IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pieter Meijers
June 30th, 2023

Outline

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

SpaceX Falcon 9 launch data were analyzed to understand the drivers of landing success and to develop a predictive model

Executive Summary

Main findings on launch success rates:

- After the 1st successful ground landing on Dec. 22nd, 2015, yearly success rates have increased
- Successful landings correlate to
  - Orbit types SSO and VLEO (>80% success rates)
  - Payloads in 2.000-5.500 kg range
  - FT and B4 boosters
  - Site KSC LC-39A (77% success rate)
- All 4 SpaceX launch sites are all US based, situated near railroads and away from populated areas
- Future landing outcomes can be best predicted using a “tree” classification model, with an accuracy of 89%, although the many ‘false positives’ remain an issue

Methodologies

- Data collection through a REST API and webscraping
- Data were exploration
  - Visually by plotting various launch parameters against each other
  - SQL-queries on the data to find specific answers
  - A geographical analysis of Launch Sites using Folium charts
  - An interactive dashboard
- Optimizing and comparing classification models

Data analysis can help calculate SpaceX launch cost by finding drivers of launch success and building a predictive model

Introduction

SpaceX is one of few private companies who launch and operate spacecraft. SpaceX has gained worldwide attention for a series of historic milestones, including being the first and only one to return a spacecraft from low-earth orbit.

Being able to return the spacecraft and reuse its first stage, allows SpaceX to operate on a much lower price point than its competitors with single-use rockets.

- SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars
- Competitors providers cost upward of 165 million dollars each

However, not all SpaceX launches result in a successful return. For a price cost estimation, a better understanding and prediction is needed of future landing outcomes.

This problem can be addressing though data analysis of historic launch data with the aim to:

- Identify key relations between success rate and parameters such as launch site, payload, booster version, orbits, etc.
- Develop a classification model to predict future landing outcomes

Section 1

Methodology

To understand future SpaceX Falcon 9 launch results, historic data were obtained, analyzed used to create a predictive model

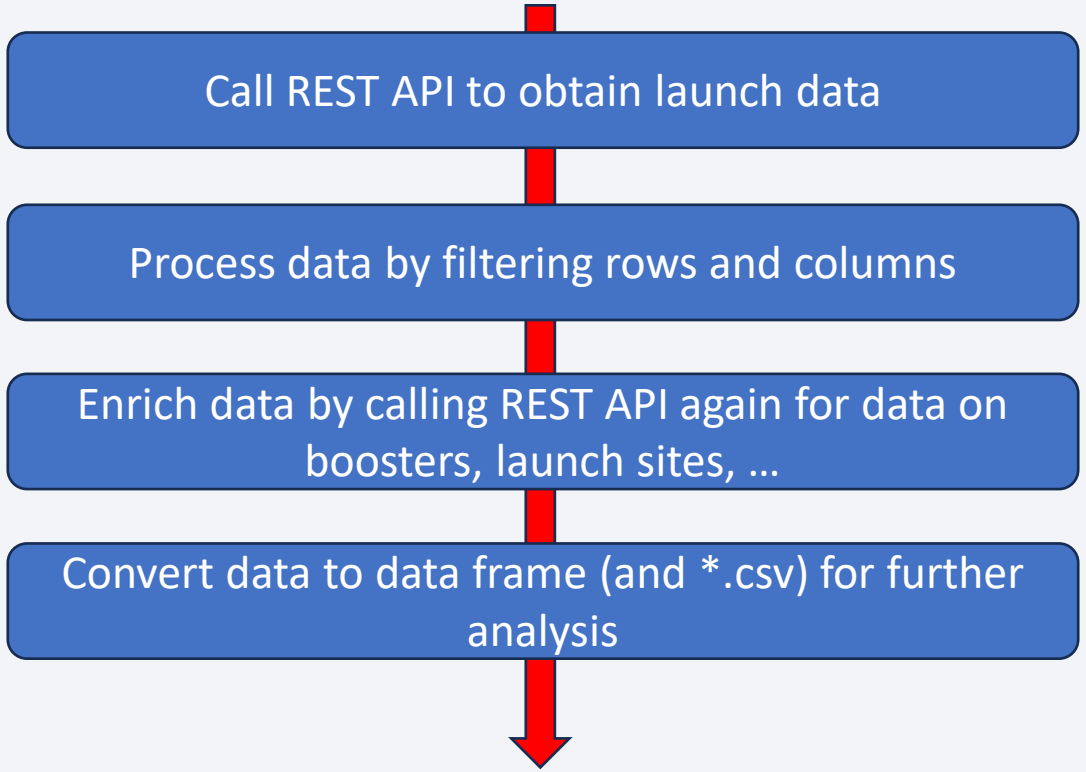
Methodology executive summary

- Data on SpaceX Falcon9 launches were collected in two ways
  - Through the SpaceX API after which a custom dataframe is built with relevant data on Falcon 9 launches
  - By scraping from Wikipedia using BeautifulSoup and to create a dataframe
- Data wrangling help find and replace missing data, format data in the right type and get a first impression
- In the Exploratory Data Analysis (EDA)
  - Six visualizations helped understand how launch parameters correlate with success rate
  - SQL data analysis was used to gain even deeper insights
- An interactive Folium map enabled an analysis of launch site success rates
- An interactive dashboard (Flask, Dash) helped analyze relations between success, launch sites, payloads and boosters
- A predict model was created by optimizing and evaluating four classification models for accuracy and results

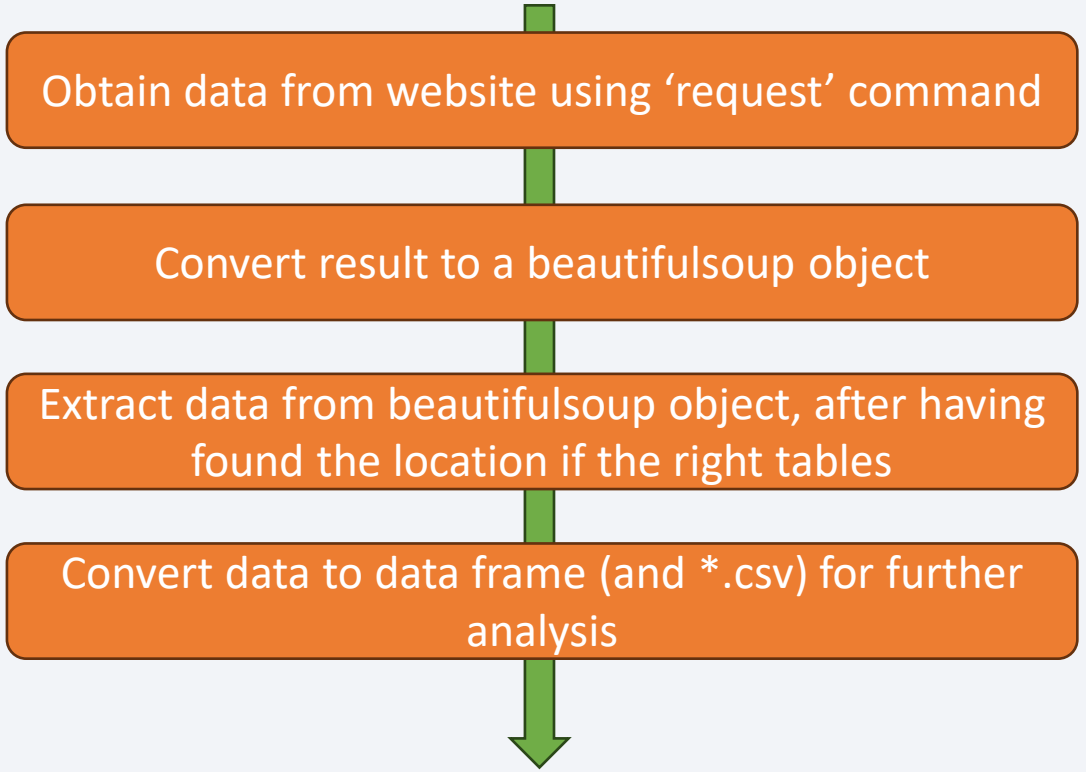
Data were obtained from the SpaceX API and by scraping the SpaceX Wikipedia page

High level data collection methods

SpaceX API



Webscraping



After obtaining launch data from the SpaceX API, a custom dataframe is built with relevant data on Falcon 9 launches

Data Collection – SpaceX API

Get data from the SpaceX API and store them in a dataframe

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
data = pd. json_normalize(response. json())
```

Limit data to relevant columns, launches with a single booster type and a single payload type

```
data = data[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]
data = data[data['cores'].map(len)==1]
data = data[data['payloads'].map(len)==1]
```

Enrich the data by obtaining details on boosters, launch sites, payloads and core data

```
getBoosterVersion(data)
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

Limit the data to launches before 13-11-2020 of type Falcon 9

```
data = data[data['date'] <= datetime.date(2020, 11, 13)]
data_falcon9 = df[df['BoosterVersion'] !='Falcon 1']
```

Replace IDs for boosters.. With their real names using custom functions that call the API

Replace missing values with the mean()

```
data_falcon9["PayloadMass"].fillna(meanPayload, inplace=True)
```

For details see https://github.com/nerdweek/SpaceX_capstone/blob/e0fddc01d70dcb4dbf9500c7e3a70e2ac047e50b/jupyter-labs-spacex-data-collection-api.ipynb

Data on SpaceX Falcon 9 launches were scraped from Wikipedia using BeautifulSoup and used to create a dataframe

Data Collection - Webscraping

Create a BeautifulSoup object from the data on a (static) WikiPedia page in SpaceX

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_F
response = requests.get(static_url)
soup = BeautifulSoup(response.content, "html.parser")
```

Find all column headers from the third table on the page

