IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Hamid Kaveh
December, 12, 2021

Outline

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

Executive Summary

- **Summary of methodologies**

❑ Data collection

❑ Data wrangling

❑ EDA with SQL

❑ EDA with data visualization

❑ Folium

❑ Machine Learning Analysis

- **Summary of all results**

❑ Data Analysis Results

❑ Charts and Graphs

❑ Prediction Results

Introduction

- **Project background and context**

  During this project we will predict if the Falcon 9 first stage will land successfully. 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 if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

- **Problems you want to find answers**

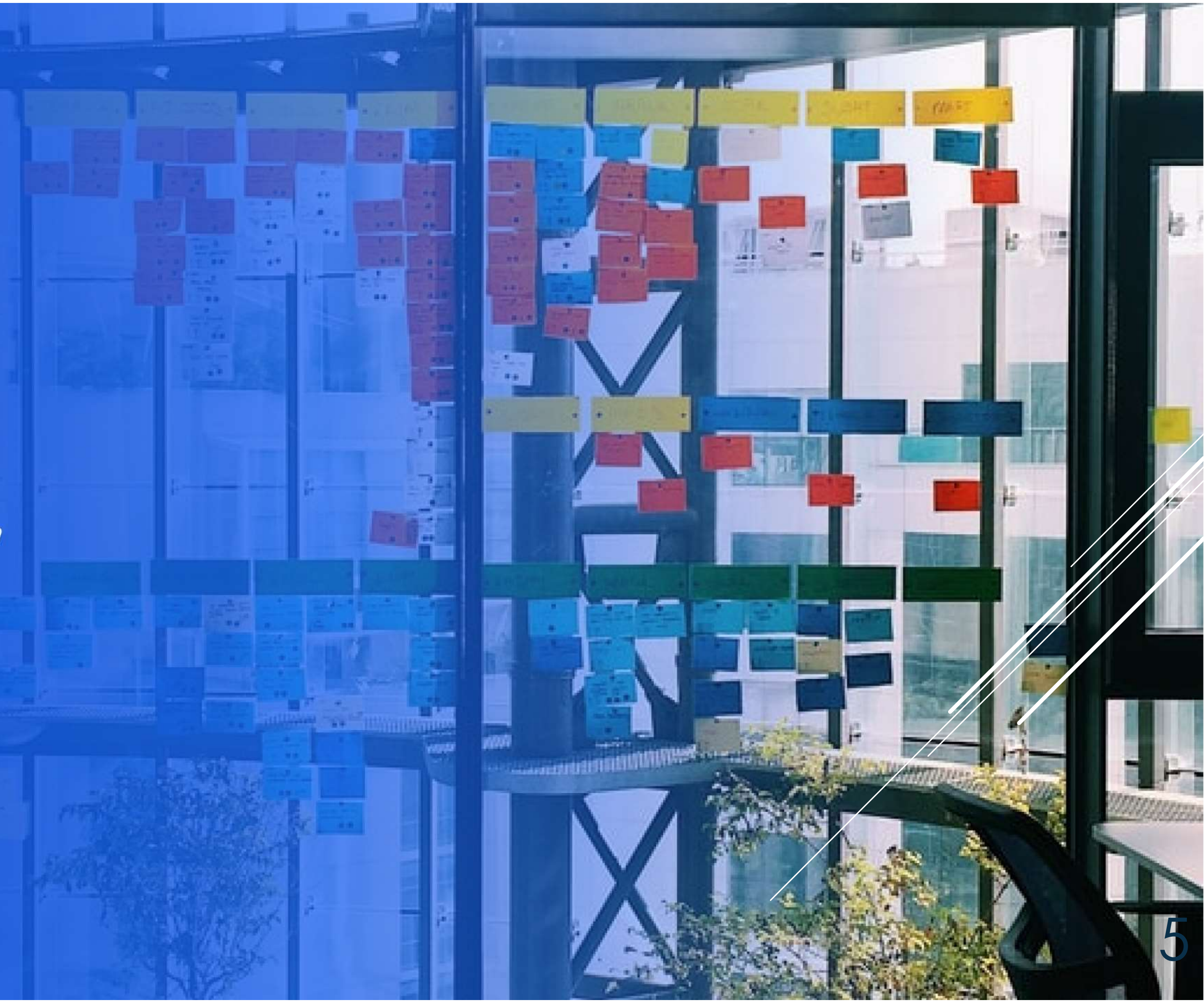
  What influences if the rocket will land successfully?

  What conditions does SpaceX have to achieve to get the best results

  and ensure the best rocket success landing rate.

Section 1

Methodology



Methodology

Executive Summary

❖Data collection methodology:

SpaceX Rest API
Web Scrapping from Wikipedia

❖Perform data wrangling

One Hot Encoding data fields for Machine Learning and dropping irrelevant columns

❖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

- **Describe how data sets were collected**.

We worked with SpaceX launch data that is gathered from the SpaceX REST API.

payload delivered, launch specifications, landing specifications, and landing outcome.

The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.

Another popular data source for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup.

SpaceX Reset API → **Getting .json file** → **Normalize Data and Converting to .csv file.**

Getting HTML file from Wikipedia → **Using BeautifulSoup for preparing data** → **Normalize Data and Converting to .csv file.**

Data Collection – SpaceX API

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

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

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

```python
# Call getBoosterVersion
getBoosterVersion(data)
```

```python
# Call getLaunchSite
getLaunchSite(data)
```

```python
# Call getPayloadData
getPayloadData(data)
```

```python
# Call getCoreData
getCoreData(data)
```

```python
# Create a data from launch_dict
data_ = pd.DataFrame.from_dict(launch_dict, orient='columns', dtype=None)
```

```python
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = data_[data_['BoosterVersion'] != 'Falcon 1']
data_falcon9['BoosterVersion'].count()
```

```python
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping

```python
page = requests.get(static_url)
```

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

```python
# Use BeautifulSoup() to create a BeautifulSoup object
soup = BeautifulSoup(page.text, 'html.parser')
```

```python
html_tables = soup.find_all('table')
```

```python
column_names = []
temp = soup.find_all('th')
for x in range(len(temp)):
    try:
     name = extract_column_from_header(temp[x])
     if (name is not None and len(name) > 0):
        column_names.append(name)
    except:
     pass
```

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

# Remove an irrelvant column
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']=[]
```

```python
df = pd.DataFrame.from_dict(launch_dict)
```

```python
df.to_csv('spacex_web_scraped.csv', index=False)
```

Data Wrangling

Describe how data were processed

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship.

