IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Alhama A. E.
January 2022

# Outline

# Executive Summary

- Methodology:
  - Data collection, data wrangling, exploratory analysis, exploratory analysis & visualization, interactive analysis with dash, predictive analysis with machine learning
- Results:
  - SpaceX launches have continuously increased the success of the first stage recovery since 2013
  - The current recovery success is about 80%
  - Major factors that affect the recovery success are: the rocket development phase & Booster version, the payload of the rocket
  - A classification machine learning model could be built and optimized to predict the first stage recovery success with an accuracy of 83%

# Introduction

Project background and context:

- Several companies offer rocket launches (space flights) for commercial use at different prices
- Among them, SpaceX offers relatively cheap ones (62 million dollar, vs 165 million dollar for other providers)
- The lower cost of the rocket launch is due to the possibility of recovering the first stage, which can be reused in the next launch

Project goal:

- Analyze data corresponding to SpaceX Launches
- Determine factors that influence a successful landing of the first stage
- Predict if a first stage will land
- Predict relative price of the launch

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection methodology:

Data collection via REST API and Web scraping

- Perform data wrangling

Identify Target information and transform into labelled data with 0 and 1 values

- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

ML models Linear Regression, Support Vector Machine, Decision Tree and K-Nearest Neighbors were built, optimized and compared

# Data Collection

- Using REST API

Datasets are collected via a HTTP GET request to the SpaceX API url

The URL https://api.spacexdata.com/v4/launches/past contains the data of past SpaceX launches and can be extracted in json format.

- Using Web Scraping from Wikipedia

HTML pages can be extracted via HTTP GET request in HTML format and afterwards parsed using BeautifulSoup objects.

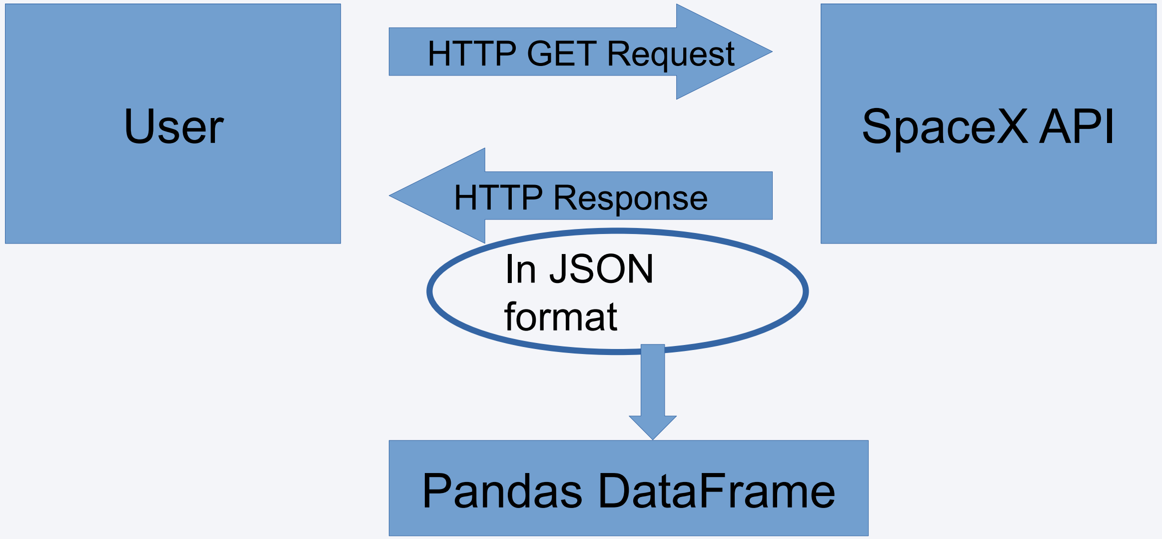
# Data Collection – SpaceX API

- REST API is used to transfer data over the internet
- The HTTP GET request is used to read data from the SpaceX API

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

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

- If the HTTP request is successful, the HTTP response shows the status code 200
- The SpaceX Server transfers the requested  data as an HTTP Response in JSON format, which can be easily transformed into a pandas DataFrame



