IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Carlos Santana
02-01-2025

Outline

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

Executive Summary

- **Summary of methodologies**
  - Data collected through SpaceX API and web-scraping.
  - Explored data using SQL, visualization, folium maps, and dashboards.
- **Summary of all results**
  - Machine Learning model predicted if the Falcon 9 first stage will land successfully.
  - All models over predicted successful landings
  - Detailed results will be provided though out the presentation.

Introduction

- **Project background and context**
- The launch of the Falcon 9 cost 62 million dollars according to advertising Competitors charge more than $165 million for similar products The key to saving costs: SpaceX renews
- **Problems you want to find answers**
- Need a model able to predict the success of a landing to calculate the cost

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
    - **Combined data from SpaceX public API and SpaceX Wikipedia page**
- Perform data wrangling
    - **Classifying true landings as successful and unsuccessful otherwise**
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - **Tuned models using GridSearchCV**
    - **Data collected were normalized, divided into training and test data sets, and evaluated by four different classification models. The accuracy of each model was assessed using various combinations of parameters.**

Data Collection

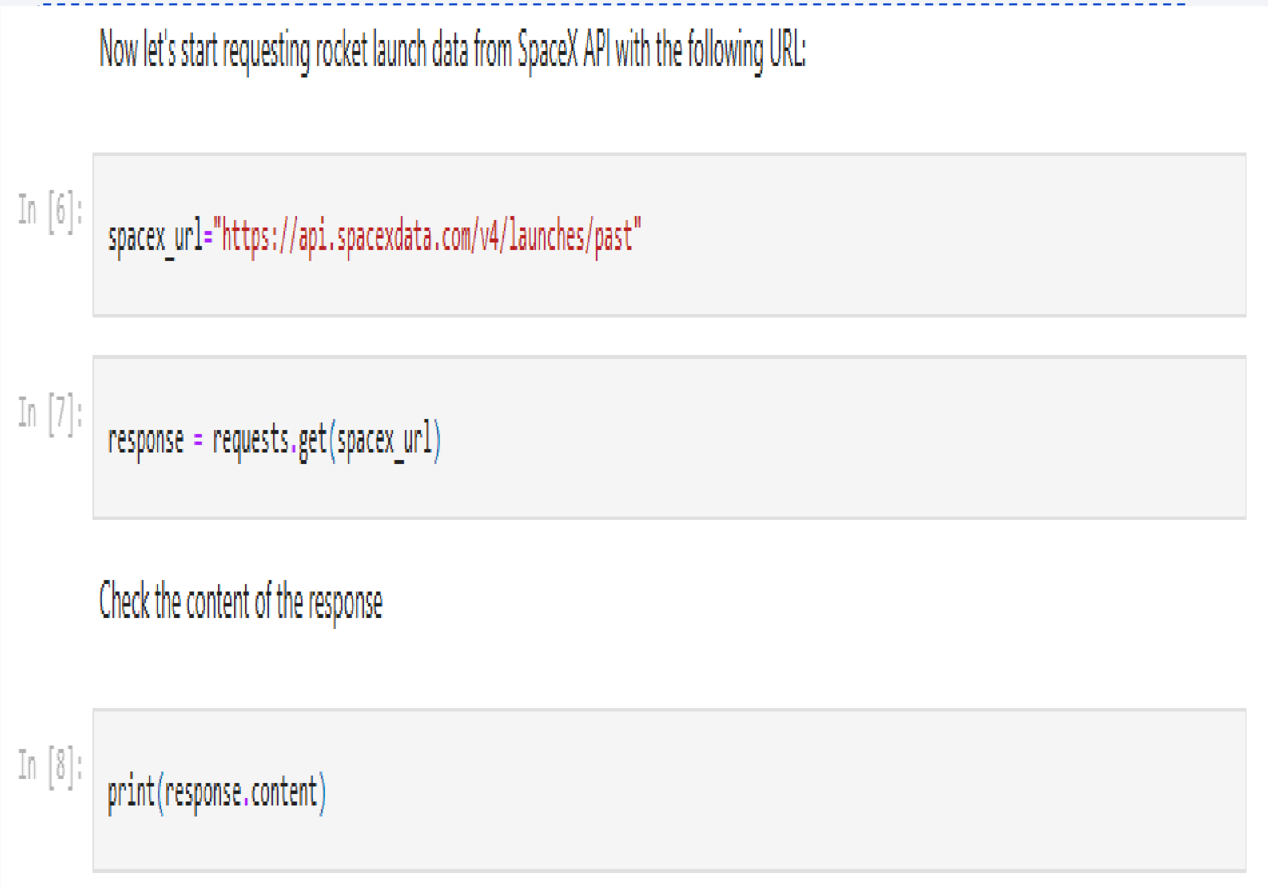
- Describe how data sets were collected.
  - Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().
  - We then cleaned the data, checked for missing values and fill in missing values where necessary.
  - In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.
  - The objective was to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for future analysis.

Data Collection – SpaceX API

- We used the get request to the SpaceX API to collect data, clean the requested data and did some basic data wrangling and formatting.

- Add the GitHub URL of the completed SpaceX API calls notebook:

- https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Now let's start requesting rocket launch data from SpaceX API with the following URL:

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

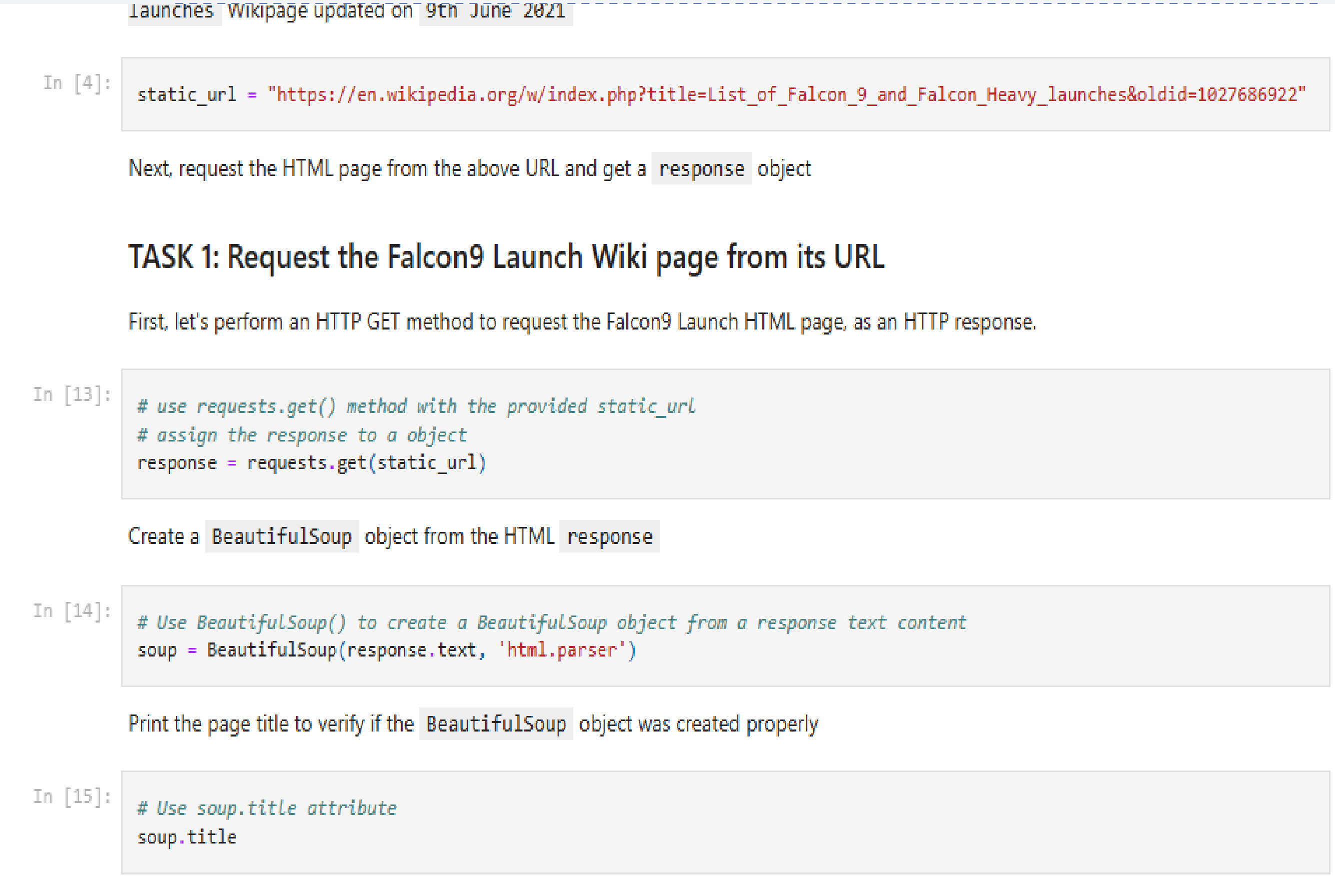
In [7]:
```
response = requests.get(spacex_url)
```