```
headers = first_launch_table.findAll("th", scope = 'col')
for header in headers:
    name = extract_column_from_header(header)
    if name is not None and len(name) > 0:
        column_names.append(name)
```

Create a dictionary with keys for each relevant parameter (column) to capture

```
launch_dict= dict.fromkeys(column_names)

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

# Let's initial the launch_dict with each
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
```

Loop through the soup object, find ‘th’ elements and add them to relevant parts of the dictionary

Convert dictionary to a data frame

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

For details see https://github.com/nerdweek/SpaceX_capstone/blob/e0fddc01d70dcb4dbf9500c7e3a70e2ac047e50b/jupyter-labs-webscraping.ipynb

Data are analyzed to get a first impression of their characteristics and content

Data Wrangling

Using data from earlier analysis, calculate the percentage of missing values per columns

```
df.isnull().sum()/df.count()*100
```

Identify columns that numerical and categorical

```
df.dtypes
```

Calculate the number of launches per site

```
df['Orbit'].value_counts()
```

Calculate the number and occurrence of mission outcome of the orbits

```
landing_outcomes = df['Outcome'].value_counts()
print(len(landing_outcomes))
landing_outcomes
```

Create a landing outcome label from Outcome column

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
landing_class = [0 if (outcome in bad_outcomes)
    else 1 for outcome in df['Outcome']]
```

For details see https://github.com/nerdweek/SpaceX_capstone/blob/e0fddc01d70dcb4dbf9500c7e3a70e2ac047e50b/labs-jupyter-spacex-Data%20wrangling.ipynb

In the Exploratory Data Analysis (EDA), six visualizations helped understand how launch parameters correlate with success rate

EDA with Data Visualization

SpaceX launch data were explored data visualizations of relationships between:

- Flight Number and Launch Site
- Payload and Launch Site
- Success rate of each orbit type
- Flight number and Orbit type
- Payload and Orbit type
- Launch success and year (yearly trend)

For details on the analysis, see section 2 and
https://github.com/nerdweek/SpaceX_capstone/blob/60ce32cb51d9319f95dbc95ab1b0ed6a6cf0b74c/jupyter-labs-eda-dataviz.ipynb

Exploratory data analysis (EDA) also included SQL data analysis, to gain even deeper insights

EDA with SQL

SpaceX launch data explorations using SQL included amongst others:

- List of unique launch sites names
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- Date of first successful landing outcome in ground pad
- Names of boosters with successful launches in drone ship and have payload mass between 4000 and 6000
- Total number of successful and failure mission outcomes
- Names of the booster versions which have carried the maximum payload mass.
- List of month names, failure landing outcomes in drone ship, booster versions, launch site for months in 2015
- Successful landing outcomes between the 04-06-2010 and 20-03-2017 ranked in descending order

For details on the analysis, see section 2 and :
https://github.com/nerdweek/SpaceX_capstone/blob/82152f95f7ec97394f2fb2d1beae3eddb69cb757/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Launch site and their success rate were analyzed in detail using interactive Folium maps

Analysis using an Interactive Map with Folium

Main findings:

- All four SpaceX launch sites are in the US, one on the US west coast, three collocated on east coast
- Launches from KSC LC-39A (west coast) have the best overall success rate
- Site CCAFS LC-40 has been most utilized
- All Launch site are located near railroad access for easy of transport and near a coastline and away from cities and highways for safety

Approach & Methods

- Launch site were plotted on a Folium world by plotting a marker and label using their LoLa-ccordinates
- Information on total launches was added using a MarkerCluster. On zooming in, this also showed all successful launches in green makers and failures in red
- After retrieving the LoLa coordinates of proximities such as railroad, coastline, highway and cities, the distance was calculated using the haversine formula. Then a line between the site and proximity was drawn with a label showing the distance

For details on the analysis, see section 3 and
https://github.com/nerdweek/SpaceX_capstone/blob/efdd2468ccb97fab68462304a3ebcef7d967ac3a/lab_jupyter_launch_site_location.ipynb

An interactive dashboard (Flask, Dash) helped analyze relations between outcome, launch sites, payloads and booster types

Build a Dashboard with Flask and Dash

Main findings:

- Site KSC LC-39A contributes highly to overall launch success
  - It has had the most successful SpaceX launches in number
  - It has a 77% success rate
- Highest success rates are achieved with
  - Payloads in 2.000-5.500 kg range
  - FT and B4 boosters

Approach & Methods

An interactive dashboard was created with Flask and Dash, that

- Showed number of successful launches per site as a pie chart
- Allowed a deep dive per site, by selecting this site in a pulldown menu. The pie chart then showed success vs. failure of the particular site
- Showed an interactice scatter plot with individual launches by success/failure and their payload. Using color, various booster types were distinguished
- Allowed for deep dive analysis by setting the payload range using a slider and by toggling booster versions

For details on the analysis, see section 4 and
https://github.com/nerdweek/SpaceX_capstone/blob/a4baac80736b52a94c18557659802ca65f3e0d3e/mod10week03_visualization.ipynb

A model to predict launch outcomes was created by evaluating four classification models and optimizing the for best results

Predictive Analysis (Classification)

Main findings:

- Four classification models were considered:
  - Linear Regression (LR)
  - Support Vector Machine (SVM)
  - Tree
  - K nearest neighbors (KNN)
- The Tree classification model is able to obtain highest accuracy
- Accuracy of other models (Linear Regression, Support Vector Machine, K nearest neighbors) were comparable
- The Tree models still has an issue with the relatively high number of false positives. This is similar to other models

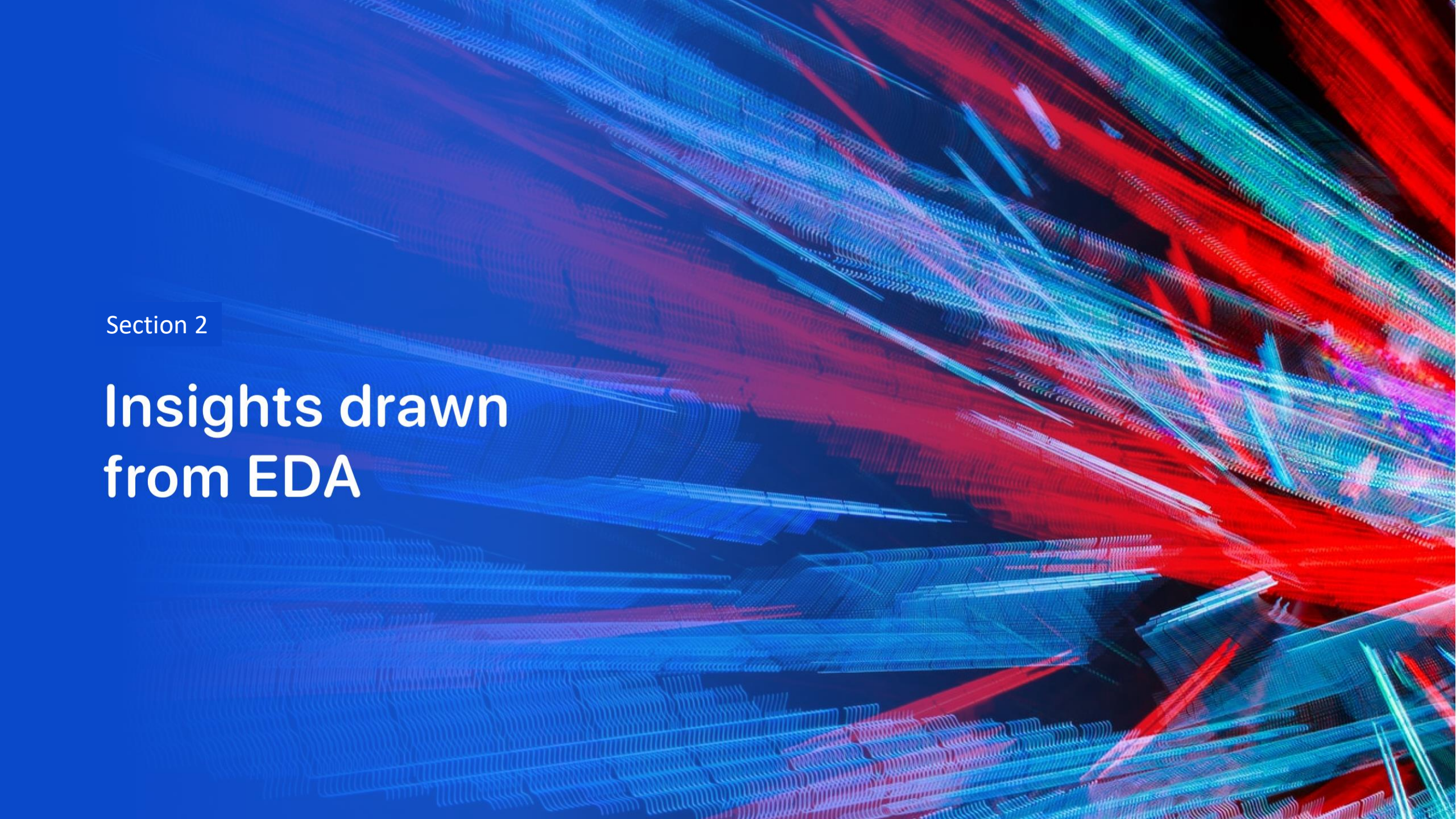
Approach & Methods

- Import launch data and normalize values
- Split data in a training data (80%) and test data (20%)
- For each of for models (See list on the left):
  - Perform a GridSearch to optimize model parameters for accuracy
  - Calculate accuracy
  - Plot and analyze confusion matrix
- Compare results for all four models

For details on the analysis, see section 5 and https://github.com/nerdweek/SpaceX_capstone/blob/7eb03362e92337b52f91a77f8438ff70b5c27846/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

