IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Thanaphon Naksri
August 1st 2025

Outline

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

Executive Summary

Methodologies

- Data Collection
- Data Wrangling
- **E**xploratory **D**ata **A**nalysis
- Map in Python w/ **Folium**
- Dashboard utilizing **Plotly Dash**

Results & Insights

- Descriptive Graphs & Queries
- Interactive Map
- Dashboard
- Prediction Model using ML

Introduction

SpaceX is the leading company in modern space industry. Their reusable rockets are proven reliable & helpful in transforming humanity's ability to access the stars.

As a Data Scientist: How can I determine/predict when the rockets can be reused?

Solution: Use publicly accessible data for building Dashboards & ML models for analyzing rocket launches

Section 1

Methodology

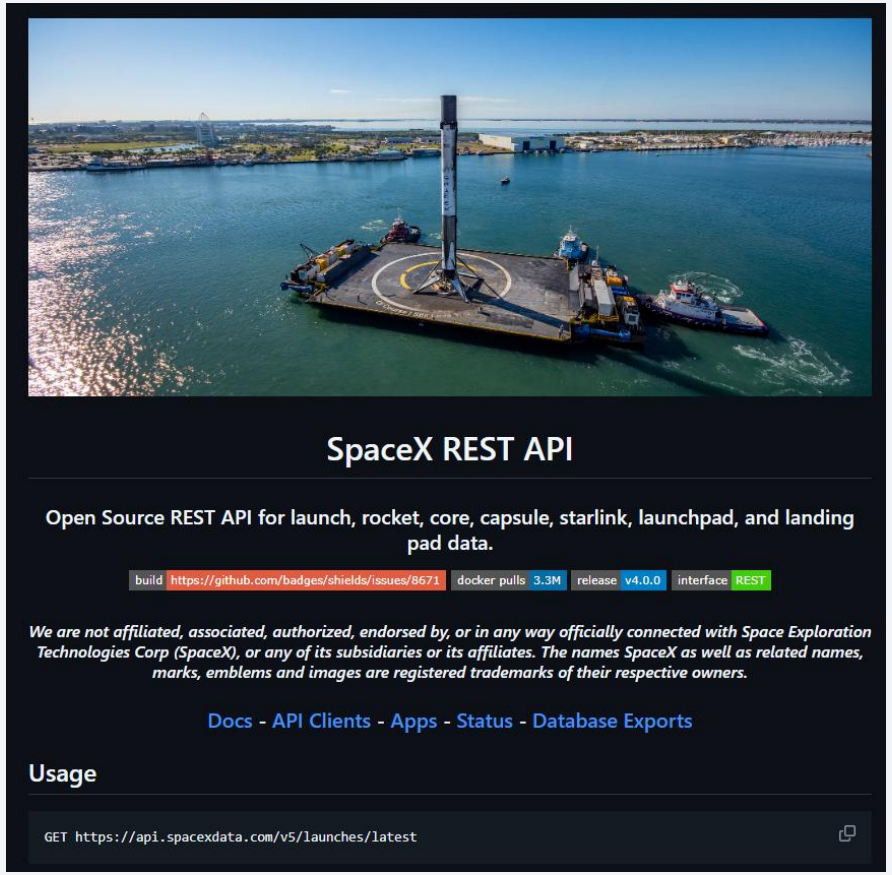
Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

REST API

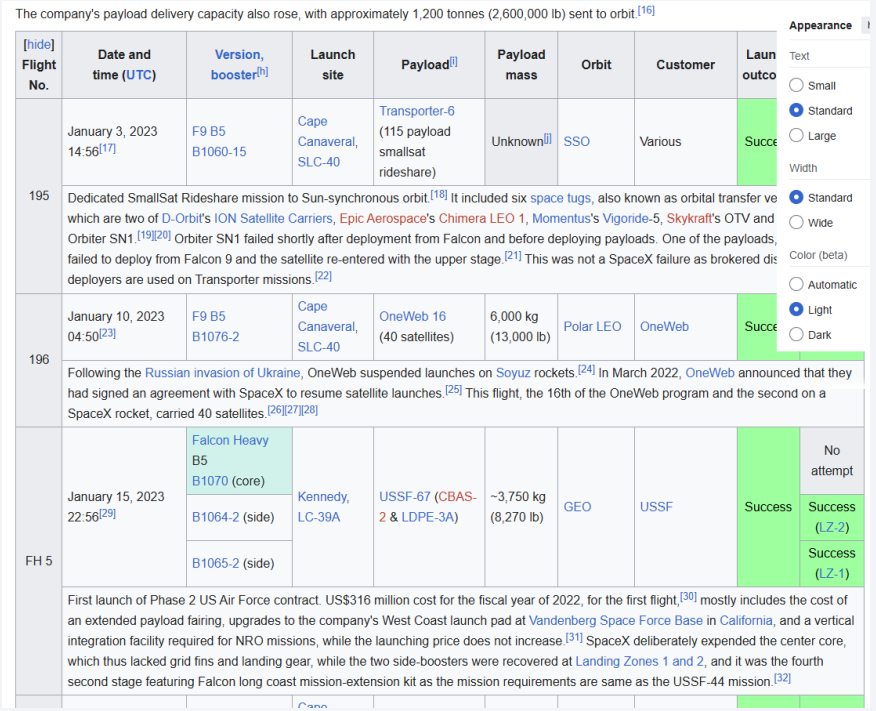


GitHub: [SpaceX-API]

Reddit: [r/SpaceX]

Wiki Scraping

Using BeautifulSoup

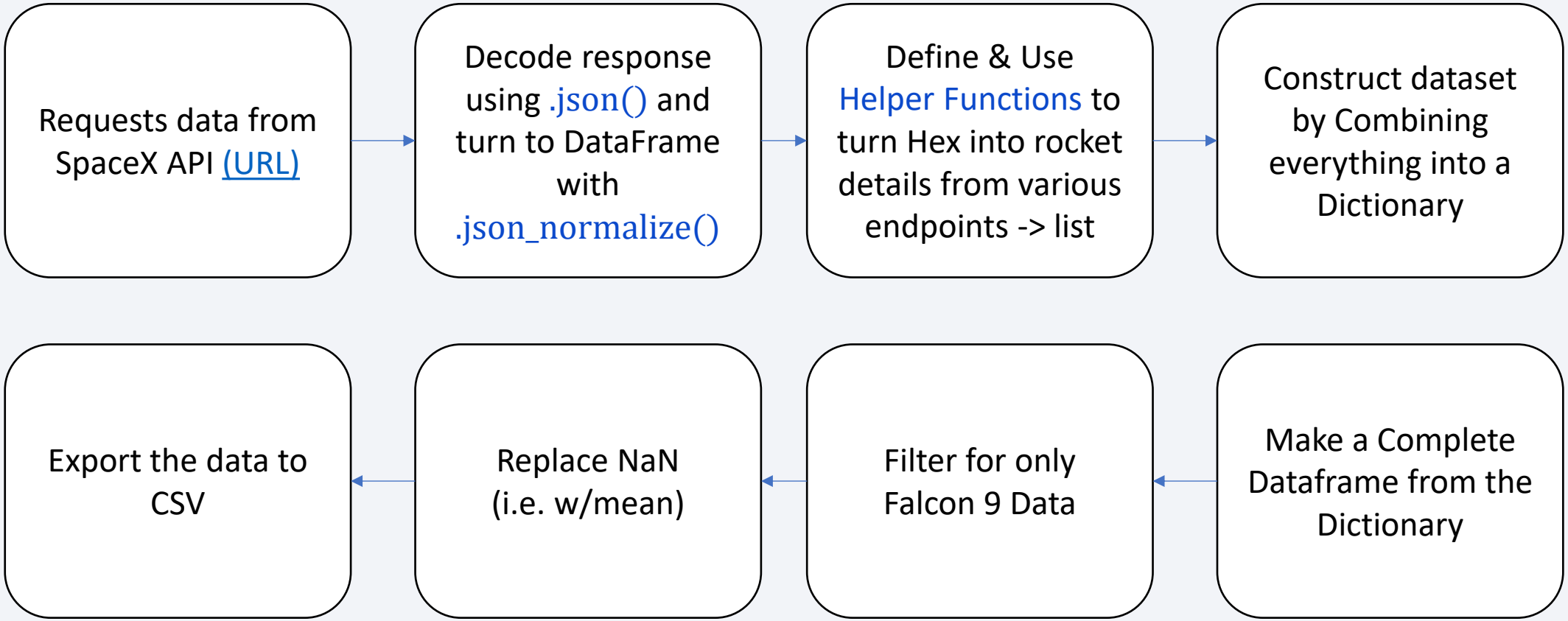

The company's payload delivery capacity also rose, with approximately 1,200 tonnes (2,600,000 lb) sent to orbit.[16]