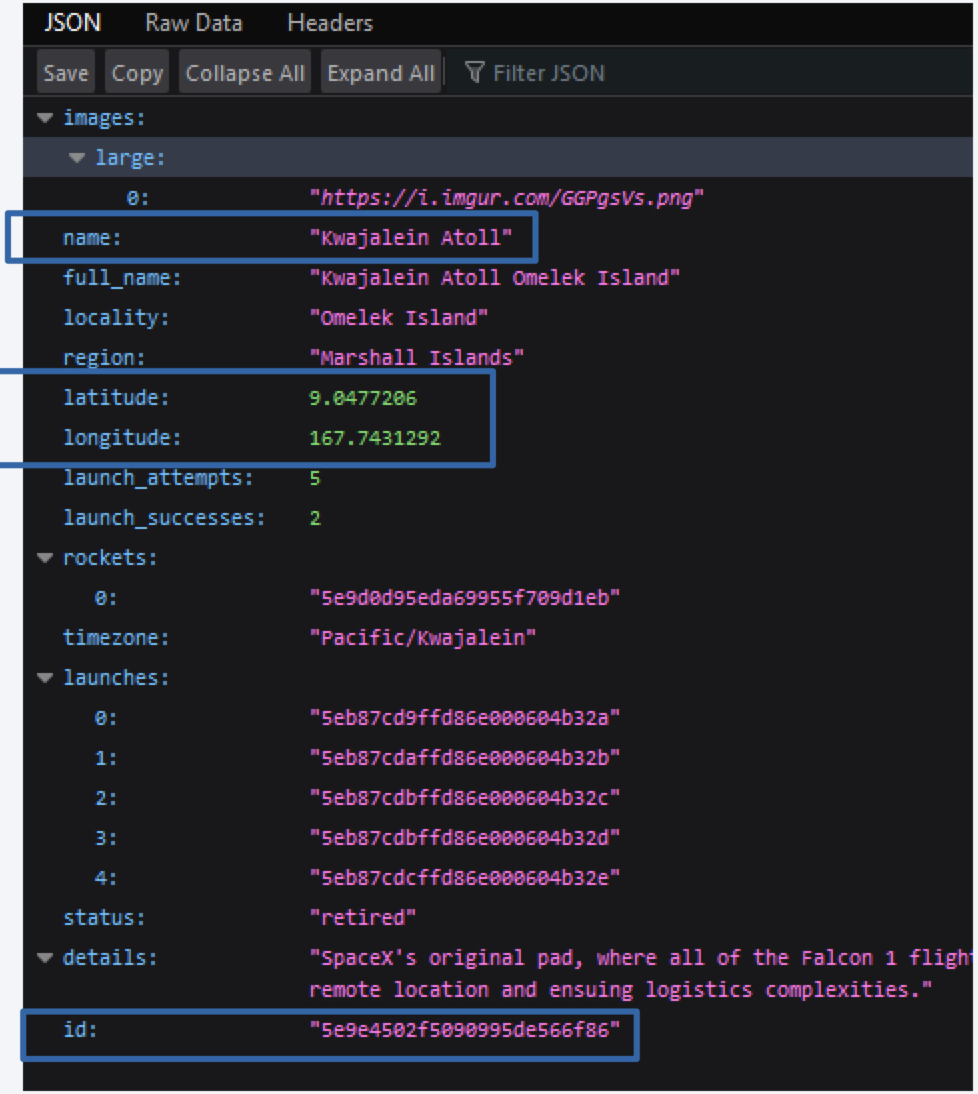

```
df = pd.json_normalize(response.json())
```



GitHub URL: https://github.com/cyesteban/capstone/blob/master/Data%20Collection%20API%20Lab.ipynb

