IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

PATRICIA
RODRÍGUEZ

Outline

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

Executive Summary

Metodology

- Data Collection and API Interaction
- Data Wrangling and Cleaning
- Exploratory Data Analysis (EDA):
- Predictive Analysis
- Visual analytics and Dashboard using Folium and Plotly Dash
- Model Building and Evaluation

Results:

- **Predictive Modeling:** Machine learning models are trained to predict the likelihood of successful rocket landings.
- **Cost Estimation:** The cost of a launch can be estimated based on the predicted landing success.
- **Performance Analysis:** The performance of different rockets and landing methods can be analyzed.

Introduction

- SpaceX, founded by Elon Musk in 2002, aimed to revolutionize space travel and make life multiplanetary. Traditional space launches were expensive and innovation was stagnant. Musk's vision of Mars colonization required drastically lower costs. SpaceX faced immense technical challenges developing reusable rockets, experiencing early launch failures. Financial risks were significant, with the company nearing bankruptcy multiple times. They entered a competitive industry, demanding constant innovation. Regulatory hurdles added further complexity. Early successes like Falcon 1's launch proved their concept. Subsequent development of Falcon 9 and Dragon spacecraft marked major milestones. These advancements significantly reduced launch costs and increased access to space.

Section 1

Methodology

Methodology

- Data Acquisition: Gathered data from two primary sources: the SpaceX REST API for programmatic access to structured launch data, and Wikipedia, utilizing web scraping with BeautifulSoup to supplement with contextual information and potentially unstructured data.
- Data Preprocessing: Executed data cleaning and transformation, addressing missing values by imputation with the mean for numerical features and assigning 0 to non-applicable fields. Applied one-hot encoding to categorical variables for normalization and to prepare data for machine learning models.
- **Exploratory Data Analysis: Conducted in-depth exploratory data analysis leveraging data visualization techniques to uncover patterns, trends, and outliers, complemented by SQL queries for data aggregation and filtering to facilitate deeper insights.**
- **Interactive Visual Analytics: Developed interactive dashboards using Folium for geospatial visualization of launch sites and trajectories, and Plotly Dash for creating dynamic and interactive charts to explore relationships between variables and enable user-driven data exploration.**
- Predictive Modeling: Developed and evaluated classification models to predict [Specify what you are predicting, e.g., launch success, landing outcome]. Implemented a comprehensive pipeline including data validation, model training using Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbors algorithms, and rigorous model evaluation using appropriate metrics to determine the best performing model.

Data Collection

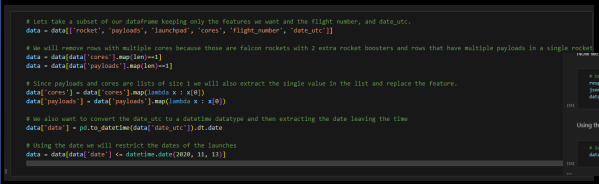
Data Acquisition Strategy: To comprehensively analyze SpaceX Falcon 9 launch data, a dual approach was employed, combining structured data retrieval via API with supplementary information gathered through web scraping to enrich the dataset for subsequent analysis.

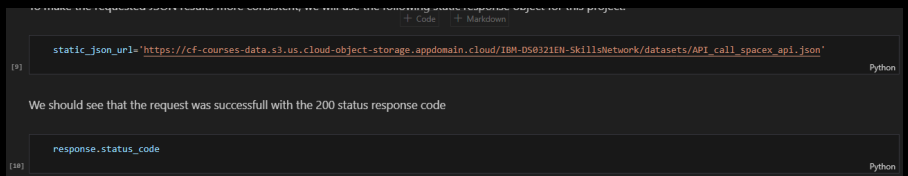
Launch records for Falcon 9 were programmatically accessed using the SpaceX REST API. A GET request was implemented to retrieve structured JSON data, which was then parsed and transformed into a Pandas DataFrame for efficient data manipulation and analysis.

Wikipedia Web Scraping: To augment the API data with contextual details, targeted web scraping was performed on relevant Wikipedia pages pertaining to specific Falcon 9 launches. Beautiful Soup was utilized to parse the HTML content, extract pertinent information, and structure it into a Pandas DataFrame for integration with the API data.

We use the get comands to collect the data and the API to get the information about the launches

Data Collection – SpaceX API





https://github.com/PatrizziaRR/IBM-SPACE-X-CAPSTONE-PROJECT/blob/main/1.Spacex_data-collection-api.ipynb

We use the web scrappingto to obtain the Falcon 9 launches with Beautifuls soup

Data Collection - Scraping

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```
[4]

```
# Use soup.title attribute
soup.find('title')
```
[7]

```
column_names = []
# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided extract_column_from_header() to get a column name
# Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names
for col in first_launch_table.find_all('th'):
    name = extract_column_from_header(col)
    if name != None and len(name) > 0:
        column_names.append(name)
```
[20]

```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url)
response.status_code
```
[5]

```
200
```