[hide] Flight No.	Date and time (UTC)	Version, booster[h]	Launch site	Payload[i]	Payload mass	Orbit	Customer	Laun outco	
195	January 3, 2023 14:56[17]	F9 B5 B1060-15	Cape Canaveral, SLC-40	Transporter-6 (115 payload smallsat rideshare)	Unknown[j]	SSO	Various	Succe	
	Dedicated SmallSat Rideshare mission to Sun-synchronous orbit.[18] It included six space tugs, also known as orbital transfer ve which are two of D-Orbit's ION Satellite Carriers, Epic Aerospace's Chimera LEO 1, Momentus's Vigoride-5, Skykraft's OTV and Orbiter SN1.[19][20] Orbiter SN1 failed shortly after deployment from Falcon and before deploying payloads. One of the payloads, failed to deploy from Falcon 9 and the satellite re-entered with the upper stage.[21] This was not a SpaceX failure as brokered dis deployers are used on Transporter missions.[22]								
196	January 10, 2023 04:50[23]	F9 B5 B1076-2	Cape Canaveral, SLC-40	OneWeb 16 (40 satellites)	6,000 kg (13,000 lb)	Polar LEO	OneWeb	Succe	
	Following the Russian invasion of Ukraine, OneWeb suspended launches on Soyuz rockets.[24] In March 2022, OneWeb announced that they had signed an agreement with SpaceX to resume satellite launches.[25] This flight, the 16th of the OneWeb program and the second on a SpaceX rocket, carried 40 satellites.[26][27][28]								
FH 5	January 15, 2023 22:56[29]	Falcon Heavy B5 B1070 (core)	Kennedy, LC-39A	USSF-67 (CBAS-2 & LDPE-3A)	~3,750 kg (8,270 lb)	GEO	USSF	Success	No attempt
		B1064-2 (side)							Success (LZ-2)
		B1065-2 (side)							Success (LZ-1)
	First launch of Phase 2 US Air Force contract. US$316 million cost for the fiscal year of 2022, for the first flight,[30] mostly includes the cost of an extended payload fairing, upgrades to the company's West Coast launch pad at Vandenberg Space Force Base in California, and a vertical integration facility required for NRO missions, while the launching price does not increase.[31] SpaceX deliberately expended the center core, which thus lacked grid fins and landing gear, while the two side-boosters were recovered at Landing Zones 1 and 2, and it was the fourth second stage featuring Falcon long coast mission-extension kit as the mission requirements are same as the USSF-44 mission.[32]								

Appearance
Text
Small
Standard
Large
Width
Standard
Wide
Color (beta)
Automatic
Light
Dark

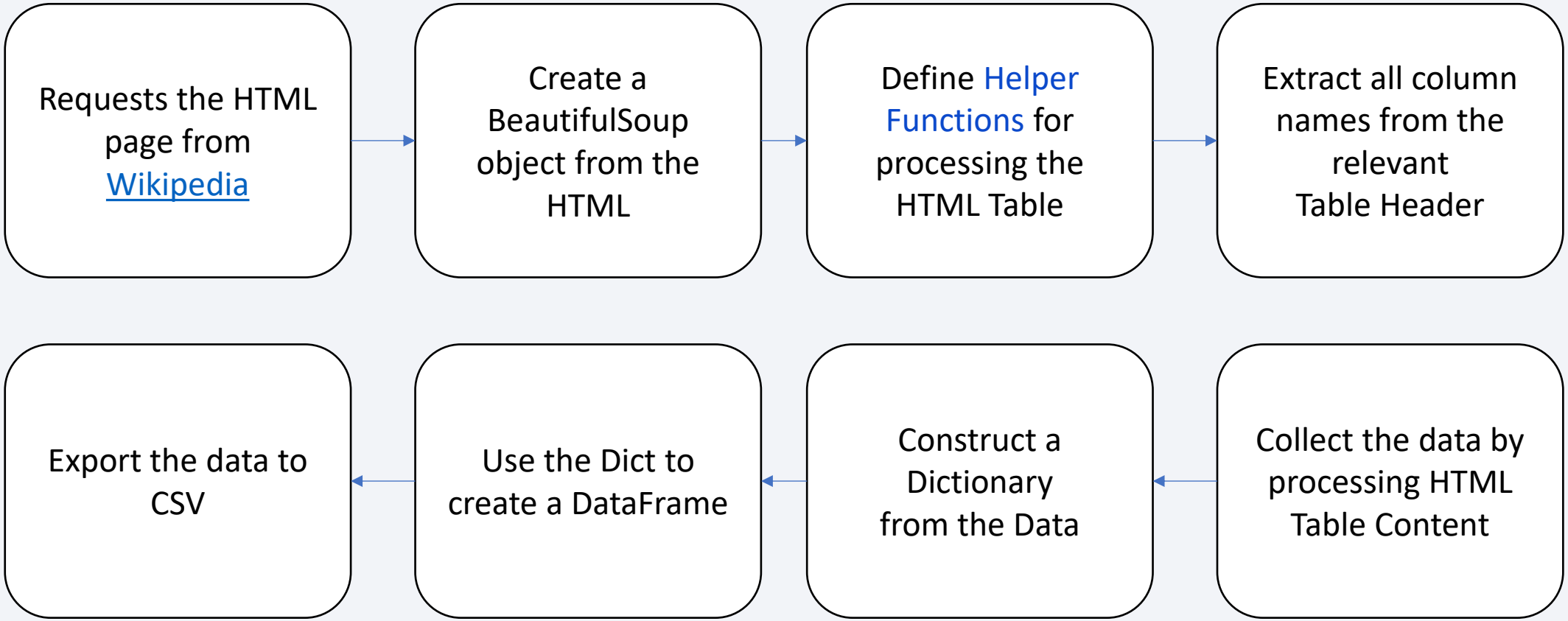
Wikipedia: [article]

Data Collection – SpaceX API



Data Collection API

Data Collection – Scraping



Data Collection Scraping

Data Wrangling

Perform simple observation on the dataset to identify key aspects

- Deal with **Null Values**
- Calculate the number of launches on each **site**
- Calculate the number and occurrence of each **orbit**
- Calculate the number and occurrence of **mission outcome per orbit type**
- **Derive mission outcome** detail into a simple Good(1) & Bad(0) Class

EDA with Data Visualization

What are we looking for?

- Correlations between each factors on the success rate of the operation
- Launch Site performances
- SpaceX’s Growth in regard to producing results

EDA with SQL

What are we looking for?

- Info on each Launch sites utilized by SpaceX in the US
- Performance Details of the Falcon 9 rocket for all operations since first developed

Build an Interactive Map with Folium

Key Functions

- **Displaying Every Launch Site in the US**
- Group **Launch Instances** into Clusters while being able to drill-through and see their Success
- **Calculate & Visualize:** Distance between a Lauch Site and its Proximities

Implementations

- **Markers with Circle, Popup, and Text**: for Displaying every **“Launch Site”**
- **Marker Cluster**: combine proximity markers into groups -> Displaying every **Launch Instances (S/F)**
- **Line**: measure & show distances between Kennedy Space Center and its proximities (i.e. coastline).

Interactive Map with Folium

Build a Dashboard with Plotly Dash

Key Functions

- **Monitoring all Launch Site associated (and used by) SpaceX**
- **Tracking** every launch instances and its associated Payload details
- **Filtering** System for Focusing on Specific Launch Site or Payload Mass Range

Implementations

- **Pie Chart for displaying**
- **Same chart** displaying total Success/Failure launches count when Selecting their respective Site
- **Drop-down & Slider** for Selecting Specific Launch Site and Payload Mass

Dashboard with Plotly-Dash

Predictive Analysis (Classification)

Feature Engineering

1. Select the **important variable** to be used in success prediction
2. There are Categorical columns? -> **One-Hot Encoding**
3. Opt. Convert all numerical values into **float** for Consistency