1. Calculate the number of launches on each site
2. Calculate the number and occurrence of each orbit
3. Calculate the number and occurrence of mission outcome per orbit type
4. Create a landing outcome label from Outcome column

EDA with Data Visualization

Summarize what charts were plotted and why you used those charts:

- **Visualize the relationship between Flight Number and Launch Site**
- **Visualize the relationship between Payload and Launch Site**
- **Visualize the relationship between success rate of each orbit type**
- **Visualize the relationship between Flight-Number and Orbit type**
- **Visualize the relationship between Payload and Orbit type**
- **Visualize the launch success yearly trend**

EDA with SQL

Using SQL for answering to:

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'CCR'
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date where the successful landing outcome in drone ship was achieved.
- Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes
- Listing the names of the booster_versions which have carried the maximum payload mass.
- Listing the records which will display the month names, successful landing_outcomesin ground pad ,booster versions, launch_sitefor the months in year 2015
- Ranking the count of successful landing_outcomesbetween the date 2010-06-04 and 2017-03-20 in descending order.

Build an Interactive Map with Folium

To visualize the Launch Data into an interactive map. We took the Latitude and Longitude Coordinates at each launch site and added a Circle Marker around each launch site with a label of the name of the launch site.

Build a Dashboard with Plotly Dash

The dashboard is built with Flask and Dash web framework:

- **Scatter Graph showing the relationship with Outcome and Payload Mass (Kg) for the different Booster Versions**

  It shows the relationship between two variables.

  It is the best method to show you a non-linear pattern.

  The range of data flow, i.e. maximum and minimum value, can be determined.

- **Pie Chart showing the total launches by a certain site/all sites**

  display relative proportions of multiple classes of data.

  size of the circle can be made proportional to the total quantity it represents.

Predictive Analysis (Classification)

BUILDING MODEL

- Loading Data
- Preprocessing
- Split our data into training and test data sets
- Decide which type of machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit datasets

EVALUATING MODEL

- Check accuracy for each model
- Plot Confusion Matrix

IMPROVING MODEL

- Feature Engineering

FINDING THE BEST PERFORMING CLASSIFICATION MODEL

Results

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

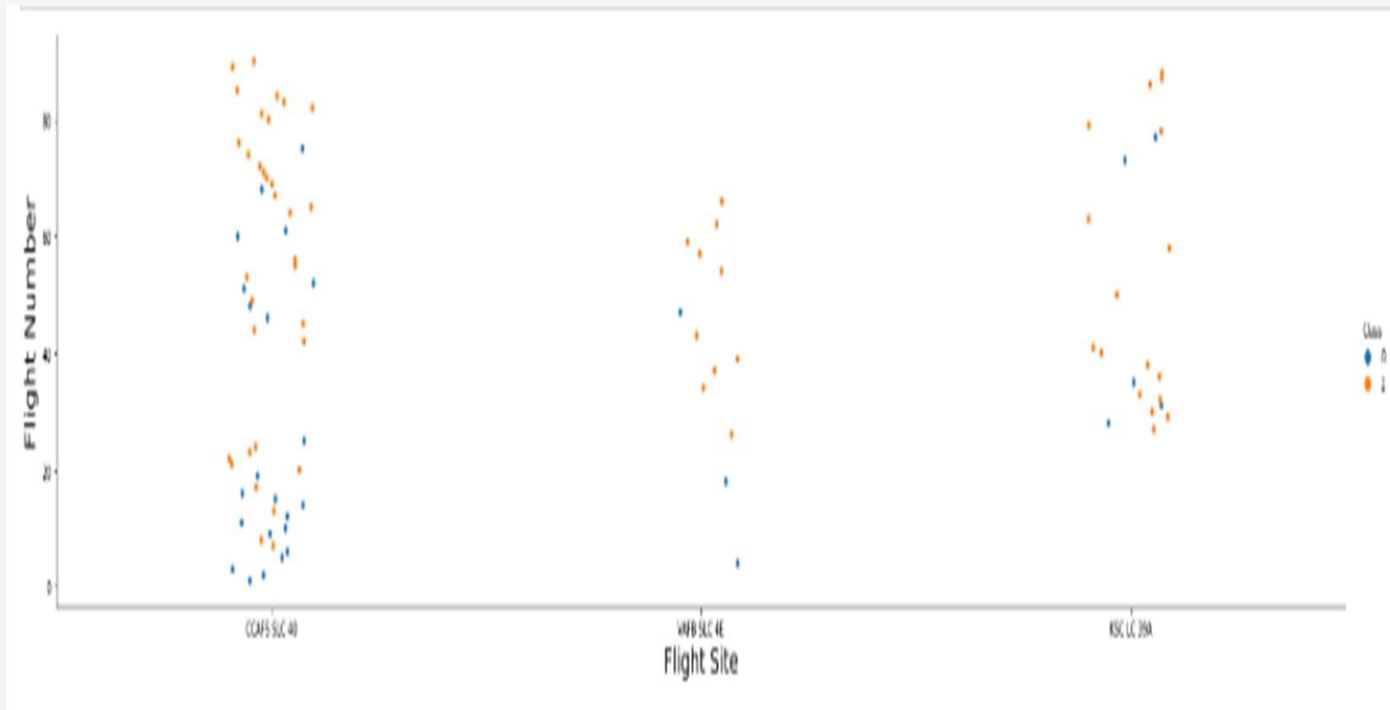
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

