

# Winning Space Race with Data Science

Space Y

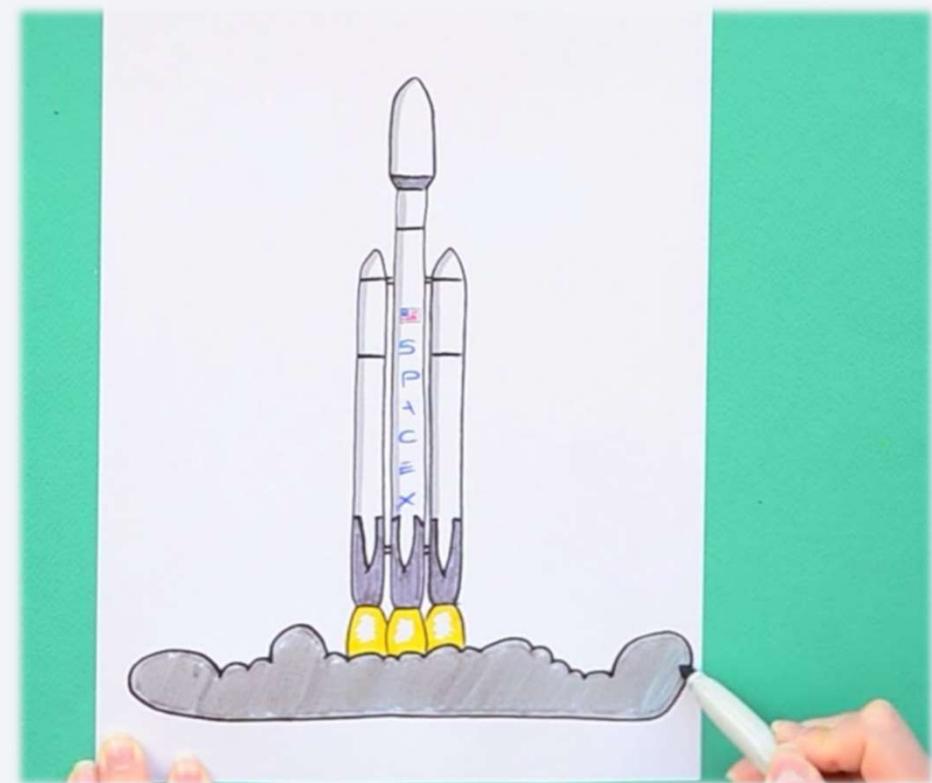
Data Scientist  
Amos LIM TY  
July 2022



# Outline

---

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



# Executive Summary

---

- **Summary Of Methodologies**
  - Data collection
  - Basic EDA and Data wrangling
  - EDA with data visualization
  - EDA with SQL
  - Building an interactive map with Folium
  - Building a Dashboard with Plotly Dash
  - Predictive analysis (Classification)
- **Summary Of All Results**
  - EDA results
  - Interactive analytics
  - Predictive analysis



# Introduction

---

- **Project Background And Context**

- 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, determining if the first stage will land can determine the cost of a launch.
- This information can be used in our company, SpaceY wants to bid against SpaceX for a rocket launch.

- **Problems That Are Worth Solving**

- What factors determine if the rocket will land successfully?
- The interaction amongst various features (e.g. rocket variables) that determine the success rate of a successful landing.
- What operating conditions need to be in place to ensure a successful landing program.

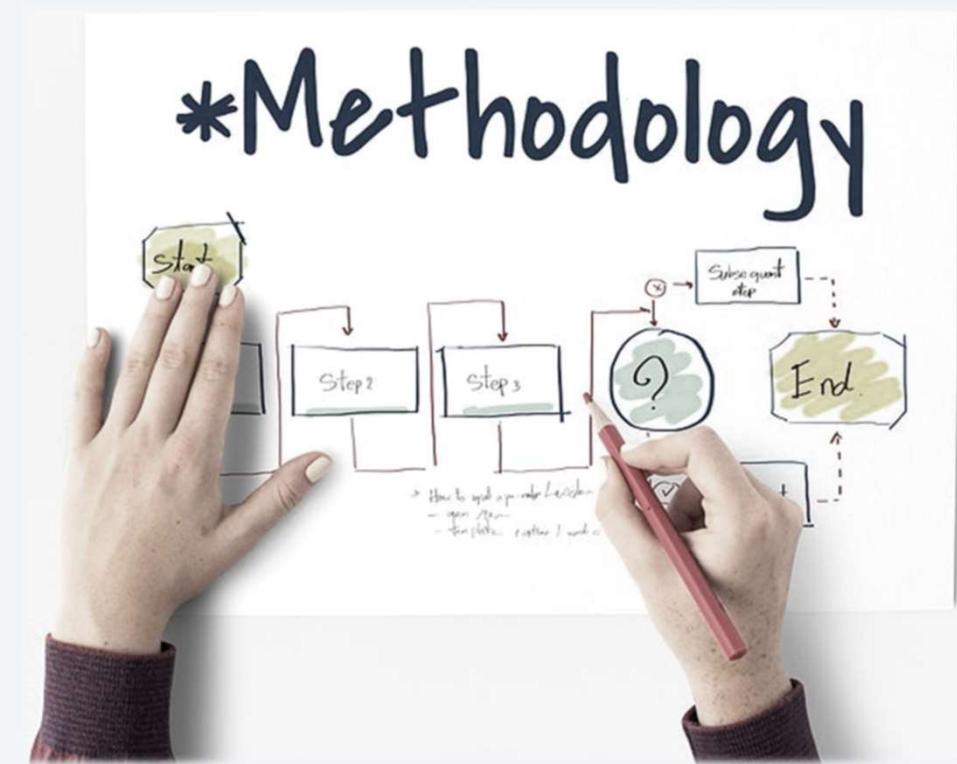


Section 1

# Methodology

# Methodology – Executive Summary

- Data Collection Methodology:
  - SpaceX Rest API (received as .json file)
  - Web Scrapping from Wikipedia
- Perform Basic Exploratory Data Analysis (EDA) & Data Wrangling
  - Data cleaning of null values and irrelevant
  - Create training labels
- Perform Exploratory Data Analysis (EDA) Using Visualization And SQL
  - One Hot Encoding applied to categorical features
- Perform Interactive Visual Analytics Using Folium And Plotly Dash
- Perform Predictive Analysis Using Classification Models
  - Built various classification models (e.g. LR, KNN, SVM, DT)
  - Evaluated for the best classification model

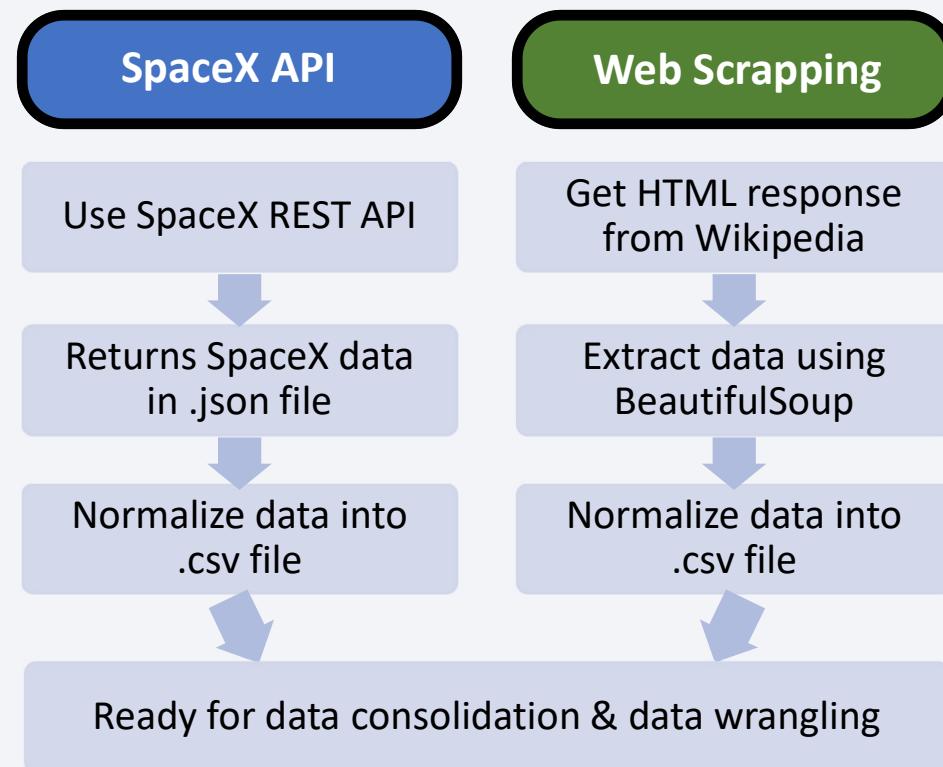


# Data Collection Overview



## Data Collection Methodology:

- SpaceX REST API
- Web Scrapping



# Data Collection – SpaceX API

## Key Phrases And Flowcharts:

1. Get a response from API
2. Convert response to a .json file  
(use 'static response object' for more consistency in this project instead)
3. Extract relevant data with custom functions
4. Create dictionary & dataframe  
(for Falcon 9 only)
5. Deal with missing values
6. Export to a .csv file

```
[29]: data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Click this icon for my notebook (API)



[https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/1\\_jupyter-labs-spacex-data-collection-api%20\(wk1-%20amos\)%20-%20v2.ipynb](https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/1_jupyter-labs-spacex-data-collection-api%20(wk1-%20amos)%20-%20v2.ipynb)

```
[6]: spacex_url="https://api.spacexdata.com/v4/launches/past"
[7]: response = requests.get(spacex_url)

[9]: static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/APIs/SpaceXAPI.json'
static_response = requests.get(static_json_url)

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

[21]: launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude':Longitude,
'Latitude':Latitude}

[22]: # Create a data from launch_dict
data = pd.DataFrame(launch_dict)

[24]: # Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = data[data.BoosterVersion == 'Falcon 9']
data_falcon9

[27]: # Calculate the mean value of PayloadMass column
Mean_PayloadMass = data_falcon9.PayloadMass.mean()
# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'] = data_falcon9['PayloadMass'].replace(np.nan, Mean_PayloadMass)

[16]: # Call getBoosterVersion
getBoosterVersion(data)