Check the content of the response

In [8]:
```
print(response.content)
```

Data Collection - Scraping

- We applied web scrapping to webscrap Falcon 9 launch records with BeautifulSoup
- We parsed the table and converted it into a pandas dataframe.
- Add the GitHub URL of the completed web scraping notebook: https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb



launches Wikipage updated on 9th June 2021

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

Next, request the HTML page from the above URL and get a `response` object

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

In [13]:
```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url)
```

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

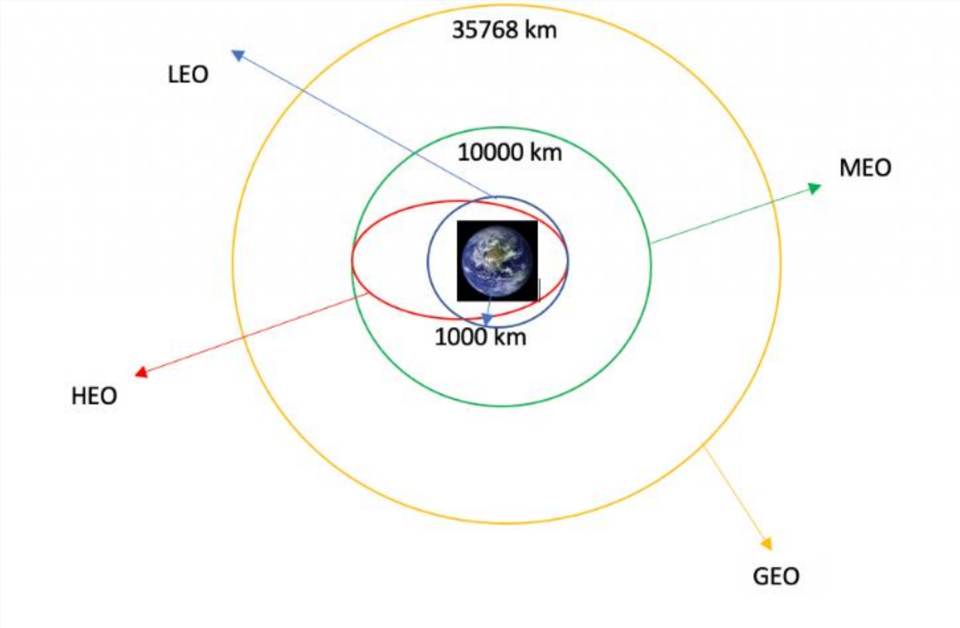
In [14]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

In [15]:
```
# Use soup.title attribute
soup.title
```

Data Wrangling

- We performed exploratory data analysis and determined the training labels.
- We created landing outcome label from outcome column and exported the results to csv.
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose:
- https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose:
- https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/edadataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose
- https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite%20(1).ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- Explain why you added those objects
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose
- https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose
- https://github.com/CARLOS2020-UNAB/Applied-Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

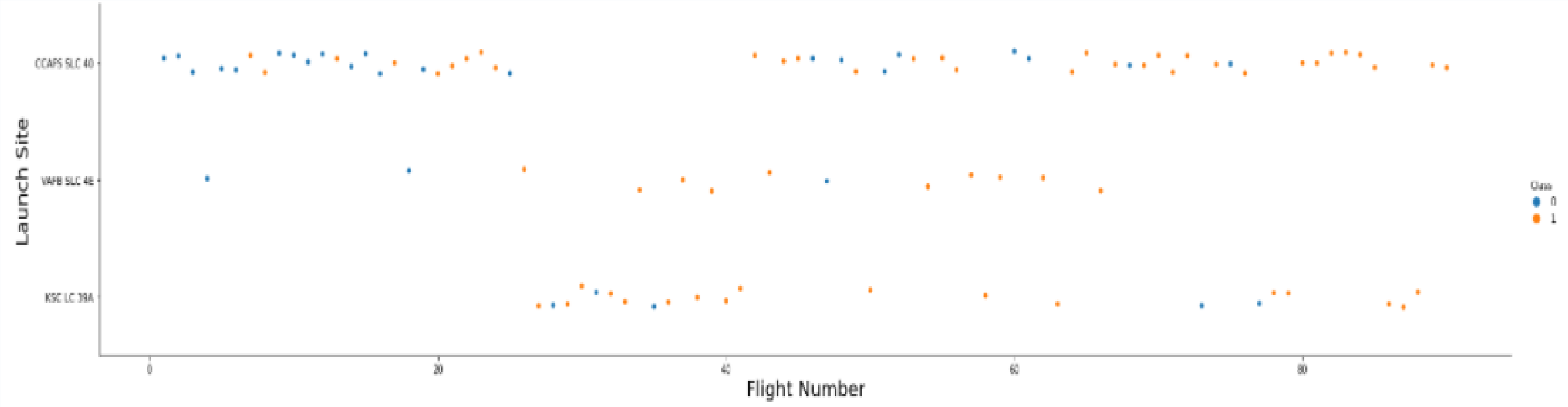
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

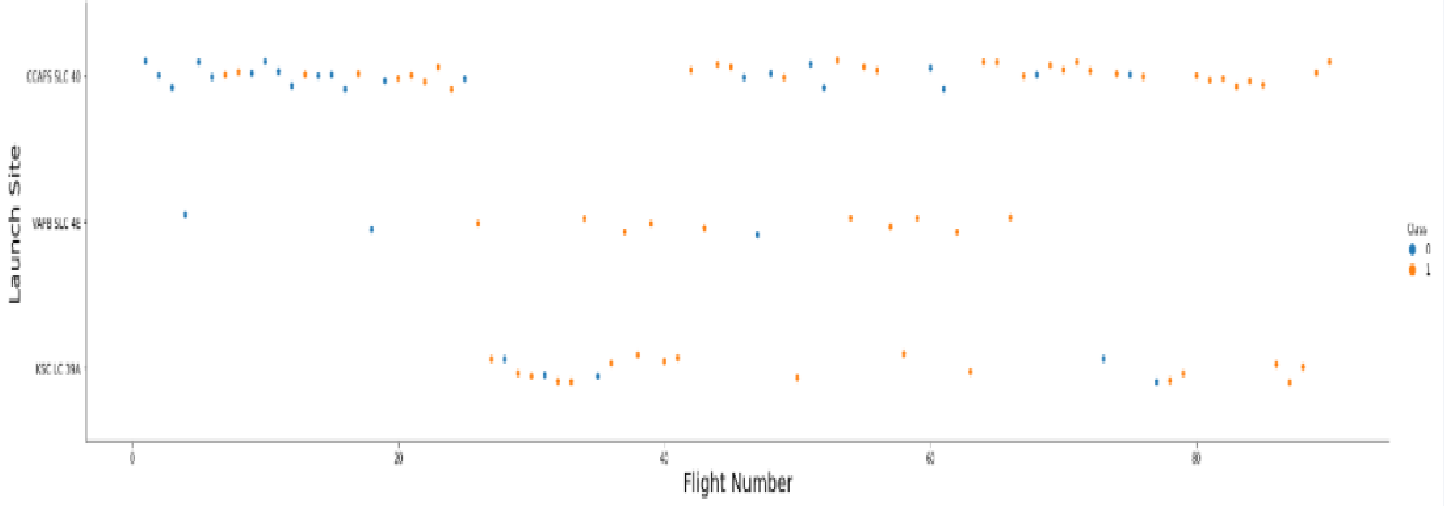
Flight Number vs. Launch Site

- From the plot, we found that the larger the flight amount at a launch site, the greater the success rate at a launch site.



