Winning Space Race with Data Science

Roman

Oct. 5 2021

Outline

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

Executive Summary

Summary of methodologies

- Data collection
- Data wrangling
- EDA with data visualization
- Building an interactive map using Foilum
- Building a Dashboard with Plotly Dash
- Predictive analysis

Summary of all results

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

Introduction

- **Project background and context**

Falcon 9 rocket launches cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.

If we can determine:

- condition for launch to be successful we can reduce launch cost
- if the first stage will land, we can determine the cost of a launch

- **Problems you want to find answers**

- rocket landed successfully/unsuccessfully? why?
- what rocket variables have relation with success rate?
- what conditions does SpaceX have to achieve high success landing rate?

Section 1

Methodology

Methodology

Executive Summary

- **Data collection methodology:**

  SpaceX Rest API

  Web scrapling https://en.wikipedia.org/

- **Performed data wrangling**

  One Hot Encoding data fields and dropping irrelevant columns

- **Performed exploratory data analysis (EDA) using visualization and SQL**

  Plotting : Scatter Graphs, Bar Graphs to show relationships between variables to show patterns of data.

- **Performed interactive visual analytics using Folium and Plotly Dash**

- **Performed predictive analysis using classification models, and models turning**

Data Collection

Data sets were collected:

- SpaceX REST API (This API will give us data about launches, including information about the rocket used,payload delivered, launch specifications, landing specifications, and landing outcome).

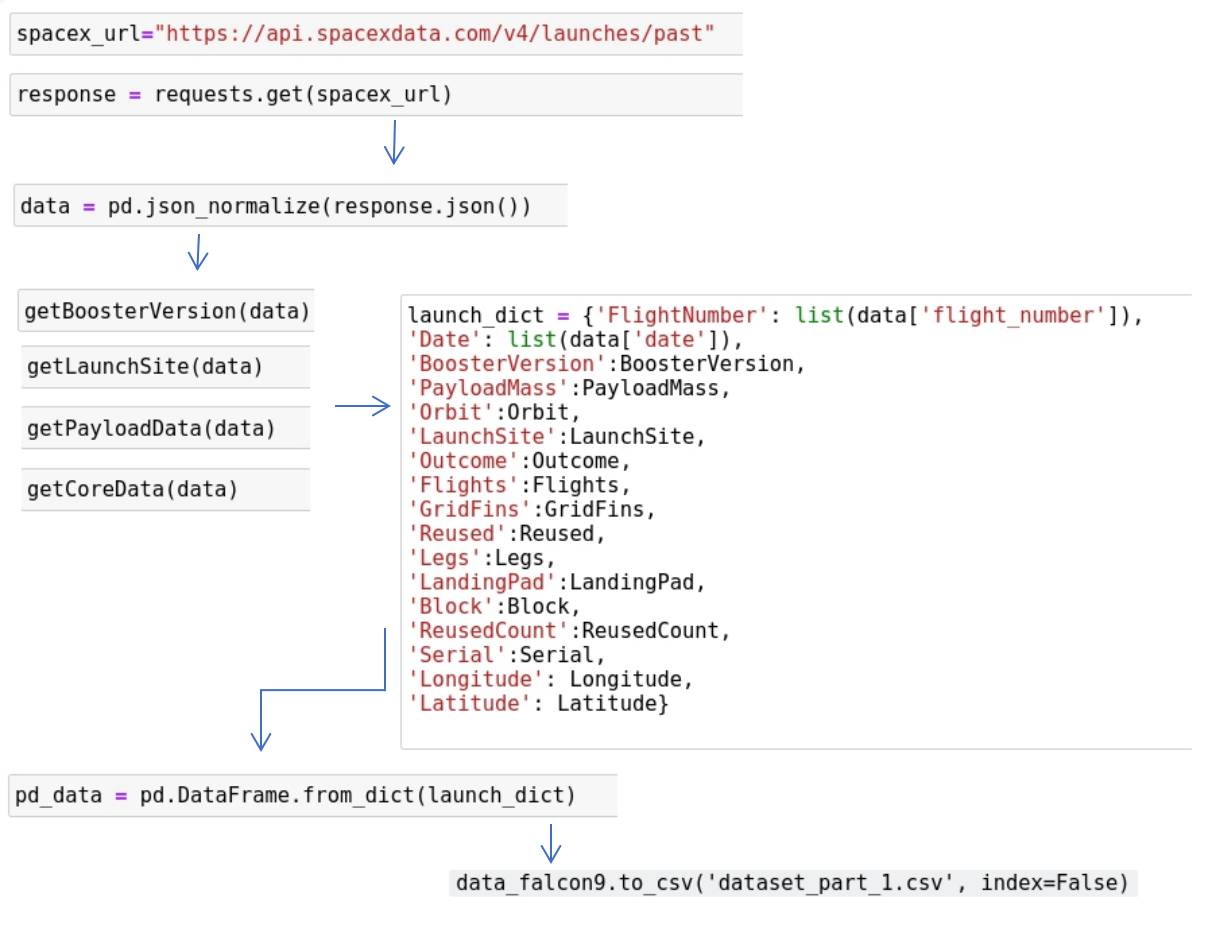
  The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.

• Alternative way for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup.

Data Collection – SpaceX API

1. Getting responce from API
2. Converting to json and normalizing
3. Apply custom functions to clean data
4. Assign list to dictionaty
5. Assign dictionary to dataframe
6. Export dataframe to csv file

GitHub



Data Collection - Scraping

1,2. Getting response from wiki page

3. Creating BS object

4. Finding tables

5. Getting column names

6. Creating dictionary

7. Filling dictionary with data

8. Converting dictionary to dataframe

9. Dataframe to .csv

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

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

```
soup = BeautifulSoup(response.text, 'html5lib'
```

```
tables_row = soup.find_all("table")
```

```
column_names = []

th_elements = first_launch_table.find_all("th")
for th in th_elements:
    name = extract_column_from_header(th)
    if name is not None and len(name) > 0:
        column_names.append(name)
```

```
launch_dict= dict.fromkeys(column_names)

del launch_dict['Date and time ( )']

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'] = []
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(so
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table h
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.
                flag=flight number.isd
```

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

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

[GitHub]

Data Wrangling

- 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

EDA with Data Visualization

Scatter Graphs:

- Flight Number VS. Payload Mass
- Flight Number VS. Launch Site
- Payload VS. Launch Site
- Orbit VS. Flight Number
- Payload VS. Orbit Type
- Orbit VS. Payload Mass

Scatter plot shows how much one variable is afected by another

GitHub

Bar charts:

- Mean VS. Orbit

Bar chart makes it easy to compare sets of data between different groups.