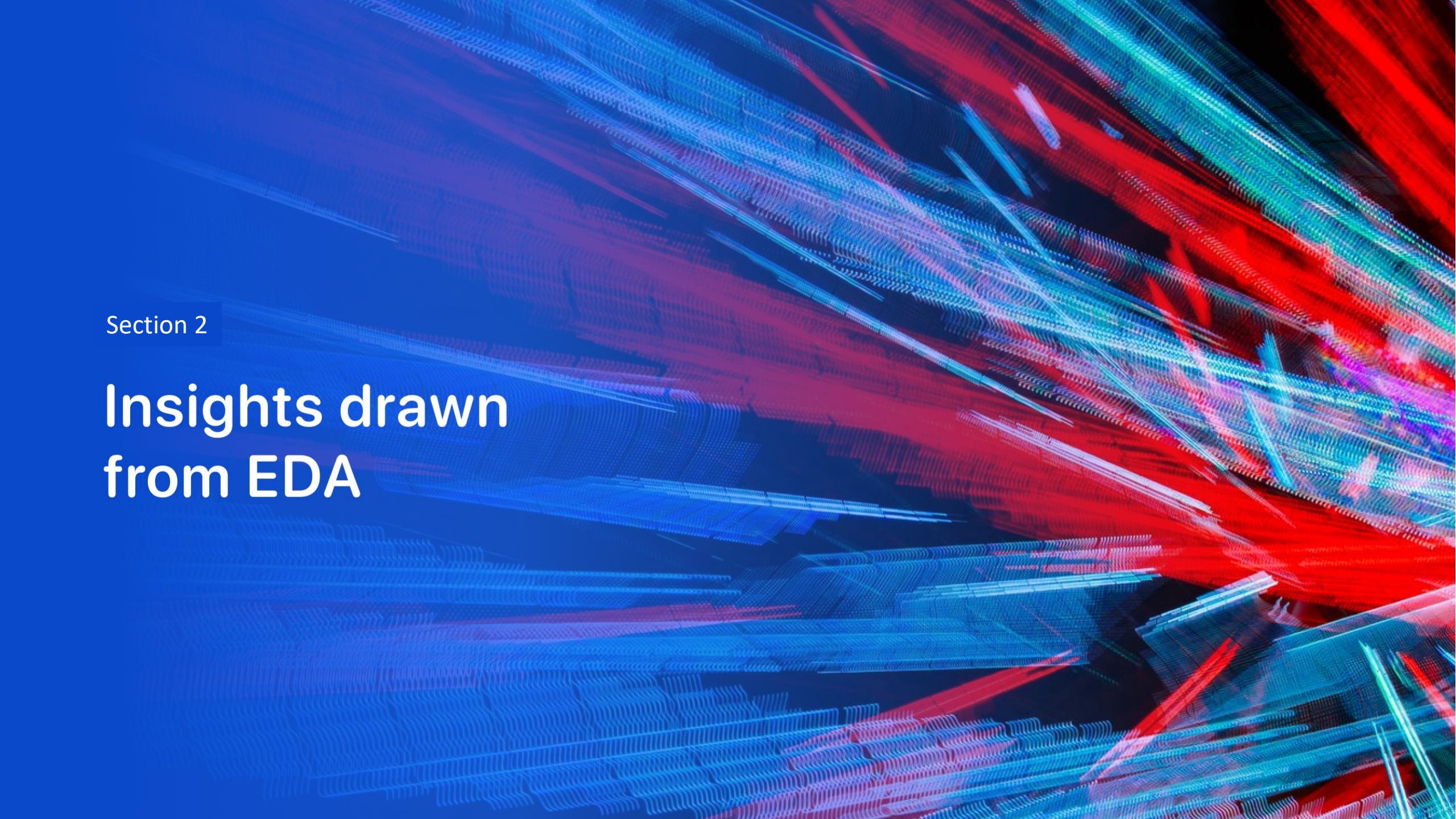
Model Development and Evaluation

1. X DataFrame for independent variables, Y for Dependent (“Success Class”)
2. Standardize the Data in X
3. Train-Test Split onto training & testing data
4. Create an ML model object then wrap with GridSearchCV to find optimal parameters
5. Generate R-Squared score and Confusion Matrix of the Model
6. Do the same for each selected ML Models
7. Compare every Models to find the most accurate one to be used

