IBM Developer
SKILLS NETWORK

SpaceY

Winning the Space Race With Data Science

Satvik Agrawal
12/29/23

OUTLINE

Executive Summary

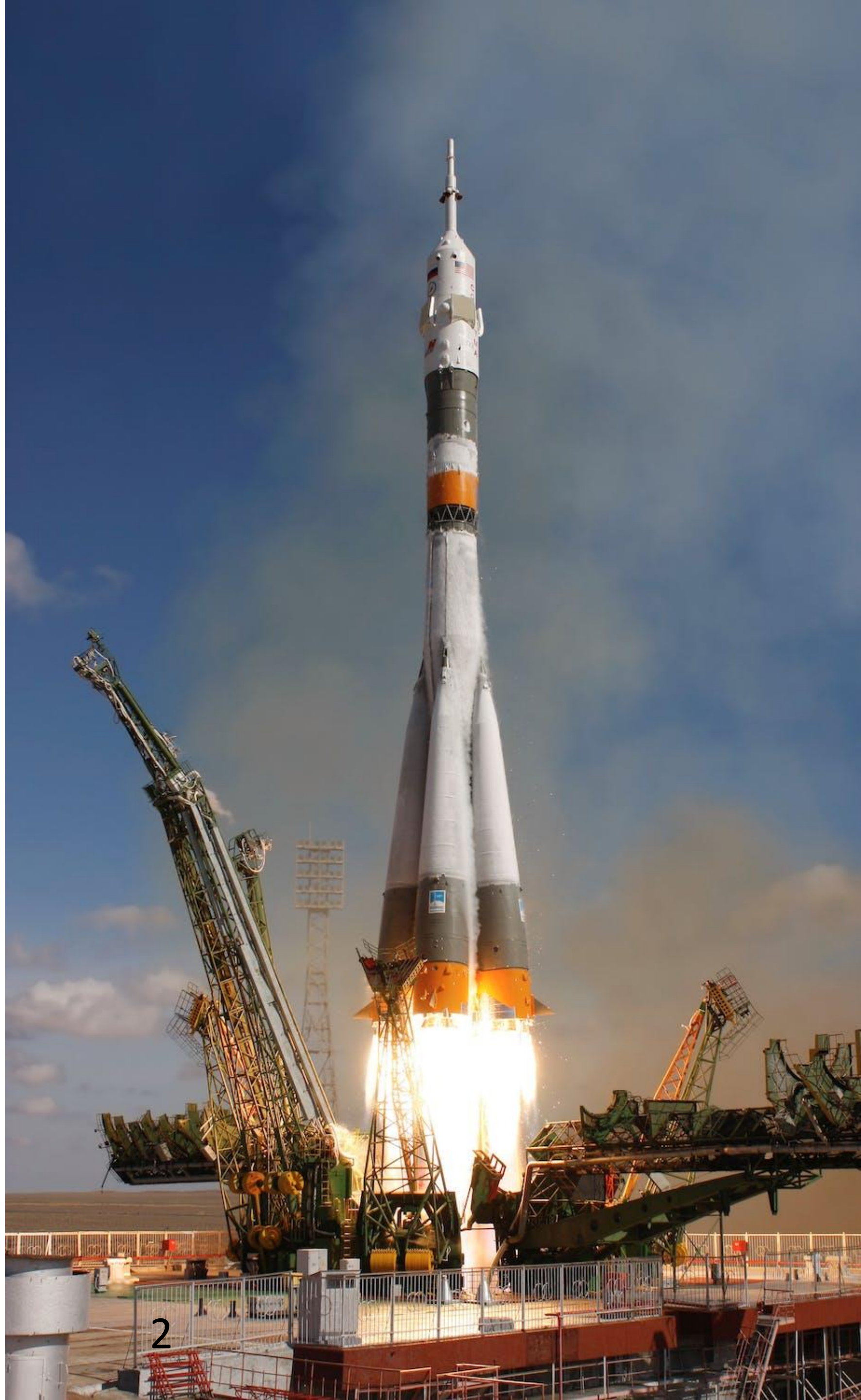
Introduction

Methodology

Results

- EDA with Visualization
- EDA with SQL
- Interactive Maps with Folium
- Plotly Dash Dashboard
- Predictive Analytics