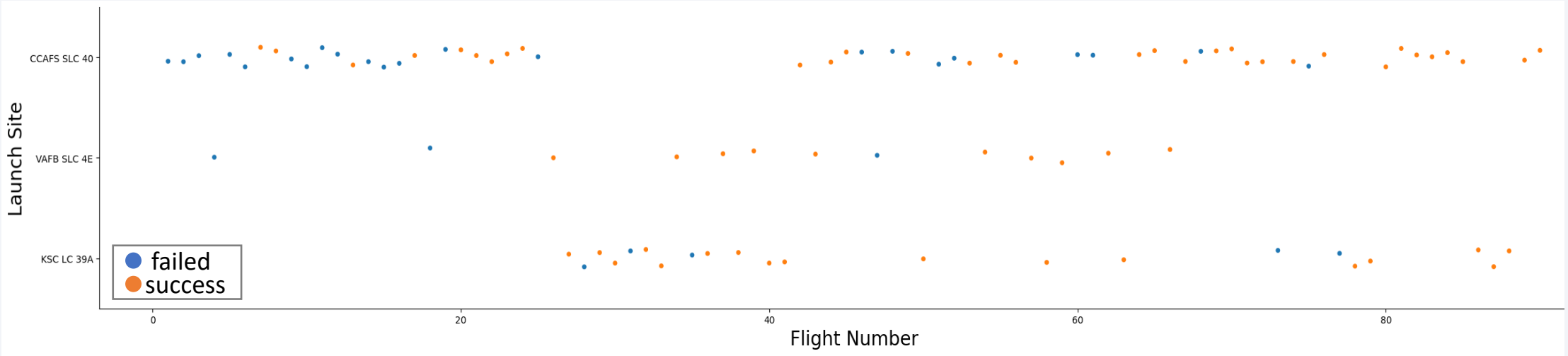
Results

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


Section 2
Insights drawn from EDA

Launch success rates increase as more launches are made, both on a macro level and per launch site

Flight Number vs. Launch Site

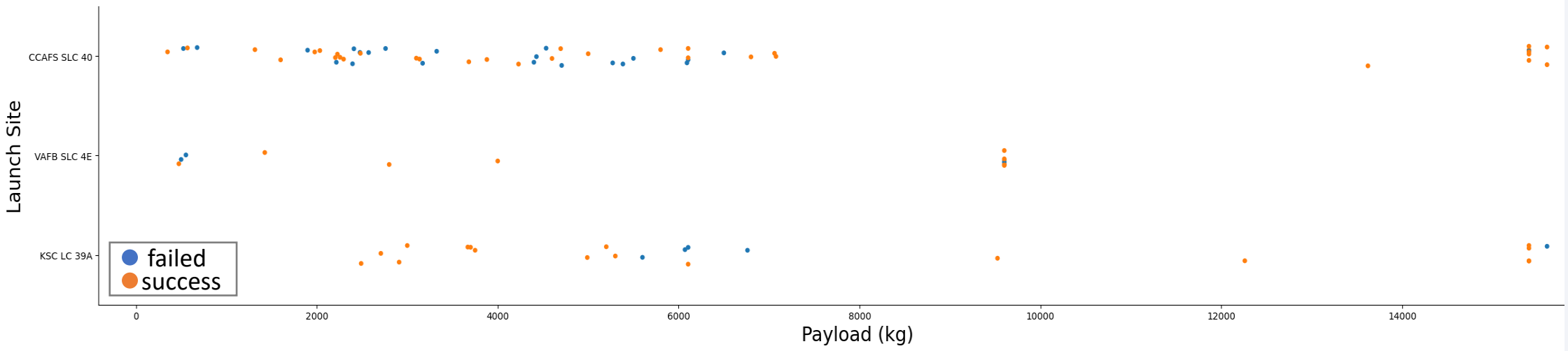


First insights:

- Success Rate increases as flights are built up, this also seems to be the case per launch site
- Variations in success rates per launch site are less obvious.
- The seemingly higher success rate of site KSC LS 39A is probably caused by it's later start (with higher flight numbers) compared to other sites

Launch success rates does not correlate strongly with payload

Payload vs. Launch Site

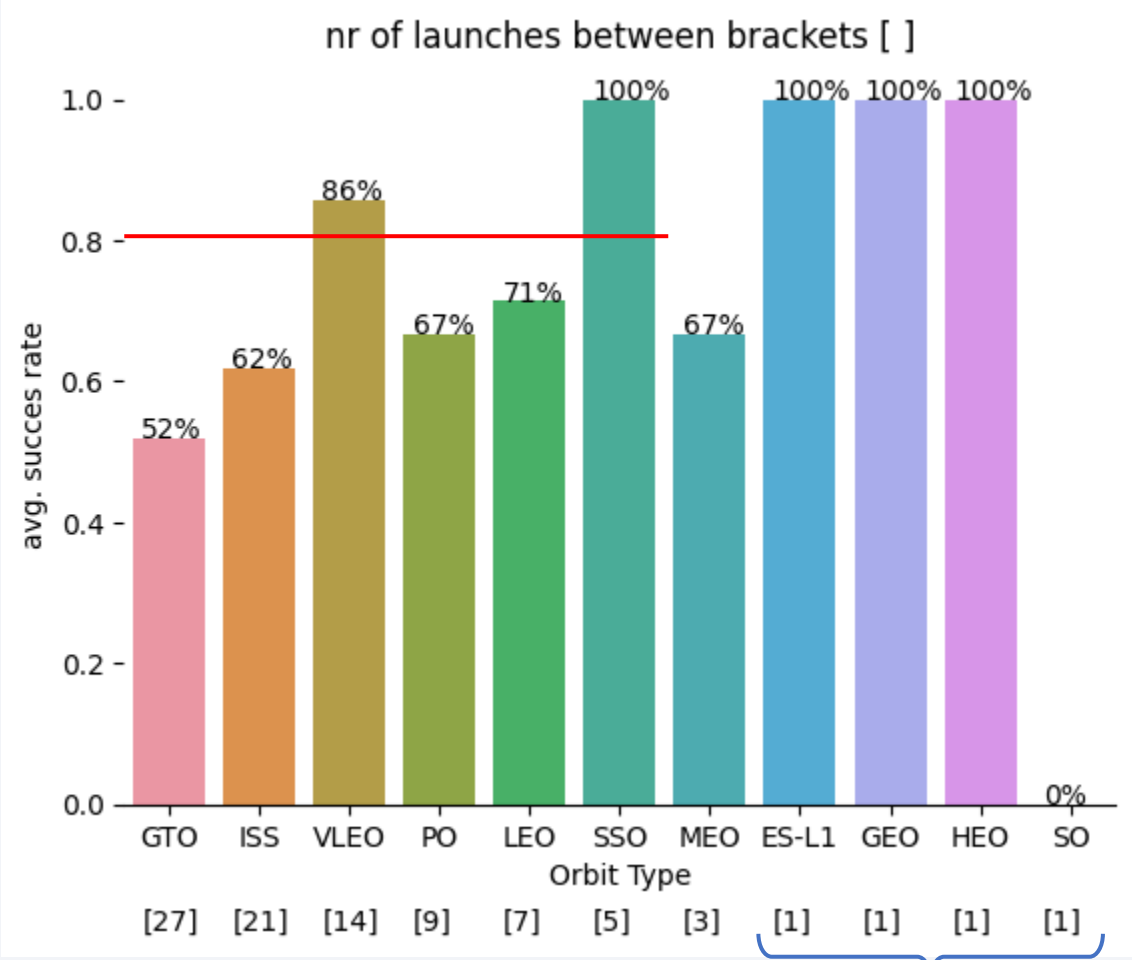


First insights:

- For the CCAPS SLC-40 site, launch success rate seems to increase with payload
- For other sites this is less clear
- It could very well be that payload has been increased as experience increased (see previous slide)

Orbit types SSO and VLEO have over 80% average success rates based on multiple launches

Average success rate per orbit type

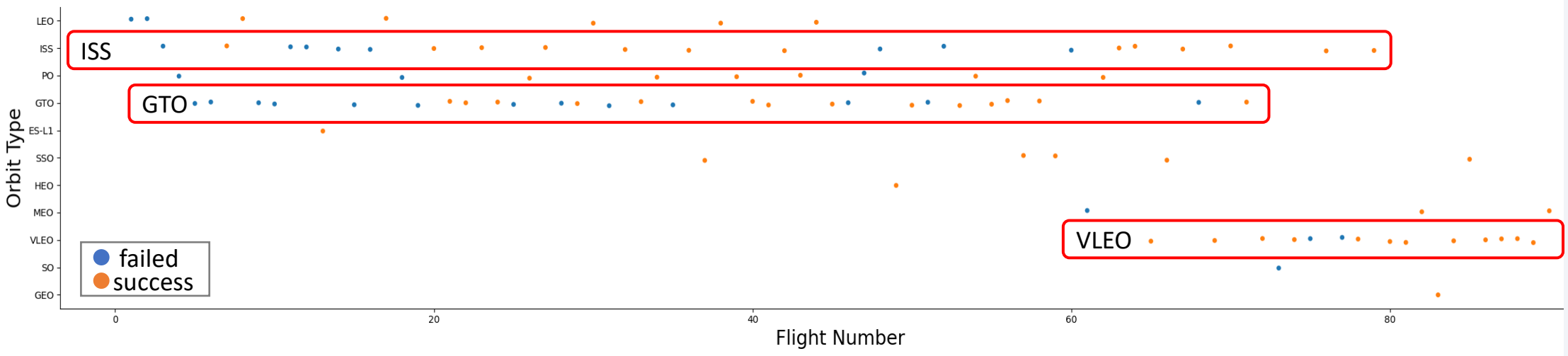


Other orbit types have either

- Numerous launches but overall lower success rates
- Only a single launch and therefore a generalized conclusion on success rates cannot be drawn

Earlier Orbit Types have been less successful than later types; In all cases, success rates increases with experience