Line graph:

- Success rate VS. Year

Line graphs show data variables and trends very clearly and can help to make predictions.

EDA with SQL

We used sql queries to answer those questions:

- Names of the unique launch sites in the space mission
- The total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- The first successful landing outcome in ground pad was acheived
- The boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- The total number of successful and failure mission outcomes
- The names of the booster_versions which have carried the maximum payload mass
- 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

GiHub

Build an Interactive Map with Folium

- We took the Latitude and Longitude Coordinates at each launch site and added a Circle Marker around each launch site with a label of the name of the launch site.
- We assigned the dataframe launch_outcomes(failures, successes) to classes 0 and 1 with Green and Red markers on the map in a MarkerCluster()
- Using Haversine's formula we calculated the distance from the Launch Site to various landmarks.

GitHub

Build a Dashboard with Plotly Dash

Scatter Graph
showing the relationship with Outcome and Payload Mass (Kg) for the different Booster
Versions
- It shows the relationship between two variables.
- It is the best method to find intuition about not linear dependency.
- The range of data flow, i.e. maximum and minimum value, can be determined.

GitHub

Pie Chart
showing the total launches by a certain site or all sites:
- display relative proportions of multiple classes of data.
- size of the circle can be made proportional to the total quantity it represents.

Red is 0(failure),
Green is 1(success)

Predictive Analysis (Classification)

Building model:

1. Load data from .csv file to Pandas and NumPy
2. Transform data
3. Spliting data on train and test set
4. Trying Logistic regression, SVM, decision tree, SVM with GridSearchCV

Model evaluating:

1. Check accuracy for each model
2. Get tuned hyperparameters for each type of algorithms
3. Plot Confusion Matrix

Model choosing:

Choosing the model with the best accuracy score

GitHub

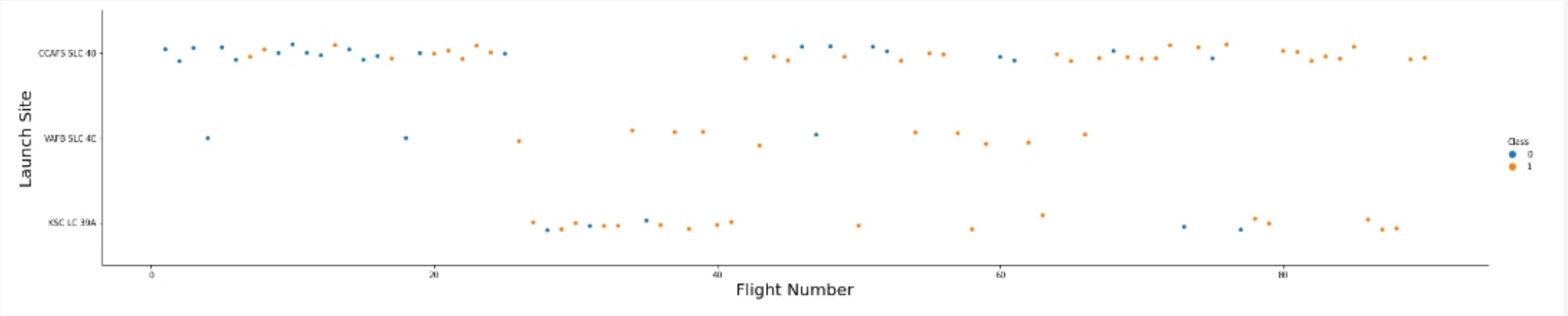
Results

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

Section 2

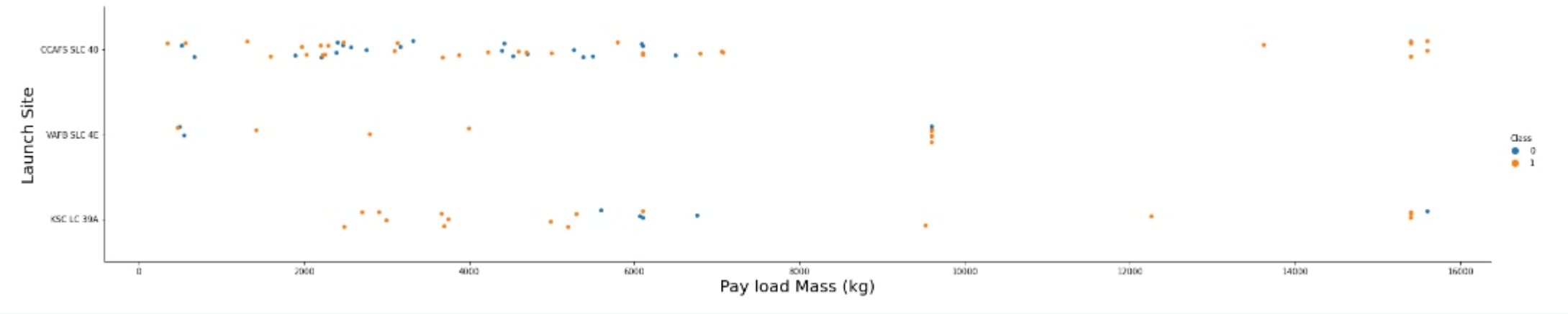
Insights drawn from EDA

Flight Number vs. Launch Site



- More recent mission have higher success rate at all sites
- CCAF SLS 40 has highest amount of launches and lowest success rate

