IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Faruk Yavuz
01/20/2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection with API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Dash board
  - Machine Learning Prediction of Landing Success
- Summary of all results
  - Exploratory Data Analysis Results
  - Interactive Analytics with Visuals
  - Predictive Analysis Results

Introduction

- Project background and context
  - Space X cost of Falcon 9 launch is 62 million dollars
  - Competition cost is around 165 million dollars
  - Space X is affordable because it can reuse the first stage
  - Competitor companies need to determine if first stage will land successfully
    - If they want to bid against Space X!
- Problems you want to find answers
  - Which factors have an effect on the landing success of the rocket?
  - The relation between features that impacts the success rate of rocket landings
  - What are the desired operational factors to maximize chances of a successful rocket landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Space X API
  - Web Scraping Wikipedia
- Perform data wrangling
  - Enriched data by adding outcome label
  - Replaced NULL values by average for total mass
  - One hot encoded categorical labels
- Perform exploratory data analysis (EDA) using visualization and SQL

Methodology

Executive Summary

- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Selected features are normalized with a standard scaler.
  - Data set is split into test and training with 0.2 test ratio
  - Logistic Regression, SVM, Decision Tree, KNN are utilized
  - 10 fold Cross Validation is applied
  - Parameter grid search is applied

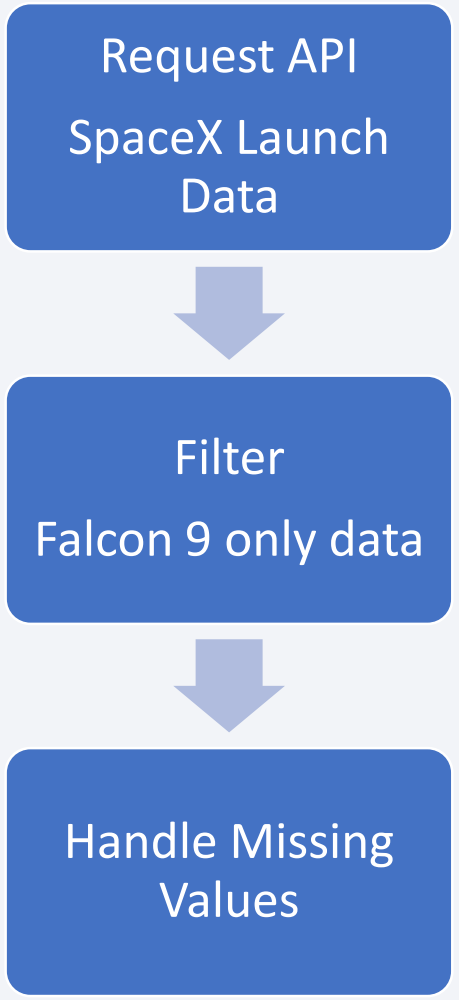
Data Collection

- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts

Data Collection - SpaceX API

- SpaceX API is utilized
- Data Filtering and processing
- Source code:

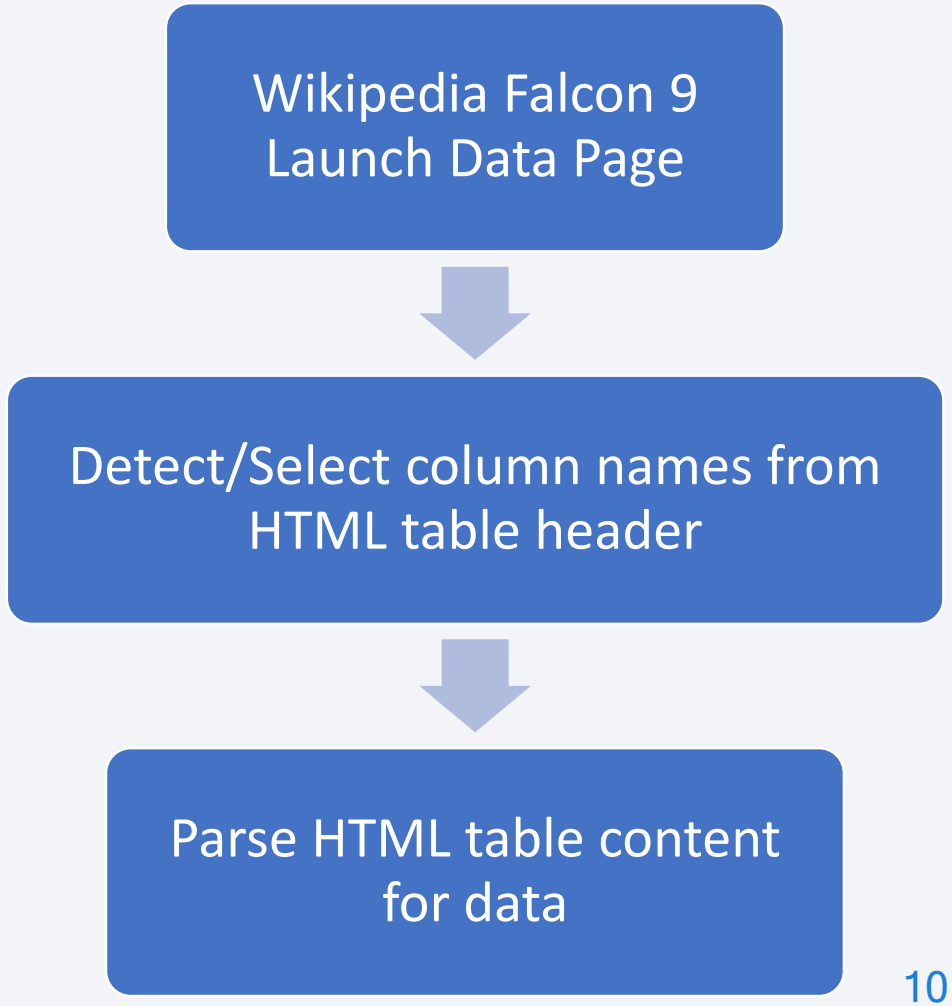
  https://github.com/Frk-Yvz/ibm_course/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- Wikipedia SpaceX Launches has data of interest
- Data is scraped with Beautiful Soup
- Source code:

  https://github.com/Frk-Yvz/ibm_course/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

- Exploratory data analysis is carried.
- Missing values and data types are analyzed.
- Landing outcomes are converted into 0's and 1's
- Source code:
  https://github.com/Frk-Yvz/ibm_course/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- Generated plots analyze relationship between Flight Number, Payload Mass, Launch Site, Orbit and Year on the success of the landing!
  - Plotted Flight Number vs Payload Mass with colored Outcome classes.
  - Plotted Flight Number and Launch Site with colored Outcome classes.
  - Plotted Payload Mass vs Launch Site with colored Outcome classes.
  - Plotted Success rate of each Orbit
  - Plotted Flight Number vs Orbit with colored Outcome classes.
  - Plotted Payload Mass vs Orbit with colored Outcome classes.
  - Plotted Success rate of each year.
  - Flight Number vs Orbit is plotted with colored Outcome classes
- Source code: https://github.com/Frk-Yvz/ibm_course/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- Generated SQL table from Pandas DataFrame created from a CSV file
  - Queried Launch Sites
  - Queried 5 records launched from ‘CCA*’
  - Queried average payload mass carried by F9 v1.1 booster
  - Queried the date of first successful landing on ground pad
  - Queried boosters successful at drone ship landing with payload mass between 4000 and 6000
  - Queried for total number of successful and failed mission outcomes
  - Queried for booster versions that carries maximum payload mass
  - Queried for boosters, launch sites, and month of the year for failed drone ship landing in 2015
  - Queried for ranking landing outcomes between 2010-06-04 and 2017-03-20
- Source code:

  https://github.com/Frk-Yvz/ibm_course/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Markers, Circles, Lines, and Marker Clusters are utilized on Folium Map.
- Markers are used to indicate locations of interest like Launch Sites.
- Circles are used to highlight some of the marked locations like Launch Sites.
- Marker Clusters are used to denote number of launches from Launch Sites.
- Lines are used to denote the distance between the two points of interest:
  - Launch site and a close by important landmark (Airport, highway, coastline etc.)
- Source code:

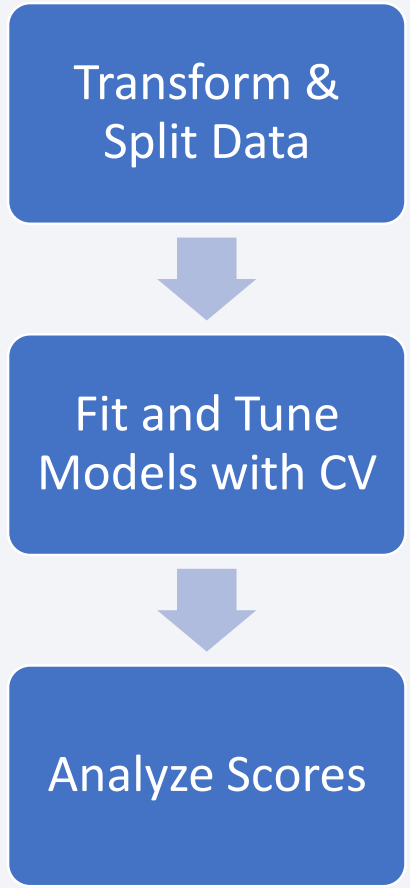
  https://github.com/Frk-Yvz/ibm_course/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- Added a dropdown to analyze ALL launch sites or specific launch sites
- Added a slider to select minimum and maximum payload of interest.
- Based on the selection of the dropdown a specific plots are generated:
  - Pie Chart:
    - Shows launches from all sites or shows successful and failed launches from the selected site
  - Scatter Plot:
    - Analyzes outcome of launch with respect to Payload Mass for each booster version
    - Shows outcomes for ALL sites or shows outcomes from a specific site.
    - The slider selection sets the interested Payload Mass range on the x axis interactively
- Added plots and interactions lets us study the following relations:
  - Launch Success per Site, Success vs Failure for selected site, Payload Mass vs Outcome for Booster types for all sites, Payload Mass vs Outcome for Booster types at selected site.
- Source code: https://github.com/Frk-Yvz/ibm_course/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Transformed the data and split it into training and test set.
- Utilized 4 different Machine Learning models.
  - Logistic Regression, SVM, Decision Tree, KNN
- Utilized grid search to find optimum parameters.
- Utilized 10 fold Cross Validation to minimize risks of under/over fitting
- Analyzed predictive capabilities to determine the best model.
- Source code: https://github.com/Frk-Yvz/ibm_course/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

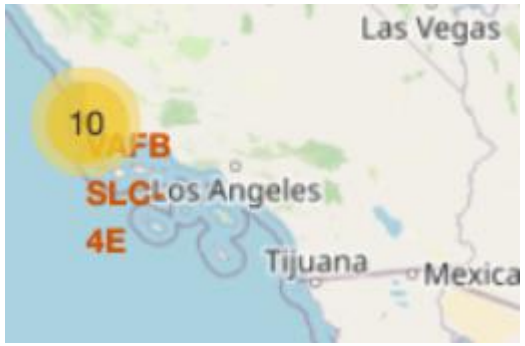


Results

- Exploratory data analysis results
  - Space X uses 4 different launch sites
  - Average payload of F9 v1.1 is 2928 kg
  - First successful landing happened in 2015 (5 years after the first launch!)
  - Many Falcon 9 boosters were successful at landing in drone ships with above average payloads
  - Very high rate of successful mission outcome is achieved.
  - Two boosters failed at landing in drone ships in 2015: F9 v1.1 B1012 and F9 v1.1 B1015
  - Landing success rate got better as years passed
- Interactive analytics demo in screenshots
- Predictive analysis results

