IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Hillson Lam
22 August 2025

# Outline

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

# Executive Summary

- Summary of methodologies
  - Data Collection and Data Wrangling
  - Exploratory Data Analysis with Data Visualization and SQL
  - Interactive Visual Analytics with Folium and Dashboard with Plotly Dash
  - Predictive Analysis (Classification)
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive Visual Analytics plots and screenshots
  - Predictive Analysis results

# Introduction

- Project background and context
  - SpaceX launches rockets relatively inexpensively that their Falcon 9 rocket launches with a cost of 62 million dollars, while others cost circa 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Thus we would like to know if the first stage lands, we can then estimate the cost of each launch. We want to know the success rate of the first stage. We want to determine the price of each launch and whether SpaceX will reuse the first stage, by gathering information about Space X launches, creating dashboards and insights and using a machine learning model

- Insights we want to know
  - Relationship between successful rate and other parameters (eg. Payload mass, launch location, and orbits)
  - What is the best analytical method, the constraints
  - The condition which allows the best opportunity for successful landing

Section 1

# Methodology

# Methodology

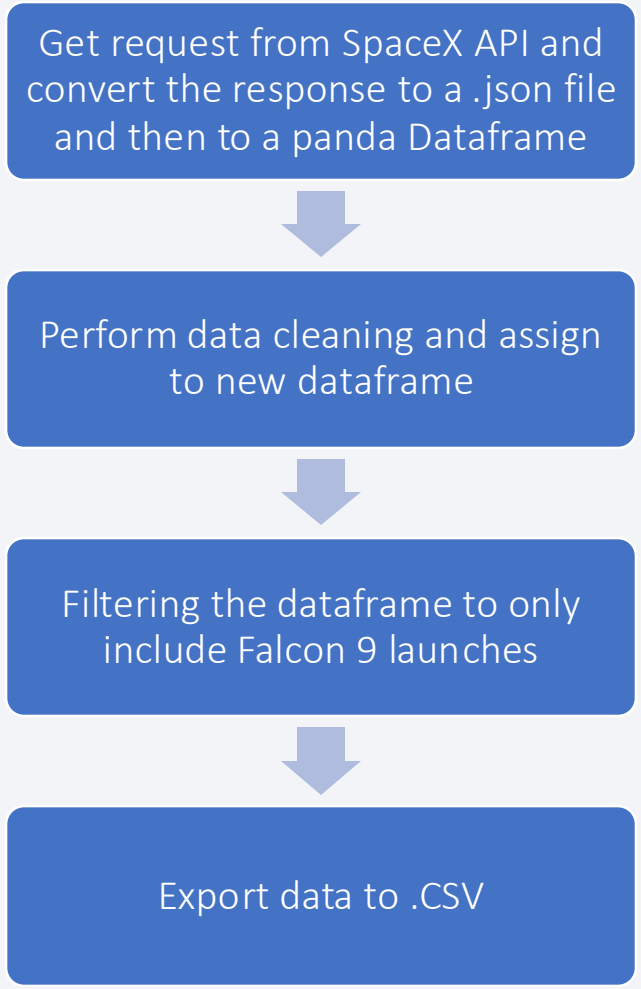
## Executive Summary

- Data collection methodology:
  - SpaceX REST API
  - Web Scraping
- Perform data wrangling
  - Filtering Data
  - Handling Missing data
  - Preparing data using One Hot Encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Fine-tuning and evaluating results from different models

# Data Collection

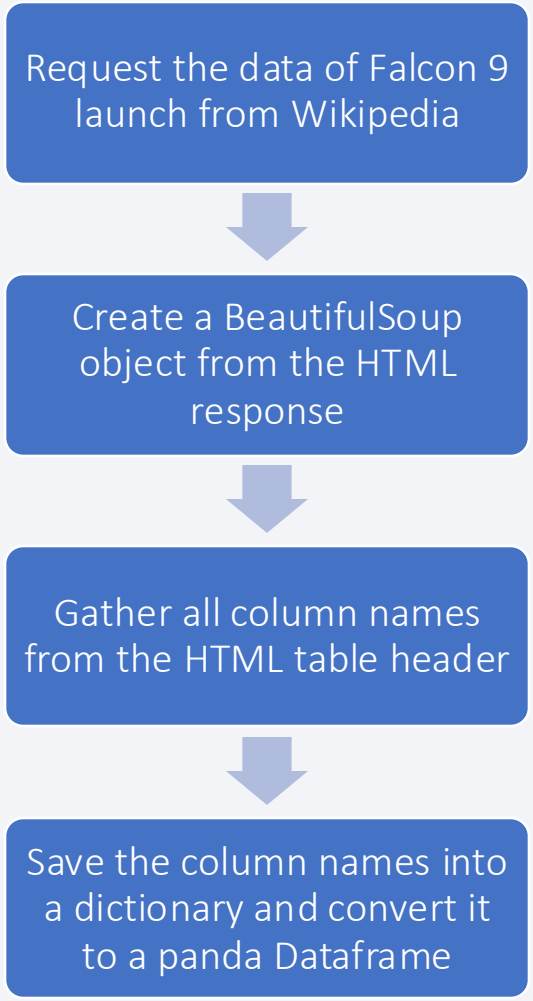
- Data collection is the process of gathering information of interest to study the research questions. In order to get the data for analysis, data collection were mainly completed in 2 different routes, namely:
  - SpaceX REST API
  - Web scrapping from Wikipedia using BeautifulSoup

# Data Collection – SpaceX API




https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/Data%20Collection%20%E2%80%93%20SpaceX%20API%E2%80%8B.ipynb

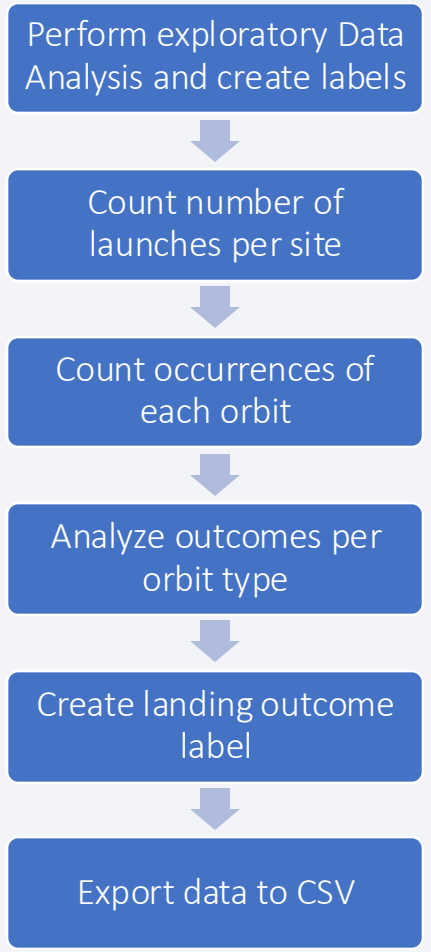
# Data Collection - Scraping




- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/Data%20Collection%20-%20Scraping.ipynb

# Data Wrangling

- In this exercise, different cases of success and failing launches. For example:
  - `True Ocean` successfully landed to a specific region of the ocean
  - `False Ocean` unsuccessfully landed to a specific region of the ocean
  - `True RTLS` successfully landed to a ground pad
  - `False RTLS` unsuccessfully landed to a ground pad
  - `True ASDS` successfully landed on a drone ship
  - `False ASDS` unsuccessfully landed on a drone ship
- We adopted 1 as successfully landed and 0 as unsuccessful to simply label the result
- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/Data%20Wrangling.ipynb



# EDA with Data Visualization

- Charts that were plotted are:
  - Scatter chart for the relationship between Flight Number and Launch Site
  - Scatter chart for the relationship between Payload Mass and Launch Site
  - Bar chart for the relationship between success rate of each orbit type
  - Scatter chart for the relationship between FlightNumber and Orbit type
  - Scatter chart for the relationship between Payload Mass and Orbit type
  - Line chart for the the launch success yearly trend
- These plots allow us to understand how the features correlates into each other and how the correct features could lead to successful landing.
  - Scatter plots demonstrate the correlation between variables. Machine learning tool can be used if relationship is identified.
  - Bar charts show comparison among different groups of data.
  - Line charts help to see trends in data over a period of time.
- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/EDA%20with%20Data%20Visualization.ipynb

# EDA with SQL

- The following are the SQL queries performed:

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first succesful landing outcome in ground pad was acheived.¶
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.
- 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 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.

- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/EDA%20with%20SQL.ipynb

# Build an Interactive Map with Folium

- The following are created and added to a folium map with the CSV file "spacex_launch_geo.csv":
  - Markers with Circle, Popup Label and Text Label for NASA Johnson Space Center and as well as all Launch site
  - Coloured markers and MarkerCluster for the success (Green)  and failed (Red) launches for each site on the map
  - Used coloured lines to display the distances between a launch site to its proximities such as railway, highway, coastline and closest city
- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/Build%20an%20Interactive%20Map%20with%20Folium%E2%80%8B.ipynb

# Build a Dashboard with Plotly Dash

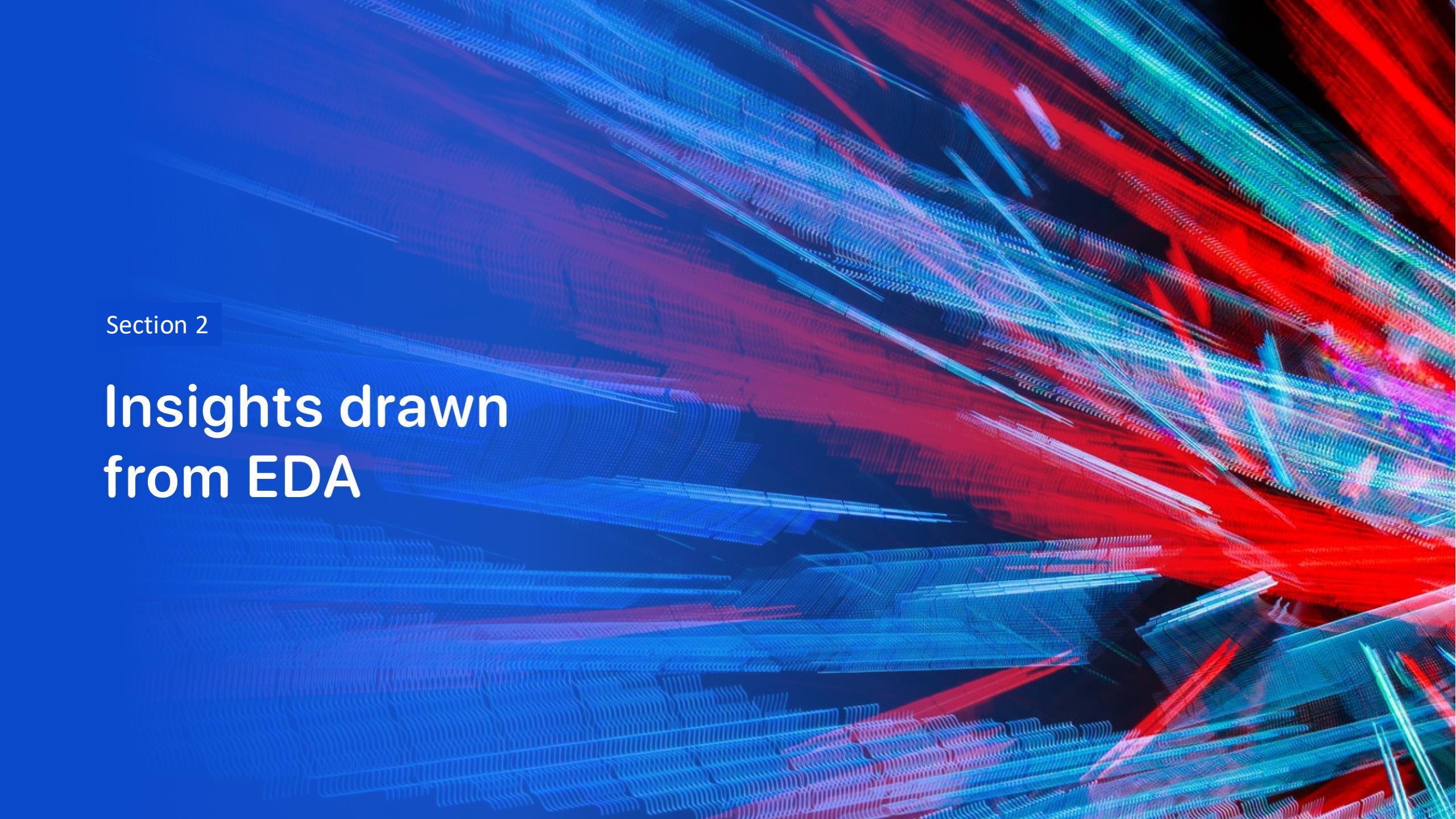
- The following plots/graphs and interactions were added to the dashboard
  - Dropdown list – to allow selection of launch site
  - Pie chart – to display the total successful launches count for all the sites and Success vs Failed counts
  - Range slider – for selecting payload
  - Scatter chart – for payload mass vs success rate for various booster versions
- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/Build%20a%20Dashboard%20with%20Plotly%20Dash.py

# Predictive Analysis (Classification)

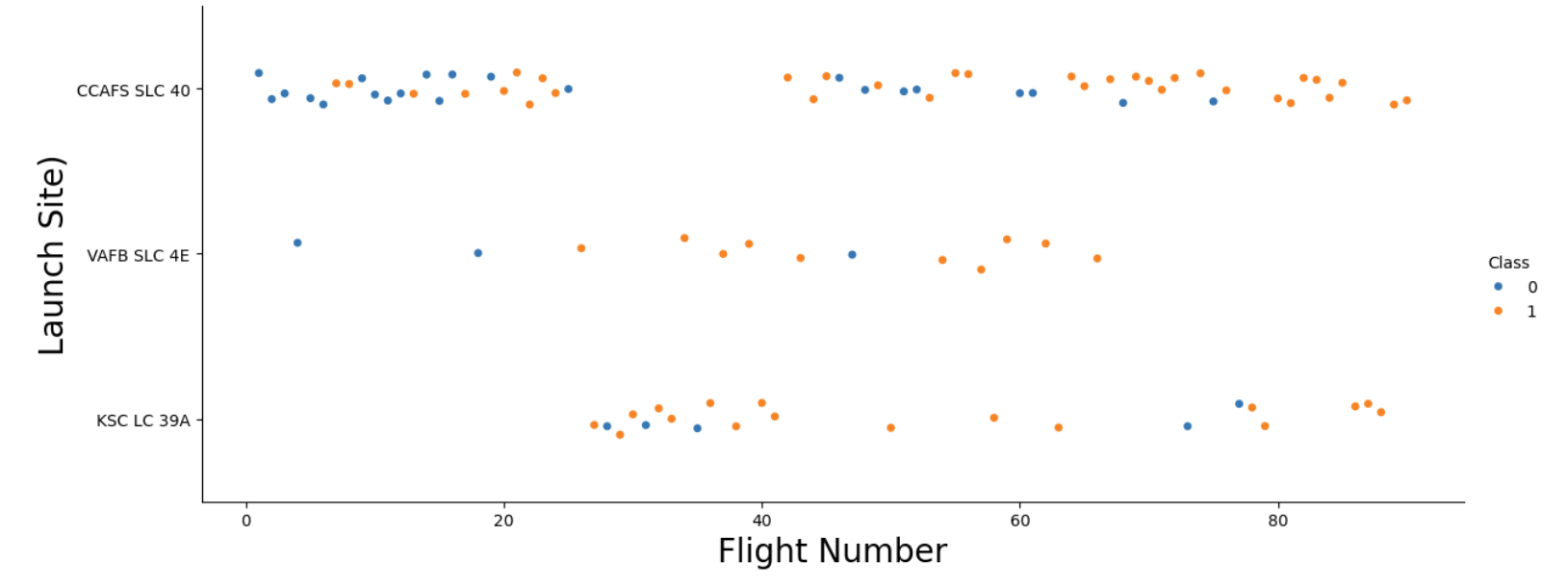
- Model Building – Load dataset into Pandas and Numpy, transform the data and divide them into training and test sets, choose the appropiate ML method and set parameter and algorithms to GridSearch CV
- Model Evaluation – Check accuracy of results and adjust parameters, and plot the confusion matrix
- Model Improvement – Feature Engineering and Algorithm Tuning
- Model Choosing – pick the best model with the highest score
- https://github.com/b210103/Applied-Data-Science-Capstone/blob/main/Predictive%20Analysis%20(Classification).ipynb