[18]: # Call getLaunchSite
getLaunchSite(data)

[19]: # Call getPayloadData
getPayloadData(data)

[20]: # Call getCoreData
getCoreData(data)
```

# Data Collection – Web Scrapping

## Key Phrases And Flowcharts:

1. Get a response from HTML and create BeautifulSoup Object
2. Find tables and extract column names
3. Create dictionary
4. Extract relevant data & append data with custom functions  
(Highlight is shown here. For full code refer to my notebook)
5. Convert to dataframe
6. Export to a .csv file

Click this icon for my notebook (web scrapping)



[https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/2\\_jupyter-labs-spacex-web-scraping%20\(wk1-%20amos\)%20-%20v2.ipynb](https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/2_jupyter-labs-spacex-web-scraping%20(wk1-%20amos)%20-%20v2.ipynb)

```
[5]: data = requests.get(static_url).text
# assign the response to a object
[6]: soup = BeautifulSoup(data, "html.parser")
[8]: html_tables = soup.find_all('table')
[10]: column_names = []
for row in first_launch_table.find_all('th'):
    name = extract_column_from_header(row)
    if name != None and len(name) > 0:
        column_names.append(name)
[12]: launch_dict = dict.fromkeys(column_names)
del launch_dict['Date and time ( )']
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
launch_dict['Version Booster'] = []
launch_dict['Booster landing'] = []
launch_dict['Date'] = []
launch_dict['Time'] = []
[13]: extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table', "wikitable plainrowheaders collapsible")):
    #get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to Launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
            else:
                flag=False
        #get table element
        row=rows.find_all('td')
[15]: df=pd.DataFrame(launch_dict)
[16]: df.to_csv('spacex_web_scraped.csv', index=False)
```

# Basic EDA and Data Wrangling

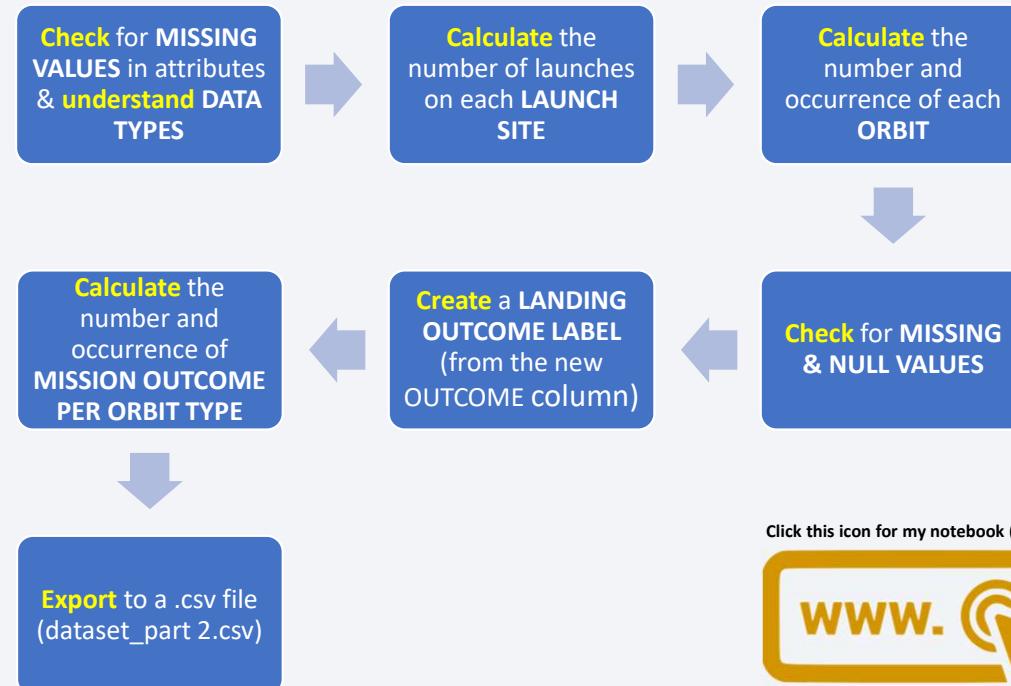


## Objectives of Data Wrangling:

- Exploratory Data Analysis
- Determine Training Labels

**Data Source:** Exported .csv file from SpaceX API  
(e.g. dataset\_part\_1.csv)

## Key Phrases And Flowcharts:



Click this icon for my notebook (data wrangling)



[https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/3\\_jupyter-labs-spacex-Data%20wrangling%20%26%20EDA%20\(wk1-%20amos\)%20-%20v3.ipynb](https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/3_jupyter-labs-spacex-Data%20wrangling%20%26%20EDA%20(wk1-%20amos)%20-%20v3.ipynb)

# EDA with Data Visualization



## Objectives of EDA with Data Visualization:

- Exploratory Data Analysis
- Preparing for Feature Engineering

Click this icon for my notebook (data visualization)



[https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/5\\_jupyter-labs-spacex-eda-dataviz%20\(wk2-%20amos\)\\_v2.ipynb](https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/5_jupyter-labs-spacex-eda-dataviz%20(wk2-%20amos)_v2.ipynb)



### Scatter Graphs Plotted:

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload vs. Launch Site
- Flight Number vs. Orbit
- Payload vs. Orbit

Scatter graph shows the relationship (aka correlation) between two variables.



### Bar Graph Plotted:

- Orbit vs. Class

Bar graph helps to compare sets of data between different category groups at a glance.



### Line Graph Plotted:

- Success Rate vs. Year

Line graph shows data variables trends and helps prediction about future data not yet recorded

Data Source: Exported and updated .csv file from SpaceX API (e.g. dataset\_part\_2.csv)

# EDA with SQL

---



Click this icon for my notebook (SQL)



