IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Duy Anh Nguyen

8 December 2021

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
- Data Collection and Data Wrangling
- EDA with Visualization and SQL
- Interactive visual analytics with Folium and Plotly Dash
- Predictive Analysis
- Summary of all results
- EDA with visualizations
- EDA with SQL
- Interactive map with Folium
- Plotly Dash dashboard
- Predictive Analysis (Classification)

Introduction

- Project background and context:

SpaceX advertises Falcon 9 rocket launches on its website with a cost of $62 million, while the cost for other providers can be as high as $165. The reusability of the first stage is the main reason behind this big difference. Hence, if we can predict whether the first stage will land, we can determine the cost of launch.

- Problems you want to find answers
- What variables determine the success of a launch?
- What Machine Learning model can be used to reliably predict the landing outcome of a launch?

Section 1

Methodology

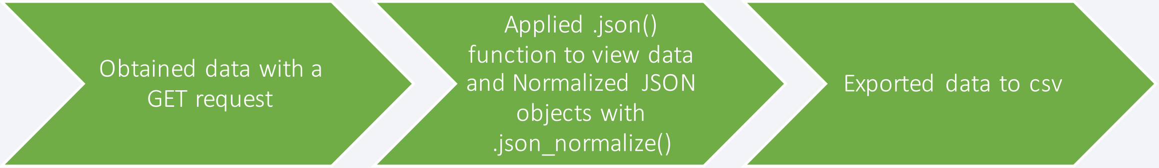
Methodology

- Data collection methodology:
  - SpaceX REST API
  - Web Scrapping from Wikipedia
- Perform data wrangling
  - Converted landing outcomes to 0 for failure and 1 for success
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Preprocessing, train/test split, train and perform models using GridSearchCV and determine the best model with the highest accuracy

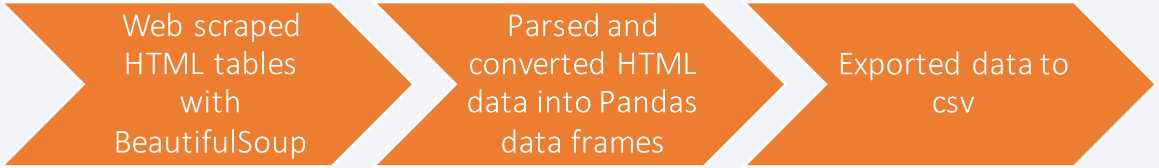
Data Collection

- Describe how data sets were collected:

- SpaceX REST API endpoint used: *api.spacexdata.com/v4/launches/past*
- Applied .json() function to view data and Normalized JSON objects with .json_normalize()
- Exported data to csv

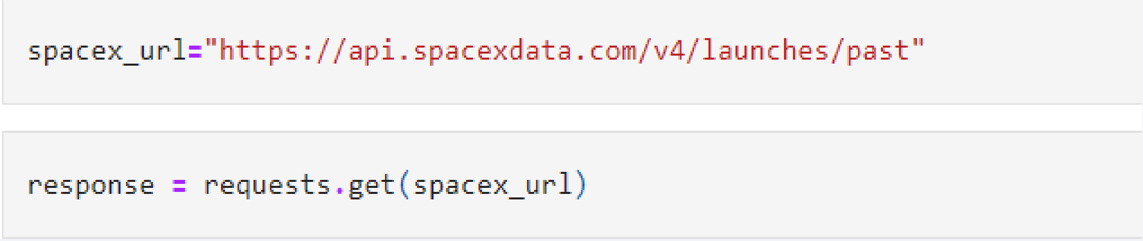

Obtained data with a GET request
Applied .json() function to view data and Normalized JSON objects with .json_normalize()
Exported data to csv

- Used BeautifulSoup library to web-scrape HTML tables
- Parsed data and converted these data into Pandas data frames
- Exported data to csv


Web scraped HTML tables with BeautifulSoup
Parsed and converted HTML data into Pandas data frames
Exported data to csv

Data Collection – SpaceX API

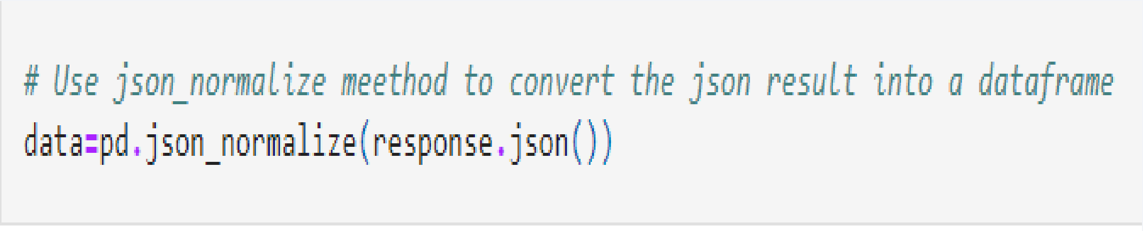
- Obtained data with a GET request



```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

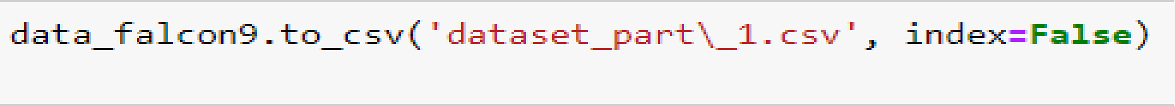
```
response = requests.get(spacex_url)
```

- Applied .json() function to view data and Normalized JSON objects with .json_normalize()



```
# Use json_normalize meethod to convert the json result into a dataframe
data=pd.json_normalize(response.json())
```

- Exported data to csv

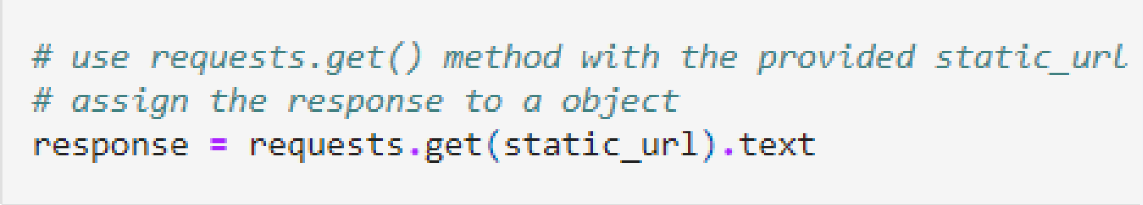


```
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

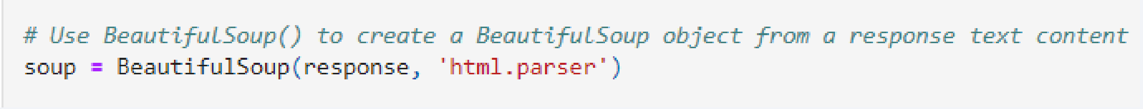
- GitHub notebook

Data Collection – Web Scraping from Wikipedia

- Web scraped HTML tables with BeautifulSoup

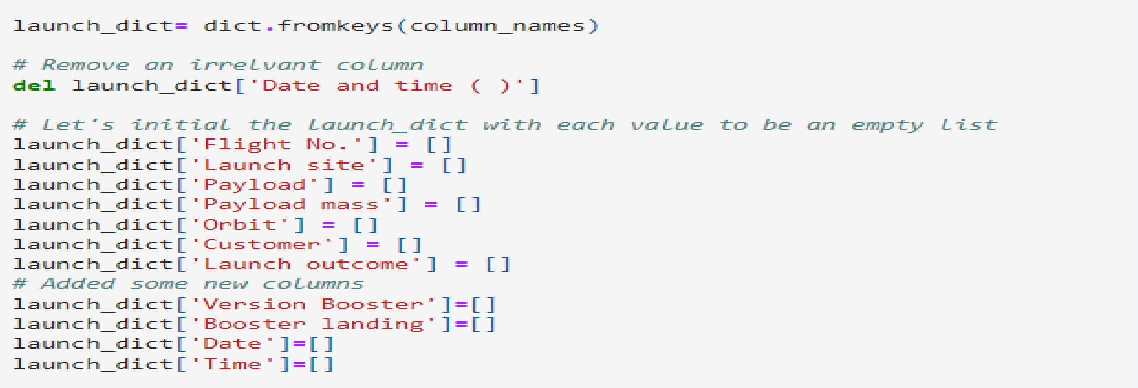


```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url).text
```

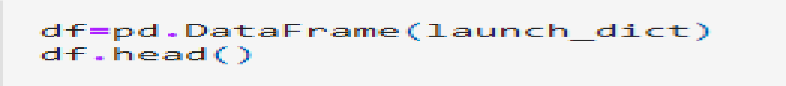


```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response, 'html.parser')
```

- Parsed data and converted these data into Pandas data frames

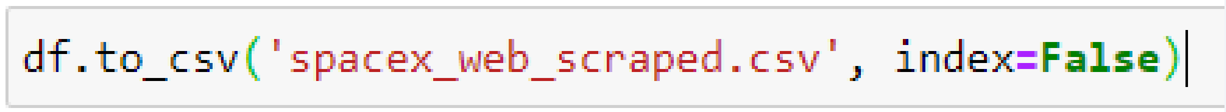


```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each value to be an empty list
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```



```
df=pd.DataFrame(launch_dict)
df.head()
```

- Exported data to csv

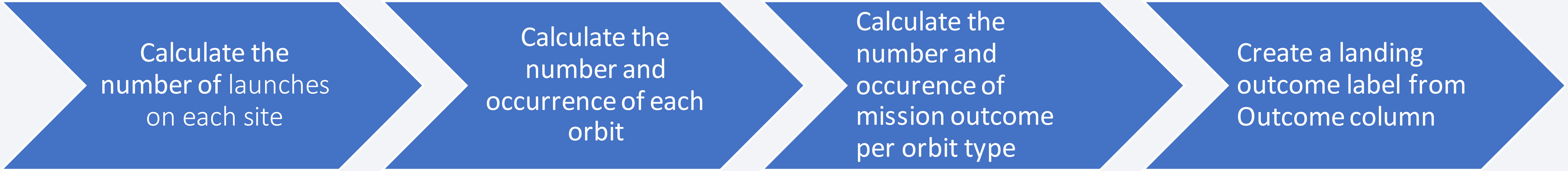