Show a scatter plot of Flight Number vs. Launch Site

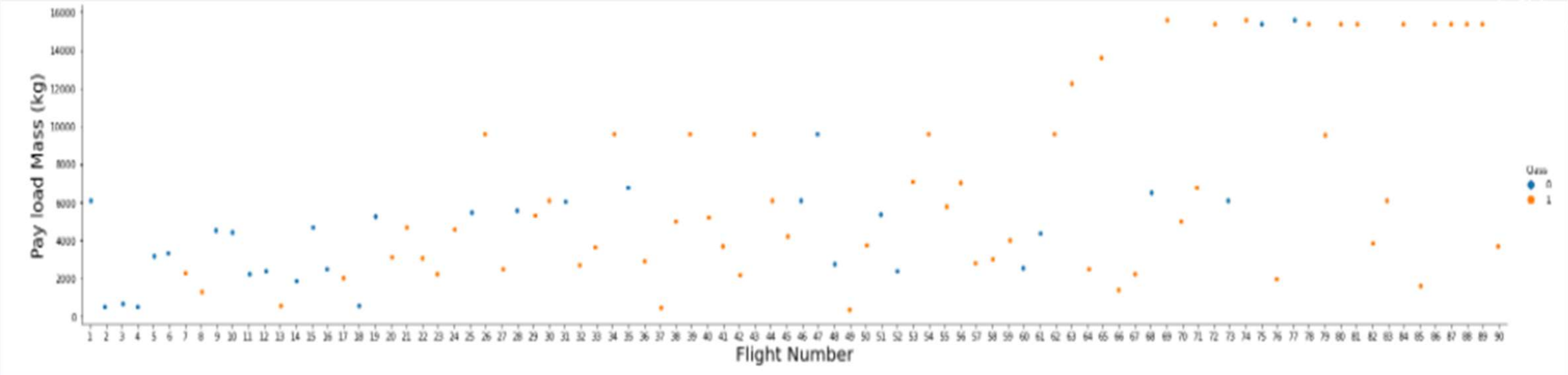
The more amount of flights at a launch site the greater the success rate at a launch site.



Payload vs. Launch Site

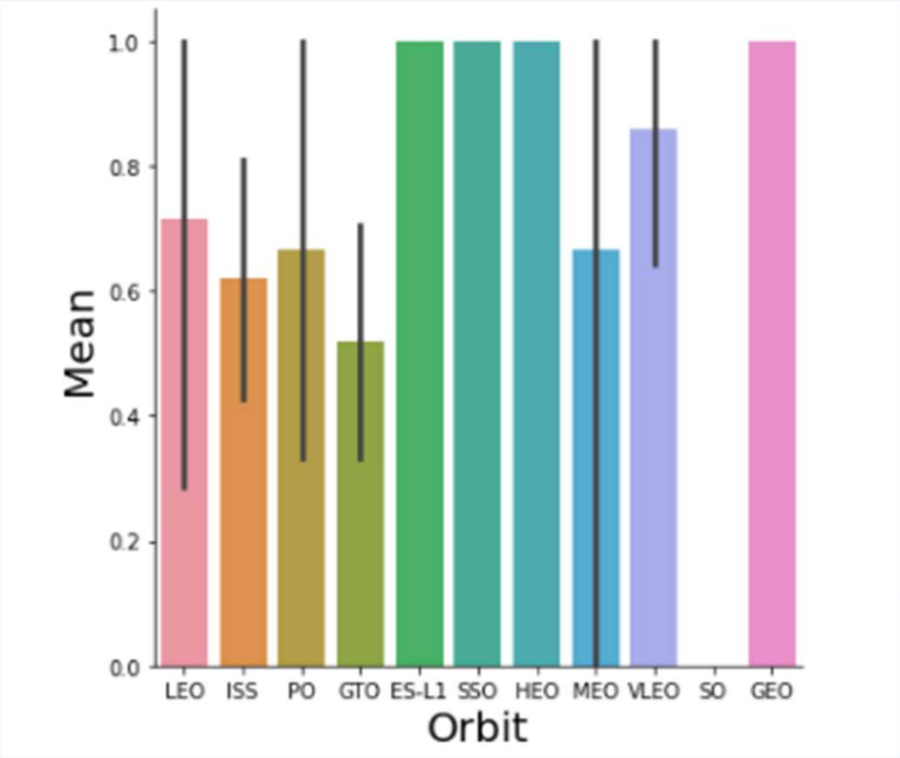
Show a scatter plot of Payload vs. Launch Site:

The greater the payload mass for Launch Site CCAFS SLC 40 the higher the success rate for the Rocket. There is not quite a clear pattern to be found using this visualization to make a decision if the Launch Site is dependent on Pay Load Mass for a success launch.



Success Rate vs. Orbit Type

Show a bar chart for the success rate of each orbit type:

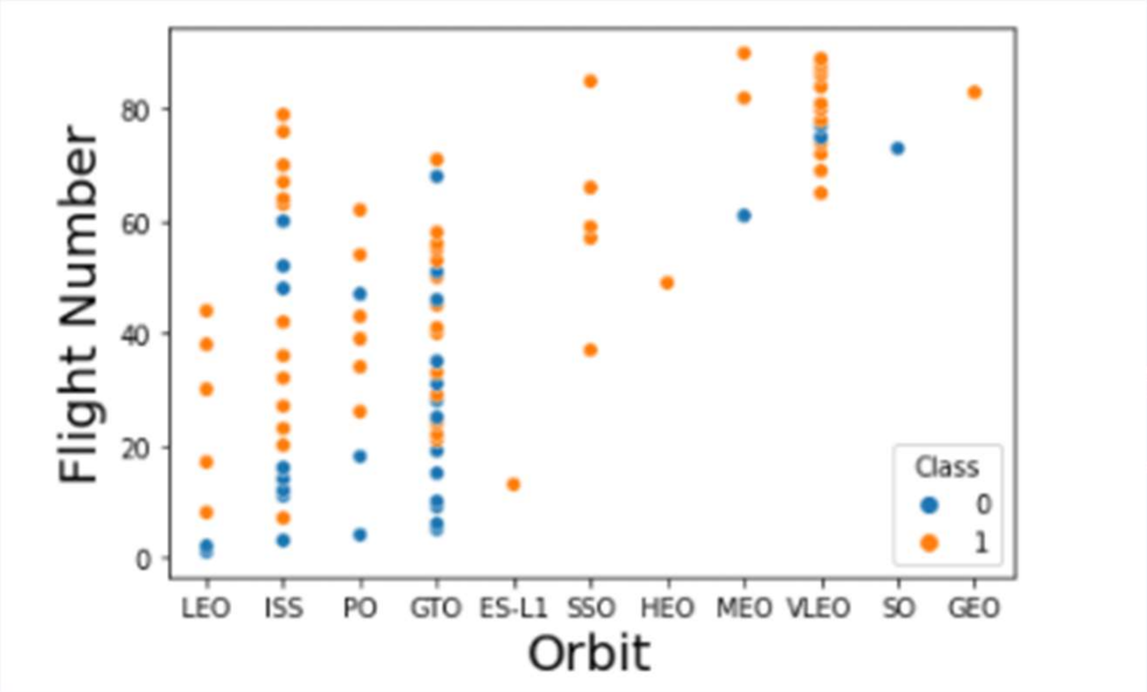


Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate.

