IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Fernando Tortosa Cid
July 1st 2024

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Proximity Analysis
- Dashboard
- Conclusion
- Appendix

Executive Summary

- Summary of methodologies
  - On the work developed to prepare this report, the following methodologies were used:
    - Data Collection and data wrangling
    - Exploratory Data Analysis through SQL, pandas and matplotlib
    - Dashboarding using Plotly Dash and Folium for maps visualization
    - Predictive analysis using classification machine learning methods for determining successfully landing of Falcon 9 1st Stage
- Summary of all results
  - Machine learning predictions reveals that the Decission Trees algorithm provides the most accurate predictions and that the flaws of the algorithm lies in the false positive predictions.

Introduction

- Project background and context
  - To setup a new space launching startup company Space Y to compete with Space X, we need to analyse the success factors driving this industry when landing reusable Stage-1 rockets as in the Falcon 9 from Space X. Identifying the features that determine success in recovering these Stage-1 rockets, can allow Space Y to operate at low costs and compete within the range of the $60M as Space X advertises.

- Problems to find answers
  - Which factors (features) of a launch determine a successful recovery of 1st Stages.
  - Using these factors, determine the success of launches and recoveries of 1st stages
  - Identify the most suitable configuration of launches to secure successful recoveries of 1st stages

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected from the SpaceX API and wikipedia sources to complete it
- Perform data wrangling
  - Data has been proved to be free from missing values and categorical features have been adapted to be used during analysis and modelling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Models have been built with scikit-learn and tuned using cross validation with the GridSearchCV class

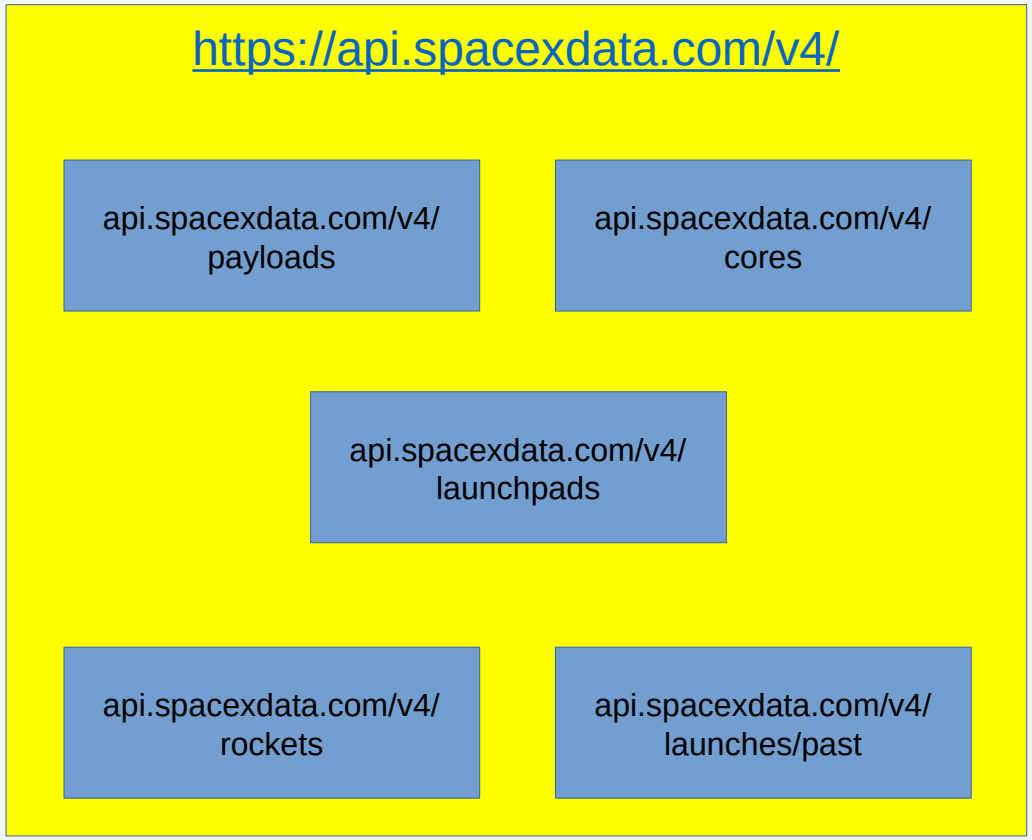
Data Collection

For collecting information two sources were used:

- Use of SpaceX API – The REST API from SpaceX included information about launches details sch as the name of the launcher, payloads, success/failure and description, events and times of occurrence, internet references, etc.
- Using webscraping of public sources like Wikipedia – This has completed the REST API data with rocket specifications and historical launch information such as customer, orbit

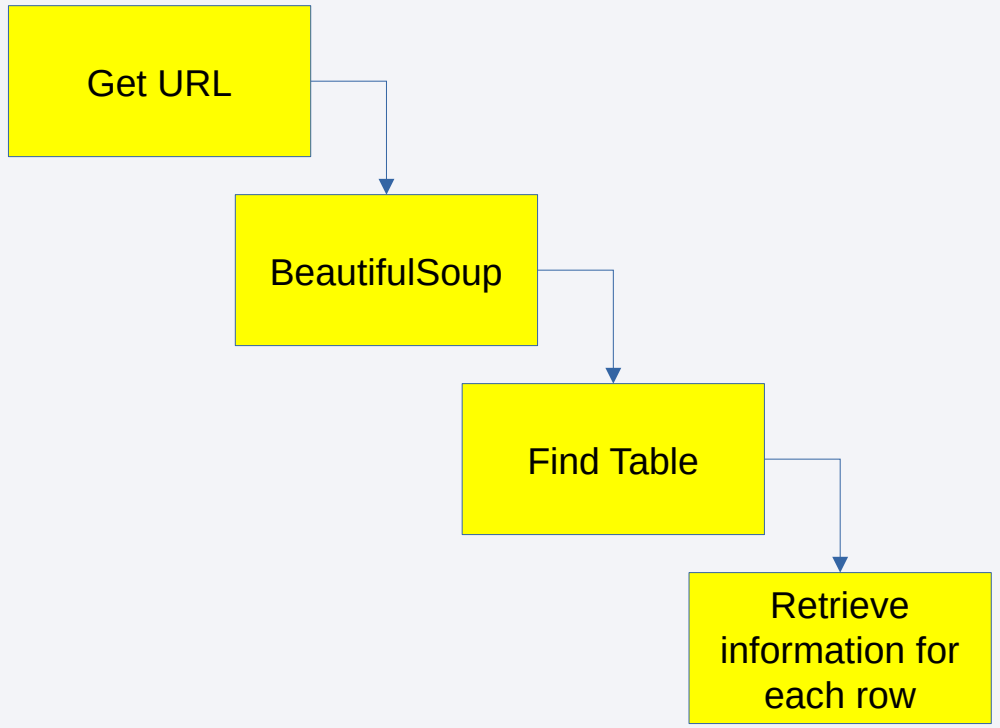
Data Collection – SpaceX API

- Data is collected from the SpaceX REST API for launch, rocket, core, capsule, starlink, launchpad and landing pad data.
- The data is accessible through http://api.spacexdata.com/v4/ with different endpoints to retrieve different information related to rockets, launchpads, payloads, core and past launches.
- The GitHub notebook can be found under https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- Data on Falcon 9 launch records is web-scrapped from Wikipedia.
- Information about date/time of launch, booster version, landing status and payload was collected.
- The GitHub notebook can be found under https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Method of discovering, structuring cleaning, enriching, validating and publishing through the following process:
  - Identification of missing values (null values)
  - Check of data types
  - Encoding of categorical features (eg: Landing Outcome)
- The GitHub notebook can be found under https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

The performed exploratory data analysis using visualization:

- Flight Number vs Payload Mass – the more massive the payload, the less likely the first stage will return
- Flight Number vs Launch Site – two sites CCAGS SLC40 and KSC LC39A are more intesively used, and more data provided
- Payload Mass vs Launch Site – Site VAFB SLC4E is not used for heavy payload mass (greater than 10,000 Kg)
- Orbit Type vs Success Rate – success rate increases for high altitude orbits (ES-L1, GEO, HEO and SSO) or very low (VLEO)
- Flight Number vs Orbit Type – In LEO orbit the success relates to the number of flights, though seems to be no relationship to GTO orbit.
- Payload Mass vs Orbit Type – successful landing rate on heavy payloads are higher for Polar, LEO & ISS
- Year vs Success Rate – tendency to increase the success rate since 2013.

The GitHub notebook can be found under
https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/edadataviz.ipynb

EDA with SQL

- Display 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 success in drone ship and have payload mass between 4000 and 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass.
- 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.