```
df.to_csv('spacex_web_scraped.csv', index=False)
```

- GitHub notebook

Data Wrangling

- Assigned 'success' landing outcome to 1, and 'failure' landing outcome to 0
- Key processes



Calculate the number of launches on each site

Calculate the number and occurrence of each orbit

Calculate the number and occurence of mission outcome per orbit type

Create a landing outcome label from Outcome column

- GitHub notebook

EDA with Data Visualization

Scatterplots:

- Flight Number vs Payload Mass
- Flight Number vs Launch Site
- Payload Mass vs Launch Site
- Orbit vs Flight Number
- Payload vs Orbit Type
- Orbit vs Payload Mass

Scatterplots show the relationship, or correlation, between 2 variables

Bar graphs:

- Mean Class vs Orbit

Bar graphs are used to how each type of Orbit differs from each other in terms of their mean class

Line graphs:

- Success Rate vs Year

Line graphs are useful in illustrating changes in data over time (trends) and making predictions about future outcomes

[GitHub notebook]

EDA with SQL

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

GitHub notebook

Build an Interactive Map with Folium

- Circles were added around each launch site with labels showing the name of each launch site.
- Green markers were added to indicate successful launches and Red markers were added to indicate unsuccessful launches. These color-labeled markers in a marker cluster help us identify which launch sites have relatively high success rate.
- Lines help measure the distance between a particular launch site and various locations on the map

- GitHub notebook

Build a Dashboard with Plotly Dash

- A pie chart was added to display the relative proportion of successful launches for each launch site
- A scatter plot was added to show the relationship between Launch Outcome and Payload Mass for different Booster versions. Scatter plots are an excellent visualization to show the relationship between 3 variables, being Launch Outcome, Payload Mass and Booster versions

- GitHub notebook

Predictive Analysis (Classification)

- Loaded the data and applied data standardization and transformation
- Train/Test split
- Set parameters for GridSearchCV and fit the models
- Evaluated the accuracy of Logistic Regression, SVM, Decision Tree and KN and selected the model with the best accuracy

Loaded, standardized and transformed data

Train/ Test split

Set parameters for GridSearchCV and fit the models