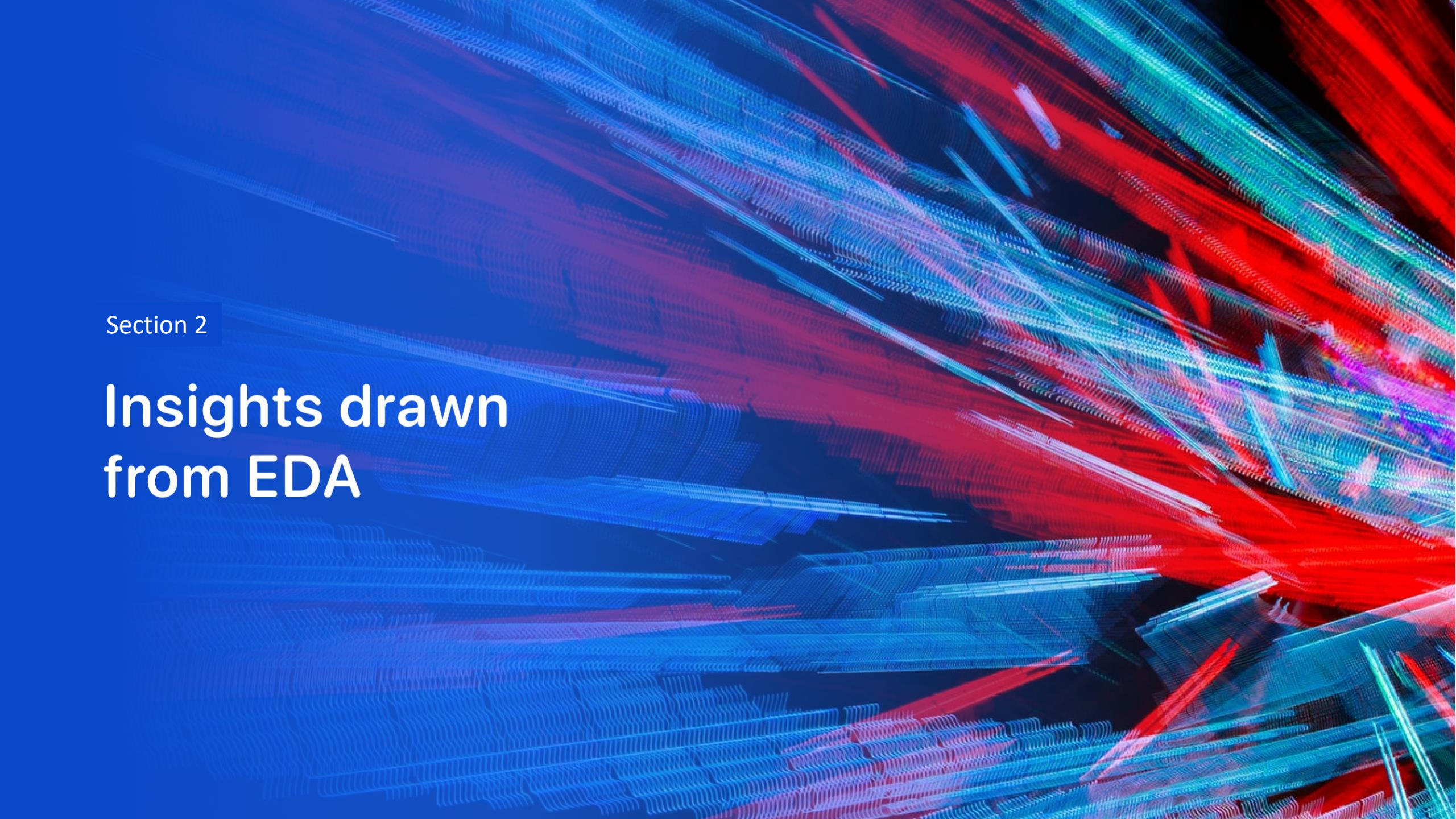
Results

- Interactive analytics demo in screenshots
  - Interactive analytics demonstrated the launch sites, their success rates and their logistic facilities and geologic features.
  - East coast hosts significantly more launches!



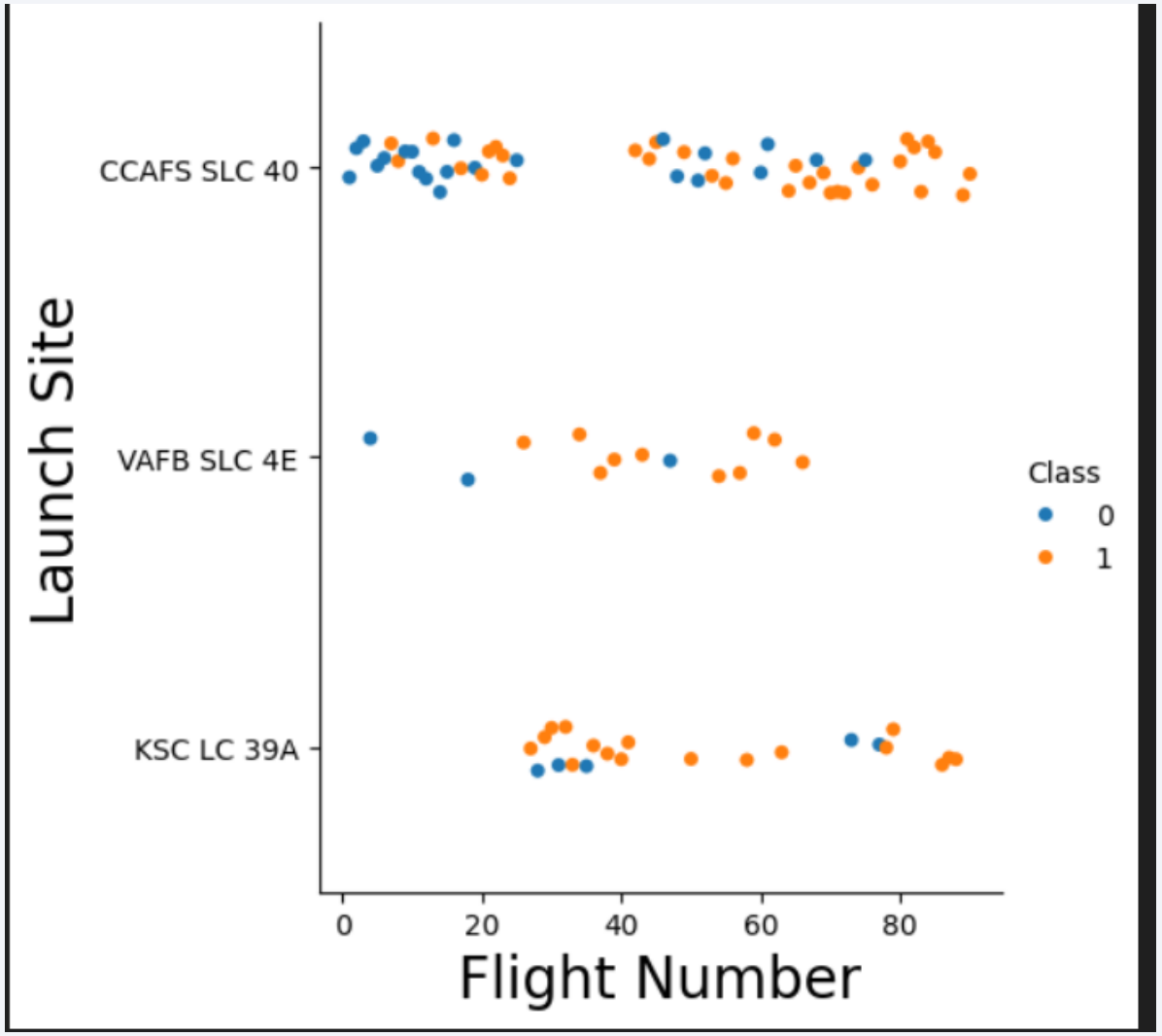
Results

- Predictive analysis results
  - Results show that Cross Validation and Parameter Tuning increases the model score.
  - Results show that the is no under fitting
  - Results show that there is no overfitting.
  - The test set performance suggests that all models are equivalent in performance.
  - The test set performance indicates we have False Positives
  - The training set performance shows that Decision Tree is better than the rest of the models.
  - Hence, I conclude that Decision Tree is the best model:
    - Training score: 0.875
    - Test score: 0.833


Section 2
Insights drawn from EDA

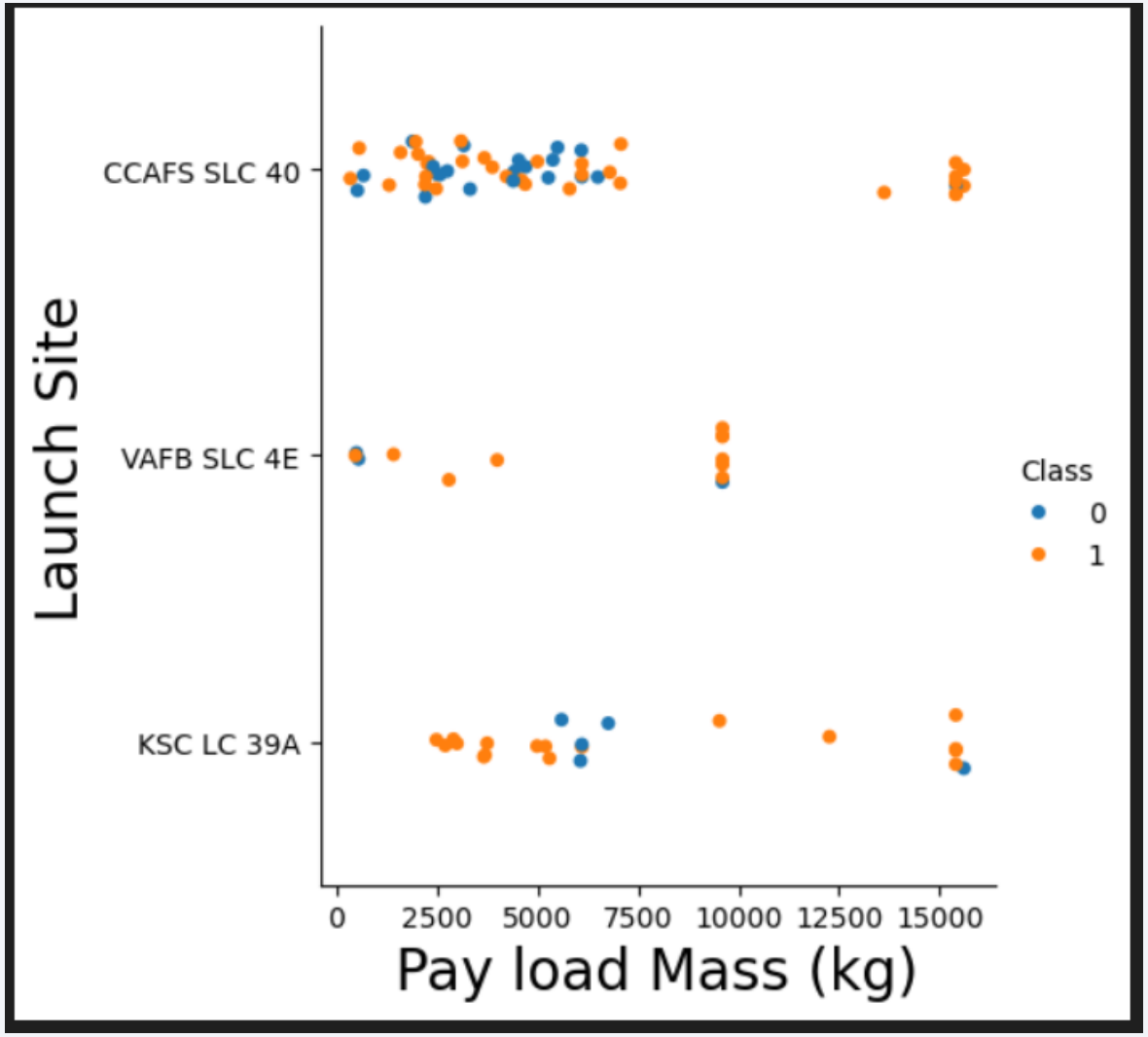
Flight Number vs. Launch Site

- Plot shows that launch sites with more flights have greater success rate



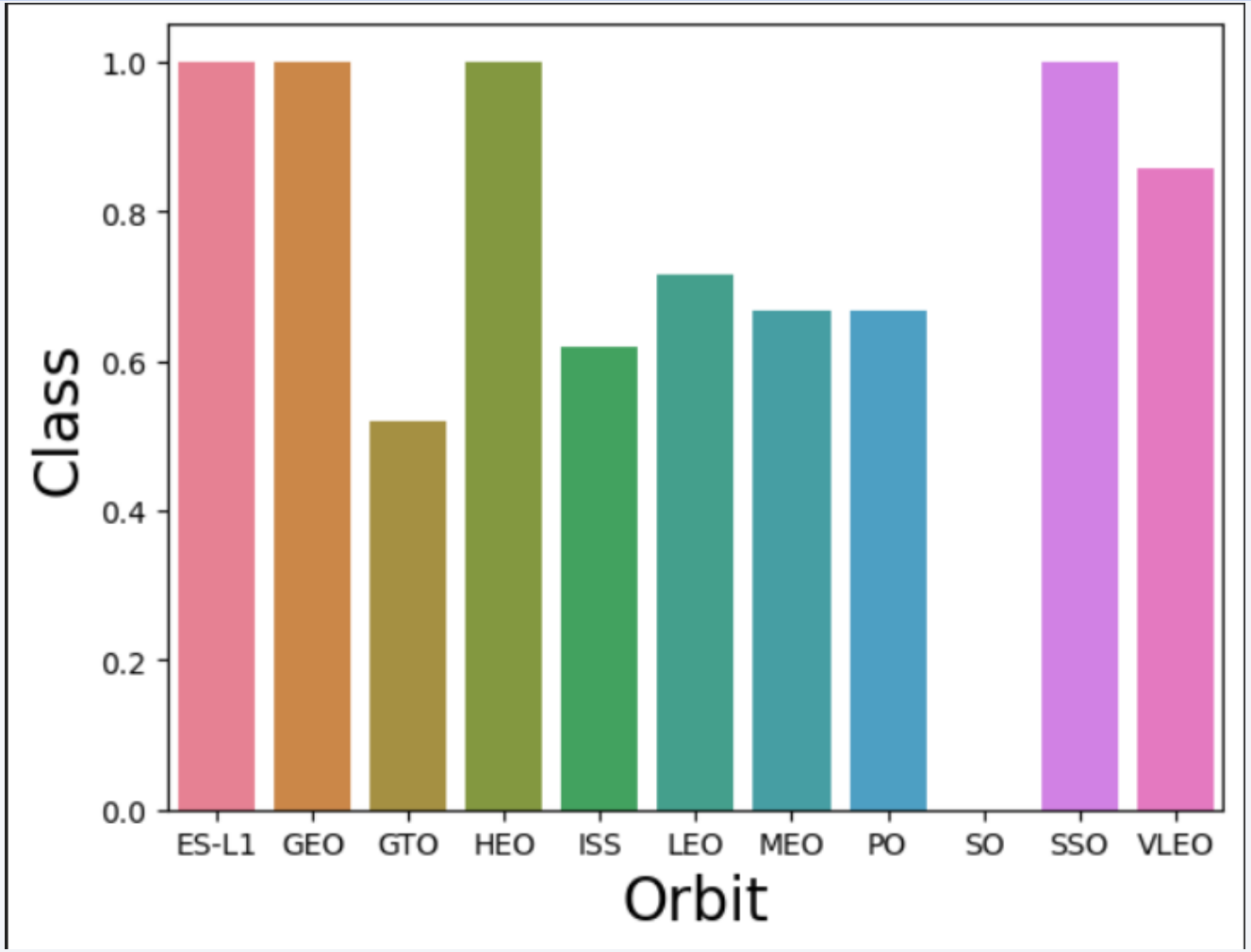
Payload vs. Launch Site

- Greater payload shows higher success rate for all sites especially CCAFS SLC 40