Conclusion

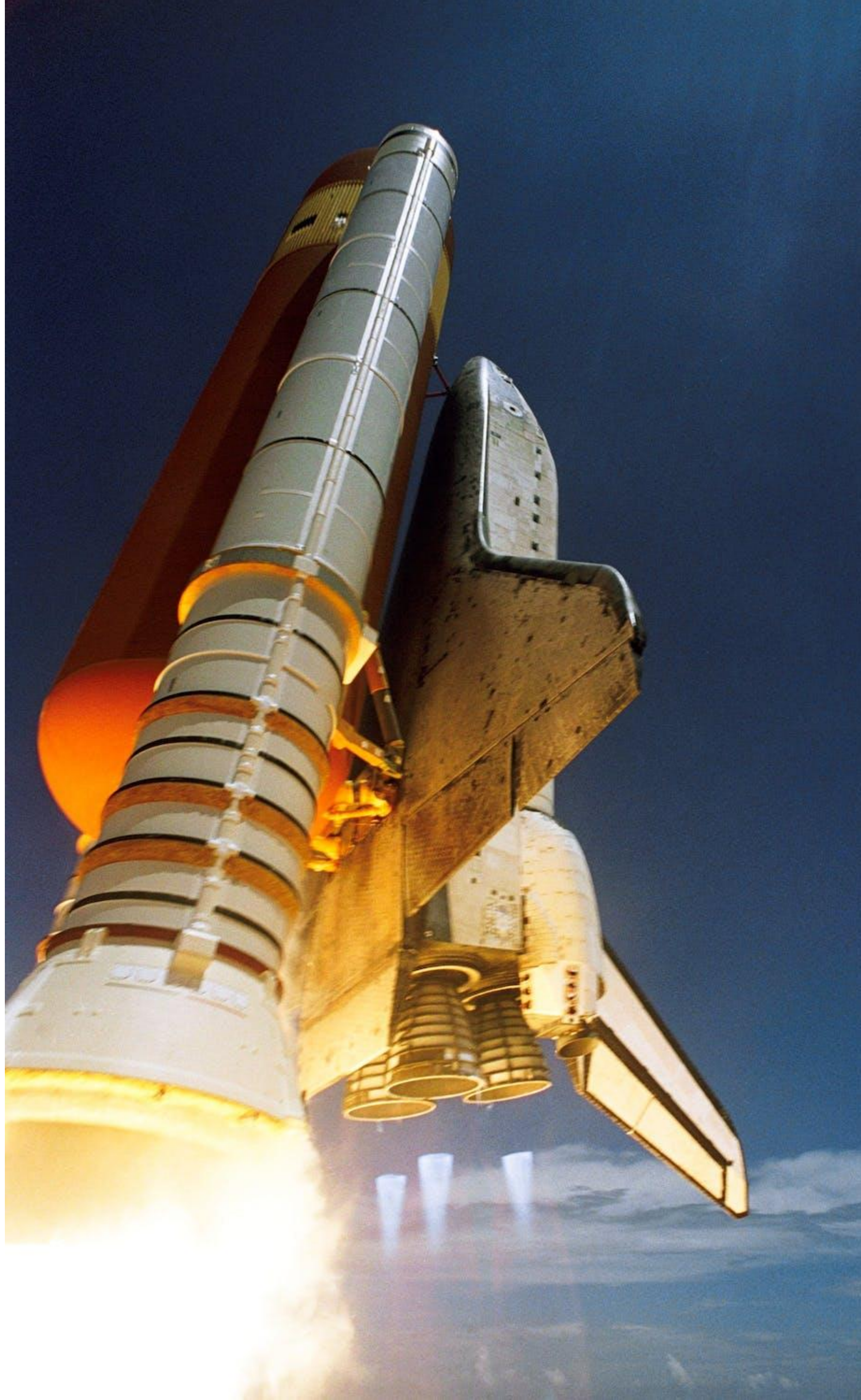


EXECUTIVE SUMMARY

Summary of Methodologies

The research undertaken attempts to identify the factors for successful first-stage rocket landing. To make this determination, the following methodologies are used:

- **Collect** data using SpaceX REST API and Web Scraping Techniques
- **Wrangle** and format data to create a categorical outcome variable.
- **Explore** the data with visualization techniques and interactive dashboards.
- **Analyze** the data with SQL querying, calculating insightful statistics such as total payload, payload range for successful launches, launch site and yearly trends.
- **Build Models** to predict landing outcomes using Logistic Regression, Support Vector Machines (SVM), and K-Nearest-Neighbor (KNN).



EXECUTIVE SUMMARY

Results

Exploratory Data Analysis:

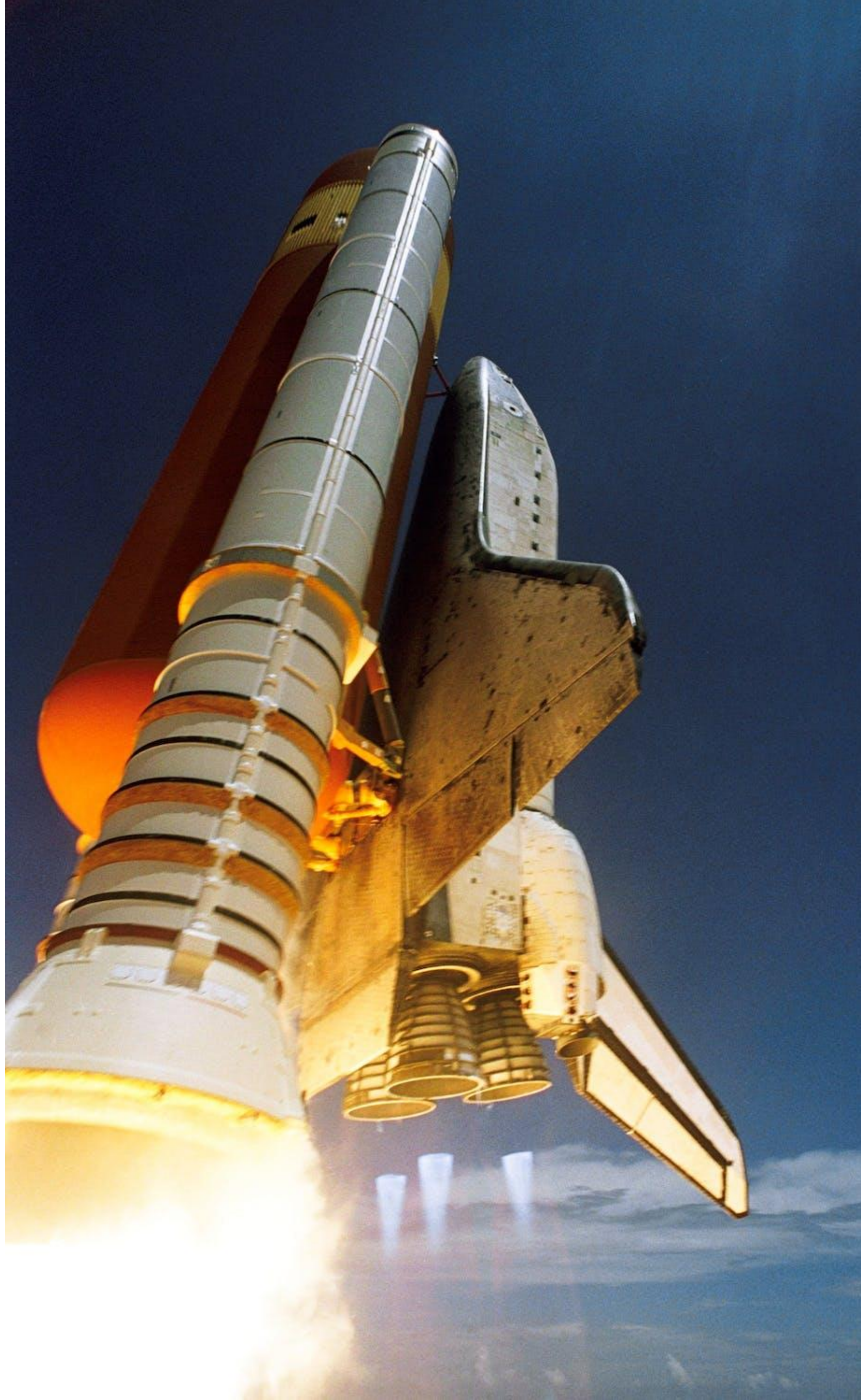
- Launch Success has improved over time
- KSC-LC-39A has the highest success rate among landing sites
- Orbits ES-L1, GEO, HEO, and SSO have a 100% success rate

Visualization/Analytics:

- Most launch sites are near the equator, and all are close to the coast

Predictive Analytics:

- All models performed similarly on the test set; the decision tree model slightly outperformed



INTRODUCTION

- SpaceX, a leading space industry, strives to make space travel inexpensive and accessible to the common person.
- Its accomplishments include sending spacecraft to the International Space Station (ISS), Launching the internet access satellite connection '*Starlink*', and sending manned missions to space.
- SpaceX has inexpensive launches due to their *novel reuse* of the first stage of their Falcon 9 rocket. The total costs estimate to around $62 million each.
- By determining if a rockets first stage will land, we can predict the price of the launch, by analyzing public data and applying machine learning models to predict whether SpaceX – or any competing company – can reuse their first stage (and effectively outbid them).