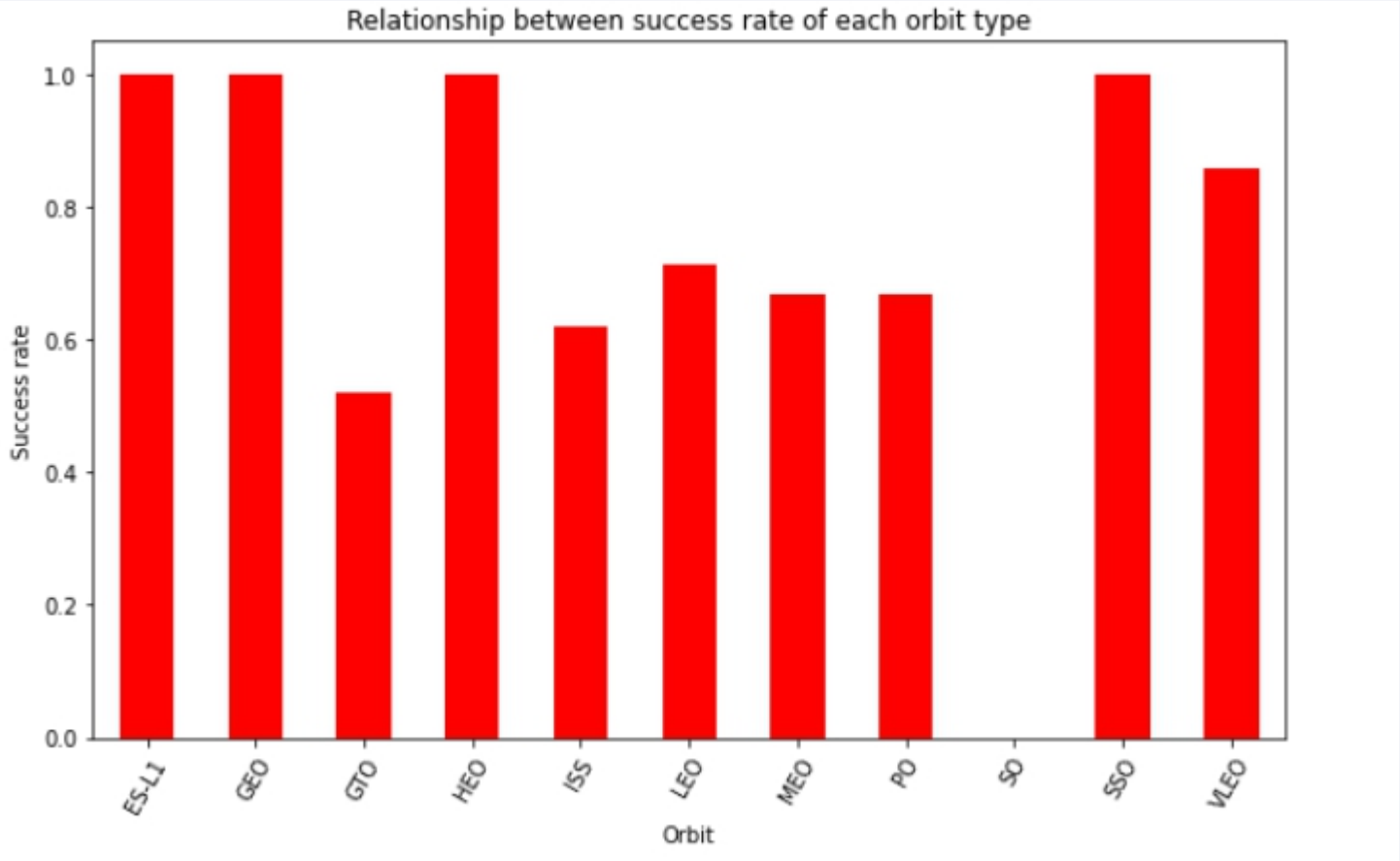
Payload vs. Launch Site



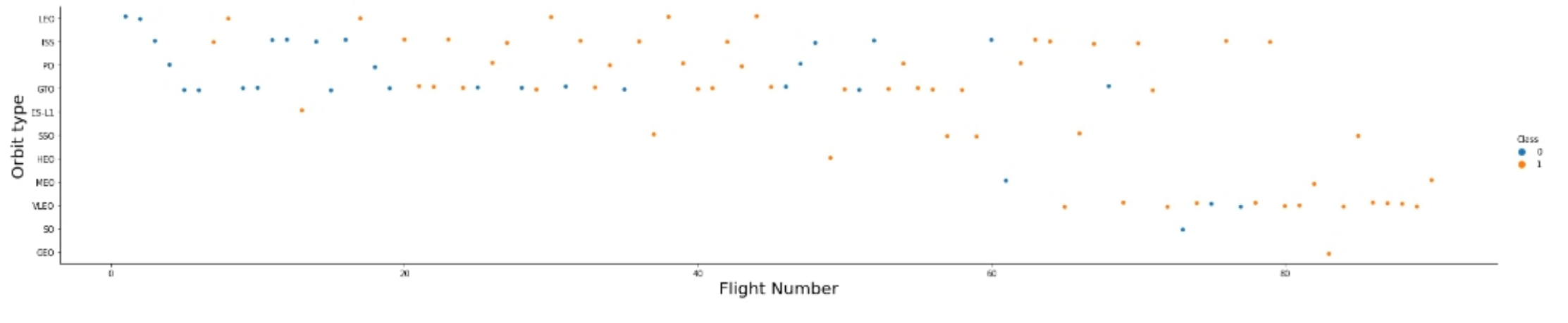
- Looks like the greater the payload mass - the higher the success rate for the Rocket.
- There is no clear pattern between Launch SIte and Payload mass

Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO have highest success rate

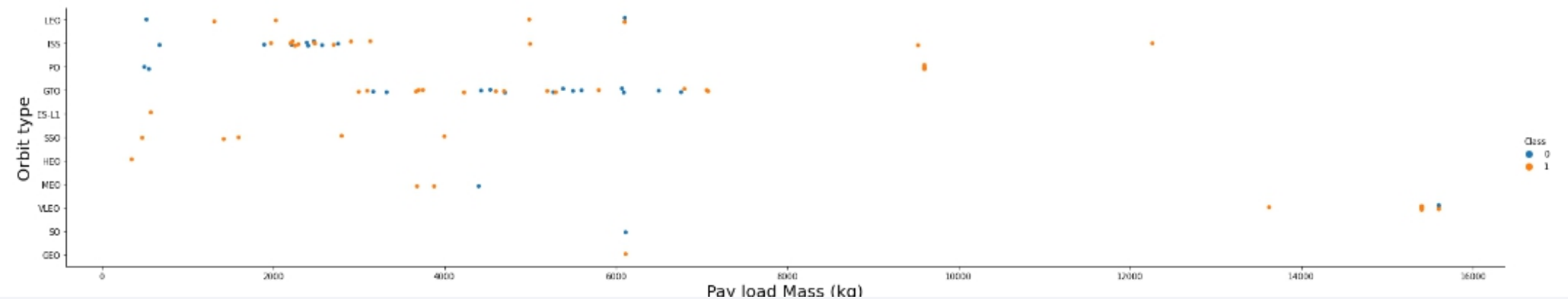


Flight Number vs. Orbit Type



- There seems to be no relationship between flight number when in GTO orbit.

