IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Tim Presland
May 2025

Outline

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

Executive Summary

This Capstone project was carried out for "Space Y", using data science techniques to assess the success rates achieved by SpaceX. SpaceX uses its "Falcon 9" rocket, with the aim of successfully landing and recovering the stage 1 rocket component. This critical part of the mission determines whether costs associated with rocket launches can be significantly reduce, thus providing competitive advantage.

In this project, the following methodologies were used:

- Data collection from the SpaceX APIs and web scraping from historical data relating to the Falcon 9 Wikipedia
- Data wrangling, to assess and evaluate the datasets and attributes available, e.g. data relating to launch sites, orbits and mission outcomes
- Exploratory Data Analysis (EDA), using data visualisations and SQL analysis
- Create interactive visual analytics, using Folium maps to describe launch sites and Dashly dashboarding interactively allow analysis of success rates
- Predictive analysis through building machine learning models, to assess the best models to use along with analysis of predictions of success outcomes

In summary, the key project findings were:

- The most successful launch sites appeared to be CCAFS LC40 and KSC LC-39A, with success rates >70%
- Payload appears to be a key factor in number of successful outcomes (the range 2000Kg to 7000Kg having most success)
- The most successful rates were achieved using orbits HEO, GEO, ES-L1 and SSO
- Success rates improved significantly between 2013 and 2017

Introduction

Project background and context

Rocket launches are potentially very expensive, with some organisations quoting over $160M to launch. SpaceX has innovated to allow recovery of the stage 1 rocket, reducing its costs to around $60M. This Capstone project was therefore carried out for "Space Y", using data science techniques to assess the success rates achieved by SpaceX and inform how Space Y can compete and provide competitive advantage.

Purpose of analysis

The aims of this data science project were to gather historical data for Space X launches in order to analyse the key features which determine and influence successful outcomes. Key findings sought were:

- Relevance and geography of launch sites, using Folium maps
- Relationships between key data attributes and outcomes, using visualisations, SQL analysis and dashboarding
- Key features included payload, launch sites and orbits
- Finally, use of machine learning models provided valuable feedback on how predictable, and using which key features, allowed the most successful predictive modelling

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Python tools were used to extract SpaceX JSON data from its API services (api.spacexdata.com)
  - Additional historical data was derived using web scraping from the Falcon 9 Wiki pages
- Perform data wrangling
  - Data was first cleaned to deal with missing data (e.g. payload mass mean value used)
  - Data was processed by creating DataFrames and filtering on key features e.g. 'payload', 'launch sites' etc.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Resulting data was used to build ML models, using GridSearch to evaluate best parameters to use
  - Models used included Logistic Regression, SVC, Decision Trees and KNN

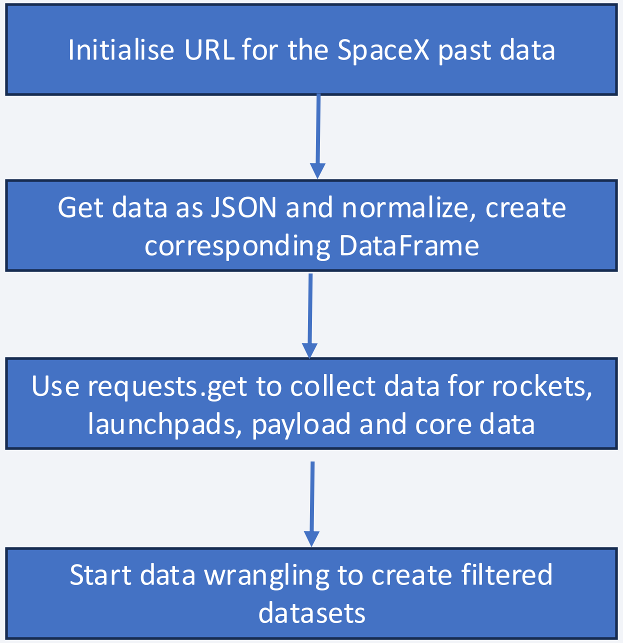
Data Collection

- Data sets were collected from SpaceX's API services (api.spacexdata.com/v4) and web scraped from the Falcon 9 Wiki pages (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)

- Following slides describe in more detail how each dataset was collected

Data Collection – SpaceX API

- SpaceX REST calls process:

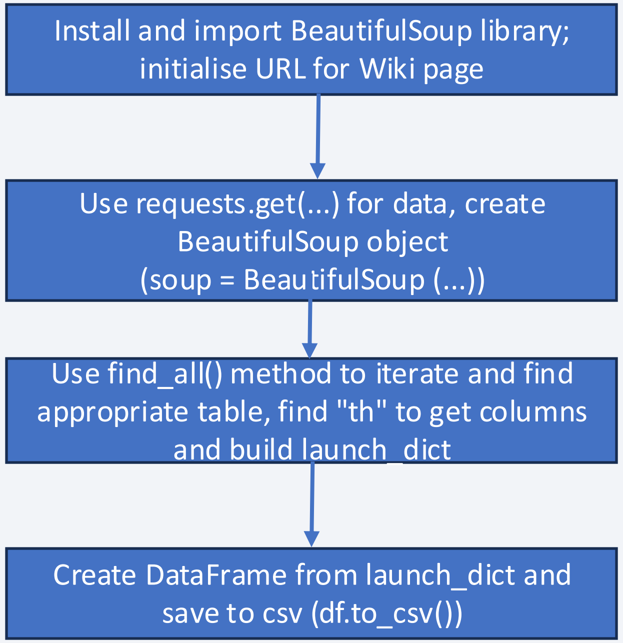

Initialise URL for the SpaceX past data
Get data as JSON and normalize, create corresponding DataFrame
Use requests.get to collect data for rockets, launchpads, payload and core data
Start data wrangling to create filtered datasets

- GitHub URL of the completed SpaceX API calls notebook:

https://github.com/timpresland/CapstoneProject/blob/main/jupyter-labs-spacex-data-collection-api-v2.ipynb

Data Collection - Scraping

- Web scraping process flowchart:



Install and import BeautifulSoup library; initialise URL for Wiki page

Use requests.get(...) for data, create BeautifulSoup object (soup = BeautifulSoup (...))

Use find_all() method to iterate and find appropriate table, find "th" to get columns and build launch_dict

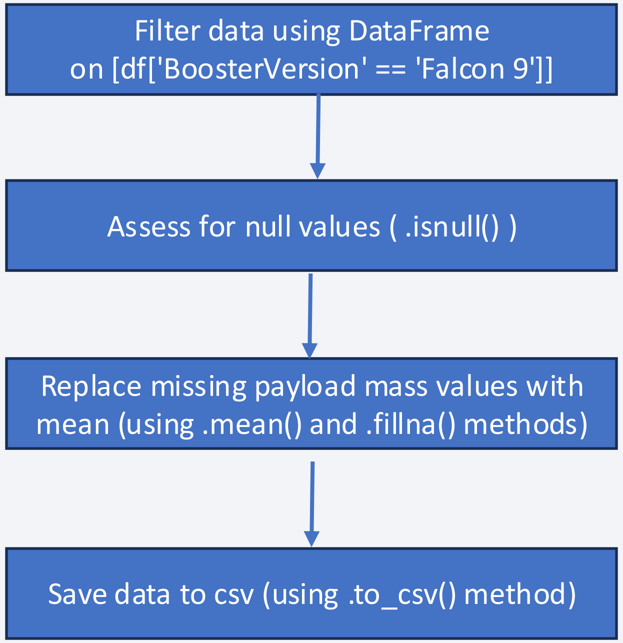
Create DataFrame from launch_dict and save to csv (df.to_csv())

- GitHub URL of the completed web scraping notebook:

https://github.com/timpresland/CapstoneProject/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

- Flowchart for data processing:



Filter data using DataFrame on [df['BoosterVersion' == 'Falcon 9']]

Assess for null values (.isnull())

Replace missing payload mass values with mean (using .mean() and .fillna() methods)

Save data to csv (using .to_csv() method)

- GitHub URL of data wrangling related notebooks:
- https://github.com/timpresland/CapstoneProject/blob/main/labs-jupyter-spacex-Data%20wrangling-v2.ipynb

EDA with Data Visualization

- The following charts were plotted. Scatter plots to analyse relationship between:

  - Payload Mass and Flight Numbers, using color to show Class (success / failure)
  - Launch Sites and Flight Numbers, using color to show Class (success / failure)
  - Payload Mass and Launch Sites, using color to show Class (success / failure)
  - Flight Numbers and Orbit, using color to show Class (success / failure)
  - Payload Mass and Orbit, using color to show Class (success / failure)

- Line plot to show success rates over time, between 2010 and 2020

- GitHub URL of completed EDA with data visualization notebook:

  https://github.com/timpresland/CapstoneProject/blob/main/jupyter-labs-eda-dataviz-v2.ipynb

EDA with SQL

- SQL queries performed to show:

  - list of launch sites
  - 5 records where launch sites begin with the string 'CCA'
  - total payload mass carried by boosters launched by NASA (CRS)
  - average payload mass carried by booster version F9 v1.1
  - date when the first succesful landing outcome in ground pad was achieved
  - names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - total number of successful and failure mission outcomes
  - booster_versions that have carried the maximum payload mass, using subquery
  - 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 between the date 2010-06-04 and 2017-03-20, in descending order