The GitHub notebook can be found under https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

The analysis of the site locations may reveal important information about the success rate. It may also depend on the location and proximities to the launch site.

- Used folium.Circle to add NASA Johnson Space Center and site locations
- Used folium.map.Marker to add text labels for circles
- Create marker clusters for each individual launch per site and added color to highlight Successful landings based on class label outcome
- Added Mouse position on the map to get coordinates for mouse over map
- Added folium.Polyline with coordinates to display distance between coastline point and sites
- Land marks such as railway line and highways and connect them to launch sites.

The GitHub notebook can be found under
https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Build a dashboard applications for presenting SpaceX Launch records Dashboard information. This website is available to view and filter data. The Dashboard is built with Flask and Dash web Framework. It contains:

- Pie Chart showing total launches by a site (or all sites) – Display of all sites distribute the percentage of launches for each site; display per site shows the rate os success for the given site.
- Scatter Graph showing the relationship with Success Rate and Payload Mass (Kg) for the different Booster Versions. This includes all sites. Information can be filtered per payload range.

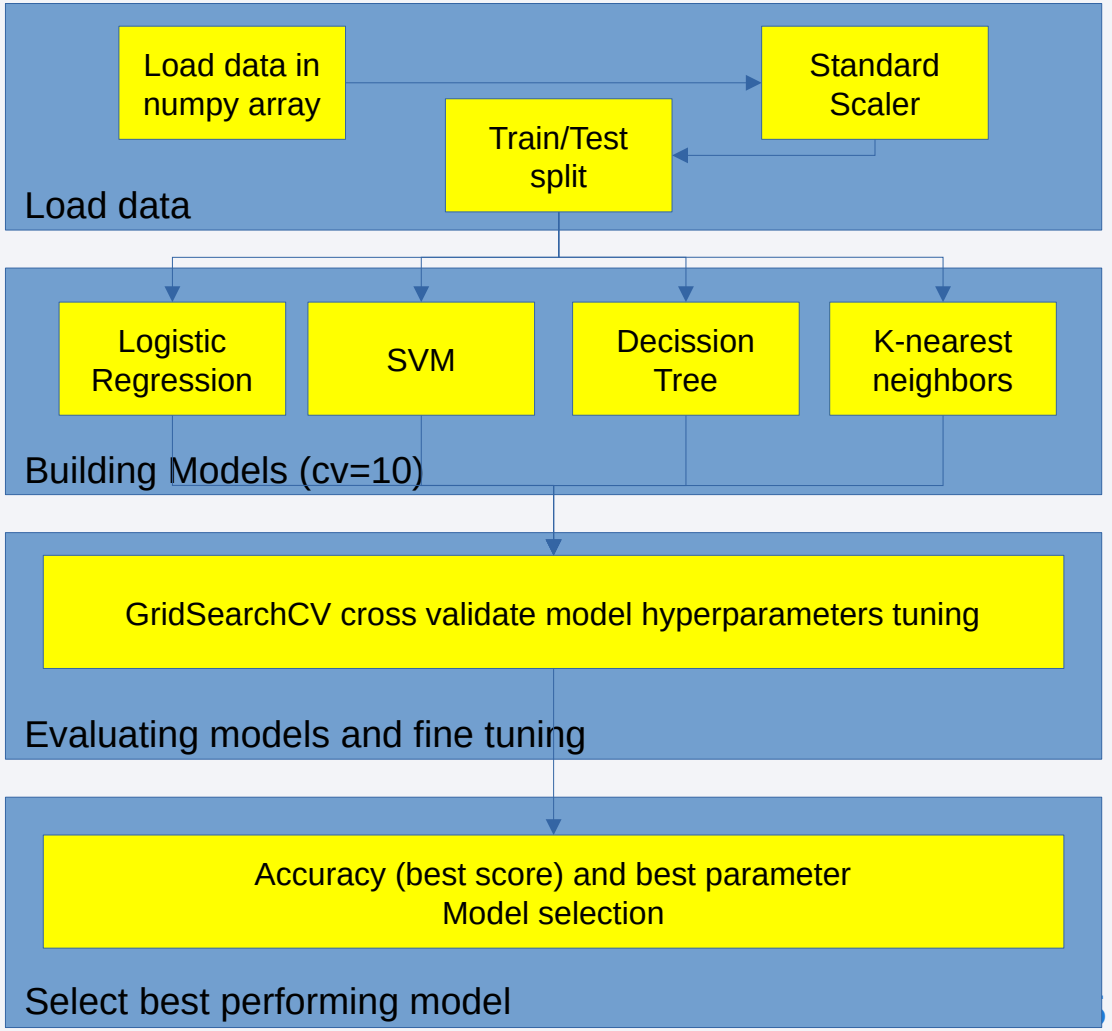
The GitHub notebook can be found under https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Process follow:

- Load data, standardise and split dataset into train/test
- Build classification models
- Evaluate models and fine tune hyperparameters using score metrics and confusion matrix
- Find the best performing classification model

The GitHub notebook can be found under https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



Results

- Exploratory data analysis results:
  - No rockets launched from VAFB-SLC site for heavy payloads.
  - No success rate for orbit SO. The highest success rate is for orbits on high altitude or very low.
  - As a rule, as increase the number of flights, increase the success rate (orbits ISS, VLEO), although this might not be the case for GTO.
  - With heavy payloads the successful landing or positive landing rate are more for Polar, VLEO and ISS.
  - The launch success rate is increasing from 2013 to 2020.

Results

- Interactive analytics:
  - All sites are located close to the sea coast
  - The majority of the launches are carried out in the east coast (KSC LC-39A, CCAFS SLC-40 and CCAFS LC-40)
  - The sites are located close to railroad or highway
  - The sites keep certain distance away from the cities

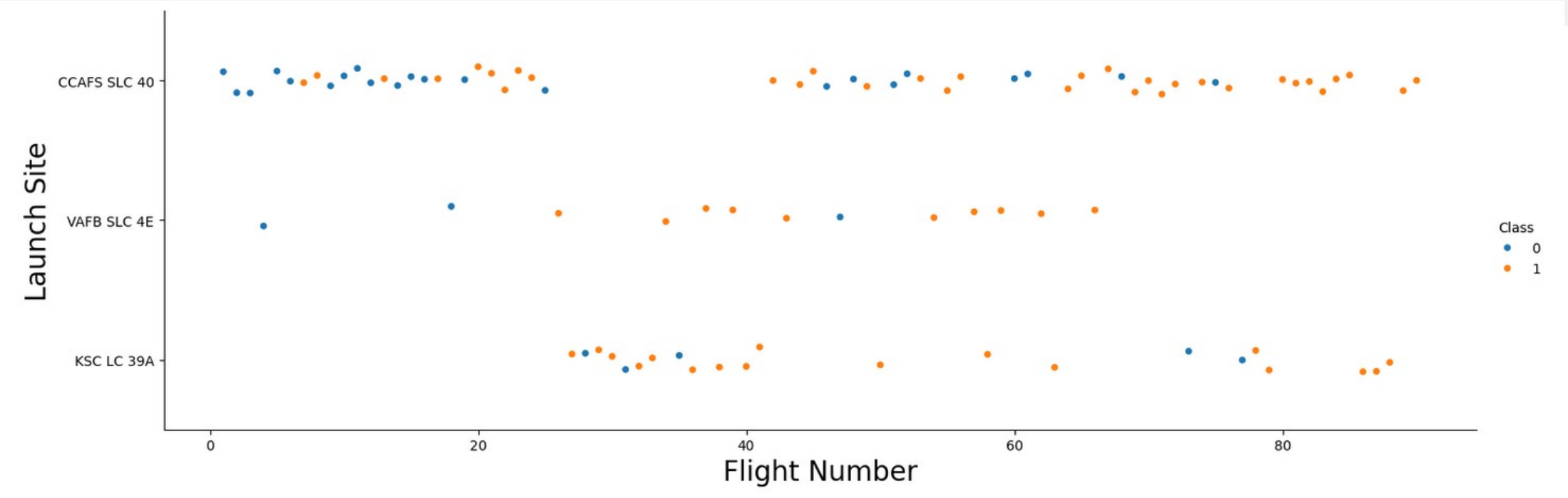
Results

- Predictive analysis results
  - Models trained: Logistic Regression, SVM, Decission Trees and K-nearest neighbors
  - All models trained were fine-tuned by using cross validation automated and resulted in a score above 80%
  - The best performance in prediction is achieved with the Decission Tree model, with a score of 86%
  - All models have one common flaw: false positives. If first stage lands successfully, the models predict it correctly. However, if landing is failed, the model is not good at predicting a fail outcome.