INTRODUCTION

The determining process will involve analysis and determination of several statistics related to the launches, including:

- Identifying what features affect the successful landing of the rocket
- The interaction between these factors
- The optimal operating conditions required to ensure a successful landing program

Methodology

Data Collection - API

- Request data from SpaceX API (rocket launch data)
- Decode response using .json() and convert to a dataframe using .json_normalize()
- Request information about the launches from SpaceX API using custom functions
- Create dictionary from the data
- Create dataframe from the dictionary
- Filter dataframe to contain only Falcon 9 launches
- Replace missing values of Payload Mass with calculated .mean()
- Export data to csv file

```python
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

✓ 0.0s Python

```python
response = requests.get(spacex_url)
```

✓ 1.0s Python

```python
# Use json_normalize meethod to convert the json result into
data = pd.json_normalize(response.json())
```

✓ 0.0s Python

```python
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}
```

```python
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = launchdf[
    launchdf['BoosterVersion'] != 'Falcon 1']
data_falcon9
```

Python

```python
# Calculate the mean value of PayloadMass column
PayloadMassMean = data_falcon9['PayloadMass'].mean()

# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(np.nan, PayloadMassMean)
```

Data Collection – Web Scraping

- Request data (Falcon 9 launch data) from Wikipedia
- Create BeautifulSoup object from HTML response
- Extract column names from HTML table header
- Collect data from parsing HTML tables
- Create dictionary from the data
- Create dataframe from the dictionary
- Export data to csv file

```python
# use requests.get() method with the provided static
# assign the response to a object
response = requests.get(static_url).text
```

Python

```python
# Use BeautifulSoup() to create a BeautifulSoup obje
soup = BeautifulSoup(response, "html.parser")
```

Python

```python
column_names = []

# Apply find_all() function with `th` element on fi
# Iterate each th element and apply the provided ext
# Append the Non-empty column name (`if name is not

table_elements = first_launch_table.find_all('th')

for th in table_elements:
    name = extract_column_from_header(th)
    if (name is not None and len(name) > 0):
        column_names.append(name)
```

Python