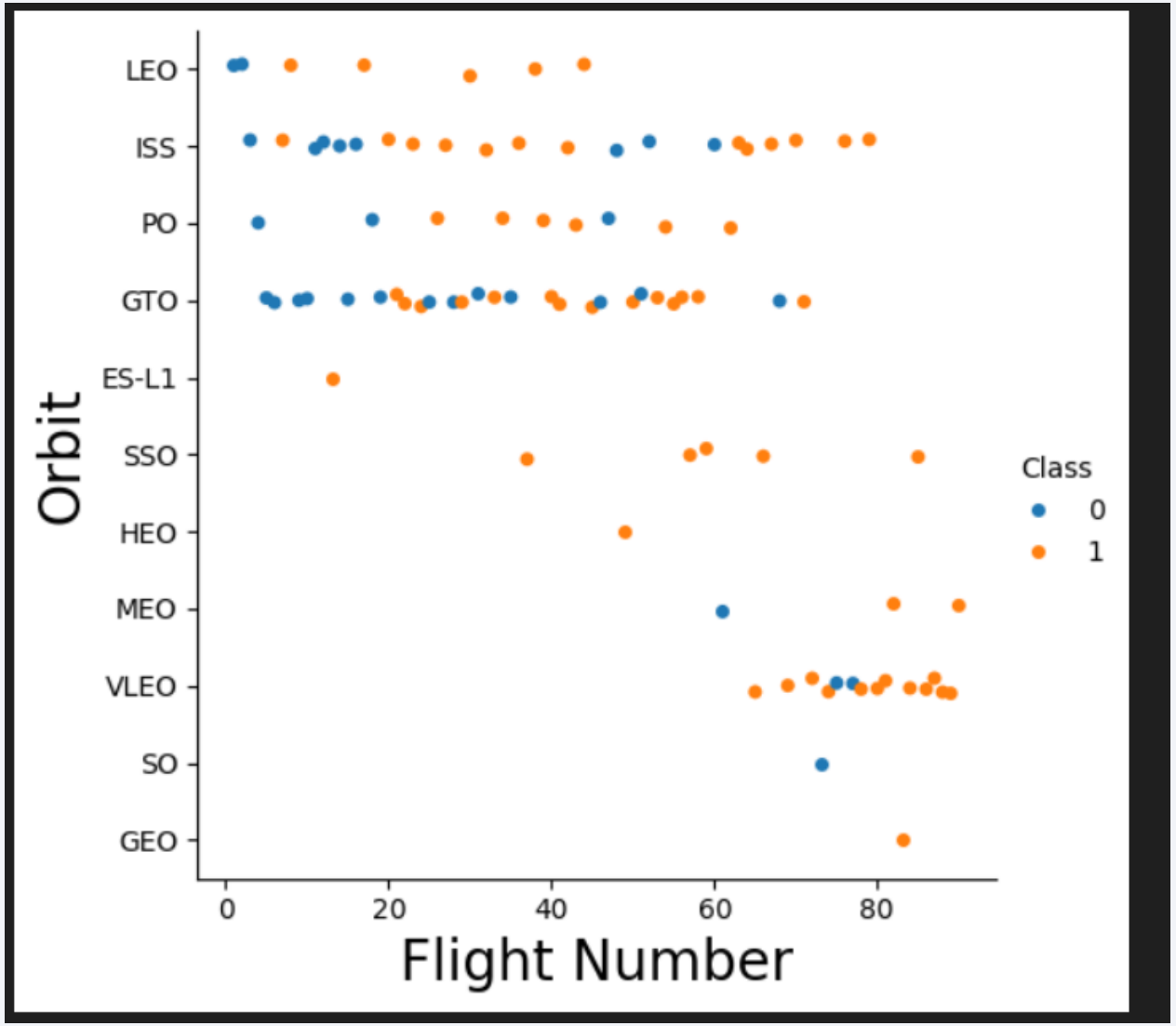
Success Rate vs. Orbit Type

- ESL1, GEO, HEO, SSO have perfect success rates
- VLEO has a high success rate
- Seems like orbit plays a role in the success of the mission!



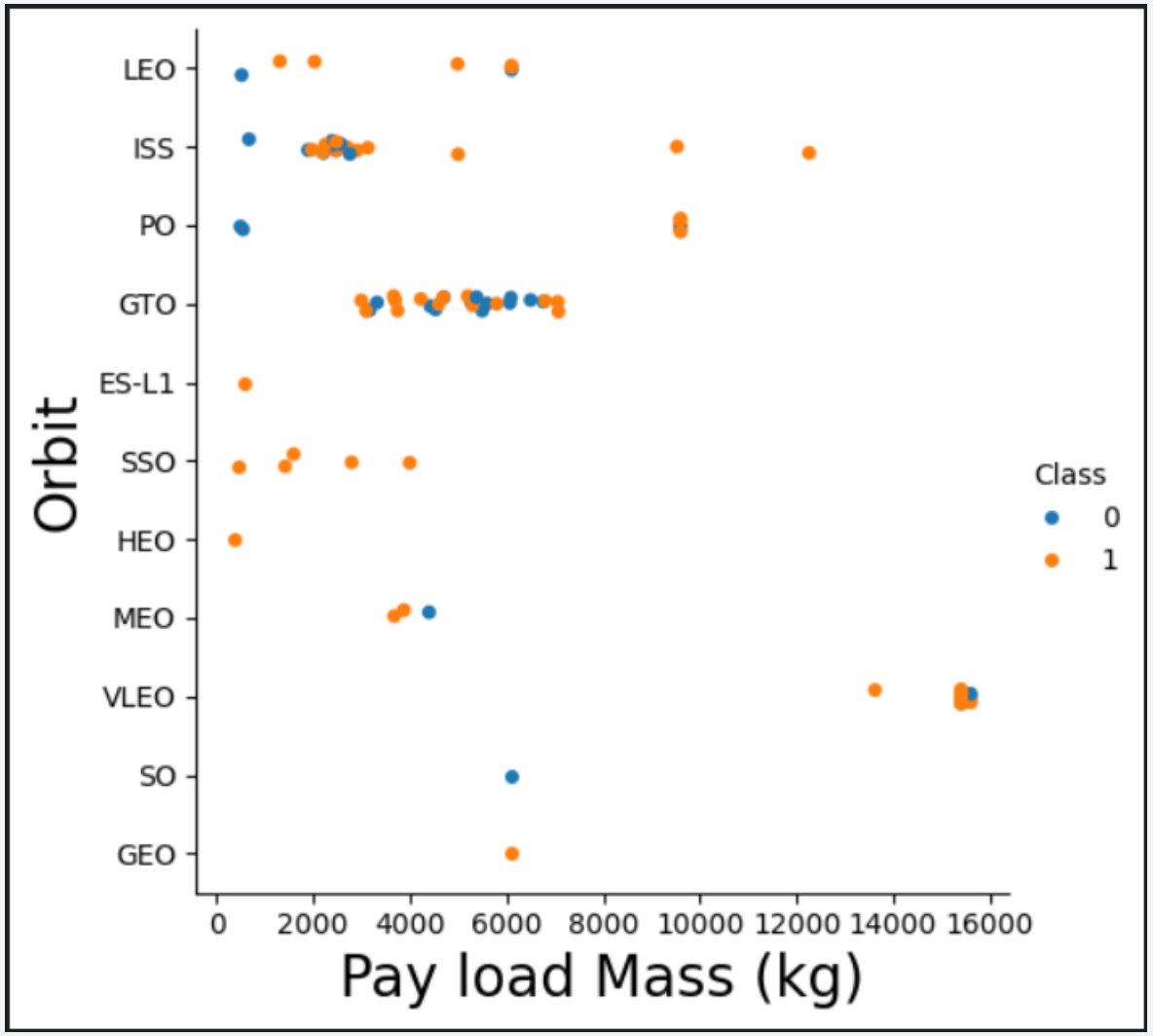
Flight Number vs. Orbit Type

- ESL1, GEO, HEO have only 1 flights despite their 100% success rate.
- Seems like success rate increases with flight number for Orbits as well.



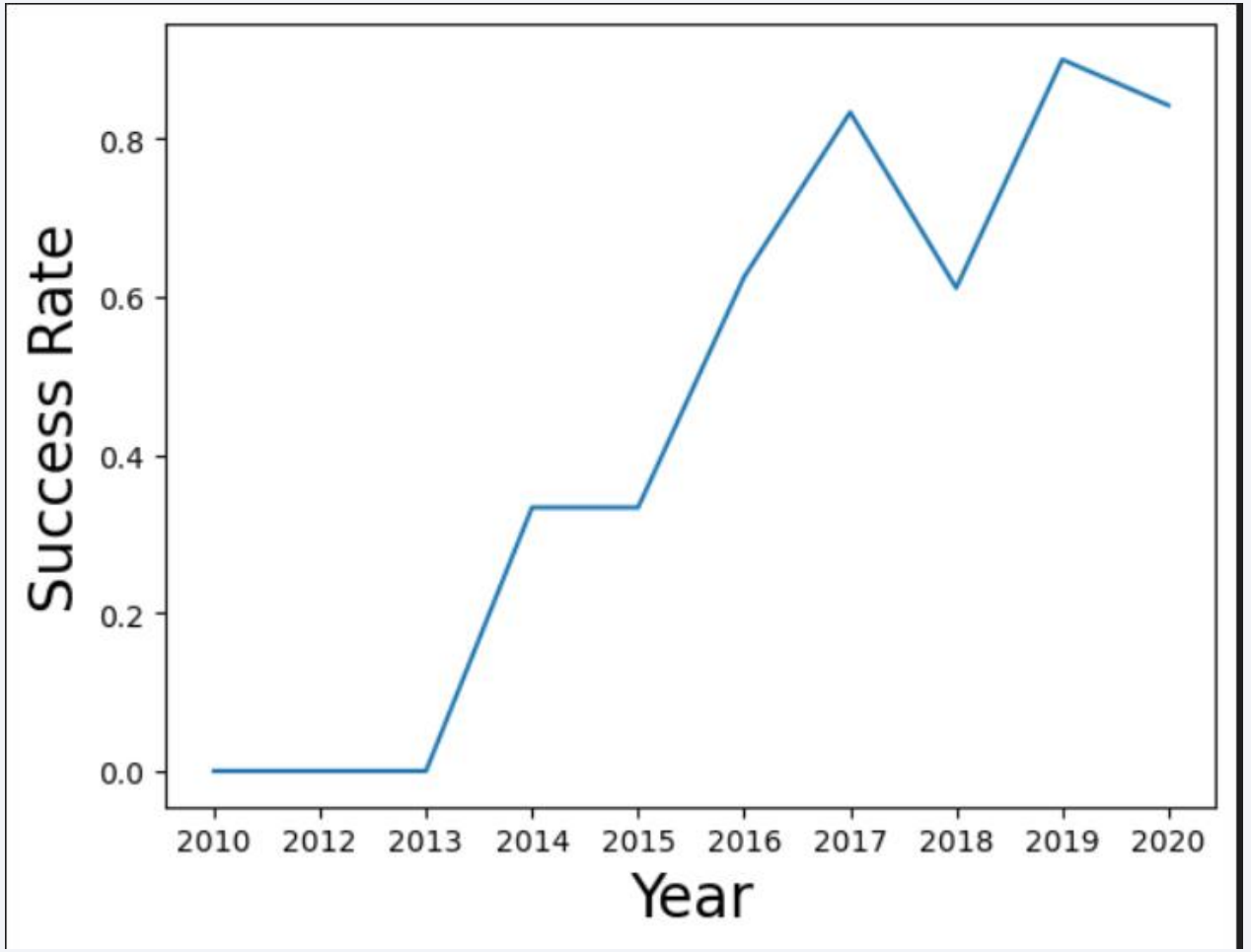
Payload vs. Orbit Type

- There is at best a weak relationship between the payload and success rate at Orbits.
- PO and ISS has higher success rate at higher payloads