Evaluated the models and selected the best model

- GitHub notebook

Results

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

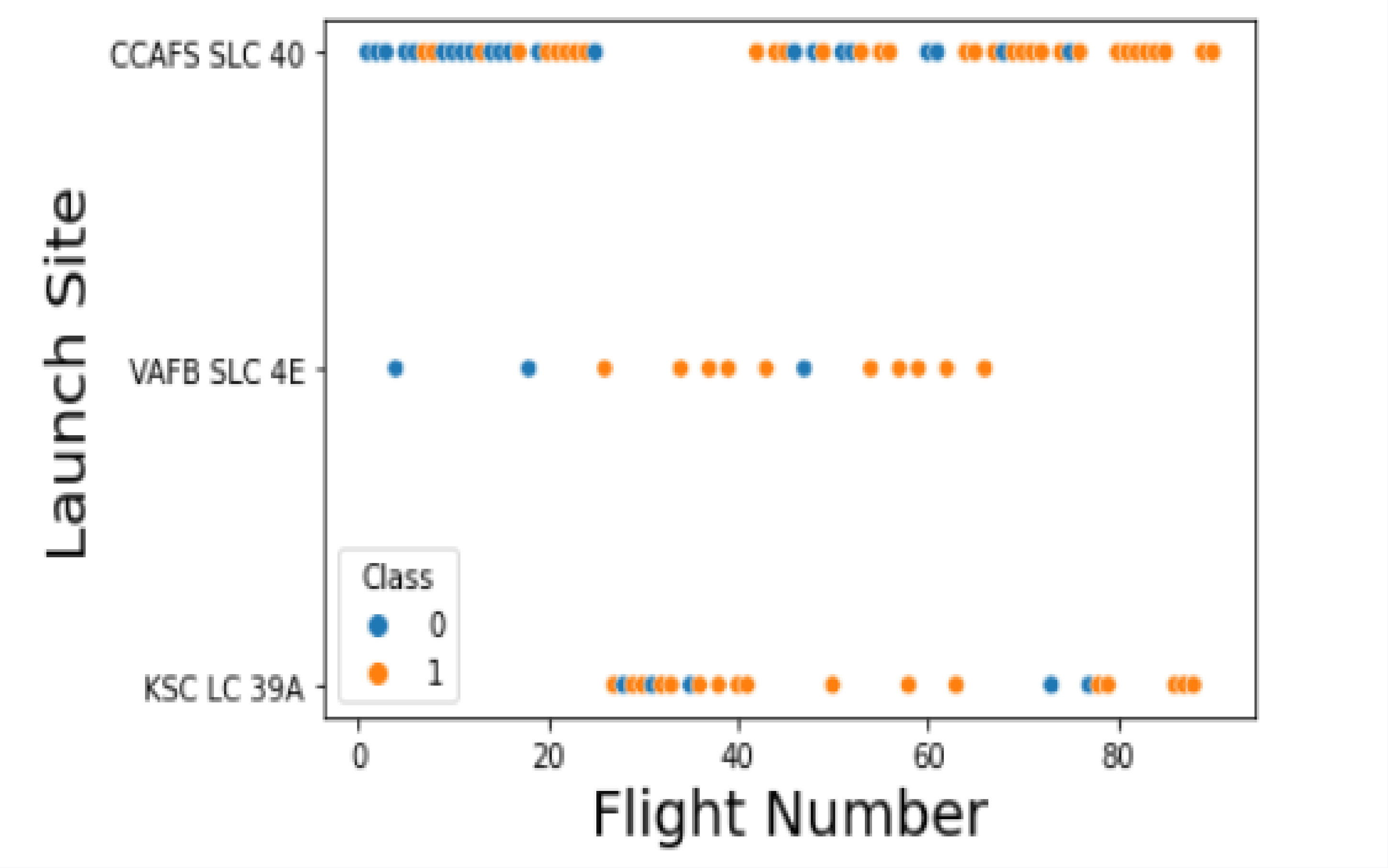
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

- Blue dots represent unsuccessful launches, while orange dots represent successful launches

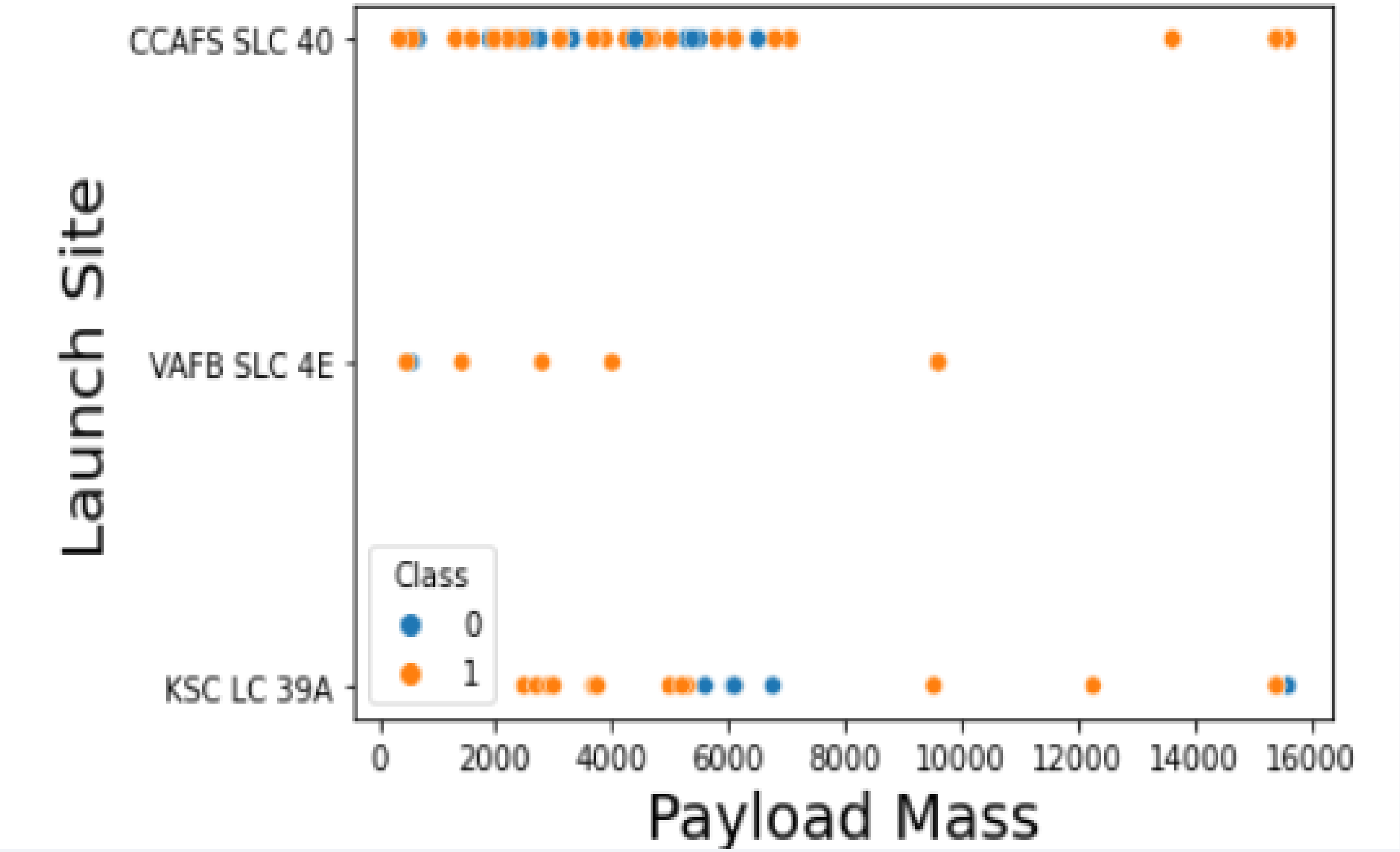
- The higher the number of flights, the better the success rate of each launch site



Payload vs. Launch Site

- Blue dots represent unsuccessful launches, while orange dots represent successful launches

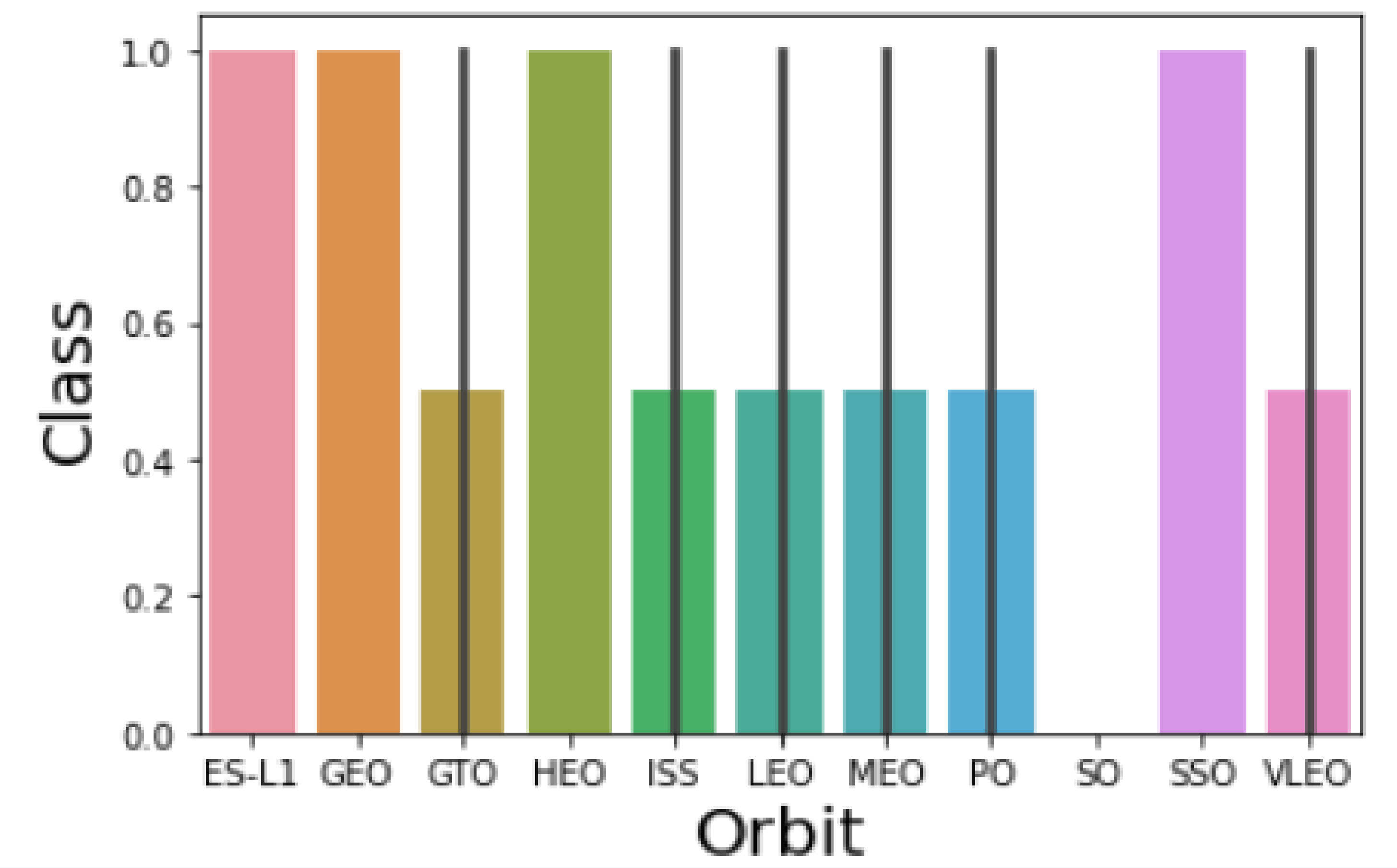
- No clear pattern can be observed from this visualization



Success Rate vs. Orbit Type

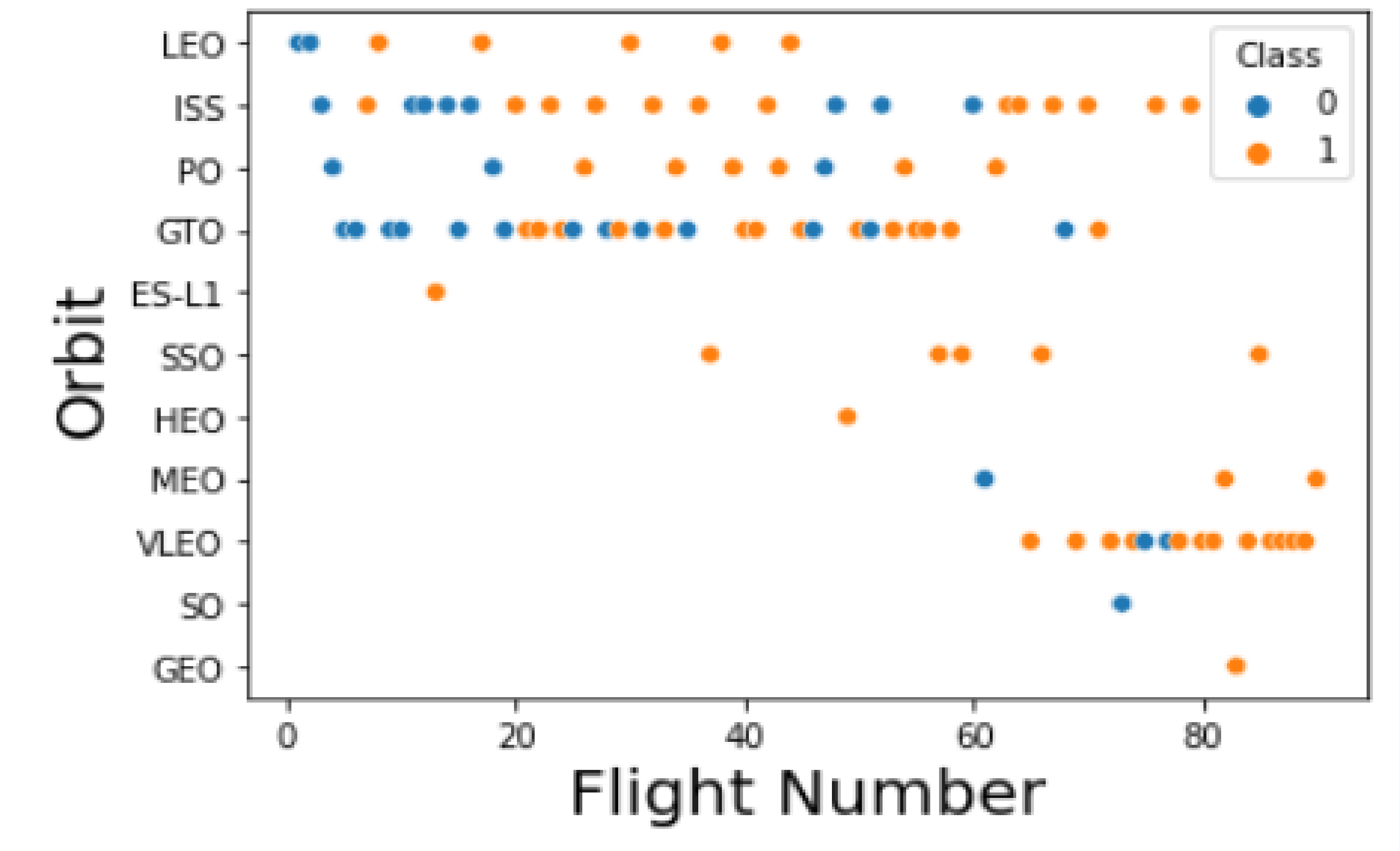
- Orbit ES-L1, GEO and HEO have the best success rates

- No success launch was recorded for SO orbit



Flight Number vs. Orbit Type

- Blue dots represent unsuccessful launches, while orange dots represent successful launches
- Launch outcome seems to be correlated to the number of flight for LEO orbit
- SSO has no unsuccessful launches

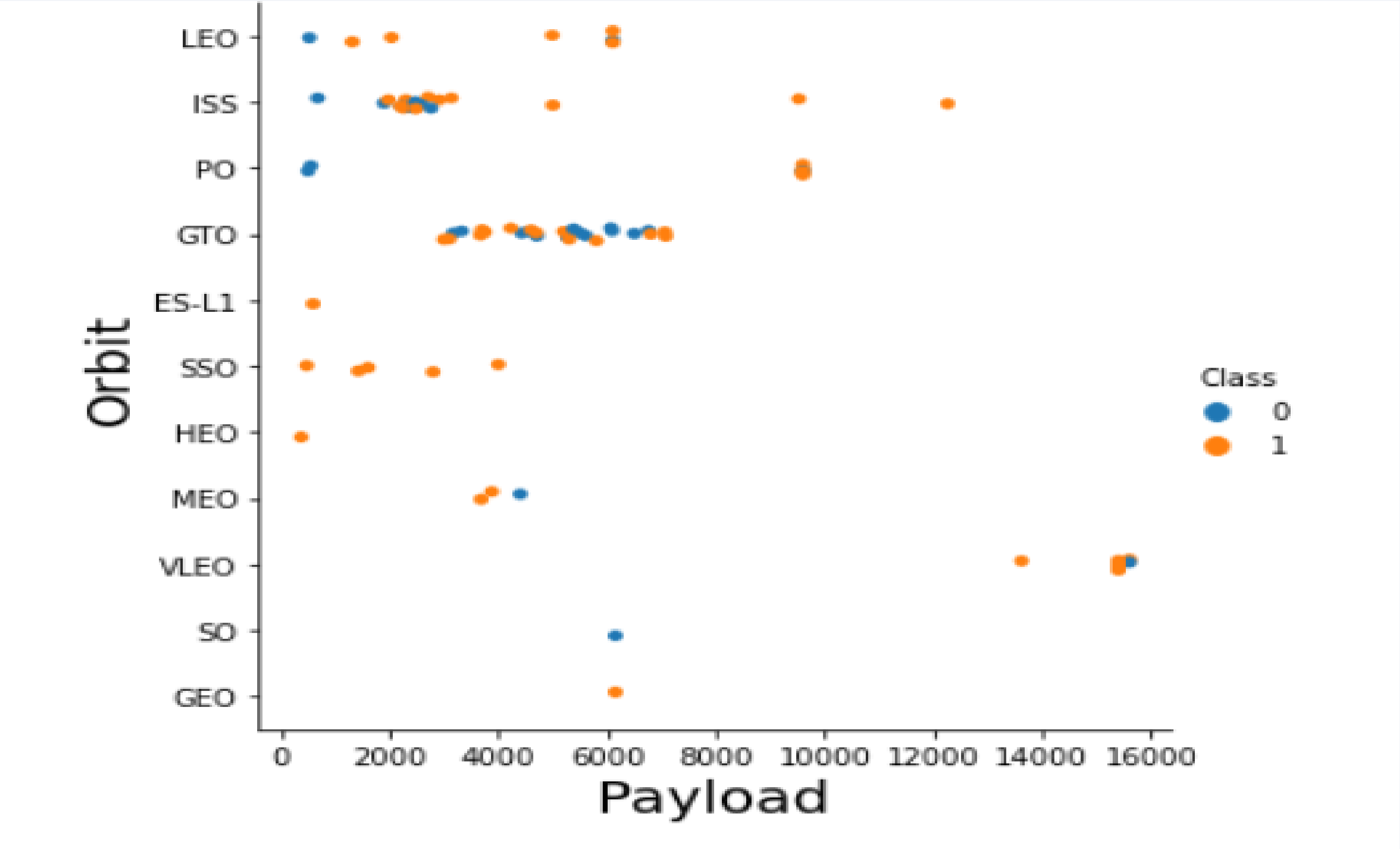


Payload vs. Orbit Type

- Blue
  dots represent unsuccessful launches, while orange dots represent successful launches

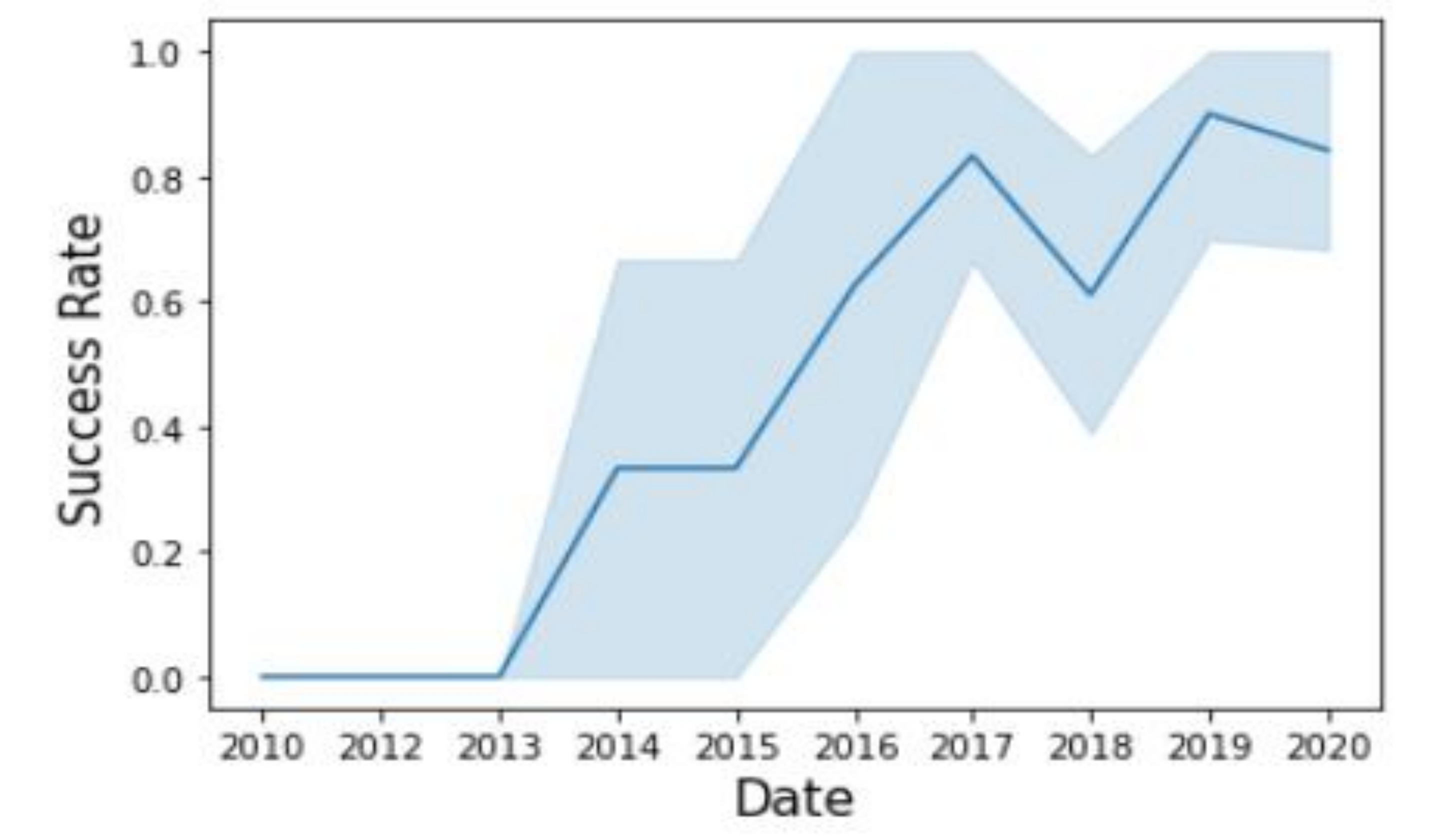
- Heavy payloads do not have negative impacts on ISS, LEO and PO orbits

- Heavy payloads have negative impacts on GTO and VLEO orbits

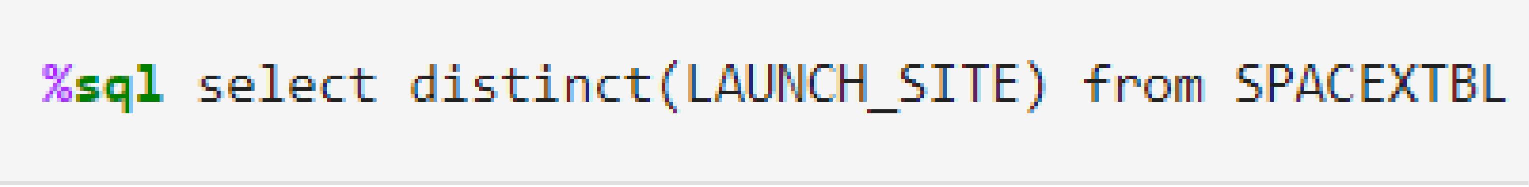


Launch Success Yearly Trend

- Success rate gradually increased from 2015-2017.
- Success rate decreased in 2018, before rising again in 2019
- Success rate dropped slightly in 2020

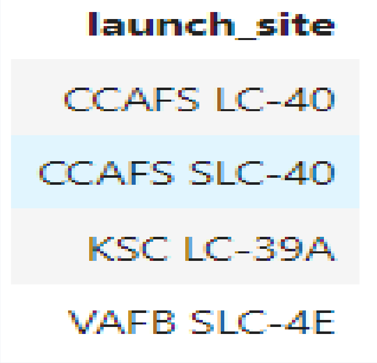


All Launch Site Names



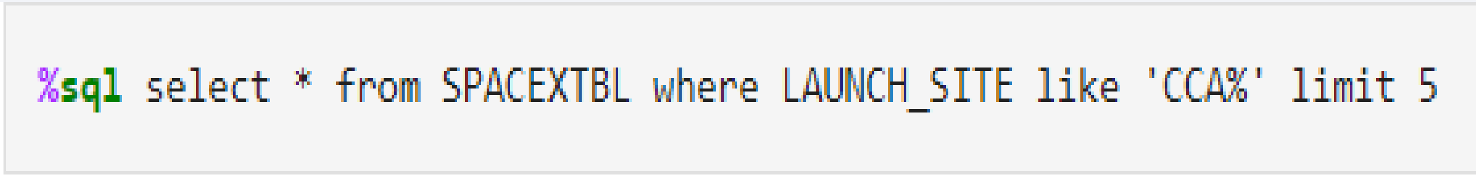
```
%sql select distinct(LAUNCH_SITE) from SPACEXTBL
```

The term 'DISTINCT' tells SQL to only returns unique values in the LAUNCH_SITE column



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

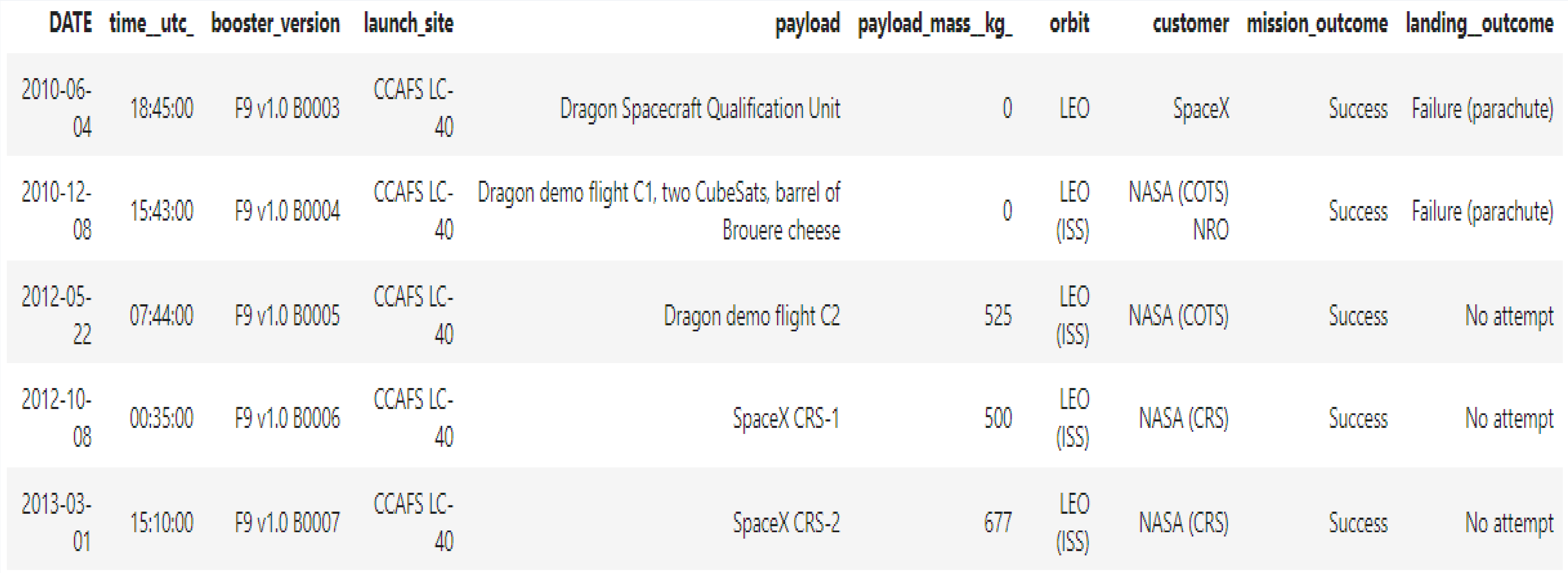
Launch Site Names Begin with 'CCA'



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

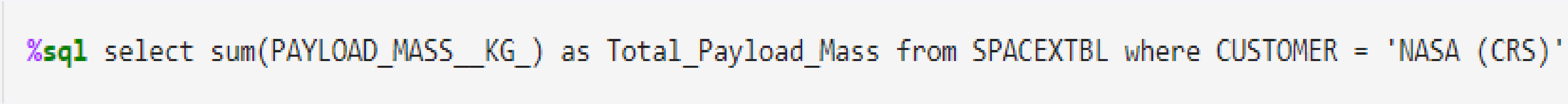
The 'CCA%' means SQL will return all records in which Launch Site starts with 'CCA'.

Limit 5 tells SQL to only take the top 5 records.



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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

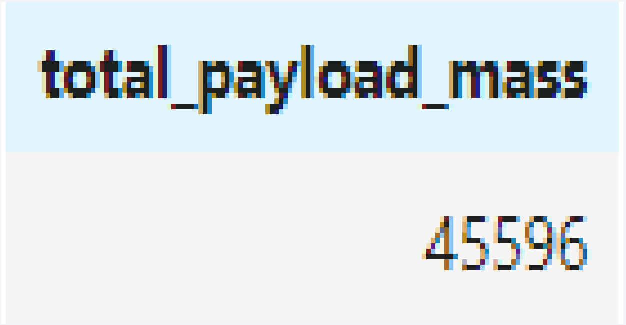
Total Payload carried by boosters launched by NASA