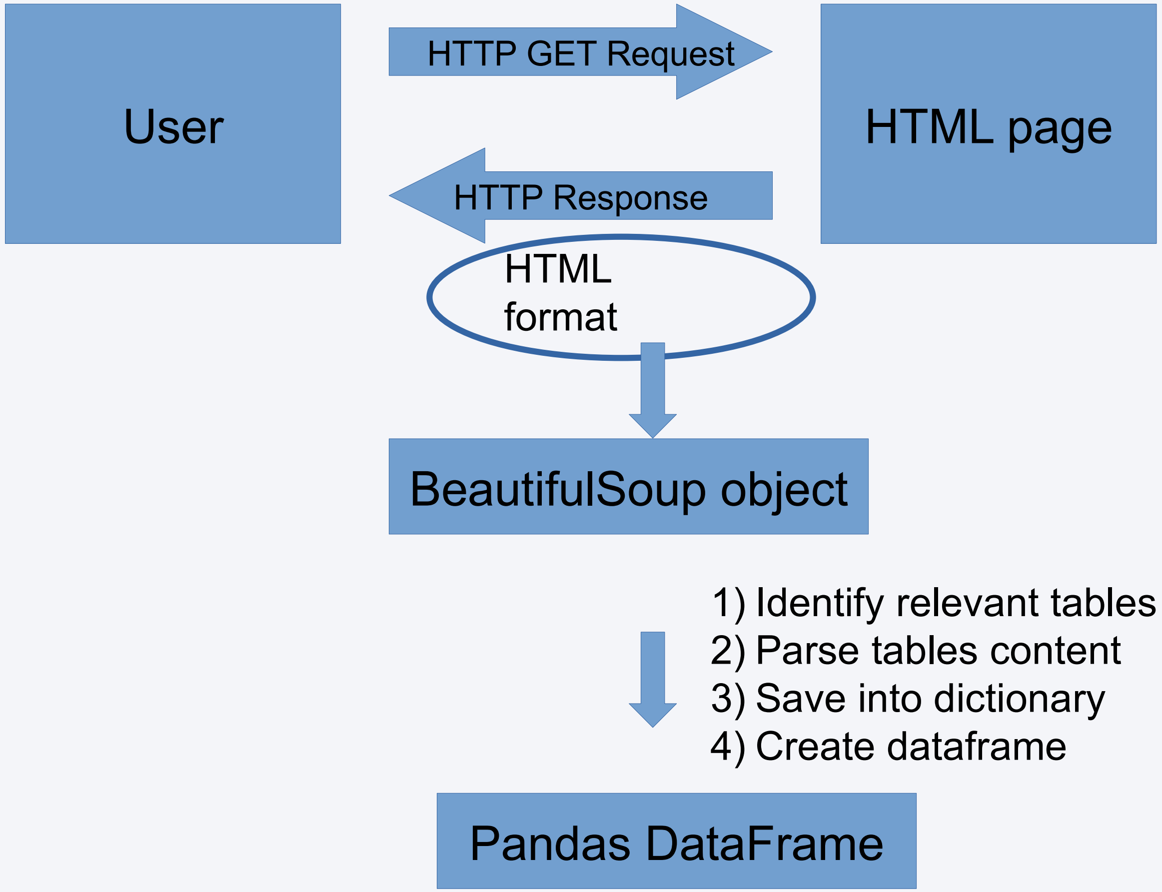
# Data Collection – SpaceX API

- Data given as an ID number under the spacex_url needs to be extracted using HTTP requests from specific urls
- Screenshot: view of the url https://api.spacexdata.com/v4/launchpads/5e9e4502f5090995de566f86/ information in json format
- This url contains information about the launch sites names and geographical coordinates
- This url is used in the function getLaunchSite
- Information about BoosterVersion, launchSite, PayloadMass etc. is collected from different urls via HTTP requests and saved to a dictionary, which is the base for the final pandas dataframe (df)
- Finally, the df is filtered to show only data of the BoosterVersion Falcon 9



# Data Collection - Scraping

- The content of webpages like Wikipedia can be collected via HTTP GET request in HTML format
- A BeautifulSoup object is created with the HTML content. The BeautifulSoup object can be searched by title, row etc
- The relevant table(s) need to be identified and parsed, and the data saved into a dictionary. Finally, a pandas DataFrame is created from the dictionary