Launch Success Yearly Trend

- As time progresses, the success rate gets better especially after 2013



All Launch Site Names

- DISTINCT keyword is used

```
%sql SELECT DISTINCT("Launch_Site") FROM "SPACEXTABLE";
```

```
 * sqlite:///my_data1.db
Done.
```

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

Launch Site Names Begin with 'CCA'

- WHERE .. LIKE ‘CCA%’ LIMIT 5 is used

```
[10]: %sql SELECT * FROM "SPACEXTABLE" WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

Done.

[10]:

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

- To choose NASA, WHERE ‘Customer’ = ‘NASA (CRS)’ is used
- Aggregate function SUM is used to calculate total mass

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") AS "Total_Payload_Mass" FROM "SPACEXTABLE" WHERE "Customer" = 'NASA (CRS)';
```

```
 * sqlite:///my_data1.db
Done.
```

Total_Payload_Mass
45596

Average Payload Mass by F9 v1.1

- To choose F9, WHERE ‘Booster_Version’ = ‘F9 v1.1’ is used
- Aggregate function AVG is used to calculate average mass

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

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM "SPACEXTABLE" WHERE "Booster_Version" = 'F9 v1.1';
```

```
 * sqlite:///my_data1.db
Done.
```

AVG("PAYLOAD_MASS__KG_")
2928.4

First Successful Ground Landing Date

- Successful landing outcomes are selected
- Aggregate function MIN is used to select first such date

```
%sql SELECT MIN("DATE") FROM "SPACEXTABLE" WHERE "Landing_Outcome" = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

MIN("DATE")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- WHERE is used to select landing outcome and AND is used to set the range of payload.

```
%sql SELECT "Booster_Version" FROM "SPACEXTABLE" WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" >
```

```
 * sqlite:///my_data1.db
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

- GROUP BY is used on ‘Mission_Outcome’ column to group unique types
- Aggregate function COUNT() is used on ‘Mission_Outcome’ to count

```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") FROM "SPACEXTABLE" GROUP BY "Mission_Outcome";
```

```
 * sqlite:///my_data1.db
Done.
```

Mission_Outcome	COUNT("Mission_Outcome")
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- Booster versions are selected with a specific Payload_Mass using WHERE.
- To find the maximum payload mass, a subquery is used with aggregate function MAX on ‘PAYLOAD_MASS_KG’

```
%sql SELECT "Booster_Version" FROM "SPACEXTABLE" WHERE "PAYLOAD_MASS__KG_" = (
```

```
 * sqlite:///my_data1.db
Done.
```

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

- WHERE clause is used to select the year and failed landing outcomes.
- Aggregate function substr is used to select specific parts of the ‘DATE’ column

```
%sql SELECT substr("DATE",6,2), "Booster_Version", "Launch_Site" FROM "SPACEXTABLE" WHERE substr("DATE",0,5) = '2015' AND "
```

```
* sqlite:///my_data1.db
Done.
```

substr("DATE",6,2)	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

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

- WHERE clause with multiple conditions is used to select between the dates 2010-06-04 and 2017-03-20
- GROUP BY is used on ‘Landing_Outcome’ to group unique outcomes and aggregate function COUNT to find counts of each group.

```
%sql SELECT "Landing_Outcome", COUNT("Landing_Outcome") AS "Count" FRO
```

```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	Count
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

Section 3

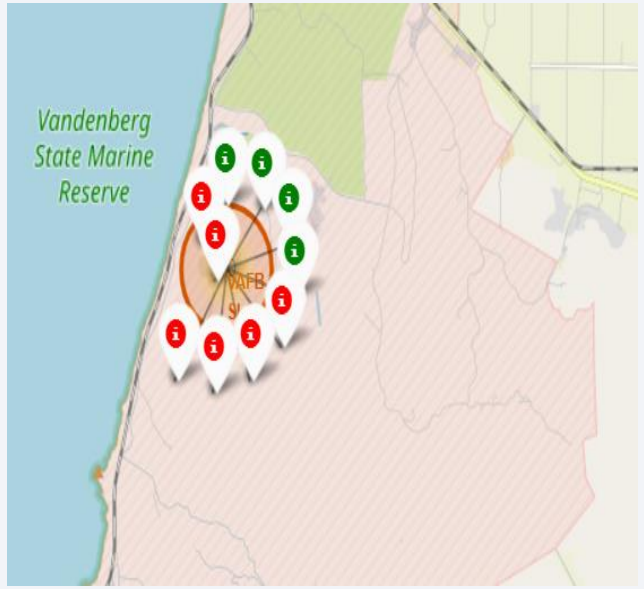
Launch Sites Proximities Analysis

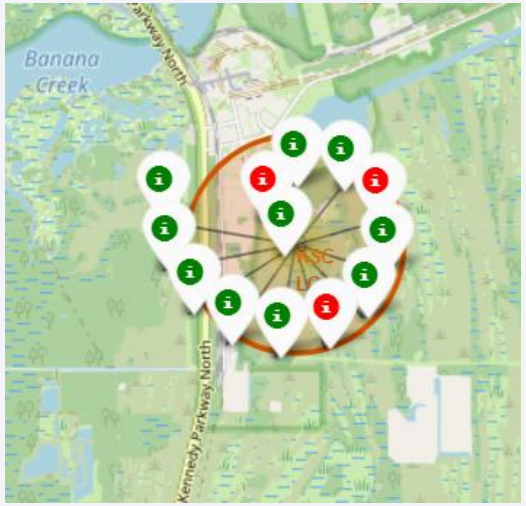
SPACE X LAUNCH SITES MAP

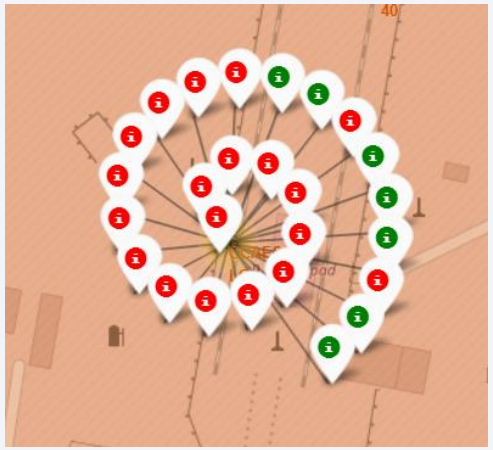
- We see that the launches are from US, the east & west coast

SPACE X LAUNCH SITES OUTCOME MAP

- We can investigate the success at each launch site graphically on this map as well