# Results

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

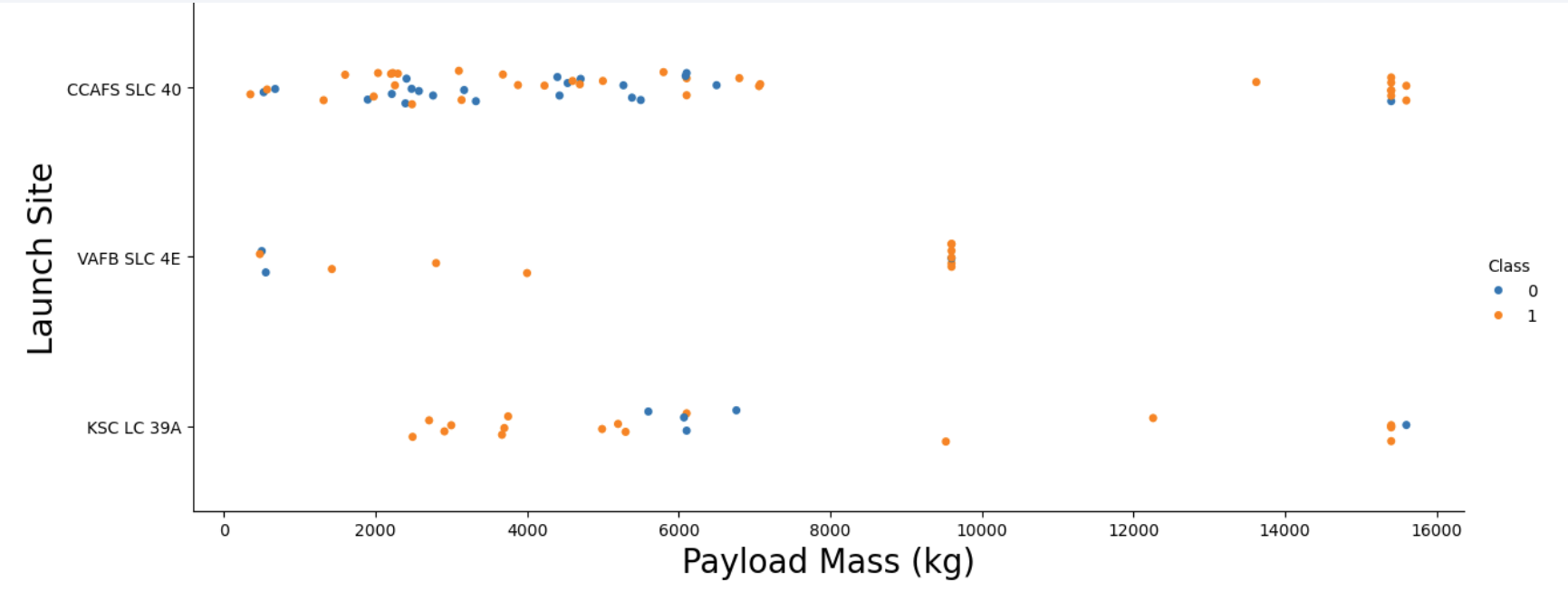

Section 2
Insights drawn from EDA

# Flight Number vs. Launch Site




- Earlier launches were mostly at CCAFS SLC 40
- Most earlier launches failed while the latest flights mostly succeeded
- Locations VAFB SLC 4E and KSC LC 39A have higher success rates.
- The trend shows the recent launches are usually successful

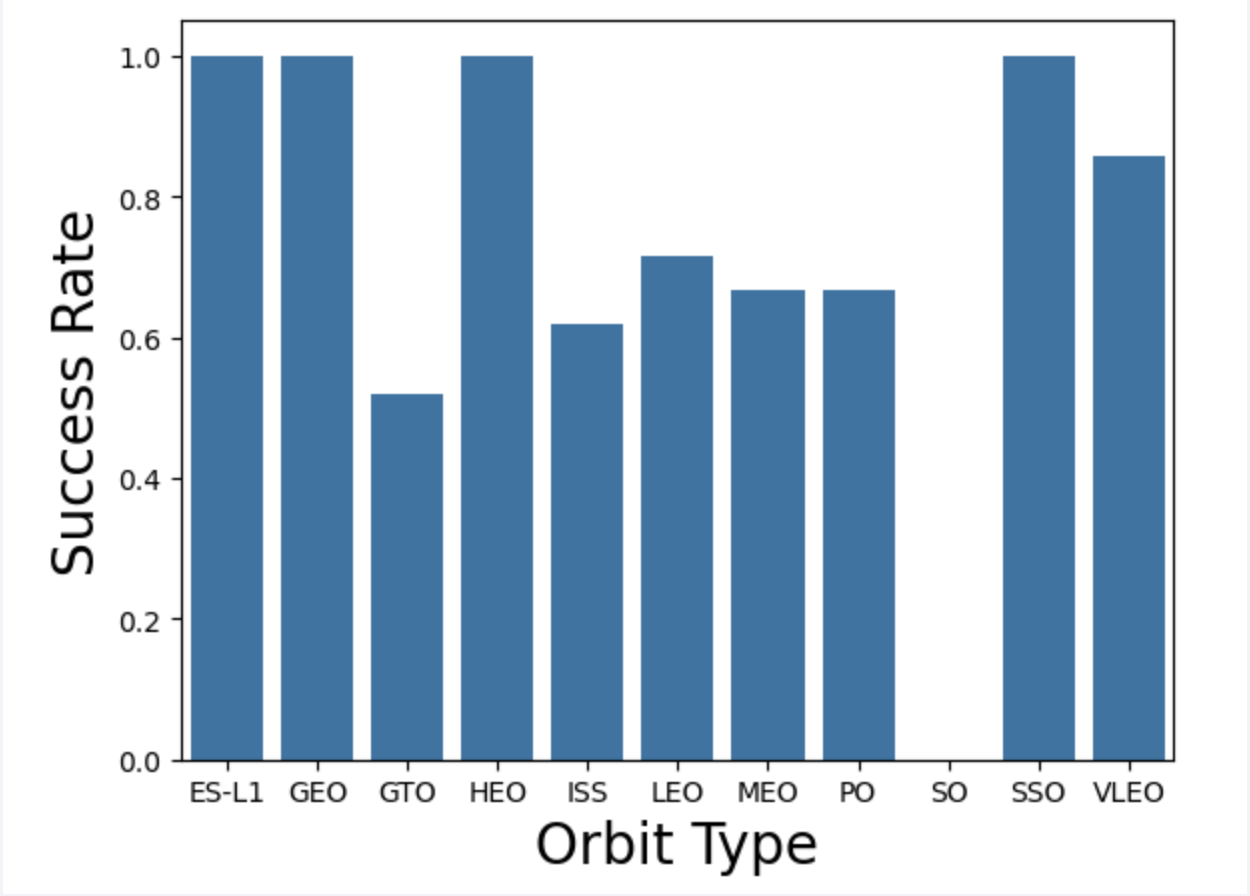
# Payload vs. Launch Site




- Most of the launches were relatively low payload with a mixture of successes and failures
- Medium and high payload flights have high successful rate
- Launches at KSC LC 39A under 5800kg performed extremely well