Predictive Analysis

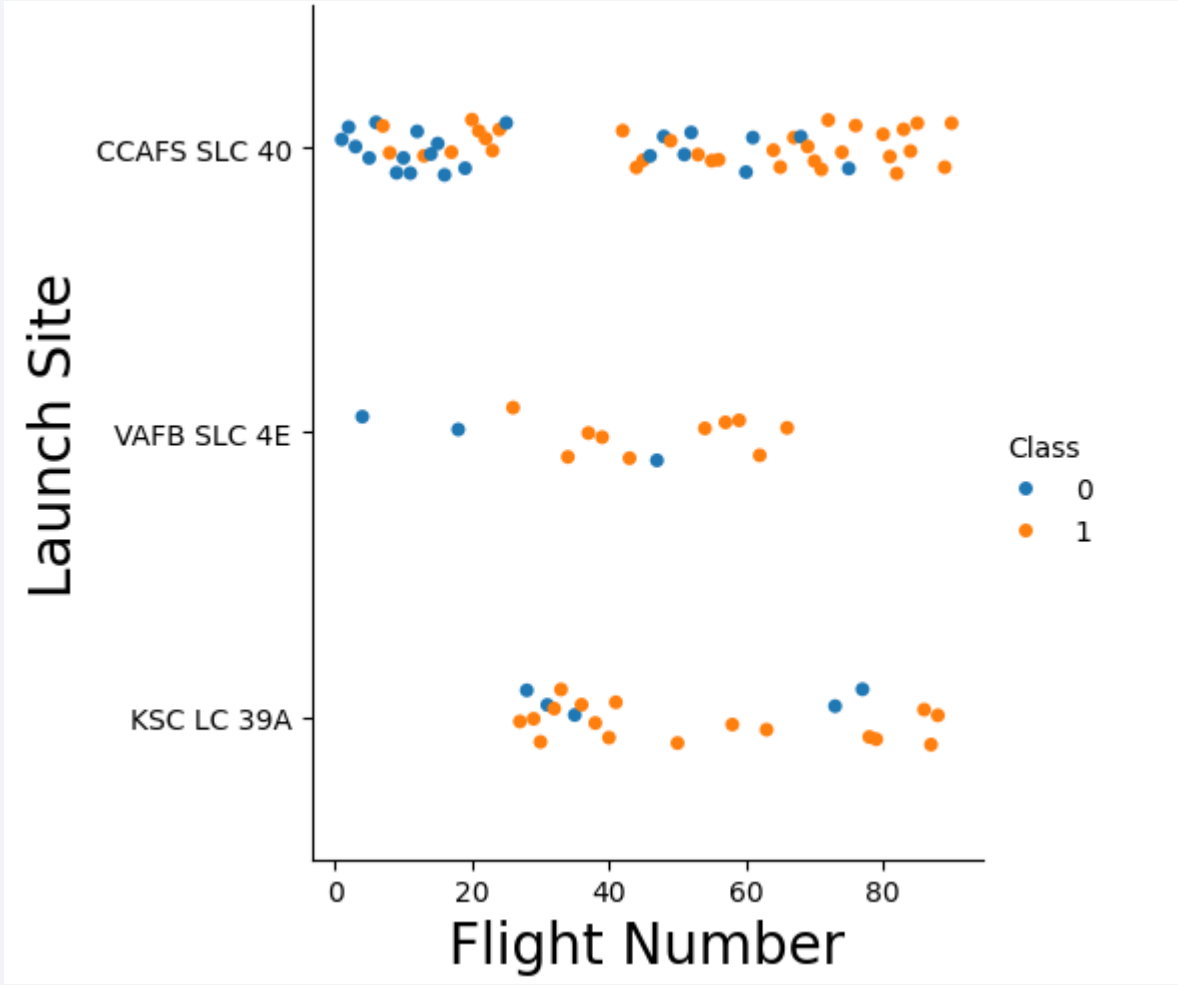
Results

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


Section 2
Insights drawn from EDA

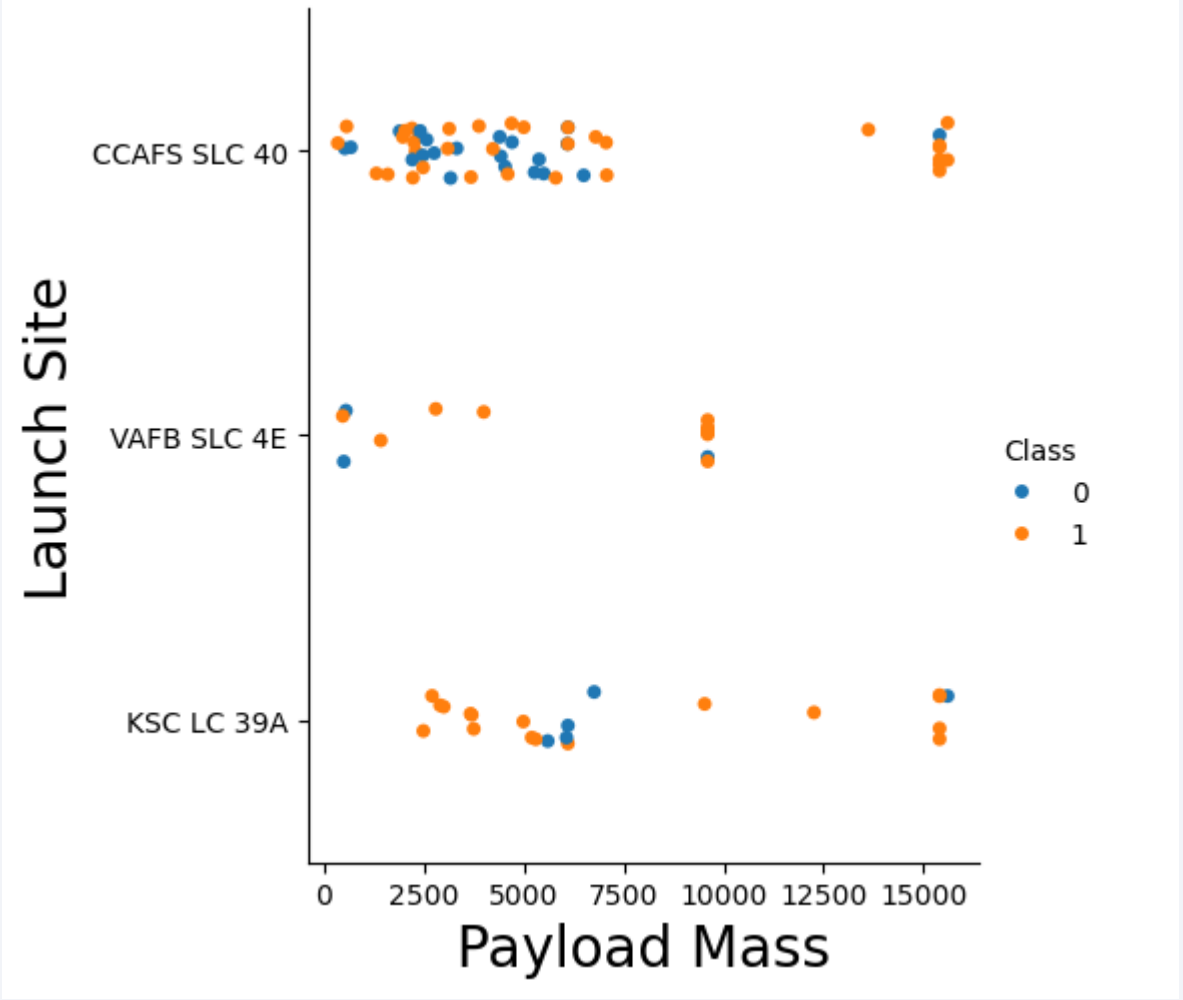
Flight Number vs. Launch Site

- Each Launch site have varying success.
- **KSC** being the most consistent



Payload vs. Launch Site

- All are capable of similar Payload Mass
- Some prefers a particular load range

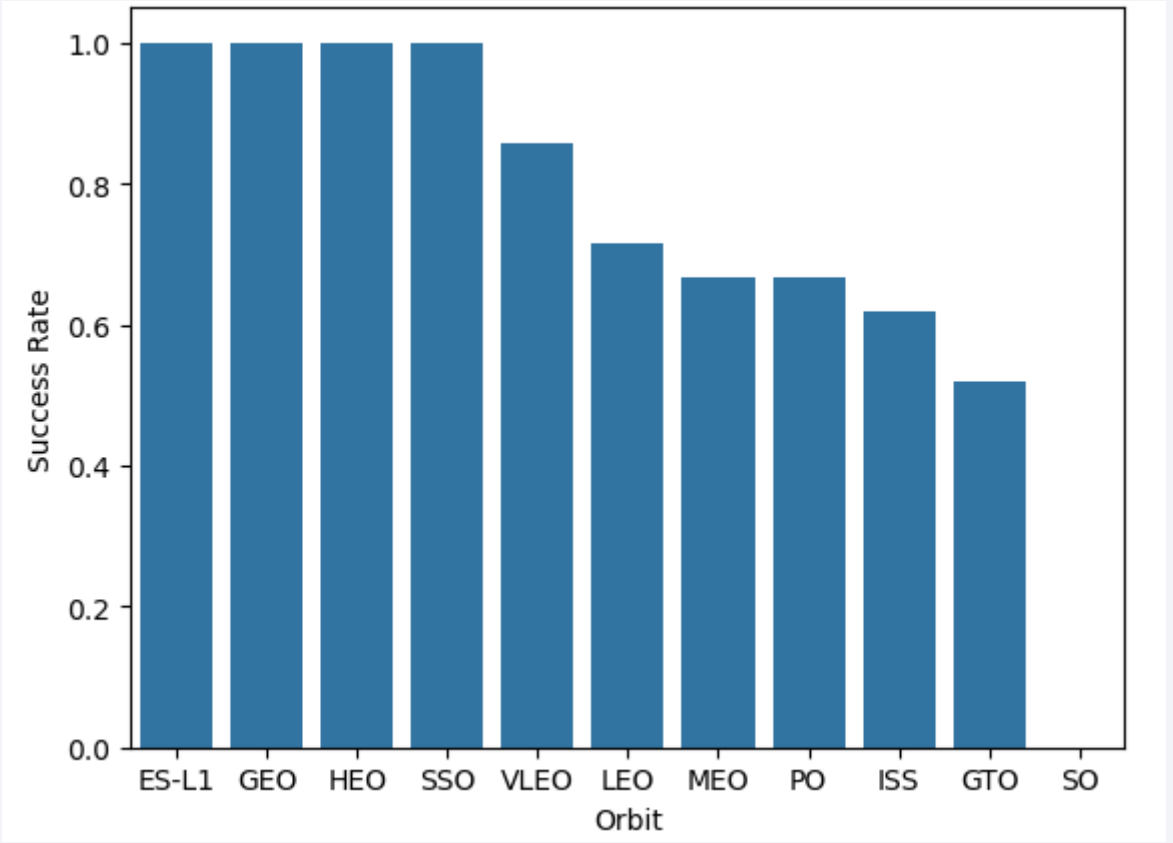


Success Rate vs. Orbit Type

- ES-L1
- GEO
- HEO
- SSO

All have **100% Success Rate**

Flights to VLEO has high Success Rate despite large attempt amount

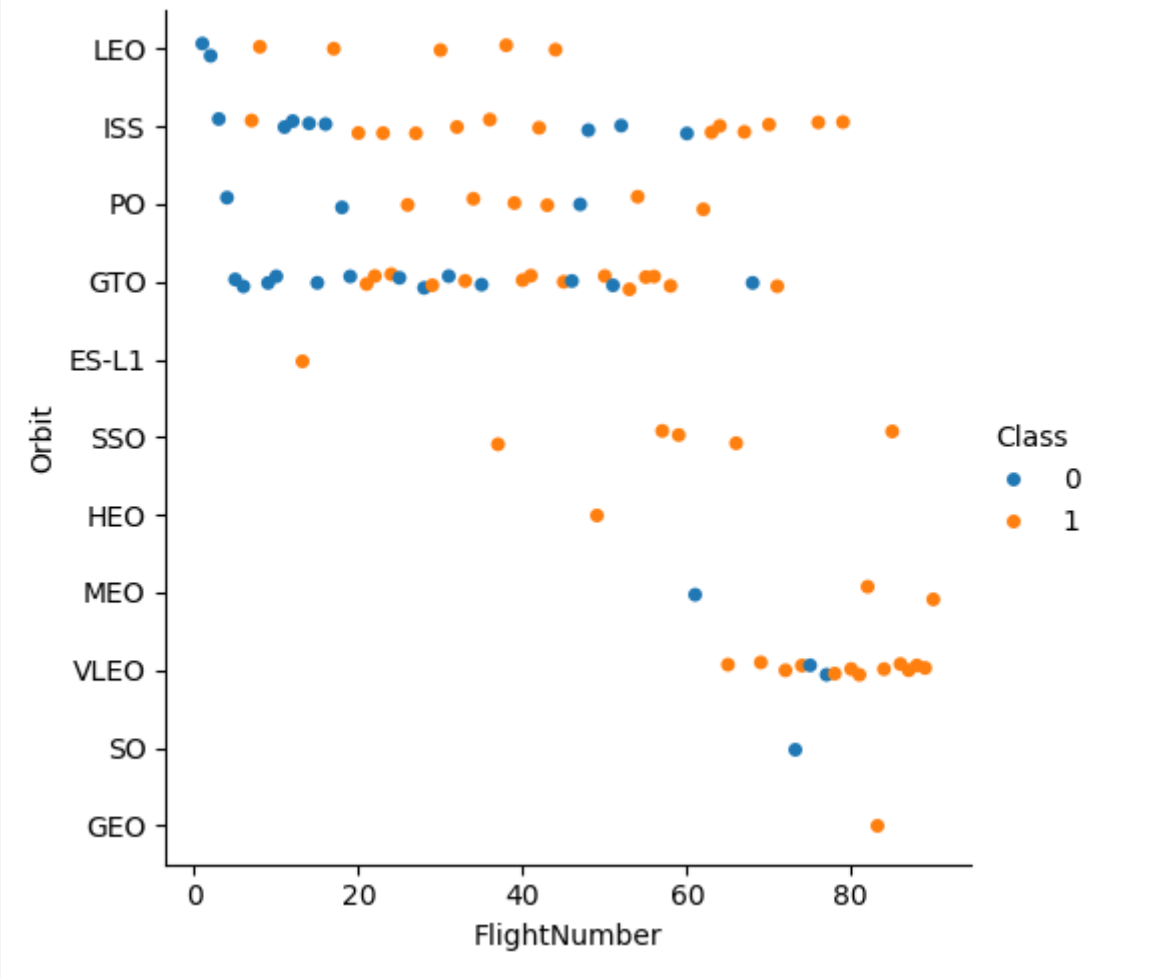


Flight Number vs. Orbit Type

Orbit Matters!