[https://github.com/amoslimty/Apple-Data-Science-Capstone-Project-SpaceX/blob/main/4\\_jupyter-labs-spacesx-eda-sql%20\(wk2-%20amos\)\\_v2.ipynb](https://github.com/amoslimty/Apple-Data-Science-Capstone-Project-SpaceX/blob/main/4_jupyter-labs-spacesx-eda-sql%20(wk2-%20amos)_v2.ipynb)

Data Source: Provided SpaceX Dataset in .csv format (extracted from Web scrapping method)

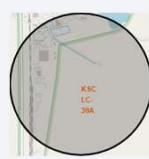
## 10 SQL Queries include:

1. Display the names of the **unique launch sites** in the space mission
2. Display 5 records where **launch sites** begin with the string 'CCA'
3. Display the **total payload mass** carried by boosters launched by **NASA (CRS)**
4. Display **average payload mass** carried by booster version **F9 v1.1**
5. List the **date** where the **first successful landing outcome** in **ground pad** was achieved.
6. List the **names of the boosters** which have **success in drone ship** have **payload mass** greater than 4000 but less than 6000
7. List the **total number of successful and failure mission outcomes**
8. List the **names of the booster\_versions** which have carried the **maximum payload mass**.
9. List the **records** which will display the **month names, failure landing\_outcomes** in **drone ship, booster versions, launch\_site** for the **months** in **year 2015**
10. **Rank the count of successful landing\_outcomes** between the date 4<sup>th</sup> Jun 2010 and 20<sup>th</sup> Mar 2017 in descending order.

# Build an Interactive Map with Folium



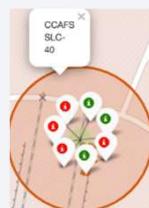
Data Source: Provided SpaceX Launch Geo Dataset in .csv format



## Mark 'Launch Sites' with Circle and Label:

- NASA Johnson Space Center at Houston
- SpaceX Launch Sites: CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, and VAFB SLC-4E.

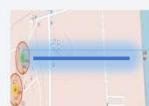
Ease of locating launch sites



## Mark Color-labeled Markers in Marker Clusters (i.e., SpaceX Launch Sites):

- Red marker – Failed Launch
- Green marker – Successful Launch

Identify the Success rate of each launch sites (Geographical Patterns)

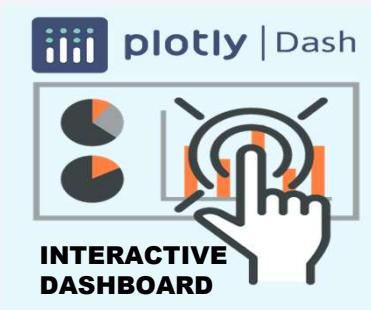


## Mark Polylines between a Launch Site and to its Proximities:

- Selected coastline point
- Closest city
- Closest railway
- Closest highway

Calculate the distances between a launch site to its proximities for logistics, safety, etc.

# Build a Dashboard with Plotly Dash



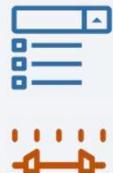
Click this icon for my notebook Plotly Dash App



[https://github.com/amoslimty/Apple-Data-Science-Capstone-Project-SpaceX/blob/main/7\\_spacex\\_dash\\_app%20\(wk3-%20amos\).py](https://github.com/amoslimty/Apple-Data-Science-Capstone-Project-SpaceX/blob/main/7_spacex_dash_app%20(wk3-%20amos).py)

Data Source: Provided SpaceX Launch Dash Dataset in .csv format

## Interactive Inputs:



- **Dropdown Menu** – To select the launch site
- **Range Slider** – To select payload range

- Keep the number of inputs to a minimum.
- Two selected input choices are intuitive to most users.

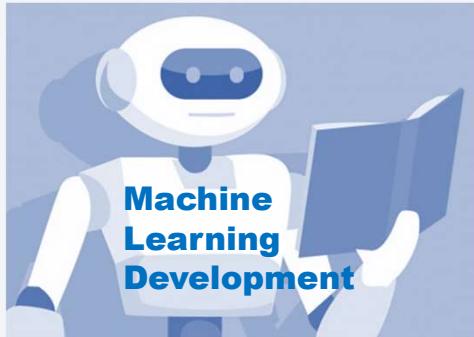
## Instant Rendering of Outputs:



- **Pie Chart** – To display successful and failed launch counts for all sites
- **Scatter Graph** – To display the correlation between payload and launch success

- Two selected instant output choices also are relatable to users.
- The instant changes in the graphs allow users to understand the cause-and-effect.

# Predictive Analysis (Classification)



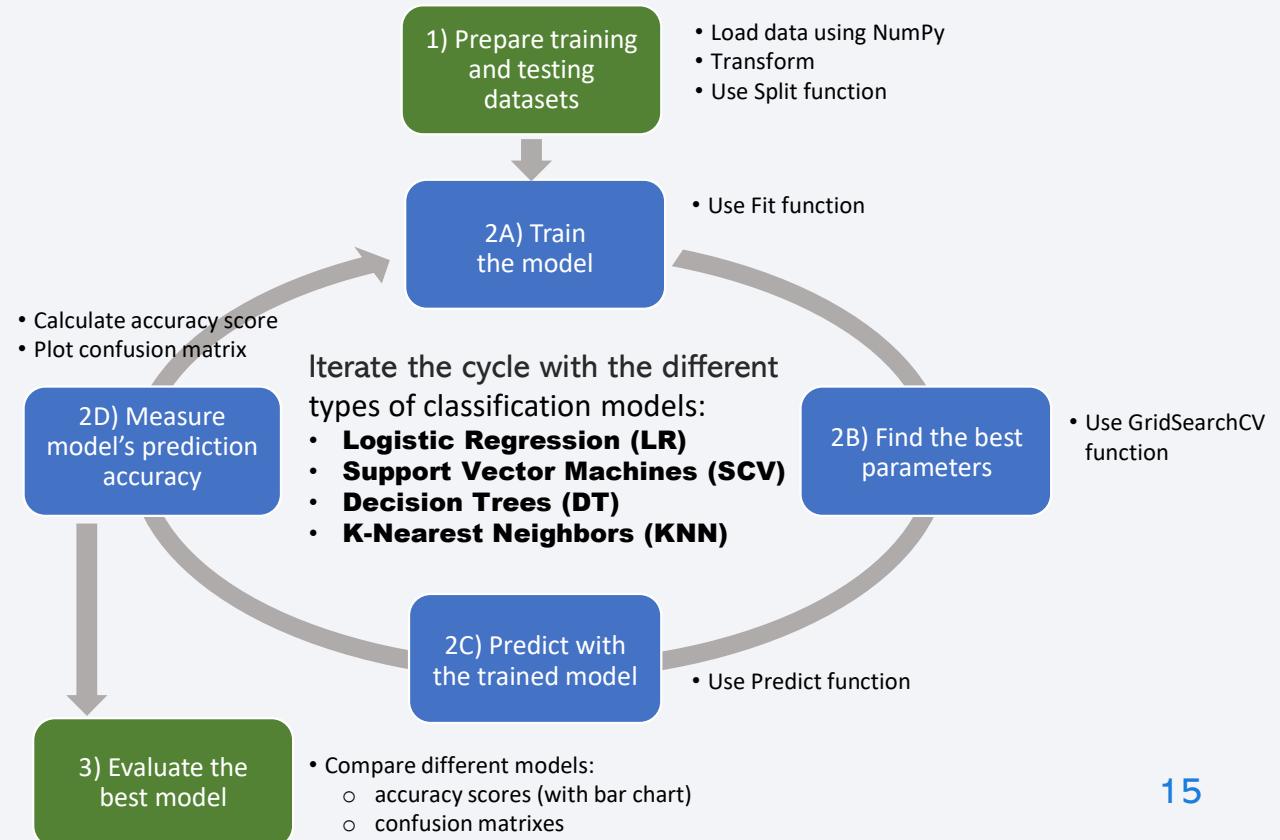
Click this icon for my notebook (Folium)