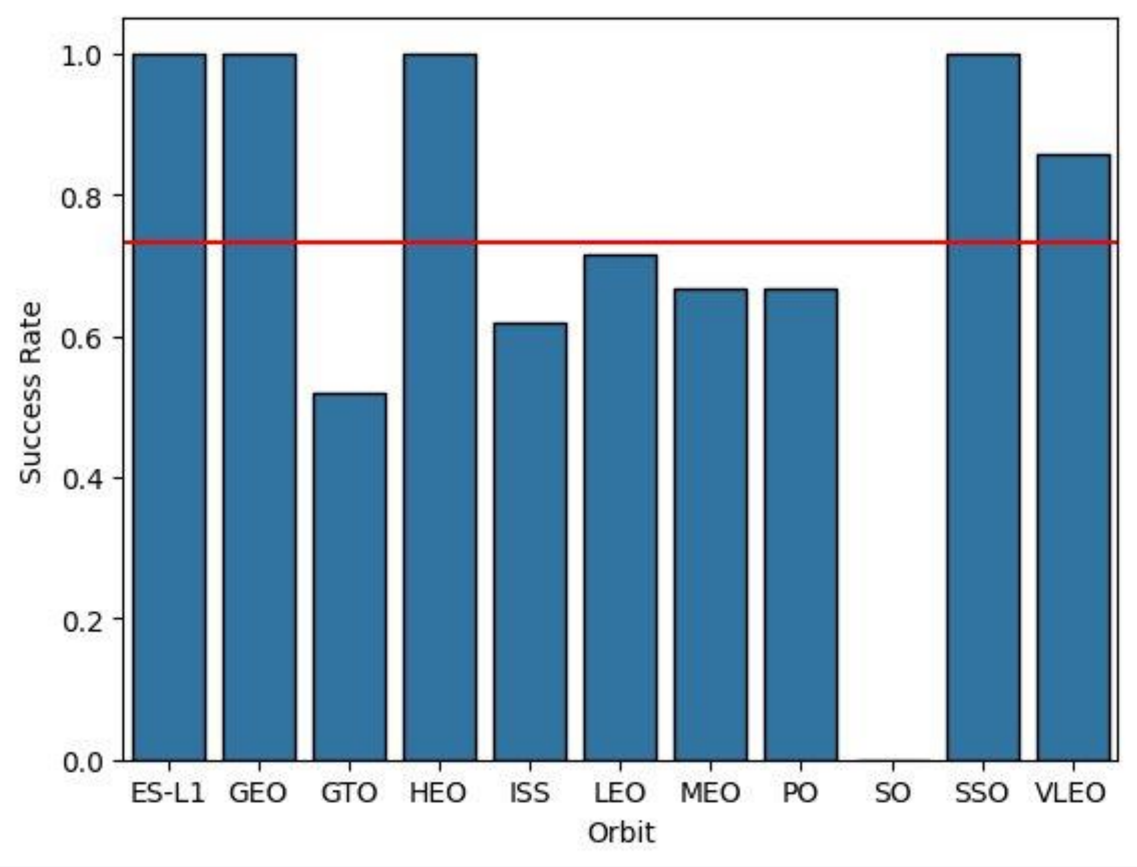
Payload vs. Launch Site

- VAFB-SLC launch site there are no rockets launched for heavy payload mass (greater than 10000).
- The higher the payload, the higher the success rate.

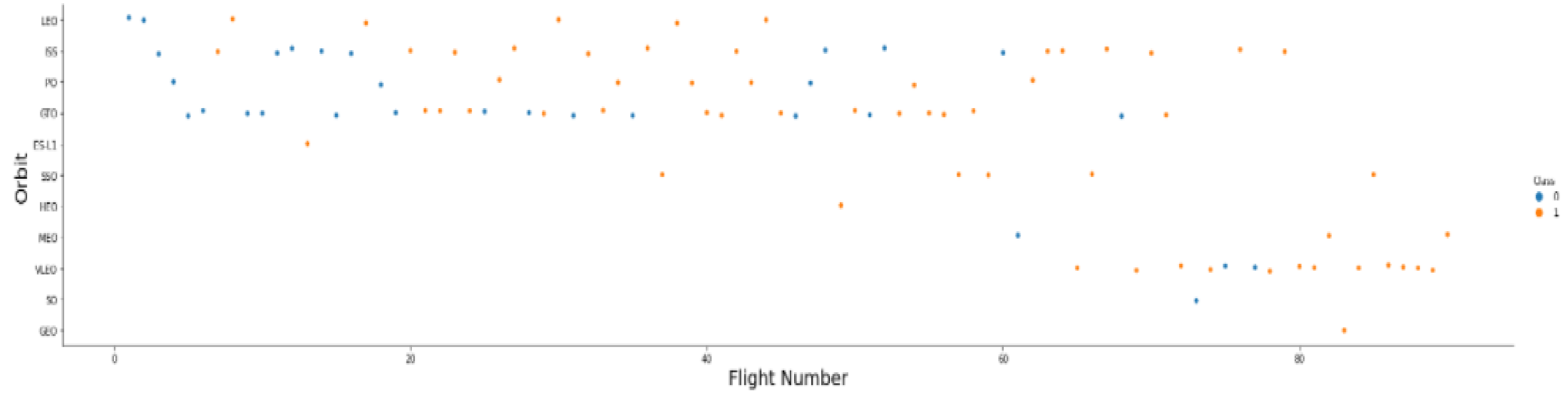


Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO and VLEO orbits have above the average success rate in stage 1 landing.
- ES-L1, GEO, HEO and SSO have 100% success rate in stage 1 landing.
- SO has no successful stage 1 landing



Flight Number vs. Orbit Type

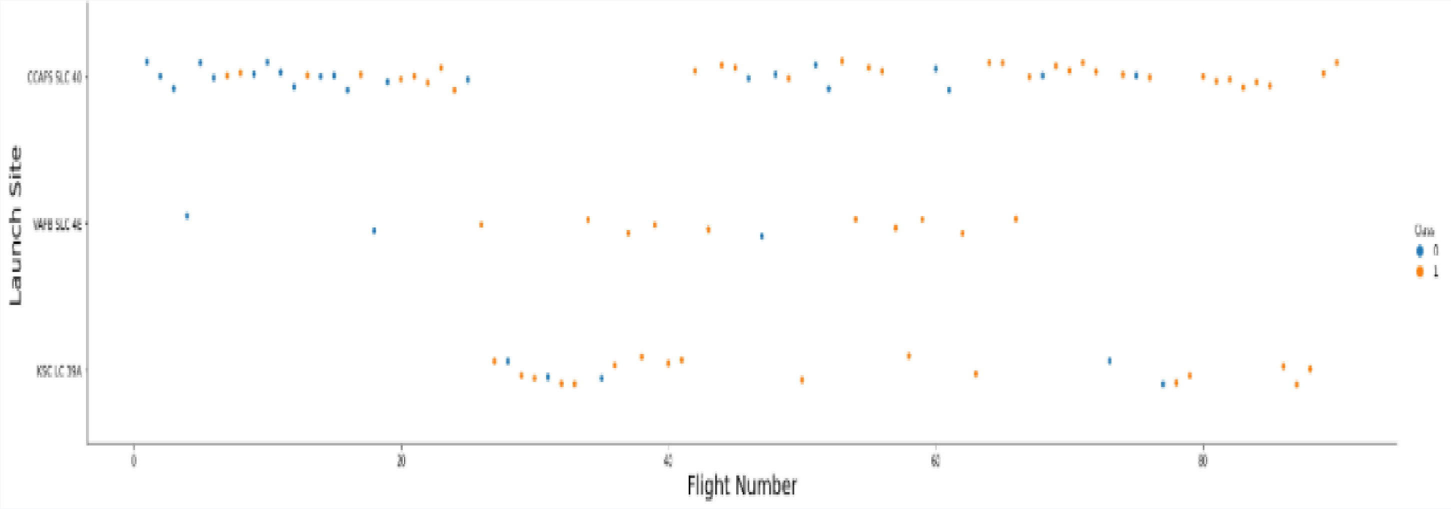


- The plot below shows the Flight Number vs. Orbit type. We observe that in the LEO orbit, success is related to the number of flights whereas in the GTO orbit, there is no relationship between flight number and the orbit.

- SO has 100% failure rate, but only one launch.