Create a `BeautifulSoup` object from the HTML `response`

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text, 'html.parser')
```
[6]

https://github.com/PatrizziaRR/IBM-SPACE-X-CAPSTONE-PROJECT/blob/main/2.Spacex_webscraping%20.ipynb

Data Wrangling

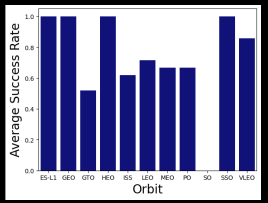
- We dove into the data to understand it better and figure out what categories or tags (labels) we needed to use to train our model.
- We counted how many rocket launches happened at each location and tracked the different paths the rockets took in space (orbits), noting how many times each path was used.
- we saved all our organized data, including these labels, into a spreadsheet file (CSV)

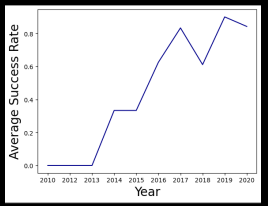
https://github.com/PatrizziaRR/IBM-SPACE-X-CAPSTONE-PROJECT/blob/main/3.Spacex-Data%20wrangling.ipynb

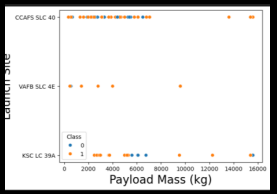
EDA with Data Visualization

- To explore potential connections, we created scatter plots visualizing how these factors relate: the number of the launch and the weight of the payload, the launch number and its location, payload weight compared to launch site, launch number and orbital path, and payload weight vs. orbital path

- To analyze success rate trends across launch sites, we used a bar chart to visualize success rates for each orbit type.
- A line chart visually represented the trend of successful launches over the years, allowing us to assess improvements in success rates.







https://github.com/PatrizziaRR/IBM-SPACE-X-CAPSTONE-PROJECT/blob/main/4.Spacex_Data_visualitation.ipynb.jupyterlite.ipynb

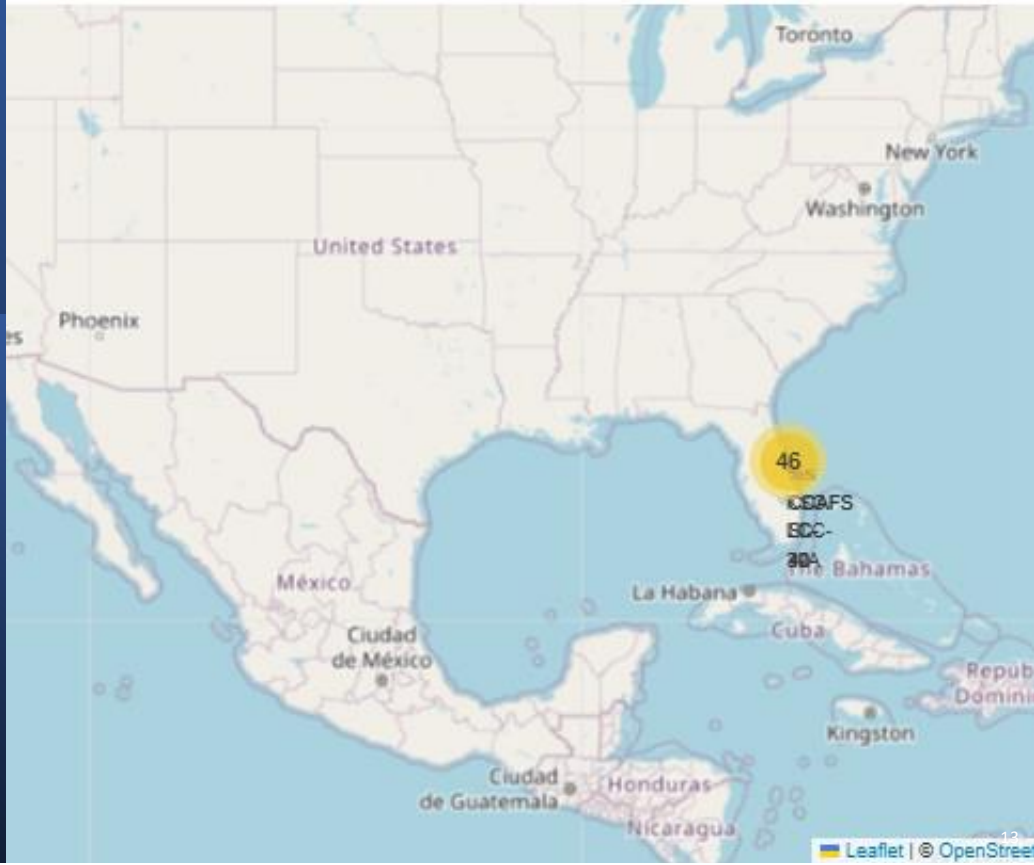
EDA with SQL

- Identify the primary launch sites used in the space mission and analyze the factors influencing launch site selection (e.g., geographical location, payload requirements, cost considerations).
- Calculate the cumulative payload mass delivered to orbit by NASA (CRS) missions using the specified boosters.
- Determine the average and median payload mass lifted by the F9 v1.1 booster variant. Compare this with the average payload mass of other booster versions and discuss the potential performance differences.
- Document the date of the first successful ground pad landing and analyze the key factors that contributed to achieving this milestone.
- dentify boosters that have successfully landed on a drone ship while carrying a payload between 4000 and 6000 kg.
- Calculate the overall mission success rate and failure rate.
- Analyze potential correlations between mission outcomes and factors such as booster version, payload mass, launch site, and weather conditions.
- Identify the booster version(s) that have demonstrated the highest payload capacity. Investigate the technical specifications of these boosters and discuss the design features that enable them to lift heavier payloads.
- Rank 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.
- Analyze the trends in landing outcomes (success/failure, drone ship/ground pad) over the specified time period. Visualize these trends and discuss the factors that may have influenced the evolution of landing strategies.

https://github.com/PatrizziaRR/IBM-SPACE-X-CAPSTONE-PROJECT/blob/main/5.Spacex_EDA_SQL.ipynb

Interactive map created with Folium to get those ideal locations for launch sites