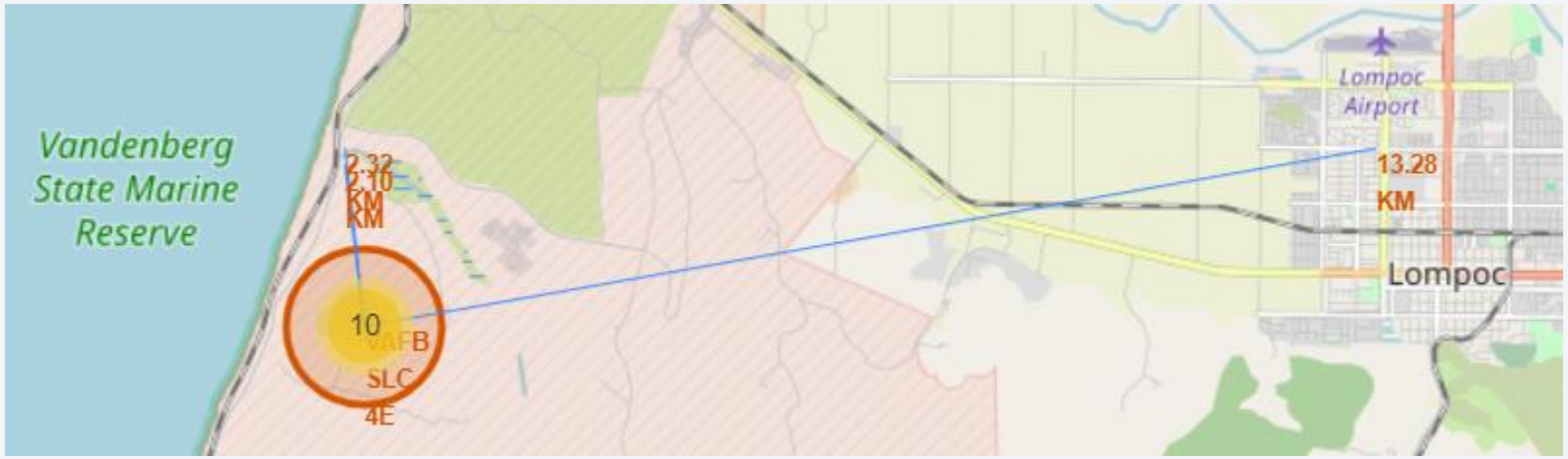


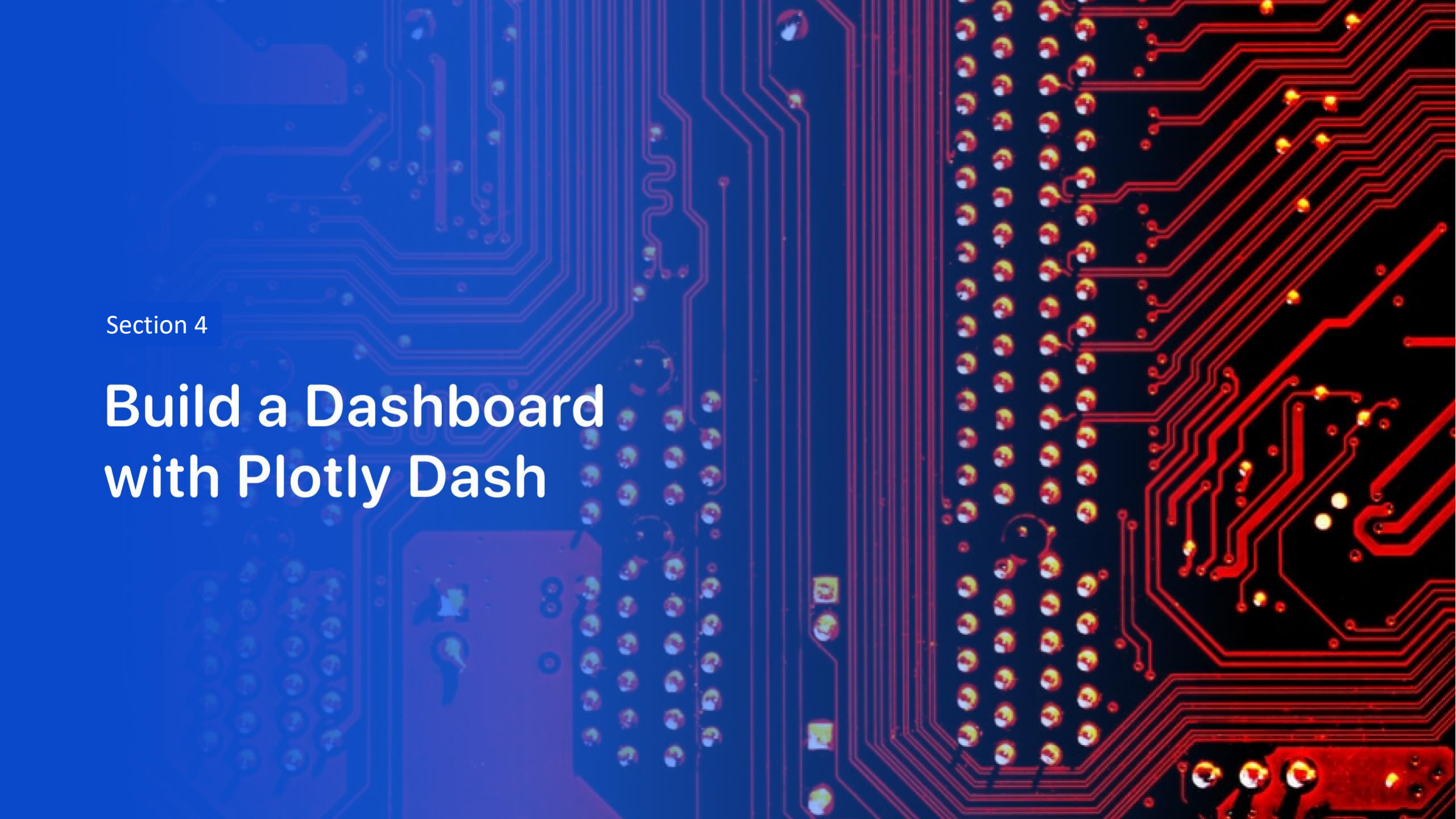




SPACE X LAUNCH SITE & PROXIMITY

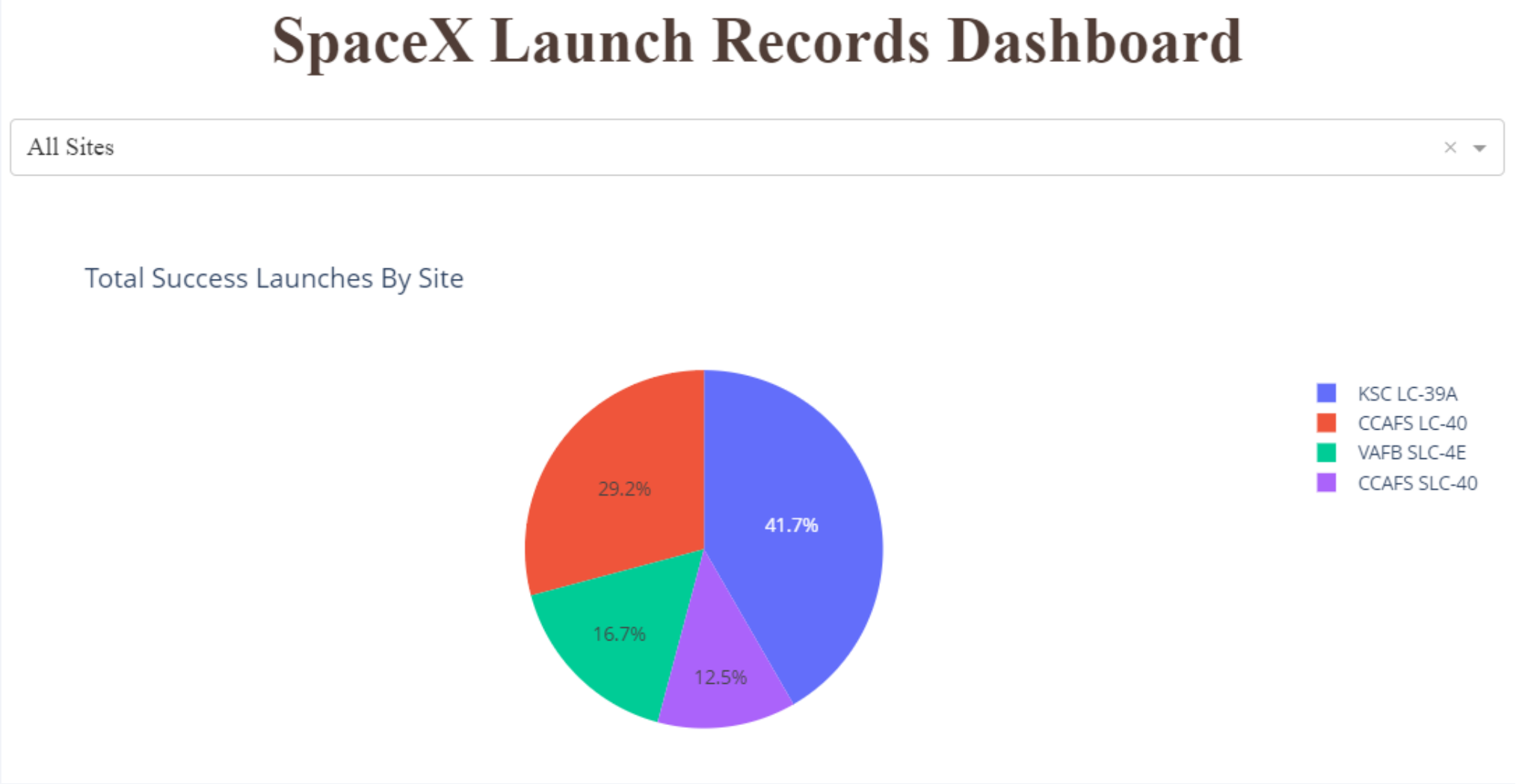
- Launch site is in close proximity to coastline
- Launch site is not in close proximity to highway
- Launch site is not in close proximity to railways
- Launch site is not in close proximity to the cities




Section 4
Build a Dashboard with Plotly Dash

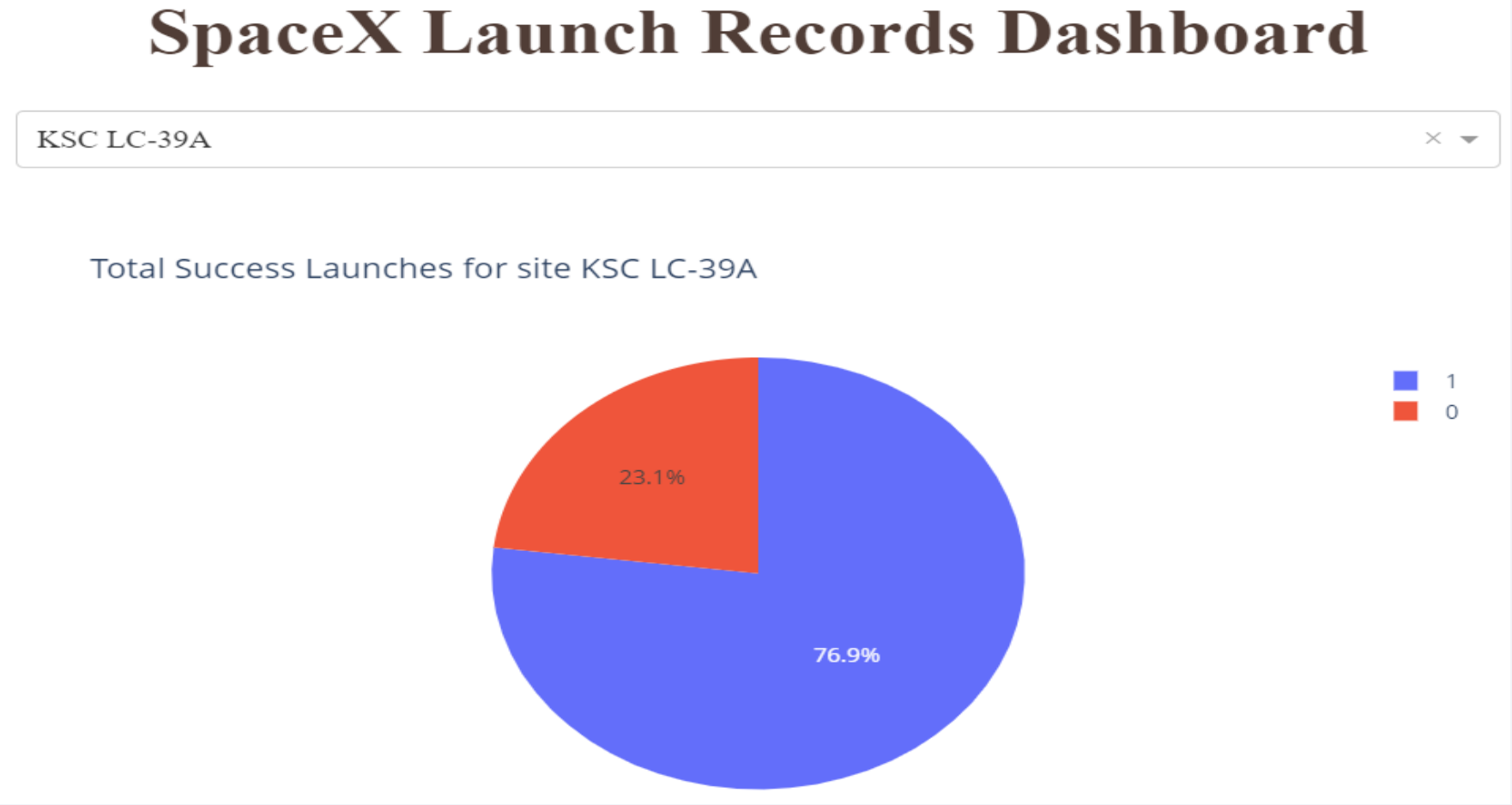
Dashboard Success Count for ALL Sites

- KSC LC-39A has the highest successful launches



Dashboard Pie chart of Highest Success Ratio Site

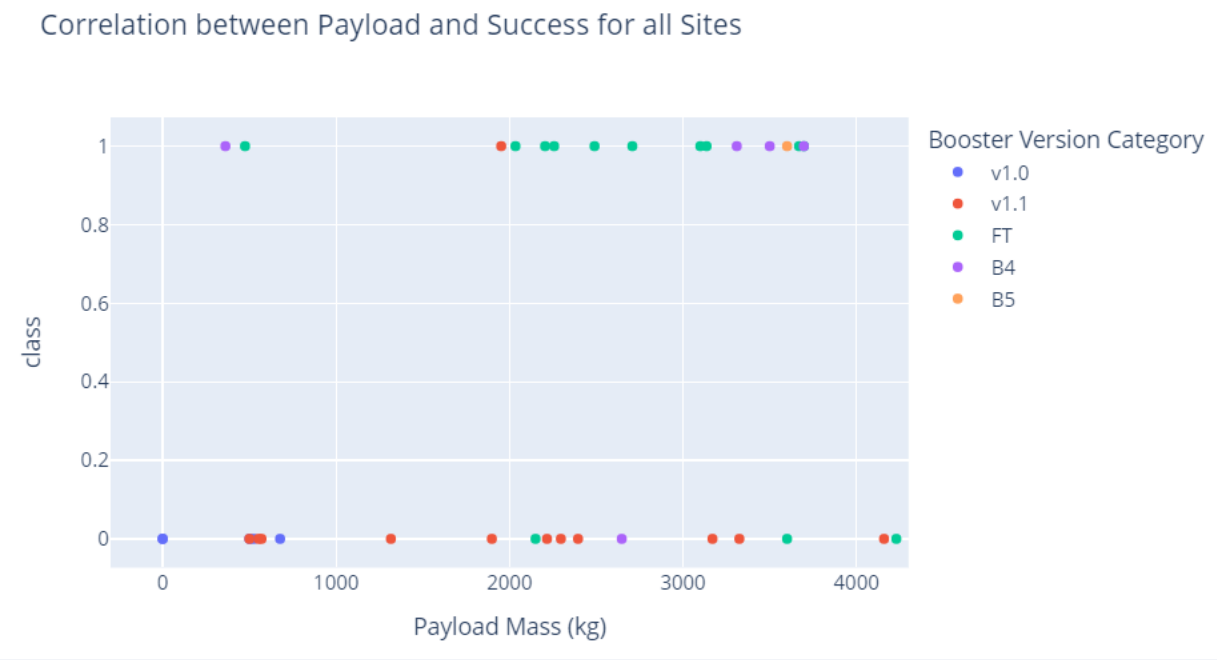
- KSC LC-39A achieved 76.9% success rate with a failure rate of 23.1%



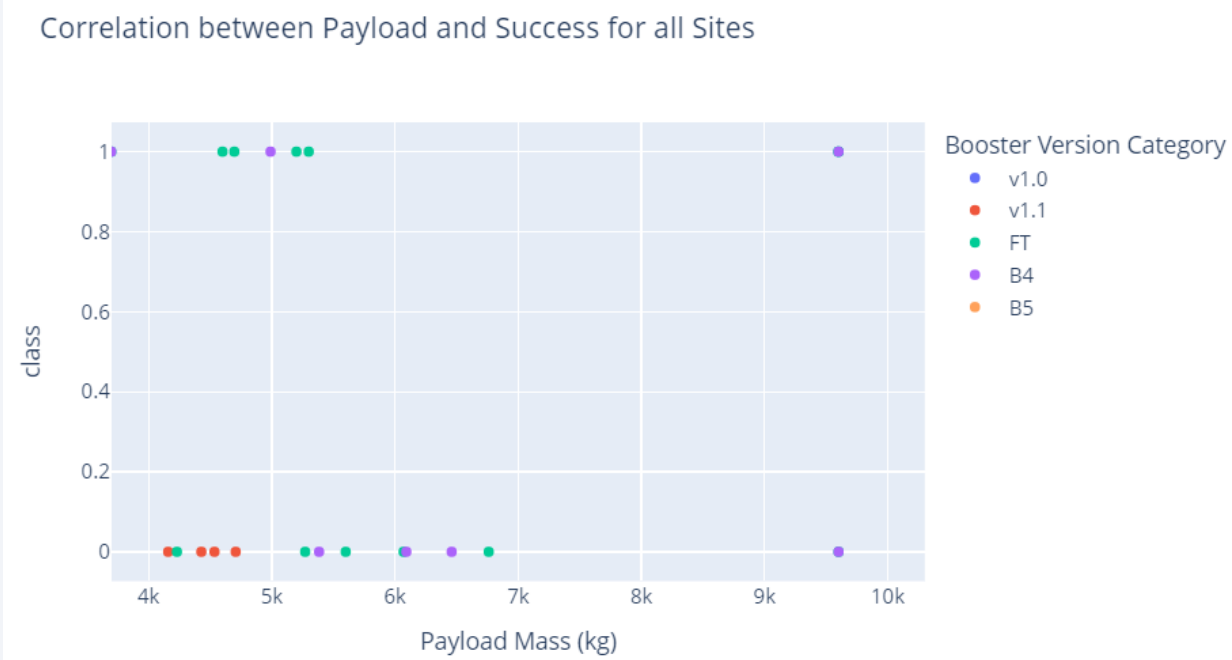
Dashboard Payload vs Launch Outcome for ALL Sites

- We see that success rates for low weighted payloads is higher!