Flight Number vs. Orbit Type

Show a scatter point of Flight number vs. Orbit type:

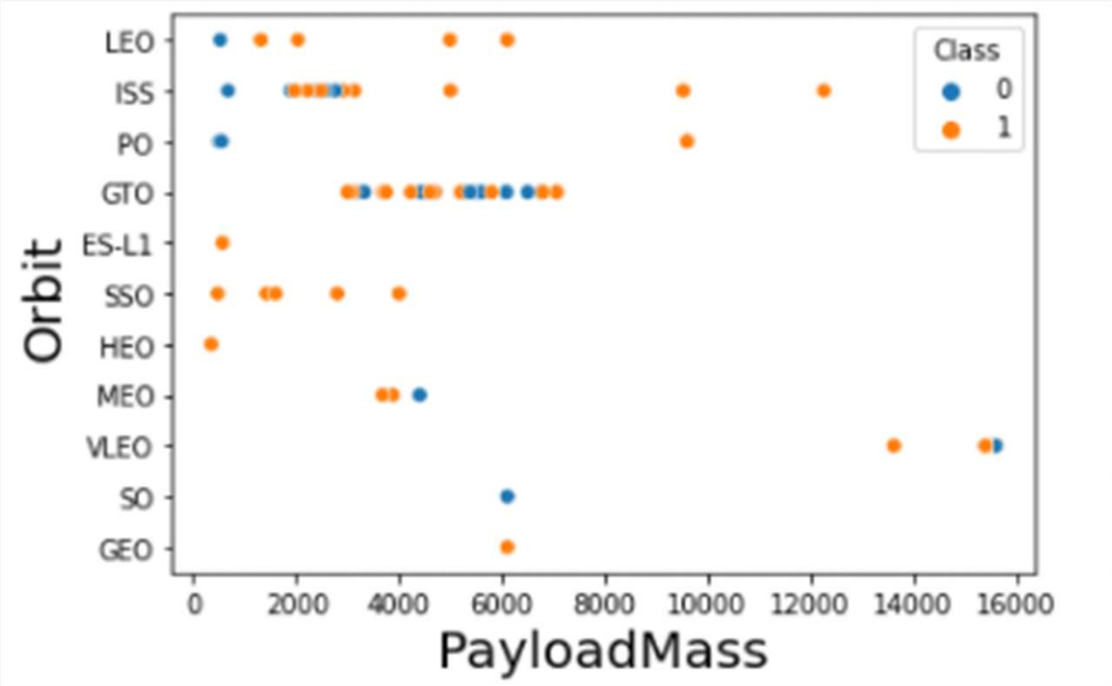
You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



Payload vs. Orbit Type

Show a scatter point of payload vs. orbit type:

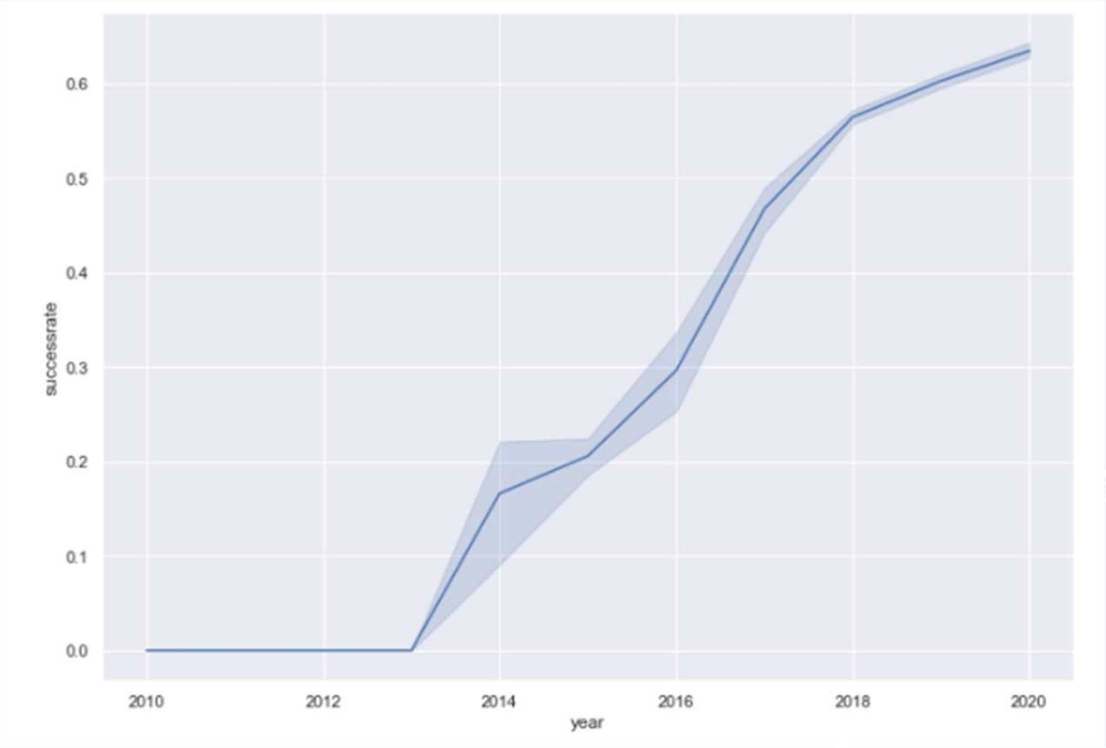
You should observe that Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.