Build an Interactive Map with Folium



Build a Dashboard with Plotly Dash

Types of charts used in analyzing launch data:

- **Success Pie Chart:** This chart shows the proportion of successful launches out of all launches for a specific launch site or for all sites combined. It's meant to provide a clear visual representation of launch success rates.
- **Payload Success Scatter Plot:** This chart explores the relationship between the mass of the payload (the cargo carried by the rocket) and whether the launch was successful. It aims to help understand if there's a link between payload size and launch outcomes.

the dashboard includes two interactive elements:

- A dropdown menu allows users to select a specific launch site or view data from all sites, enabling customized analysis.
- A slider filters the scatter plot by payload mass range, giving users dynamic control over the data for more granular exploration.

Predictive Analysis (Classification)

- Data was collected from two CSV files.Exploratory data analysis was conducted using Seaborn and Matplotlib to visualize the features and identify the class variable.Features were engineered, the data was standardized, and then split into training and testing sets.Several models, including Logistic Regression, SVM, Decision Tree, and KNN, were evaluated. GridSearchCV was used for hyperparameter tuning.Model performance was assessed using a confusion matrix, precision, recall, and cross-validation techniques.The final model was selected based on achieving the highest accuracy and recall scores.

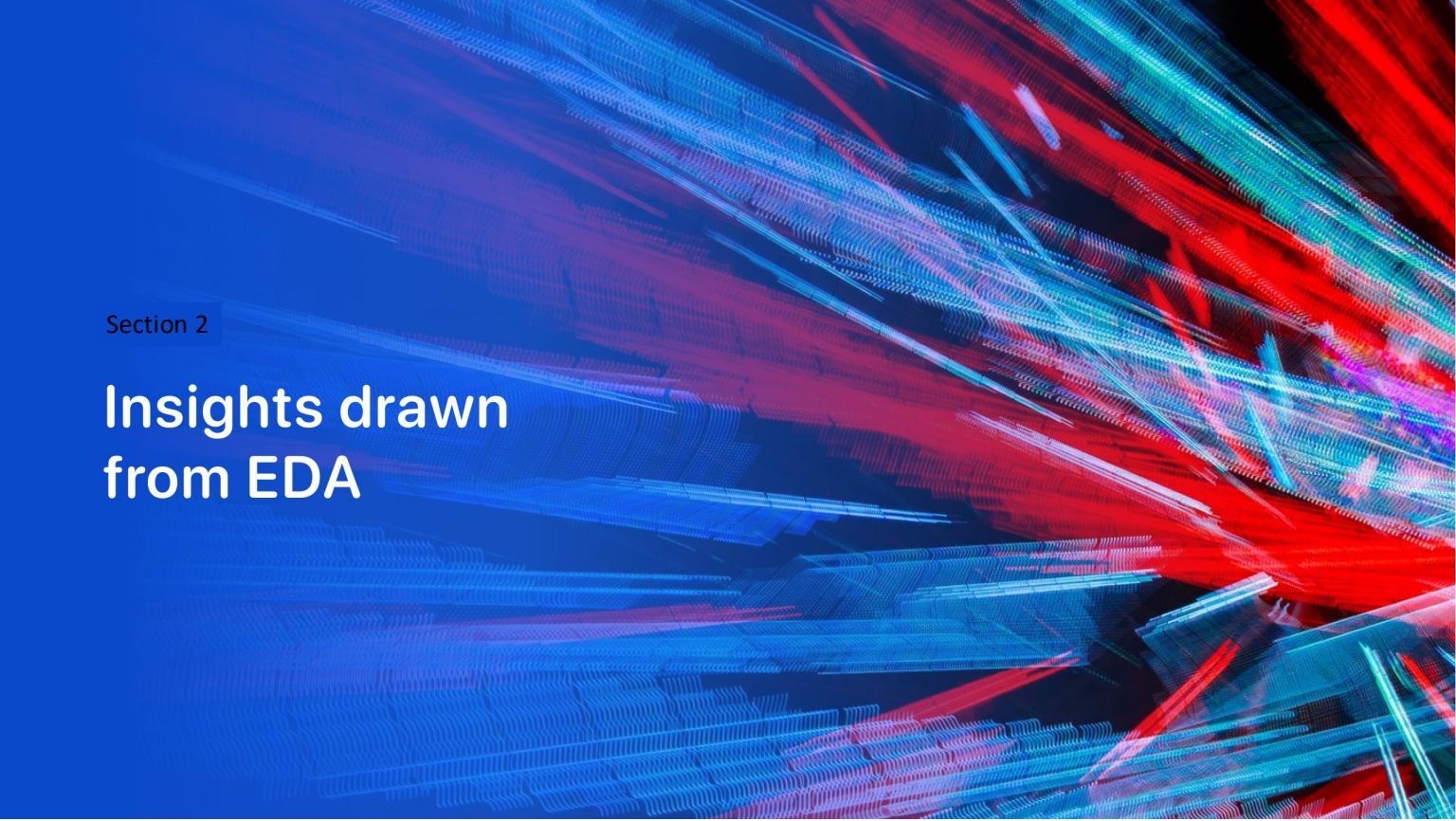
Results

Exploratory Data Analysis:

- **Visualization:** Utilized Seaborn and Matplotlib to explore feature distributions and correlations within the dataset.
- **Target Identification:** Defined the class variable as "successful landing" for predictive modeling.

Predictive Analysis:

- **Model Selection:** Selected the optimal predictive model based on achieving the highest accuracy and recall metrics.
- **Evaluation:** Assessed the model's predictive capabilities using a confusion matrix, precision, recall, and cross-validation techniques.

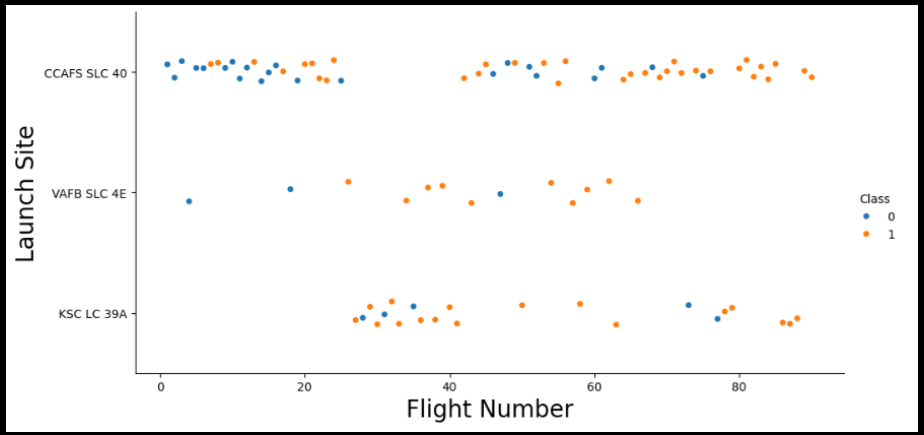


Section 2

Insights drawn from EDA

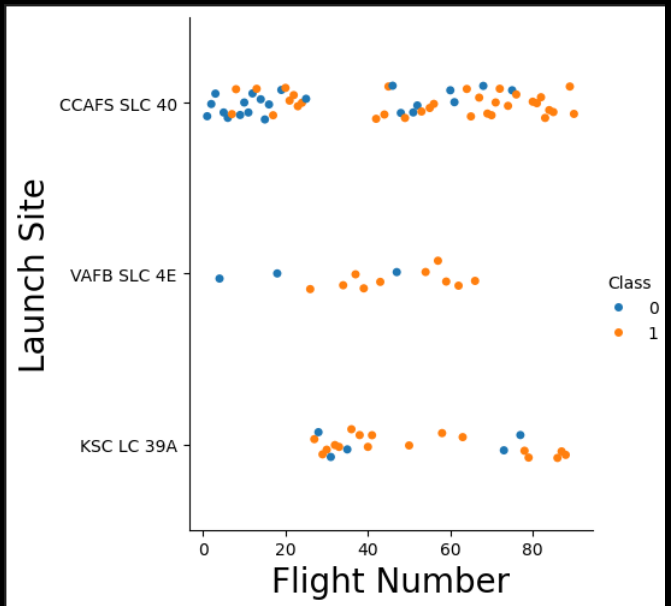
Flight Number vs. Launch Site

As many flights at a launch site the greater the success rate at a launch site.



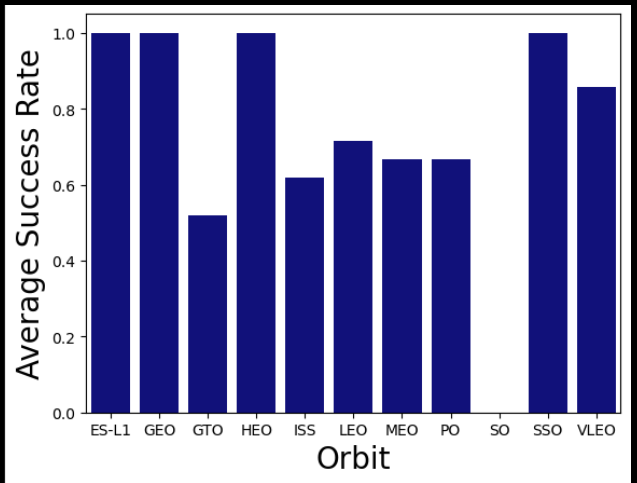
Payload vs. Launch Site

The greater the payload mass for Launch Site CCAFS SLC 40 the higher the success rate



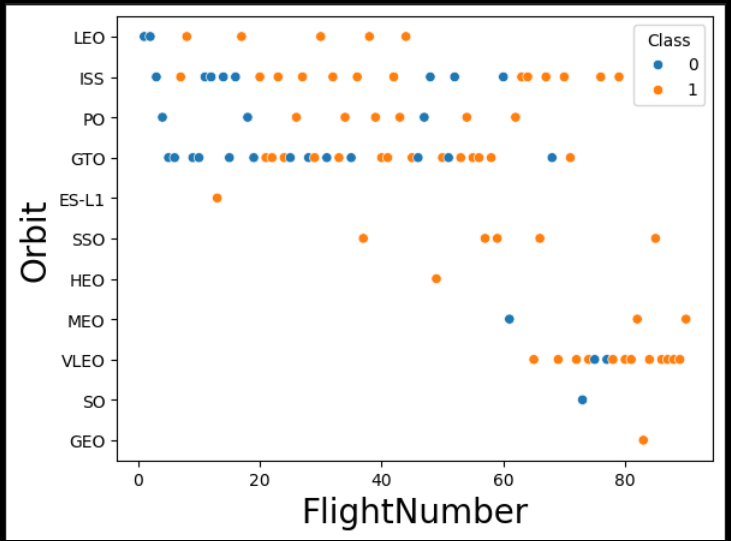
Success Rate vs. Orbit Type

Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate



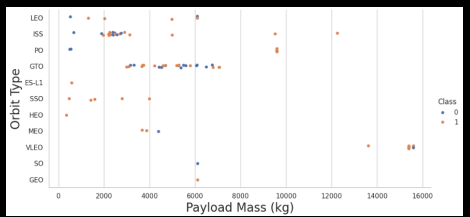
Flight Number vs. Orbit Type

LEO orbit's Success is related to the number of flights

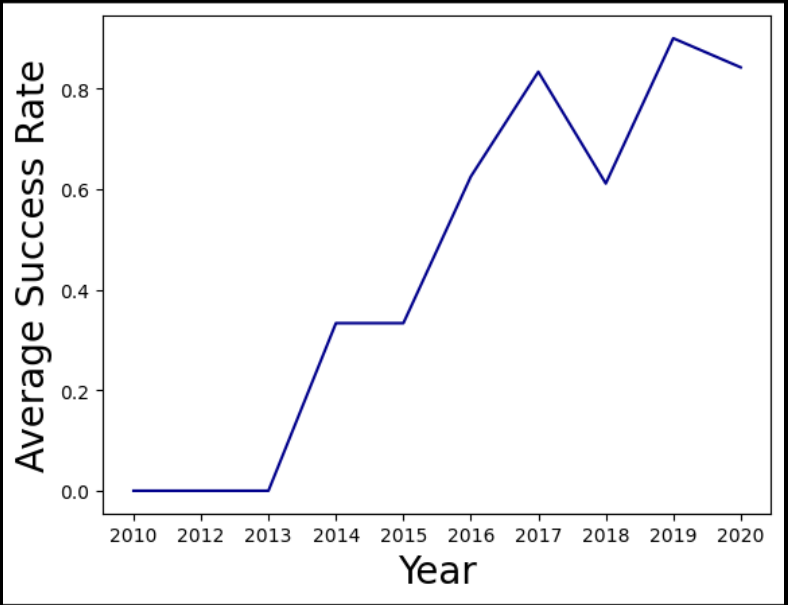


Payload vs. Orbit Type

- eavy payloads negatively influence Geostationary Transfer Orbit (GTO) launches, while positively influencing launches to Polar Low Earth Orbit (LEO), including those servicing the International Space Station (ISS)



Launch Success Yearly Trend



Sucess rate since 2013 kept increasing till 2020

All Launch Site Names

Tha launch sites are displaying by using distinct

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