```
%sql select sum(PAYLOAD_MASS__KG_) as Total_Payload_Mass from SPACEXTBL where CUSTOMER = 'NASA (CRS)'
```

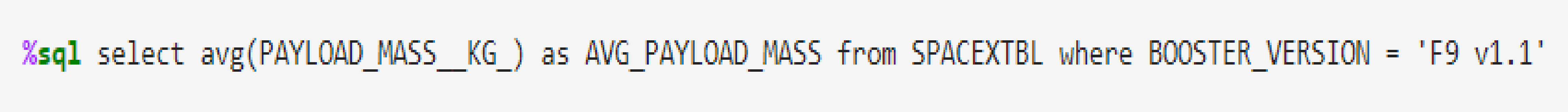
Specifying CUSTOMER = 'NASA (CRS)' commands SQL to return only records where NASA (CRS) launches the rocket.

Sum(PAYLOAD_MASS_KG_) tells SQL to calculate the sum of Payload Mass for all records that satisfy the WHERE clause



total_payload_mass
45596

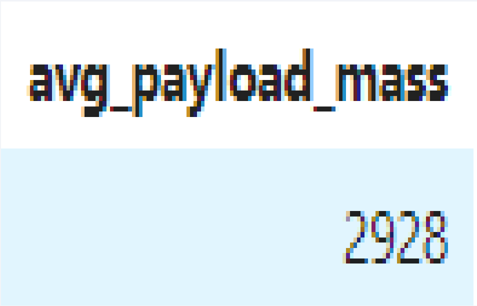
Average Payload Mass by F9 v1.1