```python
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']
```

```python
df= pd.DataFrame({
    key:pd.Series(value) for key, value
     in launch_dict.items() })
df
```

Data Wrangling

Steps

- Perform EDA and determine the data labels
- Calculate:
  - # of launches for each site
  - # and occurrence of orbits
  - # and occurrence of mission outcome per orbit type
- Create a binary variable '**class**' that will represent the landing outcome.
- Outcomes converted into 1 for a successful landing and 0 for unsuccessful landing

EDA with Visualization

Charts Created

- Flight Number vs. Payload
- Flight Number vs. Launch Site
- Payload Mass (kg) vs. Launch Site
- Payload Mass (kg) vs. Orbit Type
- Landing Outcome (‘class’) yearly trend

Analysis

- **View relationship** between features using scatter plots. This is useful for developing classification models
- **Show comparisons** using bar charts.

EDA with SQL

- The SpaceX dataset is loaded into a PostgreSQL database
- The data is queried to get insights from the data:
  - Names of unique launch sites in the mission
  - Total Payload Mass carried by boosters launched by NASA (CRS)
  - Average Payload Mass carried by Booster Version F9 v1.1
  - Total number of successful and failure mission outcomes
  - Booster Version and launch site names of failed landing outcomes in drone ship

Map with Folium

Markers indicating Launch Sites

- Added a yellow circle around **NASA JSC** with a popup label.
- Added red circles around launch sites with a popup label.

Colored Markers of Launch Outcomes

- Added colored markers of **successful** and **failure** launches at each launch site to display which site has high success rates.

Distance between a Launch Site to proximities

- Added **colored lines** to show **distance between** Launch Site **CCAFS SLC-40** to **nearest coastline, railway, highway, and city**.

Dashboard with Plotly

Dropdown List with Launch Sites

- Allow users to select all launch sites or a specific one.

Pie chart showing successful launches

- Allow user to see successful and unsuccessful launches as a percent of total

Slider of Payload Mass Range

- Allow user to select payload mass range

Scatter chart showing Payload Mass vs. Success Rate by Booster Version

- Allow user to see correlation between payload mass and launch success.

Predictive Analytics

Steps Taken

- **Create** NumPy array from ‘class’ column
- **Standardize** the data with *StandardScalar*. Fit and transform the data
- **Split** the data using *train_test_split*
- **Create** and **Apply** a *GridSearchCV* object with cv=10 (number of folds) for parameter optimization, on the following models:
  - Logistic Regression
  - Support Vector Machine
  - Decision Tree
  - K-Nearest Neighbor (KNN)

Predictive Analytics