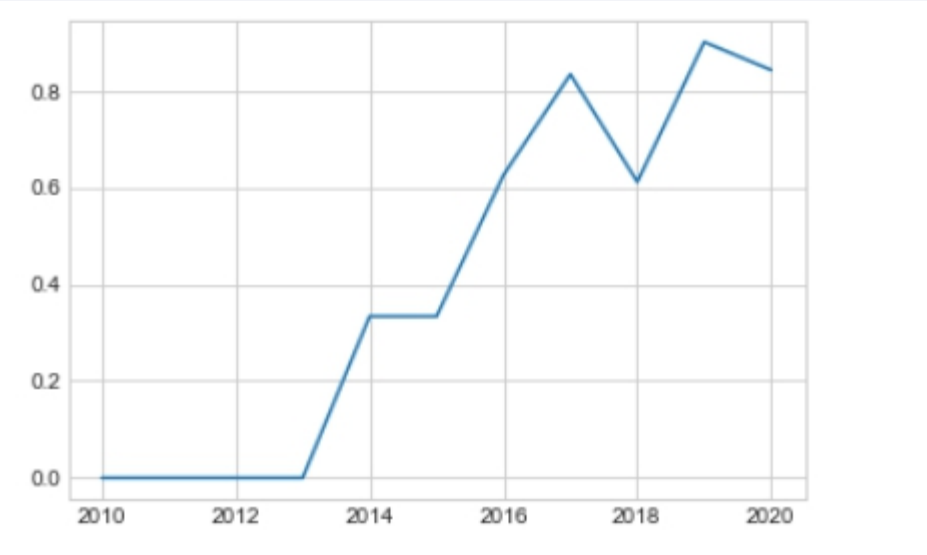
Payload vs. Orbit Type



There seems not to be relationship between Orbit type and Payload mass

Launch Success Yearly Trend

We can observe that the sucess rate since 2013 kept increasing till 2020



All Launch Site Names

The names of the unique launch sites:

SELECT DISTINCT launch_site FROM SPACEXTBL

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

Launch Site Names Begin with 'CCA'

5 records where launch sites begin with `CCA`

```
SELECT * FROM SPACEXTBL
 WHERE launch_site LIKE 'CCA%'
LIMIT 5;
```

Using the word LIMIT 5 in the query means that it will only show 5 records from teble and LIKE keyword has a wild card with the words ‘CCA%’ the percentage in the end suggests that

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 Mass

The total payload carried by boosters from NASA:

SELECT SUM(payload_mass__kg_) FROM SPACEXTBL

WHERE customer = 'NASA (CRS)'

1
45596

Using the function SUM summates the total in the column payload_mass_kg_. The WHERE clause filters the dataset to only perform calculations on Customer NASA (CRS)

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 v1.1:

SELECT AVG(payload_mass__kg_) FROM SPACEXTBL

WHERE booster_version = 'F9 v1.1'

1
2928

Using the function AVG works out the average in the column payload_mass_kg_. The WHERE clause filters the dataset to only perform calculations on Booster_version F9 v1.1

First Successful Ground Landing Date

The dates of the first successful landing outcome on ground pad:

SELECT MIN(date) FROM SPACEXTBL

WHERE landing__outcome = 'Success (ground pad)'

1
2015-12-22

Using the function MIN works out the minimum date in the column Date. The WHERE clause filters the dataset to only perform calculations on landing_outcome ‘Success (ground pad)’

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
SELECT booster_version FROM (SELECT * FROM SPACEXTBL
    WHERE payload_mass__kg_ BETWEEN 4000 AND 6000)
    WHERE landing__outcome = 'Success (drone ship)'
```

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

Using subquery to retrieve payload mass between 4000 and 6000. Result is filtered on landing outcome = ‘Successful (drone ship)’

Total Number of Successful and Failure Mission Outcomes

The total number of successful and failure mission outcomes

SELECT mission_outcome, COUNT(mission_outcome) AS total FROM SPACEXTBL

GROUP BY mission_outcome

ORDER BY total DESC

mission_outcome	total
Success	99
Failure (in flight)	1
Success (payload status unclear)	1

We group table by mission outcome and count amount of outcomes per group, order by DESC will show result in descent order

Boosters Carried Maximum Payload

List the names of the booster which have carried the maximum payload mass:

SELECT DISTINCT(booster_version) FROM SPACEXTBL

WHERE payload_mass__kg_ = (SELECT MAX(payload_mass__kg_) FROM SPACEXTBL)

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

Using the word DISTINCT in the query means that it will only show Unique values in the Booster_Version column from the table. Using sub query for retrieving maximum payload mass.

2015 Launch Records

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

```
SELECT booster_version, launch_site FROM (SELECT * FROM SPACEXTBL
                                          WHERE DATE >='01-01-2015' AND DATE < '01-01-2016')
   WHERE landing__outcome = 'Failure (drone ship)'
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

Using sub query for retrieving 2015 data table, searching this table with landing__outcome =’Failure (dron ship)’ condition

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

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order:

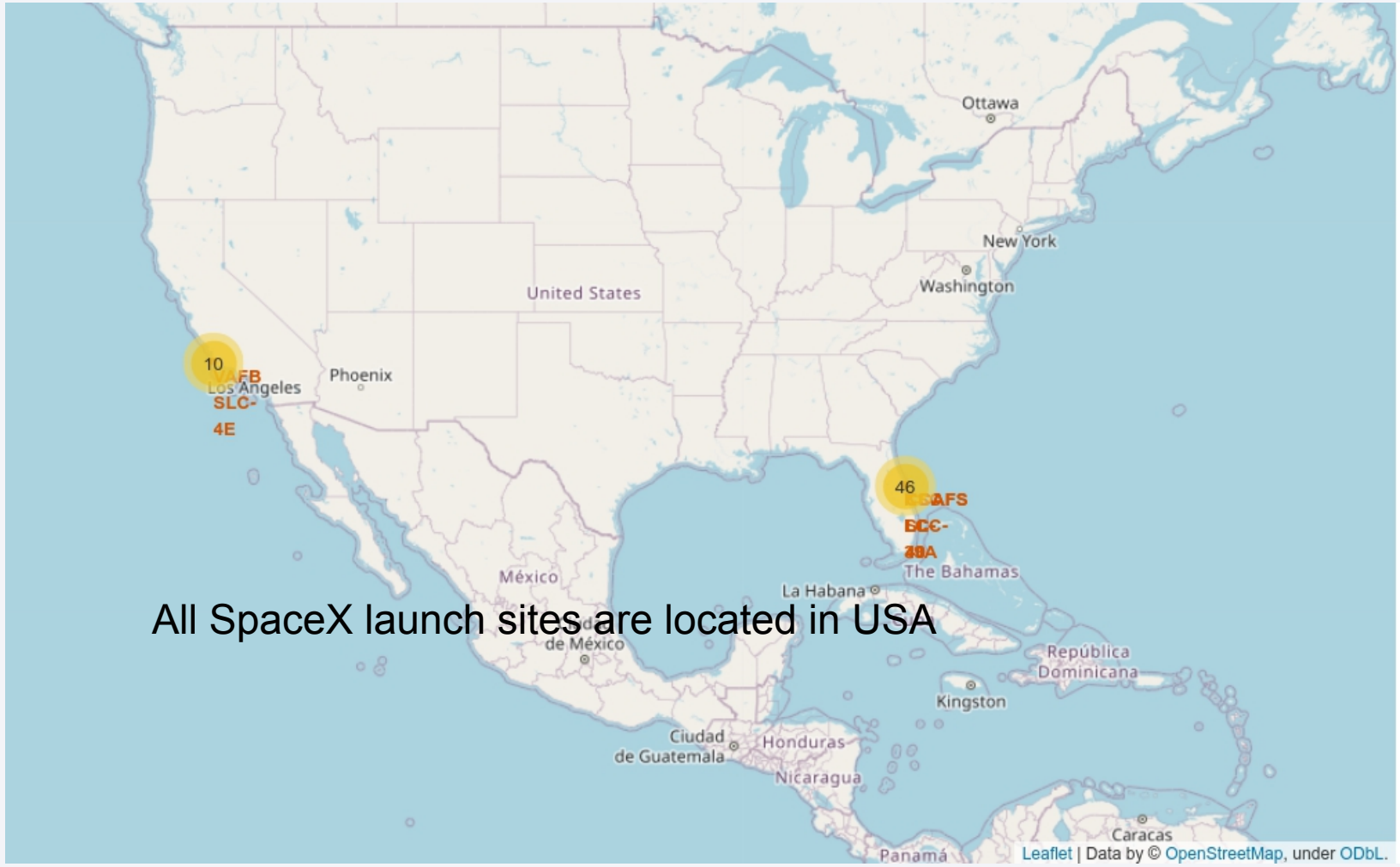
```
SELECT landing__outcome, COUNT(landing__outcome) AS total FROM (SELECT * FROM SPACEXTBL
                                                    WHERE DATE >='06-04-2010' AND DATE <= '03-20-2017')
    GROUP BY landing__outcome
    ORDER BY total DESC
```