```
%sql select avg(PAYLOAD_MASS__KG_) as AVG_PAYLOAD_MASS from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1'
```

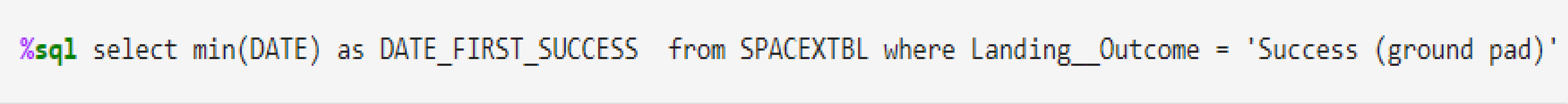
Specifying BOOSTER_VERSION = 'F9 v1.1' commands SQL to return only records where the booster version is F9 v1.1

avg(PAYLOAD_MASS_KG_) tells SQL to calculate the average of Payload Mass for all records that satisfy the WHERE clause



avg_payload_mass
2928

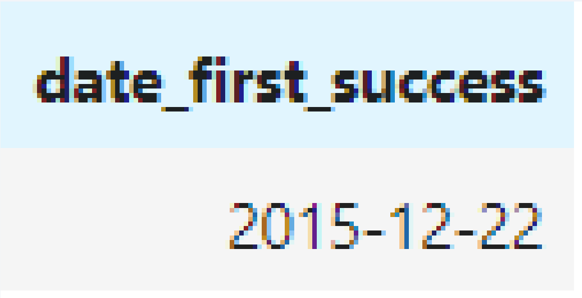
First Successful Ground Landing Date