- GitHub URL of your completed EDA with SQL notebook:

  https://github.com/timpresland/CapstoneProject/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Map objects created and added to a folium map included:

  - circle object to create a reference location to NASA Johnson Space Centre
  - marker object (with icon showing text label), highlighting the location of the NASA Johnson Space Centre
  - child objects for each launch site, with circle and marker objects, as described above
  - marker clusters, to allow individual launch records to be drilled down (e.g. red = failure, green = success)
  - distance and polyline markers to highlight point to point distance to key references on the map (e.g. launch site to coast)

- GitHub URL of the interactive map with Folium map:

  https://github.com/timpresland/CapstoneProject/blob/main/lab-jupyter-launch-site-location-v2.ipynb

Build a Dashboard with Plotly Dash

- Plots / graphs and interactions added to the dashboard:

  - dropdown control to allow "ALL" or specific launch site filtering (comparison of success rates per launch site)
  - slider control to allow filtering of payload mass
  - pie charts to show % success of the selected filter (ALL sites or one specific launch site)
  - scatter plot to show success versus failure against selected site, payload, with color to show rocket booster version

- GitHub URL of completed Plotly Dash lab:

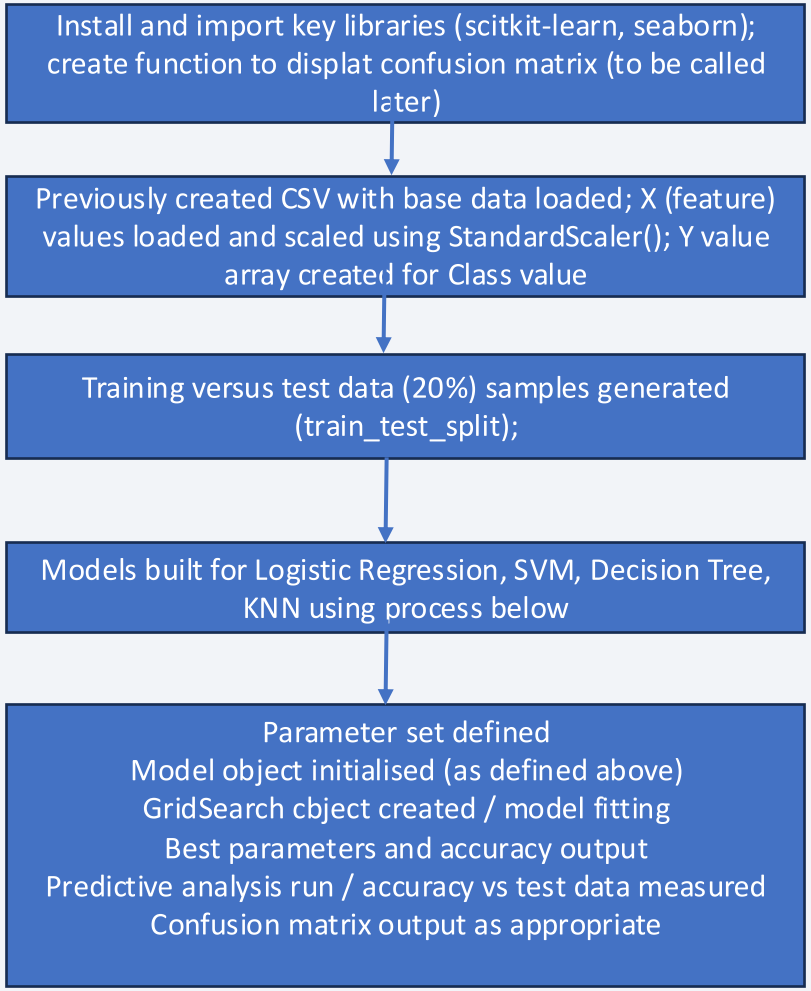
  https://github.com/timpresland/CapstoneProject/blob/main/Capstone%20-%20Dash%20Project.ipynb

Predictive Analysis (Classification)

- Methodology for build, evaluation, improvements and determining the best performing classification model shown in flowchart:

- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose:

  https://github.com/timpresland/CapstoneProject/blob/main/SpaceX-Machine-Learning-Prediction-Part-5-v1.ipynb



Install and import key libraries (scitkit-learn, seaborn); create function to displat confusion matrix (to be called later)

Previously created CSV with base data loaded; X (feature) values loaded and scaled using StandardScaler(); Y value array created for Class value

Training versus test data (20%) samples generated (train_test_split);

Models built for Logistic Regression, SVM, Decision Tree, KNN using process below

Parameter set defined
Model object initialised (as defined above)
GridSearch cbject created / model fitting
Best parameters and accuracy output
Predictive analysis run / accuracy vs test data measured
Confusion matrix output as appropriate

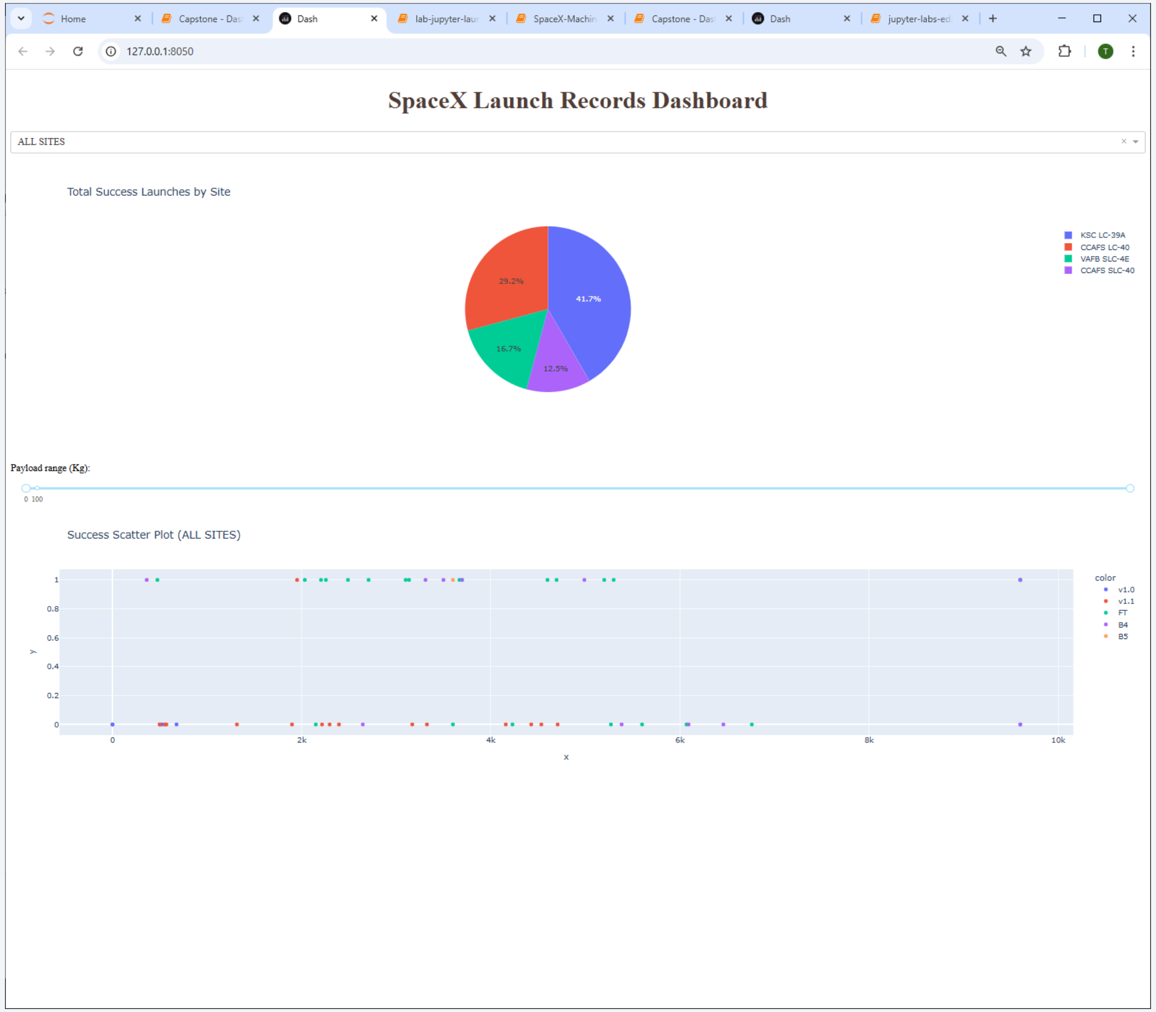
Results

- Exploratory data analysis results:

  - total 90 records, 18 test samples
  - as flight number increased, success rate improved
  - payloads of 2000-7000Kg had more success
  - HEO, GEO, ES-L1, SSO orbits had the highest success rate
  - success rates increased from 2013-2017

- Predictive analysis results:

  - accuracy scores for LR, KNN and SVC models very similar at around 85%, with test scores around 83%
  - Decision Tree best model with accuracy 87.5%, test score 94%



SpaceX Launch Records Dashboard

ALL SITES

Total Success Launches by Site

Payload range (Kg):