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

Section 4

Launch Sites Proximities Analysis

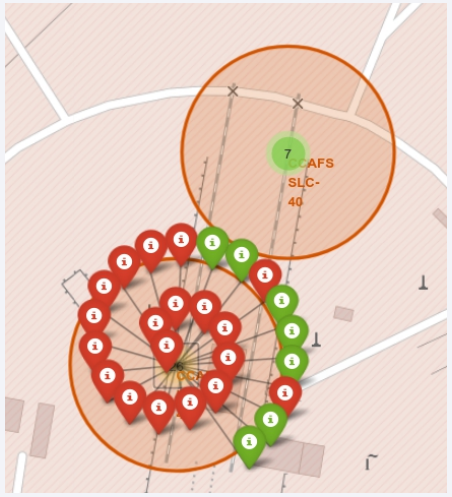
SPACEX lauch sites

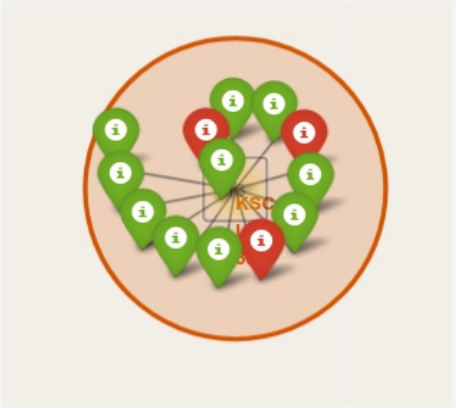


All SpaceX launch sites are located in USA

Colour labeled markers

Florida launch sites





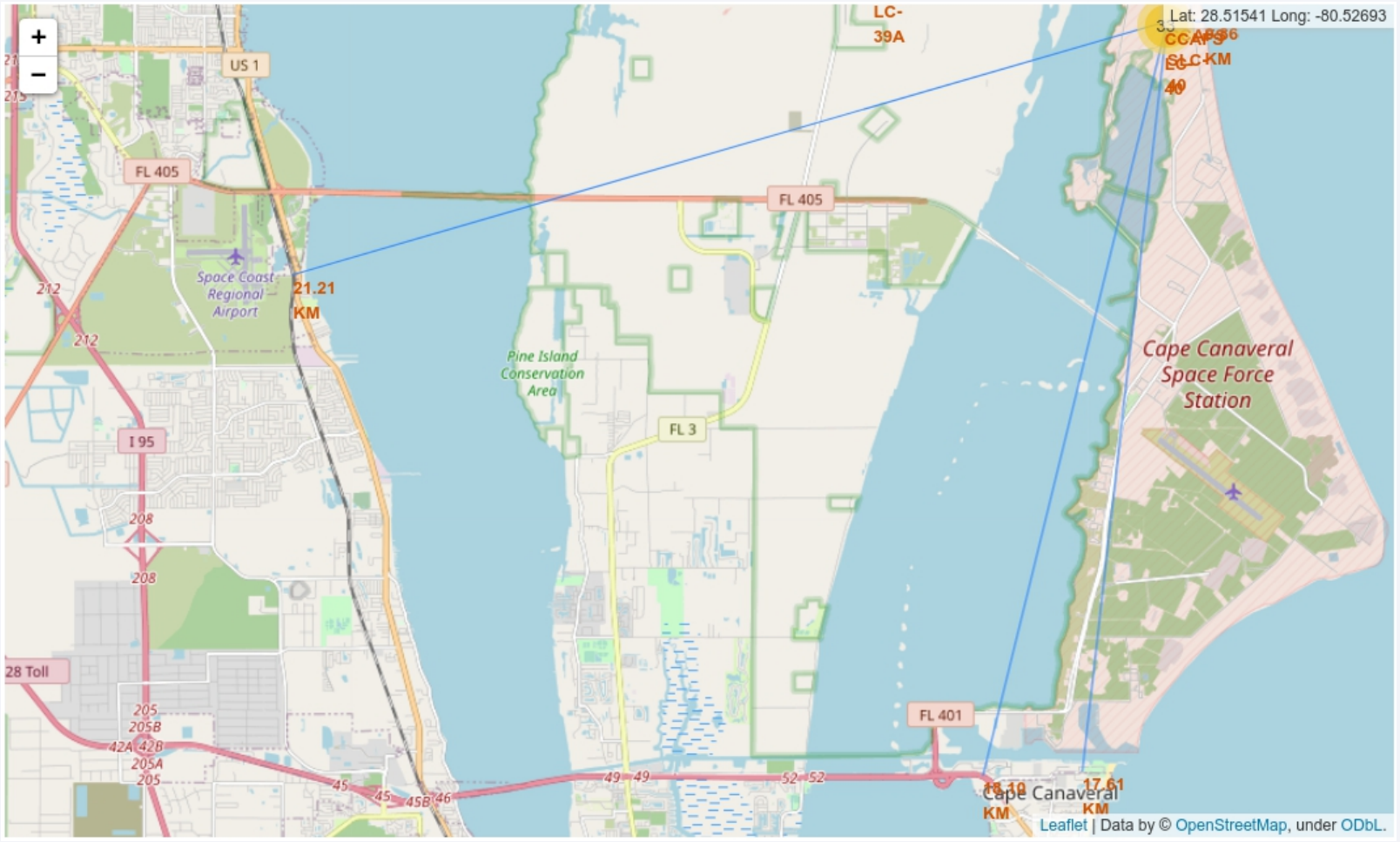
California launch site

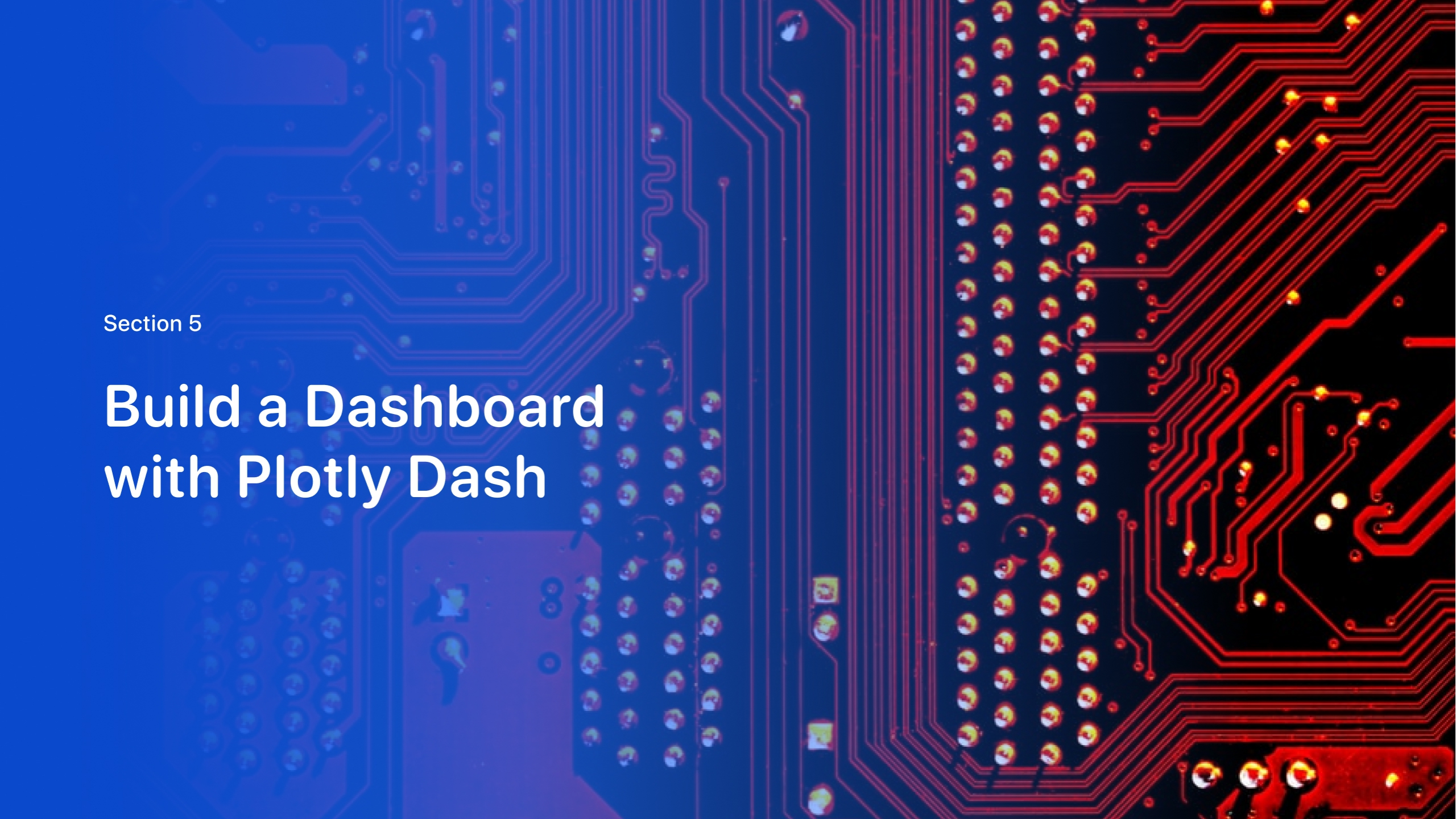
Red marker - failures
Green maeker - success