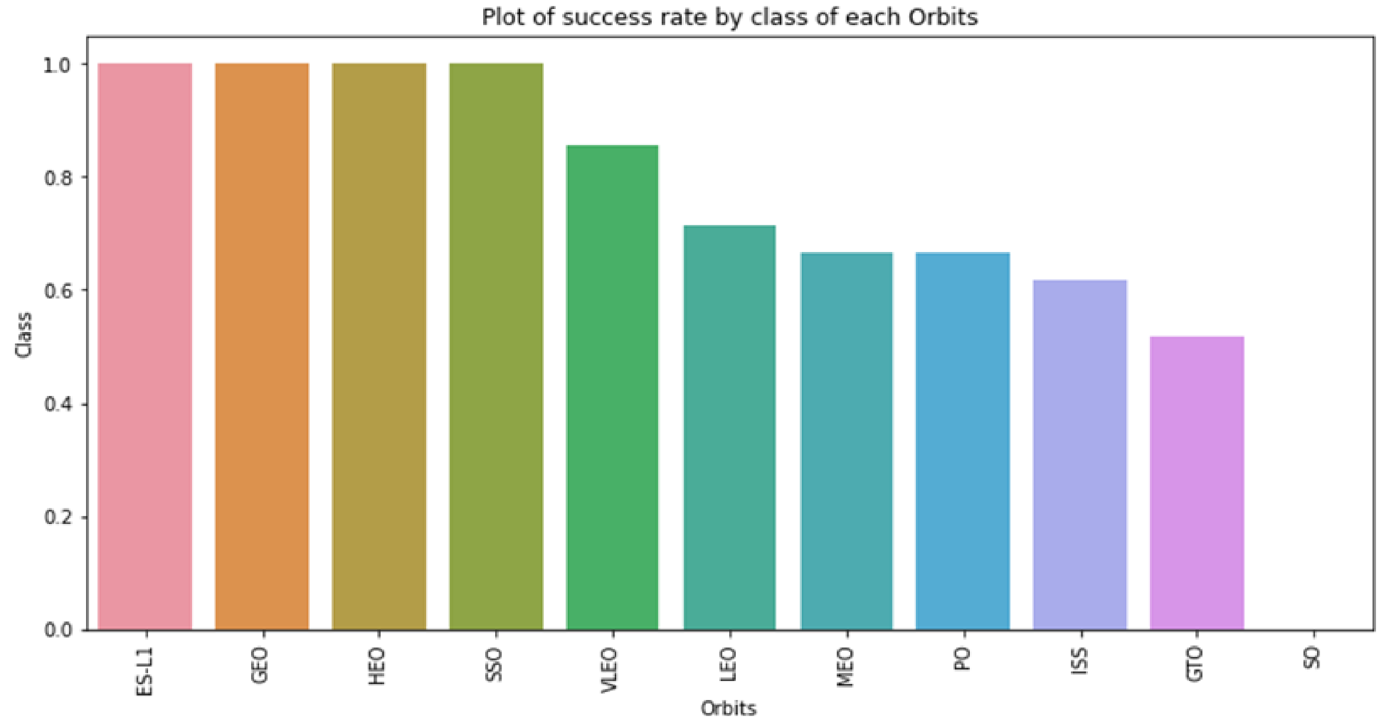
Payload vs. Orbit Type

- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- For GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.



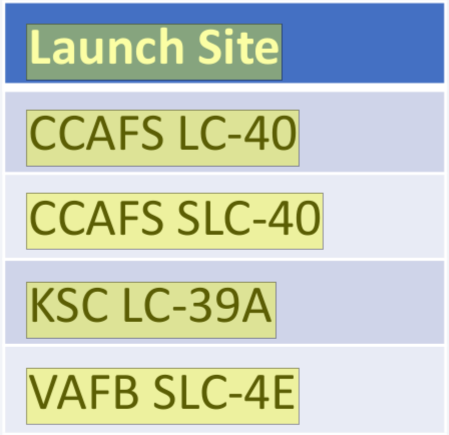
Launch Success Yearly Trend

- From the plot, we can see that ES-L1, GEO, HEO, SSO, VLEO had the most success rate.



All Launch Site Names

- According to data, there are four launch sites:



Launch Site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E

- They are obtained by selecting unique occurrences of “launch_site” values from the dataset.

Launch Site Names Begin with 'CCA'

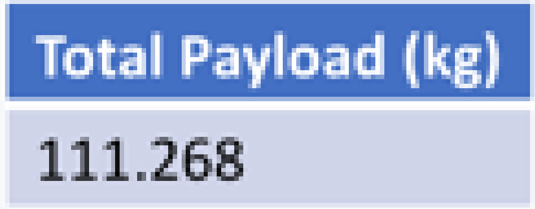
- 5 records where launch sites begin with `CCA`:

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	**CCA**FS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	**CCA**FS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	**CCA**FS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	**CCA**FS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	**CCA**FS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attemp

- Here we can see five samples of Cape Canaveral launches.

Total Payload Mass

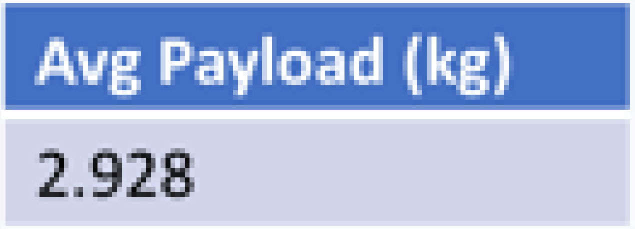
- Total payload carried by boosters from NASA:



Total Payload (kg)
111.268

Average Payload Mass by F9 v1.1

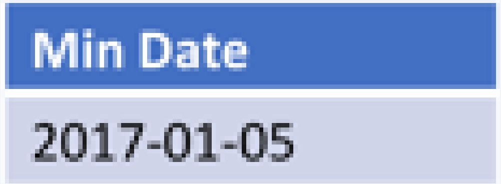
- Calculate the average payload mass carried by booster version F9 v1.1
- Payload mass we obtained the value of 2,928 kg



Avg Payload (kg)
2.928

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here:

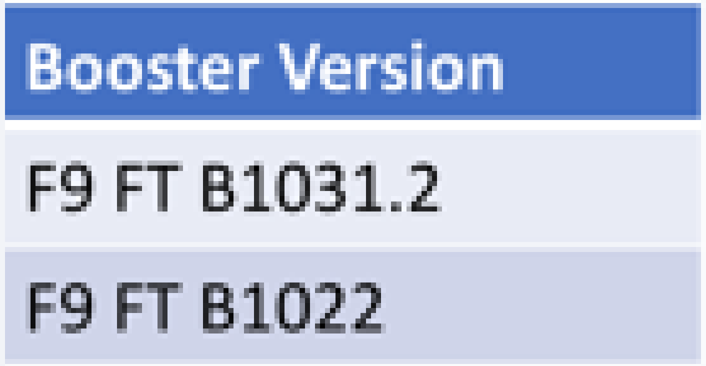


Min Date
2017-01-05

Successful Drone Ship Landing with Payload between 4000 and 6000

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

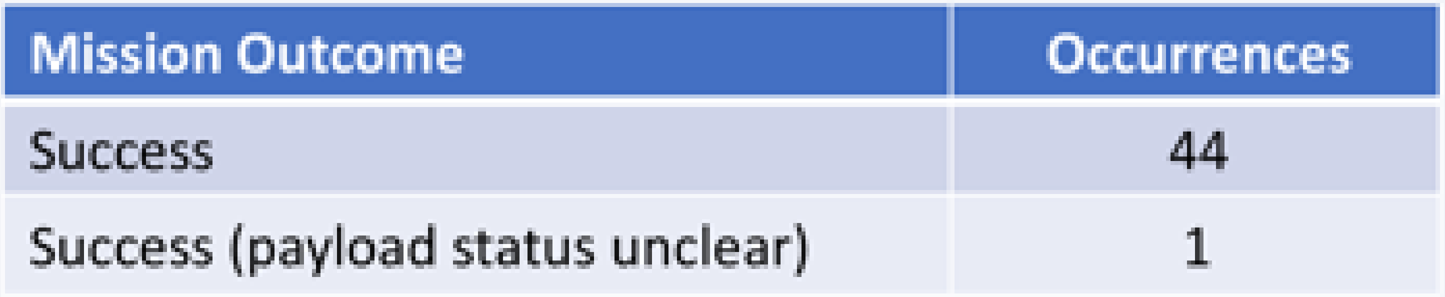
- Selecting distinct booster versions according to the filters above, these 2 are the result.