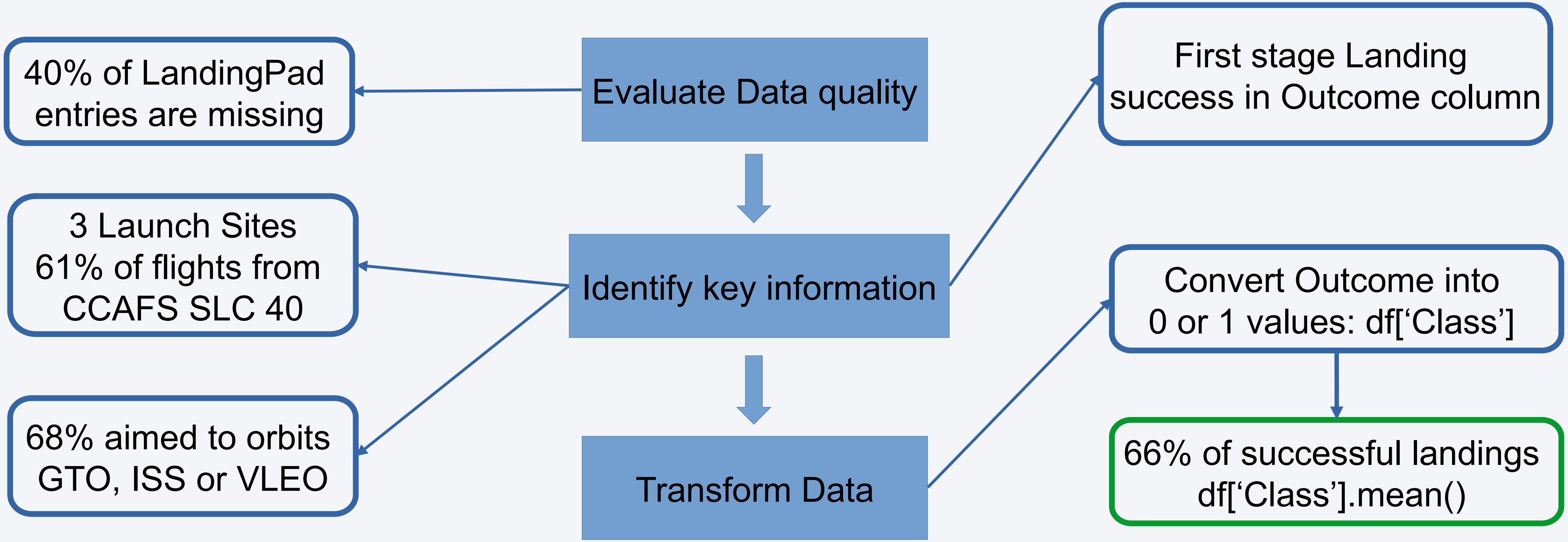



GitHub URL: https://github.com/cyesteban/capstone/blob/master/Web%20scraping.ipynb

# Data Wrangling

Purpose:

- Get important insights into data (missing values, key information)
- Transform key data into a label that can be used for exploratory analysis & ML






GitHub URL: https://github.com/cyesteban/capstone/blob/master/Data%20Wrangling.ipynb

# EDA with Data Visualization

The main variables that we have are the Flight number (equiv to time), the PayloadMass, Orbit, LaunchSite, ReusedCount (how many times the first stage has been reused)

In order to learn about possible correlations between these variables, the following charts have been plotted:

- Flight number vs Launch Site: evolution of landing success in every site over time
- Payload vs Launch Site: are some Launch Sites more suitable for heavier rockets?
- Success rate vs orbit type: does the orbit type determine the success?
- Flight number vs orbit type: does the success rate increase/decrease with time for a specific orbit?
- Payload vs orbit type: how does the payload affect success rate for different orbits?
- Landing success yearly trend: evolution of success rate over time
- Block vs Flight number and Block vs ReusedCount: how do the different development versions of the Falcon 9 Booster affect reusability of the first stage and landing success?

Finally, the data is processed in order to covert categorical columns into dummy values which will be later used for ML prediction

GitHub URL: https://github.com/cyesteban/capstone/blob/master/EDA%20with%20Visualization.ipynb

# EDA with SQL

SQL Queries used:

1) %sql select distinct LAUNCH_SITE from SPACEXDATASET;
2) %sql select * from SPACEXDATASET WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
3) %sql select SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS_KG from SPACEXDATASET WHERE CUSTOMER LIKE '%NASA%';
4) %sql select AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS_KG from SPACEXDATASET WHERE BOOSTER_VERSION LIKE '%F9 v1.1%';
5) %sql select MIN(DATE) AS MIN_DATE from SPACEXDATASET WHERE LCASE(LANDING__OUTCOME) LIKE '%success (ground%';
6) %sql select BOOSTER_VERSION from SPACEXDATASET WHERE LANDING__OUTCOME LIKE '%Success (drone ship)%' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
7) %sql SELECT SUBSTRING(MISSION_OUTCOME,1,7) AS MISSION_OUTCOME_, COUNT(SUBSTRING(MISSION_OUTCOME,1,7)) AS VALUE_COUNT FROM SPACEXDATASET GROUP BY SUBSTRING(MISSION_OUTCOME,1,7);
8) %sql SELECT BOOSTER_VERSION, PAYLOAD_MASS__KG_ FROM SPACEXDATASET WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET);
9) %sql SELECT TO_CHAR(TO_DATE(MONTH(DATE), 'MM'), 'MONTH') AS MONTH, LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXDATASET WHERE (DATE LIKE '2017%' AND LANDING__OUTCOME LIKE '%Success (ground pad)%');
10) %sql SELECT YEAR(DATE) AS YEAR, COUNT(LANDING__OUTCOME) AS LANDING_COUNT FROM SPACEXDATASET WHERE (LANDING__OUTCOME LIKE '%Success%' AND DATE BETWEEN '2010-06-04' AND '2017-03-20') GROUP BY YEAR(DATE) ORDER BY LANDING_COUNT DESC;

GitHub URL: https://github.com/cyesteban/capstone/blob/master/EDA%20with%20SQL%20bis.ipynb

# Build an Interactive Map with Folium

Goal:

- Mark the launch sites on the map for visualization
- Mark the number of successful and failed landings for each site
- Calculate distances between the sites and geographic references (cities, roads, etc.)