Payload 0kg – 4000kg



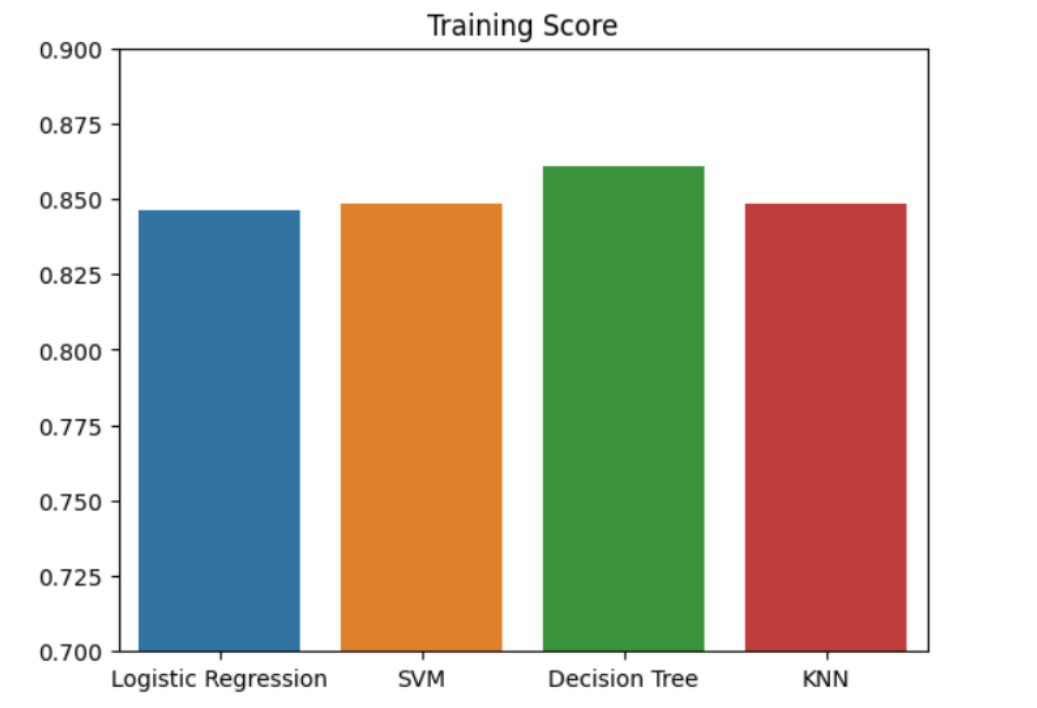
Payload 4000kg – 10000kg

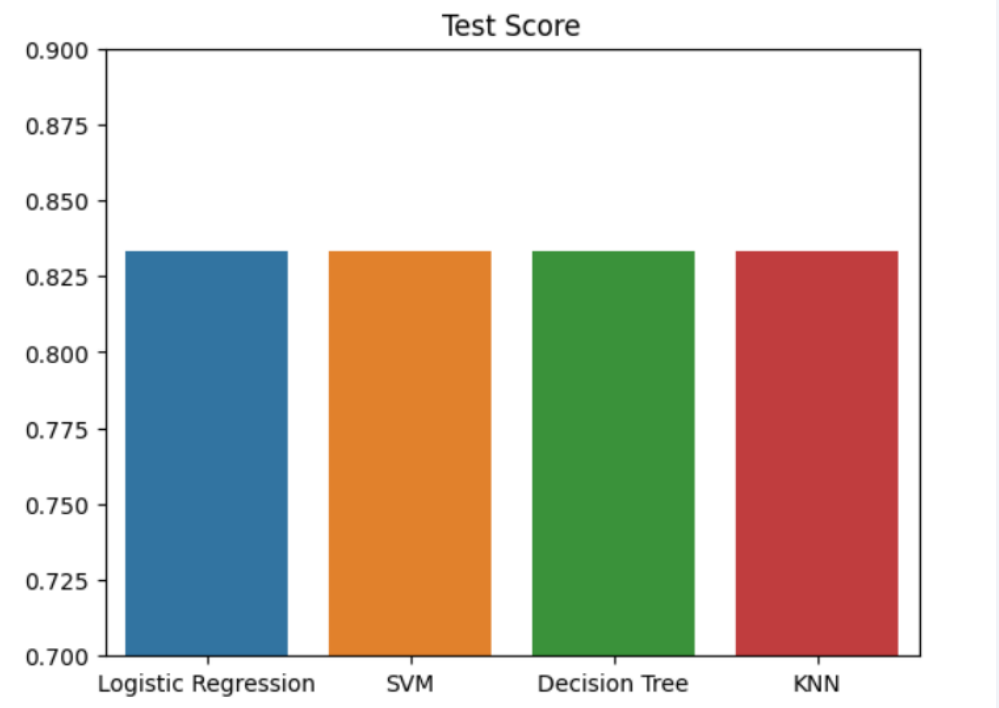


Section 5

Predictive Analysis (Classification)

Classification Accuracy



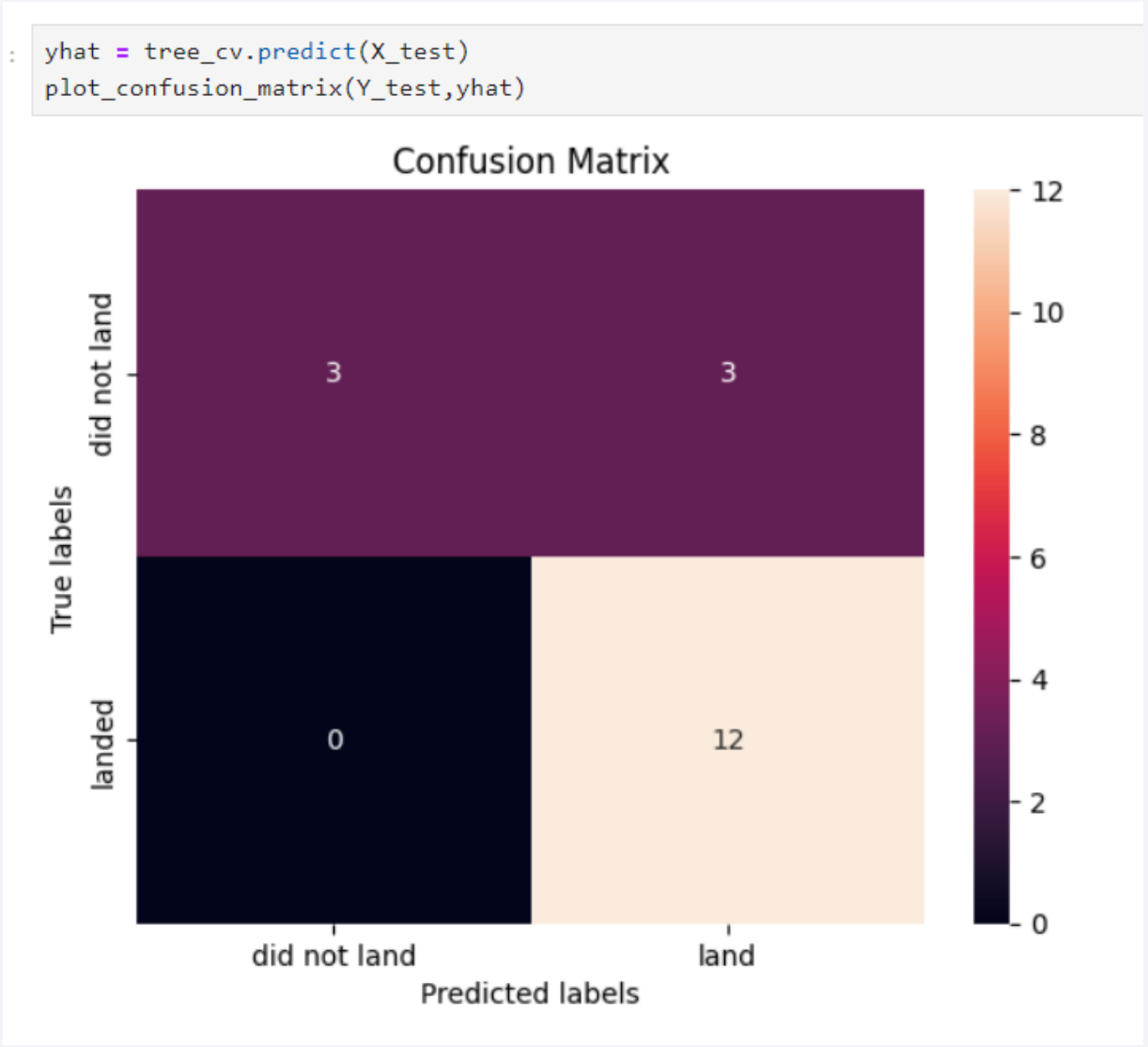


- Models have same test score. Decision Tree model has slightly higher training score. Hence, I am picking Decision Tree as the best model.
- Actually, we need more test data to see if the performance difference seen in training carries to the test performance!

Confusion Matrix

- Confusion matrix shows that the model can be improved on False Positive estimations. Meaning that it predicts successful landing whereas it is actually a failure for 3 cases.

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



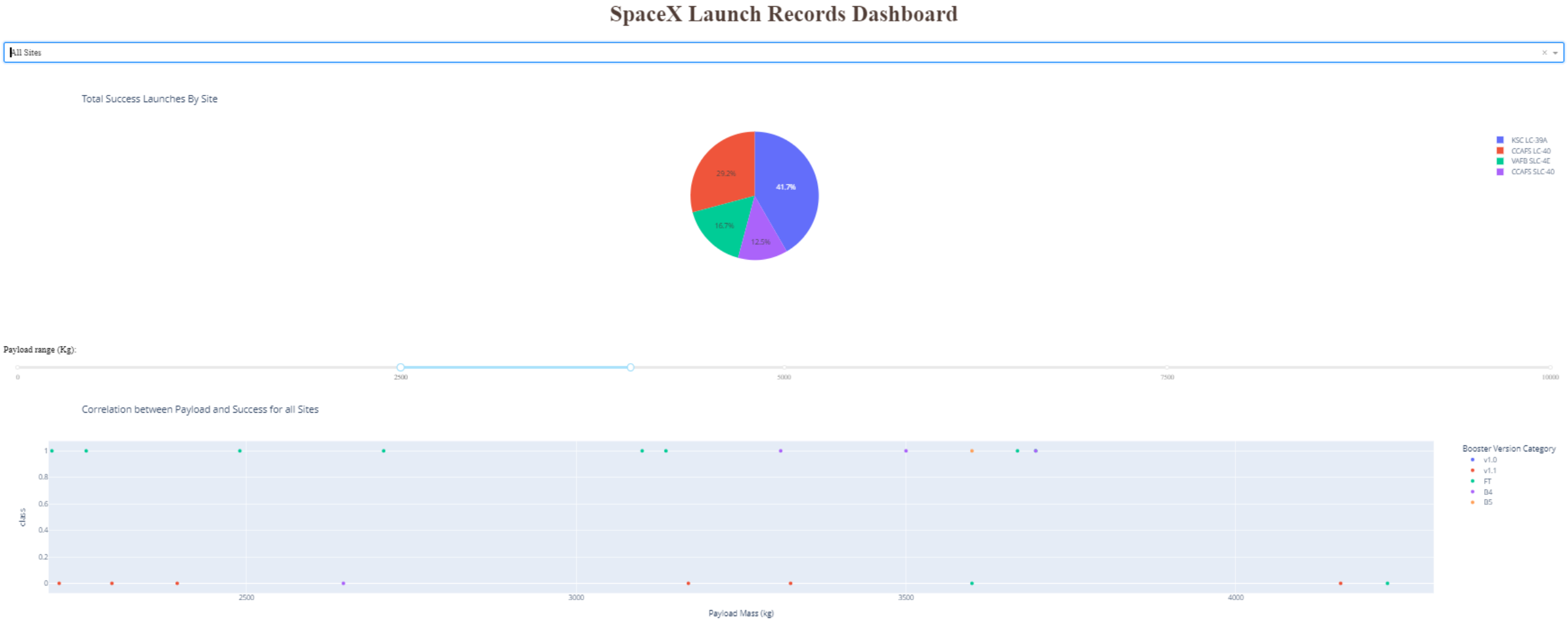
Conclusions

- There is a positive correlation between the number of flights operated from a site and the success rates.
- Orbits ESL1, GEO, HEO, SSO have perfect success rates.
- Success rate increases with flight number for Orbits as well.
- KSC LC-39A has the highest successful launches.
- As time progresses, the success rate gets better especially after 2013.