Booster Version
F9 FT B1031.2
F9 FT B1022

Total Number of Successful and Failure Mission Outcomes

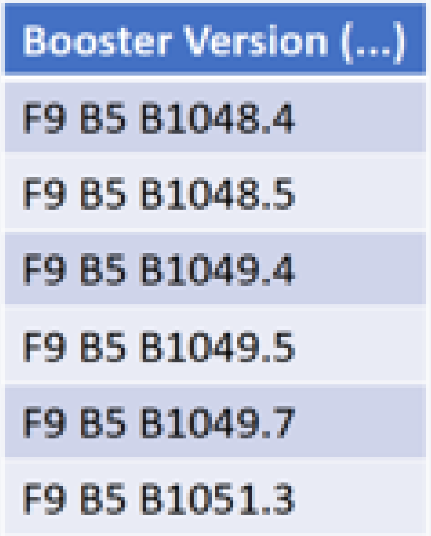
- Calculate the total number of successful and failure mission outcomes



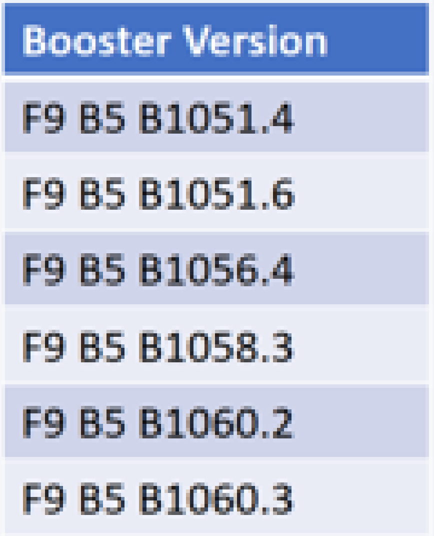
Mission Outcome	Occurrences
Success	44
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here



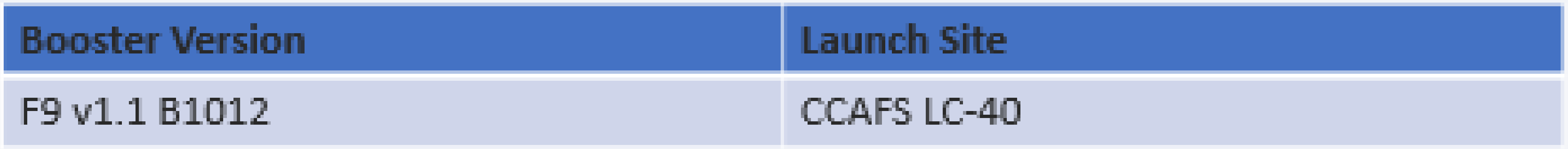
Booster Version (...)
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3



Booster Version
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2015 Launch Records

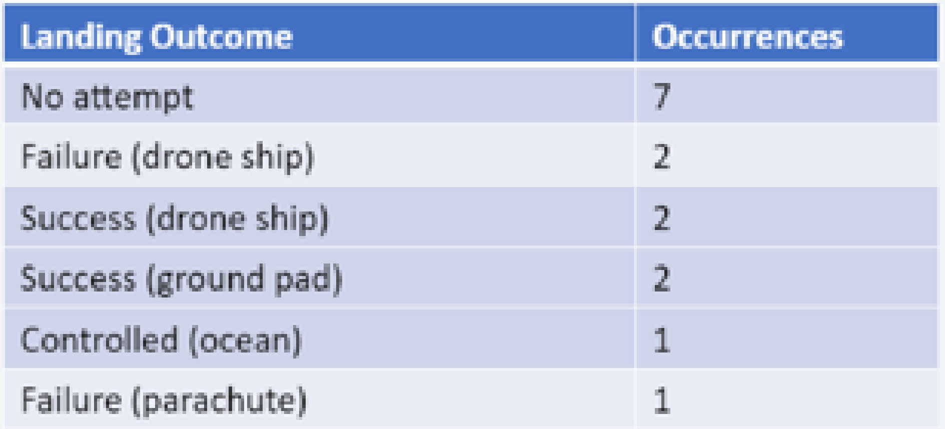
- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015



Booster Version	Launch Site
F9 v1.1 B1012	CCAFS LC-40

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 orde



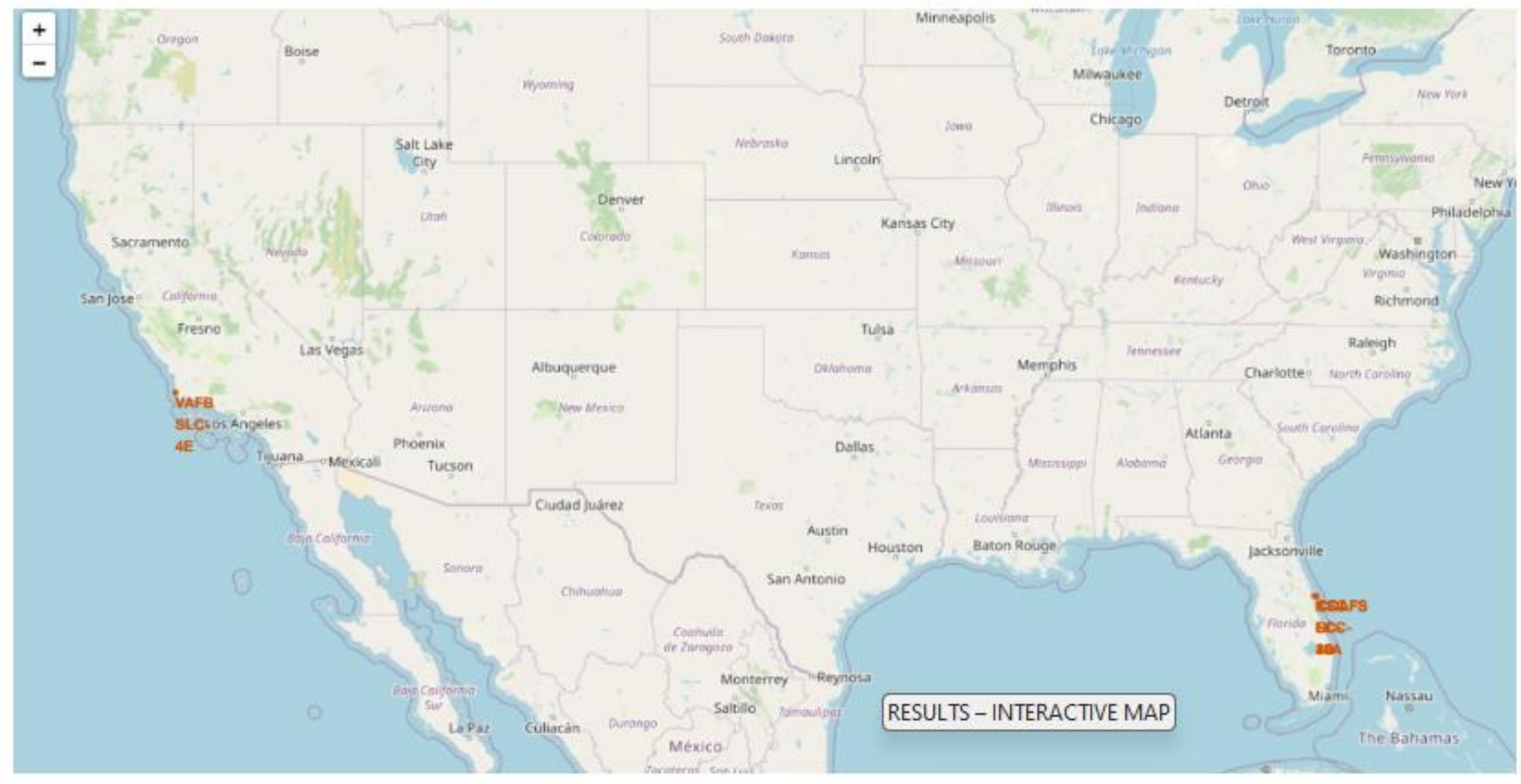
Landing Outcome	Occurrences
No attempt	7
Failure (drone ship)	2
Success (drone ship)	2
Success (ground pad)	2
Controlled (ocean)	1
Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