Flight Number vs. Orbit Type

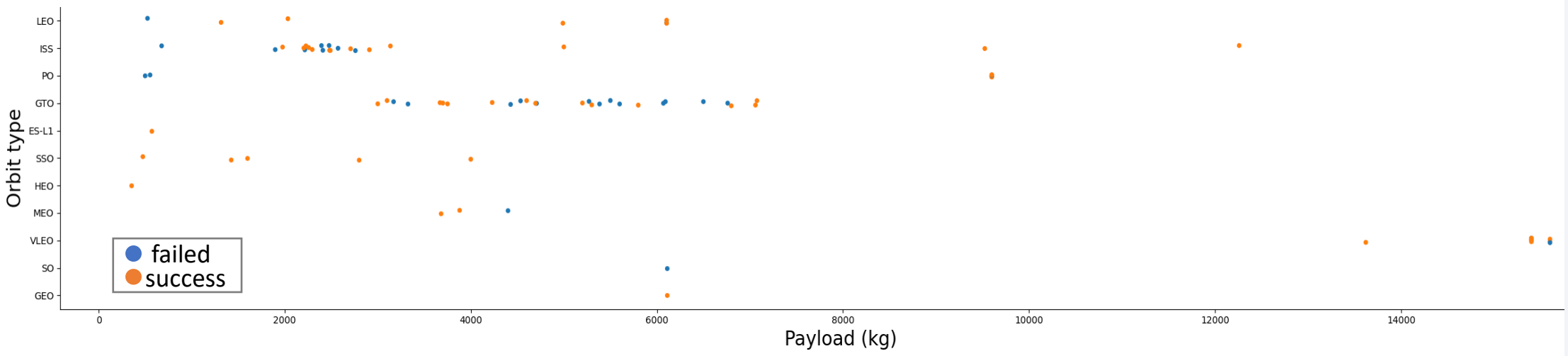


At various points in the SpaceX program, new orbit types have been introduced. For ISS, GTO and VLEO, sufficient launches have taken place to draw some conclusions

- Success rates increases as more flights are made, this also seems to be true per orbit types
- Orbit type that have been introduced earlier (ISS 62% , GTO 52%) have lower overall success rates than those introduced later in the program (VLEO 86%)

Launch success rates does not correlate strongly with payload; Most payloads range between 2000 and 7000 kg

Payload vs. Orbit Type

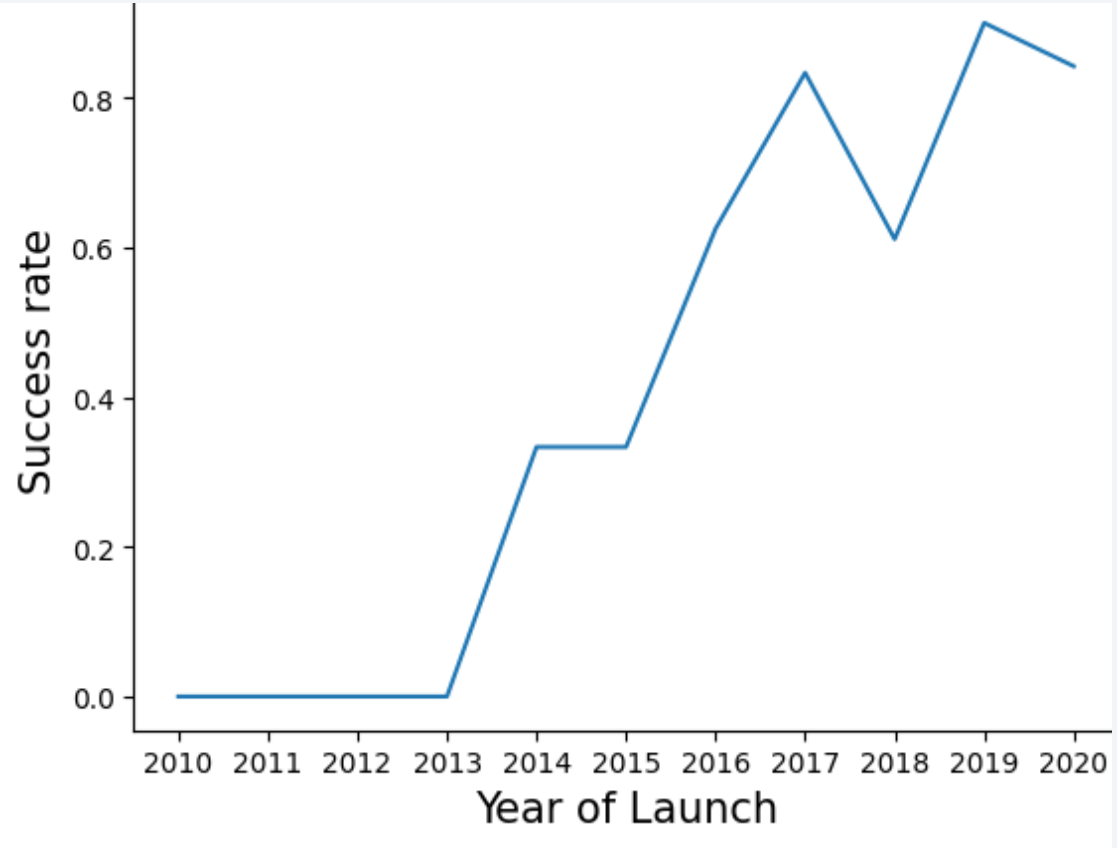


First insights:

- For some orbit type, success rate increases with payload: LEO, ISS,
- For some orbit type, success rate decreases with payload: GTO
- For all others, the relation is less clear due to lack of data

Average launch success rate increase greatly over the years

Launch Success Yearly Trend



Please note that the number of launches also increases per year. This is consistent with the earlier conclusion that success rate increased dramatically with flight numbers

year	Class
2010	1
2012	1
2013	3
2014	6
2015	6
2016	8
2017	18
2018	18
2019	10
2020	19

Four unique launch sites have been used

All Launch Site Names

There are four distinct Launch Site Names

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

SQL query & explanation

Note the use of DISTINCT to make a list of unique values

```
SELECT DISTINCT Launch_Site
FROM SPACEXTBL
```

Details of first 5 flights from a launch sites starting with ‘CCA’

Launch Site Names Begin with 'CCA’ – details of first 5 launches

4-06-10	18:45	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
8-12-10	15:43	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS)	Success	Failure (parachute)
22-05-12	7:44	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
8-10-12	0:35	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
1-03-13	15:10	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

SQL query & explanation

Note the use of LIKE and the %-sign as a wildcard to capture all site starting with ‘CCA’

```
SELECT *
FROM SPACEXTBL
WHERE "Launch_Site" like "CCA%"
Limit 5
```

NASA boosters carried a total payload mass of 45,596 kg

Total Payload Mass

SQL query & explanation

Note the use of

- SUM to add up all the payloads
- WHERE to only do so for NASA boosters

```
SELECT SUM(PAYLOAD_MASS__KG_)
FROM 'SPACEXTBL'
WHERE Customer is "NASA (CRS)"
```

Booster version F9 v1.1 (variants) carried 2,535 kg payload on average

Average Payload Mass by F9 v1.1

SQL query & explanation

Note the use of

- AVG to average payload mass
- WHERE and LIKE to filter for F9 v1.1. This is needed since in the data F9 v1.1 is often followed by the booster serial number. For example: F9 v1.1, F9 v1.1 B1011, F9 v1.1 B1010

```
SELECT AVG(PAYLOAD_MASS__KG_)
FROM 'SPACEXTBL'
WHERE "Booster_Version" LIKE "F9 v1.1%"
```

The first launch with a successful ground landing took place on December 22nd, 2015

First Successful Ground Landing Date

SQL query & explanation

Note the use of

- MIN to find the first date
- WHERE and LIKE to filter for success, since the Landing Outcome filed often contains additional details on the success or failure

```
SELECT MIN("Date")
FROM 'SPACEXTBL'
WHERE "Landing_Outcome" Like "Success%"
```

Four boosters have had success in drone ship landing with a payload between 4,000 and 6,000 kg

Boosters with successful drone ship landing and payload between 4000 and 6000

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

SQL query & explanation

Note the use of BETWEEN to further specify the payload range

```
SELECT "Booster_Version"
FROM 'SPACEXTBL'
WHERE "Landing_Outcome" = "Success (drone ship)"
AND "PAYLOAD_MASS__KG_" BETWEEN 4000.0 AND 6000.0
```

100 of the 101 launches had a successful mission outcome

Total Number of Successful and Failure Mission Outcomes

Mission_Outcome	COUNT("Mission_Outcome")
None	0
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

SQL query & explanation

Using GROUP BY ‘mission outcome’, you have to manually aggregate “success”-outcomes:

```
SELECT "Mission_Outcome", COUNT("Mission_Outcome")
FROM 'SPACEXTBL'
GROUP BY "Mission_Outcome"
```

You can eliminate this step by using a WHERE and LIKE clause, but then you have to do the same thing for failures:

```
SELECT "Mission_Outcome", COUNT("Mission_Outcome")
FROM 'SPACEXTBL'
WHERE "Mission_Outcome" LIKE 'Success%'
```

```
SELECT "Mission_Outcome", COUNT("Mission_Outcome")
FROM 'SPACEXTBL'
WHERE "Mission_Outcome" LIKE 'Failure%'
```

12 booster versions carried the max. payload of 15,600 kg

Boosters Carried Maximum Payload

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

SQL query & explanation

Booster versions are selected where the payload mass is the max payload mass. Note that this max payload mass is not explicitly mentioned. It is included as a subquery in the WHERE clause.

```
SELECT "Booster_Version"
FROM 'SPACEXTBL'
WHERE "PAYLOAD_MASS__KG_" = (
    SELECT MAX("PAYLOAD_MASS__KG_")
    FROM 'SPACEXTBL')
GROUP BY "Booster_Version"
```