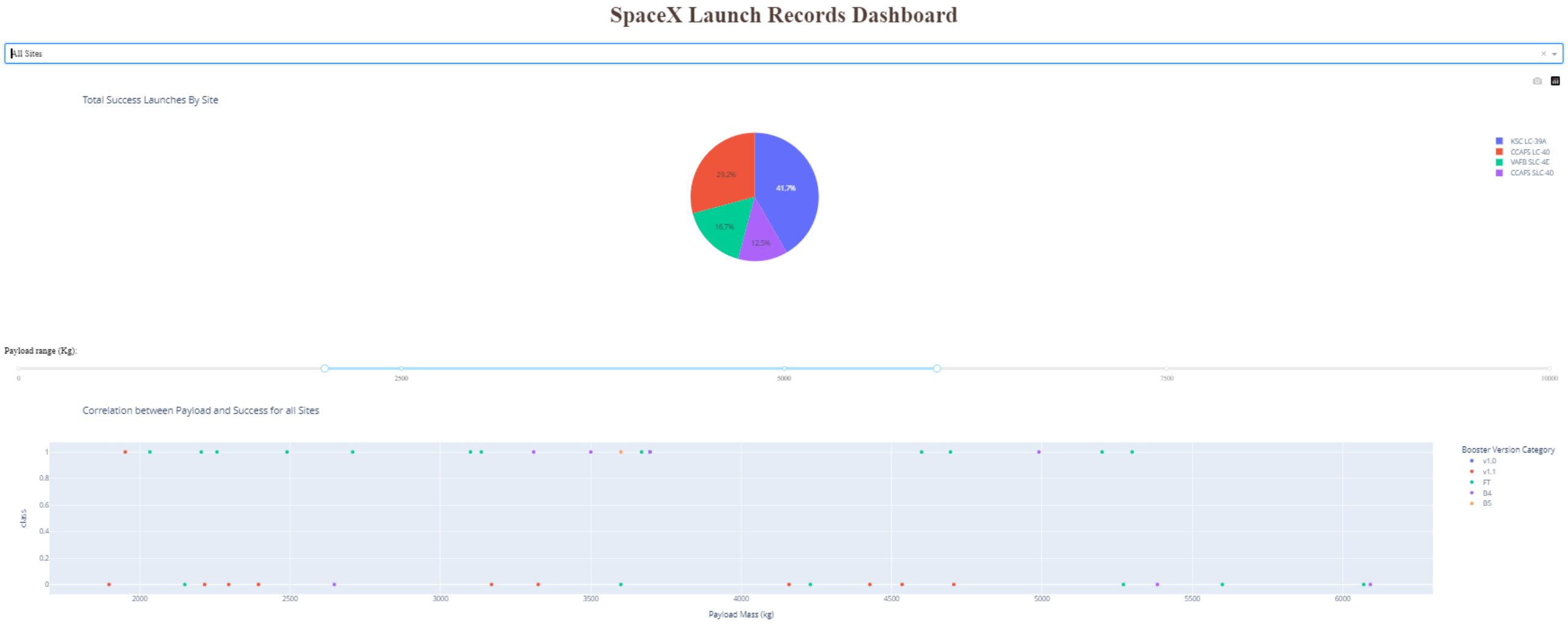
Appendix - SQL Queries

- Task 1: SELECT DISTINCT("Launch_Site") FROM "SPACEXTABLE";
- Task 2: SELECT * FROM "SPACEXTABLE" WHERE "Launch_Site" LIKE 'CCA%';
- Task 3: SELECT SUM("PAYLOAD_MASS__KG_") AS "Total_Payload_Mass" FROM "SPACEXTABLE" WHERE "Customer" = 'NASA (CRS)';
- Task 4: SELECT AVG("PAYLOAD_MASS__KG_") FROM "SPACEXTABLE" WHERE "Booster_Version" = 'F9 v1.1';
- Task 5: SELECT MIN("DATE") FROM "SPACEXTABLE" WHERE "Landing_Outcome" = 'Success (ground pad)';
- Task 6: SELECT "Booster_Version" FROM "SPACEXTABLE" WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS__KG_" < 6000;
- Task 7: SELECT "Mission_Outcome", COUNT("Mission_Outcome") FROM "SPACEXTABLE" GROUP BY "Mission_Outcome";
- Task 8: SELECT "Booster_Version" FROM "SPACEXTABLE" WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM "SPACEXTABLE");
- Task 9: SELECT substr("DATE",6,2), "Booster_Version", "Launch_Site" FROM "SPACEXTABLE" WHERE substr("DATE",0,5) = '2015' AND "Landing_Outcome" = 'Failure (drone ship)' ;
- Task 10: SELECT "Landing_Outcome", COUNT("Landing_Outcome") AS "Count" FROM "SPACEXTABLE" WHERE "DATE" >= '2010-06-04' AND "DATE" <= '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY "Count" DESC;