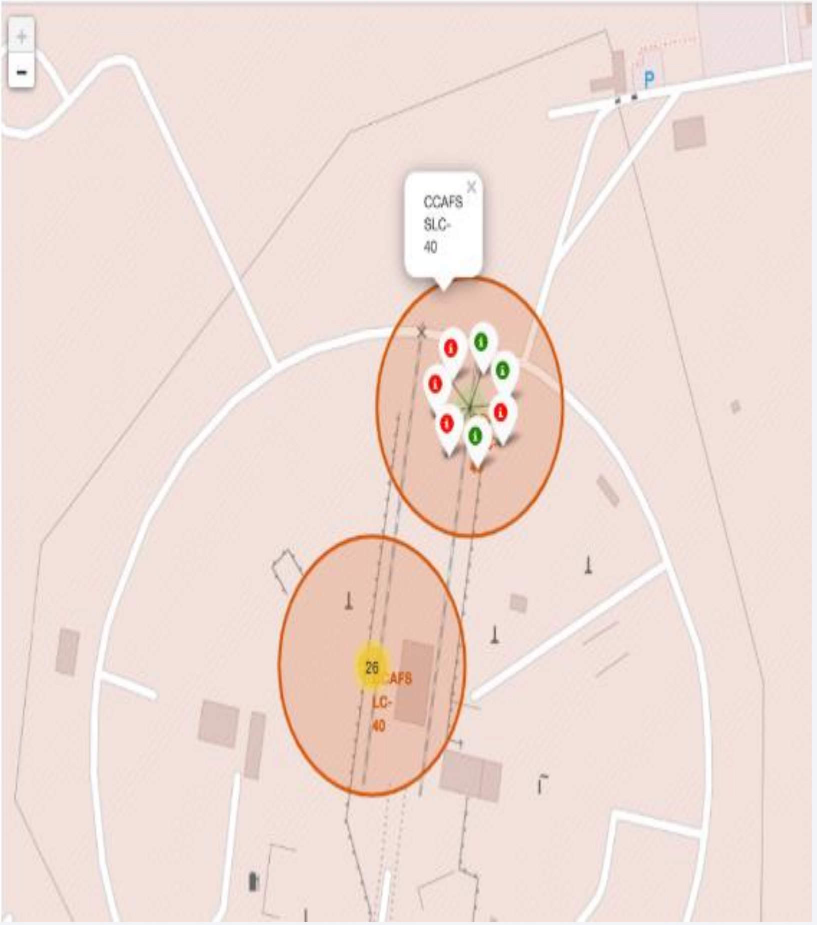
Results-interactive MAP

- SpaceX launch sites are located in the United States of America, one in the coast of California and the rest near the coast of Florida

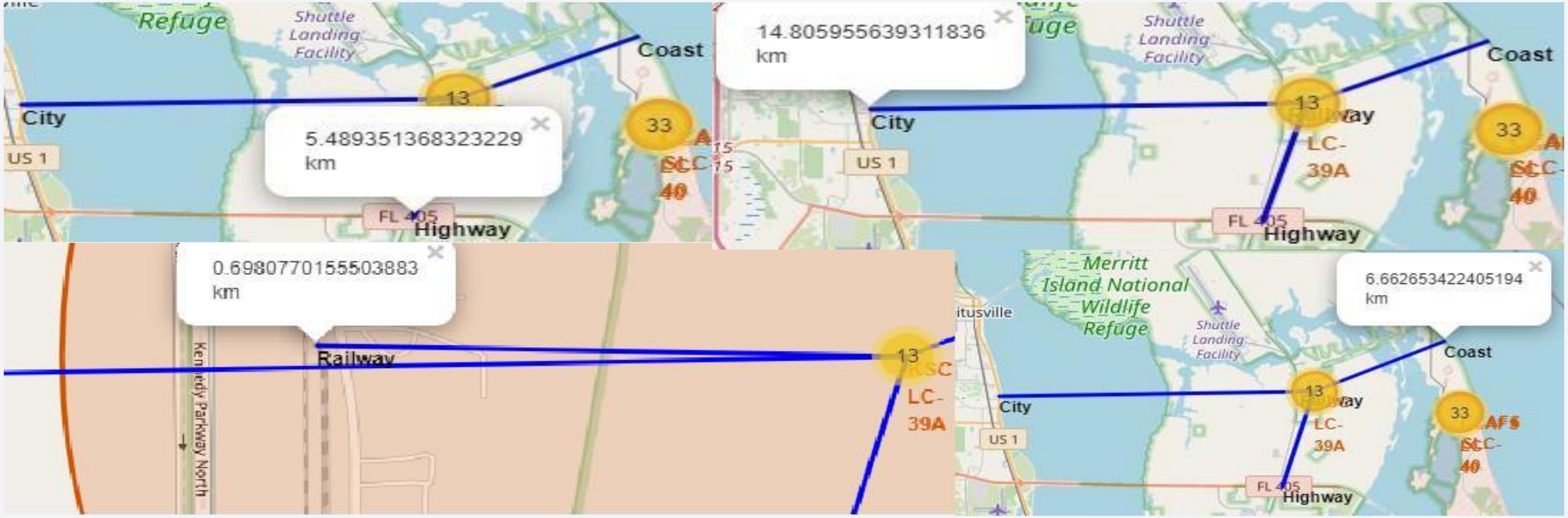


Color-Coded Launch Markers

- Green marker shows successful launches and red marker show failures.