Section 2

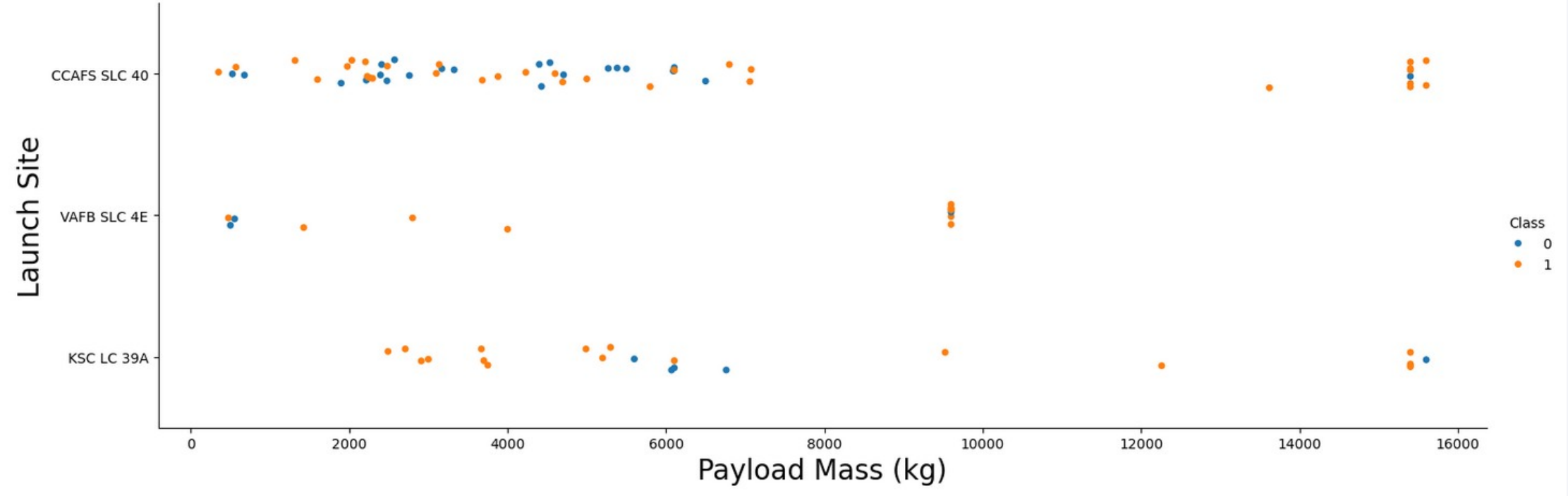
Insights drawn from EDA

Flight Number vs. Launch Site



The more successful sites are CCAFS SLC 40 and KSC LC 39A, although they are the most used. Recently, these are the sites used for launches and not the VAFB SLC 4E anymore.