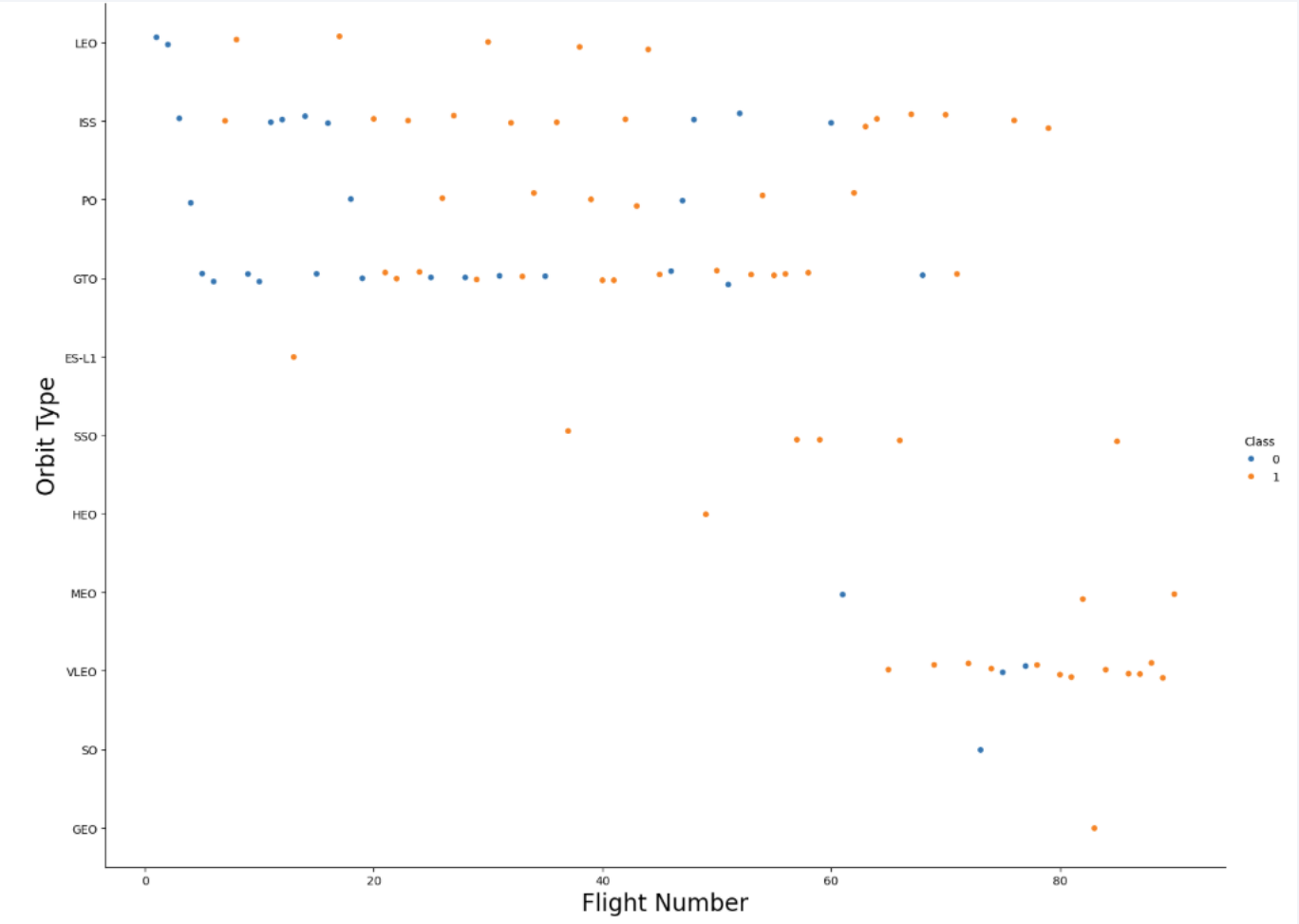
# Success Rate vs. Orbit Type

- No flight had succeeded at SO
- 100% success rate for ES-L1, GEO, HEO and SSO
- Except SO, all the launches had a at least 50% success rate



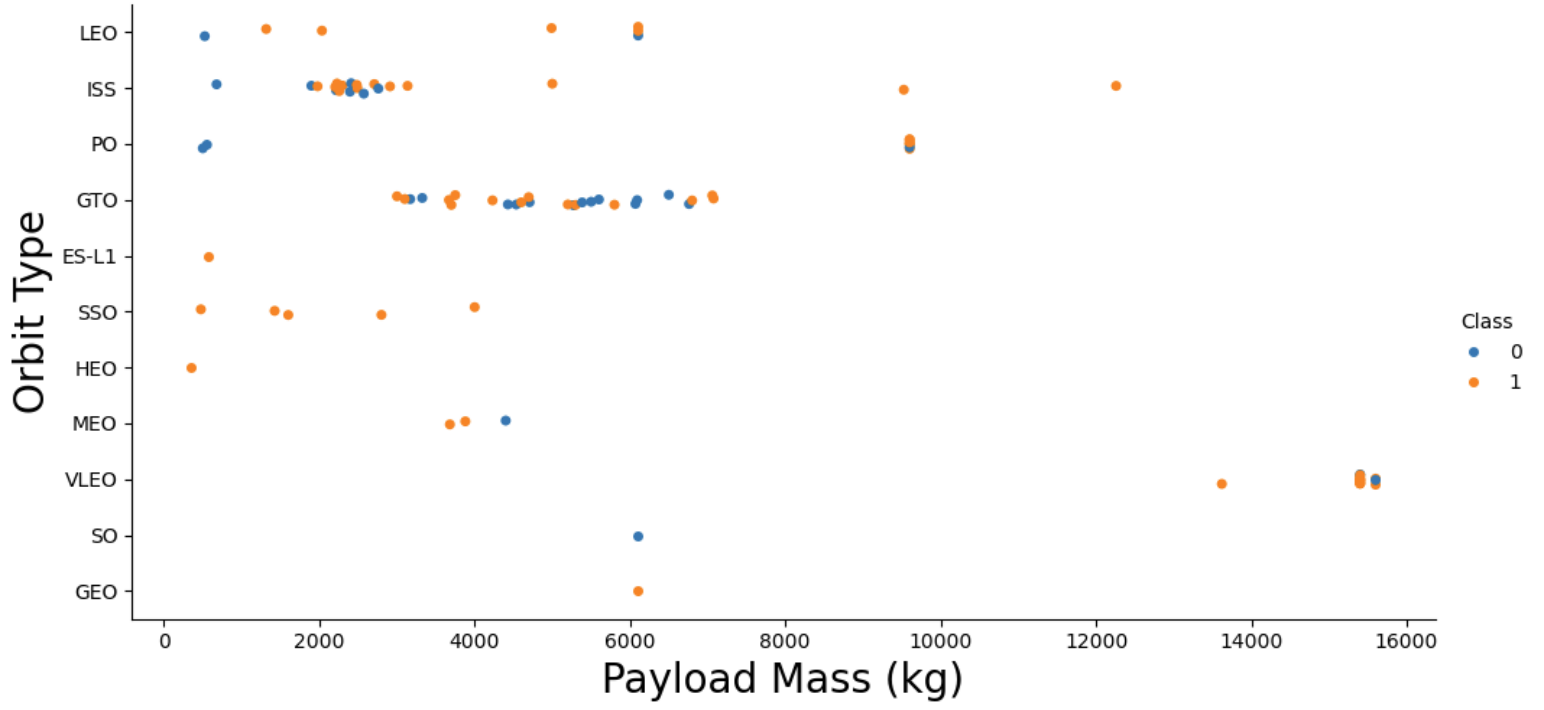
# Flight Number vs. Orbit Type

- Most flights were going to the ISS and GTO orbits
- For the LEO Orbit, it shows postive relationship between success rate and flight numbers.