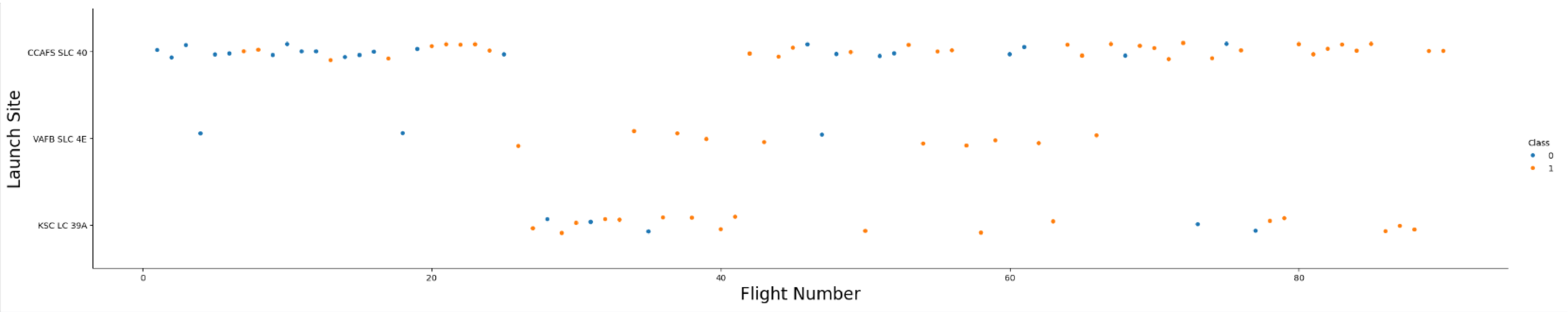
Success Scatter Plot (ALL SITES)

Section 2

Insights drawn from EDA

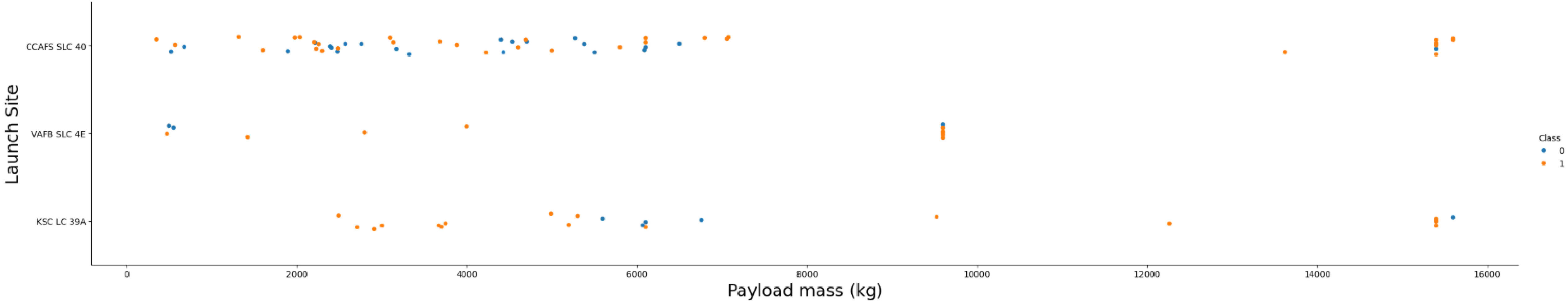
Flight Number vs. Launch Site

- Scatter plot of Flight Number vs. Launch Site
- CCAFS SLC40 shows most commonly used site, from earliest to latest flight numbers
- All sites show an improving success rate with time



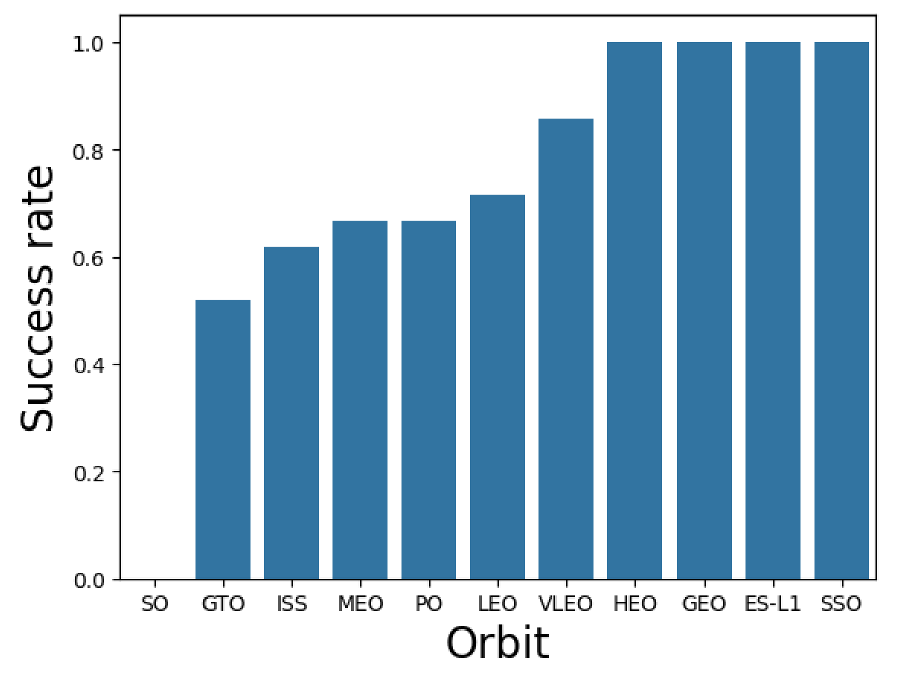
Payload vs. Launch Site

- Scatter plot of Payload vs. Launch Site
- Most successful missions are with payload mass between 2000 and 6000 Kg
- Success rates good with larger payloads but fewer occurences



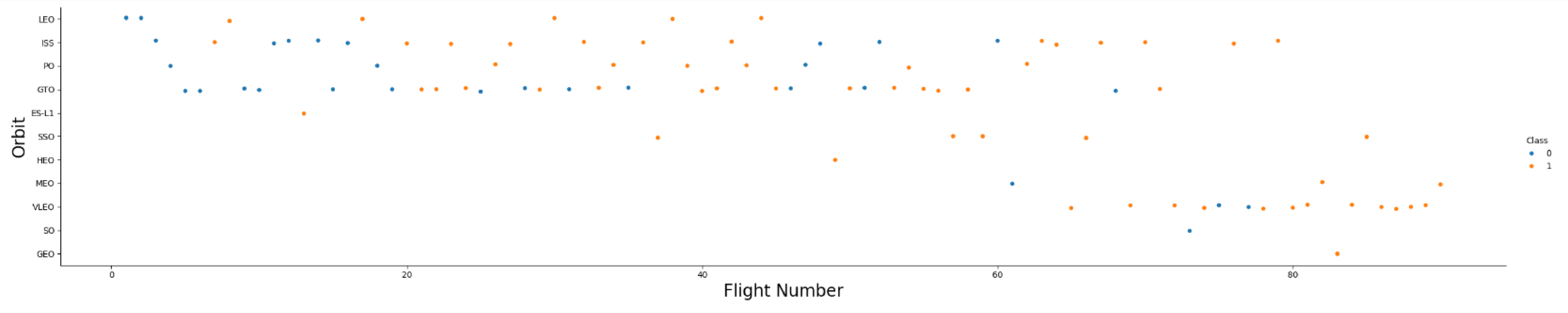
Success Rate vs. Orbit Type

- Bar chart for the success rate of each orbit type
- HEO, GEO, ES-L1 and SSO 100% success rate
- SO and GTO orbits had poorest success rate



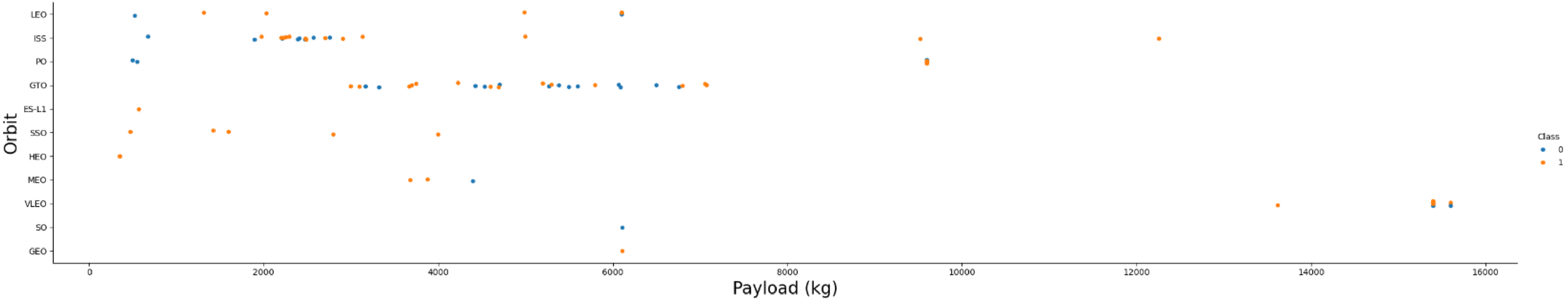
Flight Number vs. Orbit Type

- Scatter point of Flight number vs. Orbit type
- Latest flights were very successful, particularly using the VLEO orbit
- Earliest flights showed poor success rates