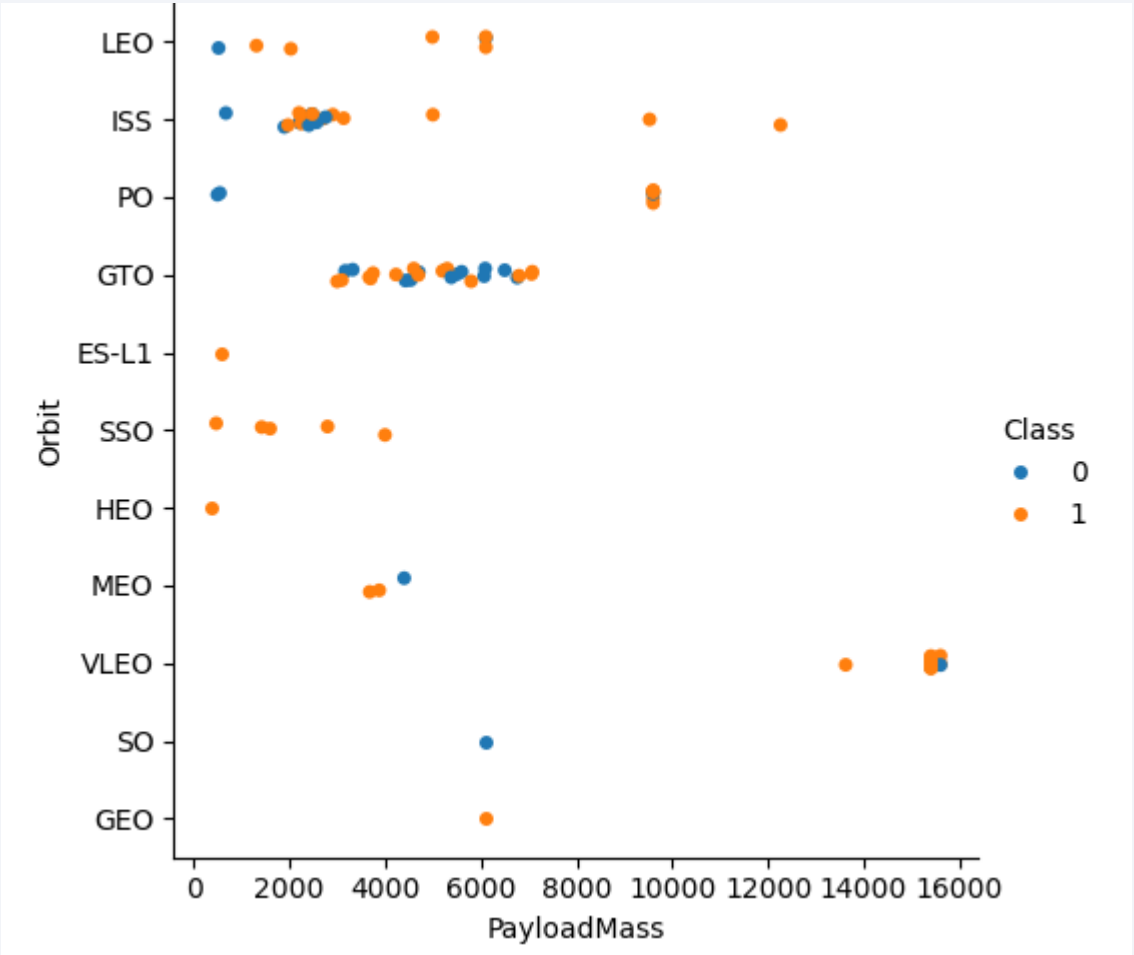
Some having a high success rate

Some produces split results



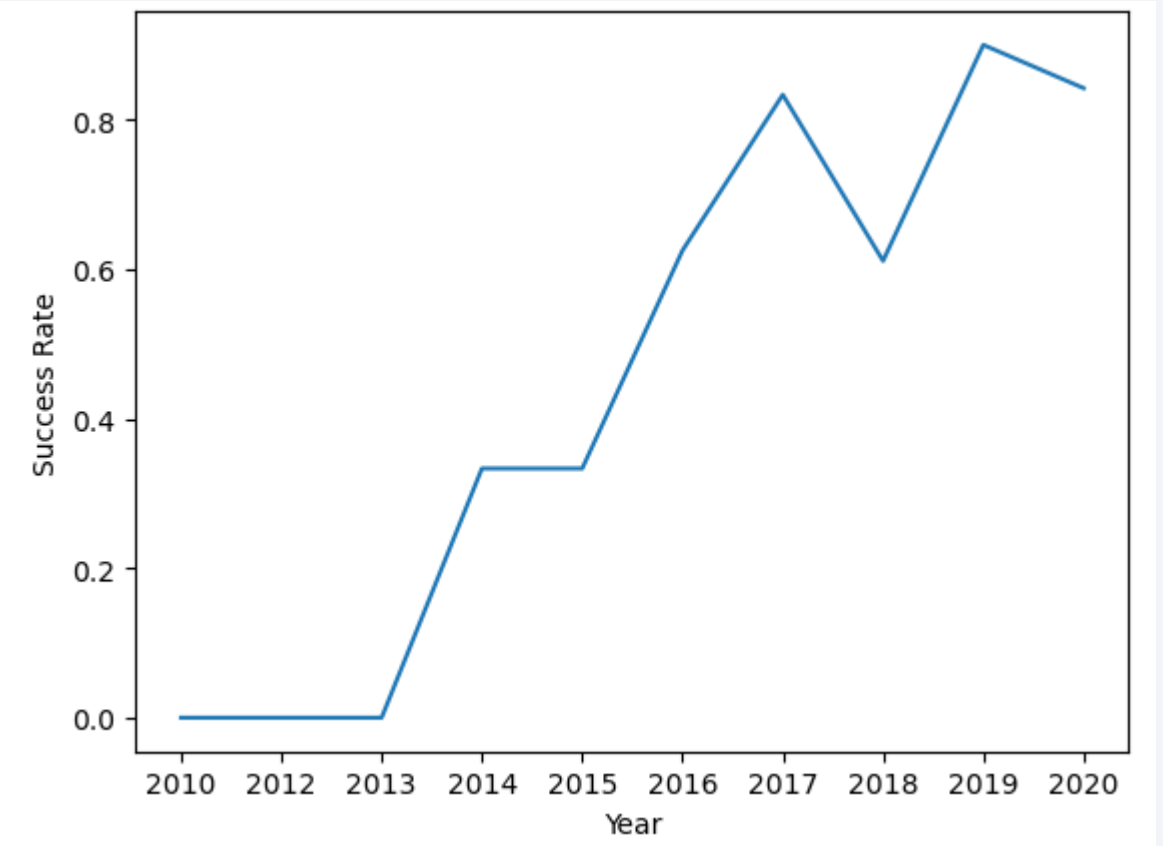
Payload vs. Orbit Type

- In **GTO** it is difficult to distinguish results
- But in most orbits, there is a **limit** to how much you can carry without failure.



Launch Success Yearly Trend

- With time, SpaceX has improved their capability to safely launch & retrieve Falcon 9 rockets.



All Launch Site Names

Display the names of the unique launch sites in the space mission

```
cur.execute("SELECT DISTINCT Launch_Site FROM SPACEXTABLE")
cur.fetchall()
```

```
[('CCAFS LC-40',), ('VAFB SLC-4E',), ('KSC LC-39A',), ('CCAFS SLC-40',)]
```

SpaceX utilizes 4 Launch Sites in the US for their Operations

Launch Site Names Begin with 'KSC'

Display 5 records where launch sites begin with the string 'KSC'

```
#cur.execute("SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'KSC%' LIMIT 5")
#print(cur.fetchall())
query = "SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'KSC%' LIMIT 5"
results = pd.read_sql_query(query, con)
results.head()
```

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
1	2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2	2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
3	2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
4	2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

With the **Kennedy Space Station** & its surrounding area being most prominent

Total Payload Mass

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

```
cur.execute("SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer == 'NASA (CRS)' ")
cur.fetchall()
```

```
[(45596,)]
```

NASA, the government funded industry Giant, being one of their main Customer

Average Payload Mass by F9 v1.1

Display average payload mass carried by booster version F9 v1.1

```
cur.execute("SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%' ")
cur.fetchall()
```

```
[(2534.6666666666665,)]
```

First Successful Ground Landing Date

List the date where the succesful landing outcome in drone ship was acheived.

Hint:Use min function

```
cur.execute("SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome == 'Success (drone ship)'")
cur.fetchall()
```

```
[('2016-04-08',)]
```

In 2016, SpaceX successful launch & land rockets

while on a moving **drone ship** at sea

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000

```
#Script for query then turn into df
query = "SELECT * FROM SPACEXTABLE WHERE Landing_Outcome == 'Success (ground pad)' AND (PAYLOAD_MASS__KG_   BETWEEN 4000 AND 6000)"
#cur.execute(query)
#cur.fetchall()
results = pd.read_sql_query(query, con)
results.head()
```

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
1	2017-09-07	14:00:00	F9 B4 B1040.1	KSC LC-39A	Boeing X-37B OTV-5	4990	LEO	U.S. Air Force	Success	Success (ground pad)
2	2018-01-08	1:00:00	F9 B4 B1043.1	CCAFS SLC-40	Zuma	5000	LEO	Northrop Grumman	Success (payload status unclear)	Success (ground pad)

After that event they continued to successfully operate on various grounds

Marking a technical revolutions in the Space Travel Industry

Total Number of Successful and Failure Mission Outcomes

SpaceX consistently produced More Success than Failure even in Testing

List the total number of successful and failure mission outcomes

```
query = "SELECT Mission_Outcome, COUNT(*) FROM SPACEXTABLE GROUP BY Mission_Outcome"
results = pd.read_sql_query(query, con)
results.head()
```

	Mission_Outcome	COUNT(*)
0	Failure (in flight)	1
1	Success	98
2	Success	1
3	Success (payload status unclear)	1

Boosters Carried Maximum Payload

List all the booster_versions that have carried the maximum payload mass. Use a subquery.

```
query = """SELECT "Booster_Version", "Payload_Mass__KG_" AS 'Maximum Load'
FROM SPACEXTABLE
WHERE "Payload_Mass__KG_" = (
    SELECT MAX("Payload_Mass__KG_") FROM SPACEXTABLE
);"""
results = pd.read_sql_query(query, con)
results.head(100)
```

	Booster_Version	Maximum Load
0	F9 B5 B1048.4	15600
1	F9 B5 B1049.4	15600
2	F9 B5 B1051.3	15600
3	F9 B5 B1056.4	15600
4	F9 B5 B1048.5	15600
5	F9 B5 B1051.4	15600
6	F9 B5 B1049.5	15600
7	F9 B5 B1060.2	15600
8	F9 B5 B1058.3	15600
9	F9 B5 B1051.6	15600
10	F9 B5 B1060.3	15600
11	F9 B5 B1049.7	15600

Falcon 9's B5 variations

having the Capability to Carries loads up to **15.6 Tons!**

2017 Launch Records

List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017

Note: SQLLite does not support monthnames. So you need to use substr(Date,6,2) for month, substr(Date,9,2) for date, substr(Date,0,5),='2017' for year.

```
query = """ SELECT substr(Date,6,2) AS Month , Landing_Outcome, Booster_Version, Launch_Site
FROM SPACEXTABLE
WHERE substr(Date,0,5)='2017' AND Landing_Outcome LIKE 'Success%';
"""
results = pd.read_sql_query(query, con)
results.head()
```