```
%%sql
SELECT
    DISTINCT("Launch_Site")
FROM
    SPACEXTABLE
```

Launch Site Names Begin with 'CCA'

Using SQL the launch sites begin with the string ‘CCA’ are displayed

```
%%sql
SELECT
    *
FROM
    SPACEXTABLE
WHERE
    "Launch_Site" LIKE 'CCA%'
LIMIT 5
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

```
%%sql
SELECT
    SUM("PAYLOAD_MASS__KG_") AS total_payload_mass,
    "Customer"
FROM
    SPACEXTABLE
WHERE
    "Customer" = 'NASA (CRS)'
GROUP BY "Customer"
```

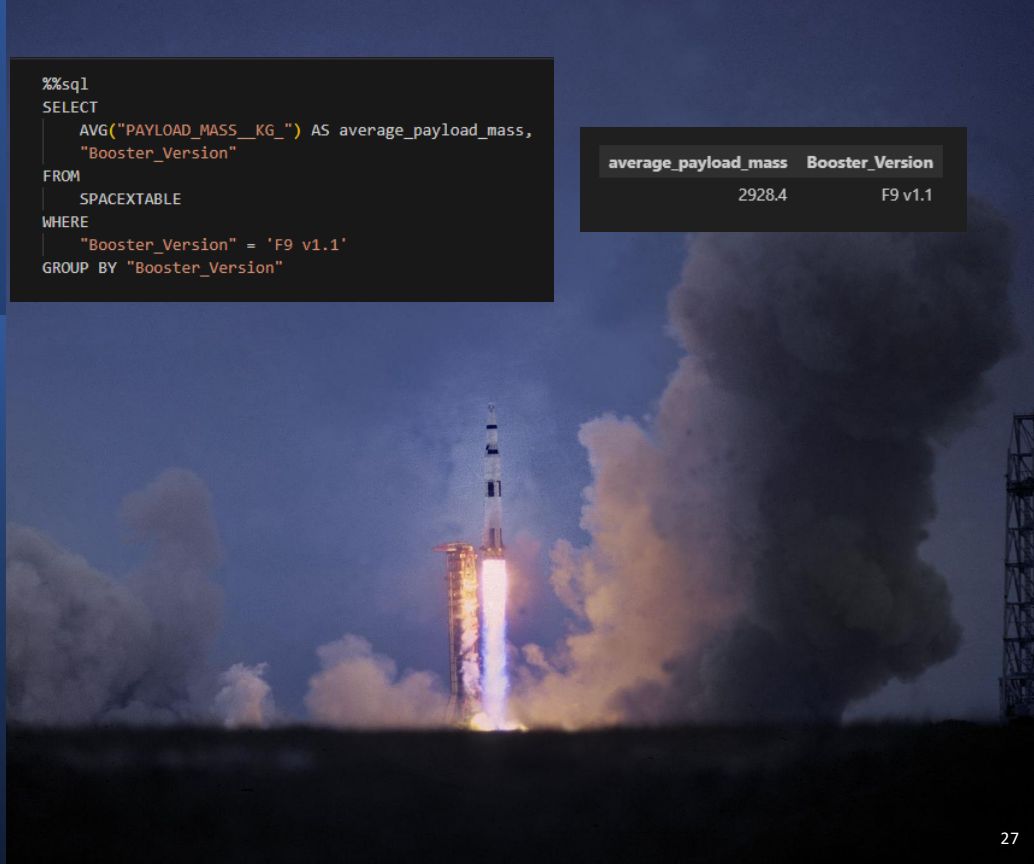
total_payload_mass	Customer
45596	NASA (CRS)