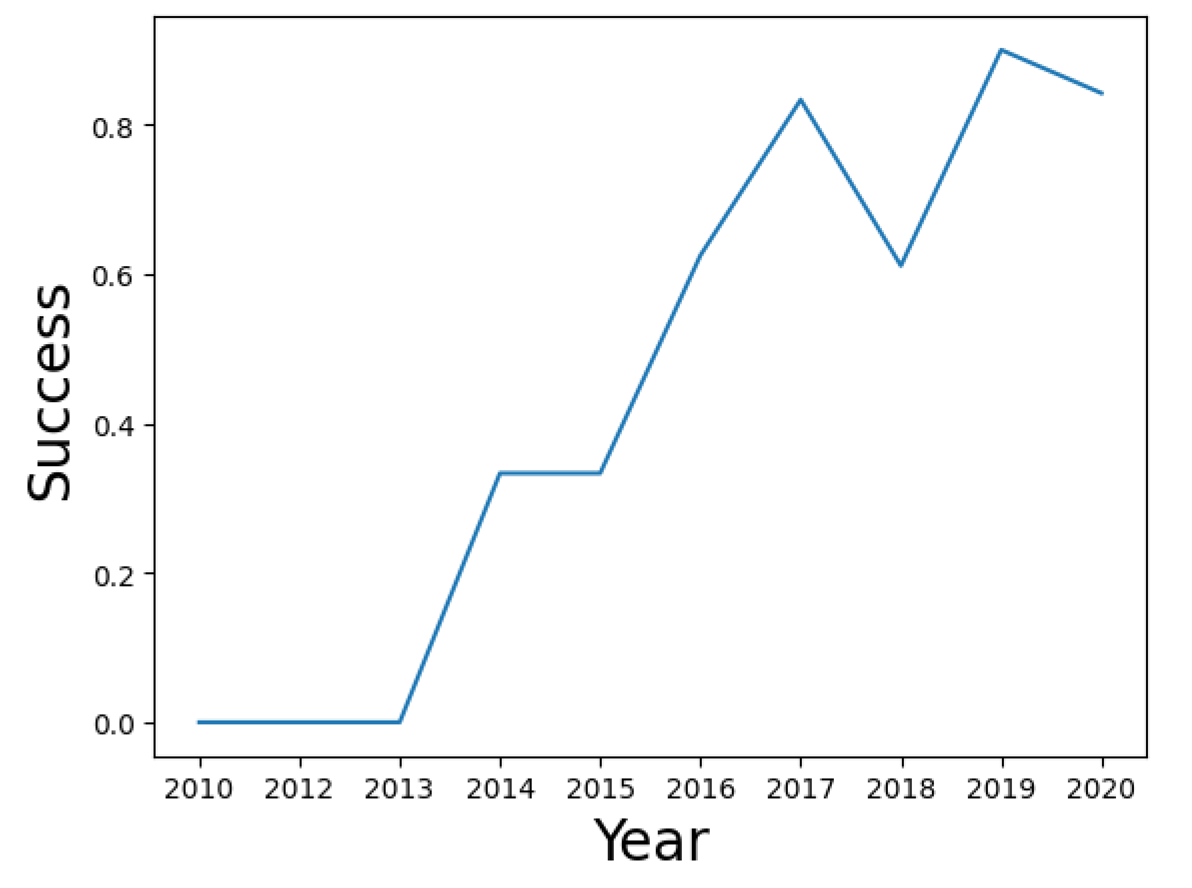
Payload vs. Orbit Type

- Scatter point of payload vs. orbit type
- Most successes with payload between 2000 and 6000 Kg
- High failure rates evident on the ISS and GTO orbits, with little correlation against payload size



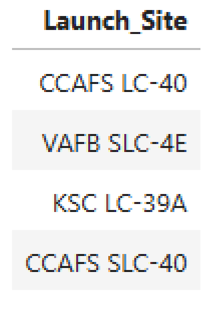
Launch Success Yearly Trend

- Line chart of yearly average success rate
- Poor success rates between 2011 and 2013
- Rapidly increasing success rates from 2013 to 2017
- Success rates appear to be plateauing after 2017



All Launch Site Names

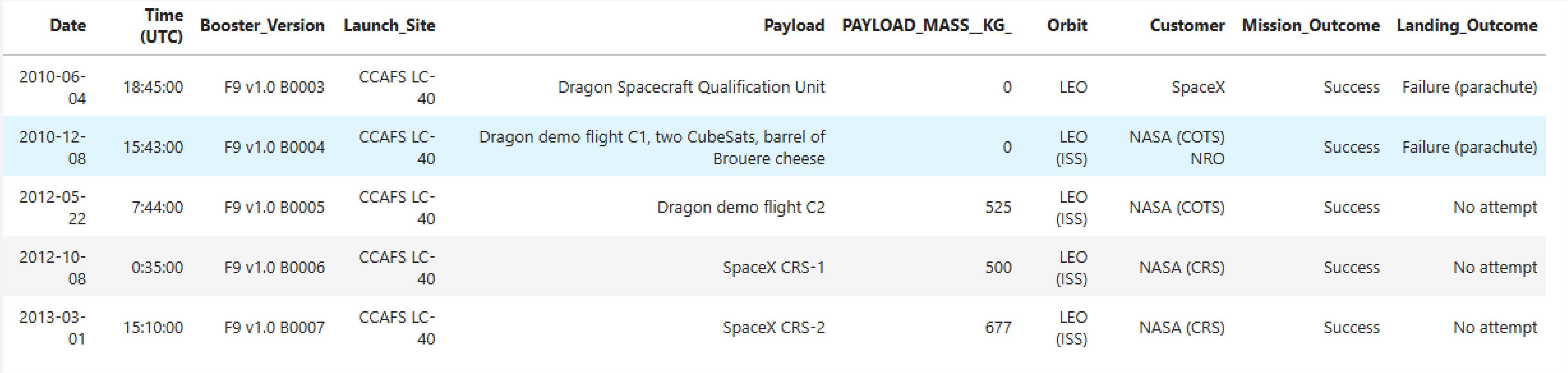
- Unique launch sites: CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40
- Distinct select query using "Launch_Site" column used to provide unique list of sites



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

Launch Site Names Begin with 'CCA'

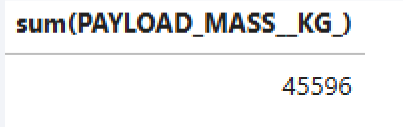
- 5 records where launch sites begin with `CCA`
- Select query used with the "LIKE" operator to find site names ("Launch_Site" column) prefixed with "CCA"



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	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

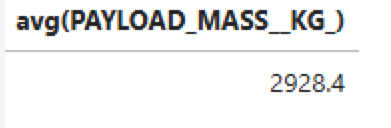
- Total payload carried by boosters from NASA = 45,596Kg
- "sum" operator used in select query on "PAYLOAD_MASS_KG" column, where clause to filter on NASA site only



sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

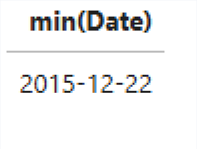
- Average payload mass carried by booster version F9 v1.1 = 2928.4Kg
- "avg" operator used in query, filtered on column "Booster_Version" where equal to "F9 v1.1"



avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

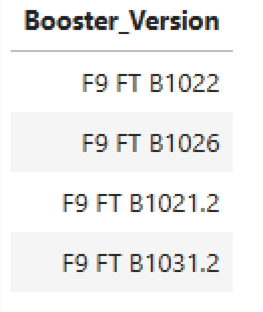
- Dates of the first successful landing outcome on ground pad = 22 December 2015
- "min" operator used on "Landing_Outcome" column; "LIKE" clause used to filter on "%Success (ground pad)%" criteria



min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

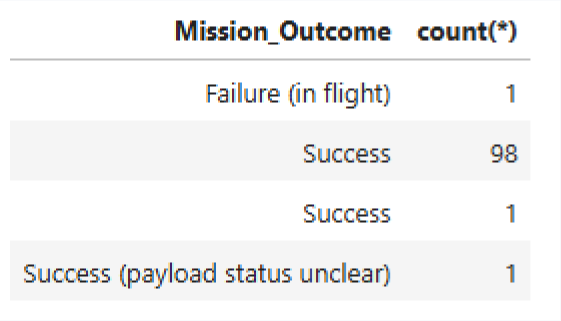
- Names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 (F9 FT B1022, F9 FT B1026, F9 FT B1021.2, F9 FT B1031.2)
- Greater and less than operators used (<, >) to filter result using "PAYLOAD_MASS_KG" column
- "Landing_Outcome" column filtered using LIKE criteria for "%Success (drone ship)%"



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

Total Number of Successful and Failure Mission Outcomes

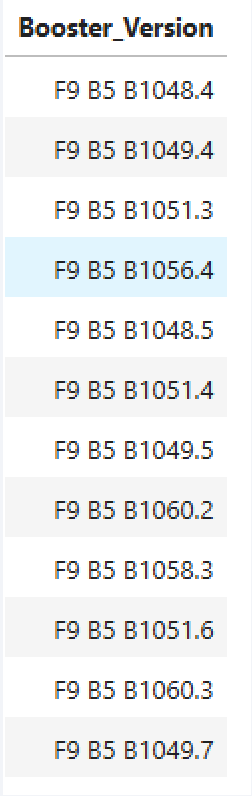
- Total number of successful (100) and failure (1) mission outcomes
- Distinct "Mission_Outcome" values used to group sum of records for each