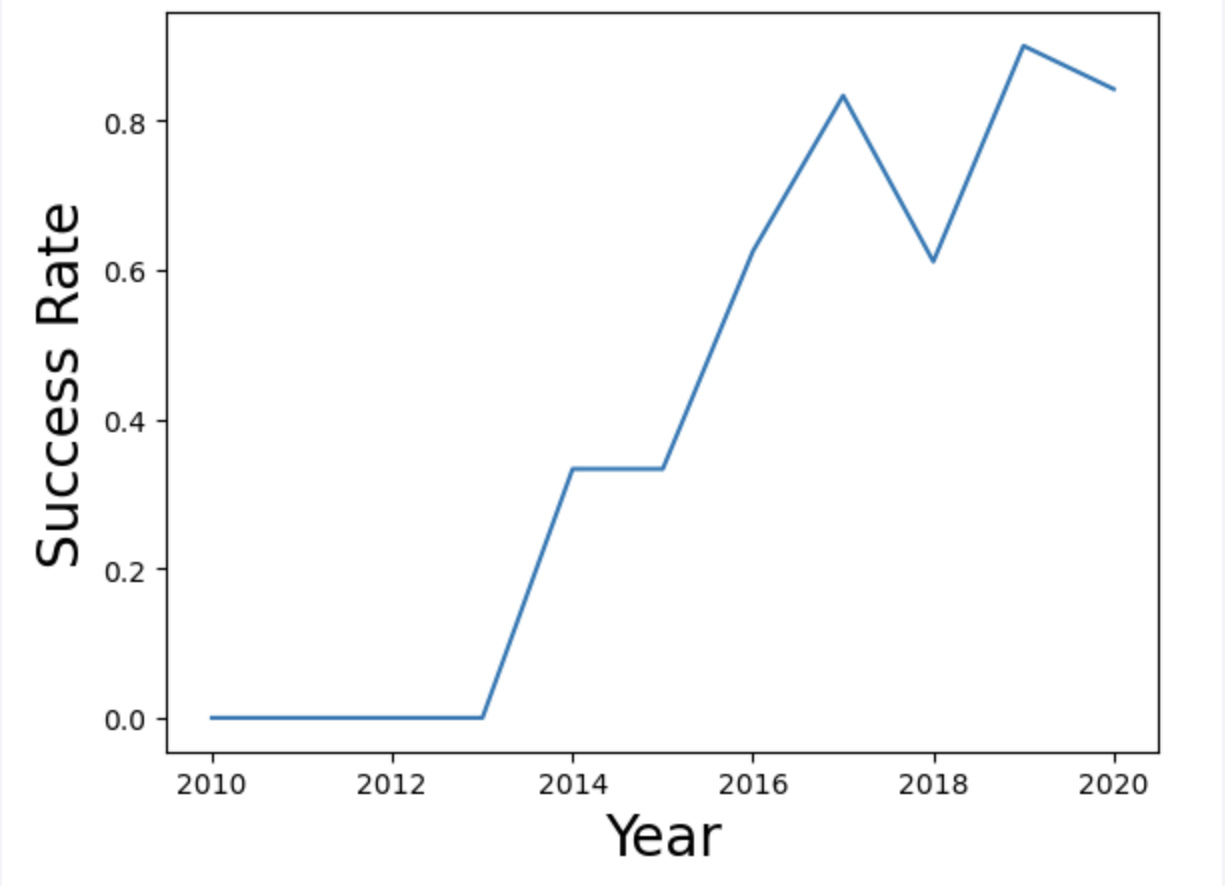
# Payload vs. Orbit Type

- Majority of flights completed under 8000kg
- SSO had a 100% success rate
- VLEO had the heaviest payload flights



# Launch Success Yearly Trend

- Success rate grows as time went along
- Success rate dropped in 2018 and 2020



# All Launch Site Names

```
[13]: q = pd.read_sql('select distinct Launch_Site from spacexdata', conn)
      q
```

[13]:

| | Launch_Site |
|---|---|
| 0 | CCAFS LC-40 |
| 1 | VAFB SLC-4E |
| 2 | KSC LC-39A |
| 3 | CCAFS SLC-40 |

- The result from the query shows the 4 launching sites.

# Launch Site Names Begin with 'CCA'

[14]:
```
q = pd.read_sql("select * from spacexdata where Launch_Site like 'CCA%' limit 5", conn)
q
```

[14]:

| | index | Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_Outcome |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 2010-06-04 | 18:45:00 | F9 v1.0 B0003 | CCAFS LC-40 | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success | Failure (parachute) |
| **1** | 1 | 2010-12-08 | 15:43:00 | F9 v1.0 B0004 | CCAFS LC-40 | Dragon demo flight C1, two CubeSats, barrel of... | 0 | LEO (ISS) | NASA (COTS) NRO | Success | Failure (parachute) |
| **2** | 2 | 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 |
| **3** | 3 | 2012-10-08 | 0:35:00 | F9 v1.0 B0006 | CCAFS LC-40 | SpaceX CRS-1 | 500 | LEO (ISS) | NASA (CRS) | Success | No attempt |
| **4** | 4 | 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 result from the query shows 5 records where launch sites begin with `CCA`

# Total Payload Mass

```
[15]: q = pd.read_sql("select sum(PAYLOAD_MASS__KG_) from spacexdata where Customer='NASA (CRS)'", conn)
      q
```

[15]:

| | sum(PAYLOAD_MASS__KG_) |
|---|---|
| 0 | 45596 |

- The total payload carried by boosters from NASA was calculated.

# Average Payload Mass by F9 v1.1

```
[18]: q = pd.read_sql("select avg(PAYLOAD_MASS__KG_) from spacexdata where Booster_Version='F9 v1.1'", conn)
      q
```

[18]:

| | avg(PAYLOAD_MASS__KG_) |
|---|---|
| 0 | 2928.4 |

- The average payload mass carried by boosters version F9 v1.1 was calculated.

# First Successful Ground Landing Date

```
[21]: q = pd.read_sql("select min(Date) from spacexdata where Landing_Outcome='Success (ground pad)'", conn)
      q
```

[21]:

| | min(Date) |
|---|---|
| 0 | 2015-12-22 |

- The result from the query shows the dates of the first successful landing outcome on ground pad

# Successful Drone Ship Landing with Payload between 4000 and 6000

```
[22]: q = pd.read_sql("select distinct Booster_Version from spacexdata where Landing_Outcome='Success (drone ship)' and PAYLOAD_MASS__KG_ between 4000 and 6000", conn)
      q
```

[22]:

| | Booster_Version |
|---|---|
| 0 | F9 FT B1022 |
| 1 | F9 FT B1026 |
| 2 | F9 FT B1021.2 |
| 3 | F9 FT B1031.2 |

- This query result lists the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

# Total Number of Successful and Failure Mission Outcomes

```
[23]: q = pd.read_sql("select substr(Mission_Outcome,1,7) as Mission_Outcome, count(*) from spacexdata  group by 1", conn)
      q
```

[23]:

| | Mission_Outcome | count(*) |
|---|---|---|
| 0 | Failure | 1 |
| 1 | Success | 100 |

- The result from the query shows the total number of successful and failure mission outcomes

# Boosters Carried Maximum Payload

```
[24]: q = pd.read_sql("select distinct Booster_Version from spacexdata where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from spacexdata)", conn)
      q
```

[24]:

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

- Tthe names of the booster which have carried the maximum payload mass were listed here.

# 2015 Launch Records

```
[27]: q = pd.read_sql("select distinct Landing_Outcome, Booster_Version, Launch_Site from spacexdata where Landing_Outcome='Failure (drone ship)'", conn)
      q
```

[27]:

| | Landing_Outcome | Booster_Version | Launch_Site |
|---|---|---|---|
| 0 | Failure (drone ship) | F9 v1.1 B1012 | CCAFS LC-40 |
| 1 | Failure (drone ship) | F9 v1.1 B1015 | CCAFS LC-40 |
| 2 | Failure (drone ship) | F9 v1.1 B1017 | VAFB SLC-4E |
| 3 | Failure (drone ship) | F9 FT B1020 | CCAFS LC-40 |
| 4 | Failure (drone ship) | F9 FT B1024 | CCAFS LC-40 |

- The above shows the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

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

```
[28]: q = pd.read_sql("select Landing_Outcome, count(*) from spacexdata where Date between '2011-06-04' and '2017-03-20' group by Landing_Outcome order by 2 desc", conn)
      q
```

[28]:

| | Landing_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 | Precluded (drone ship) | 1 |

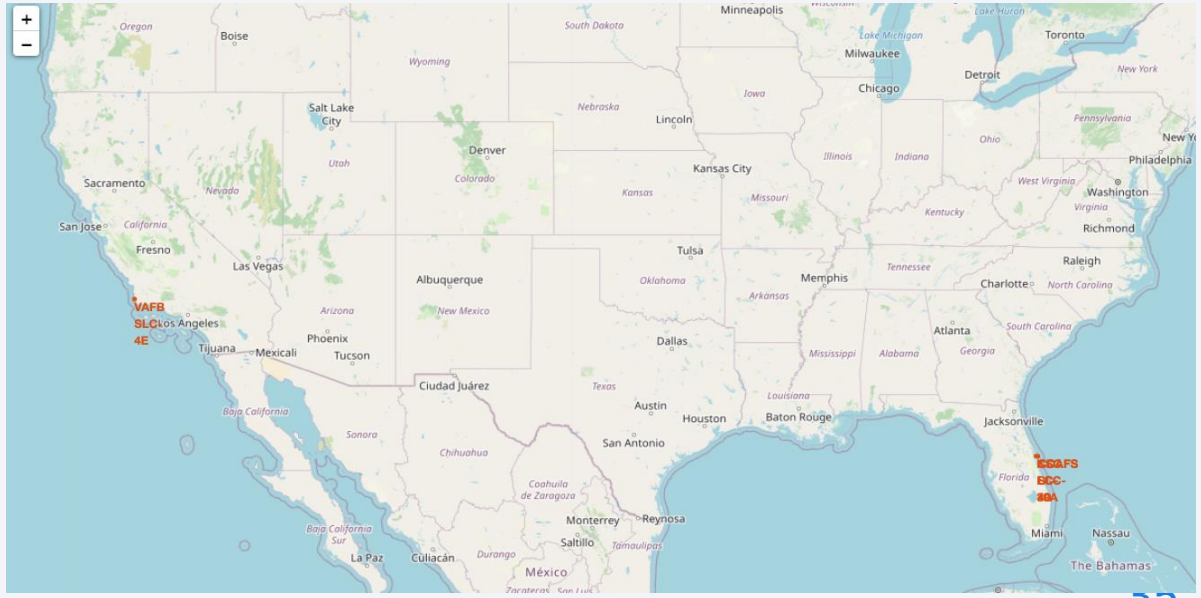
- This is the ranking of the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