```
%sql select min(DATE) as DATE_FIRST_SUCCESS  from SPACEXTBL where Landing__Outcome = 'Success (ground pad)'
```

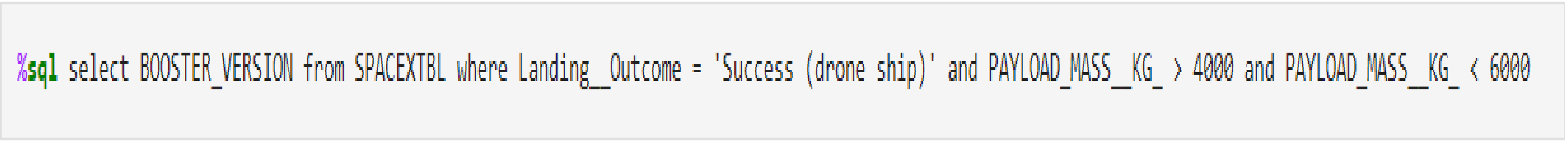
Specifying Landing_Outcome = 'Success (ground pad)' commands SQL to return only records where the landing outcome is success, and the rocket landed on ground pads.

min(DATE) tells SQL to return the min date (earliest date) that satisfy the WHERE clause



date_first_success
2015-12-22

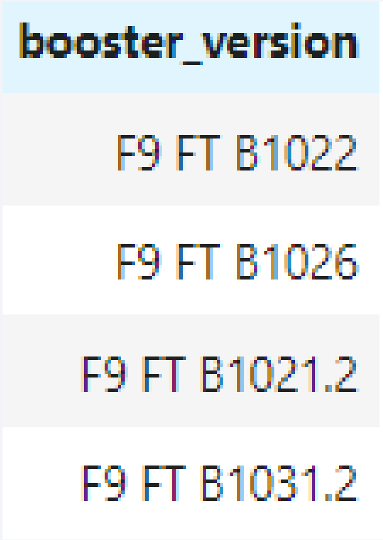
Successful Drone Ship Landing with Payload between 4000 and 6000



```
%sql select BOOSTER_VERSION from SPACEXTBL where Landing__Outcome = 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

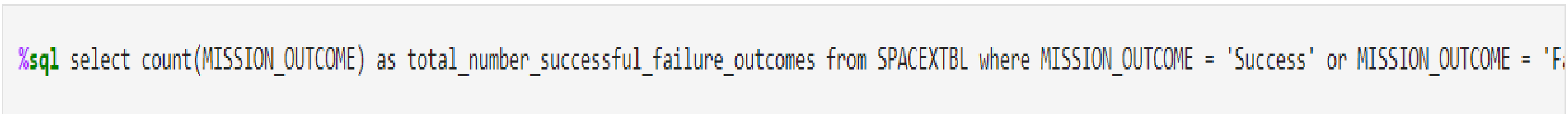
Specifying Landing_Outcome= 'Success (drone ship) and PAYLOAD_MASS_KG_ >4000 and PAYLOAD_MASS_KG_ <6000 commands SQL to return only records where these 3 conditions must be satisfied.

The rocket must have a successful landing on a drone ship, as well as a payload mass greater than 4000 and less than 6000



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

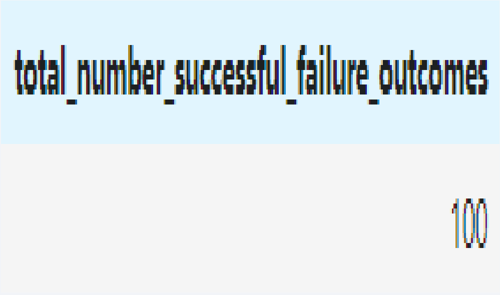
Total Number of Successful and Failure Mission Outcomes



```
%sql select count(MISSION_OUTCOME) as total_number_successful_failure_outcomes from SPACEXTBL where MISSION_OUTCOME = 'Success' or MISSION_OUTCOME = 'F
```

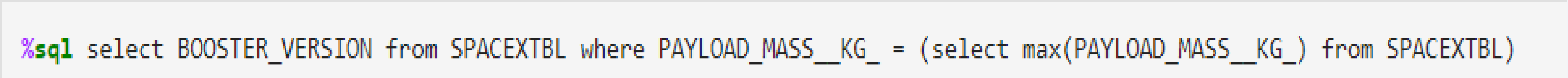
Specifying OR commands SQL to return only records Mission Outcome can either be Success or Failure.

'Count' is used to count the number of records that satisfy the WHERE clause



total_number_successful_failure_outcomes
100

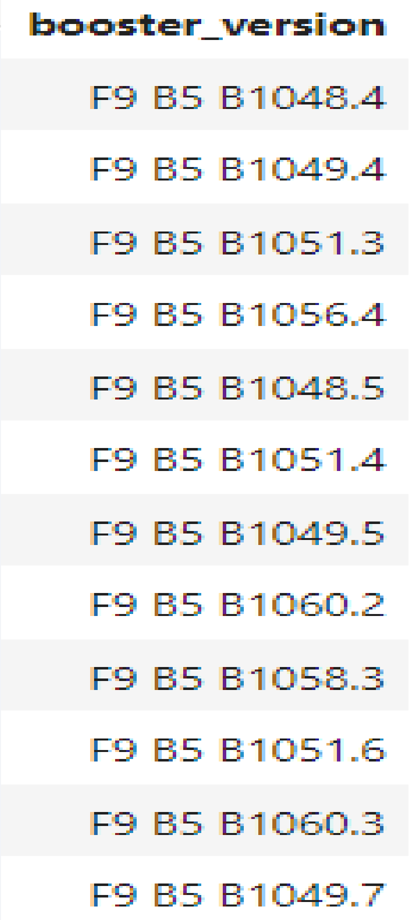
Boosters Carried Maximum Payload



```
%sql select BOOSTER_VERSION from SPACEXTBL where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

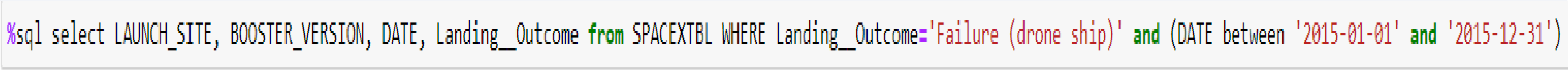
In this query, the subquery (select max(PAYLOAD_MASS__KG_) from SPACEXTBL) will be executed first. Hence, SQL will first obtain the max value of Payload Mass.

SQL then finds the booster version of rockets where the payload max equals to the value obtained from the subquery



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

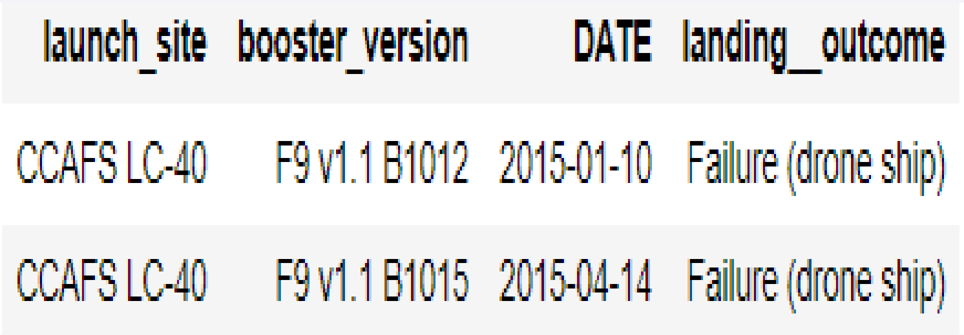
2015 Launch Records



```
%sql select LAUNCH_SITE, BOOSTER_VERSION, DATE, Landing__Outcome from SPACEXTBL WHERE Landing__Outcome='Failure (drone ship)' and (DATE between '2015-01-01' and '2015-12-31')
```

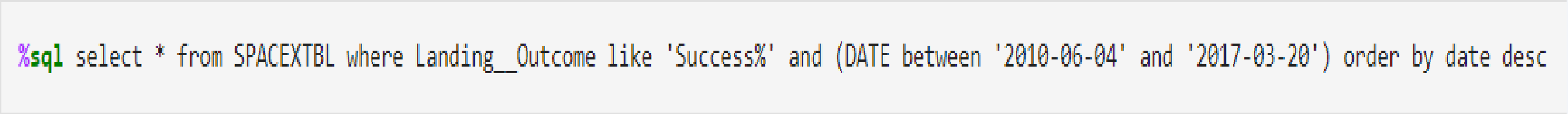
Specifying Landing__Outcome='Failure (drone ship)' and (DATE between '2015-01-01' and '2015-12-31') in the WHERE clause commands SQL to return only records where these 3 conditions must be satisfied.

The rocket must be unsuccessful when landing on a drone ship, and this failure must occurred between 01/01/2015 and 31/12/2015



launch_site	booster_version	DATE	landing__outcome
CCAFS LC-40	F9 v1.1 B1012	2015-01-10	Failure (drone ship)
CCAFS LC-40	F9 v1.1 B1015	2015-04-14	Failure (drone ship)

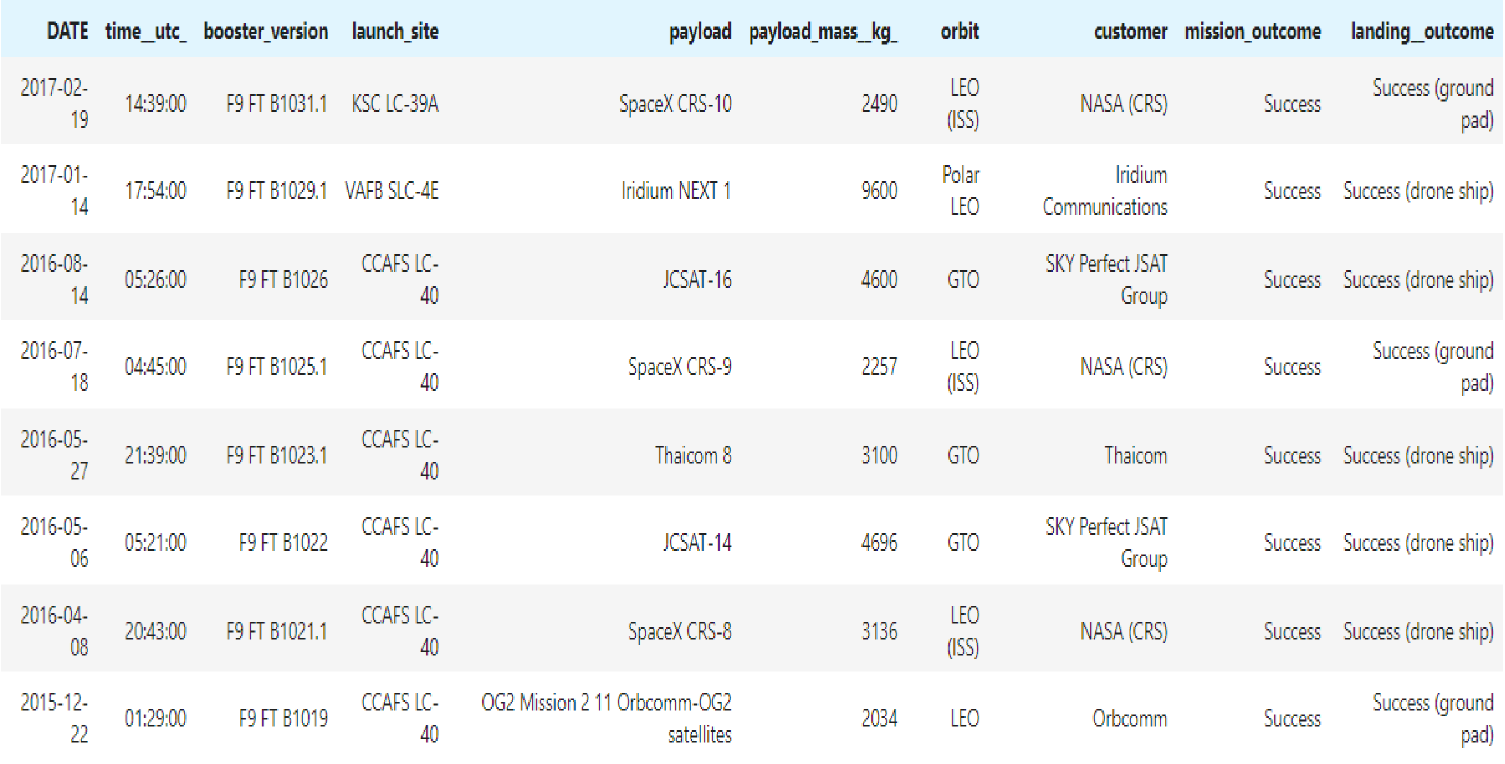
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20