	Month	Landing_Outcome	Booster_Version	Launch_Site
0	01	Success (drone ship)	F9 FT B1029.1	VAFB SLC-4E
1	02	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
2	03	Success (drone ship)	F9 FT B1021.2	KSC LC-39A
3	05	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
4	06	Success (ground pad)	F9 FT B1035.1	KSC LC-39A

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

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

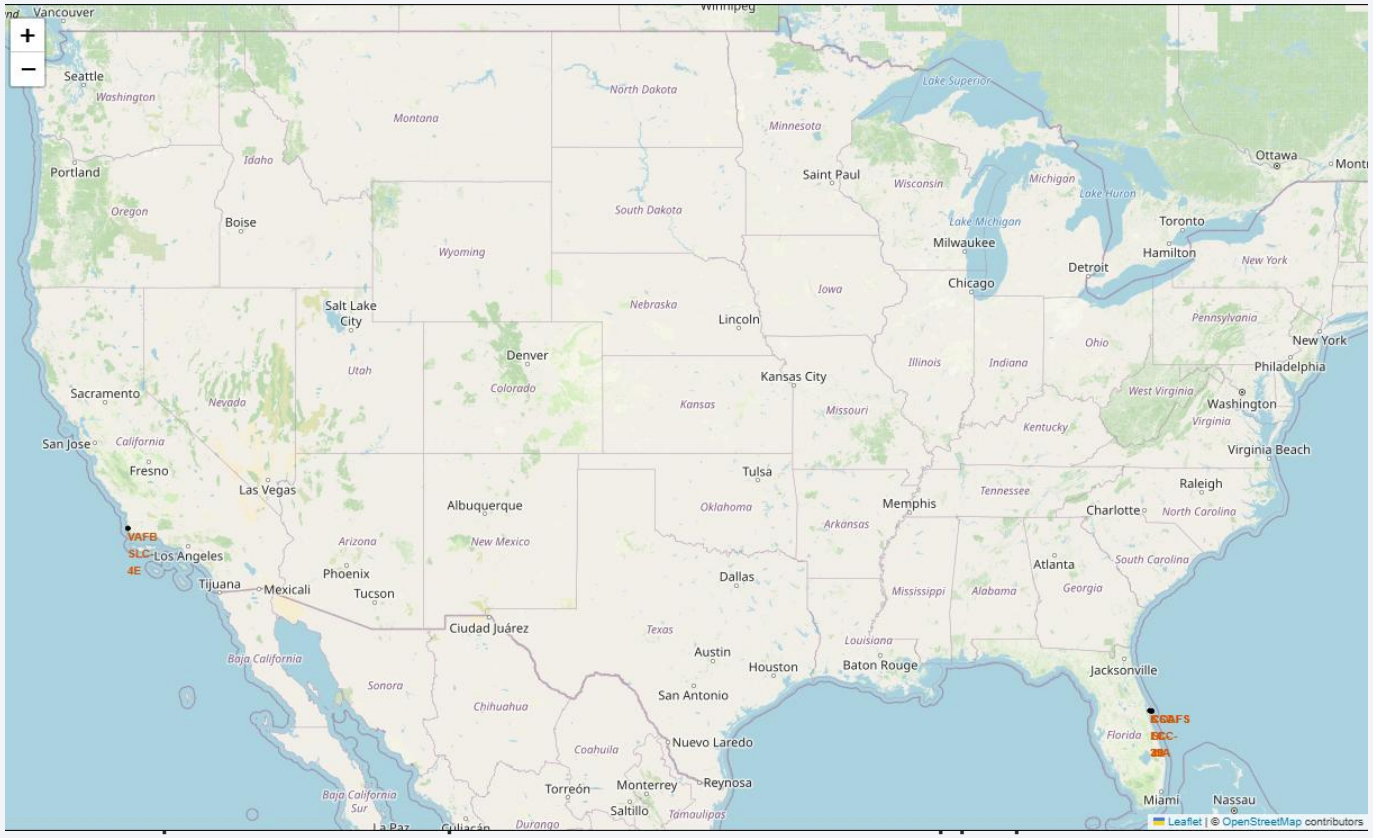
```
query = """ SELECT Landing_Outcome, COUNT(*) AS Outcome_Count
FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Outcome_Count DESC;
"""
results = pd.read_sql_query(query, con)
results.head(10)
```

	Landing_Outcome	Outcome_Count
0	No attempt	10
1	Success (drone ship)	5
2	Failure (drone ship)	5
3	Success (ground pad)	3
4	Controlled (ocean)	3
5	Uncontrolled (ocean)	2
6	Failure (parachute)	2
7	Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