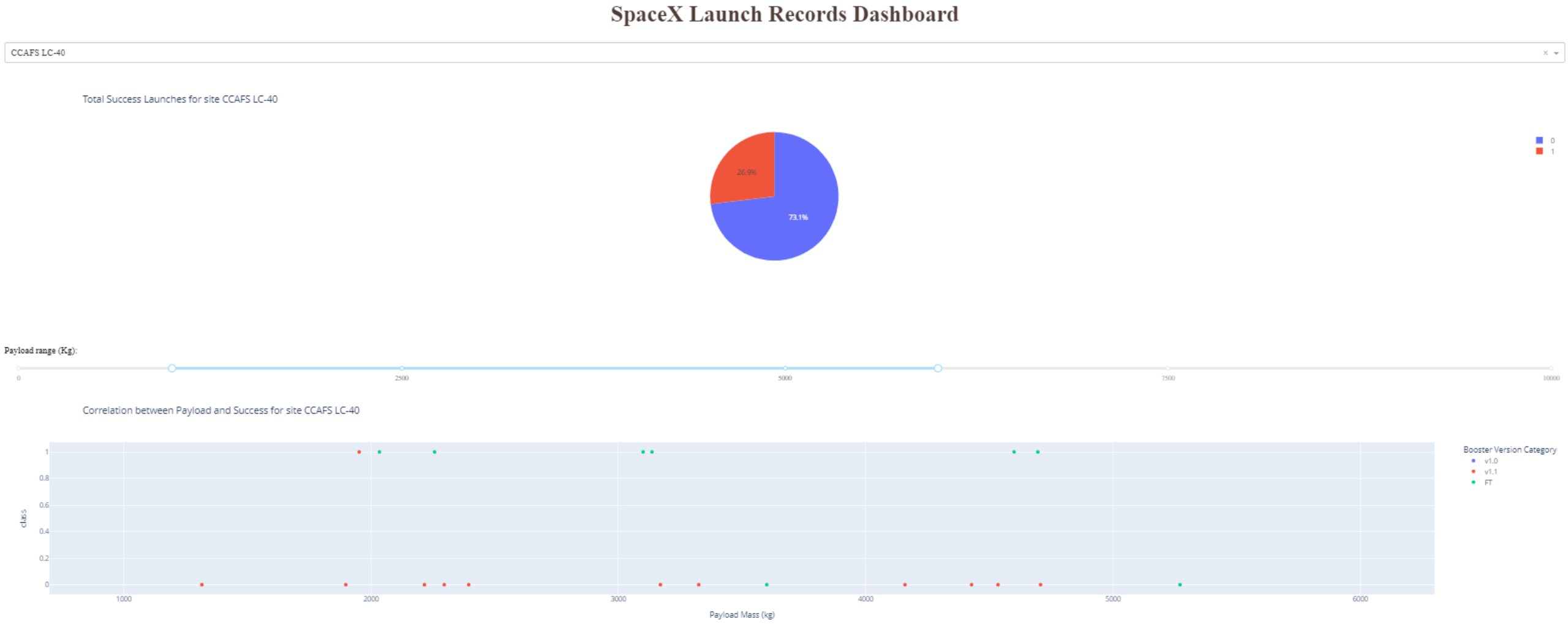
Appendix - Dashboard Visuals



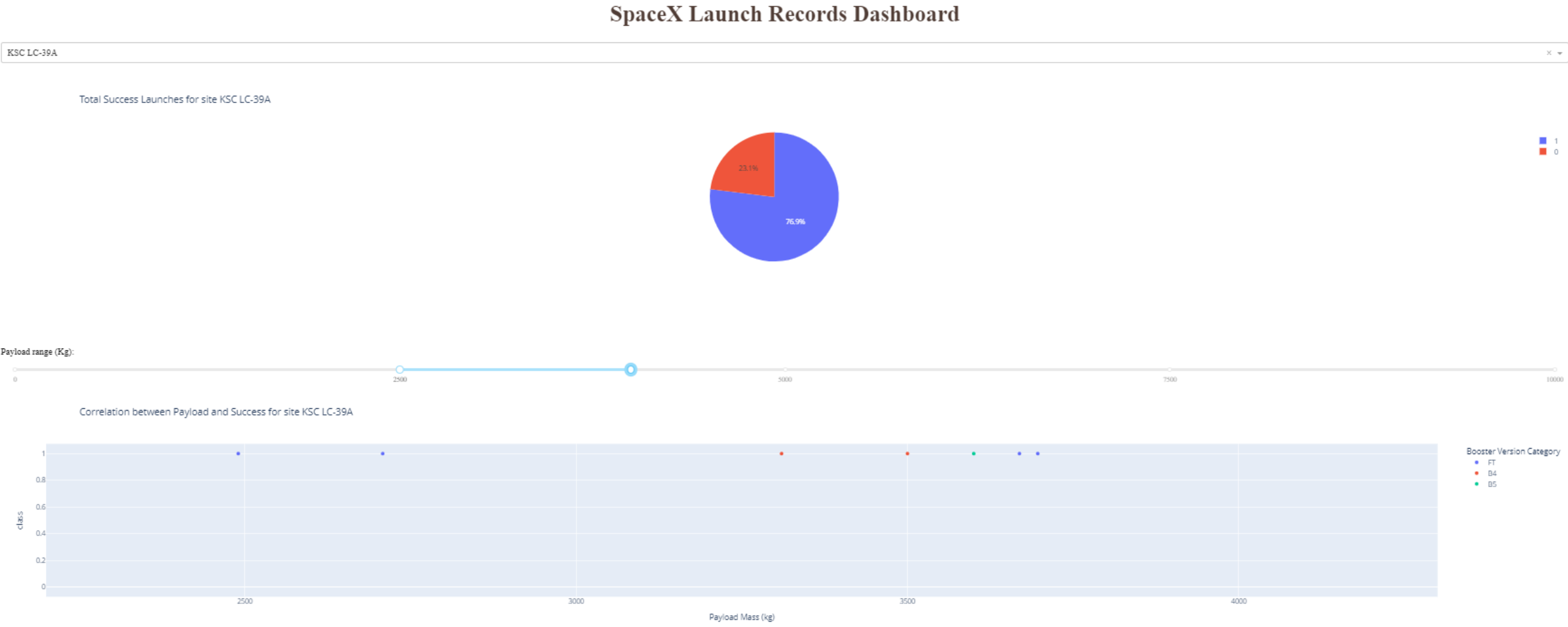
Appendix - Dashboard Visuals



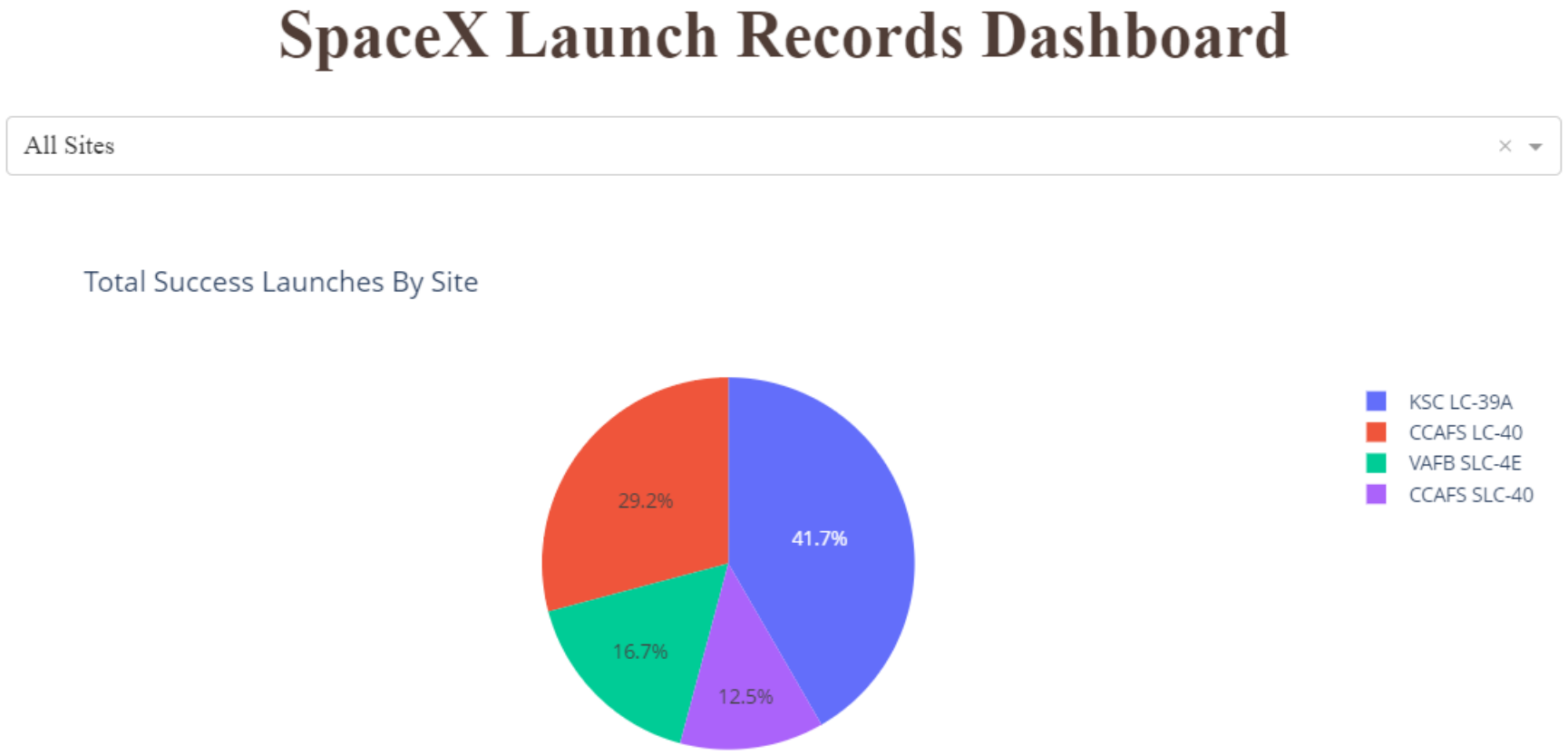
Appendix - Dashboard Visuals



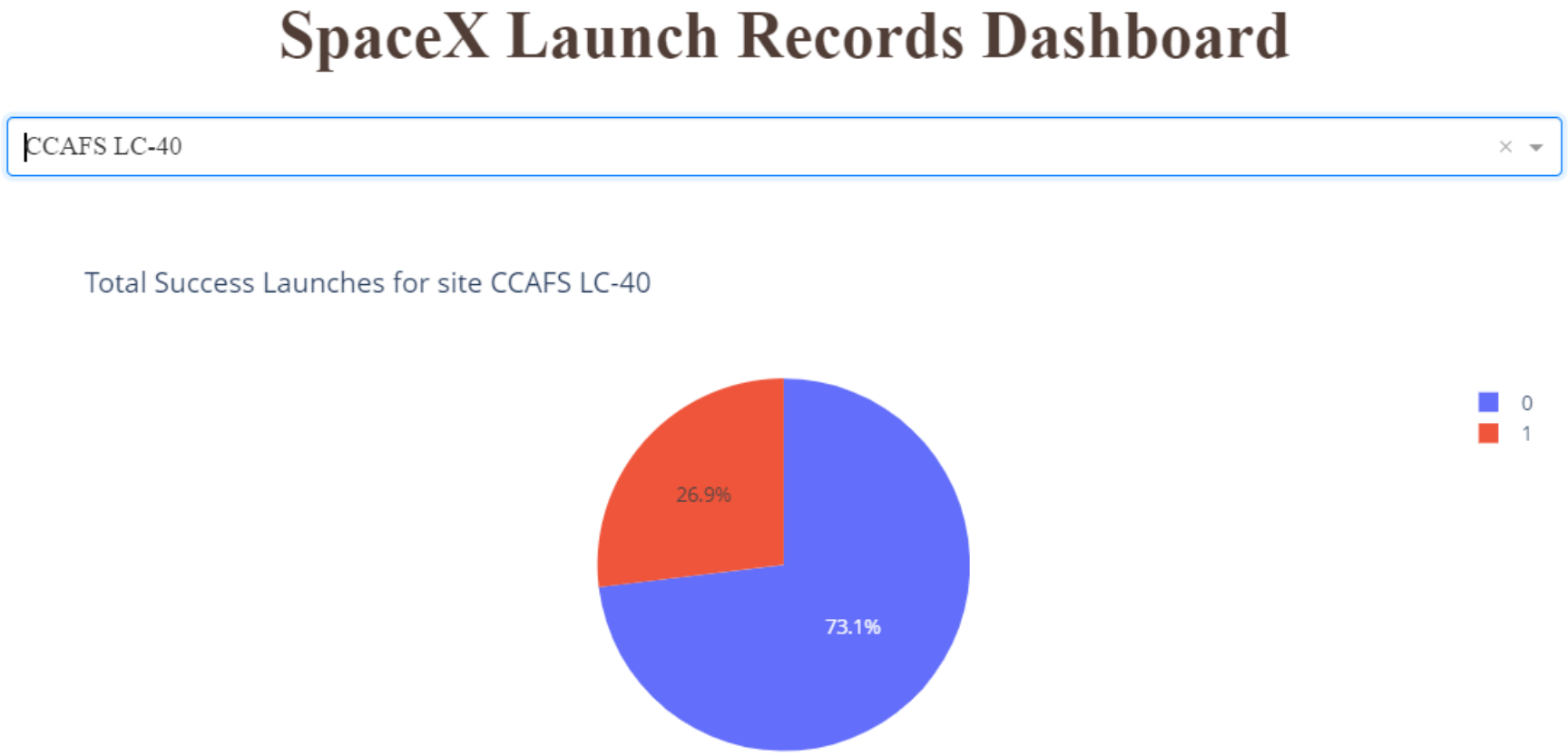
Appendix - Dashboard Visuals



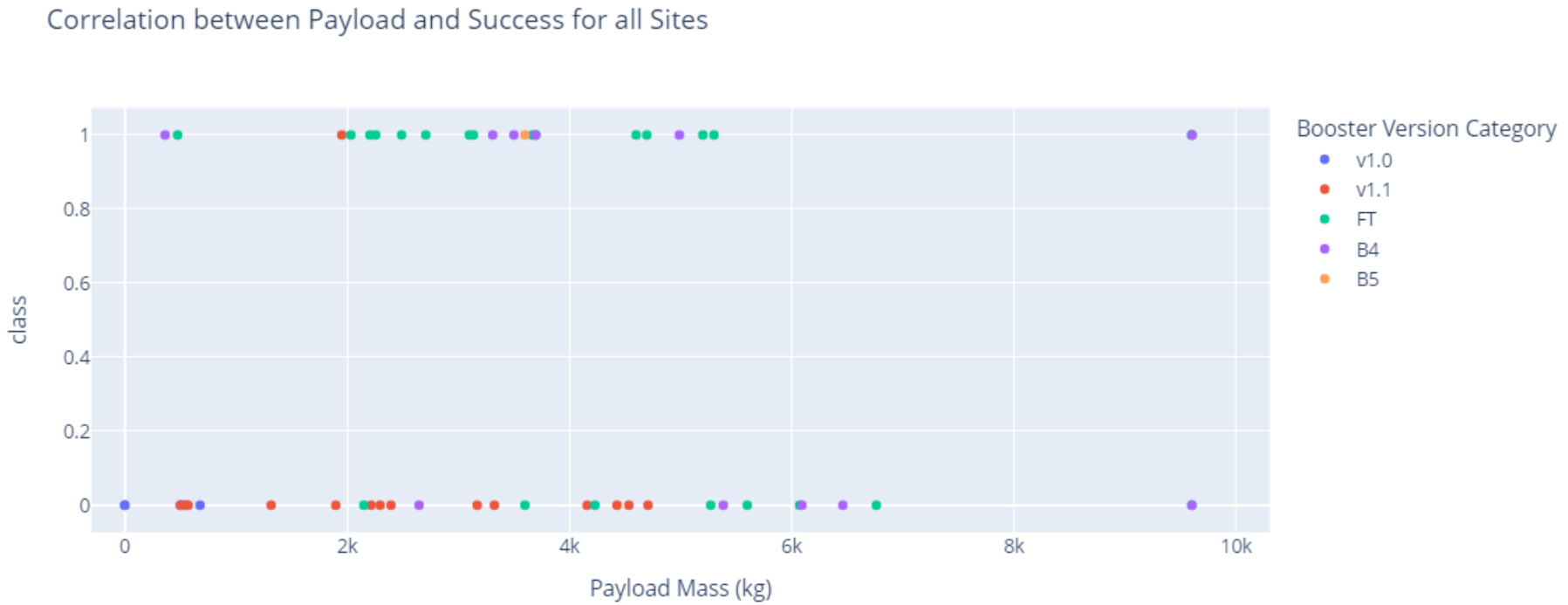
Appendix - Dashboard Visuals



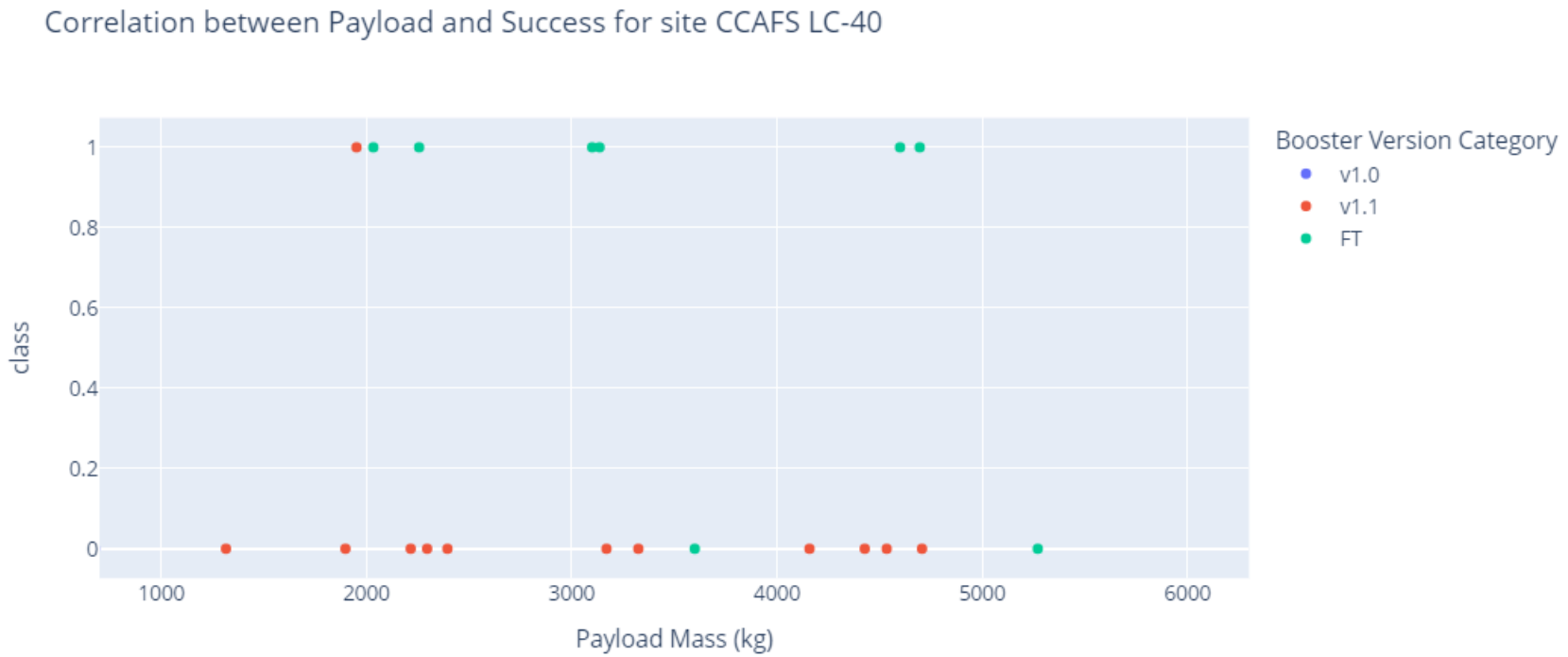
Appendix - Dashboard Visuals



Appendix - Dashboard Visuals



Appendix - Dashboard Visuals



Thank you!