```
%sql select * from SPACEXTBL where Landing__Outcome like 'Success%' and (DATE between '2010-06-04' and '2017-03-20') order by date desc
```

The WHERE clause has 3 conditions including Landing_Outcome that starts with Success and dated between 2010-06-04 and 2017-03-20.

Order by date desc ranks records based on date in descending order.

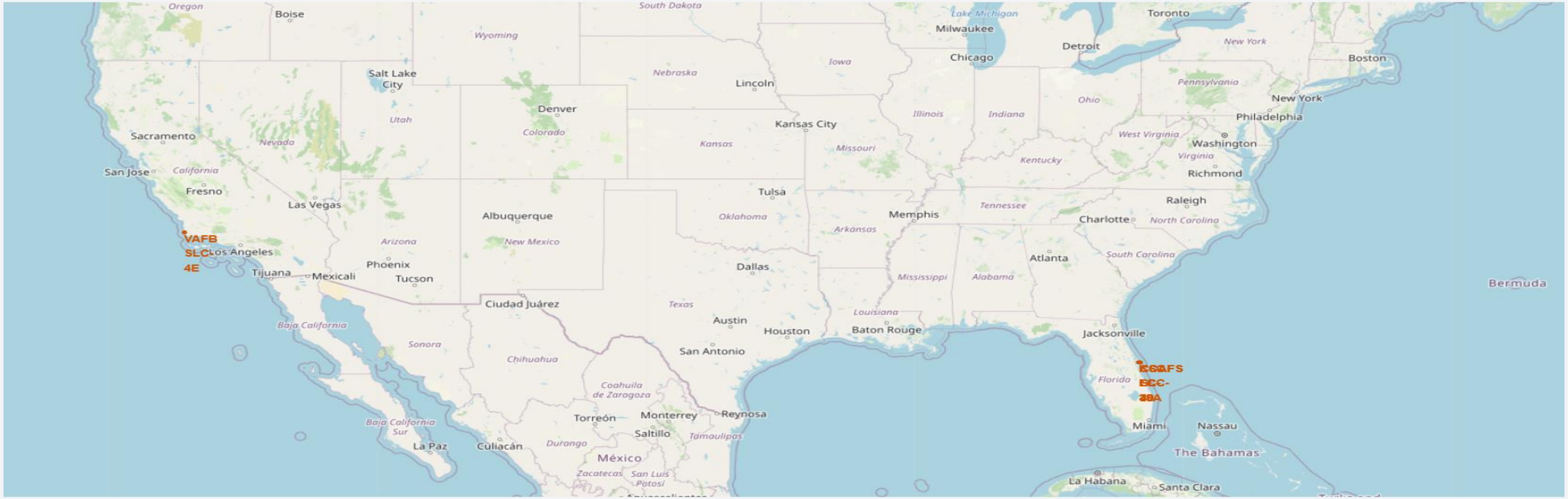


DATE	time_utc_	booster_version	launch_site	payload	payload_mass_kg_	orbit	customer	mission_outcome	landing_outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-01-14	17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (drone ship)
2016-08-14	05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-07-18	04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
2016-05-06	05:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)
2015-12-22	01:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)

Section 4

Launch Sites Proximities Analysis

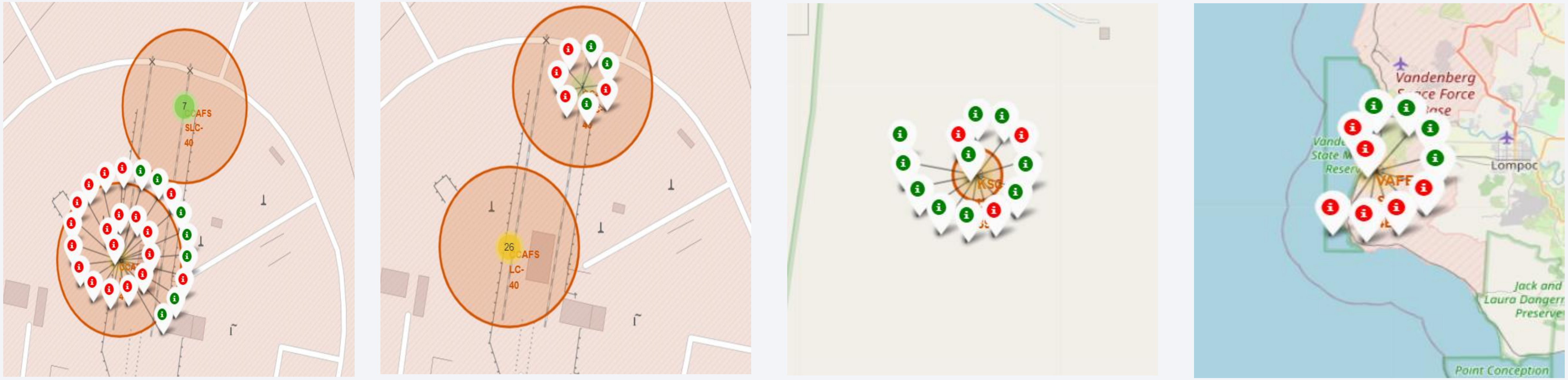
Map of all launch sites



All launch site locations are indicated by red circles with site names as labels

3 launch sites are located on the East Coast, while only 1 launch site is located on the West Coast

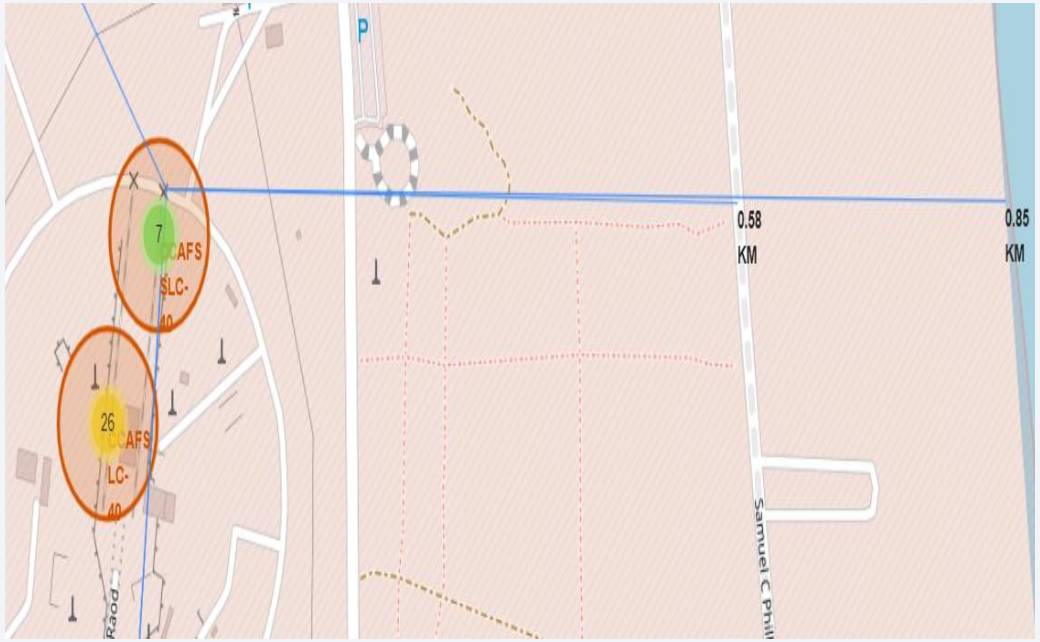
Color-labeled markers

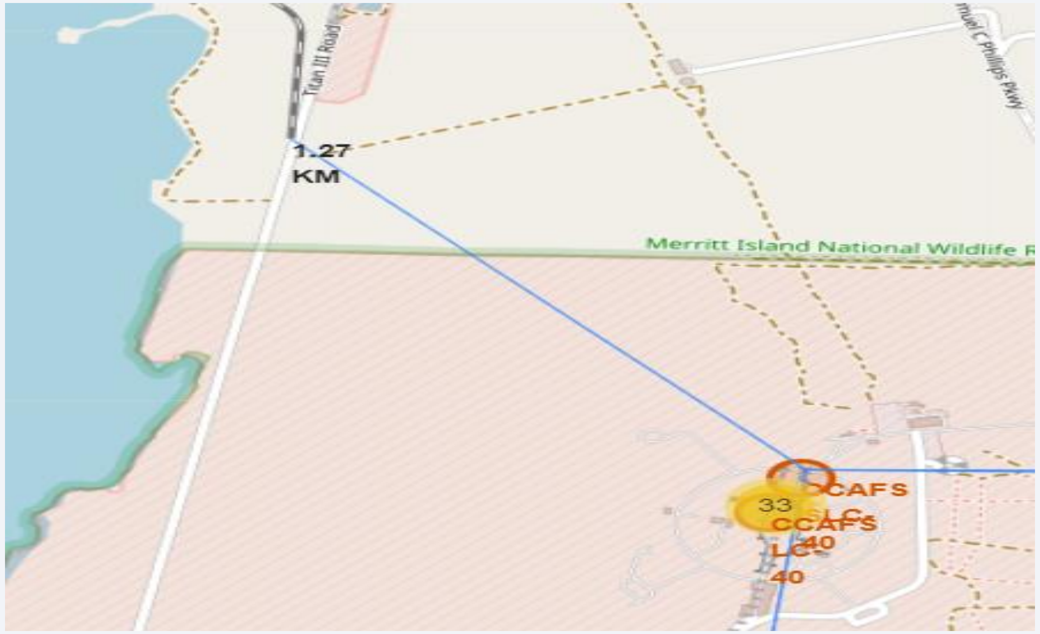


Red markers indicate failure landing outcomes, while green markers indicate successful landing outcomes

KSC LC-39A has the highest success rate with 10/13 successful landing

Launch distance from CCAFS SLC-40



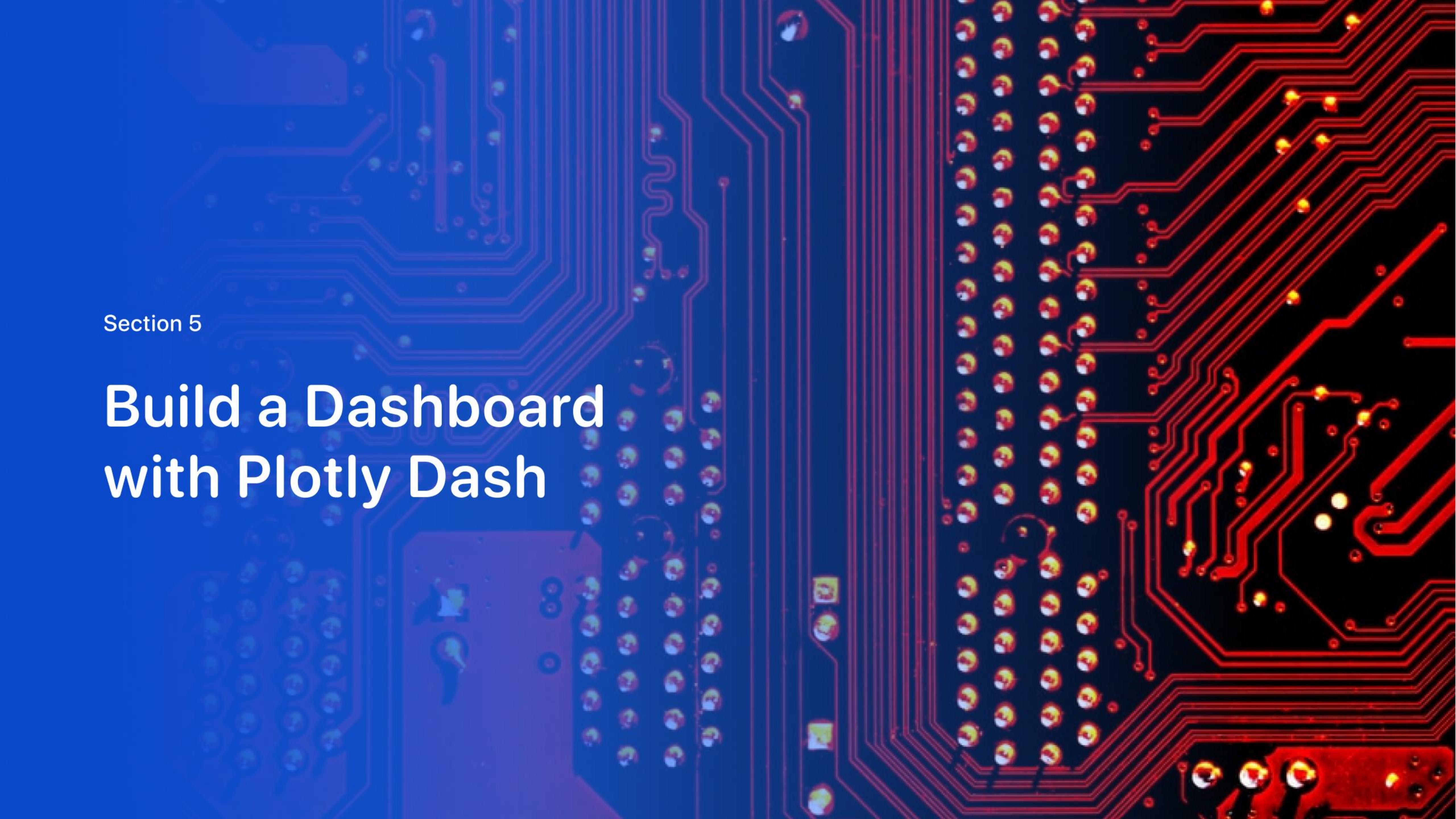


The blue lines measure the distance between launch site CCAFS SLC-40 vs coastline, the nearest highway and railways.

Distance from CCAFS SLC-40 to coastline is 0.85km

Distance from CCAFS SLC-40 to the nearest highway is 0.58km

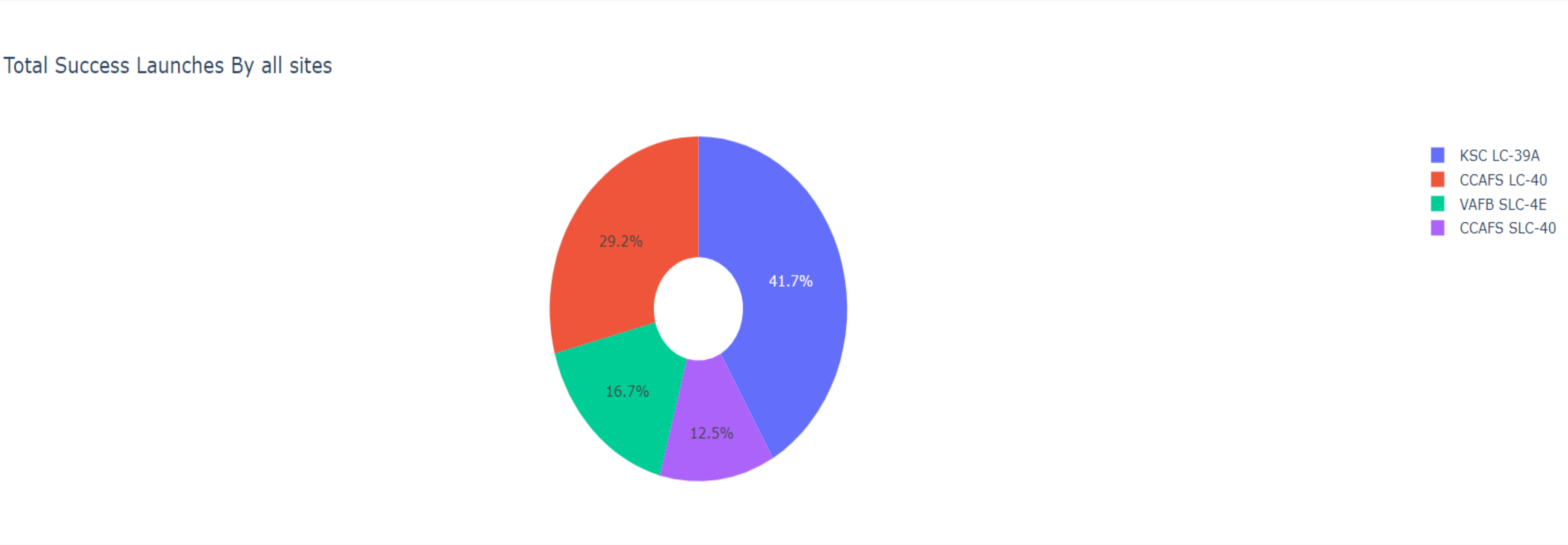
Distance from CCAFS SLC-40 to the nearest railway is 1.27km



Section 5

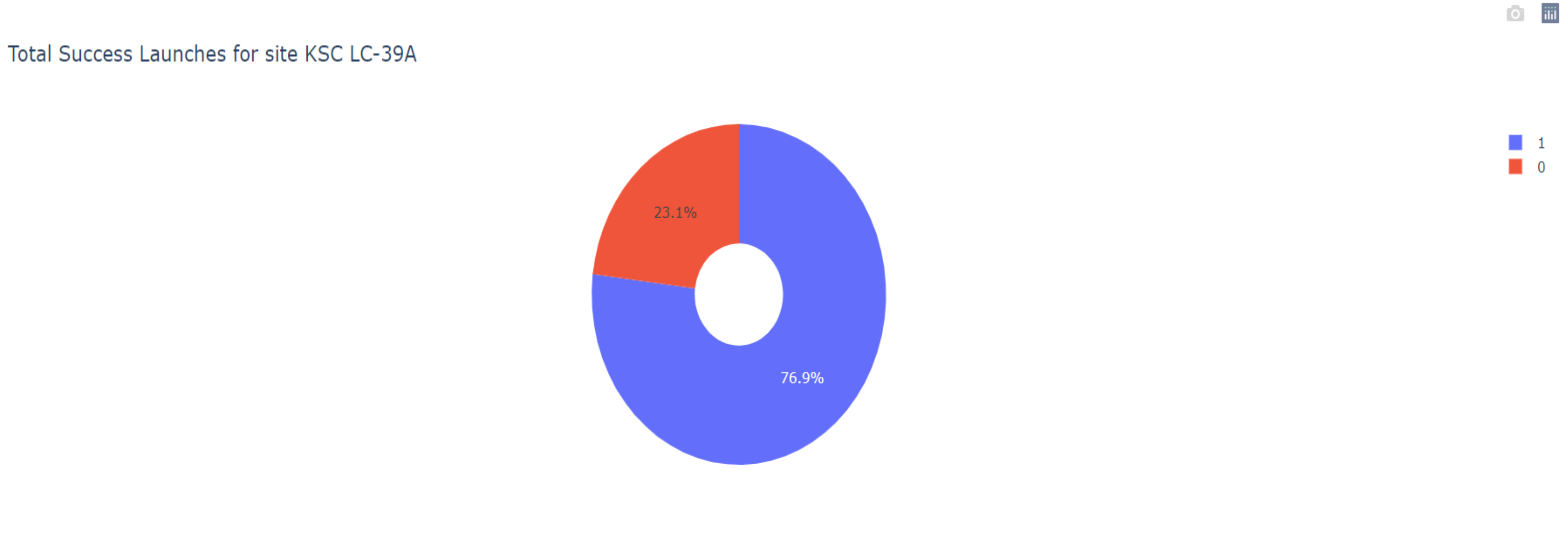