Mission_Outcome	count(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

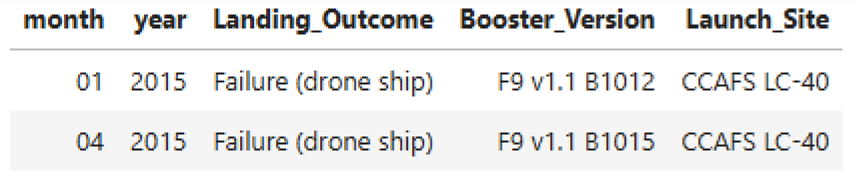
- Names of the booster which have carried the maximum payload mass = see below right!
- "max" operator used to create subquery to find maximum payload value
- Max value used as filter in select query to find all booster versions corresponding to thay payload mass



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

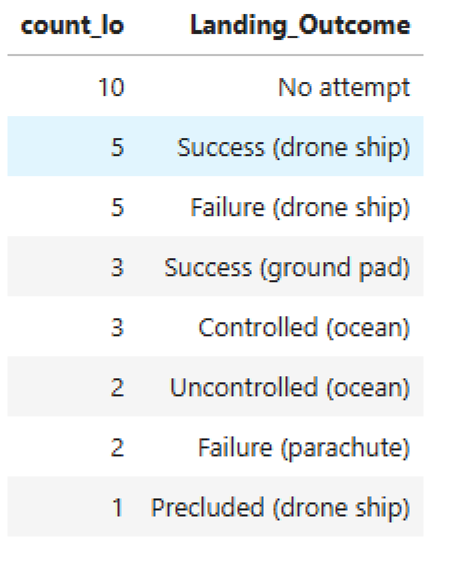
- Failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Select query used with criteria for year (2015) and "LIKE" clause to isolate failure on drone ships
- Substring functions used to separate date into month and year values



month	year	Landing_Outcome	Booster_Version	Launch_Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Rankings for 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
- Counts of "Landing_Outcome" column values used to group each type
- Date filtered using "between" operator
- Descending operator ("DESC") used to rank highest to lowest



count_lo	Landing_Outcome
10	No attempt
5	Success (drone ship)
5	Failure (drone ship)
3	Success (ground pad)
3	Controlled (ocean)
2	Uncontrolled (ocean)
2	Failure (parachute)
1	Precluded (drone ship)

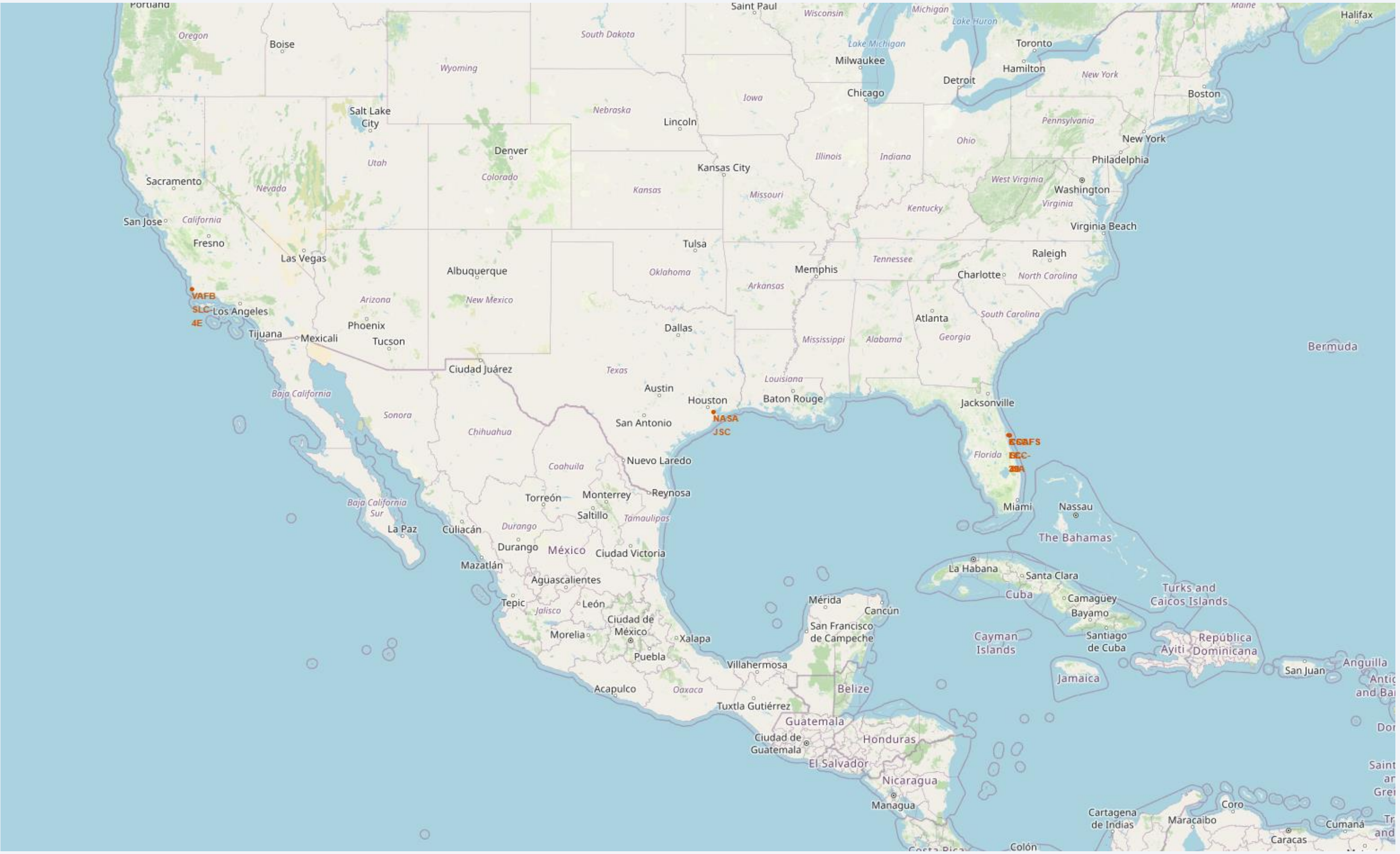
Section 3

Launch Sites Proximities Analysis

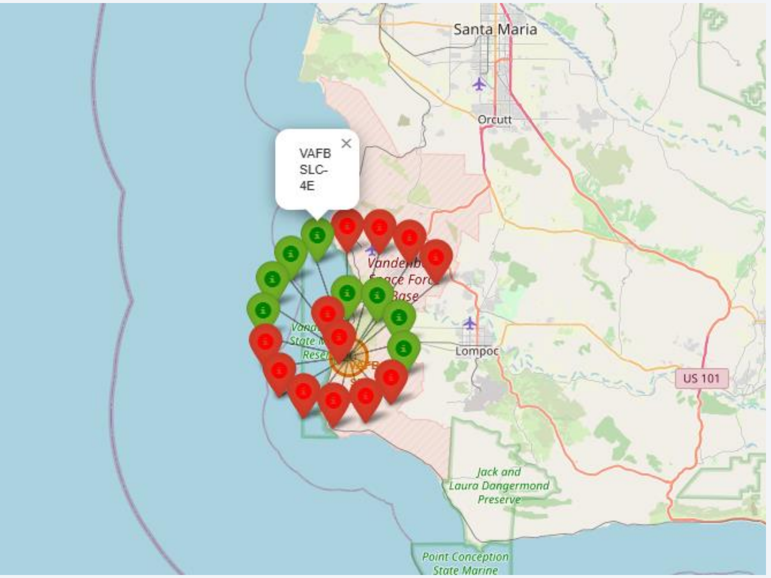
Folium Map Screenshot – Launch Site Locations

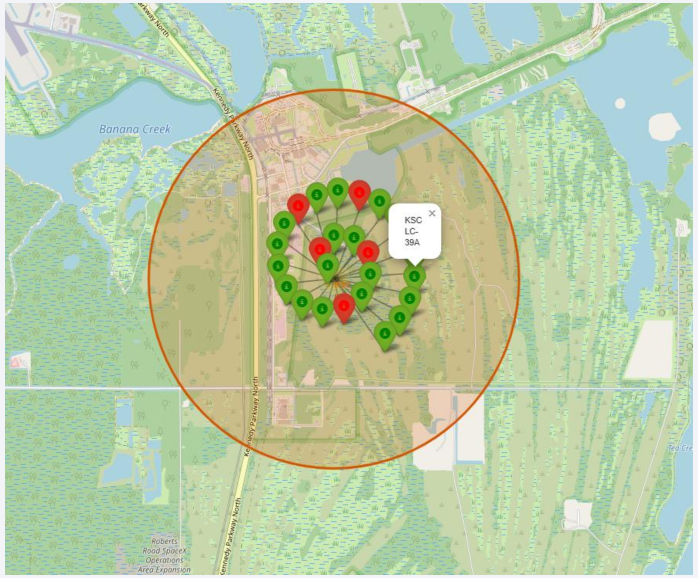
- Key findings:

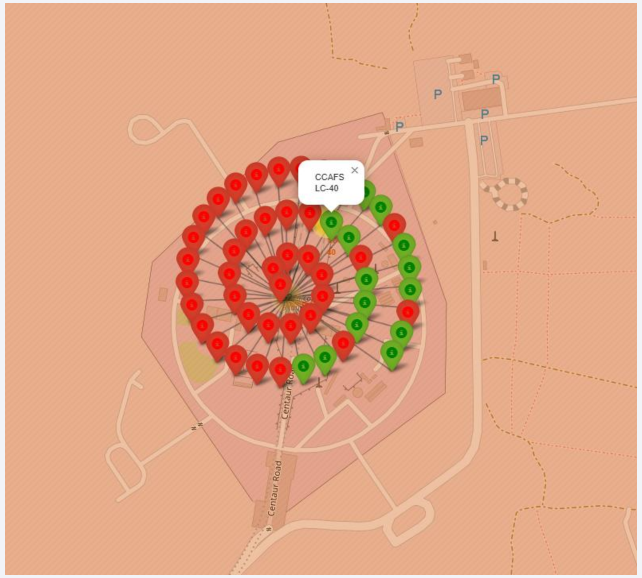
  - launch sites tend to be on the coast
  - sites are not in city areas
  - sites are close to key infrastructure, such as roads, railways and airports



Folium Map Screenshot – Launch Outcomes


Santa Maria
Orcutt
VAFB SLC-4E
Vandenb
Space Forc
Base
Lompoc
US 101
Jack and Laura Dangermond Preserve
Point Conception State Marine





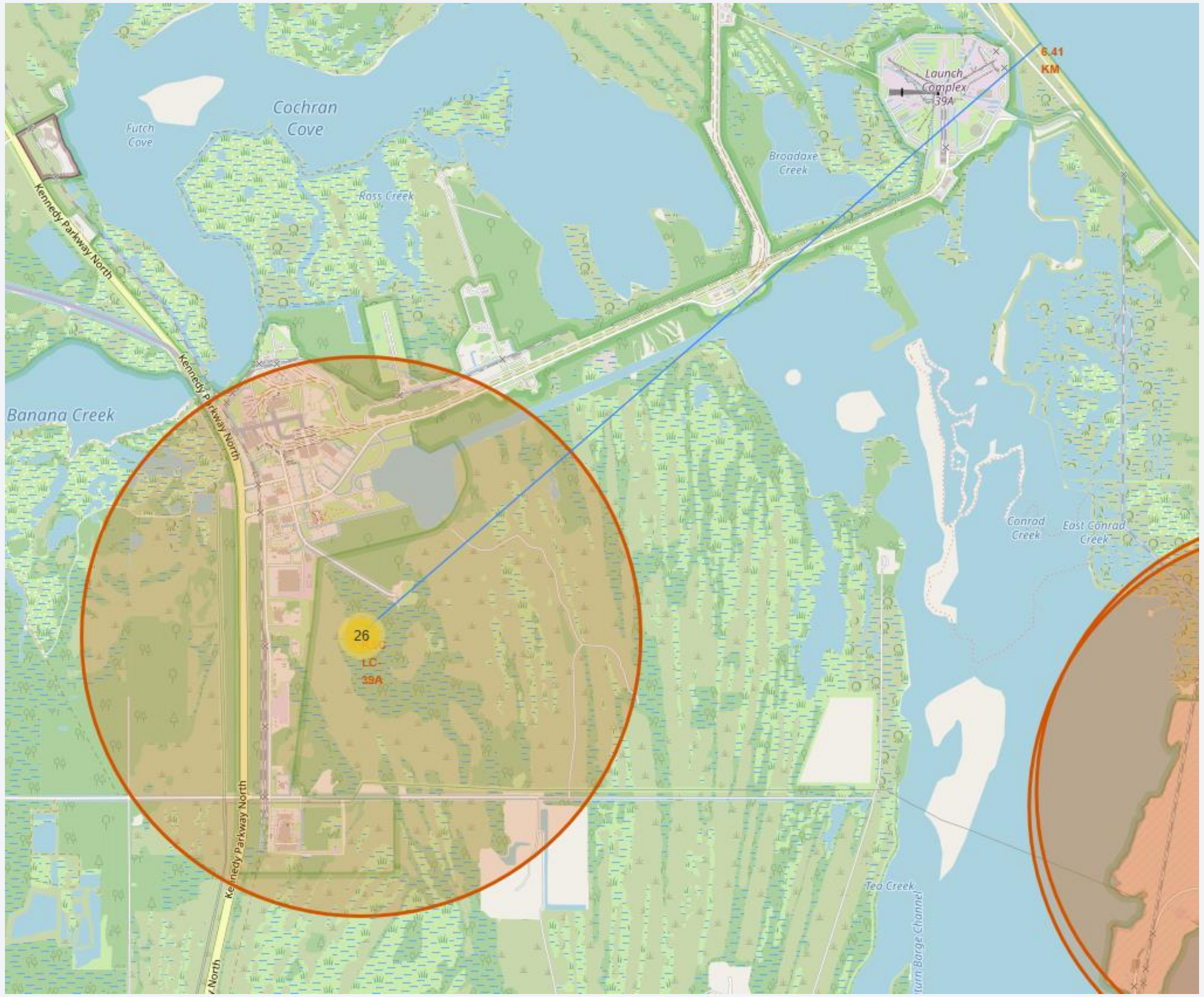
- Key findings:

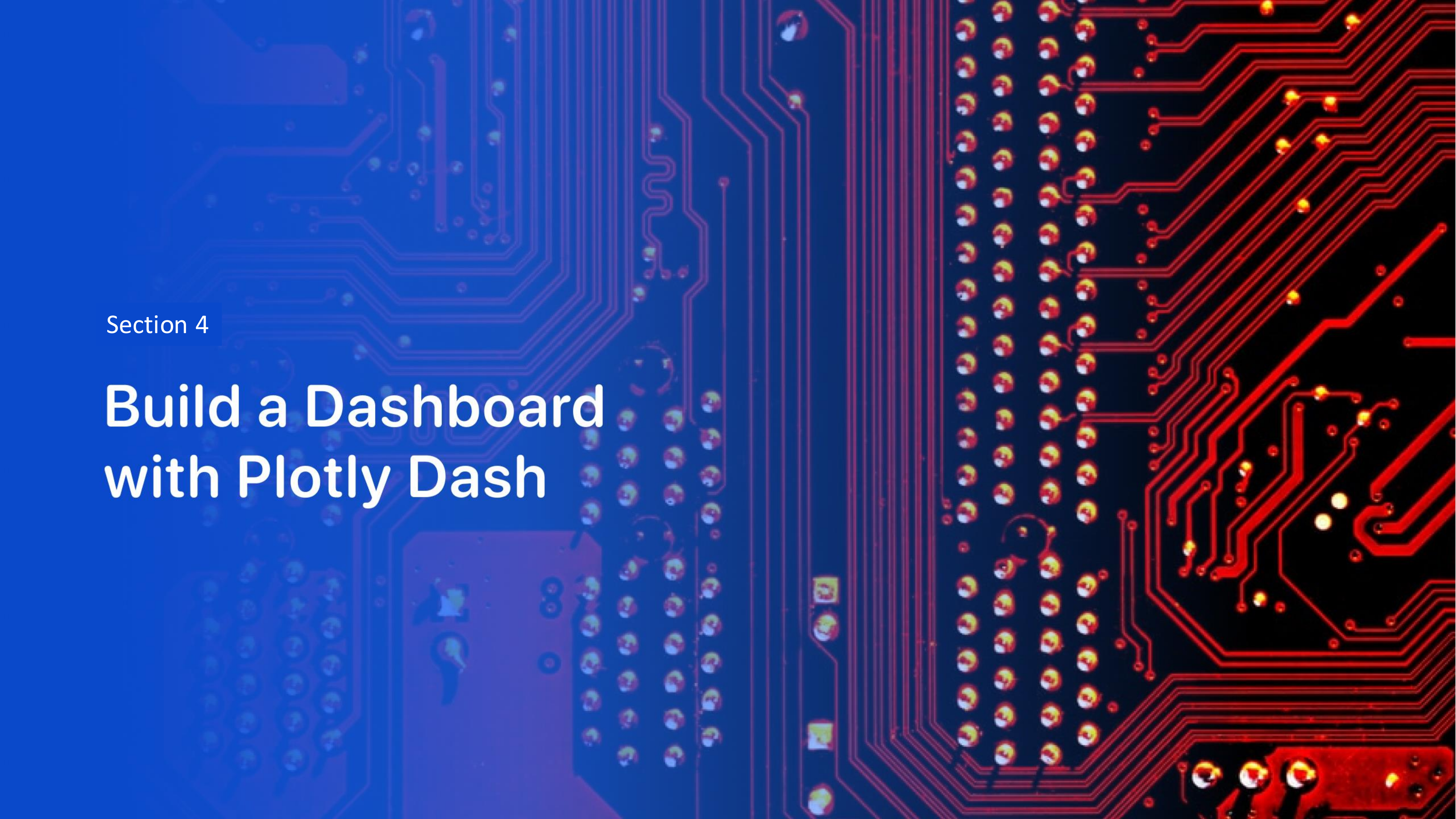
  - CCAFS LC-40 site has lower success rate than other sites
  - VAFB site has mixed performance
  - KSC LC-39A clearly has better success rate