Launch Success Yearly Trend

Show a line chart of yearly average success rate:

you can observe that the success rate since 2013 kept increasing till 2020



All Launch Site Names

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

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

- Using the word DISTINCT in the query means that it will only show Unique values in the Launch_Site column from tblSpaceX:

- `select DISTINCT [Launch_Site] from [dbo].[Spacex]`

Launch Site Names Begin with 'CCA'

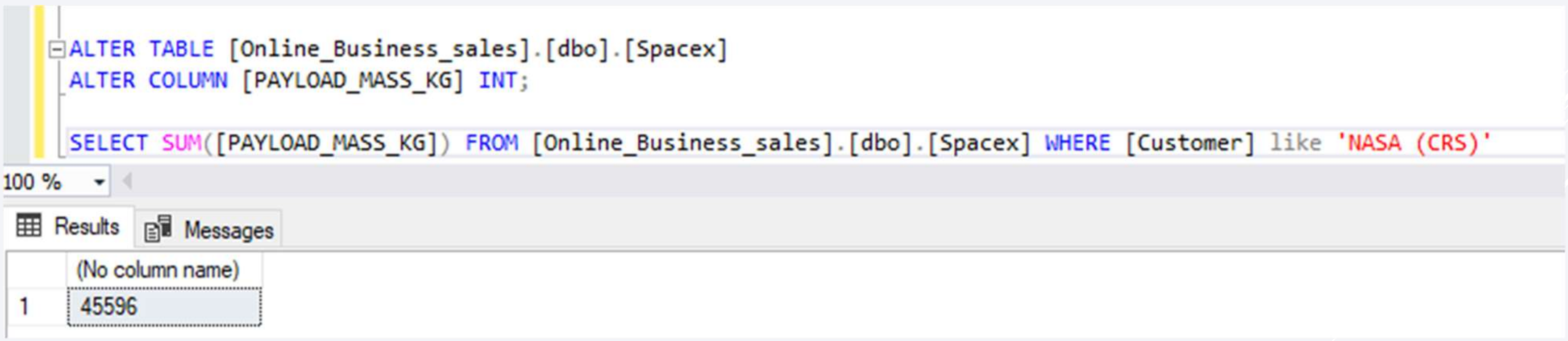
Using the word TOP 5 in the query means that it will only show 5 records from tblSpaceX and LIKE keyword has a wild card with the words ‘CCA%’ the percentage in the end suggests that the Launch_Site name must start with CCA.

- **SELECT “column_name” FROM “table_name” WHERE “column_name” LIKE {PATTERN};**

	Date	Time_UTC	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS_KG	Orbit	Customer	Mission_Outcome	Landing_Outcome
1	04-06-2010	2021-12-13 18:45:00.0000000	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2	08-12-2010	2021-12-13 15:43:00.0000000	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of B...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
3	22-05-2012	2021-12-13 07:44:00.0000000	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
4	08-10-2012	2021-12-13 00:35:00.0000000	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
5	01-03-2013	2021-12-13 15:10:00.0000000	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- SELECT SUM([PAYLOAD_MASS_KG]) FROM [dbo].[Spacex] WHERE [Customer] like 'NASA (CRS)'



Average Payload Mass by F9 v1.1

- seleSELECT AVG([PAYLOAD_MASS_KG]) FROM [dbo].[Spacex] WHERE [Booster_Version] = 'F9v1.1'

	(No column name)
1	2928

First Successful Ground Landing Date

- ```
  SELECT MIN([Date]) FROM [dbo].[Spacex] WHERE [Landing_Outcome] like 'Success(droneship)'
  ```

	(No column name)
1	06-05-2016
  ```

Successful Drone Ship Landing with Payload between 4000 and 6000

```
SELECT [Booster_Version] FROM [Online_Business_sales].[dbo].[Spacex] WHERE
([Landing_Outcome] like'Success(groundpad)') AND ([PAYLOAD_MASS_KG]>4000) AND
([PAYLOAD_MASS_KG]<6000)
```

Total Number of Successful and Failure Mission Outcomes

```
Select (SELECT Count([Mission_Outcome]) from [dbo].[Spacex] where [Mission_Outcome] LIKE
'%Success%') as Successful_Mission_Outcomes ,
(SELECT Count(Mission_Outcome) from [dbo].[Spacex] where Mission_Outcome LIKE '%Failure%')
as Failure_Mission_Coutcomes
```

	Successful_Mission_Outcomes	Failure_Mission_Coutcomes
1	100	1

Boosters Carried Maximum Payload

- SELECT DISTINCT [Booster_Version], MAX([PAYLOAD_MASS_KG]) AS [MaximumPayloadMass] FROM [dbo].[Spacex] GROUP BY [Booster_Version] ORDER BY [MaximumPayloadMass] DESC

Results | Messages

	Booster_Version	MaximumPayloadMass
1	F9 B5 B1048.4	15600
2	F9 B5 B1048.5	15600
3	F9 B5 B1049.4	15600
4	F9 B5 B1049.5	15600
5	F9 B5 B1049.7	15600
6	F9 B5 B1051.3	15600
7	F9 B5 B1051.4	15600
8	F9 B5 B1051.6	15600
9	F9 B5 B1056.4	15600
10	F9 B5 B1058.3	15600
11	F9 B5 B1060.2	15600
12	F9 B5 B1060.3	15600
13	F9 B5 B1049.6	15440
14	F9 B5 B1059.3	15410
15	F9 B5 B1051.5	14932
16	F9 B5 B1049.3	13620
17	F9 B5B1058.1	12530
18	F9 B5B1061.1	12500
19	F9 B5B1051.1	12055
20	F9 B5 B1046.4	12050
21	F9 B4 B1041.2	9600
22	F9 B4 B1041.1	9600

2015 Launch Records