To find the max payload, use the query below

```
SELECT MAX("PAYLOAD_MASS__KG_")
FROM 'SPACEXTBL'
```

In 2015 there where 2 failed drone ship landings in January and April respectively

2015 Launch Records

MTH	Landing_Outcome	Booster_Version	Launch_Site	Date
Jan	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015-01-10 00:00:00
Apr	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015-04-14 00:00:00

SQL query & explanation

- Main challenge is to return the Month in text, where the date column has type “date”. I solved this by using the month number to calculate an index for a string with 3 letter month names
- The other, minor, challenge is to filter on “year”, which is again not a separate column

```
SELECT substr('JanFebMarAprMayJunJulAugSepOctNovDec',
              1 + 3*strftime('%m', "Date"), -3) AS "MTH",
             "Landing_Outcome", "Booster_Version", "Launch_Site", "Date"
FROM 'SPACEXTBL'
WHERE SUBSTR("Date", 0,5) = '2015'
AND "Landing_Outcome" = "Failure (drone ship)"
```

Between 2010-06-04 and 2017-03-20, 10 of 32 launches had a successful landing outcome

Ranked landing outcomes between 2010-06-04 and 2017-03-20

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

SQL query & explanation

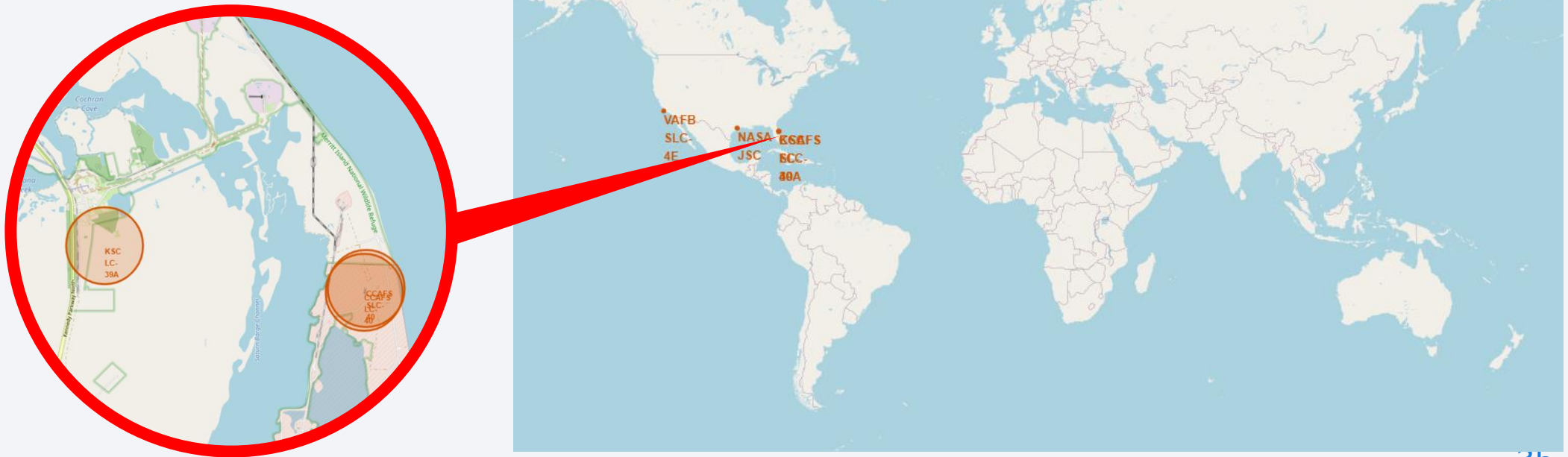
Use BETWEEN to indicate the date range

```
SELECT "Landing_Outcome", count(*)
FROM 'SPACEXTBL'
WHERE "DATE"
    BETWEEN '2010-06-04'
    AND '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY count(*) DESC;
```


Section 3
Launch Sites
Proximities Analysis

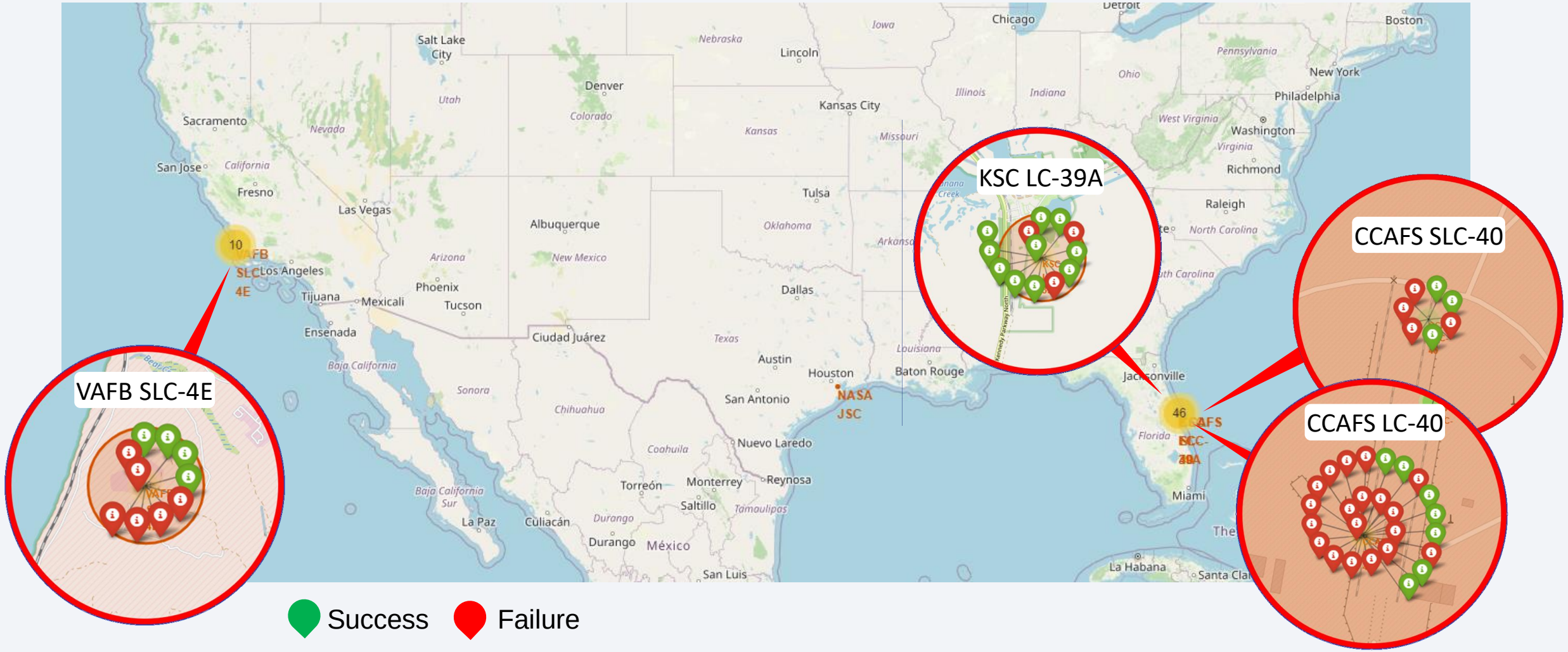
All four SpaceX launch sites are in the US, one on the US west coast, three collocated on east coast

SpaceX launch sites



Launches from KSC LC-39A have the best overall success rate; CCAFS LC-40 has been most utilized