Average Payload Mass by F9 v1.1

We display the average payload mass by F9v1,1

```
%%sql
SELECT
    AVG("PAYLOAD_MASS__KG_") AS average_payload_mass,
    "Booster_Version"
FROM
    SPACEXTABLE
WHERE
    "Booster_Version" = 'F9 v1.1'
GROUP BY "Booster_Version"
```

average_payload_mass	Booster_Version
2928.4	F9 v1.1



First Successful Ground Landing Date

Using the ‘Date ‘and the ‘Landing _Outcome’ we find the first successful ground landing date

```
%%sql
SELECT
    "Date",
    "Landing_Outcome"
FROM
    SPACEXTABLE
WHERE
    "Landing_Outcome" = 'Success (ground pad)'
ORDER BY "Date" ASC
LIMIT 1
```

Date	Landing_Outcome
2015-12-22	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

Using select we find those boosters which have success in drone ship and have payload mass greater tan 4000 but les tan 6000

```
%%sql
SELECT
    "Booster_Version",
    "Landing_Outcome",
    "PAYLOAD_MASS__KG_"
FROM
    SPACEXTABLE
WHERE
    "Landing_Outcome" = 'Success (drone ship)' AND
    "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000
GROUP BY "Booster_Version"
```

Booster_Version	Landing_Outcome	PAYLOAD_MASS__KG_
F9 FT B1021.2	Success (drone ship)	5300
F9 FT B1031.2	Success (drone ship)	5200
F9 FT B1022	Success (drone ship)	4696
F9 FT B1026	Success (drone ship)	4600

Total Number of Successful and Failure Mission Outcomes

These are the total number of succesful and failures misión outcomes running select from SPACEXTABLE

```
Click to add a breakpoint
SELECT
    "Mission_Outcome",
    COUNT("Mission_Outcome")
FROM
    SPACEXTABLE
GROUP BY "Mission_Outcome"
```

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

Boosters Carried Maximum P ayload

```
%%sql
SELECT
    "Booster_Version",
    "PAYLOAD_MASS__KG_"
FROM
    SPACEXTABLE
WHERE "PAYLOAD_MASS__KG_" = (SELECT
                                MIN("PAYLOAD_MASS__KG_")
                            FROM
                                SPACEXTABLE)
```

Booster_Version	PAYLOAD_MASS__KG_
F9 v1.0 B0003	0
F9 v1.0 B0004	0

Using SQL we obtain boosters carried Maximum, payload

2015 Launch Records

```
%%sql
SELECT
    substr("Date", 0, 5) AS year,
    substr("Date", 6, 2) AS n_month,
    substr('JanFebMarAprMayJunJulAugSepOctNovDec', 1 + 3 * strftime('%m', date("Date")), -3) AS month,
    "Launch_Site",
    "Landing_Outcome",
    "Booster_Version"
FROM
    SPACEXTABLE
WHERE
    "Landing_Outcome" = 'Failure (drone ship)' AND
    substr("Date", 0, 5) = '2015'
```

year	n_month	month	Launch_Site	Landing_Outcome	Booster_Version
2015	01	Jan	CCAFS LC-40	Failure (drone ship)	F9 v1.1 B1012
2015	04	Apr	CCAFS LC-40	Failure (drone ship)	F9 v1.1 B1015

Now we 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.

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

```
%%sql
SELECT
    COUNT("Landing_Outcome") AS n_landing_outcomes,
    "Landing_Outcome"
FROM
    (SELECT
        "Date",
        "Landing_Outcome"
    FROM
        SPACEXTABLE
    WHERE
        ("Landing_Outcome" = 'Success (ground pad)' OR "Landing_Outcome" = 'Failure (drone ship)') AND
         ("Date" > '2010-06-04' AND "Date" < '2017-03-20'))
GROUP BY "Landing_Outcome"
ORDER BY n_landing_outcomes DESC
```

n_landing_outcomes	Landing_Outcome
5	Failure (drone ship)
3	Success (ground pad)

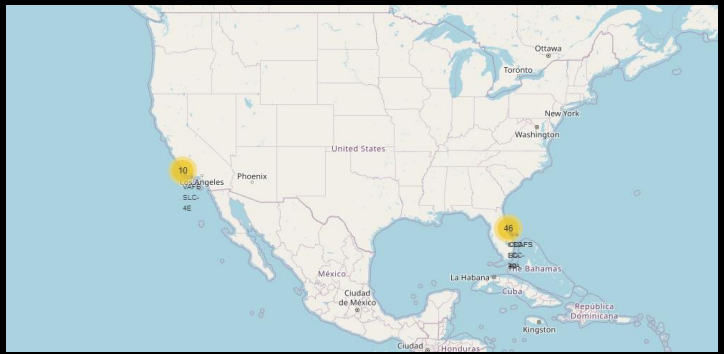
These are the landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