Main Launch Sites in the US



West Coast vs **East Coast**

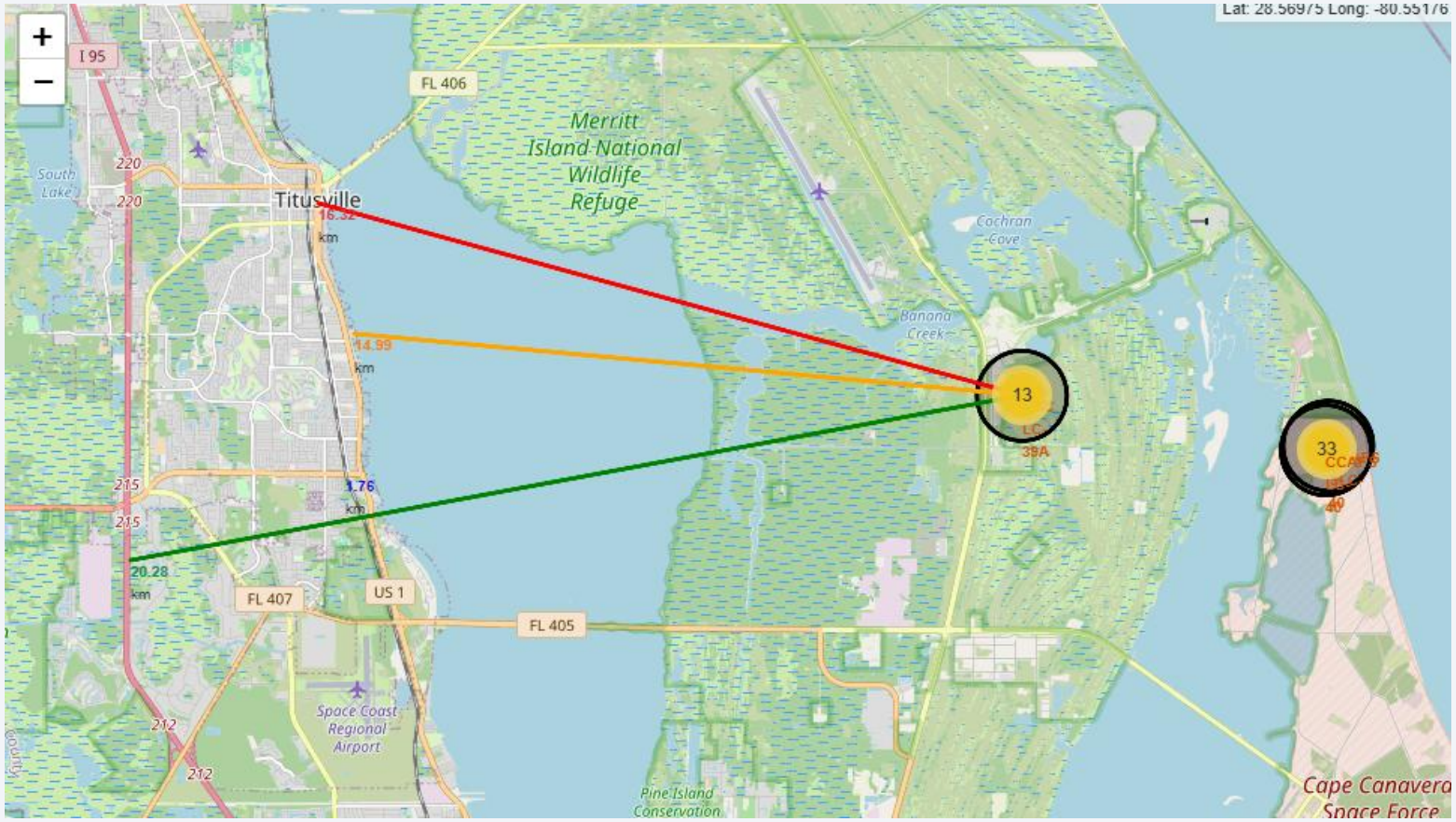
2 Main Launch Area

Clustering all launches with Success Indicator

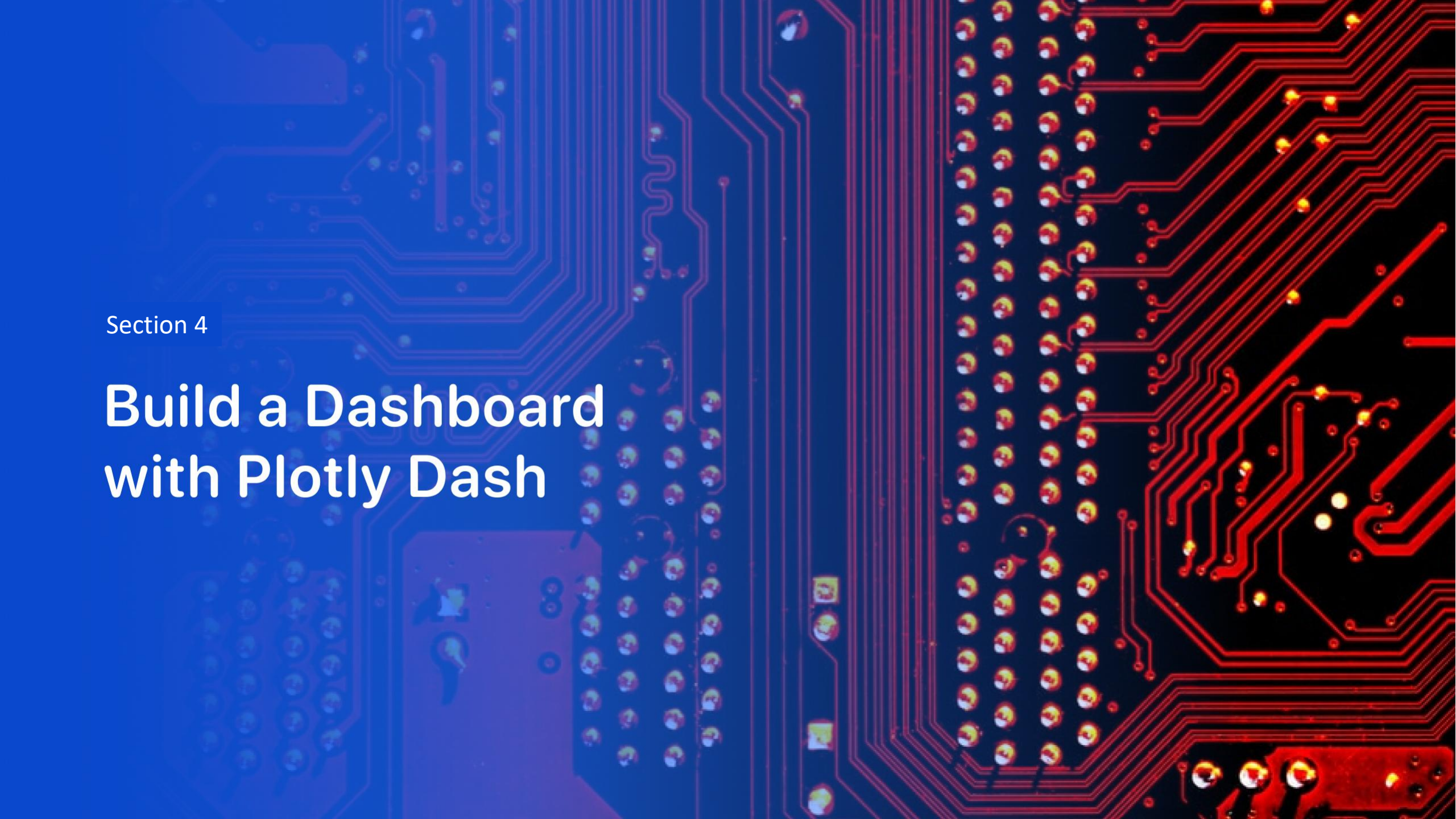
Majority of launches

are from the Cape Canaveral Area

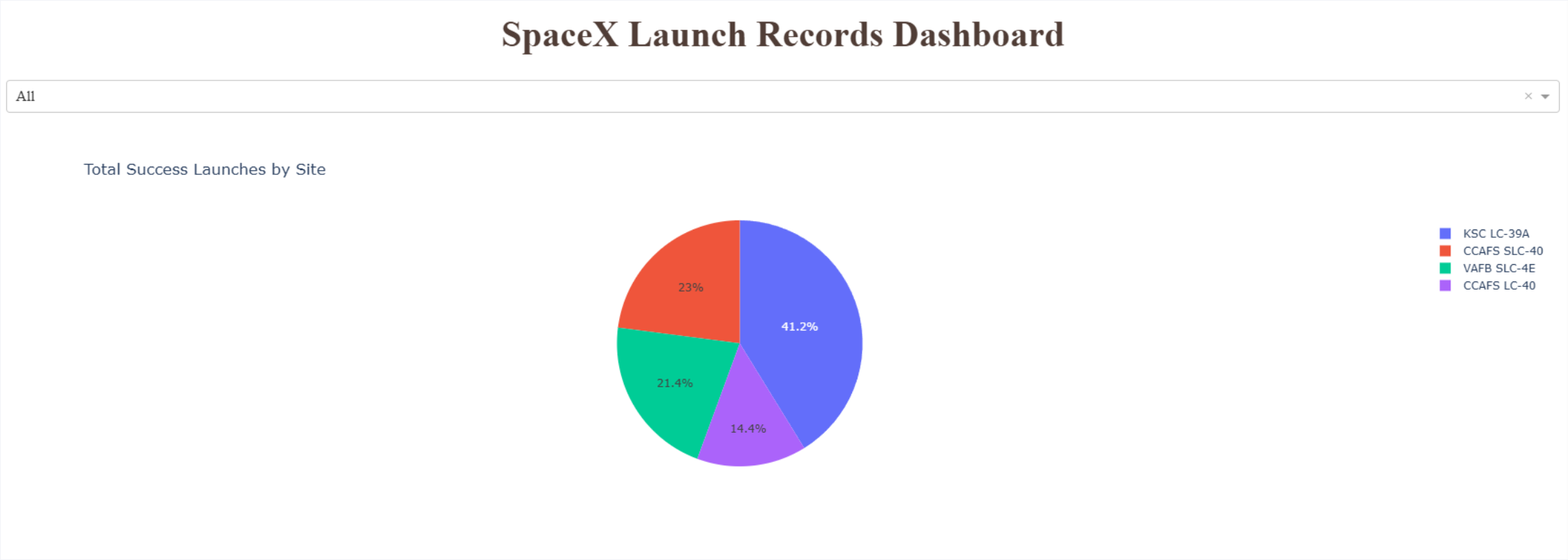
Distance to Site's Proximities



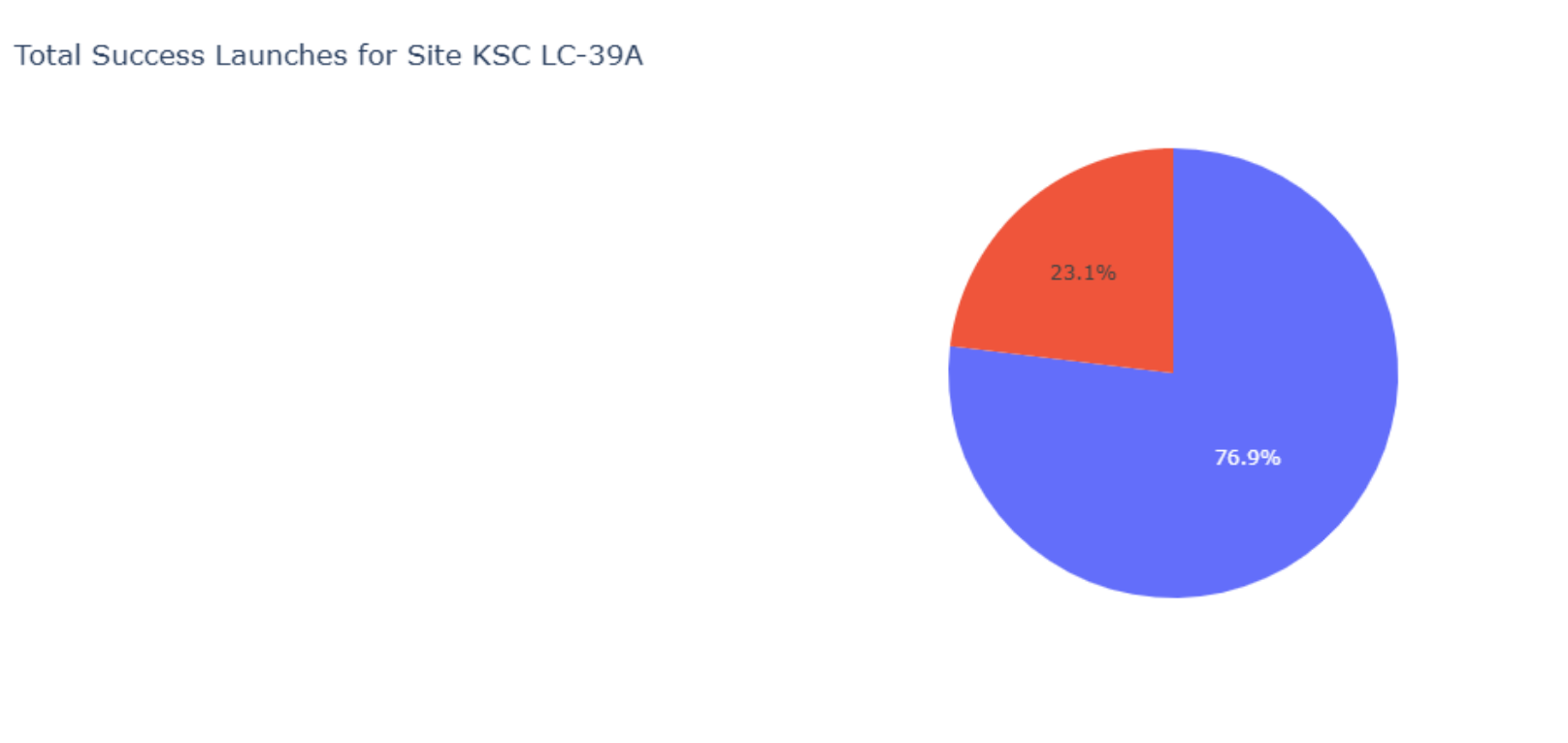
Example for proximities including railway, coastline, highway, etc.