[https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/8\\_jupyter-labs-spaces-machine%20learning%20prediction\(wk4-%20amos\) v3.ipynb](https://github.com/amoslimty/Applied-Data-Science-Capstone-Project-SpaceX/blob/main/8_jupyter-labs-spaces-machine%20learning%20prediction(wk4-%20amos) v3.ipynb)

**Data Source:** Exported and updated .csv file from SpaceX API (e.g. dataset\_part\_2.csv and dataset\_part\_3.csv)

# Classification Model Development Process:

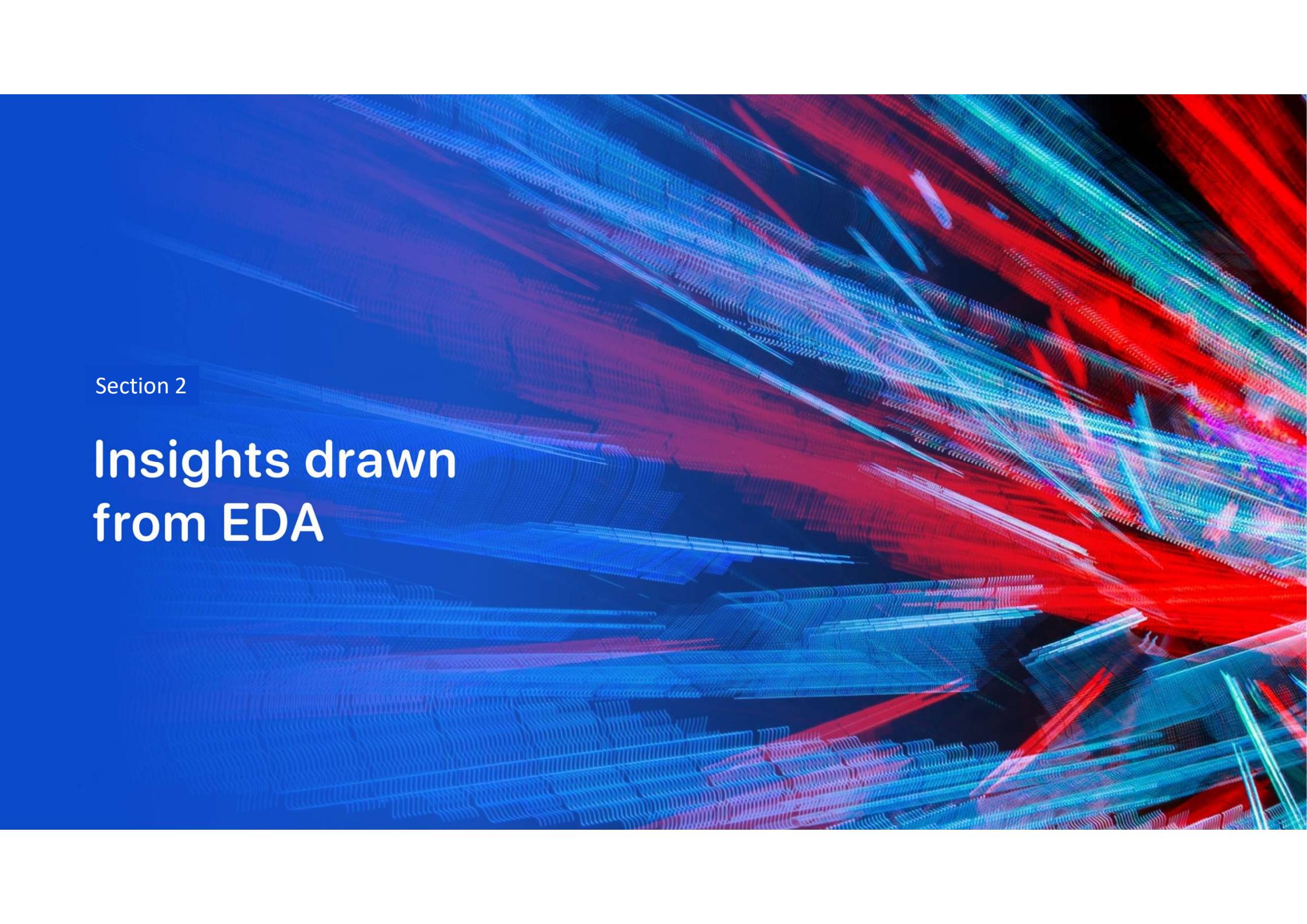


# Key Results

---

- **Exploratory Data Analysis (EDA) Results**
  - Orbit types of **ES-L1**, **GEO**, **HEO**, and **SSO** are among the highest success rate.
  - The **larger the flight volume** at a launch site, the **greater the success rate** at the launch site over time. Space X has mastered the launch business and reached sustainable success from 2013 onwards.
- **Interactive Analytics Results**
  - **KSC LC-39A** site in **Florida** has the best success rate.
  - **Lighter payloads range** (particularly between 2,000Kg to 4,000Kg) has a better success rate.
- **Predictive Analysis Results**
  - The **K-Nearest Neighbour (KNN)** classification model is the best prediction algorithm for this assignment.



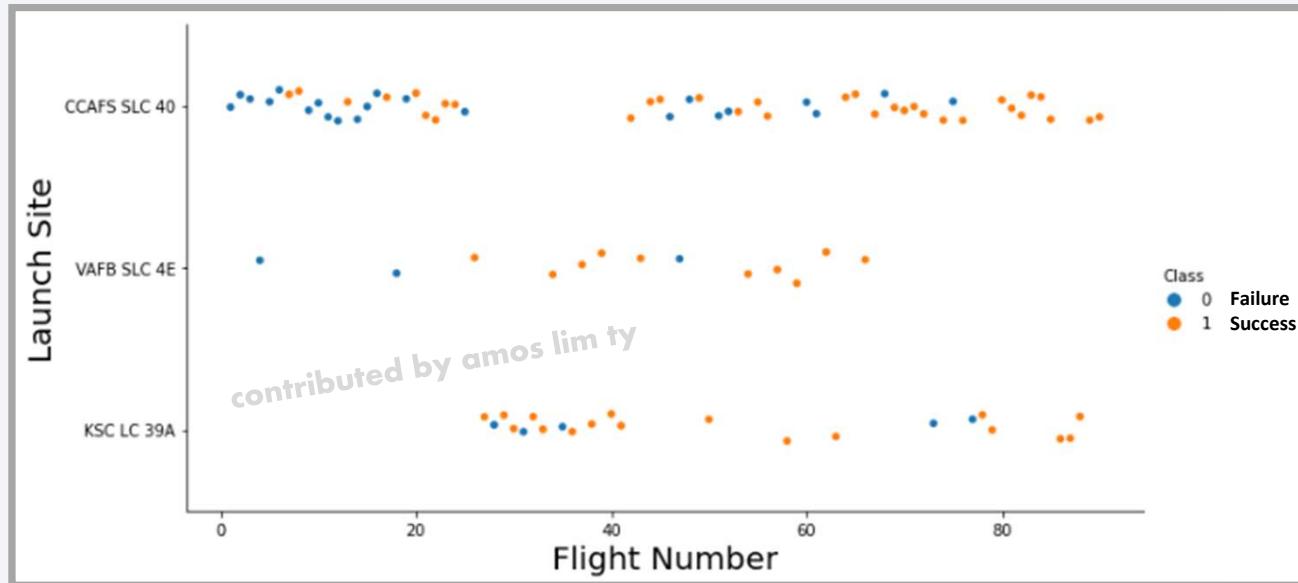
The background of the slide features a complex, abstract pattern of glowing lines. The lines are primarily blue and red, with some green and white highlights. They are arranged in a way that suggests depth and motion, resembling a 3D space filled with data or energy. The lines are thin and have a slight glow, creating a futuristic and dynamic visual effect.

Section 2

## Insights drawn from EDA



# Flight Number vs. Launch Site

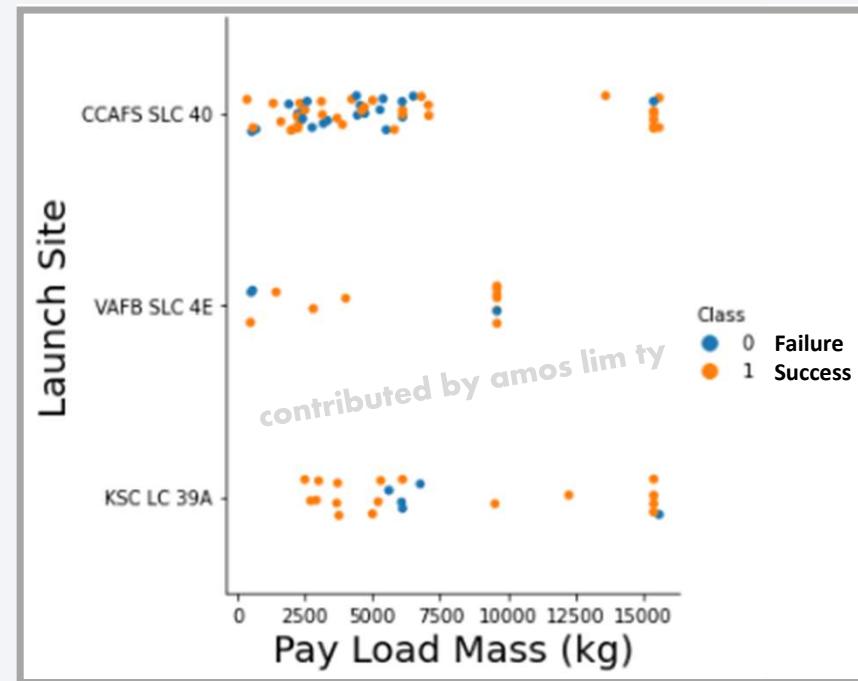


- From the scatter plot, as the flight volume increased gradually at any launch site, the success rate also increased at the launch site.
- Flight volume (aka launches) from the CCAFS SLC 40 site was significantly higher than other sites.



# Payload vs. Launch Site

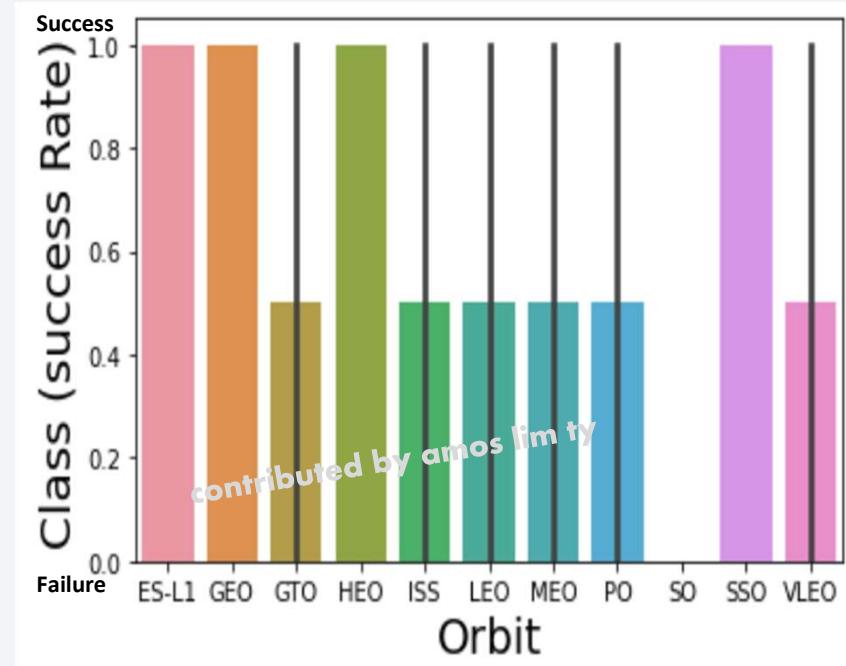
- Most launches with lower payloads (<7,500 kg) were from CCAFS SLC 40.
- The higher the payload mass, the higher the success rate for all launch sites. Particularly for payload mass >10,000 kg.
- For payload mass <7,500 kg, KSC LC39A site had better success rate than CCAFS SLC 40 site but KSC LC39A site had fewer launches than CCAFS SLC 40 site.





# Success Rate vs. Orbit Type

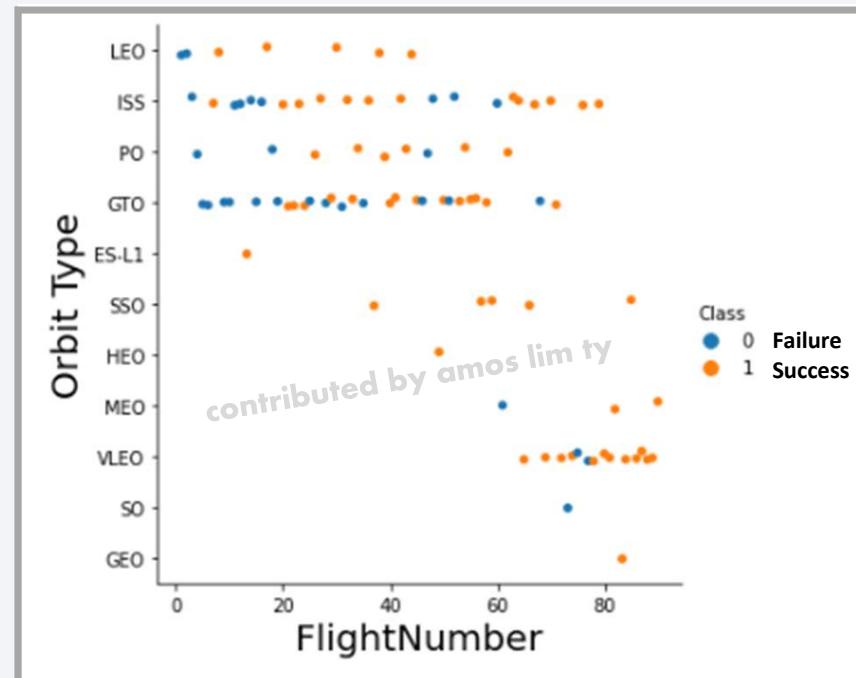
- From the bar chart, the orbit types of ES-L1, GEO, HEO, SSO are among the highest success rate.





# Flight Number vs. Orbit Type

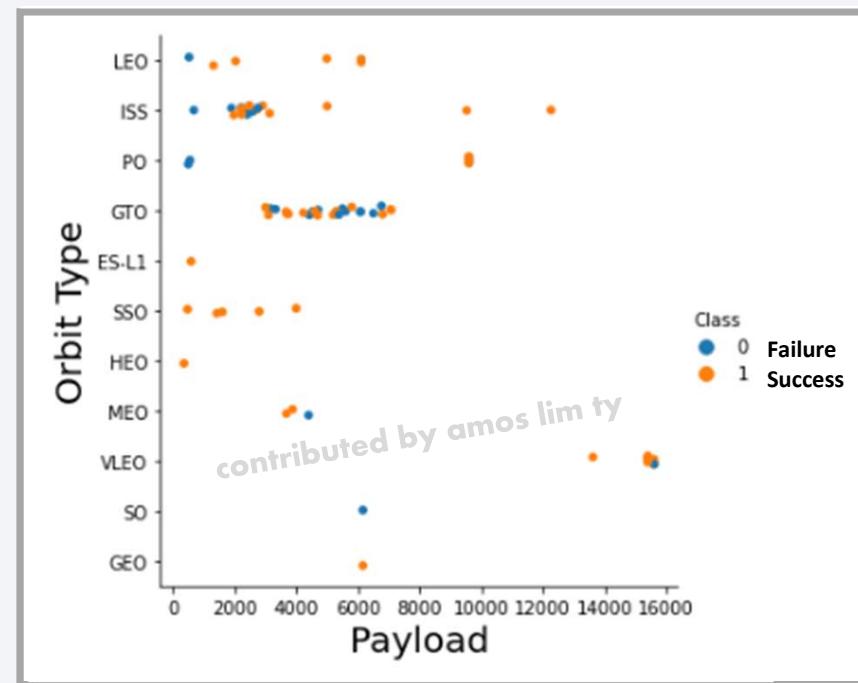
- There are more successes, as the number of flights increases over time for LEO orbit.
- There seems to be no correlation between flight number and success rate for GTO orbit.
- There is a trend of more VLEO launches in recent times compared to GTO, LEO, ISS, & PO orbits launches.





# Payload vs. Orbit Type

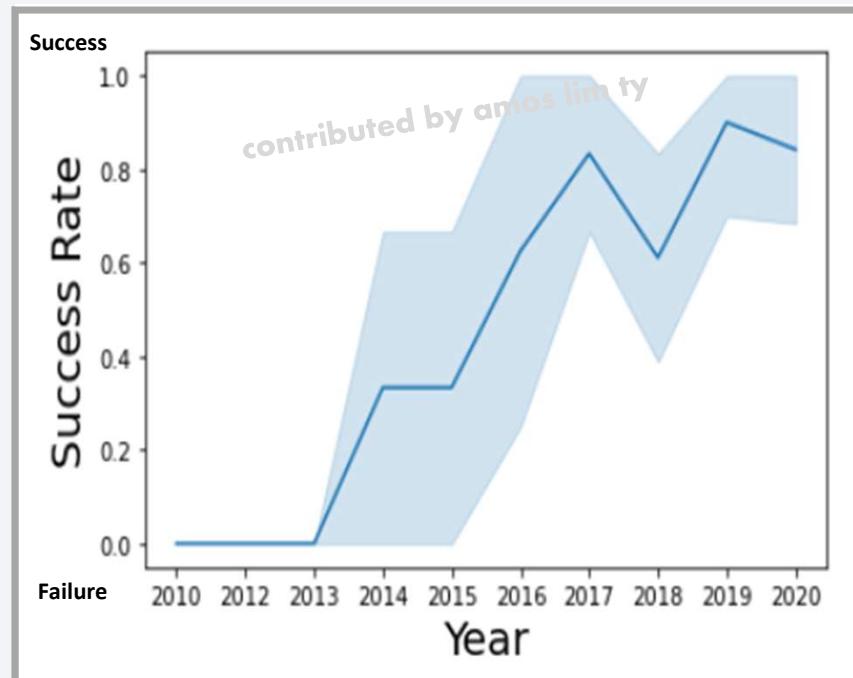
- There is a high concentration of launches but no correlation of success rate.
  - **ISS orbit:** concentrated around 2,000 kg to 4,000 kg payload
  - **GTO orbit:** concentration around 3,000 kg to 7,000 kg payload
- For SSO orbit, regardless of the range of payload, it achieved 100% success so far.
- There is no failure after crossing a certain threshold of payload:
  - **LEO orbit:** > 1,000 kg payload
  - **ISS orbit:** > 4,000 kg payload
  - **PO orbit:** > 10,000 kg payload





# Launch Success Yearly Trend

- From the line chart, it is observed that the success rate has increased significantly since 2013 and has stabilized since 2017, potentially due to advances in technology and lessons learned from failures.





# All Launch Site Names

## SQL Query