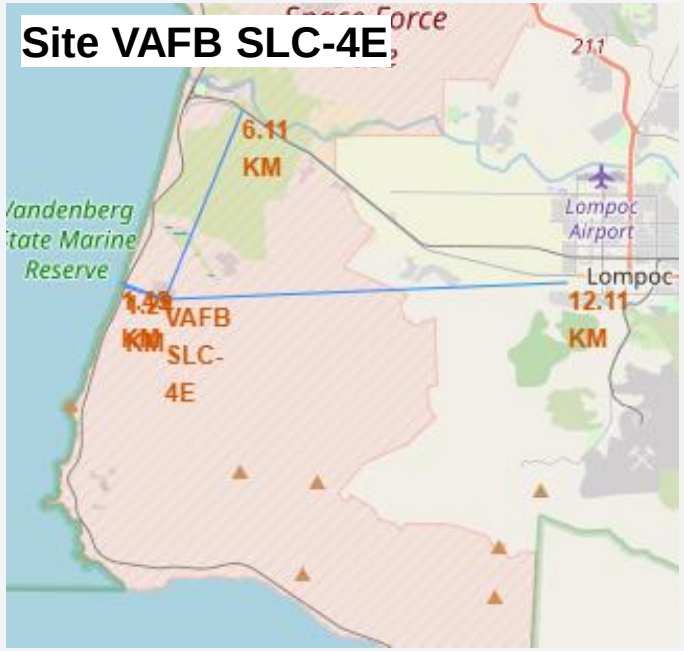
Launch outcomes per launch site

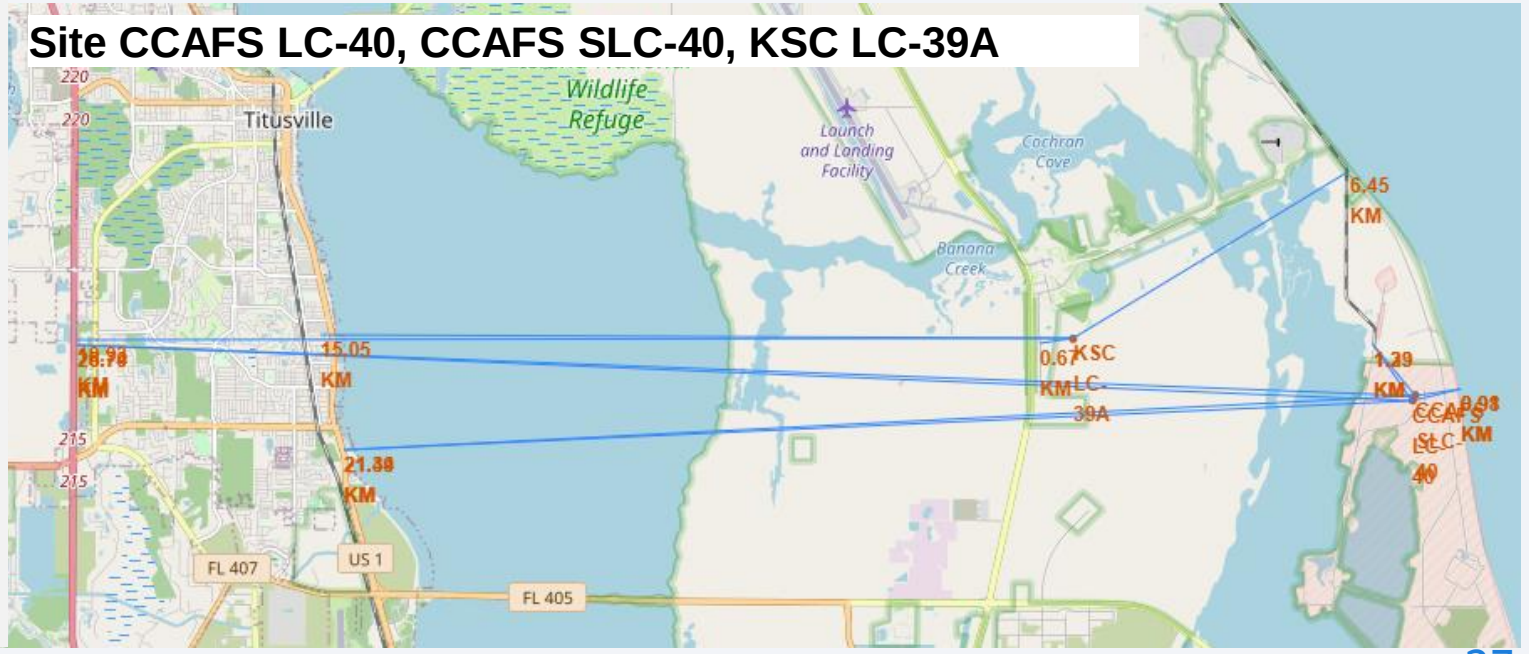


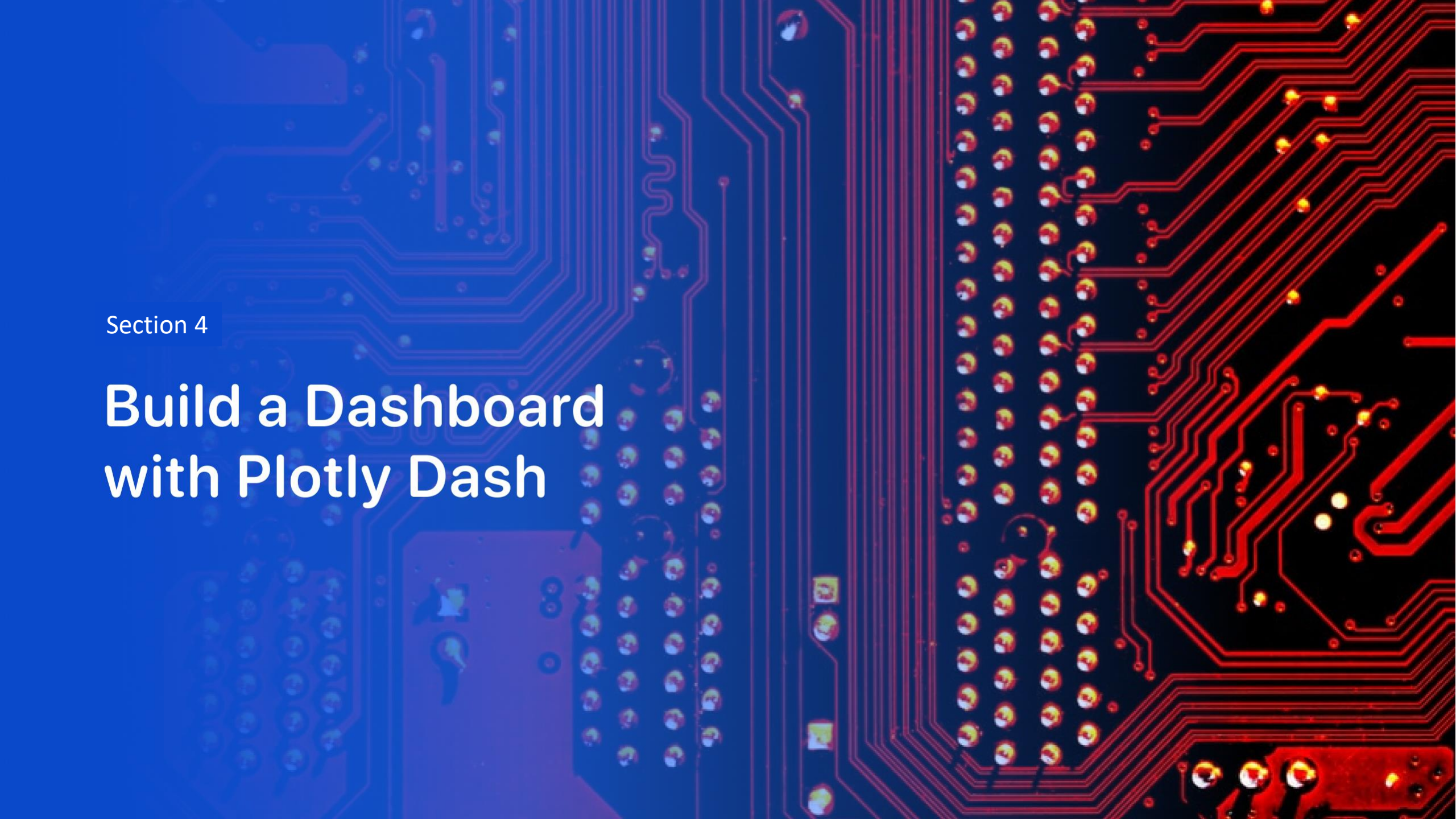
Launch site are located near railroad access and the coastline, while keeping from cities and highways

Distances between launch site nearest coastline, railroad track, highway and cities

- Launch sites are located within 2 km (on average) from railroads, for easy of transport
- Launch sites are located such they pose minimal danger to other: Near the coast (<3 km on average) and at least 18 km (on average) from cities and highways