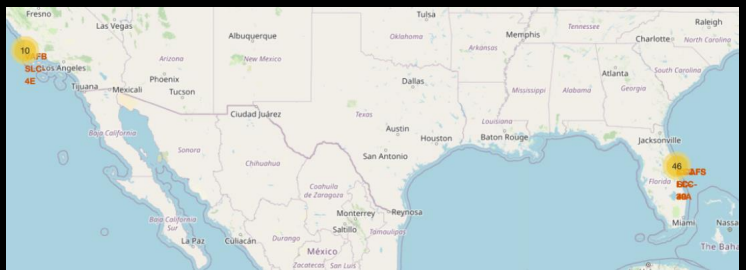
Section 3

Launch Sites Proximities Analysis

SpaceX Falcon9 Launch sites

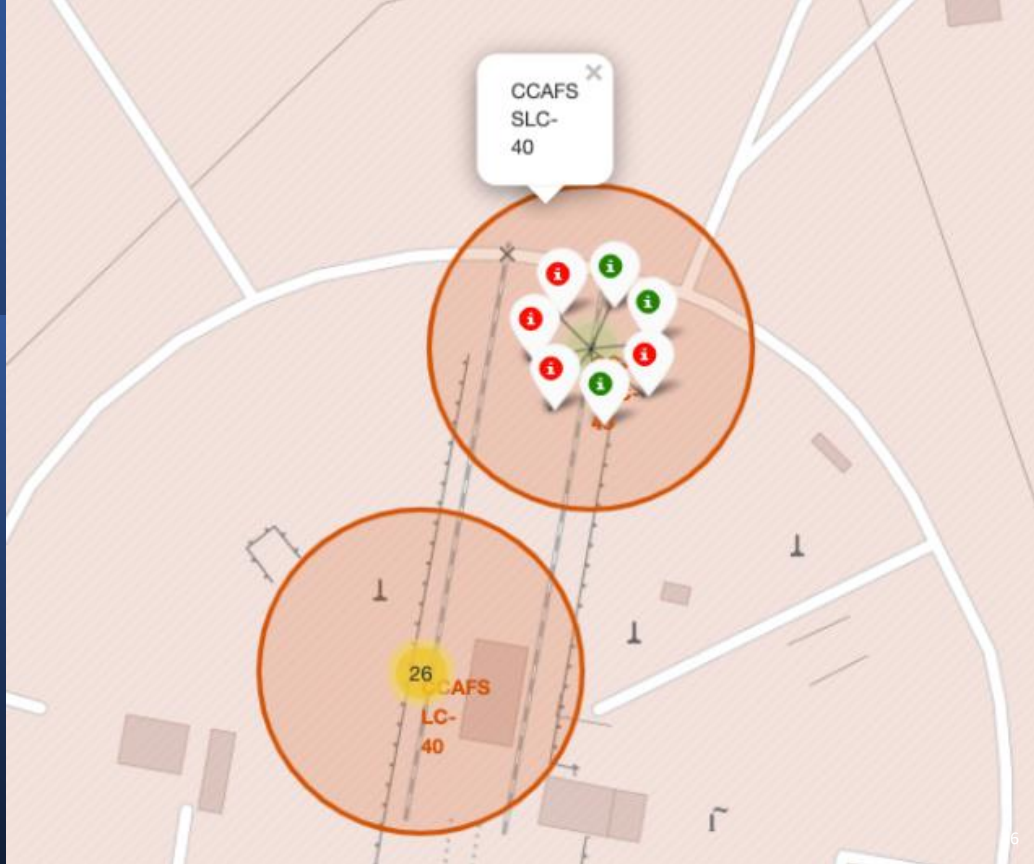
Folium Map indictes those places where lauchs took place