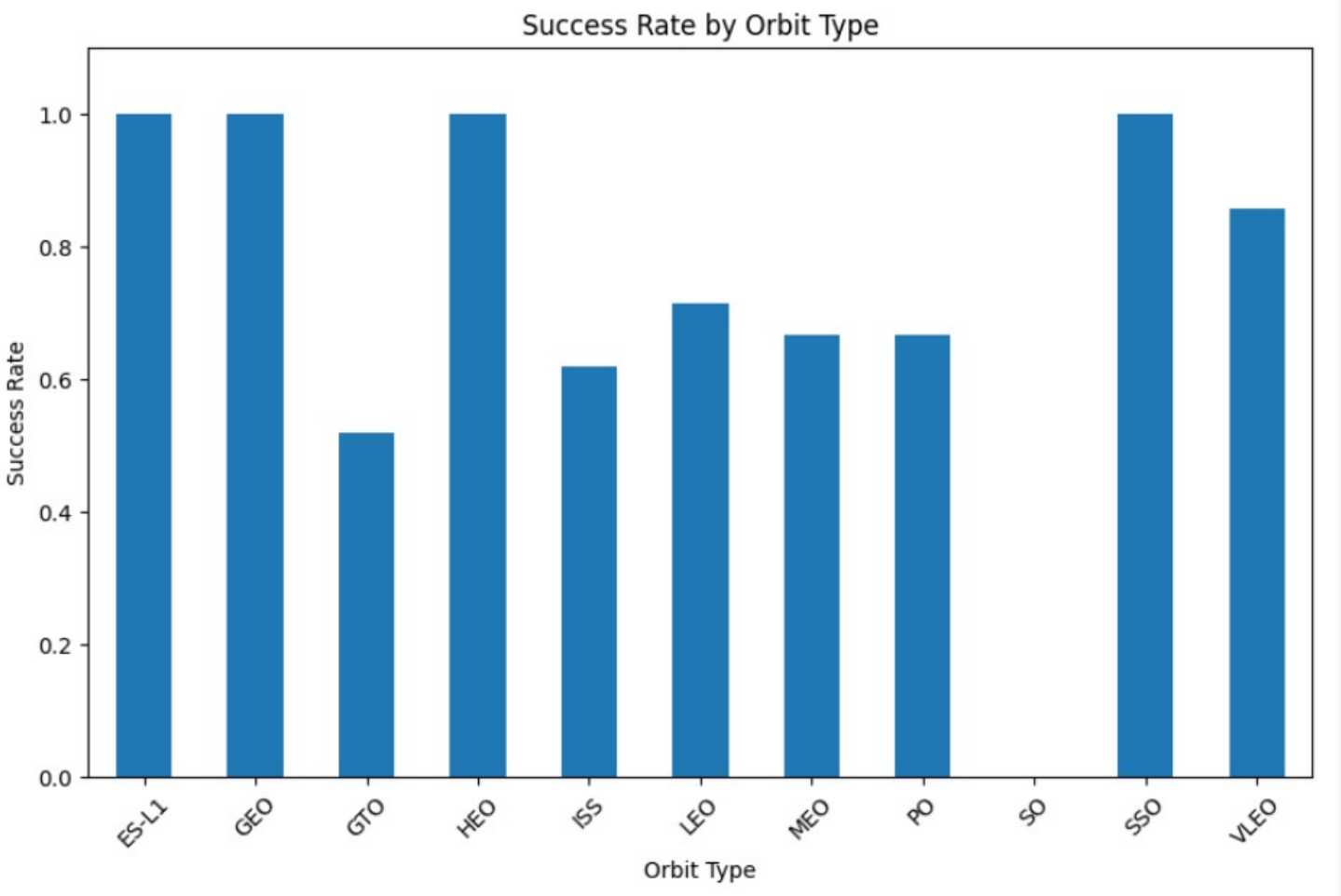
Payload vs. Launch Site



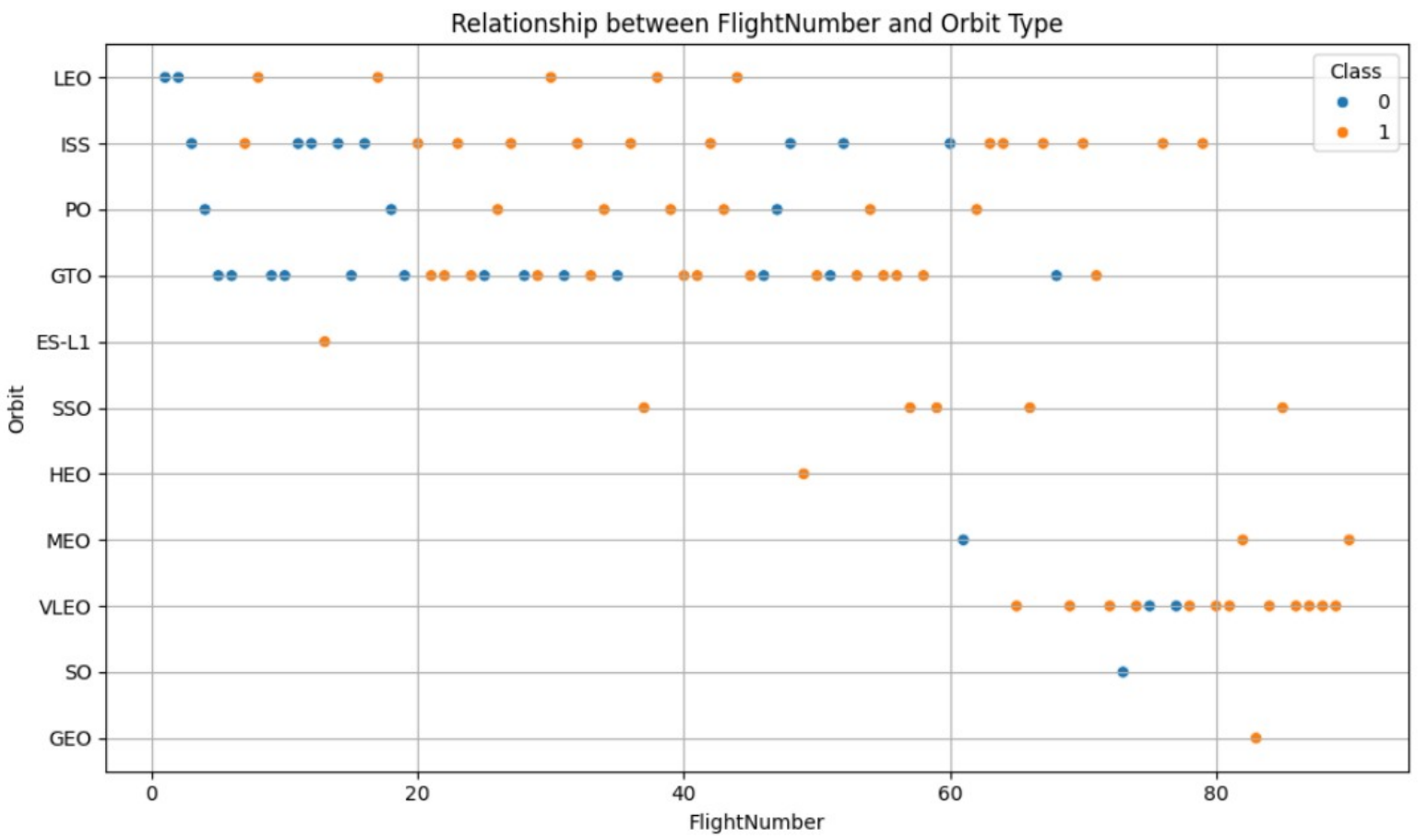
The sites CCAFS SLC 40 and KSC LC 39A are the only sites used for heavy load landings. For low loads the sites CCAFS SLC 40 and VAFB SLC 4E are used instead.

Success Rate vs. Orbit Type

More success launches are for the orbits ES-L1, GEO, HEO and SSO, but no success rate for orbit SO.

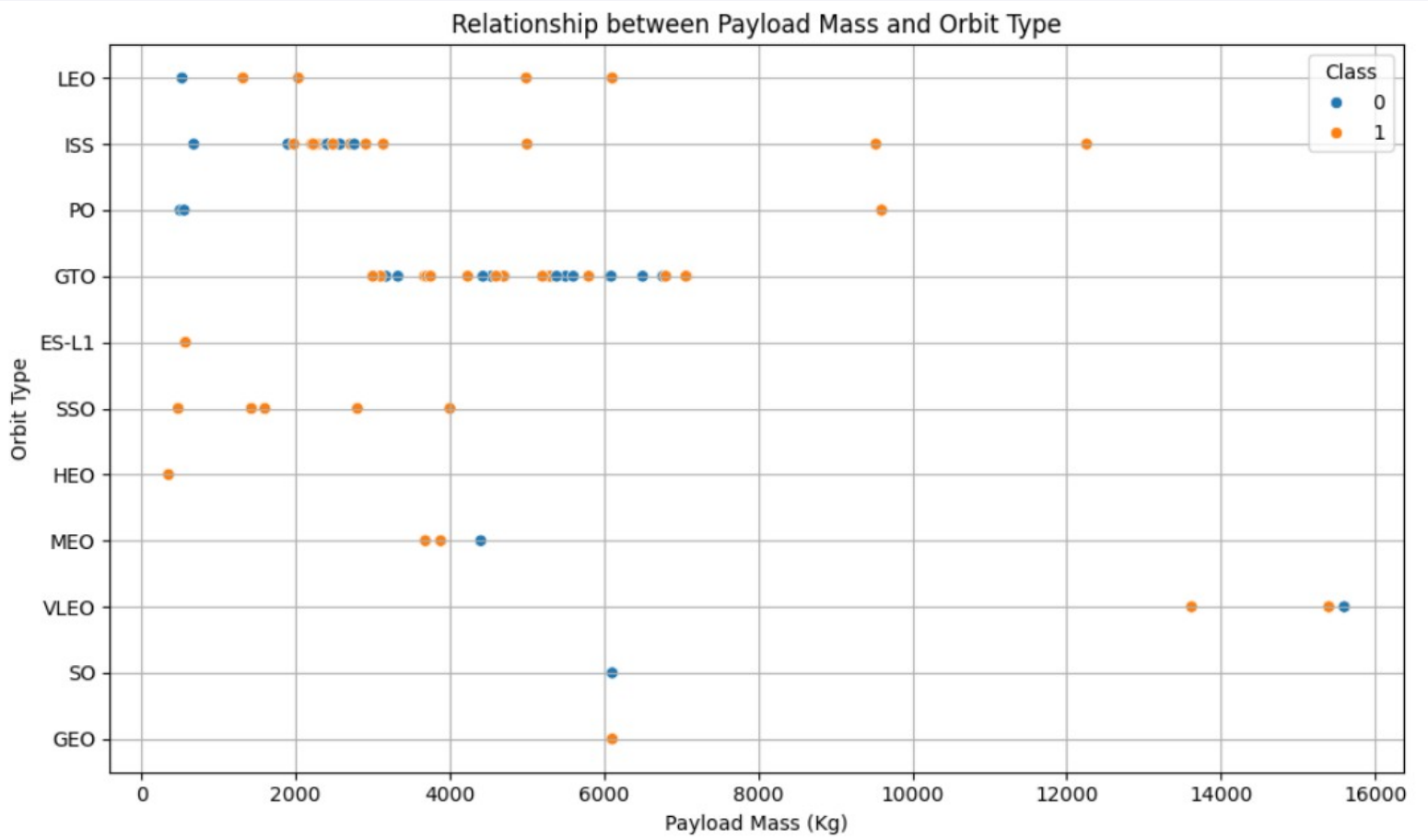


Flight Number vs. Orbit Type



There is a clear correlation between orbit and flight number for LEO orbit. However, for other orbits, this correlation is not strong, as for the ISS or GTO orbits.