Section 3

# Launch Sites Proximities Analysis

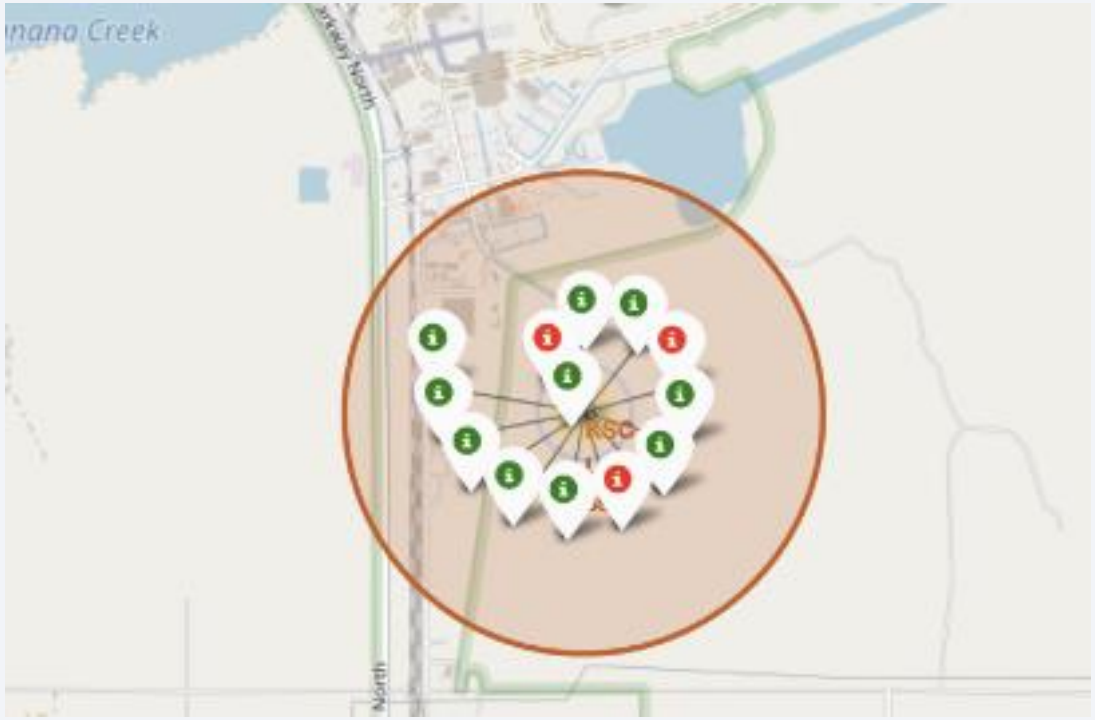
# All launch sites' location on global map using Folium

- The launching locations were chosen to be on the east coast (Florida) and the west coast (California).
- The launching location is near the equator.
- The launching location is near coastline



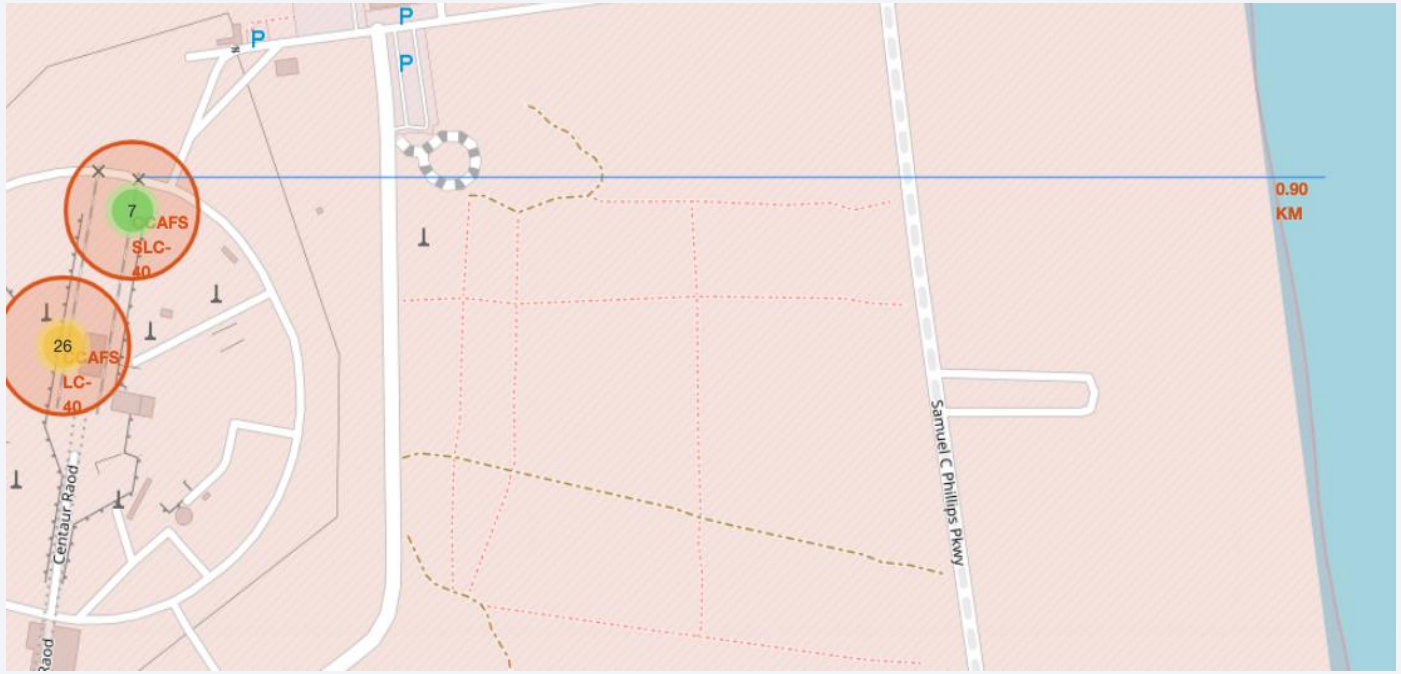
# Marking success/failed launches for each site