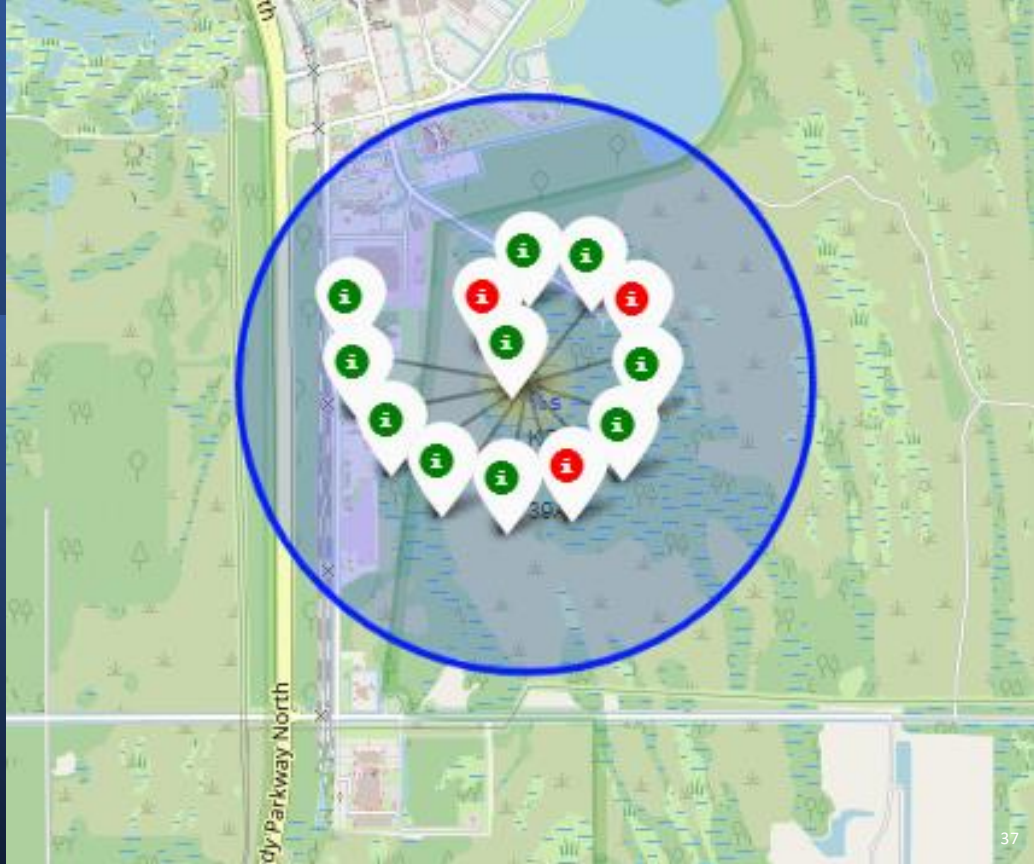
Successful Launch Sites

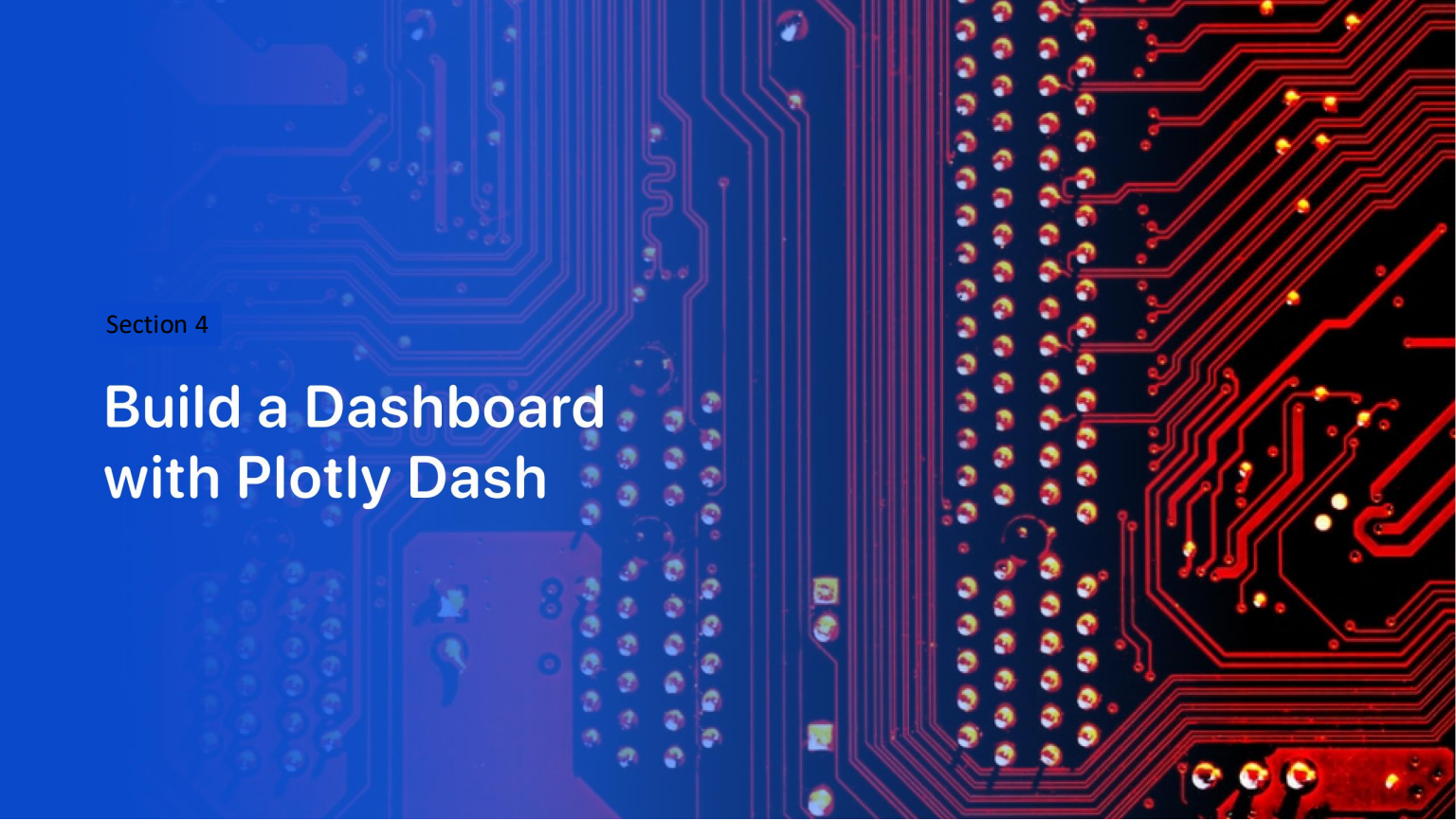
Zoom in on the data points indicating successful and failed launches.



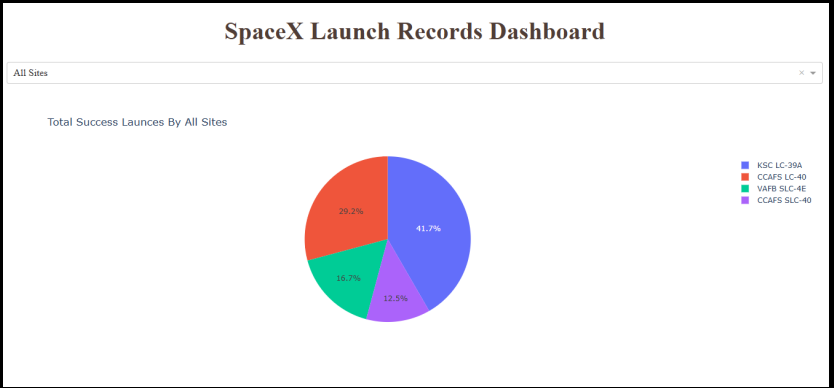
Successful launch sites

KSC obtain 10 succesful launches against 3 failed




Section 4
Build a Dashboard with Plotly Dash

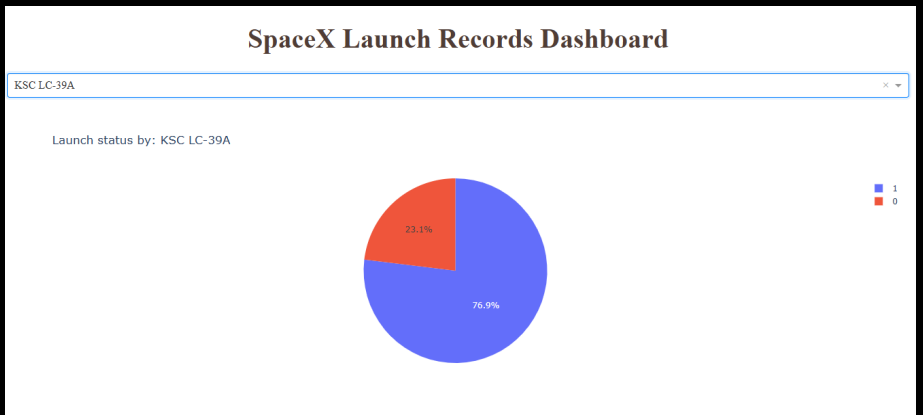
Success launch rate per site



- In this chart we can see the success lauch rate per site, where KSC LC-39A has the highest percentage

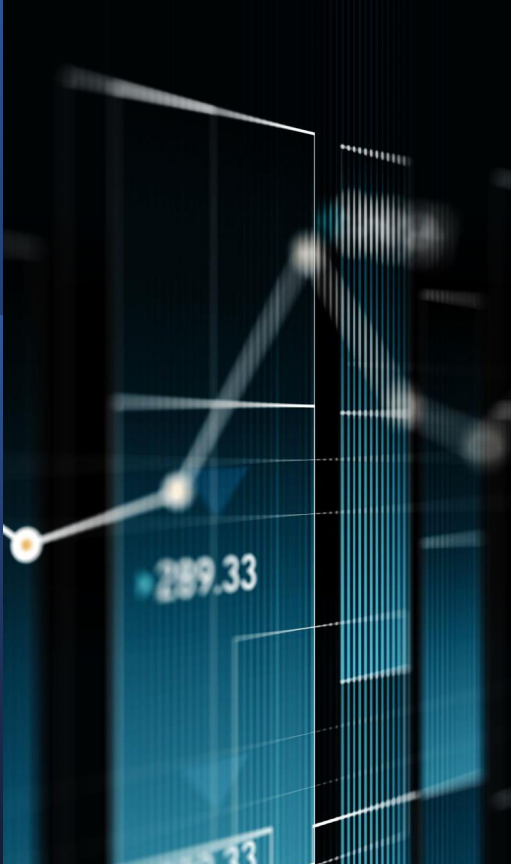
Maximum rate of successful launches

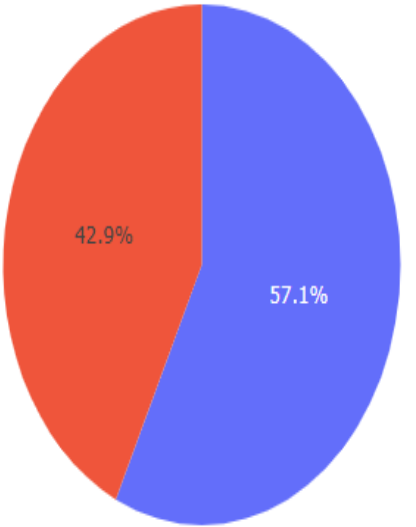
KSC LC-39ª has a 76% rate of successful launches