```
[6]: %sql select distinct(LAUNCH_SITE) from SPACEXTBL
```



```
[6]: launch_site
```

---

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

### QUERY EXPLANATION

Using the word DISTINCT in the query means that it will only show Unique values in the Launch\_Site column from tblSpaceX

Unique Launch Site	Location	Interesting Facts
Cape Canaveral Space Launch Complex 40 (CCAF SLC-40) • Previously Launch Complex 40 (LC-40)	Florida, U.S. 28.562106°N 80.577180°W	After 2007, the US Air Force leased the complex to SpaceX to launch the Falcon 9 rocket. As of May 2022, there have been 90 launches of the Falcon 9 from the complex.
Kennedy Space Center Launch Complex 39A (KSC LC-39A)	Florida, U.S. 28°36'30.2"N 80°36'15.6"W	In 2013, NASA announced that SpaceX as the new commercial tenant for 20-year exclusive lease. SpaceX will shift most of SpaceX's NASA launches to LC-39A, including commercial cargo and crew missions to the International Space Station.
Vandenberg Space Launch Complex 4 (VAFB SLC-4E)	California, U.S. 34.633°N 120.613°W	SpaceX refurbished SLC-4E for Falcon 9 launches in a 24-month process that began in early 2011. By late 2012, SpaceX scheduled the initial launch from the Vandenberg pad would be in 2013, with the larger variant Falcon 9 v1.1.

Source: [www.wikipedia.org](http://www.wikipedia.org)



# Launch Site Names Begin with 'CCA'

## SQL Query

```
[7]: %sql select * from SPACEXTBL where LAUNCH_SITE like 'CCA%' limit 5
```



### QUERY EXPLANATION

Used the word LIMIT 5 in the query means that it will only show 5 records from tblSpaceX and LIKE keyword has a wild card with the words 'CCA%' the percentage, in the end, suggests that the Launch\_Site name must start with CCA.

[7]:	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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt	
2012-10-08	00:35:00	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	

The “Cape Canaveral Space Launch Complex 40” site had previously been renamed. However, the site has consistently labelled with the starting code “CCA”. Therefore, this SQL query is useful to specifically search for records from “Cape Canaveral Space Launch Complex 40” site.

Source: [https://en.wikipedia.org/wiki/Cape\\_Canaveral\\_Space\\_Launch\\_Complex\\_40](https://en.wikipedia.org/wiki/Cape_Canaveral_Space_Launch_Complex_40)



# Total Payload Mass (by customer NASA (CRS) )

## SQL Query

```
[8]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where CUSTOMER = 'NASA (CRS)'
```



```
[8]: 1  
-----  
45596
```

### QUERY EXPLANATION

Using the function SUM summates the total in the column PAYLOAD\_MASS\_KG\_. The WHERE clause filters the dataset to only perform calculations on Customer NASA (CRS)

Commercial Resupply Services (CRS) are a series of flights awarded by NASA to deliver cargo and supplies to the International Space Station (ISS) on a commercially operated spacecraft. The first CRS contracts were signed in 2008 and awarded \$1.6 billion to SpaceX for twelve cargo Dragon and \$1.9 billion to Orbital Sciences for eight Cygnus flights, covering deliveries to 2016. The Falcon 9 and Antares rockets were also developed under the CRS program to deliver cargo spacecraft to the ISS.

Source: [https://en.wikipedia.org/wiki/Commercial\\_Resupply\\_Services](https://en.wikipedia.org/wiki/Commercial_Resupply_Services)



# Average Payload Mass by F9 v1.1

## SQL Query

```
[9]: %sql select avg(PAYLOAD_MASS_KG_) from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1'
```



```
[9]: 1
```

```
2928
```

### QUERY EXPLANATION

Using the function AVG works out the average in the column PAYLOAD\_MASS\_KG\_. The WHERE clause filters the dataset to only perform calculations on Booster\_version F9 v1.1

Falcon 9 v1.1 was the second version of SpaceX's Falcon 9 orbital launch vehicle. The rocket was developed in 2011–2013, made its maiden launch in September 2013, and its final flight in January 2016. The Falcon 9 rocket was fully designed, manufactured, and operated by SpaceX. Following the second Commercial Resupply Services (CRS) launch, the initial version Falcon 9 v1.0 was retired from use and replaced by the v1.1 version.

Source: [https://en.wikipedia.org/wiki/Falcon\\_9\\_v1.1](https://en.wikipedia.org/wiki/Falcon_9_v1.1)



# First Successful Ground Landing Date

## SQL Query

```
[10]: %sql select min(DATE) from SPACEXTBL where Landing__Outcome = 'Success (ground pad)'
```



```
[10]:
```

2015-12-22

### QUERY EXPLANATION

Using the function MIN works out the minimum date in the column Date. The WHERE clause filters the dataset to only perform calculations on Landing\_Outcome Success (ground pad)

The Economist ☰ Menu Weekly edition Q Search

Science & technology / Down to Earth

### SpaceX finally achieves its first successful landing

Elon Musk's early Christmas present to himself—and the world—is a reusable rocket



Dec 22nd 2015 Share

Technological achievements are, it seems, a bit like buses. You wait for ever, and then two come along at once. In November, an American company called Blue Origin sent a rocket to the edge of space before landing it back on the pad from which it had lifted off. That was an impressive trick. But it has just been trumped by another such firm, SpaceX, which has done the same thing with part of a rocket destined for orbit—a much harder task.

SpaceX's vehicle, one of its Falcon 9 rockets, was sent on its way from Cape Canaveral in Florida at 0129 GMT on December 22nd. This, in

**Source:**  
<https://www.economist.com/science-and-technology/2015/12/22/spacex-finally-achieves-its-first-successful-landing>

# Successful Drone Ship Landing with Payload between 4000 and 6000



## SQL Query

```
[31]: %sql select distinct(BOOSTER_VERSION) from SPACEXTBL where LANDING__OUTCOME = 'Success (drone ship)'\n    AND PAYLOAD_MASS__KG_ between 4000 and 6000
```



```
[31]: booster_version\n\nP9 FT B1021.2\nP9 FT B1031.2\nF9 FT B1022\nF9 FT B1026
```

## QUERY EXPLANATION

Selecting only Booster Version. The WHERE clause filters the dataset to Landing\_Outcome = Success (drone ship). The AND clause specifies additional filter conditions Payload\_MASS\_KG\_ > 4000 AND Payload\_MASS\_KG\_ < 6000

Of Course I Still Love You (OCISLY) is an **autonomous spaceport dronechip (ASDS)** that is operated out of the Port of Long Beach, California. Of Course I Still Love You was previously based in Florida from 2015 to 2021.

Source: <https://spacexfleet.com/of-course-i-still-love-you/>

JCSAT-2B, known as JCSAT-14 before commissioning, is a geostationary communications satellite operated by SKY Perfect JSAT Group and designed and manufactured by SSL on the SSL 1300 platform. It had a **launch weight of 4,696.2 kg**, a power production capacity of 9 to 9.9 kW at end of life and a 15-year design life.

Source: <https://en.wikipedia.org/wiki/JCSAT-2B>

SES-10 is based on the three axis stabilised Eurostar-3000 satellite bus. It has a **mass of 5,282 kg**, produces 13 kW of power and has a design life of 15 years.

Source: <https://en.wikipedia.org/wiki/SES-10>

Source: [https://en.wikipedia.org/wiki/List\\_of\\_Falcon\\_9\\_first-stage\\_boosters](https://en.wikipedia.org/wiki/List_of_Falcon_9_first-stage_boosters)

S/N	Version	Launch date (UTC) <sup>[5]</sup>	Flight No. <sup>[a]</sup>	Turnaround	Payload <sup>[b]</sup>	Launch	Landing	Status
B1021	FT	8 April 2016	F9-023	—	Dragon C110 (CRS-8) <sup>[26]</sup>	Success (40)	Success (OCISLY)	Retired <sup>[27]</sup> To be displayed at Cape Canaveral <sup>[28]</sup> [dubious – discuss]
		30 March 2017	F9-032 △	356 days	SES-10 <sup>[28]</sup>	Success (39A)	Success (OCISLY) [29][30]	
B1022	FT	6 May 2016	F9-024	—	JCSAT-14	Success (40)	Success (OCISLY)	Retired
B1026	FT	14 August 2016	F9-028	—	JCSAT-16	Success (40)	Success (OCISLY) [37]	Retired <sup>[33]</sup>
B1031	FT	19 February 2017	F9-030	—	Dragon C112 (CRS-10) <sup>[48]</sup>	Success (39A)	Success (LZ-1) <sup>[49]</sup>	Retired <sup>[33]</sup>
		11 October 2017	F9-043 △	234 days	SES-11 <sup>[49]</sup>	Success (39A)	Success (OCISLY)	

# Total Number of Successful and Failure Mission Outcomes



## SQL Query