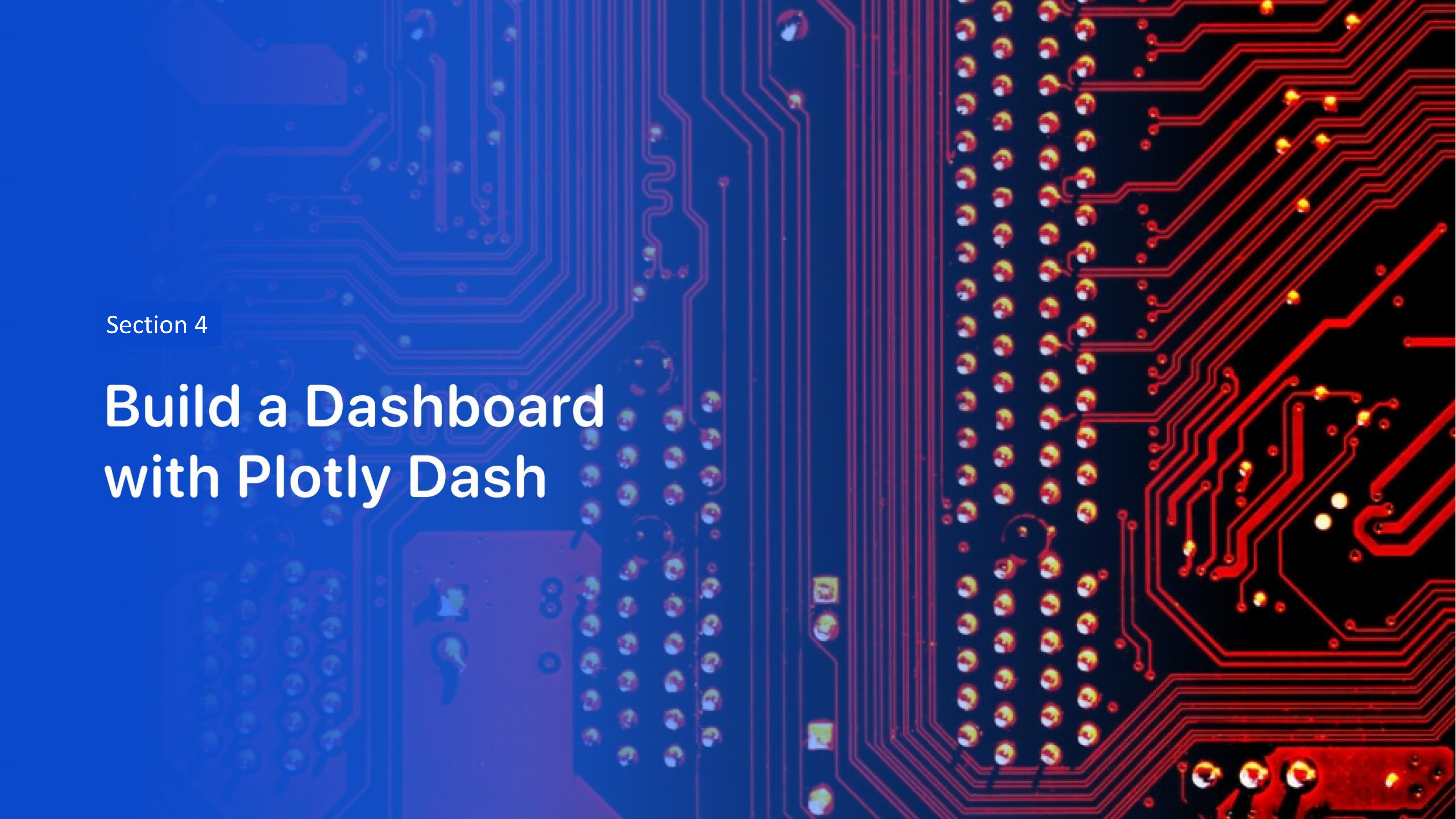


Key Location proximities

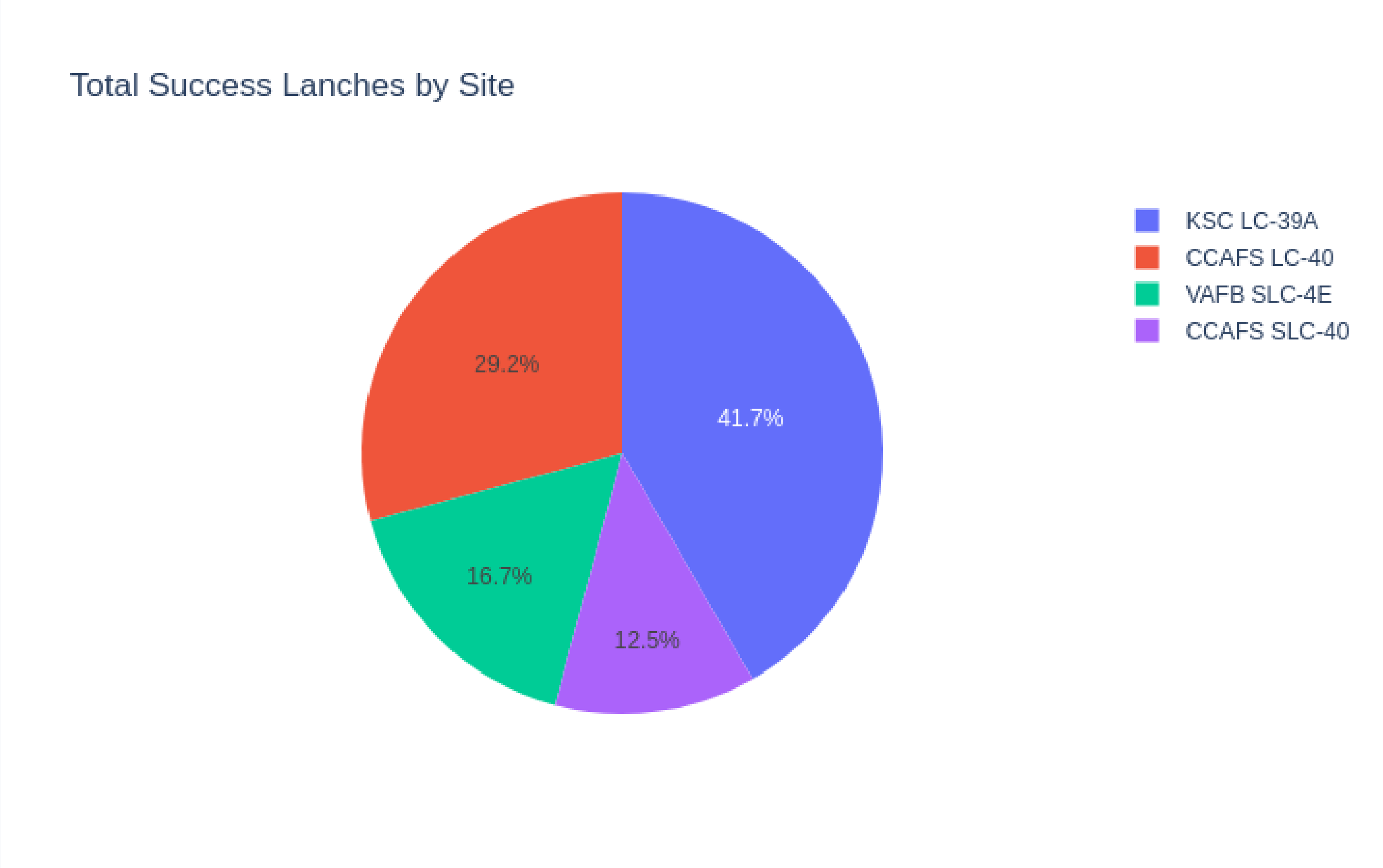


Section 4

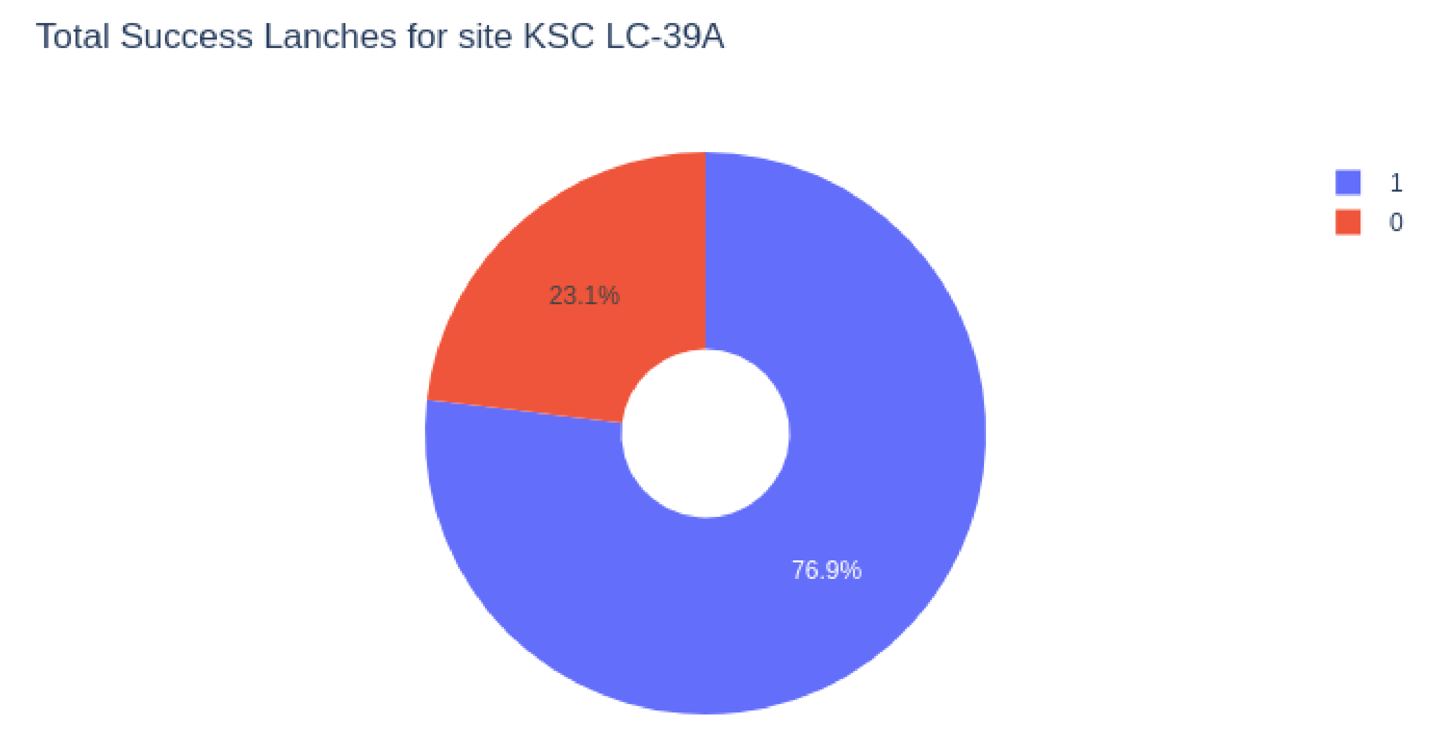
Build a Dashboard with Plotly Dash



Launch success count for all sites

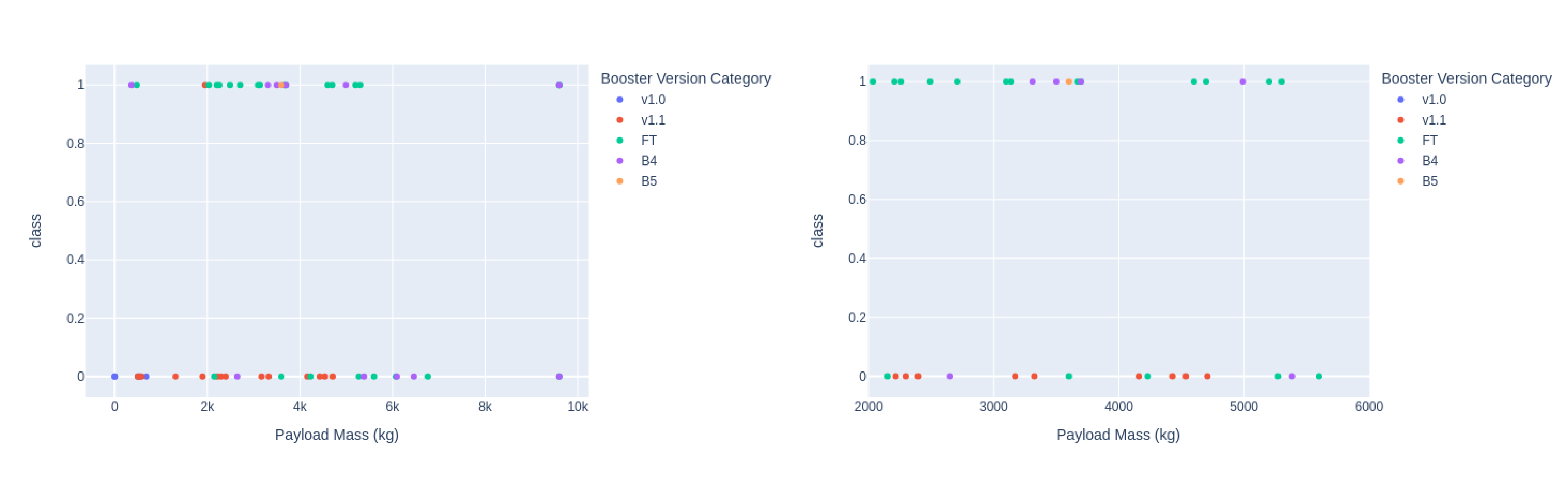


Launch site with the highest launch success ratio



KSC LC-39A has a 76.9% success rate

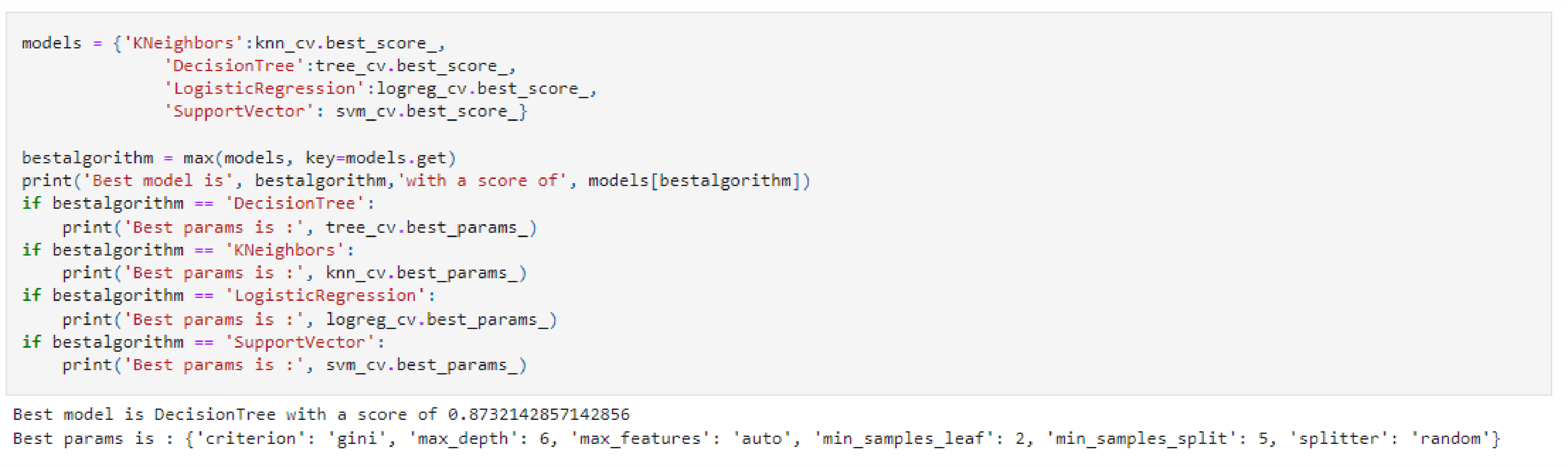
Payload vs. Launch outcome scatter plot for all sites



Section 5

Predictive Analysis (Classification)

Classification Accuracy



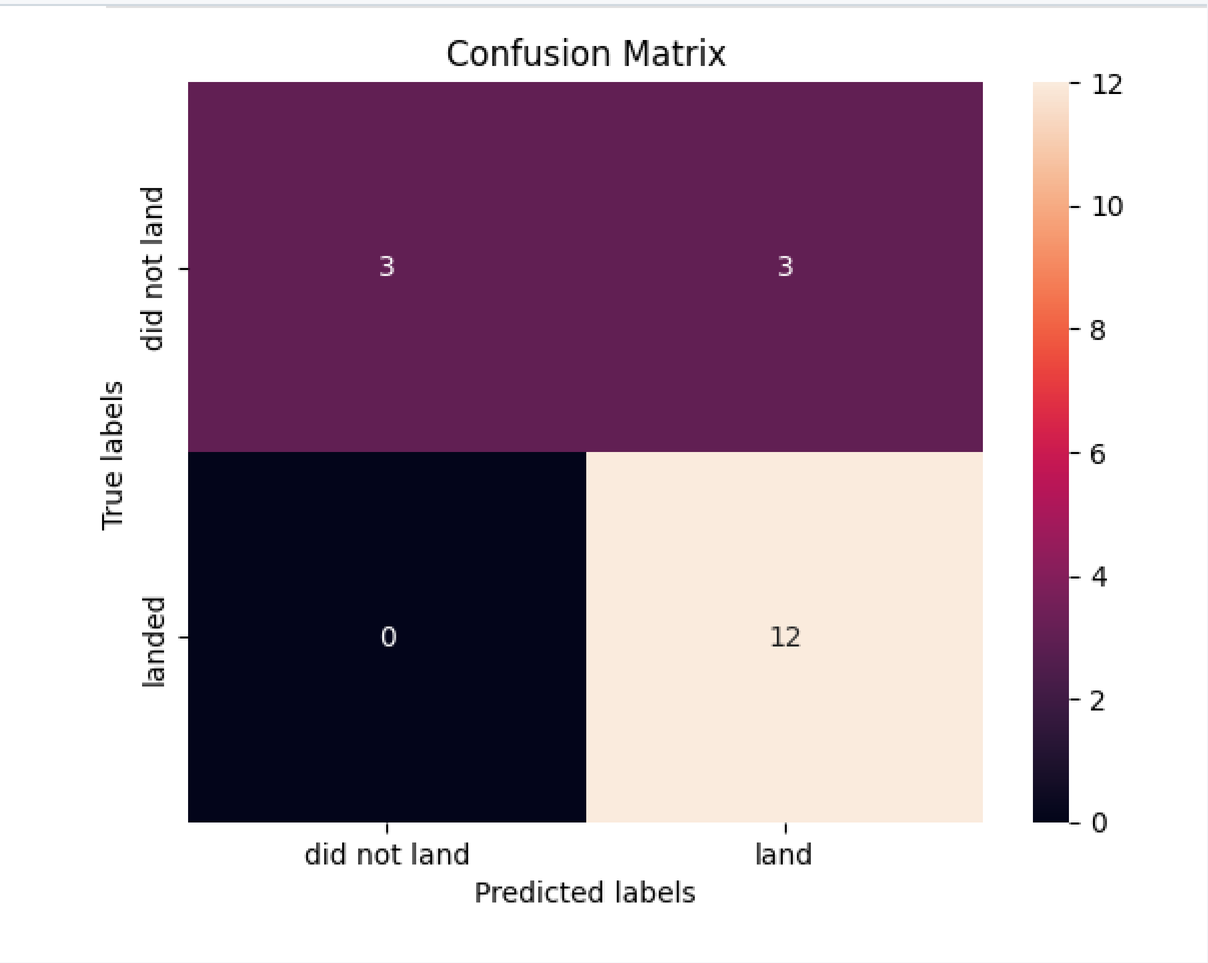
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
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 0.8732142857142856
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 5, 'splitter': 'random'}
```

Confusion Matrix

- The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.



Conclusions

- The Decision tree classifier is the best machine learning algorithm for this task.
- Launch success rate started to increase in 2013 till 2020.
- Higher payload launches are more successful
- KSC LC-39A is the launch site with the highest success rate

Thank you!