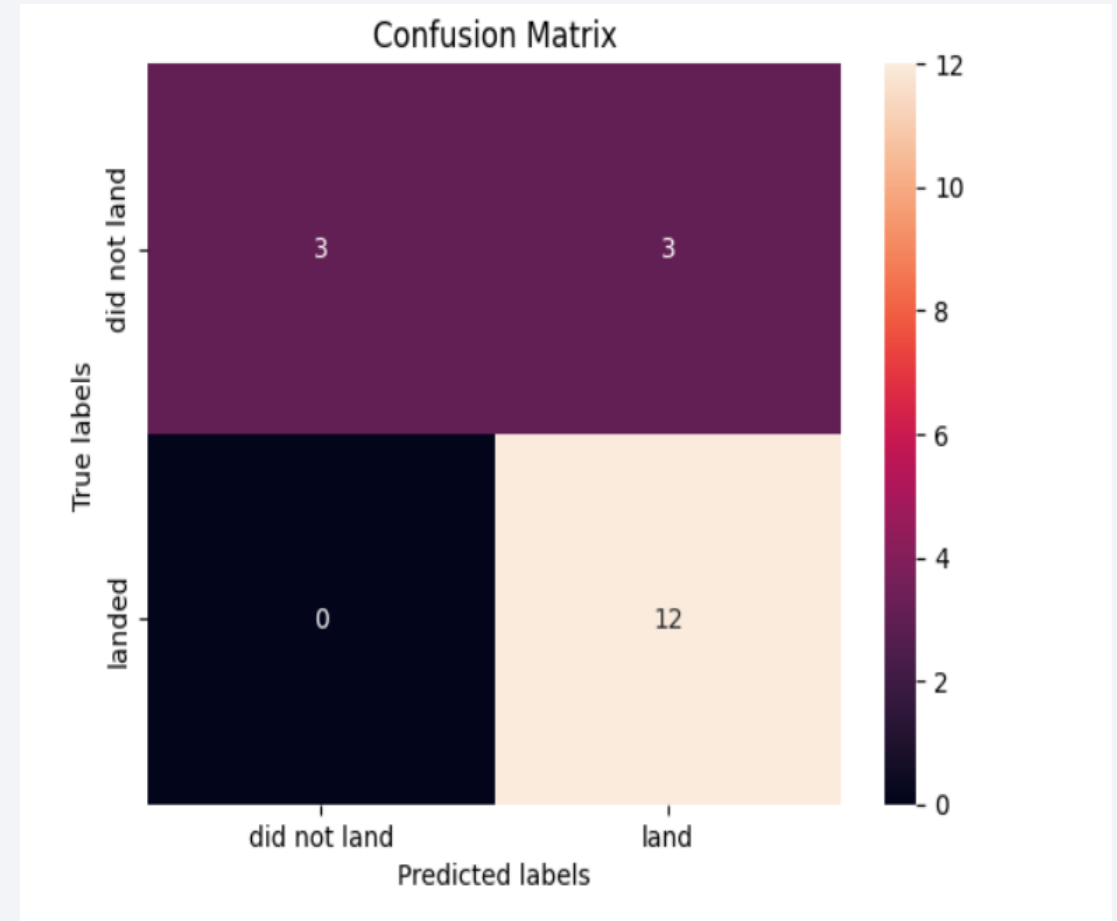
Scores and Accuracy of the Entire Data Set

	LogReg	SVM	Tree	KNN
<b>Jaccard_Score</b>	0.833333	0.845070	0.882353	0.819444
<b>F1_Score</b>	0.909091	0.916031	0.937500	0.900763
<b>Accuracy</b>	0.866667	0.877778	0.911111	0.855556

# Confusion Matrix

- Explanation:

Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the major problem is false positives.





# Conclusions

---

- Decision Tree Model is the best algorithm for this dataset.
- Launches with a low payload mass show better results than launches with a larger payload mass.
- Most of launch sites are in proximity to the Equator line and all the sites are in very close proximity to the coast.
- The success rate of launches increases over the years.
- KSC LC-39A has the highest success rate of the launches from all the sites.
- Orbits ES-L1, GEO, HEO and SSO have 100% success rate.

# Appendix

---

- Special Thanks to:
  - [Instructors](#)
  - [Coursera](#)
  - [IBM](#)

Thank you!