```
[13]: %sql select MISSION_OUTCOME, count(*) as Counter from SPACEXTBL group by MISSION_OUTCOME
```



	mission_outcome	counter
	Failure (in flight)	1
	Success	99
	Success (payload status unclear)	1

## QUERY EXPLANATION

To display the result set listing the outcome and its count of total number of outcome types found in dataset (i.e. failure (in flight), success, and success (payload status unclear)), we add the "group by" clause to the select statement. The "group by" clause groups a result into subsets that has matching values for one or more columns.



# Boosters Carried Maximum Payload

## SQL Query

```
[33]: %sql select distinct(BOOSTER_VERSION), PAYLOAD_MASS__KG_ from SPACEXTBL \
    PAYLOAD_MASS__KG_ = ( max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```



booster_version	payload_mass_kg
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

## QUERY EXPLANATION

To determine the booster that has carried the maximum payload using a subquery in the WHERE clause and the MAX() function.

# 2015 Launch Records of Failure Drone Ship Landing



## SQL Query

```
[19]: %sql TO_CHAR(TO_DATE(MONTH("DATE"), 'MM'), 'MONTH') as month_name_in_2015, \
    LANDING_OUTCOME, BOOSTER_VERSION, LAUNCH_SITE \
    from SPACEXTBL where LANDING_OUTCOME = 'Failure (drone ship)' AND DATE like '2015%'
```



```
[19]: month_name_in_2015 landing_outcome booster_version launch_site
      JANUARY Failure (drone ship) F9 v1.1 B1012 CCAFS LC-40
      APRIL Failure (drone ship) F9 v1.1 B1015 CCAFS LC-40
```

## QUERY EXPLANATION

To perform a series of conversions in order to extract the month in string format.

- The MONTH( ) function returns the month part for a given date (a number from 1 to 12).
- TO\_DATE converts a formatted date string to a date integer.
- TO\_CHAR performs the reverse operation; it converts a date integer to a formatted date string.

The WHERE clause filters the dataset to Landing\_Outcome = Failure (drone ship). The AND clause specifies additional filter conditions with LIKE keyword has a wild card with the words '2015%' the percentage, in the end, suggests that the Date must start with the year 2015.

Source: [https://en.wikipedia.org/wiki/List\\_of\\_Falcon\\_9\\_first-stage\\_boosters](https://en.wikipedia.org/wiki/List_of_Falcon_9_first-stage_boosters)

S/N <sup>[a]</sup>	Version	Launch date (UTC) <sup>[5]</sup>	Flight No.	Payload <sup>[b]</sup>	Launch	Landing	Status
B1012	v1.1	10 January 2015	F9-014	Dragon C107 (CRS-5)	Success (40)	Failure	Destroyed
B1013	v1.1	11 February 2015	F9-015	DSCOVR	Success (40)	Controlled (ocean)	Expendeed
B1014	v1.1	2 March 2015	F9-016	AUSSAT 3A / Eutelsat 115 West B	Success (40)	No attempt <sup>[18]</sup>	Expendeed
B1015	v1.1	14 April 2015	F9-017	Dragon C108 (CRS-6)	Success (40)	Failure	Destroyed
B1016	v1.1	27 April 2015	F9-018	TürkmenAlem 52°E / MonacoSAT	Success (40)	No attempt <sup>[18]</sup>	Expendeed

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



## SQL Query

```
[30]: %sql select LANDING_OUTCOME, count(*) as counter_btwn_2010Jun04_to_2017Mar20 \ 
    DATE between '2010-06-04' and '2017-03-20' \
    group by LANDING_OUTCOME order by counter_btwn_2010Jun04_to_2017Mar20 desc
```



landing_outcome	counter_btwn_2010jun04_to_2017mar20
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

## QUERY EXPLANATION

Selected Landing outcomes and the COUNT of landing outcomes from the data and used the WHERE clause to filter for landing outcomes BETWEEN 2010-06-04 to 2010-03-20. Also applied the GROUP BY clause to group the landing outcomes and the ORDER BY clause to order the grouped landing outcome in DESCENDING ORDER.

A nighttime satellite view of Earth from space, showing city lights and auroras.

Section 3

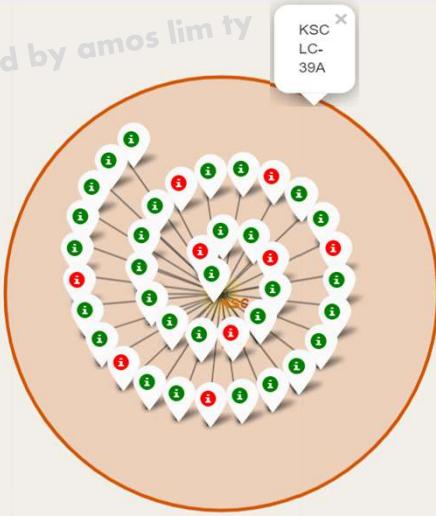
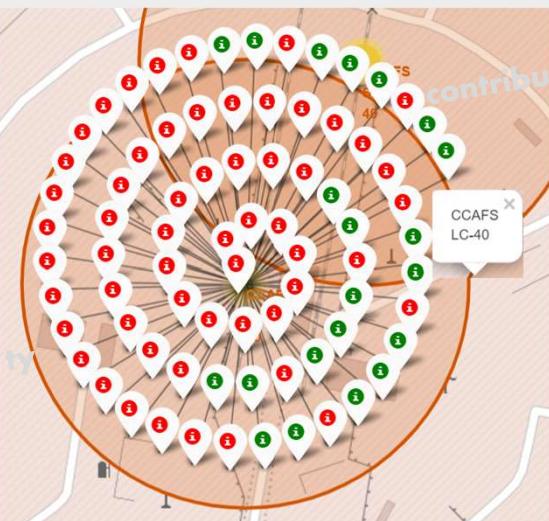
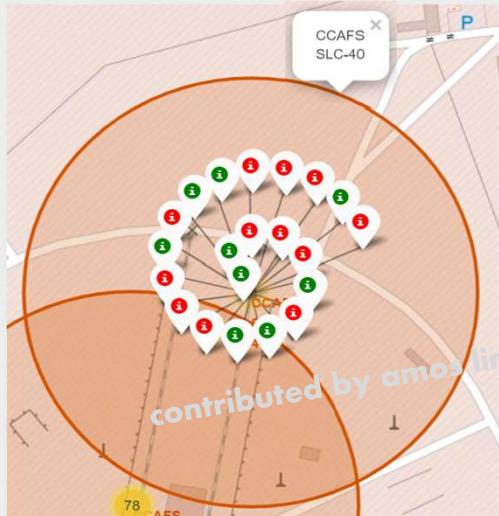
# Launch Sites Proximities Analysis

# Global Locations Of All Space X Launch Sites

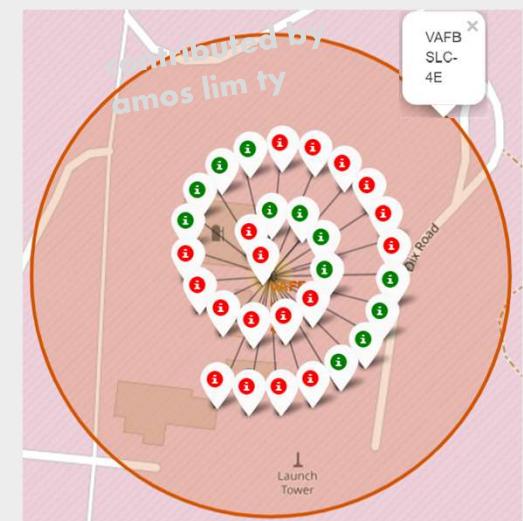


# Launch Success Rate Analysis by Sites

## Florida Launch Sites



## California Launch Site



- CCAFS LC-40 had the **worst** success rate, but it also clocked the **most** launches.
- KSC LC-39A has the **best** success rate.