Steps Taken

- **Calculate** accuracy on the test data using various score methods for all models
- **Assess** the *Confusion Matrix* for all models
- **Identify** the best model using *Jaccard_Score*, *F1_Score,* and accuracy.


RESULTS
NASA

RESULTS SUMMARY

EXPLORATORY DATA ANALYSIS

- Launch Success has improved over time
- KSC LC-39A has highest success rate among landing sites
- Orbits ES-L1, GEO, HEO, and SSO have 100% success rate

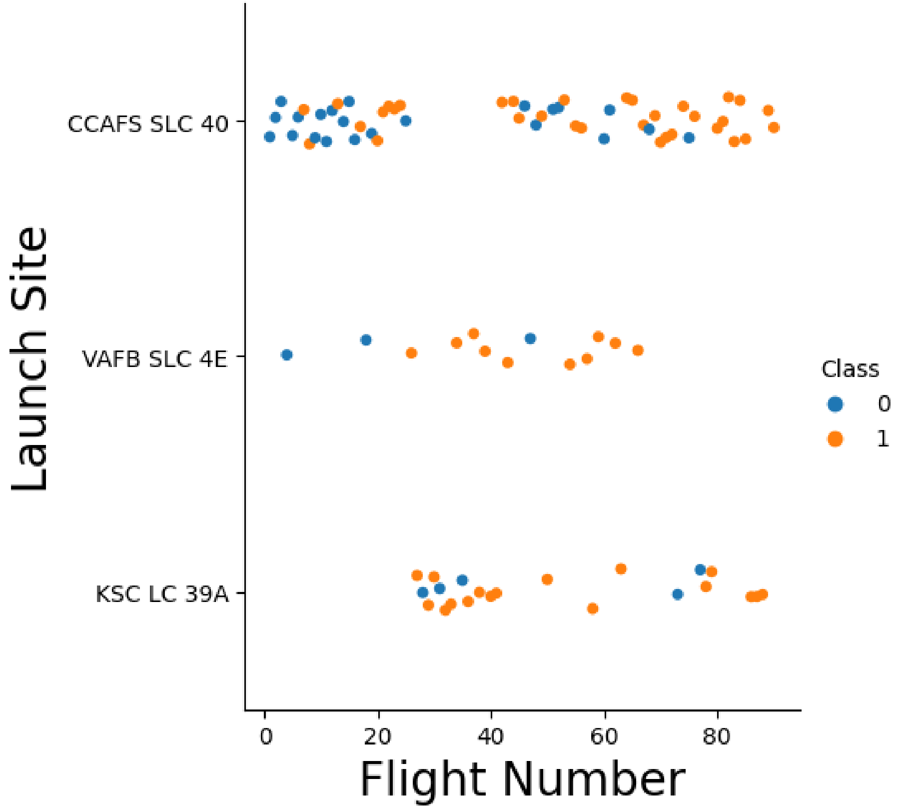
VISUAL ANALYTICS

- Most launch sites are near the equator, and all are near the coastline
- Launch sites are far enough away to prevent infrastructural damage due to a failed launch, while still close enough to optimize transport and travel costs

Flight Number vs. Launch Site

EDA RESULTS:

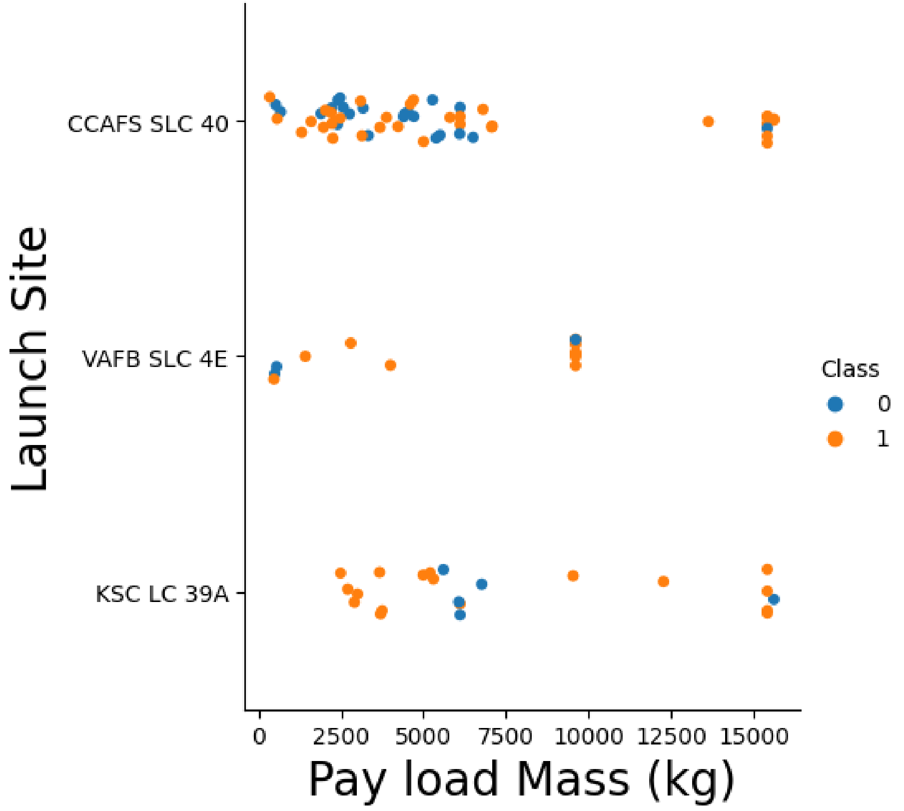
- **Earlier Flights had a lower success rate** (blue = fail)
- **Later Flights had a higher success rate** (orange = success)
- Around **half** of the launches were from CCAFS SLC-40 launch site
- VAFB SLC 4E and KSC LC 39A have higher success rates
- We can infer that new launches have a higher success rate



Payload vs. Launch Site

EDA RESULTS:

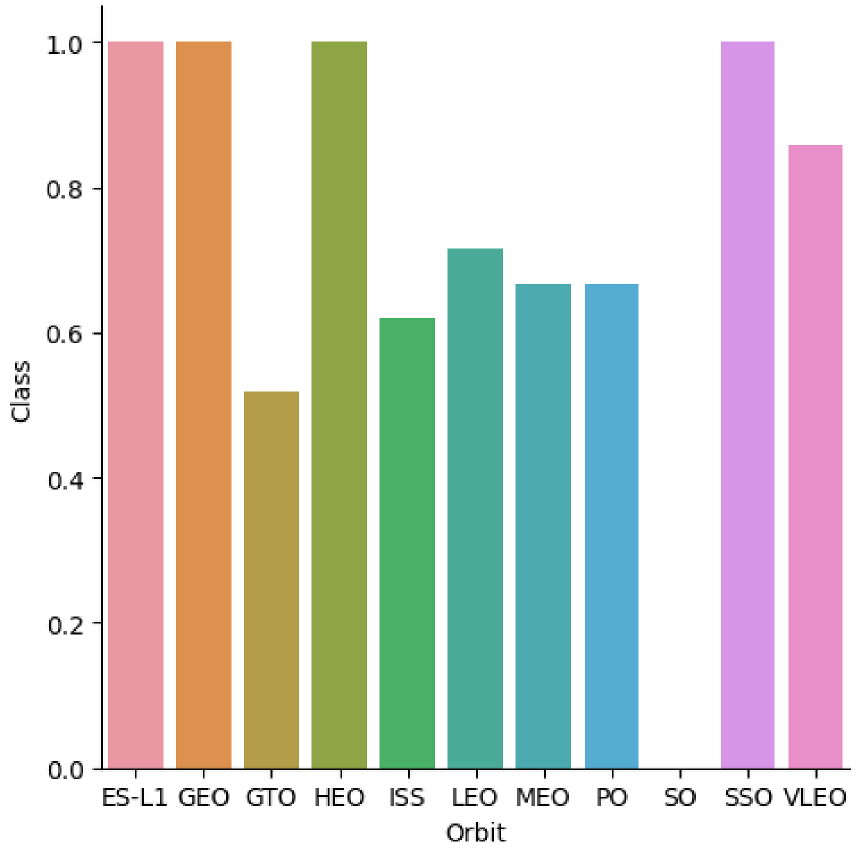
- Typically, the **higher the payload mass** (kg), higher the **success rate**
- Most launches with a payload **greater than 7,000 kg** were **successful**
- **KSC LC 39A has a 100% success rate** for launches **less than 5,500 kg**
- VAFB SKC 4E has **not launched** anything **greater than ~10,000 kg**



Success Rate by Orbit

EDA RESULTS:

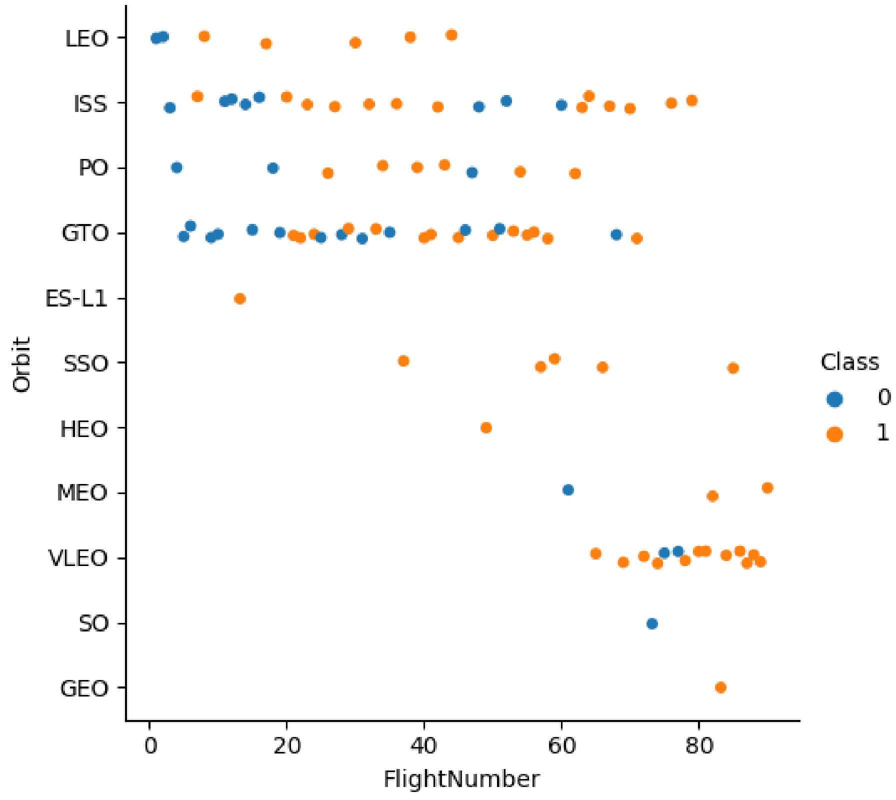
- 100% Success Rate: ES-L1, GEO, HEO, SSO
- 50-80% Success Rate: GTO, ISS, LEO, MEO, PEO
- 0% Success Rate: SO



Flight Number vs. Orbit

EDA RESULTS:

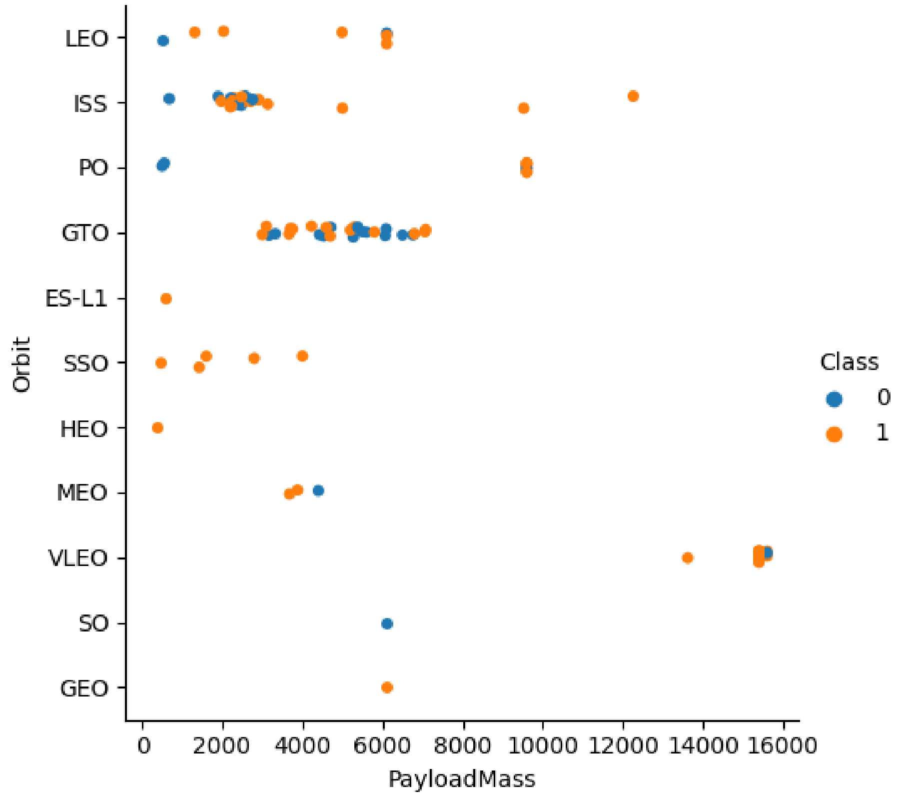
- **The success rate** typically **increases** with the number of flights for each orbit
- This relationship is highly apparent for the **LEO orbit**
- The **GTO orbit**, however, does not follow this trend



Payload vs. Orbit

EDA RESULTS:

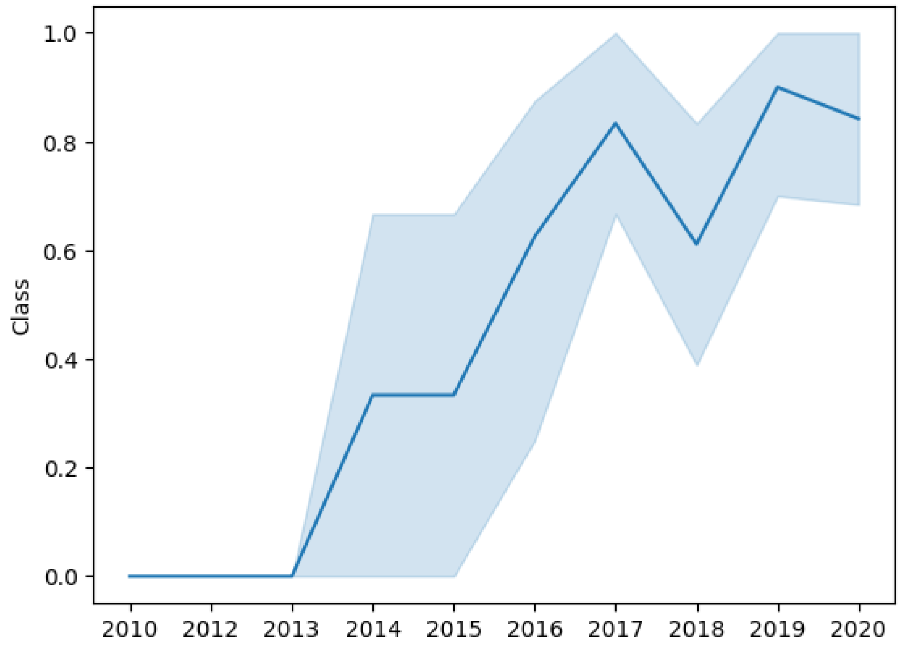
- Heavy payloads are better with LEO, ISS and PO orbits
- The GTO orbit has mixed success with heavier payloads



Launch Success over Time

EDA RESULTS:

- The success rate improved from 2013-2017 and 2018-2019
- The success rate decreased from 2017-2018 and from 2019-2020
- Overall, the success rate has improved since 2013



Launch Site Information

Records with Launch Site name starting with 'CCA'

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

Launch Site Names:

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

Payload Mass

Total Payload Mass:

- **45,596 kg** (total) carried by boosters launched by NASA (CRS)

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Customer == "NASA (CRS)"
```