Payload vs. Orbit Type

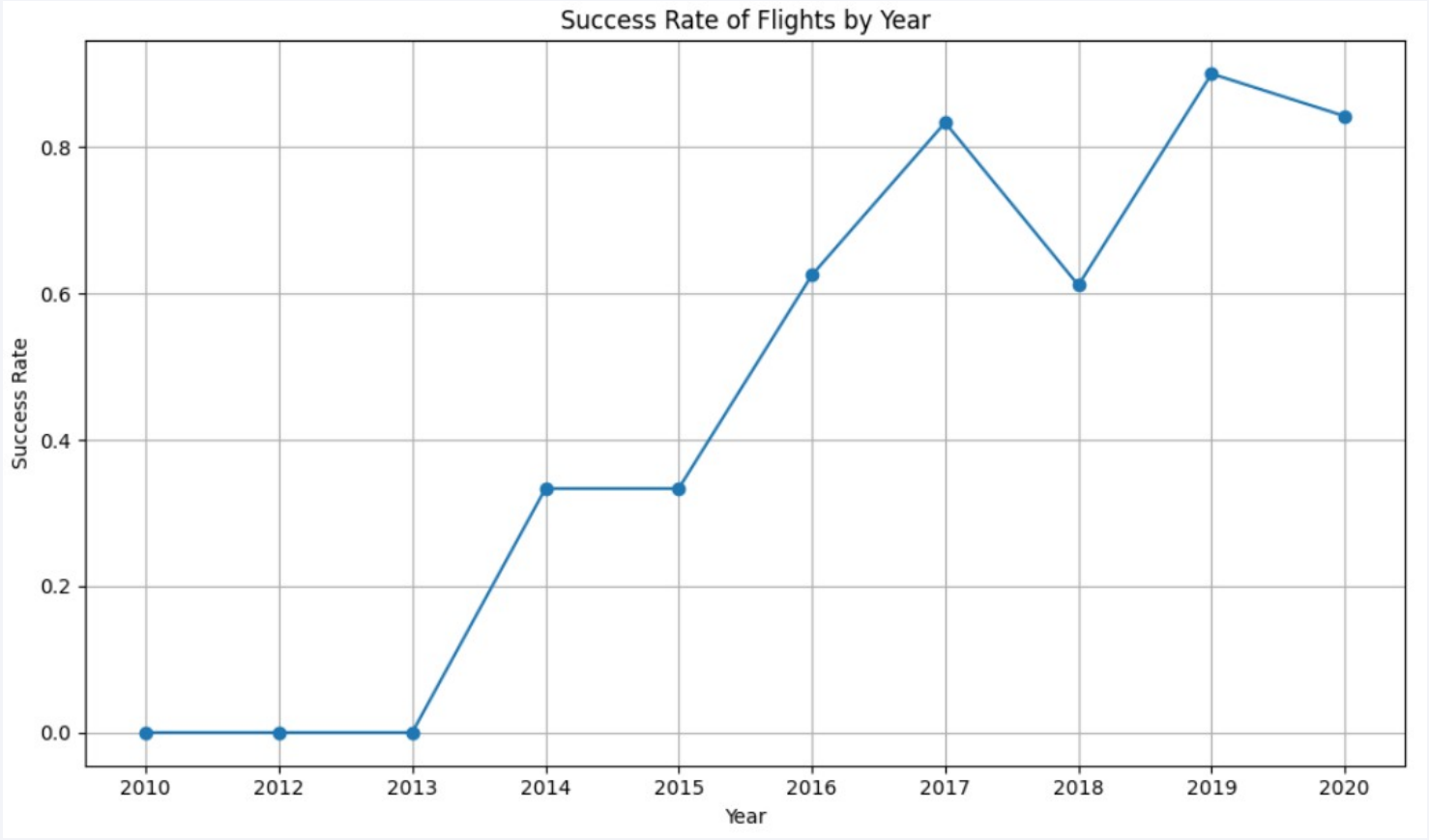


For some orbits (LEO, ISS and POLAR) the high payloads correlate to successful landings.

For GTO, there is no correlation between the features.

Launch Success Yearly Trend

- The success rate since 2013 keeps a tendency to increase until 2020.



All Launch Site Names

SQL Query

```
select distinct Launch_Site
from SPACEXTBL
```

Explanation

DISTINCT only shows unique values

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

Launch Site Names Begin with 'CCA'

SQL Query

```
select *
from SPACEXTBL
where Launch_Site like 'CCA%'
limit 5
```

Explanation

`where Launch_Site like 'CCA%'` only shows values where launch_site begins with 'CCA'

`Limit 5` only shows the first 5 values

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

Total Payload Mass

SQL Query

```
select Customer, sum(PAYLOAD_MASS__KG_) as Total_payload
from SPACEXTBL
where Customer='NASA (CRS)'
```

Explanation

`sum(PAYLOAD_MASS__KG_) as Total_payload` calculates sum of the payload mass

Customer	Total_payload
NASA (CRS)	45596

Average Payload Mass by F9 v1.1

SQL Query

```
select Booster_Version, avg(PAYLOAD_MASS__KG_) as Average_payload
from SPACEXTBL
where Booster_Version like 'F9 v1.1%'
```

Explanation

`avg(PAYLOAD_MASS__KG_)` calculates the average of the payload mass

Booster_Version	Average_payload
F9 v1.1 B1003	2534.6666666666665

First Successful Ground Landing Date

SQL Query

```
select min(Date) as First_Landing_Date, Landing_Outcome
from SPACEXTBL
where Landing_Outcome like 'Success (ground pad)%'
```

Explanation

`min(Date) as First_Landing_Date` calculates the earliest date

First_Landing_Date	Landing_Outcome
2015-12-22	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL Query

```
select Booster_Version
from SPACEXTBL
where Landing_Outcome like 'Success (drone ship)' and
PAYLOAD_MASS__KG_ between 4000 and 6000
```

Explanation

`Landing_Outcome like 'Success (drone ship)'` select the required outcome

`PAYLOAD_MASS__KG_ between 4000 and 6000` filter the payload

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

SQL Query

```
select Mission_Outcome, count(Mission_Outcome) as Total_Mission_Outcome
from SPACEXTBL
where Mission_Outcome like '%Success%' or Mission_Outcome like '%Failure%'
group by Mission_Outcome
```

Explanation

`Mission_Outcome like '%Success%' or Mission_Outcome like '%Failure%'`

Filter by Success or Failure

Mission_Outcome	Total_Mission_Outcome
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

SQL Query

```
select distinct Booster_Version
from SPACEXTBL
where PAYLOAD_MASS__KG_ = (
    select max(PAYLOAD_MASS__KG_)
    from SPACEXTBL
    )
```

Explanation

Subquery selects th max payload.

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

2015 Launch Records

SQL Query

```
select substr(Date,6,2) as month, Landing_Outcome, Booster_Version,
Launch_Site
from SPACEXTBL
where substr(Date,0,5)='2015' and Landing_Outcome='Failure (drone
ship)'
```

Explanation

`substr(Date,6,2)`	gets a substring to get the month
`substr(Date,0,5)`	gets a substring to get the year

month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

SQL Query

```
select Landing_Outcome, count(*) Number_Landings
from SPACEXTBL
where Date between '2010-06-04' and '2017-03-20'
group by Landing_Outcome
order by Number_Landings desc
```

Landing_Outcome	Number_Landings
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

Explanation

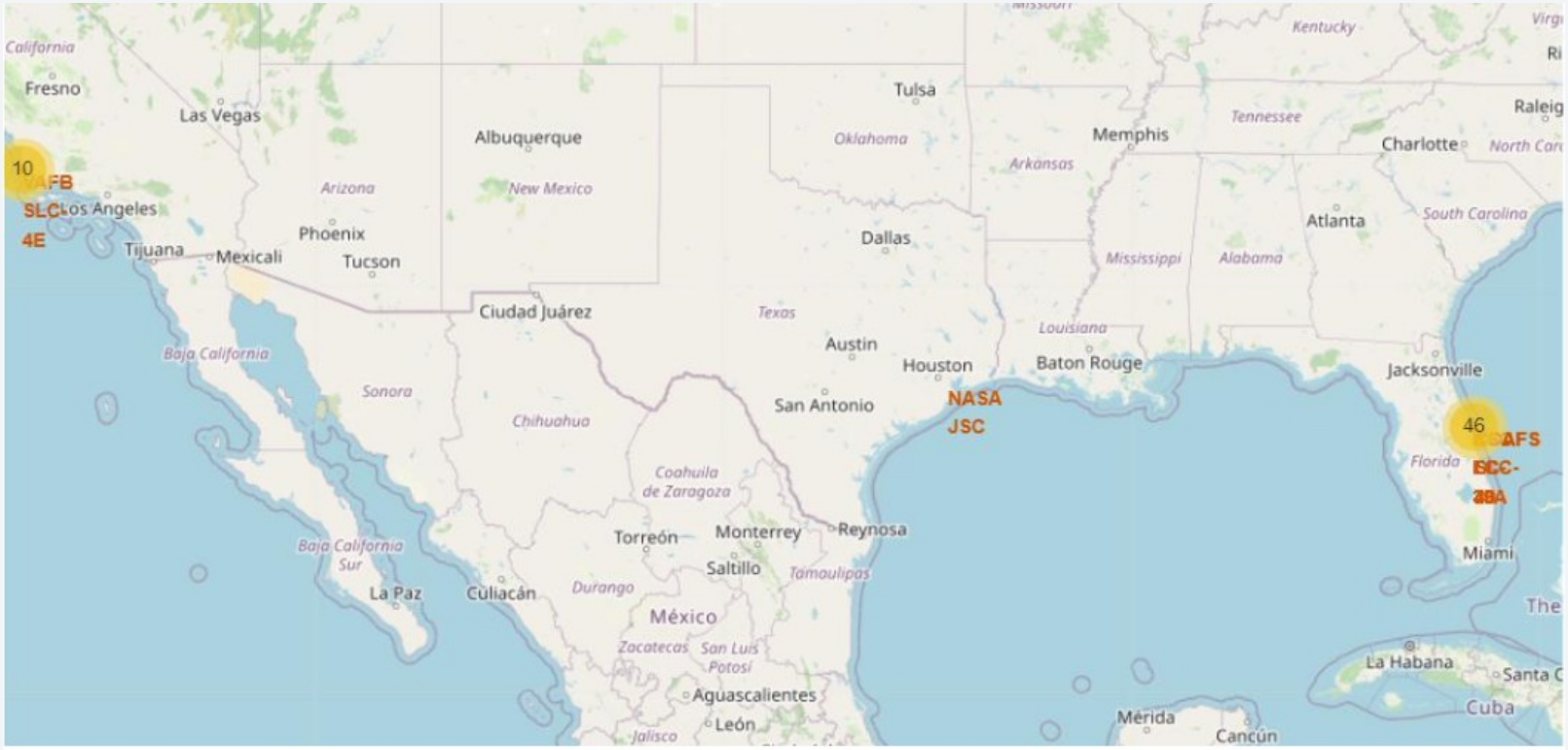
`Date between '2010-06-04' and '2017-03-20'` selects a date in a range

`group by Landing_Outcome` permits count on this field.