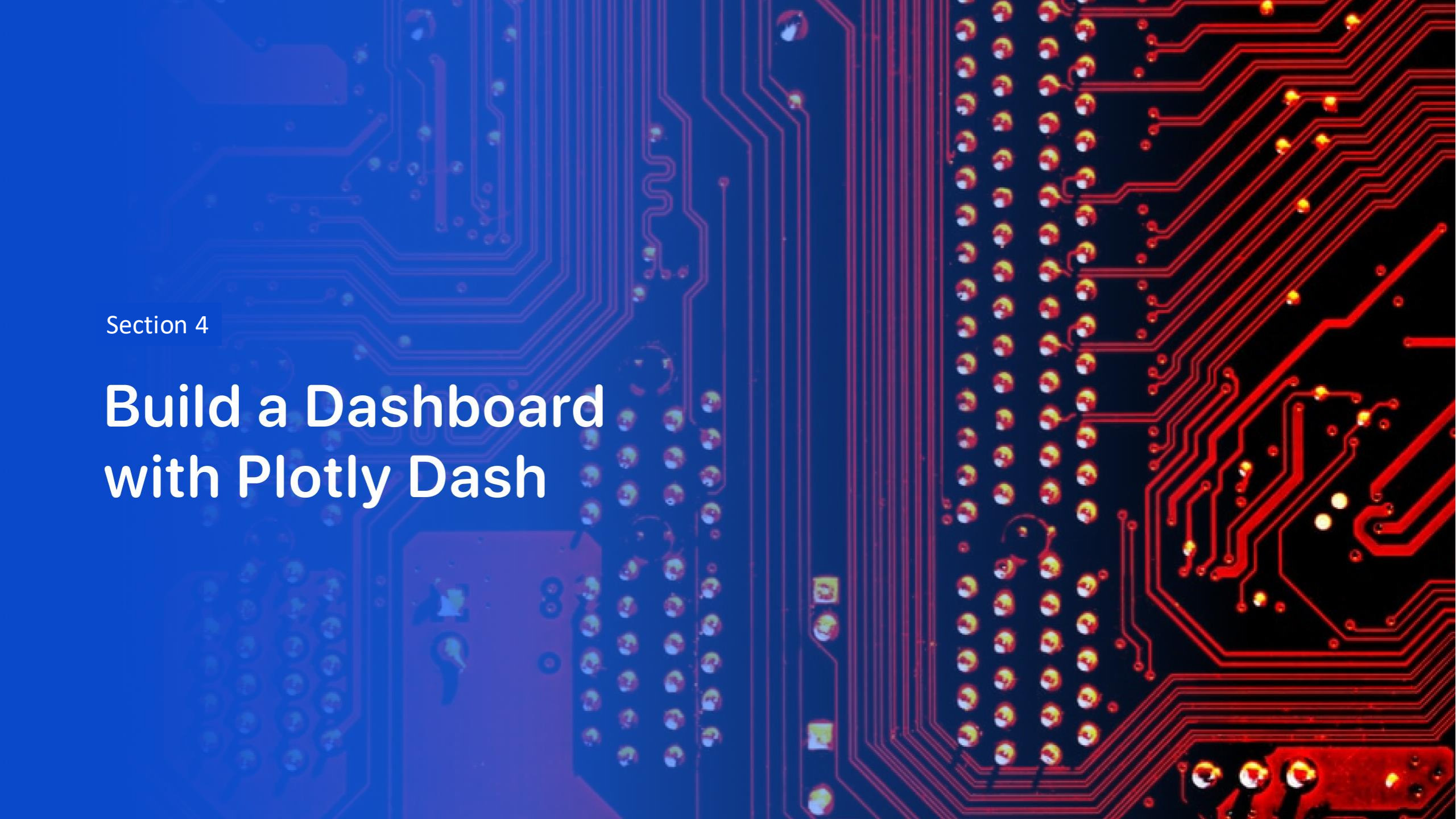
- From this technique we can understand the total number launches at a site and easily differentiate the successful flights (green) and the failing flights (red)



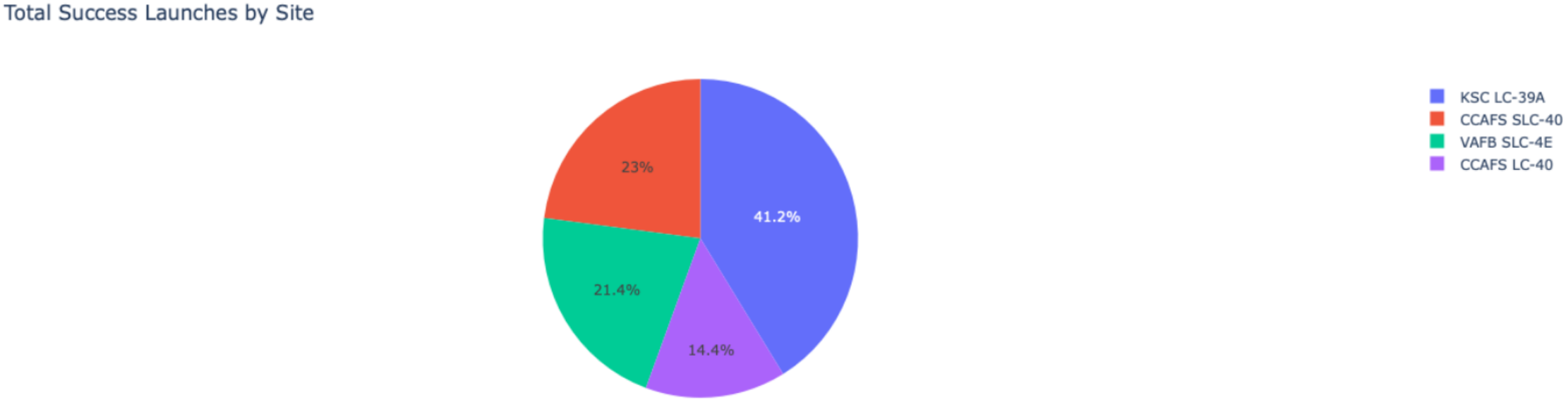
# Measuring distance between launch sites and other places

- As shown in the photo we can measure the distance on Folium, thus we are able to measure the distance of nearby facilities (such as railway, highway and coastline) in relation to the launch sites.



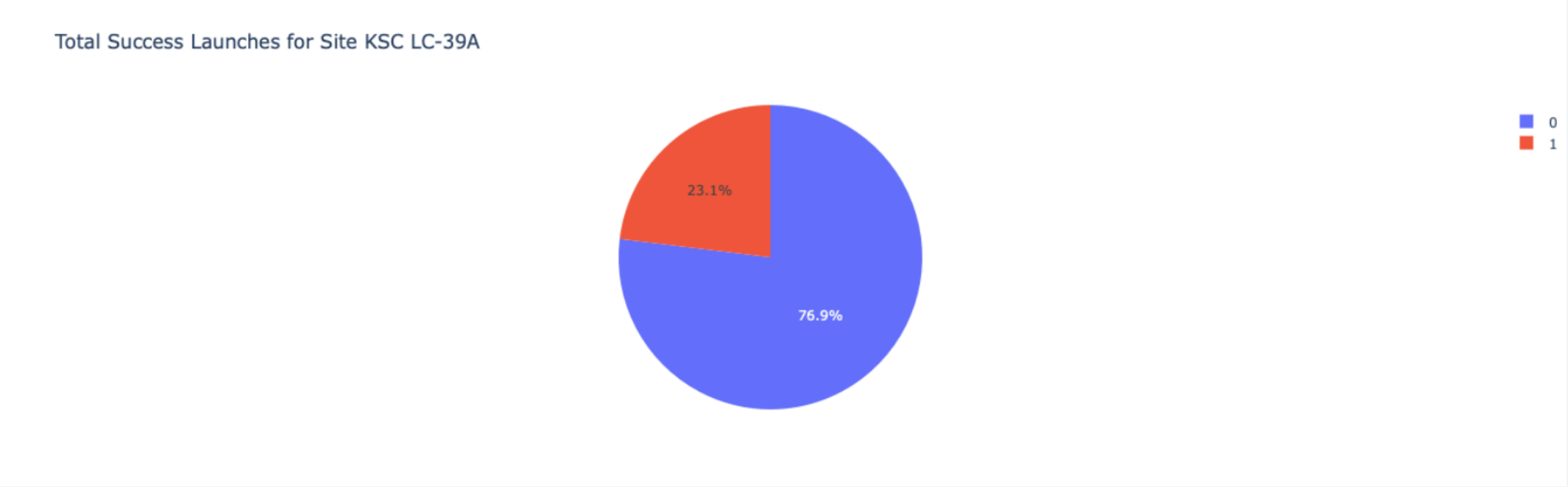

Section 4
Build a Dashboard with Plotly Dash

# Successful Launches by Site




- This pie chart shows the successful launches by site, we can that KSC LC-39A has the most successful launches.