* sqlite:///my_data1.db
Done.

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass:

2,928 kg (Average) carried by Booster Version F9 v1.1

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTABLE
WHERE Booster_Version = "F9 v1.1"
```

* sqlite:///my_data1.db
Done.

AVG(PAYLOAD_MASS__KG_)
2928.4

Landing and Mission Info

1st Successful Landing in Ground Pad:

- 12/22/2015

```
%%sql

SELECT Date
FROM SPACEXTABLE
WHERE Landing_Outcome = "Success (ground pad)"
LIMIT 1
```

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

Date
2015-12-22

Booster Drone Ship Landing:

- Booster mass greater than 4,000 but less than 6,000

```
%%sql
SELECT payload
FROM SPACEXTABLE
WHERE
    Landing_Outcome = "Success (drone ship)"
    AND
    PAYLOAD_MASS__KG_ > 4000
    AND
    PAYLOAD_MASS__KG_ < 6000
```

Payload
JCSAT-14
JCSAT-16
SES-10
SES-11 / EchoStar 105

Total number of Successful and Failed Mission Outcomes:

- 1 Failure in Flight
- 99 Success
- 1 Success (Payload Status Unclear

```
%%sql

SELECT Mission_Outcome, COUNT(*) as Total_Count
FROM SPACEXTABLE
GROUP BY Mission_Outcome
```

Boosters

Carrying Max Payload:

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

Query:

```
%%sql

SELECT Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ = (
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTABLE
)
```

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

Failed Landings on Drone Ship

In 2015:

Month	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

```
%%sql

SELECT substr(Date, 6, 2) AS Month, Booster_Version, Launch_Site
FROM SPACEXTABLE
WHERE
    Date LIKE "2015%"
    AND
    Landing_Outcome LIKE "Failure%"
```

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

Count of Successful Landings

Between 2010-06-04 and 2017-03-20 in descending order:

Landing_Outcome	TOTAL_COUNT
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