Folium Map Screenshot – Coastal Proximity

- Key findings:

  - launch site only 6.4Km from coast
  - close to key infrastructure, such as highway and railway

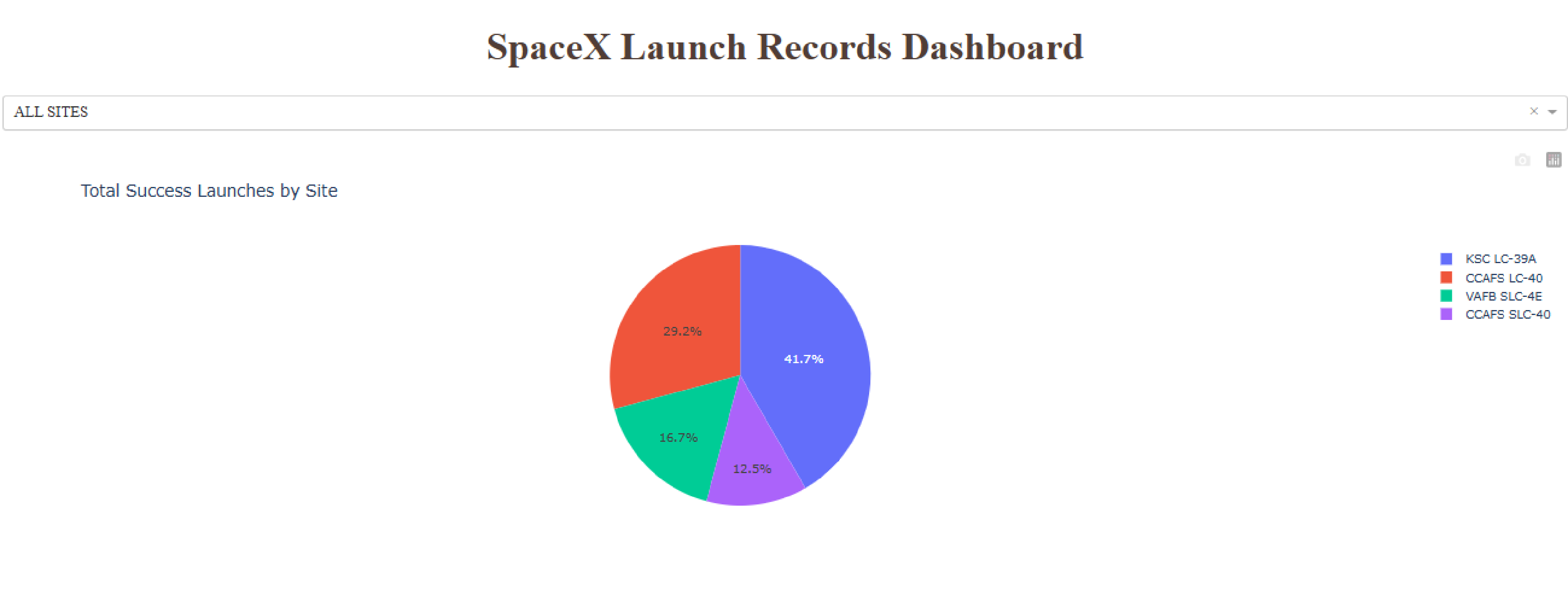




Section 4

Build a Dashboard with Plotly Dash

Dashboard Screenshot – Launch Success (All Sites)



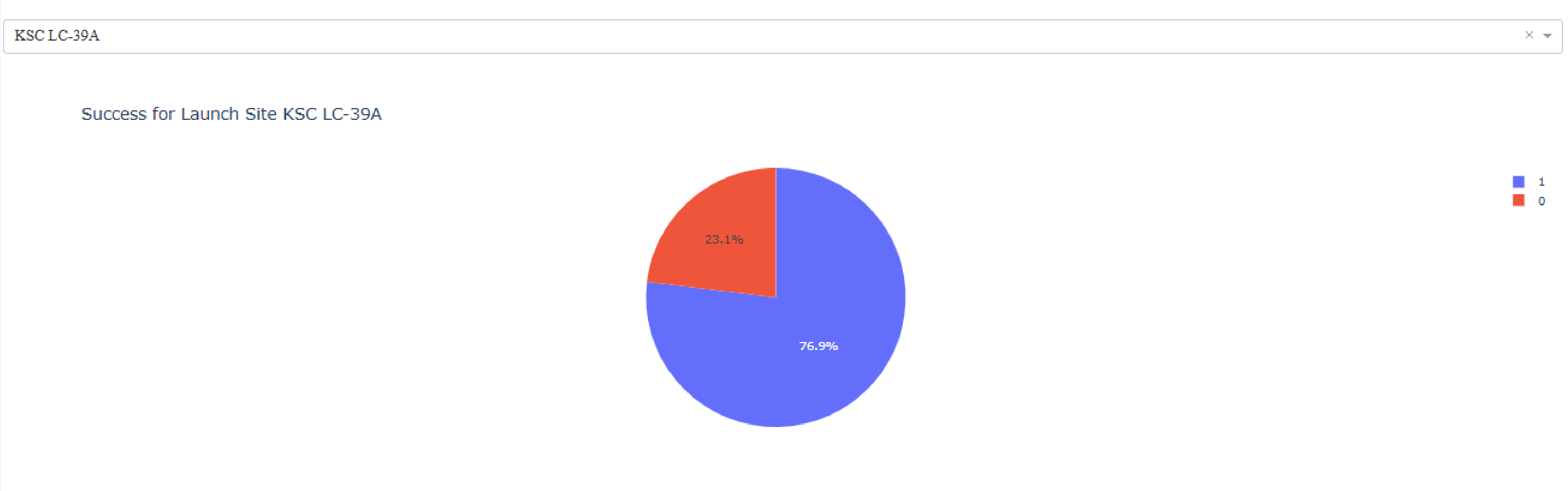
SpaceX Launch Records Dashboard

ALL SITES

- Key observations:

  - KSC LC-39A has the highest success rate (41.7% of all successful launches took place there)
  - VAFB SLC-4E and CCAFS SLC-40 poorest (<20%)

Dashboard Screenshot – Most Successful Launch Site



KSC LC-39A

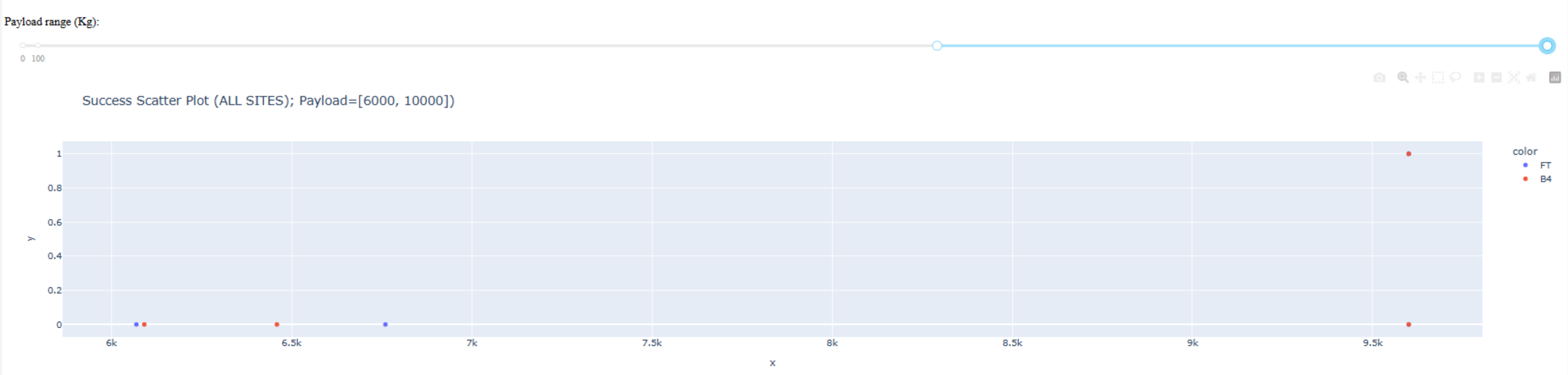
Success for Launch Site KSC LC-39A

- Key observations:

  - KSC LC-39A has a high success rates (~77% of its launches succeeded)