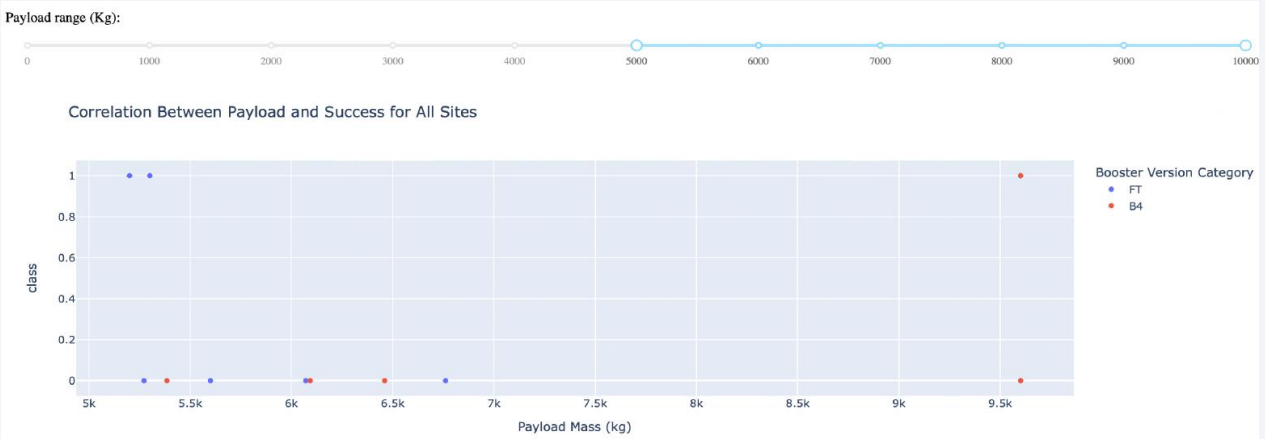
# Total Successful Launches for Site KSC LC-39A




- This shows the launch successful rate (76.9%) at KSC LC-39A

# Payload vs. Launch Outcome scatter plot for all sites

- These 2 scatter chart tell that lower payload might contribute higher launch successful rate. In particular, payload between 2000kg to 5000kg has the highest rate of success.

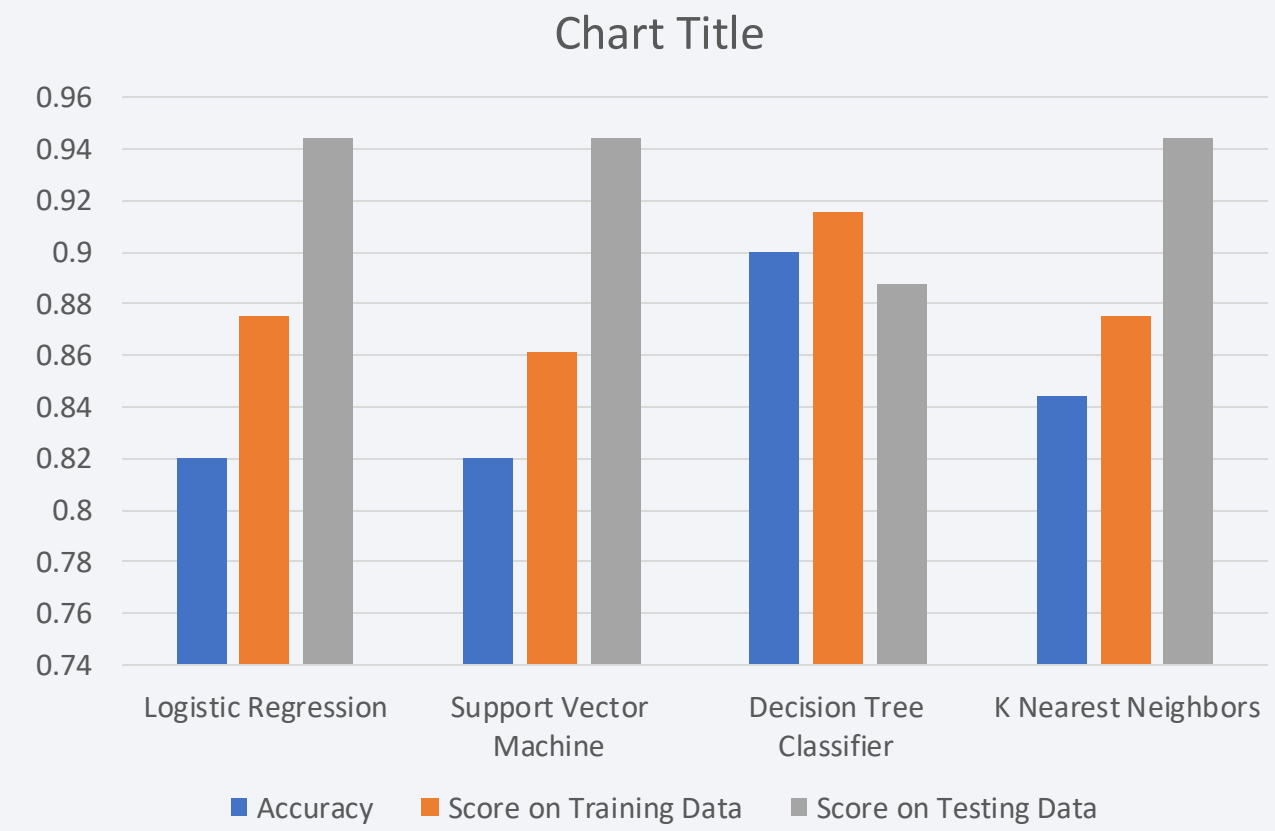



Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

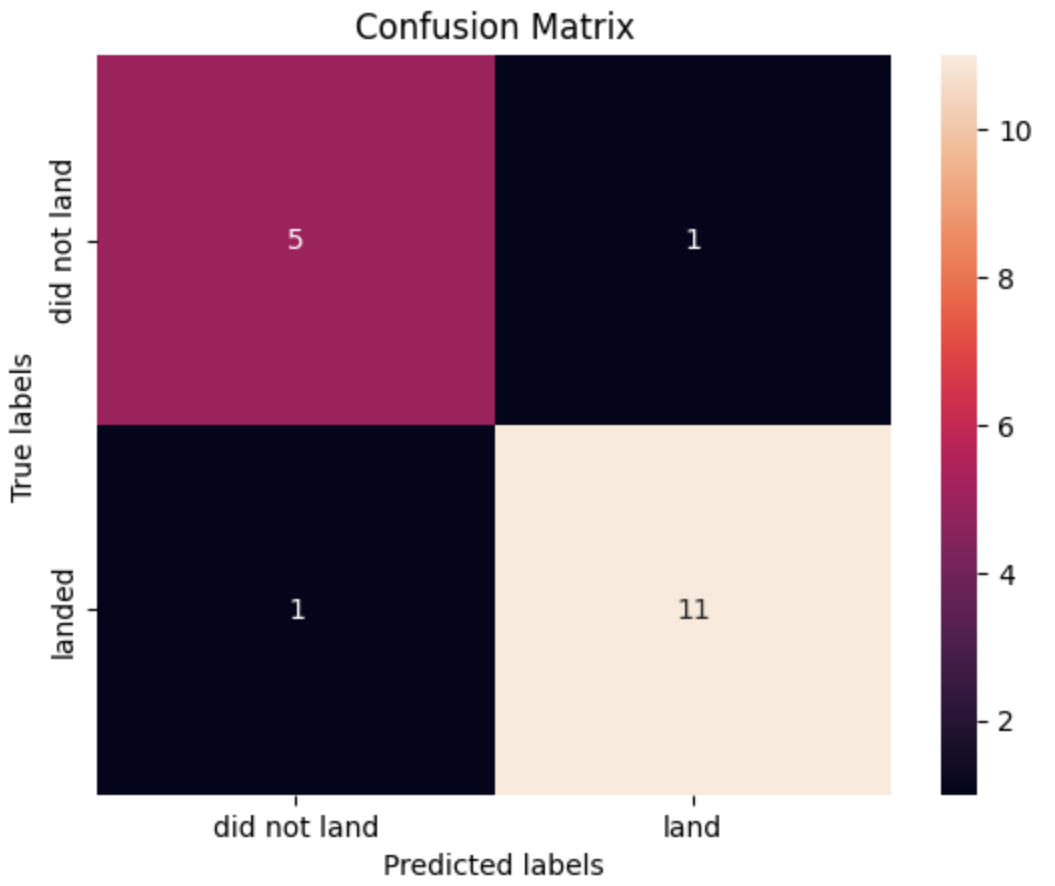
- This bar chart compares the four models
- From the accuracy we can tell that Decision Tree has the most accurate result



# Confusion Matrix

- The confusion matrix of the best performing model (Decision Tree) shows 11 True Positives, 5 True Negatives, 1 False Positive and 1 False Negative

```
[25]: yhat = tree_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



# Conclusions

- Decision Tree is the best model.
- Orbits GEO, HEO, SSO and ES-L1 have perfect launch successful rate.
- In general, the lower the payload, the more success us the launches
- The success rate climb up over the years.
- Most launch location are next to the coastline and near Equator.
- KSC LC-39A has the highest success rates of all launching sites.

# Appendix

- https://www.coursera.org/learn/applied-data-science-capstone/home/welcome
- https://github.com/b210103/Applied-Data-Science-Capstone

# Thank you!