```
%%sql

SELECT Landing_Outcome, COUNT(*) AS TOTAL_COUNT
FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY TOTAL_COUNT DESC
```

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


Launch site
Analysis

Launch Sites

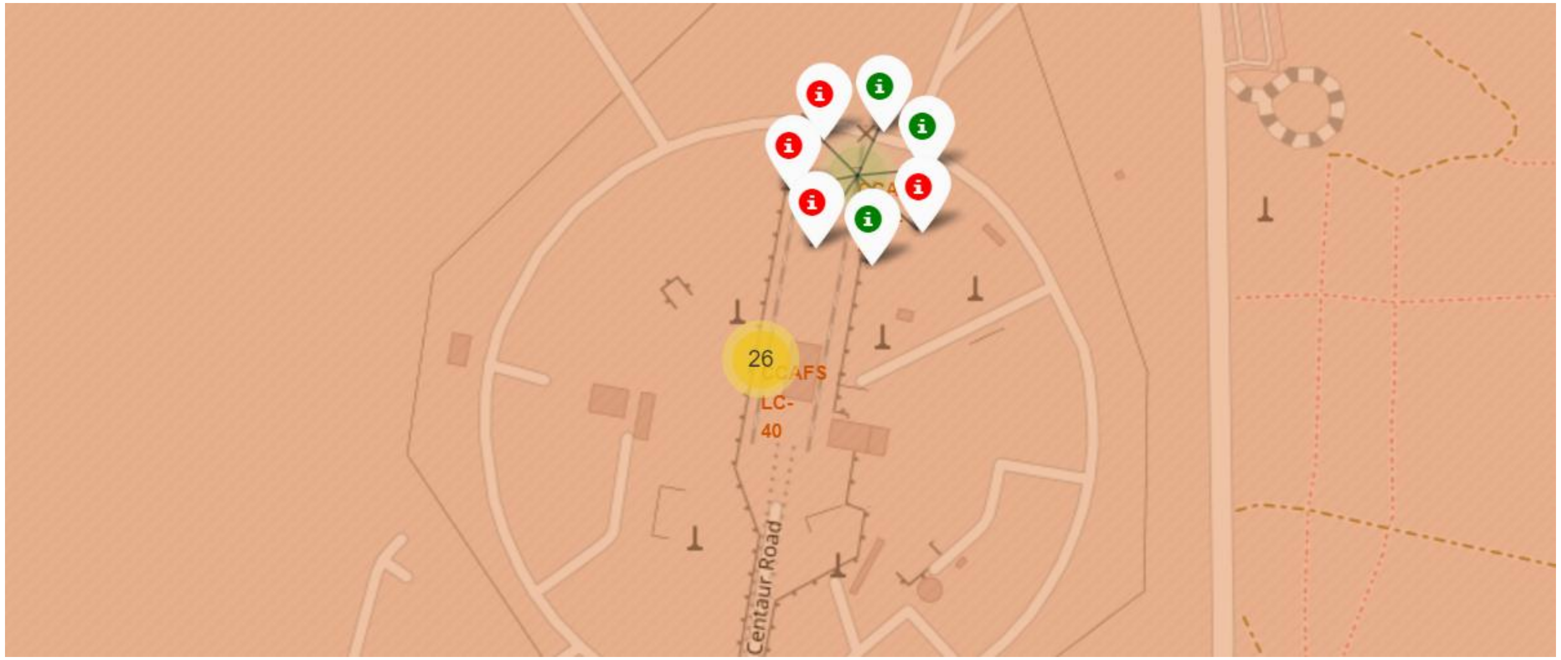
With Markers

- **Near Equator:** the closer the launch site to the equator, the easier it is to launch to equatorial orbit. Rockets launched from this region get a **Natural Boost** that helps **save costs**.

Launch Outcomes

At each Launch site:

- **Green** markers for Successful launches, **Red** for unsuccessful launches.
- Launch site **CCAFS LSC-40** has a **3/7 success rate (42.9%)**



Distance to Proximities

For CCAFS LSC-40:

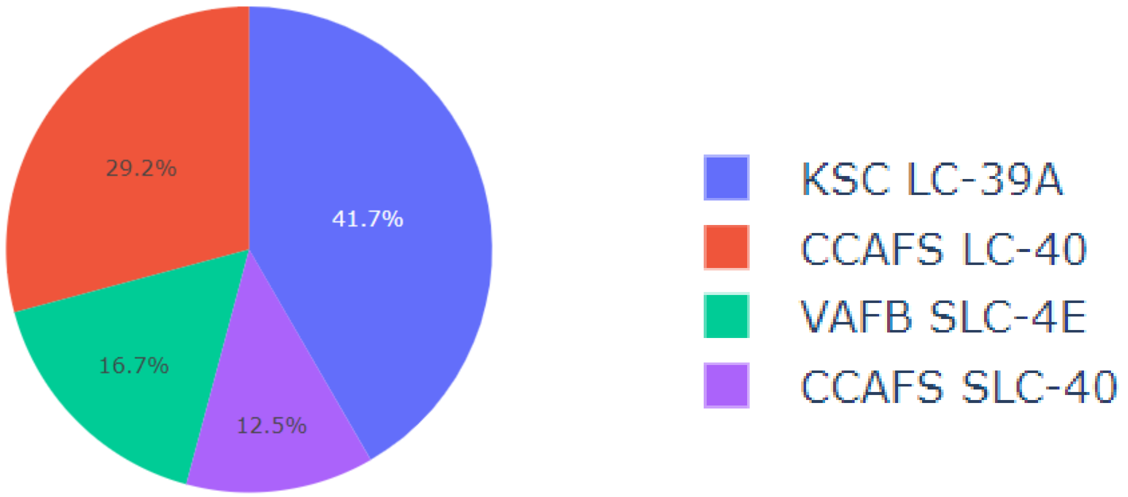
- **.86 km** from nearest Coastline
- **21.96 km** from nearest railway
- **23.23 km** from nearest city
- **26.88 km** from nearest highway


Dashboard

Launch success

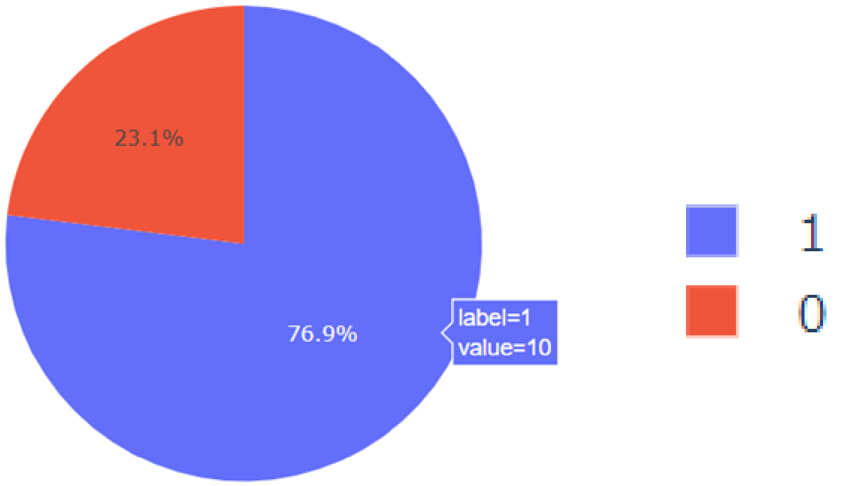
As percent of total:

- KSC LC-39A has the **most successful launches** amongst launch sites (**41.2%**)



As percent of total:

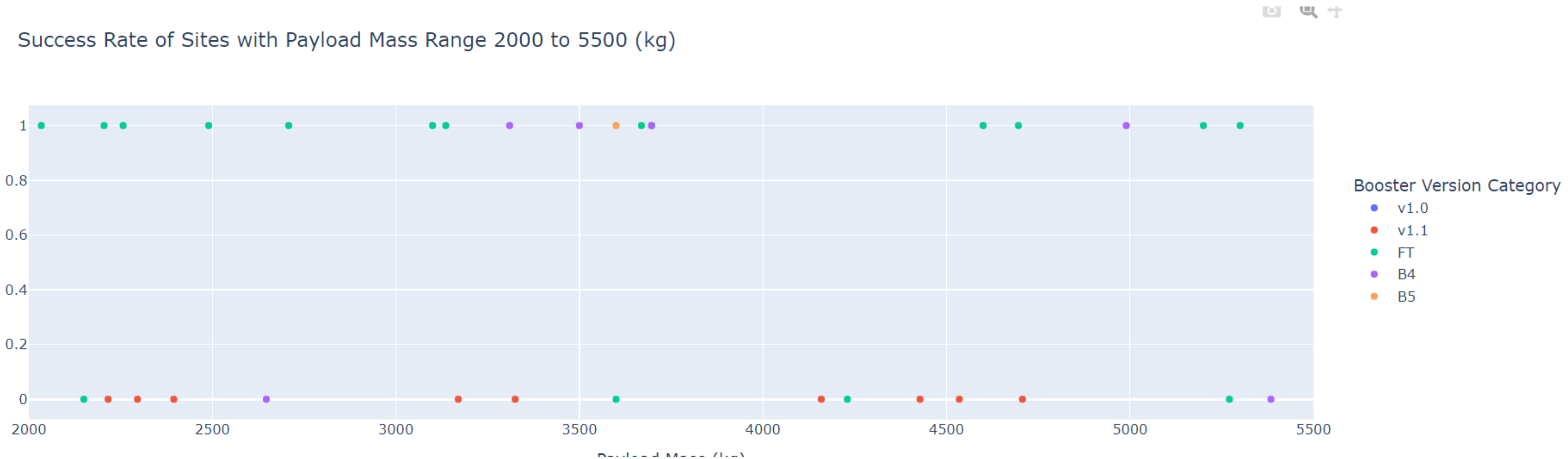
- **KSC LC-39A** has the highest **Success Rate** amongst launch sites (**76.9%**), with **10** successful launches and **3** failed ones.



Payload Mass and Success

By Booster Version:

- Payloads between **2,000 kg** and **5,000 kg** have the **highest success rate**
- Here, 1 indicates successful outcome and 0 indicates a failed one.



Predictive Analytics

Classification

Accuracy:

- **All** the **Models** performed at about the same level and had the same scores and **accuracy**. This is likely due to the **small size** of the dataset. The **Decision Tree** model slightly outperformed the rest.

- **.best_score_** is the average of all **cv folds** for a single combination of the parameters.

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.800000	0.800000	0.800000	0.800000
F1_Score	0.888889	0.888889	0.888889	0.888889
Accuracy	0.833333	0.833333	0.833333	0.833333

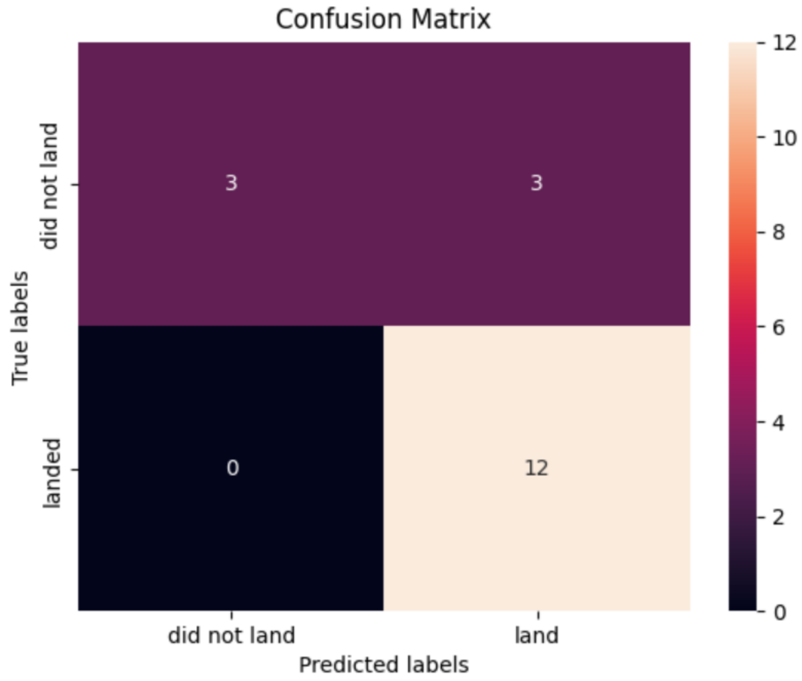
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.9017857142857144
Best params is : {'criterion': 'gini', 'max_depth': 18, 'max_feat
ures': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 2, 'sp
litter': 'best'}
```