Section 3

Launch Sites Proximities Analysis

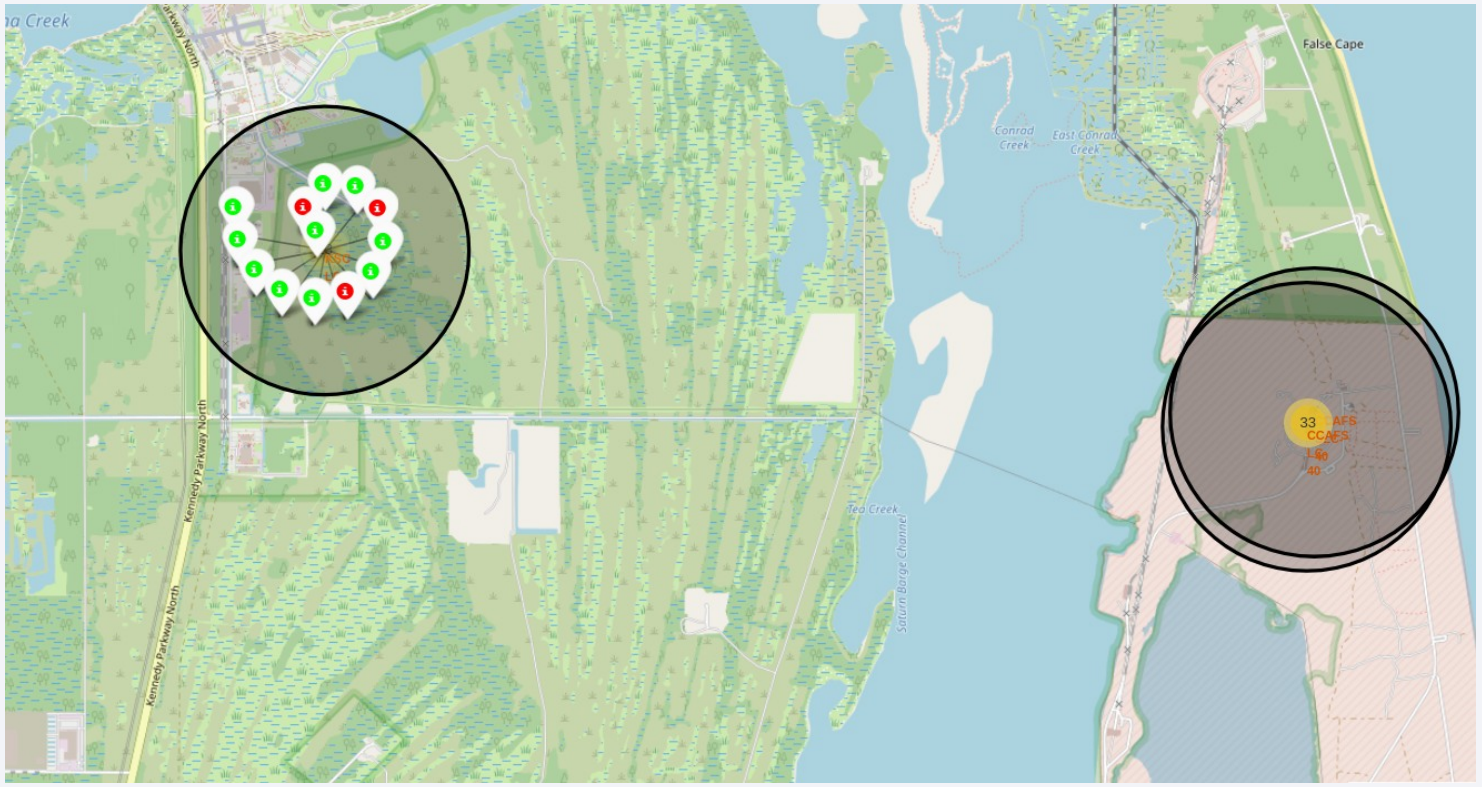
Proximity analysis global map



The analysis reveals that the site locations are far away between them. However, all of them are located close to the sea coast.

Successful sites

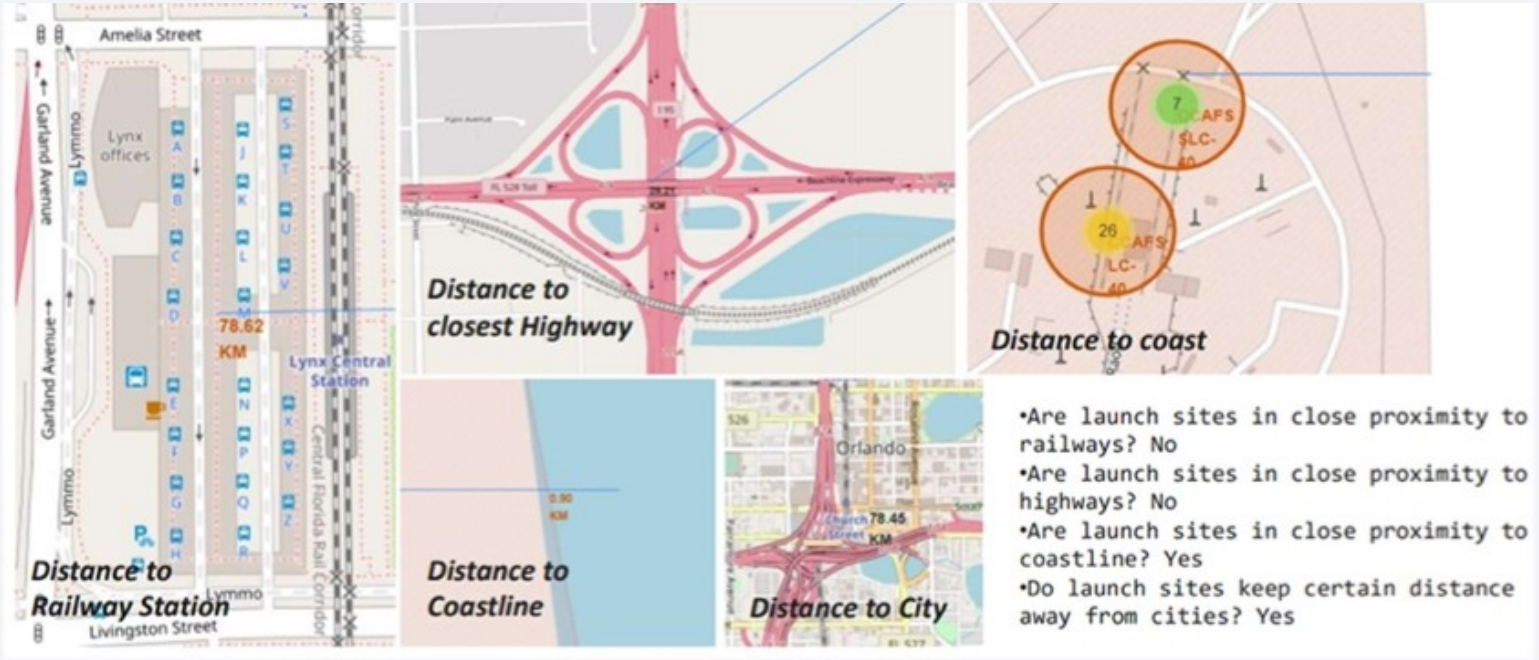
The KSC LC 39A shows to be one of the sites with the most successful launches.