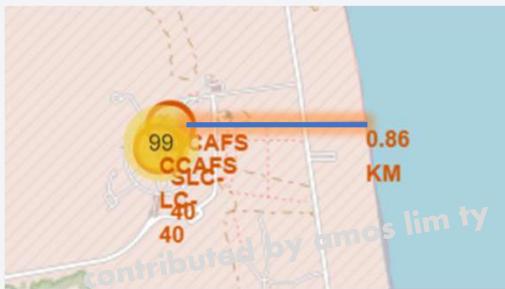
### Color Maker Legend

 Success

 Failure

# Proximities Analysis - CCAFS SLC-40 Site Only

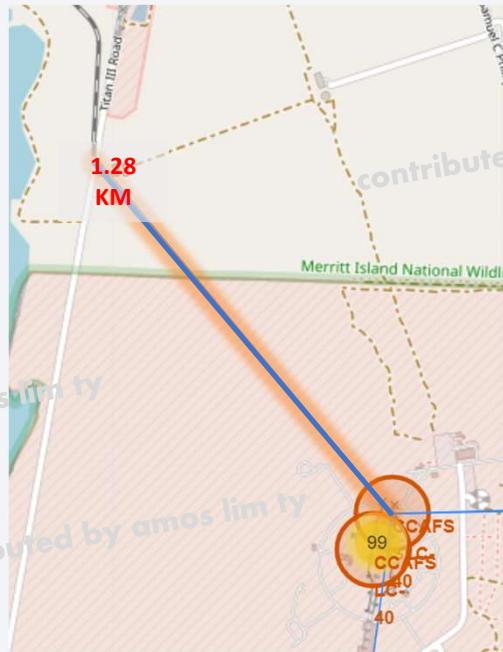
Distance to  
Closest Coastline



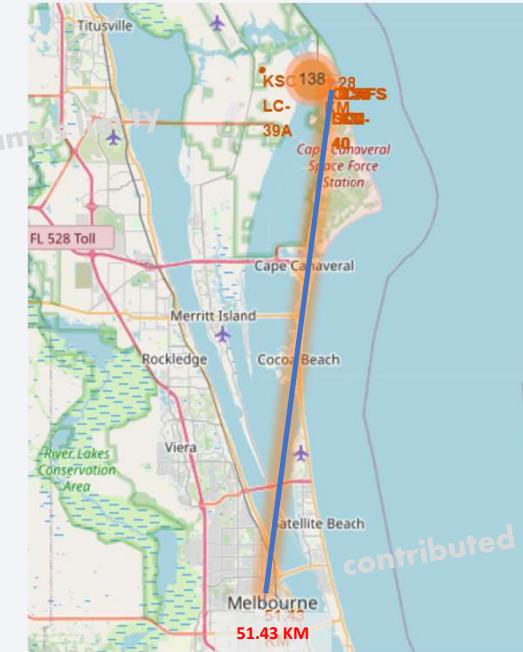
Distance to  
Closest Highway



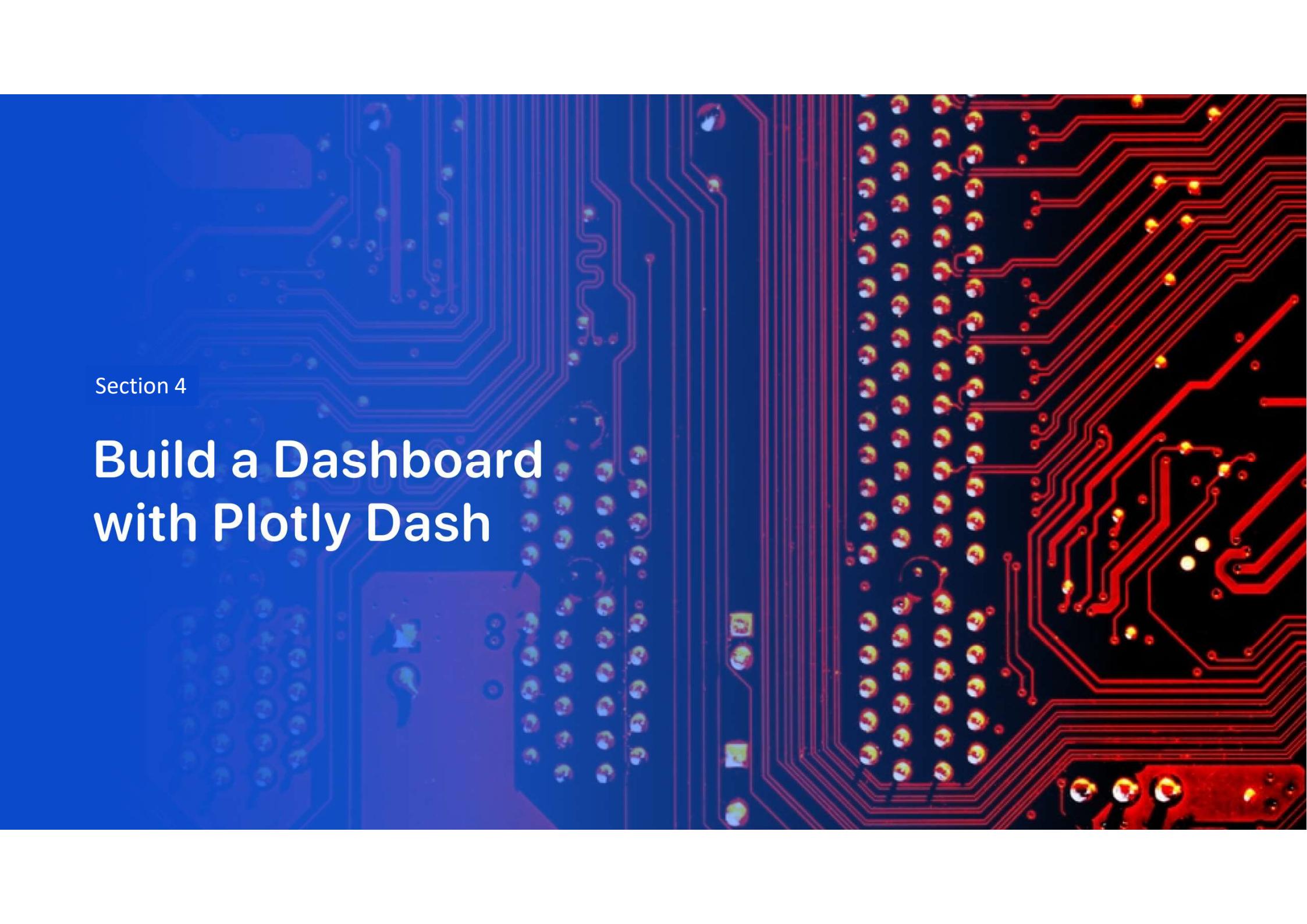
Distance to  
Closest Railway



Distance to  
Closest City



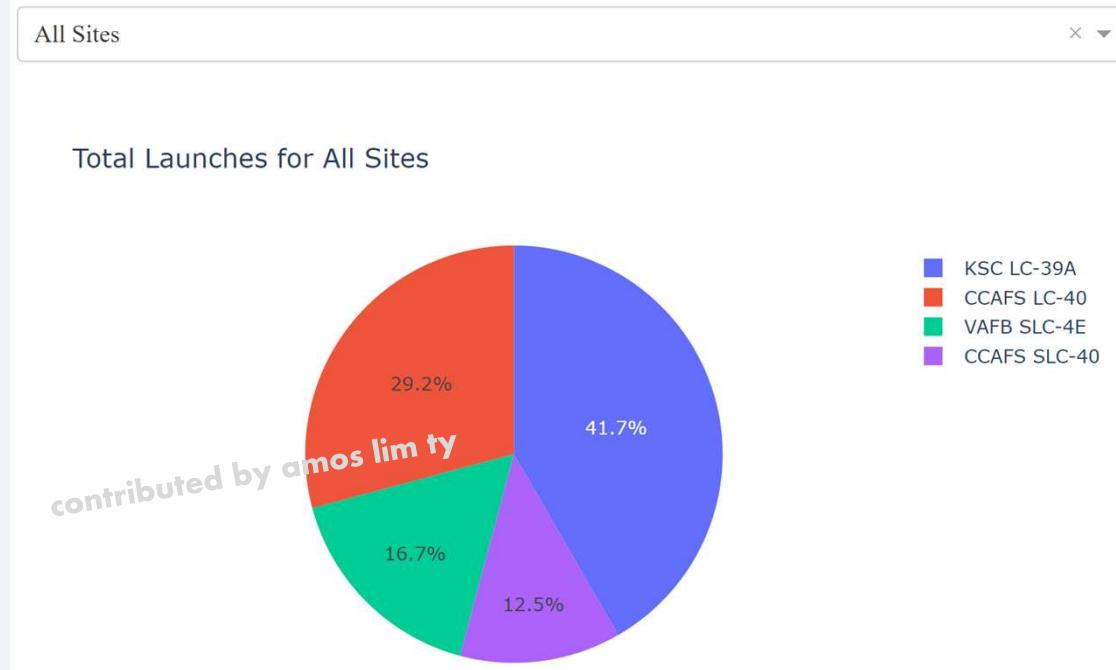
- Observed that the transport infrastructures (e.g. highway, railway, etc) are accessible to the site, which is typically  $< 1.5\text{km}$  in proximity. It is believed for better logistic efficiency.
- Site is far ( $>50\text{ km}$ ) from the closest city, which supports the safety reason.
- All the site proximities consideraten is necessary requirements but might not contribute to the launch success rate directly



Section 4

## Build a Dashboard with Plotly Dash

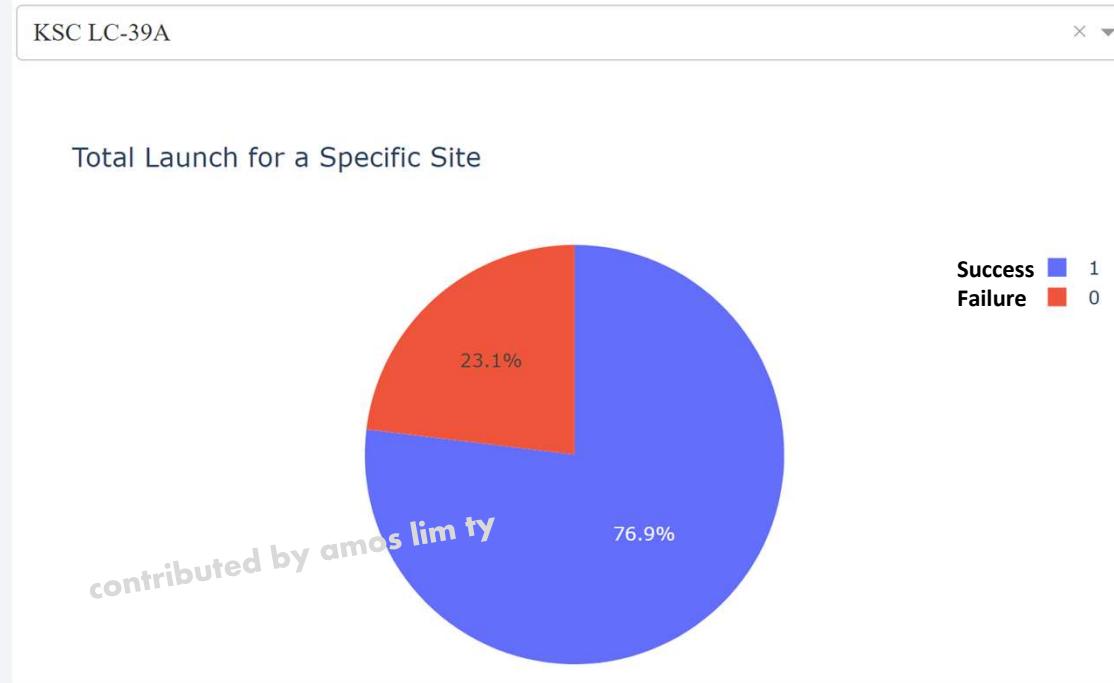
# Overall Sites' Successful Launches Contribution



- **KSC LC-39A** had the highest contribution (**41.7%**) to successful launches when compared to all sites.