Confusion Matrices

Performance Summary:

- A **Confusion Matrix** summarizes the performance of a classification algorithm
- All the confusion matrices were identical
- The fact that there are False Positives (Type 1 error) is not good
- Confusion Matrix Outputs:
  - 12 True Positive
  - 3 True Negative
  - 3 False Positive
  - 0 False Negative
- **Precision** = TP / (TP + FP)
  - 12 / 15 = 0.80
- **Recall** = TP / (TP + FN)
  - 12 / 12 = 1.00
- **F1_Score** = 2 * (Precision * Recall) / (Precision + Recall)
  - 2 * (0.80 + 1.00) / (0.80 + 1.00) = 0.89
- **Accuracy** = (TP + TN) / (TP + TN + FP + FN) = 0.833



CONCLUSION

Research:

- **Model Performance**: The models performed similarly on the test set with the Decision Tree model slightly outperforming
- **Equator**: Most of the launch sites are near the equator to minimize cost
- **Coast**: All the launch sites are near the coast
- **Launch Success**: Increases over Time
- **KSC LSC-39A**: Has the highest success rate among launch sites. Has a 100% success rate for launches less than 5,000 kg
- **Orbits**: ES-L1, GEO, HEO, and SSO have a 100% success rate
- **Payload Mass**: Across all launch sites, the higher the payload mass (kg), the higher the success rate

CONCLUSION

Things to Consider:

- **Dataset**: A larger dataset will help build on the predictive analytics results to help understand if the findings can be generalizable to a larger dataset
- **Feature Analysis/PCA**: Additional feature analysis or principal component analysis should be conducted to see if it can help improve accuracy