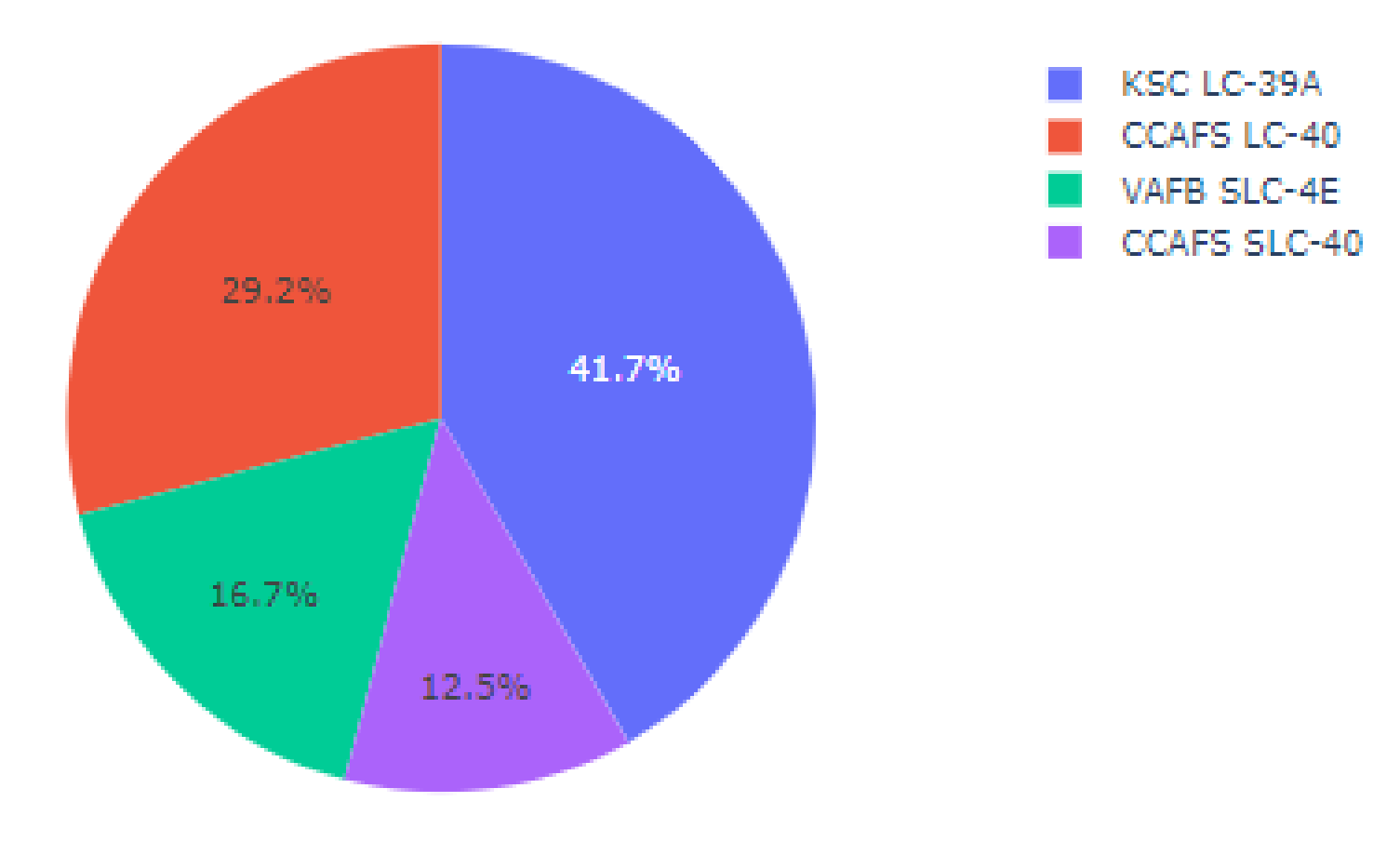





Section 4
Build a Dashboard with Plotly Dash

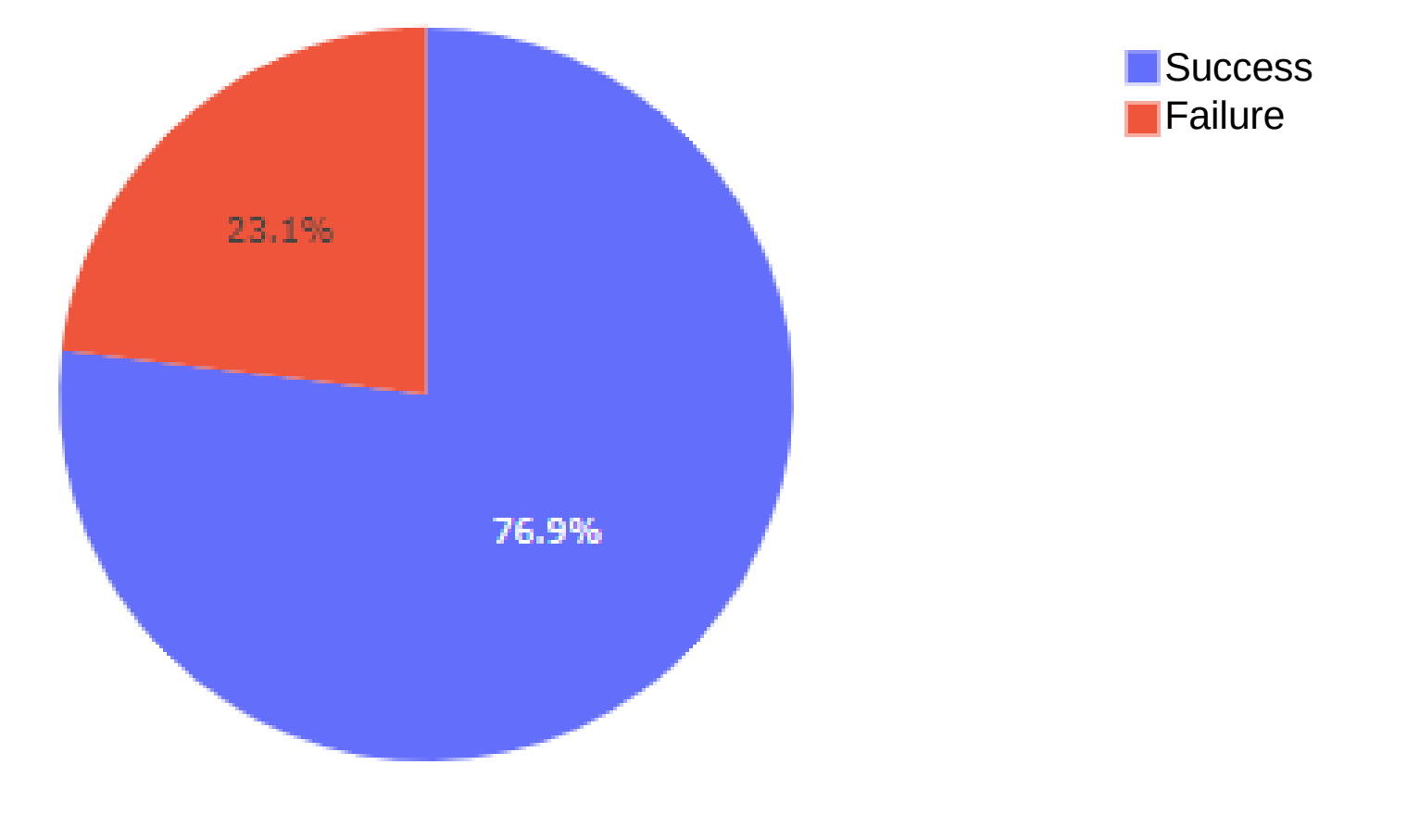
Site KSC LC-39A has had the most successful SpaceX launches

Total number of successful launches per site



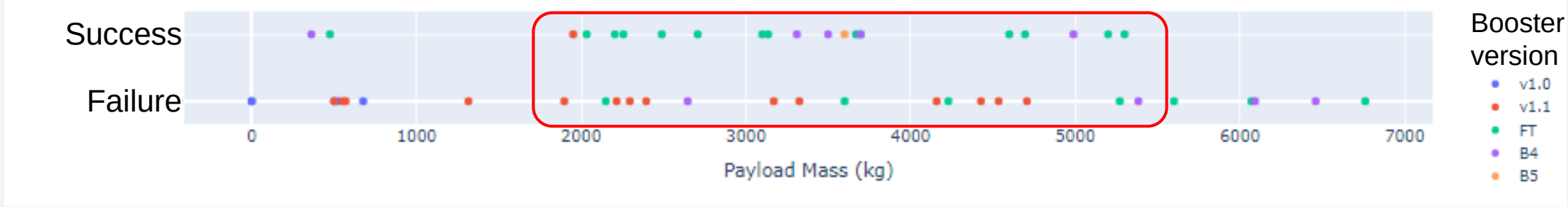
Site KSC LC-39A has a 77% success rate

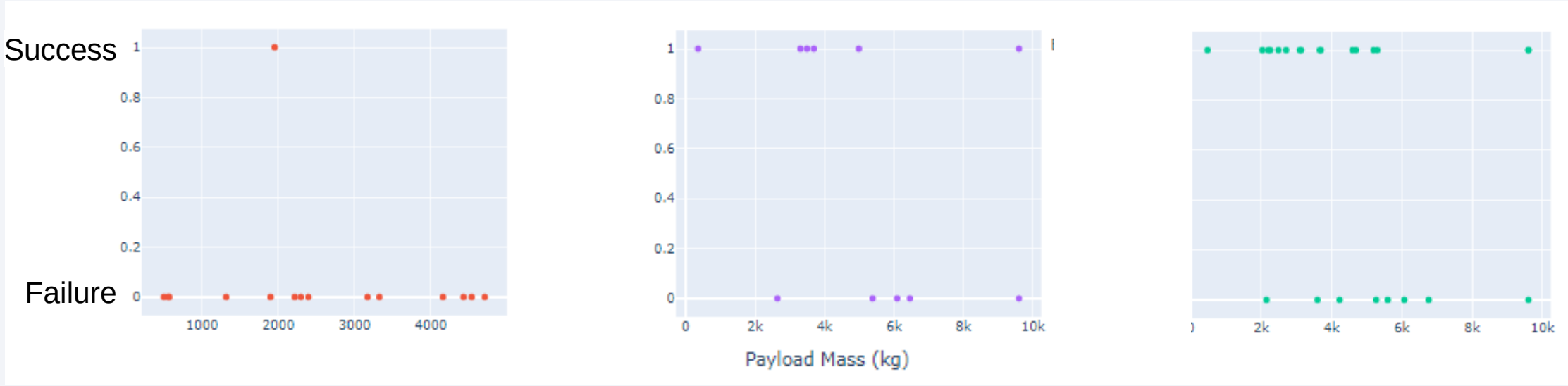
KSC LC-39A launces by outcome



Highest success rates are achieved with payloads in 2.000-5.500 kg range and the FT and B4 booster[1]

Launch outcome for different payloads an booster versions





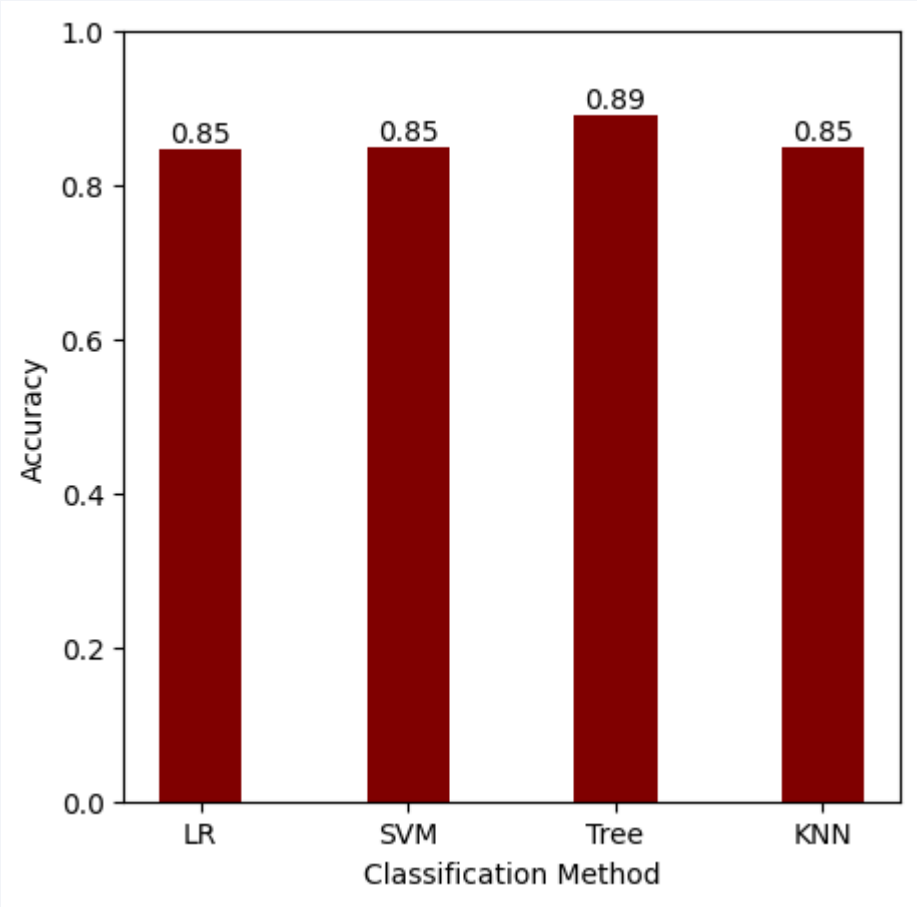
1: data for B5 and V1.1 not considered, since the were used for respectively 1 and 4 launches