CCAFS SLC-40 has the minimum percentage of successful lauches with a 42.9%

Minimum rate of successful launches





Section 5

Predictive Analysis (Classification)

Classification Accuracy


```
accuracy_test = [logreg_score, svm_cv_score, tree_cv_score, knn_cv_score]
accuracy_train = [logreg_cv.best_score_, svm_cv.best_score_, tree_cv.best_score_, knn_cv.best_score_]
best_parameters_of_train= [logreg_cv.best_params_, svm_cv.best_params_, tree_cv.best_params_, knn_cv.best_params_]
model = ['Logistic Regression','Support Vector Machine', 'Decision Tree', 'K Nearest Neighbour']
data= {'Model': model, 'Accuracy_Test': accuracy_test, 'Accuracy_Train_Cv': accuracy_train}
df=pd.DataFrame(data)
df
```

Python

	Model	Accuracy_Test	Accuracy_Train_Cv
0	Logistic Regression	0.833333	0.846429
1	Support Vector Machine	0.833333	0.848214
2	Decision Tree	0.833333	0.862500
3	K Nearest Neighbour	0.833333	0.848214

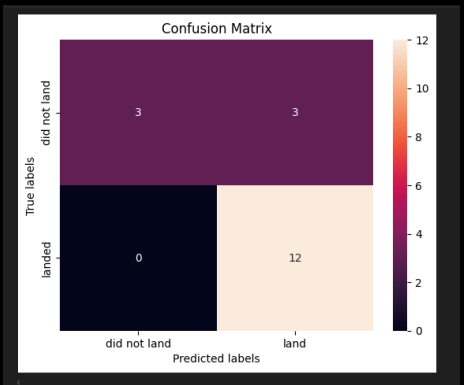
```
df_melted = df.melt(id_vars=['Model'], var_name='Dataset', value_name='Accuracy')
df_melted['Dataset'] = df_melted['Dataset'].replace({'Accuracy_Train_Cv': 'Train', 'Accuracy_Test': 'Test'})
df_melted
```

Python

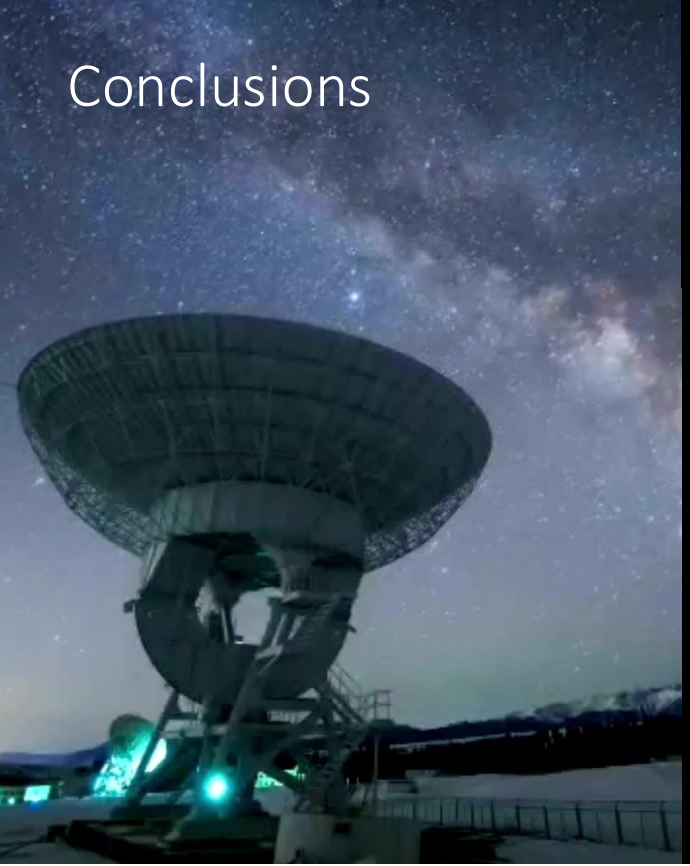
	Model	Dataset	Accuracy
0	Logistic Regression	Test	0.833333
1	Support Vector Machine	Test	0.833333
2	Decision Tree	Test	0.833333
3	K Nearest Neighbour	Test	0.833333
4	Logistic Regression	Train	0.846429
5	Support Vector Machine	Train	0.848214
6	Decision Tree	Train	0.862500

Confusion Matrix

- A confusion matrix is a table used to evaluate the performance of a classification model. It helps visualize how well the model correctly predicted the outcomes (in this case, whether a rocket launch landed successfully or not) compared to the actual outcomes.
- This confusion matrix shows that the model has a moderate accuracy of 66.7%. When it predicts a landing, it tends to be correct 80% of the time (precision). However, it also misses 20% of the actual successful landings (recall).



Conclusions



1.Launch Site Success Rates: The analysis revealed varying success rates across different launch sites. This suggests that factors like location, infrastructure, and environmental conditions can significantly influence the outcome of a launch.
2.Orbit Type and Success: Certain orbit types, such as Polar LEO (including missions to the International Space Station), demonstrated higher success rates compared to others, such as GTO. This highlights the challenges associated with specific orbital trajectories.
3.Payload Mass Influence: The relationship between payload mass and launch success was complex and varied depending on the orbit type. While heavier payloads appeared to negatively affect GTO launches, they showed a positive influence on Polar LEO and ISS missions.
4.Predictive Modeling: Machine learning models, such as Logistic Regression, SVM, Decision Tree, and KNN, were successfully trained to predict launch outcomes based on factors like launch site, orbit type, and payload mass.
5.Model Performance: The selected model achieved a moderate accuracy of 66.7% in predicting launch outcomes. However, further analysis of metrics like precision and recall revealed potential areas for improvement, particularly in minimizing false negatives.
6.Data Visualization: Tools like Seaborn and Matplotlib were instrumental in visualizing data patterns, correlations, and feature distributions. This facilitated exploratory data analysis and informed model selection.
7.Interactive Dashboard: The development of an interactive dashboard with features like dropdown menus and sliders enhanced data exploration and allowed for customized analysis based on user-selected parameters.
8.Future Work: Further research could investigate the specific factors contributing to the varying success rates across launch sites and orbit types. Additionally, refining the predictive model to improve accuracy and minimize false negatives could enhance its practical applications.
Bonus Conclusion:
9.Value of Data Analysis: The project demonstrated the value of data analysis and machine learning in understanding complex phenomena like rocket launches and predicting their outcomes. This knowledge can inform decision-making and improve future space missions.

Thank you!