Build a Dashboard with Plotly Dash

Launch Success Count for all sites



KSC LC-39A has the highest success rate among all launch sites accounting for 41.7% of all successful landing. KSC LC-39A is followed by CCAPS LC-40 which accounts for 29.2% of all successful landing.

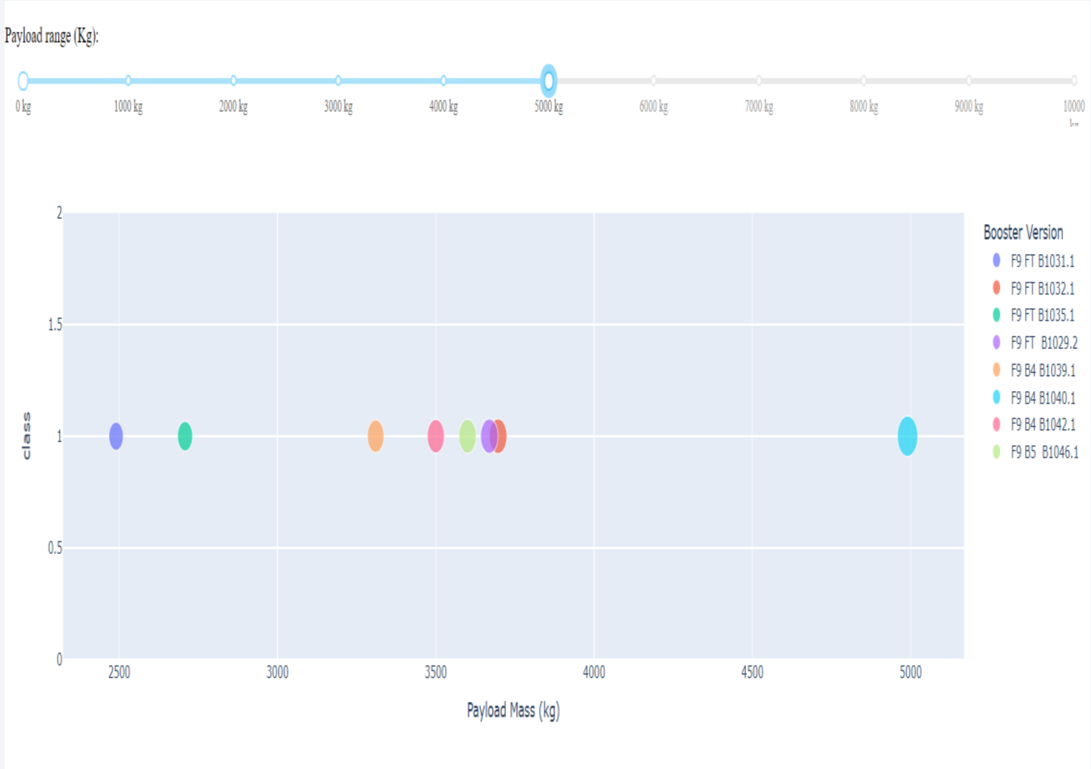
Launch Site with the highest success ratio

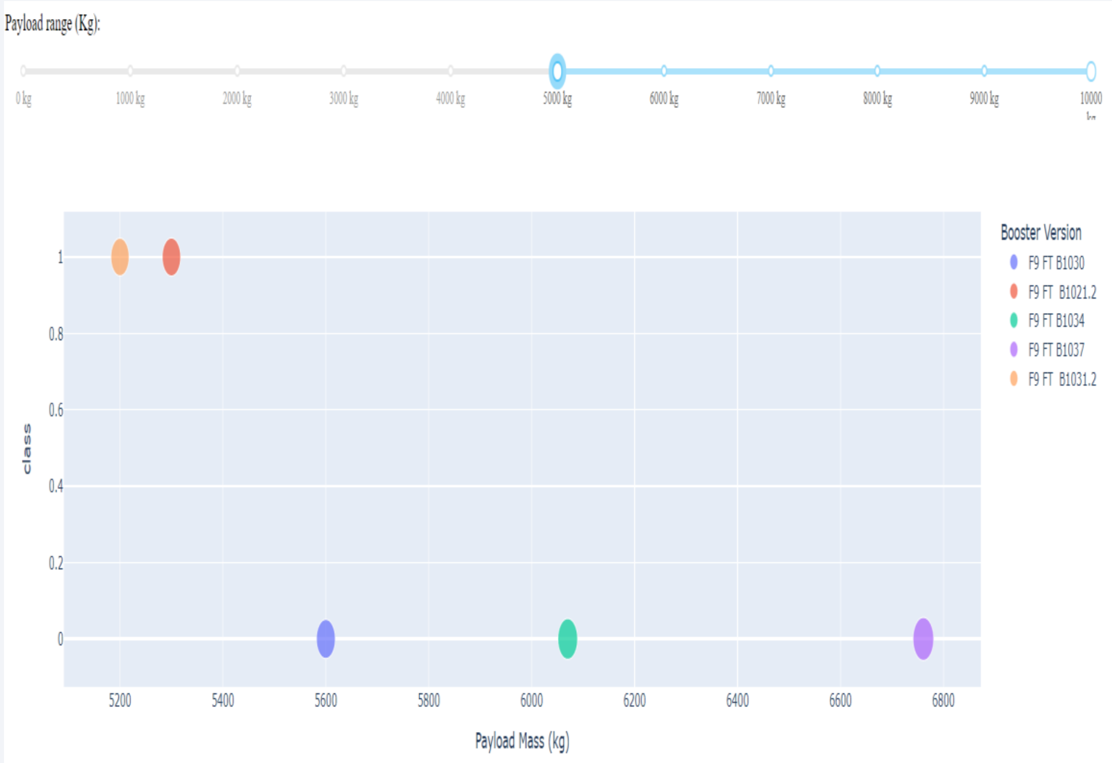


Total Success Launches for site KSC LC-39A

KSC LC-39A has the highest success rate among all launch sites. 76.9% of all launches from this site landed successfully.

Success rate by Payload Mass





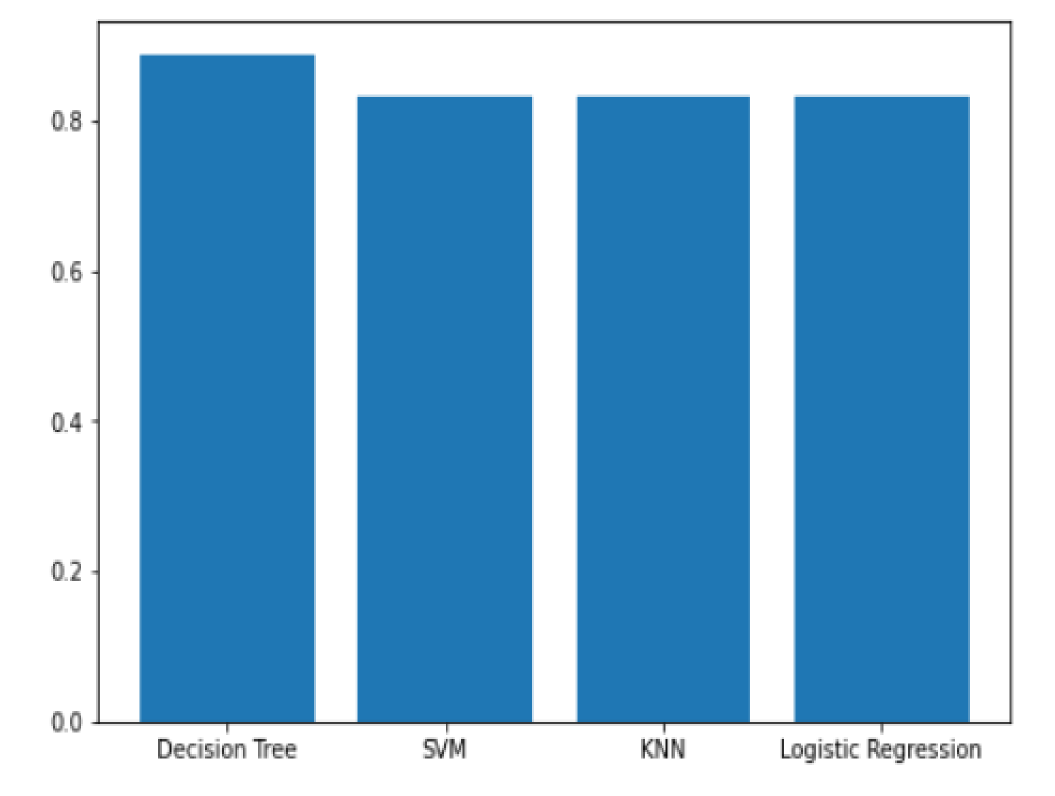
Launches with a lower Payload Mass (0-5000kg) have a higher success rate than launches with a higher Payload Mass (>5000kg)

Section 6

Predictive Analysis (Classification)

Classification Accuracy

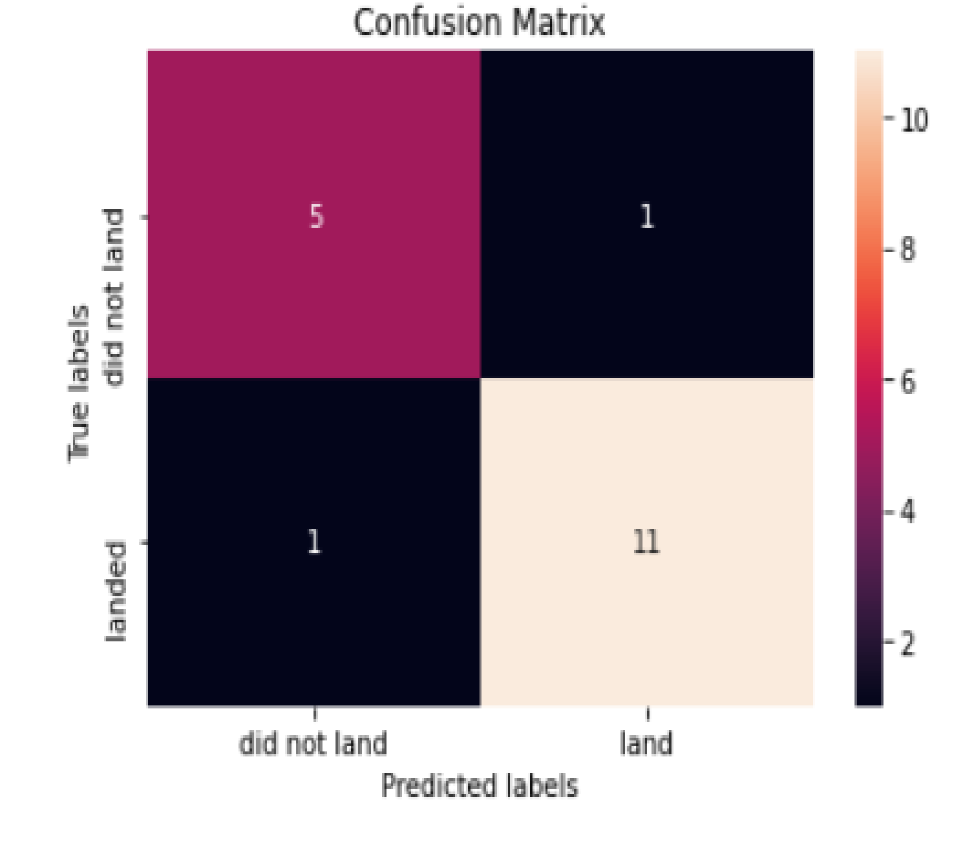
- Decision Tree achieved the highest accuracy score at 0.8889
- SVM, KNN and Logistic Regression all achieved an accuracy score of 0.8333
- After tuning with GridSearchCV, the accuracy of KNN, SVM and Logistic Regression increased, while the accuracy of Decision Tree dropped.



Confusion Matrix

The Decision Tree model correctly predicted 5 Failure Outcomes (TN) and 12 Successful Outcomes (TP), as well as incorrectly predicted 1 Successful Outcomes (FP) and 1 Failure Outcome (FN)

Classification accuracy = (TP+TN)/(TP+TN+FP+FN)=16/18=0.8889



Conclusions

- KSC LC-39A has the most successful landing outcome of all launch sites
- Launches with a low Payload Mass (<5000kg) have a higher success rate than launches with a high Payload Mass (>5000kg)
- Launch sites are placed relatively close to highways and railways, and far away from metropolitan areas
- Orbit GEO, HEO, SSO, and ES-L1 have the highest success rates
- Decision Tree is the best Machine Learning model to predict the landing outcome

Thank you!