Section 5

Predictive Analysis (Classification)

The Tree model has highest model accuracy of 89%; other models perform at 85%

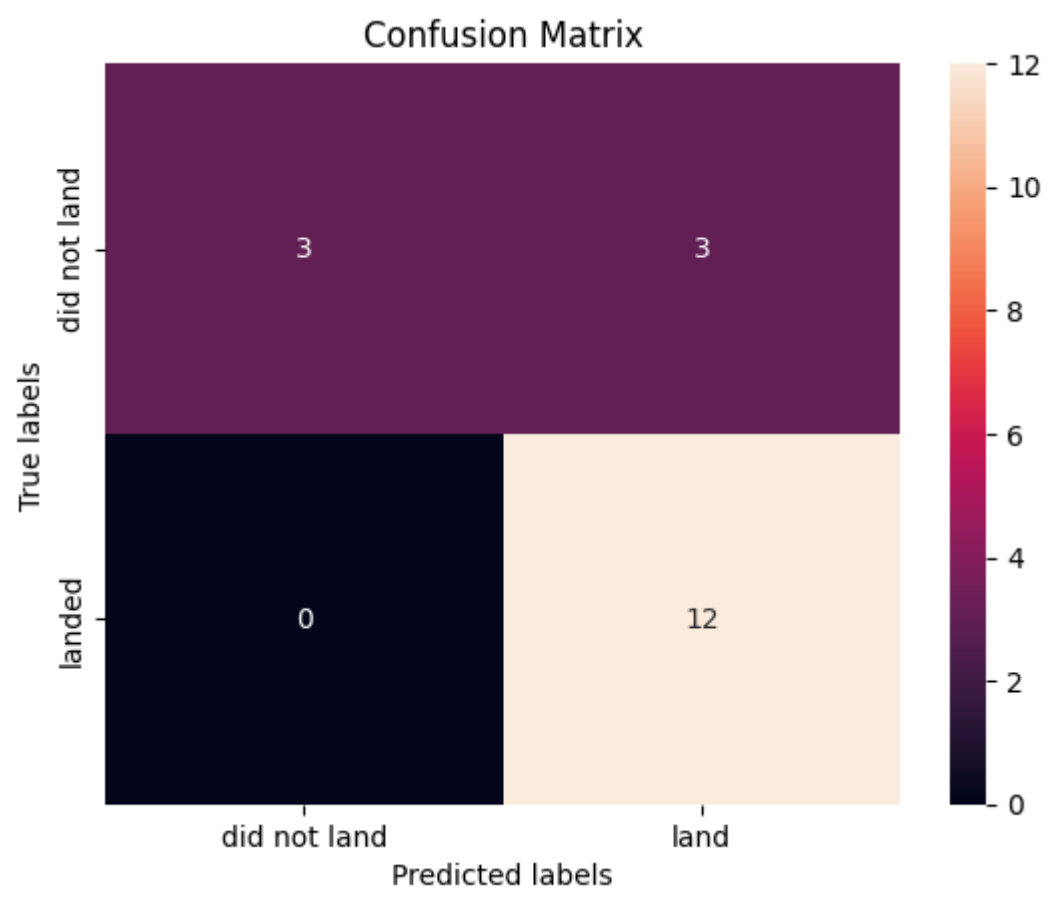
Accuracy per classification model after optimization



- Four models have been evaluated:
  - Linear Regression (LR)
  - Support Vector Machine (SVM)
  - Tree
  - K nearest neighbors (KNN)
- Parameters for each model were optimized using a GridSearch
- The model accuracy was calculated after the optimization

‘Tree’, the best performing model, still has an issue with relative high number of ‘false positives’

Confusion matrix of Tree model



- The Tree model is quite capable at predicting successful launches
- However, failed launches are classified as failures 50% of the time and as success also 50% of the time

Conclusions

- After the 1st successful ground landing on Dec. 22nd, 2015, yearly success rates have increased
- Successful landings correlate to
    - Orbit types SSO and VLEO (>80% success rates)
    - Payloads in 2.000-5.500 kg range
    - FT and B4 boosters
    - Site KSC LC-39A (77% success rate)
- All 4 SpaceX launch sites are all US based, situated near railroads and away from populated areas
- Future landing outcomes can be best predicted using a “tree” classification model, with an accuracy of 89%, although the many ‘false positives’ remain an issue

Appendix – work around to obtaining LoLa coordinates through Google Earth rather than writing then down

- When analyzing launch sites in Folium, it is needed to establish the LoLa coordinates of some proximities of each site: nearest railroad, nearest highway, nearest city, nearest coastline.
- This would result in writing down LoLa coordinates for 16 points (4 sites x 4 proximities)
- An alternative is to use data from google earth that can then be used to in python

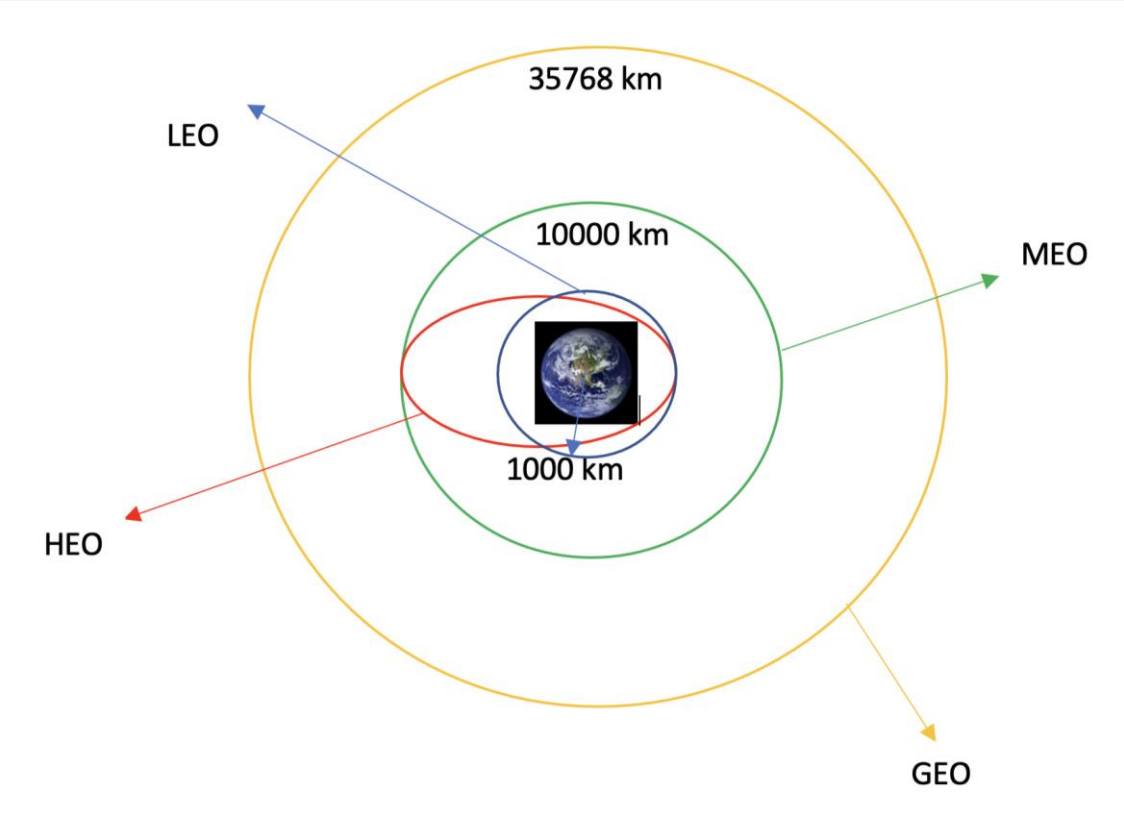
Steps:

- Go to Google earth and find a launch site
- Create a polygon starting and ending at the launch site via the proximities
- Export the polygon to a *.kml file
- Open the *.kml in a text editor or vsCode and copy the lola-coordinates


```
39                  <LinearRing>
40                      <coordinates>
41                          -80.57682456226971,28.56317239709365,0
                            -80.58530603383508,28.57203256679918,0
                            -80.79437791597293,28.54788144335043,0
                            -80.56759839892578,28.56423087844411,0
                            -80.7960388651244,28.55349638149724,0
                            -80.5762676471339,28.56216454628229,0
                            -80.57682456226971,28.56317239709365,0
42                      </coordinates>
43                  </LinearRing>
```

Appendix – LEO, HEO, MEO and GEO orbits

Orbit types



- LEO: Low Earth orbit (LEO) is an Earth-centred orbit with an altitude of 2,000 km or less. Most of the manmade objects in outer space are in LEO.
- HEO A highly elliptical orbit, is an elliptic orbit with high eccentricity, usually referring to one around Earth.
- MEO Geocentric orbits ranging in altitude from 2,000 km to just below geosynchronous orbit at 35,786 kilometers (22,236 mi). Also known as an intermediate circular orbit
- GEO It is a circular geosynchronous orbit 35,786 km (22,236 miles) above Earth's equator and following the direction of Earth's rotation [10

Thank you!