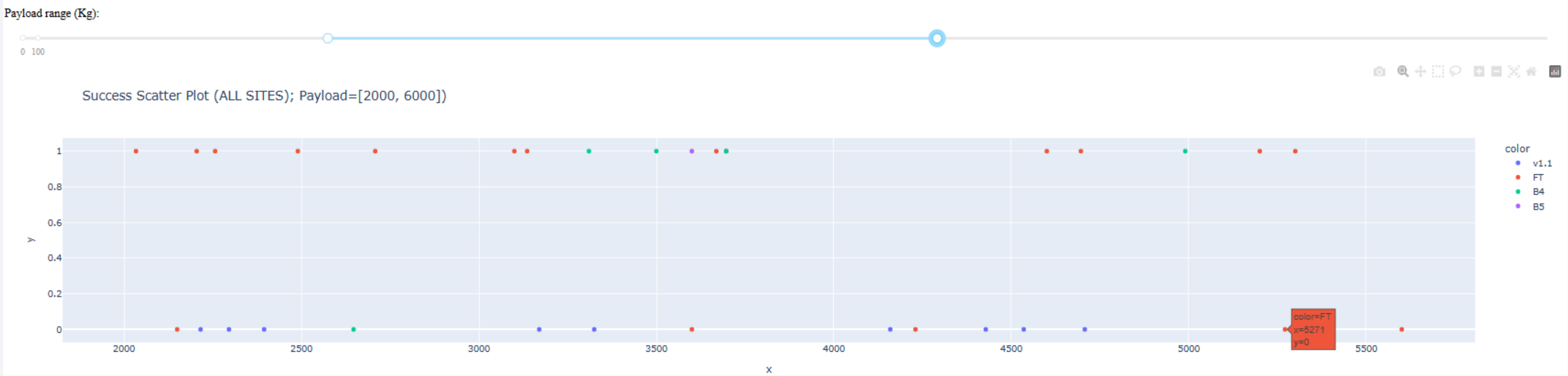
Dashboard Screenshot – Payload Ranges (All Sites)

- Top scatter shows that higher payload ranges have poor success rates
- Bottom scatter shows much better success rate at lower payload range (in this case 2,000 to 6,000 Kg)



Payload range (Kg):

Success Scatter Plot (ALL SITES); Payload=[6000, 10000])



Payload range (Kg):

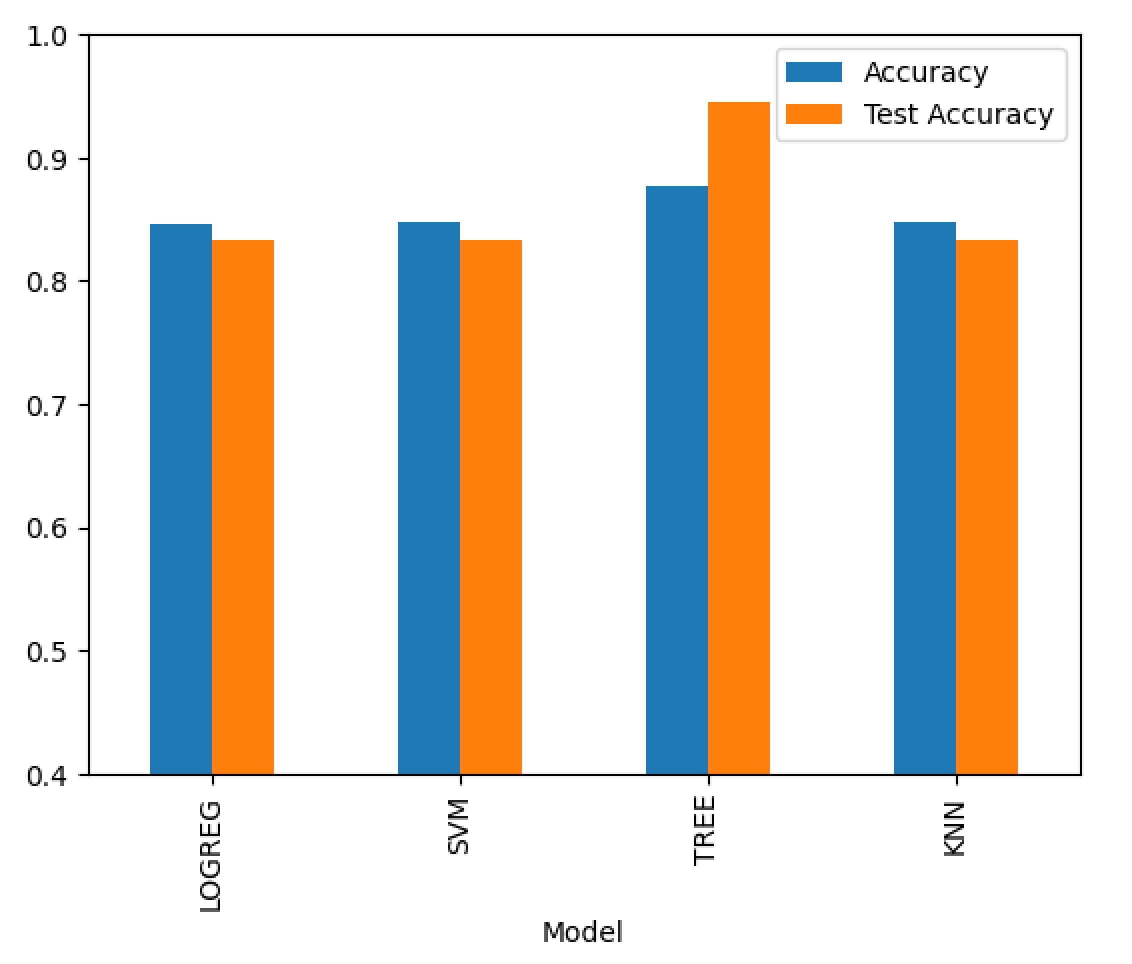
Success Scatter Plot (ALL SITES); Payload=[2000, 6000])

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Visualisations using bar chart to show model accuracies:

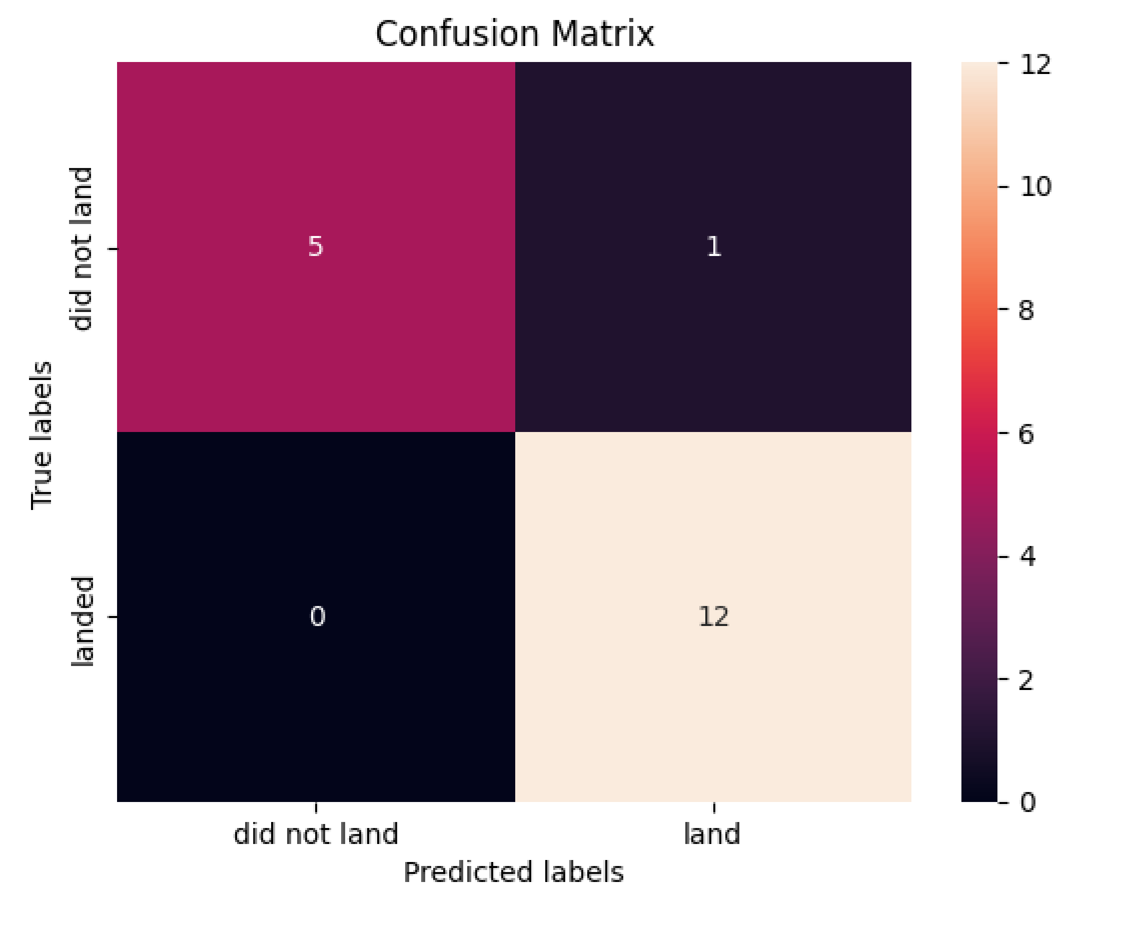


- Decision Tree has the best accuracy with >90% on test data
- LogReg, SVM and KNN all similar with accuracy around 85%

Confusion Matrix

- Confusion matrix of best performing model

- Results in 12 true positives, 5 true negatives, zero false negatives and only1 false positive



Conclusions

- Decision Tree classification proved to be the most successful model
- The tree also performed better than expected against the test data
- The logistic regression, SVM and KNN models were all moderately good and similar in results, but not as good as the decision tree

Appendix

- URL to full GitHub repository:

  https://github.com/timpresland/CapstoneProject

Thank you!