Section 4
Build a Dashboard with Plotly Dash

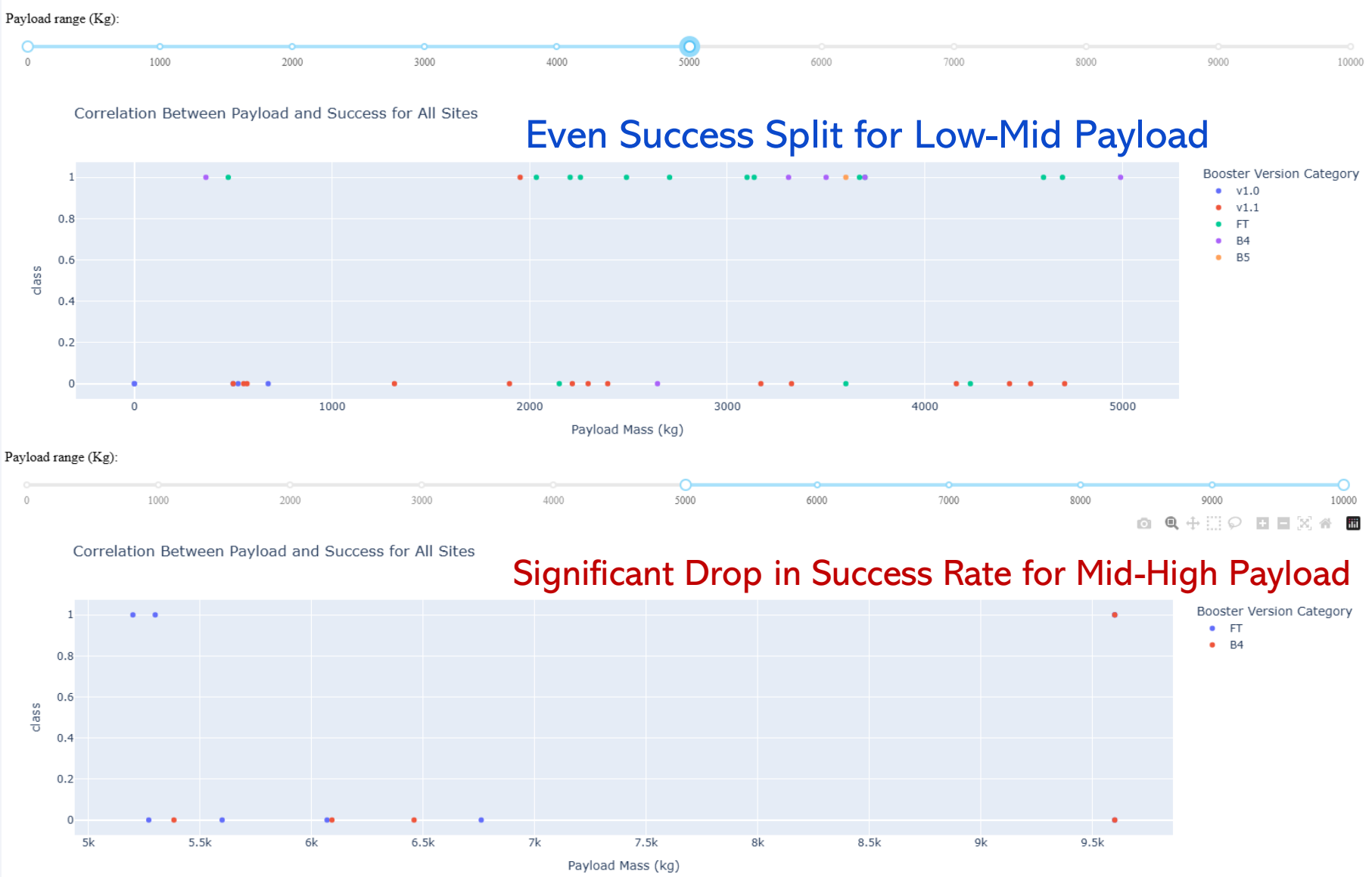
How Successful are these sites?



Kennedy Space Center leads in Launch Success!



Does Payload affect the Launch Outcome?



Section 5

Predictive Analysis (Classification)

Classification Accuracy

Tested Models

- Logistic Regression
- Support Vector Matrix
- Tree
- K-Nearest-Neighbor

```
from sklearn.metrics import jaccard_score, f1_score
# Testing Jaccard & F1 score for prediction on the whole Dataset
jaccard_scores = [
                  jaccard_score(Y, logreg_cv.predict(X), average='binary'),
                  jaccard_score(Y, svm_cv.predict(X), average='binary'),
                  jaccard_score(Y, tree_cv.predict(X), average='binary'),
                  jaccard_score(Y, knn_cv.predict(X), average='binary'),
                 ]
```

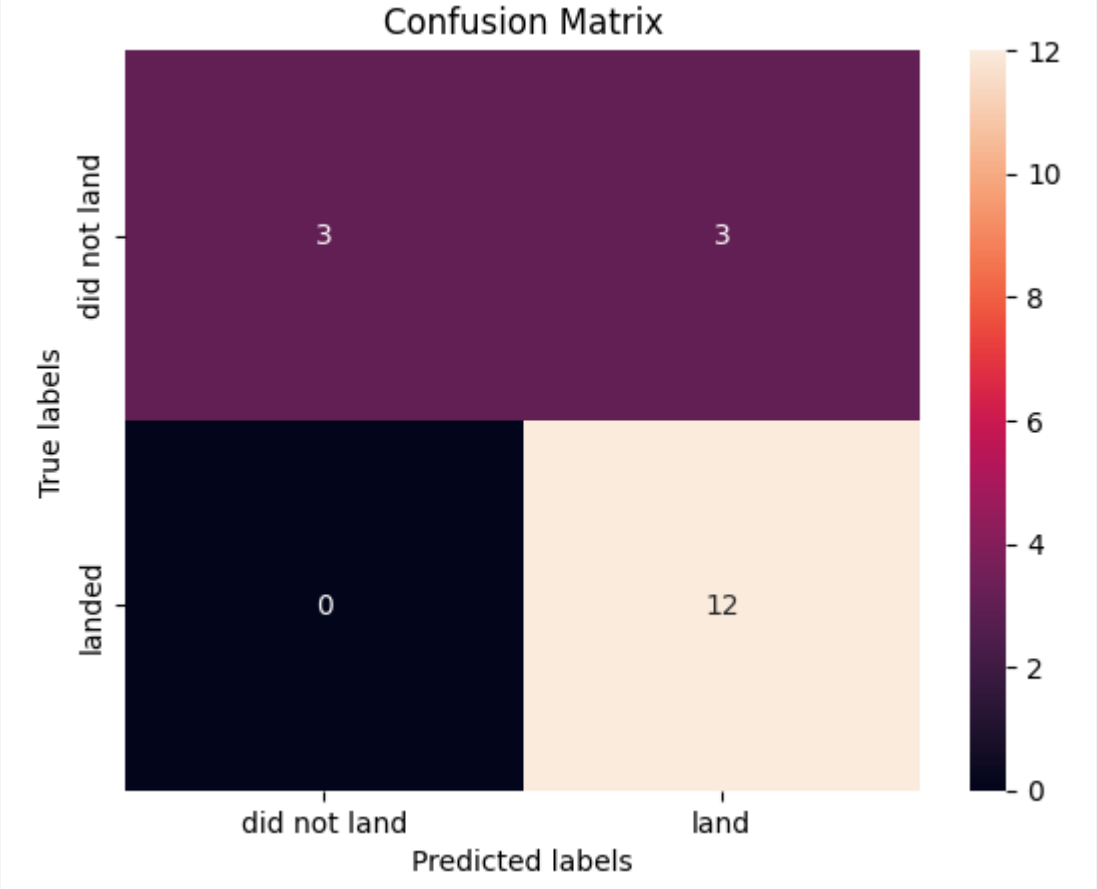
Comparing each ML Models' Prediction ability

```
f1_scores = [
              f1_score(Y, logreg_cv.predict(X), average='binary'),
              f1_score(Y, svm_cv.predict(X), average='binary'),
              f1_score(Y, tree_cv.predict(X), average='binary'),
              f1_score(Y, knn_cv.predict(X), average='binary'),
             ]

accuracy = [logreg_cv.score(X, Y), svm_cv.score(X, Y), tree_cv.score(X, Y), knn_cv.score(X, Y)]

scores = pd.DataFrame(np.array([jaccard_scores, f1_scores, accuracy]),
                      index=['Jaccard_Score', 'F1_Score', 'Accuracy'],
                      columns=['LogReg', 'SVM', 'Tree', 'KNN'])
scores
```

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.833333	0.845070	0.840580	0.819444
F1_Score	0.909091	0.916031	0.913386	0.900763
Accuracy	0.866667	0.877778	0.877778	0.855556

SVM leads in all scoring metrics!

Confusion Matrix



Same matrix displayed in all 4 Models

- ~83% of the prediction is **Correct**
- **Mistakes in predicting False Positives**
- **Low distinction between 4 Models**

(May be due to a Relatively low amount of test data)

Conclusions

- **Orbit height** matters to the operation's success
- SpaceX provides **various agency** with improved launching capabilities
- SpaceX's Rocket **Performance improves** with time
- **Kennedy Space Center leads** in Launch Performance (Mainly launches on the East)
- Using **SVM** Method is sufficient for Prediction, but Having more Samples is ideal.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!