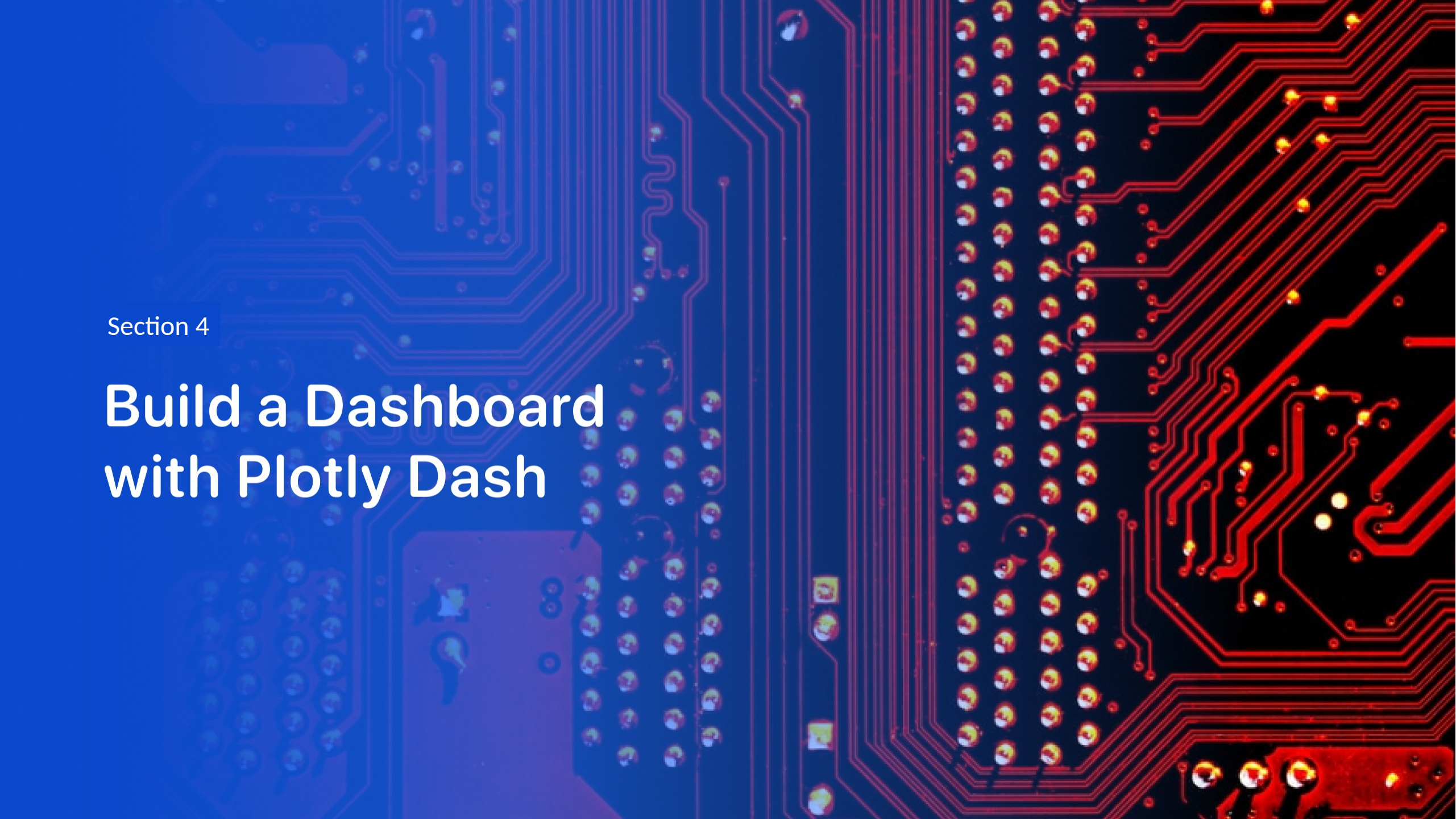


Launch sites and proximity to landmarks

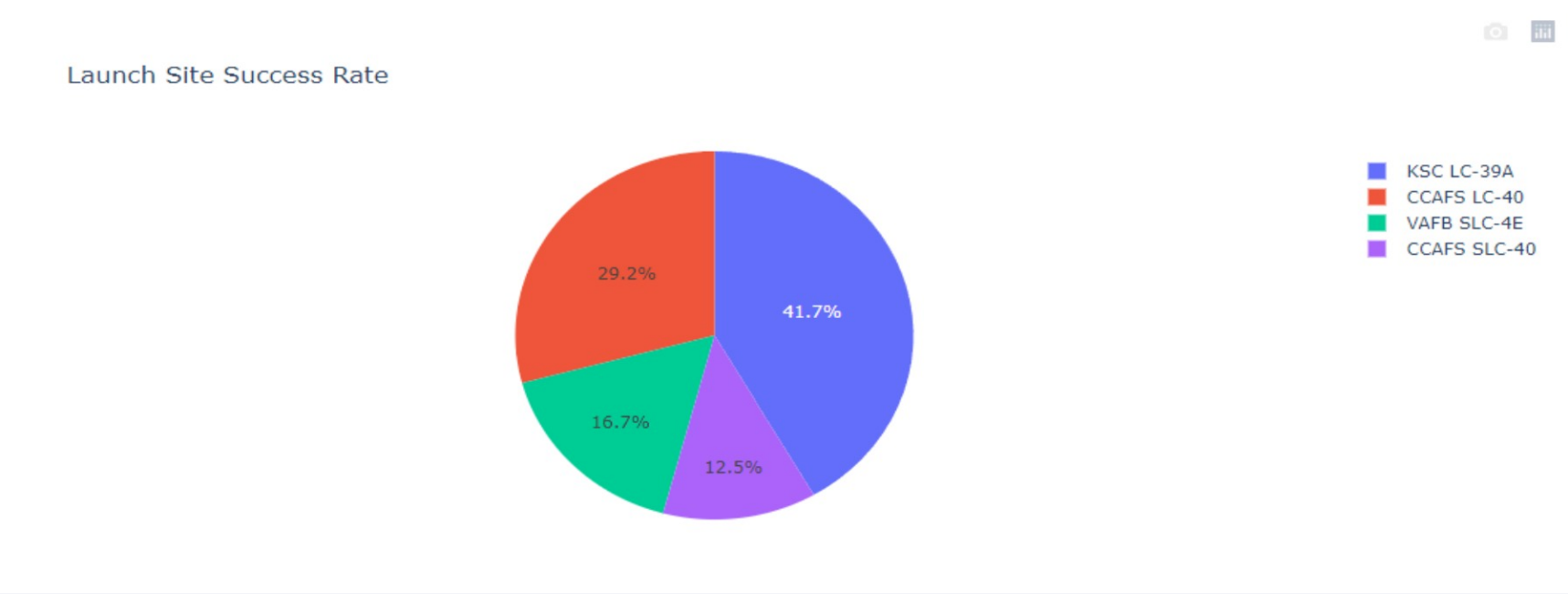
Every site is located in areas near to specific landmarks. Thus, the sites are located:

- Close to the coast
- Away from highways
- Away from railroads
- Away from cities.