```
SELECT DATENAME(month, DATEADD(month, MONTH(CONVERT(date, Date, 105)), 0)- 1) AS
Month, [Booster_Version], [Launch_Site], [Landing_Outcome]
FROM [dbo].[Spacex]
WHERE ([Landing_Outcome] LIKE N'%Success %') AND (YEAR(CONVERT(date, Date, 105)) =
'2015')
```

Results | Messages

	Month	Booster_Version	Launch_Site	Landing_Outcome
1	December	F9 FT B1019	CCAFS LC-40	Success (ground pad)

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

- SELECT COUNT(Landing_Outcome) FROM [dbo].[Spacex] WHERE (Landing_Outcome
- LIKE '%Success%') AND (Date > '04-06-2010') AND (Date < '20-03-2017')

Results | Messages

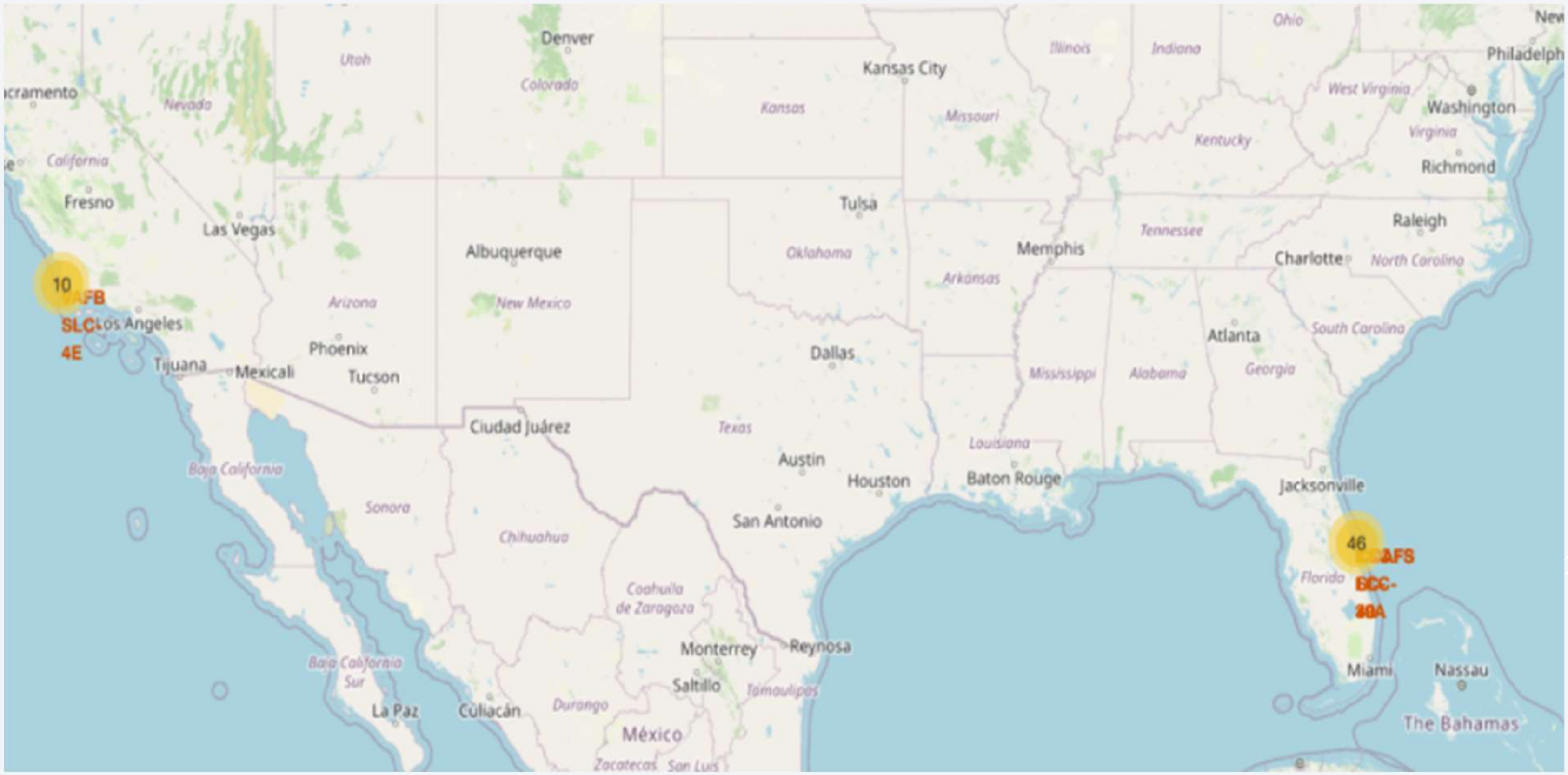
	(No column name)
1	34

Section 4

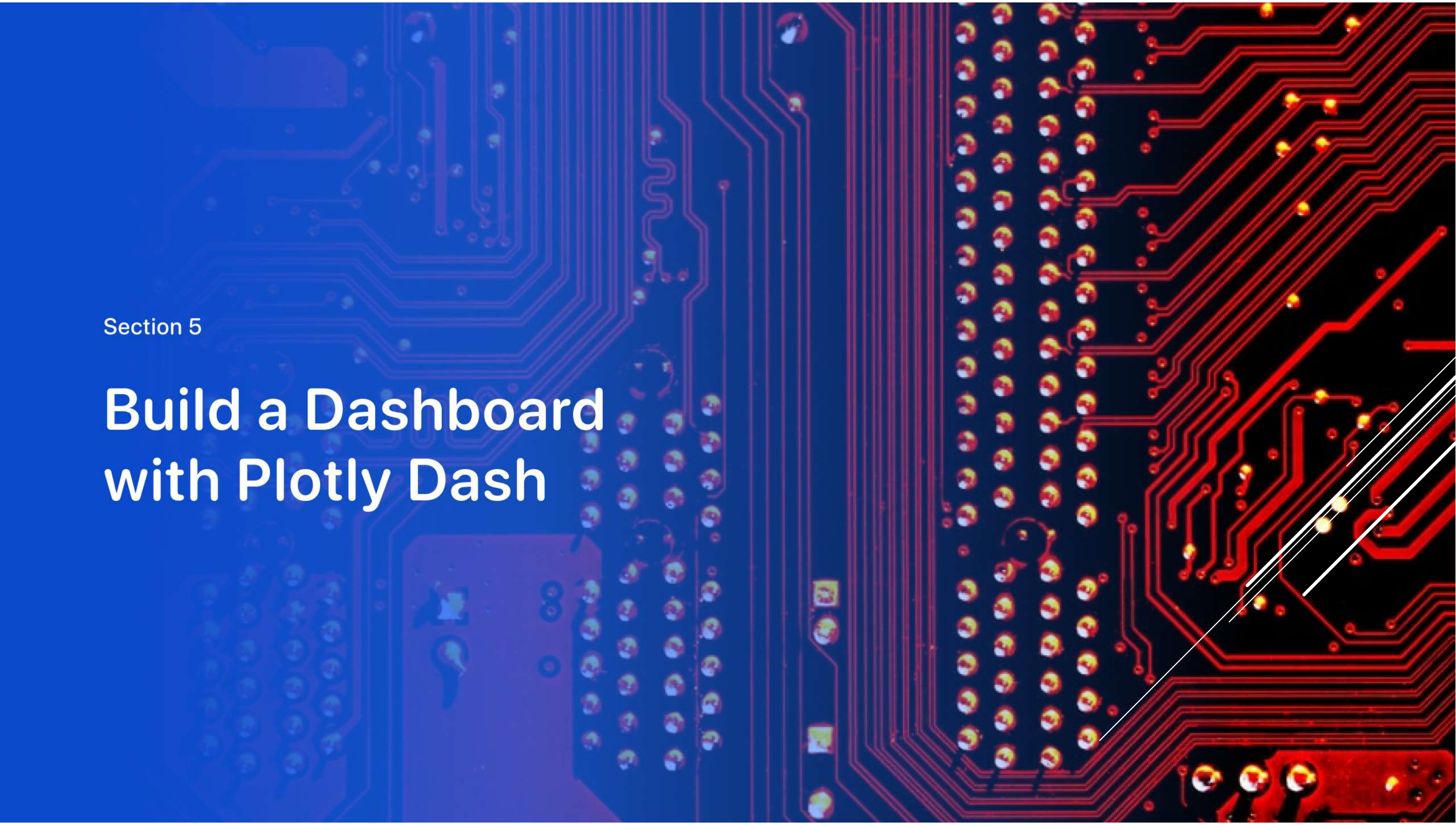
Launch Sites Proximities Analysis

Folium Map: All launch sites global map markers

Folium Map



Folium Map: Color Labelled Markers

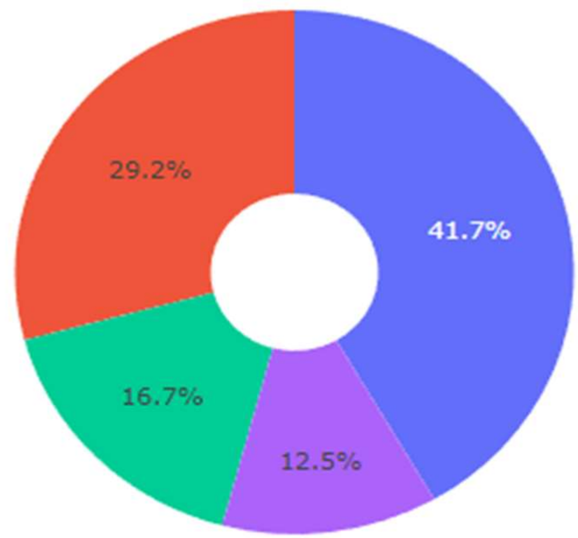


Section 5

Build a Dashboard with Plotly Dash

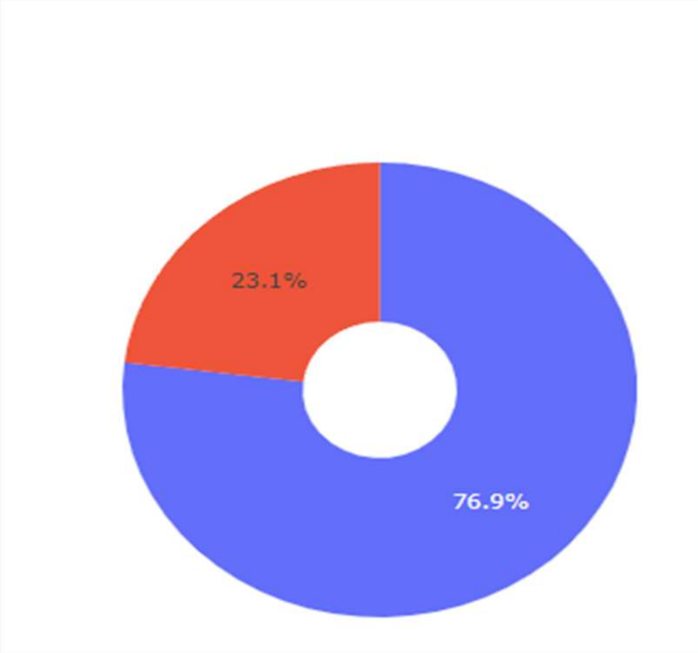
launch success count for all sites

We can see that KSC LC-39A had the most successful launches from all the sites