Data Source: Provided SpaceX Launch Dash Dataset in .csv format

# Success Rate of Site With the Overall Highest Successful Launches Contribution



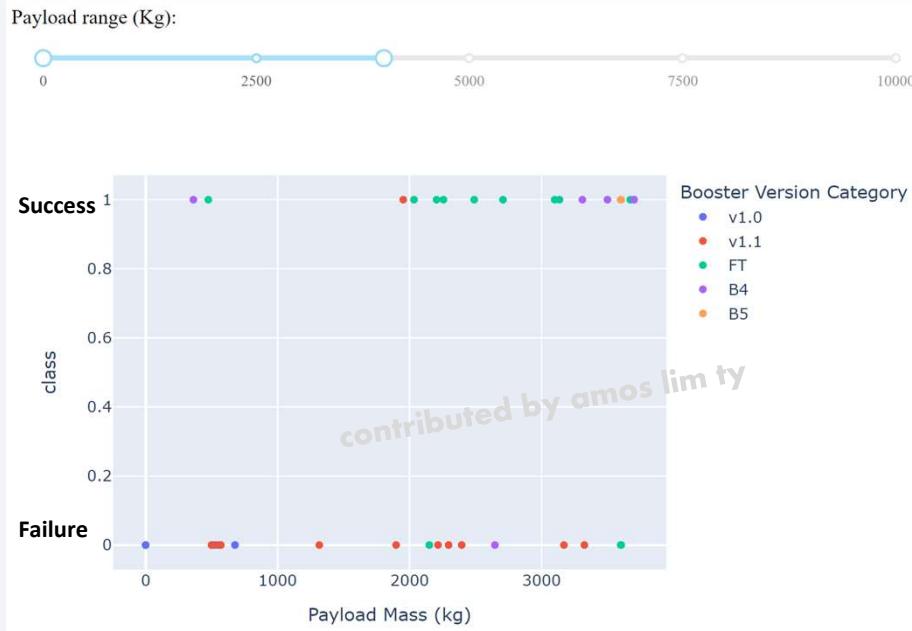
- KSC LC-39A site itself had achieved **76.9%** successful launches and **23.1%** of failed launches.

Data Source: Provided SpaceX Launch Dash Dataset in .csv format

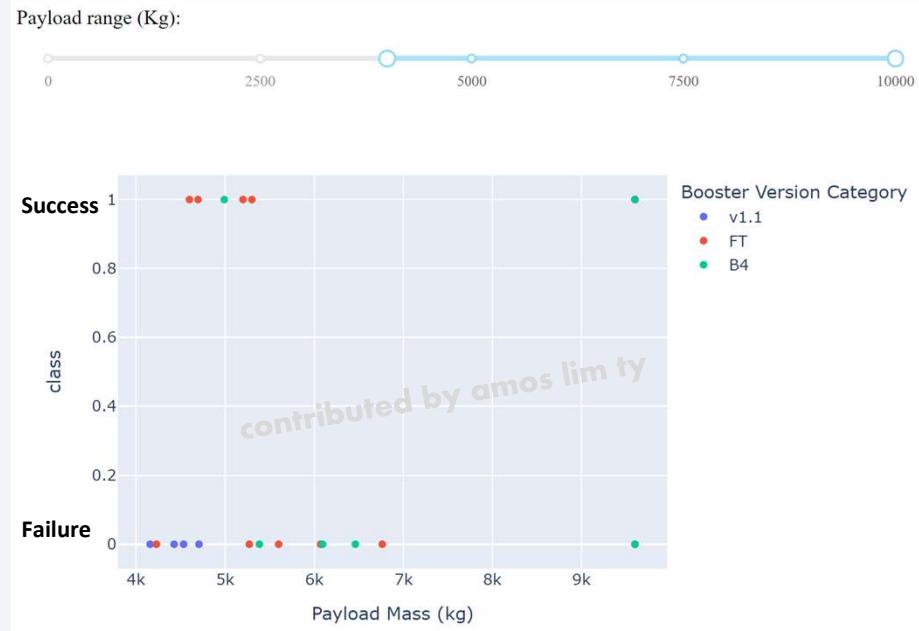
# Different Payload Vs. Launch Outcome For All Sites



## Lighter Payload: 0KG to 4,000KG



## Heavier Payload: 4,001KG to 10,000KG



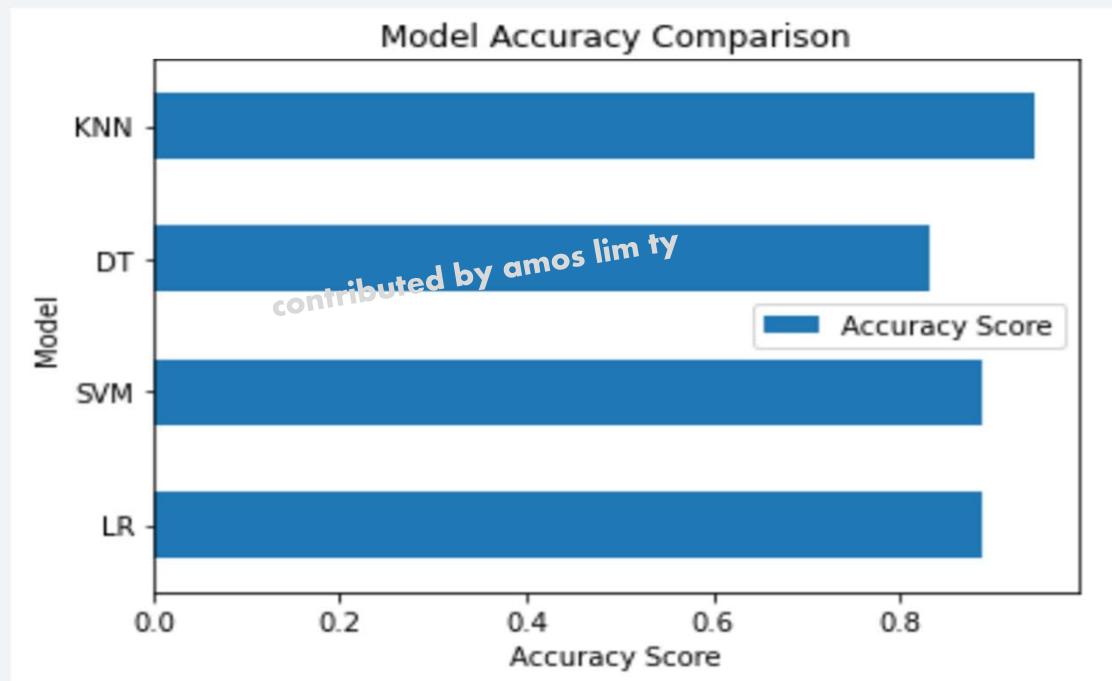
- There were more successful launches with lighter payloads, particularly between 2,000Kg to 4,000Kg.
- There were more Booster Version Categories for lighter payloads.

The background of the slide features a dynamic, abstract design. It consists of several curved, glowing lines in shades of blue and yellow, creating a sense of motion and depth. The lines are thicker in the center and taper off towards the edges, with the blue lines on the left and yellow lines on the right. The overall effect is reminiscent of a tunnel or a high-speed travel through a digital space.

Section 5

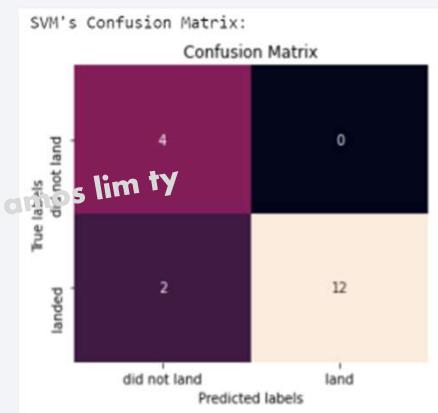
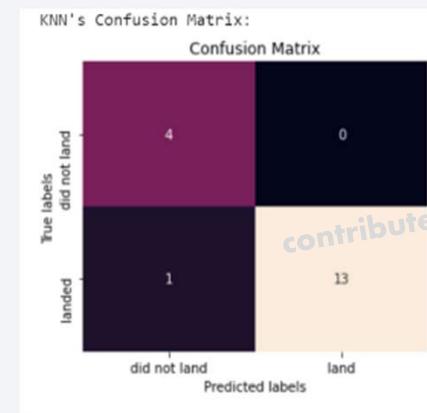
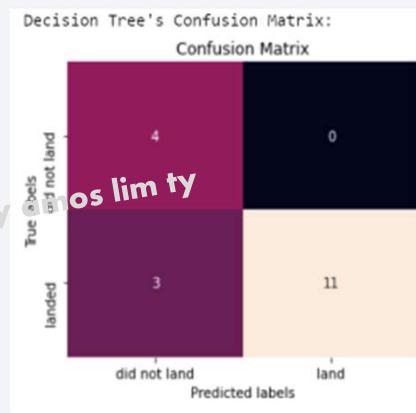
# Predictive Analysis (Classification)

# Classification Accuracy Comparison



- **K-Nearest Neighbour (KNN)** is the best classification model for achieving **94.44% accuracy** on the test data.

# Confusion Matrix Comparison



- The confusion matrix for the **K-Nearest Neighbour (KNN)** classification model shows the highest of 'True Positive' (i.e. 13) and the lowest of 'False Negative' (i.e. 1).

		Predicted Values	
		Negative	Positive
Actual Values	Negative	TN	FP
	Positive	FN	TP

# Conclusions

---

- SpaceY could bid against SpaceX for a rocket launch because of our ability to predict the first-stage landing outcomes.
- Each successful first stage landing is worth about 100 million dollars. This is a significant opportunity cost for every launch.
- We discovered there are 10-15 factors that could help to build the prediction model and there are strong correlations among them.
- We also observed that Elon Musk increased his chances of success through failure. While we can shorten the process by learning about his failures, “Failure is still truly life’s greatest teacher”.



Thank you!