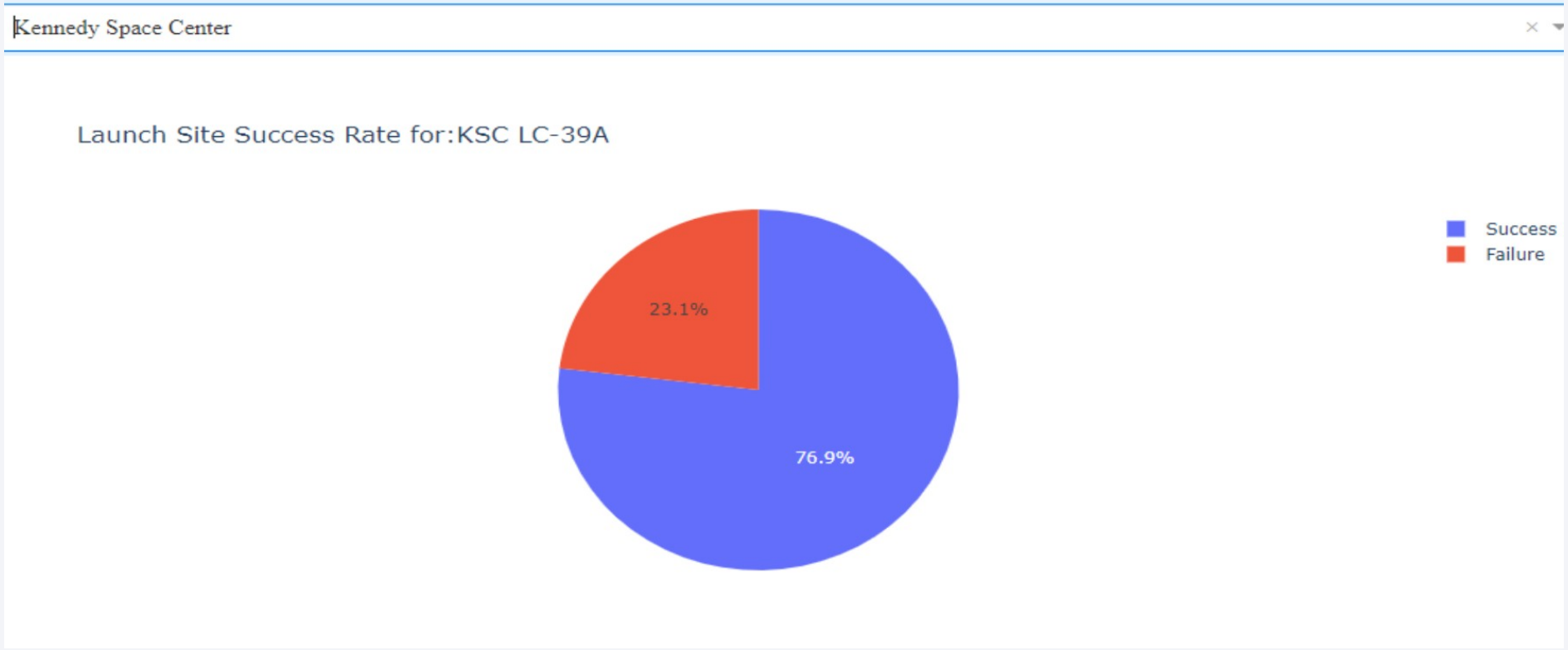

Section 4
Build a Dashboard with Plotly Dash

Total success launces by all sites



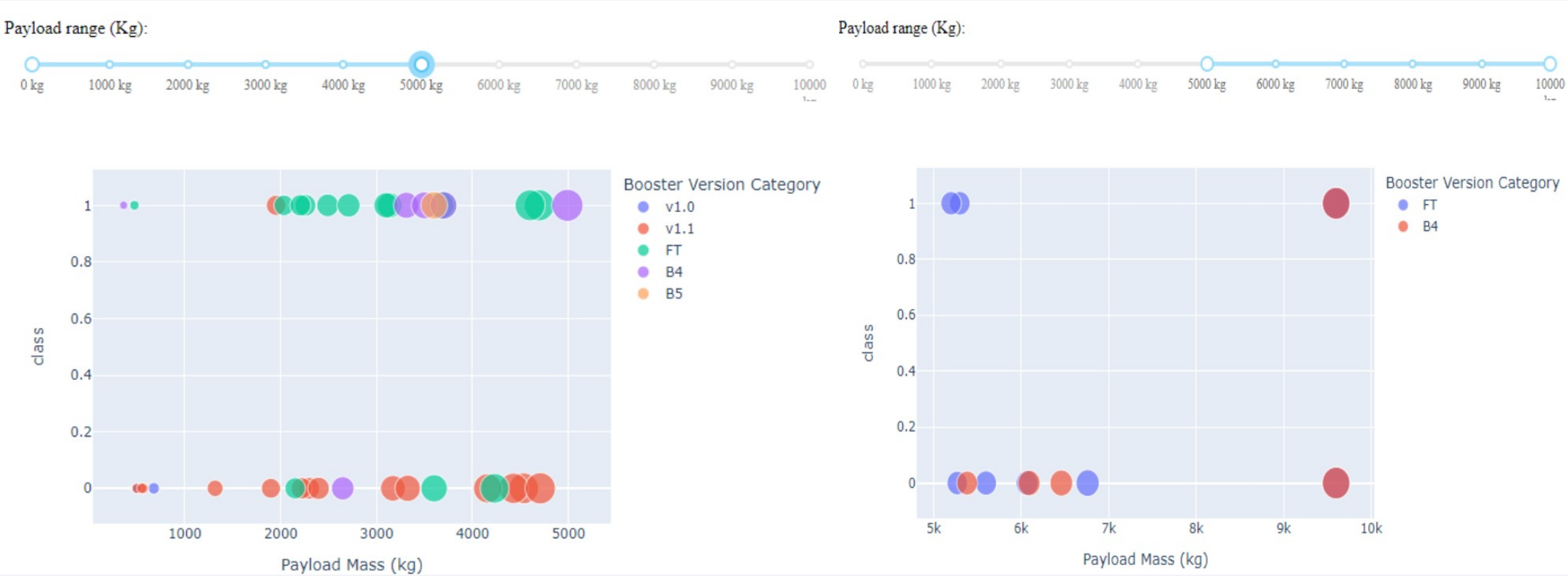
The highest successful launches are from site KSC LC-39A.

Success rate for KSC LC-39A



Success rate for the KSC LC-39A space centre is 76,9%, leaving a failure rate is 23,1%

Total success launches for all sites



Higher successful launches for Low weight payloads(0-5000kg) vs Low successful launches for Low weight payloads(5000-10000kg). Boosters FT and B4 high success rate in both cases

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The highest accuracy in the prediction is the Decission Tree, although all models predict with more than 84% accuracy.

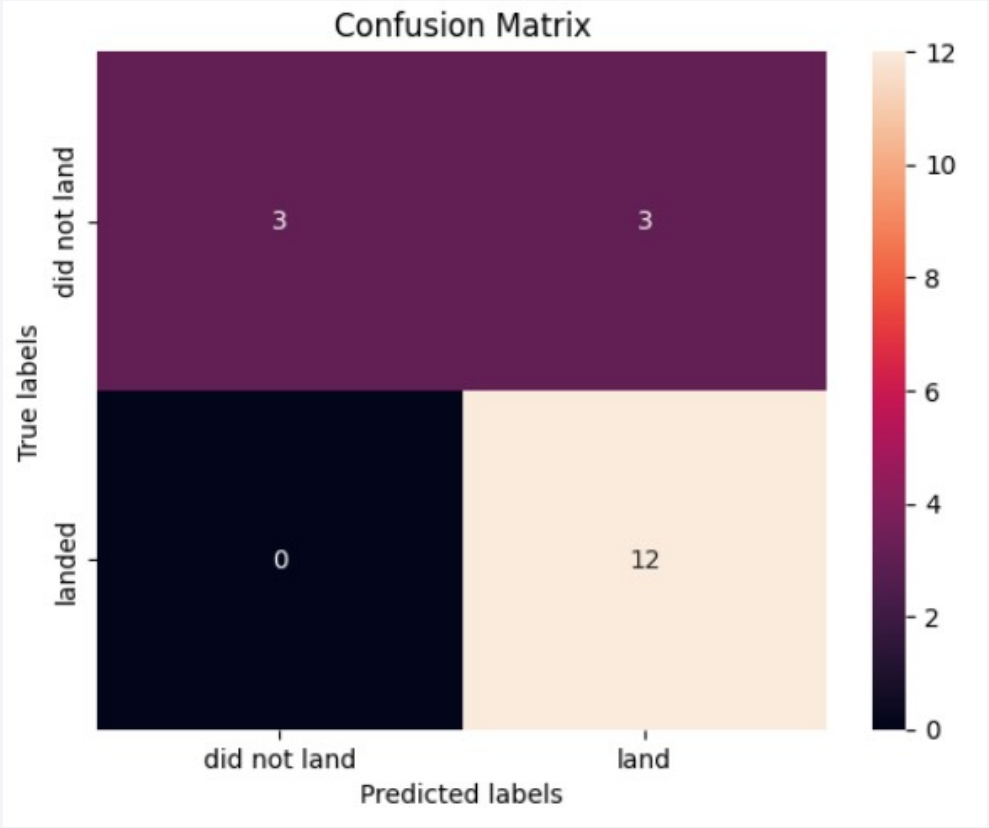
```
logreg_score = logreg_cv.best_score_
svm_score = svm_cv.best_score_
tree_score = tree_cv.best_score_
knn_score = knn_cv.best_score_

print(f"Logistic Regression Score: {logreg_score}")
print(f"SVM Score: {svm_score}")
print(f"Decission Tree Score: {tree_score}")
print(f"KNN Score: {knn_score}")
```

```
Logistic Regression Score: 0.8464285714285713
SVM Score: 0.8482142857142856
Decission Tree Score: 0.8625
KNN Score: 0.8482142857142858
```

Confusion Matrix

The best performing model is the decission tree, with the best score of 86%. But also the confusion matrix for this model reveals, that there is a tendency in the model for false positives. That means, that for successful landings, the model predicts it correctly. However, for unsuccessful landings, the model may predict it incorrectly.



Conclusions

- The Decission Tree Classifier model trained with the given dataset is the best performer model for predicting outcome rates.
- Low weighted payloads perform better than the heavier payloads
- The success rates for SpaceX launches is directly proportional time
- In a few years it can be forecasted that launches will perform better
- The site KSC LC-39A had the most successful launches records from all
- Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate

Appendix

List of Notebooks used in the analysis:

Data Collection: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Wrangling: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

Webscraping: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb

SQL Lite: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Data Visualization: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/edadataviz.ipynb

Maps Analysis: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Dash Application: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/spacex_dash_app.py

Machine Learning Models: https://github.com/TheBlackmad/IBM-Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Thank you!