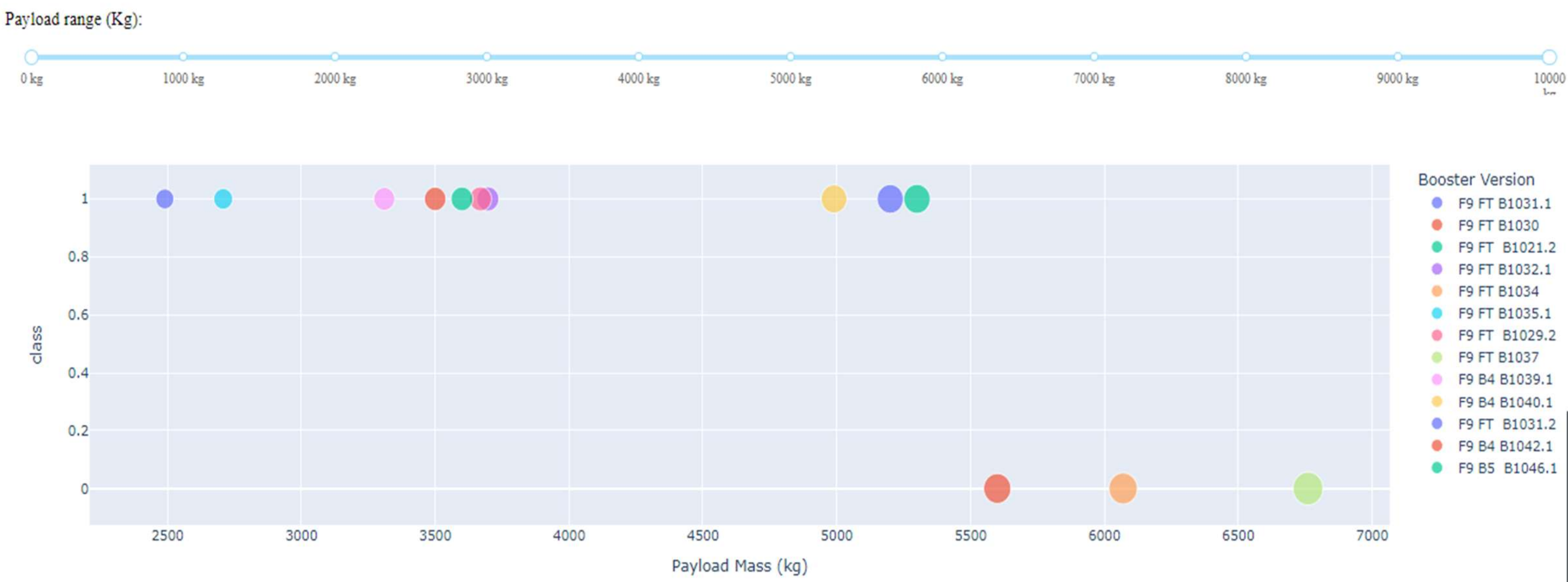
The Pie chart for the launch site with highest launch success ratio

KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate



Payload vs. Launch Outcome scatter plot for all sites

- We can see the success rates for low weighted payloads is higher than the heavy weighted payloads



Section 6

Predictive Analysis (Classification)

Classification Accuracy

Visualize the built model accuracy for all built classification models, in a bar chart

- logistic regression: `accuracy : 0.8464285714285713`
- SVM: `accuracy: 0.8333333333333334`
- Tree: `accuracy : 0.875`
- KNN: `accuracy : 0.8482142857142858`

Best Model:

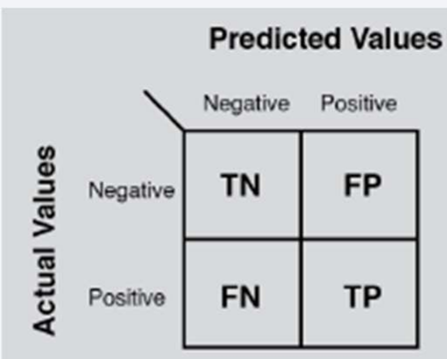
After selecting the best hyperparameters for the decision tree classifier using the validation data, we achieved 87.5% accuracy on the test data.

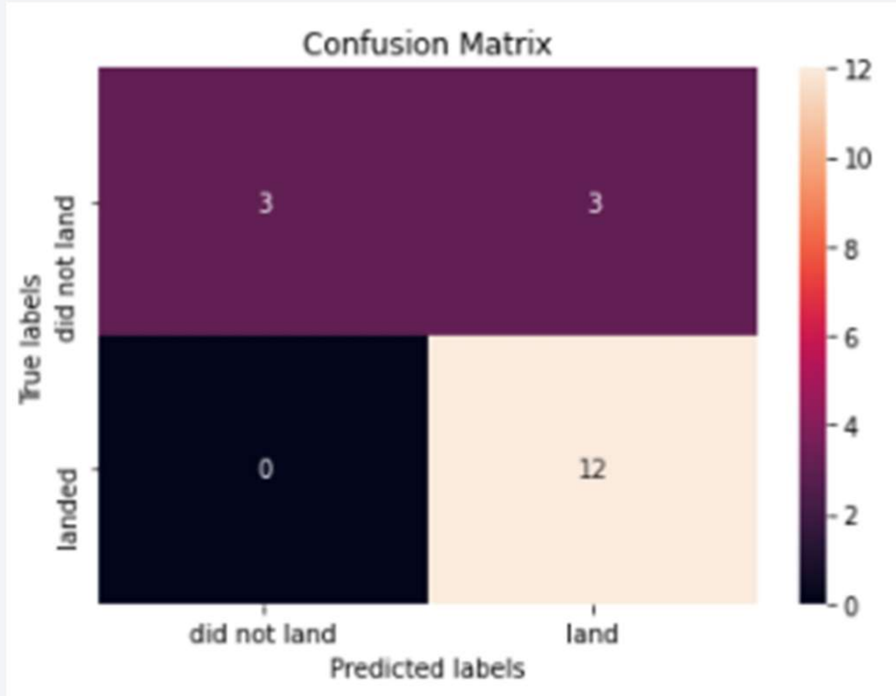
```
[33]: Scoring = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg.best_score_}
      bestalgorithm = max(Scoring, key=Scoring.get)
      print('Best Algorithm is',bestalgorithm,'with a score of',Scoring[bestalgorithm])
      if bestalgorithm == 'Tree':
          print('Best Params is :',tree_cv.best_params_)
      if bestalgorithm == 'KNN':
          print('Best Params is :',knn_cv.best_params_)
      if bestalgorithm == 'LogisticRegression':
          print('Best Params is :',logreg.best_params_)

      Best Algorithm is Tree with a score of 0.875
      Best Params is : {'criterion': 'entropy', 'max_depth': 4, 'max_features': 'auto', 'min_samples_leaf': 1, 'min_samples_split': 10, 'splitter': 'random'}
```

Confusion Matrix

- A confusion matrix is **a table that** is often used to describe the performance of a classification model on a set of test data for which the true values are known.





Conclusions

- **We can see that KSC LC-39A had the most successful launches from all the sites.**
- **Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate.**
- **The Tree Classifier Algorithm is the best for Machine Learning for this dataset.**

Appendix

For doing this project we use Jupyter Lab for our coding in Python language and Microsoft SQL Server Management Studio 18 for Query Language.


Thank you!