Closest points

Closest:

- railway - 21.21 km
- city - 17.61 km
- highway - 18.1 km
- coastline - 0.86 km

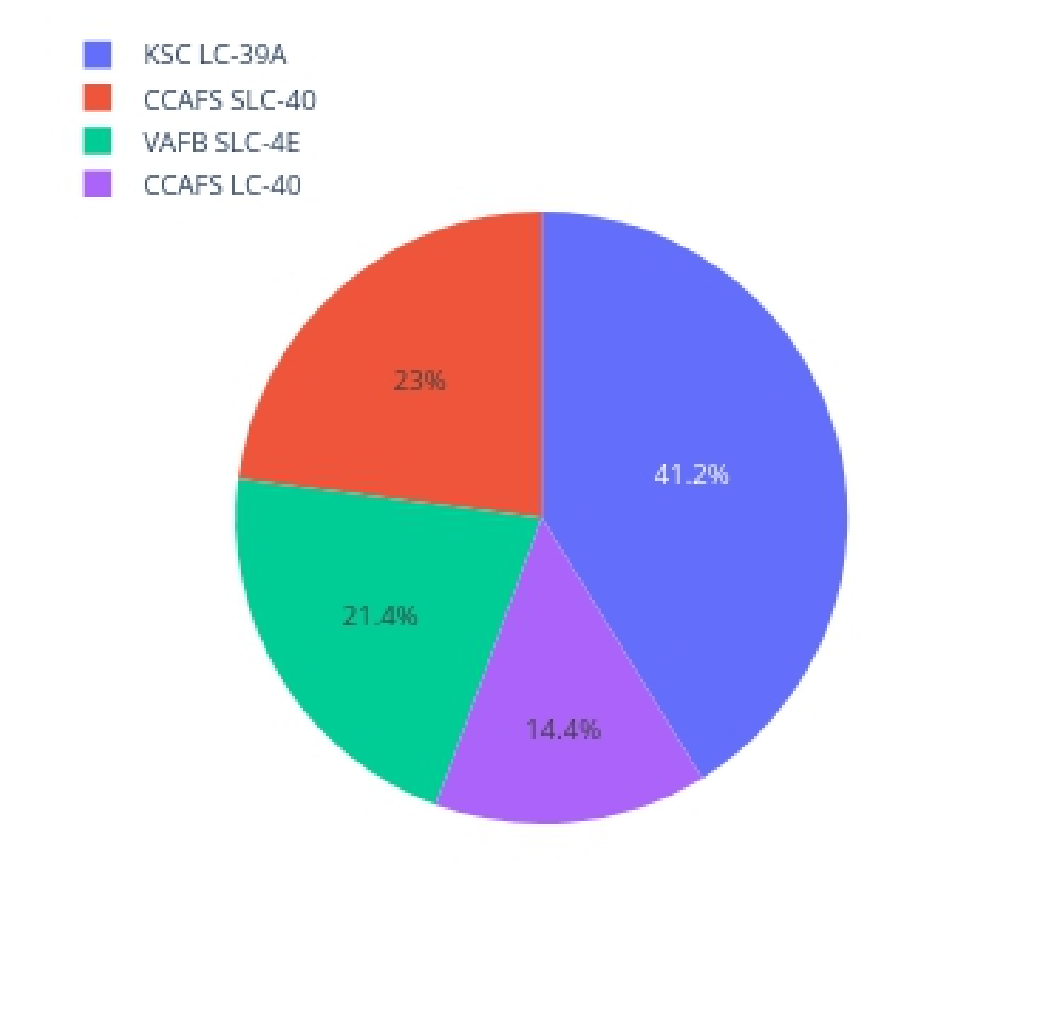




Section 5

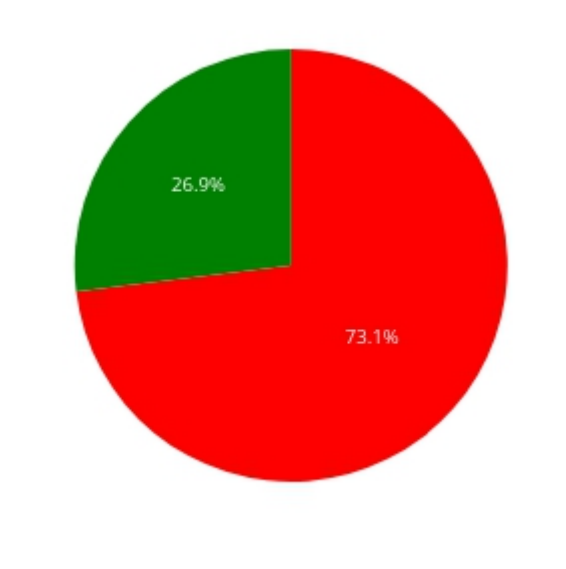
Build a Dashboard with Plotly Dash

SpaceX Launch Records all sites



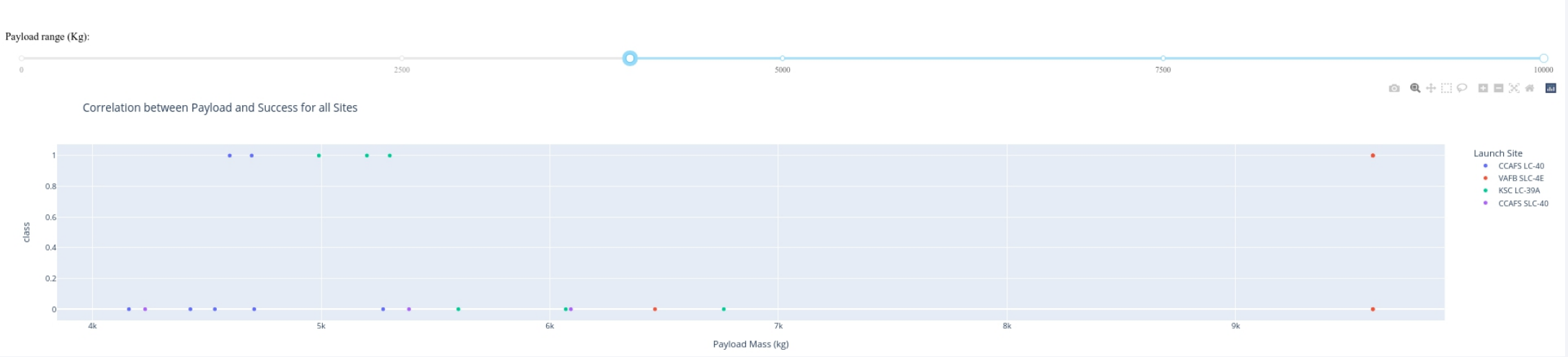
KSC LC-39A had the
most successful launches from all the sites

CCAFS LC-40



CCAFS LC-40 have has lowest success rate - 26,9%

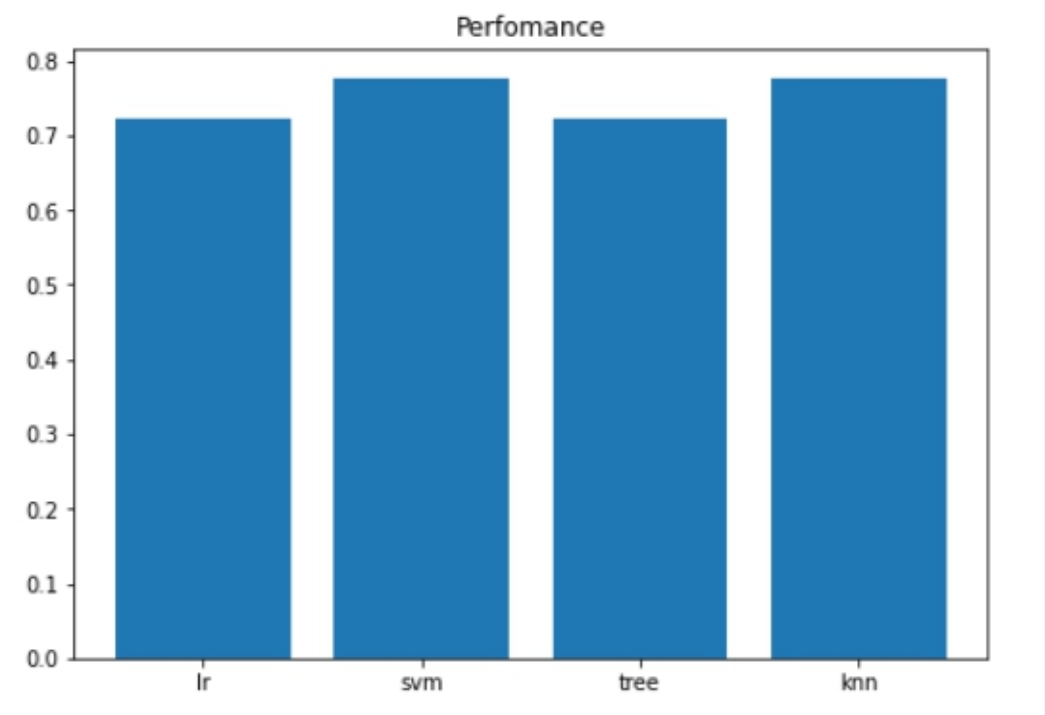
SpaceX Launch Records



Section 6

Predictive Analysis (Classification)

Classification Accuracy

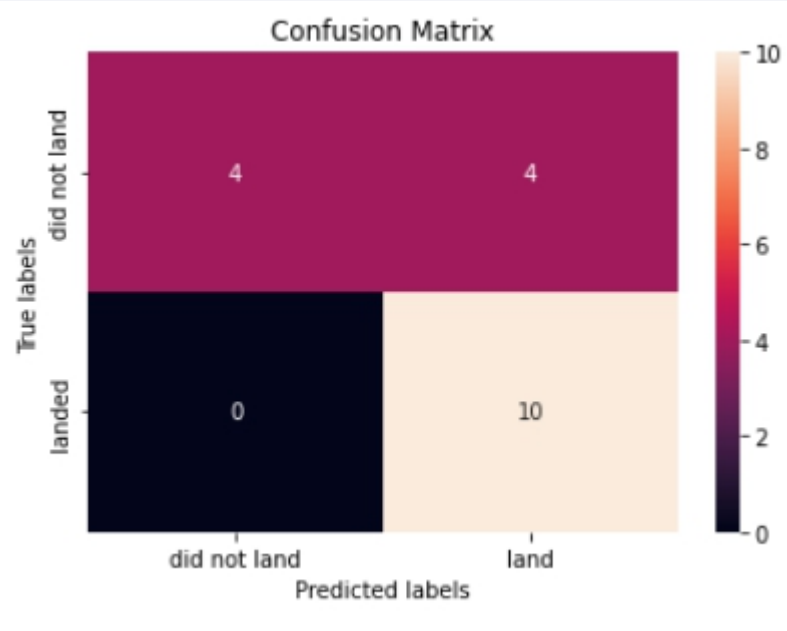


SVM and **KNN** have same test perfomance = **0.78**

KNN, parametes = {'algorithm': 'auto', 'n_neighbors': 4, 'p': 1}

SVM, parameters = {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'}

Confusion Matrix for SVM and Tree



Main problem of models - false positive

Conclusions

- The Tree Classifier Algorithm and Support Vector Machine are best for Machine Learning for this dataset.
- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches
- We can see that KSC LC-39A had the most successful launches from all the sites(least successful - CCAF LC-40)
- Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate

Thank you!