A Folium map was created and the following objects were added:

- Circles with popup labels: to mark the launch sites and show their names when clicking on them,
- Markers: to add labels to the map with the shortened name of the sites
- Marker cluster: to show several labels corresponding to the same coordinates that can be interactively expanded
- Mouse position coordinates in order to read/show the coordinates of a specific point on the map
- Line between coordinate points to show distances

GitHub URL: https://github.com/cyesteban/capstone/blob/master/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

# Build a Dashboard with Plotly Dash

A Dashboard has been built containing:

- A dropdown menu with the names of the Launch Sites
- A pie chart which takes the input from the dropdown menu and shows either the percentage of successful landings corresponding to each site (if all sites chosen) or the percentage of successful and failed landings for a specific site. This gives an overview of the success rate of each site
- A range slider to define a range for the payload
- A scatter plot to correlate the payload with the landing success for different Booster versions

GitHub URL: https://github.com/cyesteban/capstone/blob/master/SpaceX_dash.py

# Predictive Analysis (Classification)

- Data processing and Machine Learning model generation & evaluation

| Preprocessing | Split into Train and Test data | Train model(s) on Train data | Evaluate each model on Test data |
|---|---|---|---|
| Convert features into array with standardized distribution of values | Train data: 80% of Datapoints<br>Test data: 20 % of datapoints | GridSearchCV used to find best parameters for each model | Evaluation metrics: accuracy, f1, jaccards Score, log loss |

- The results of the different models are compared and the best one identified

GitHub URL: https://github.com/cyesteban/capstone/blob/master/Machine%20Learning%20Prediction.ipynb

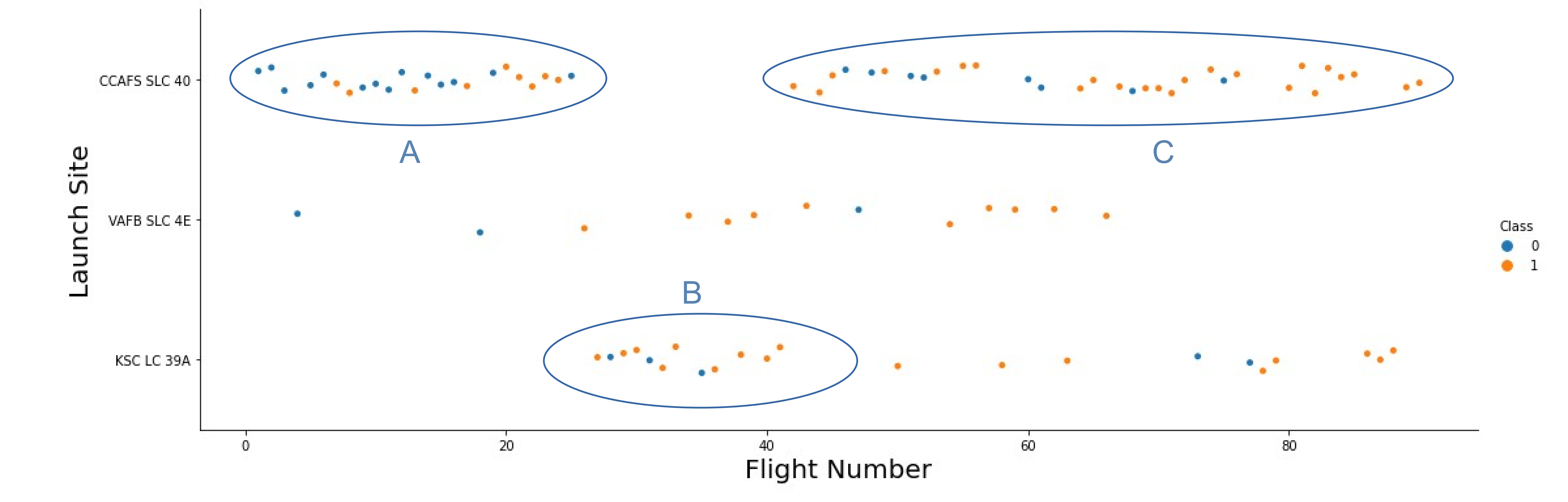
# Results

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

Section 2

# Insights drawn from EDA

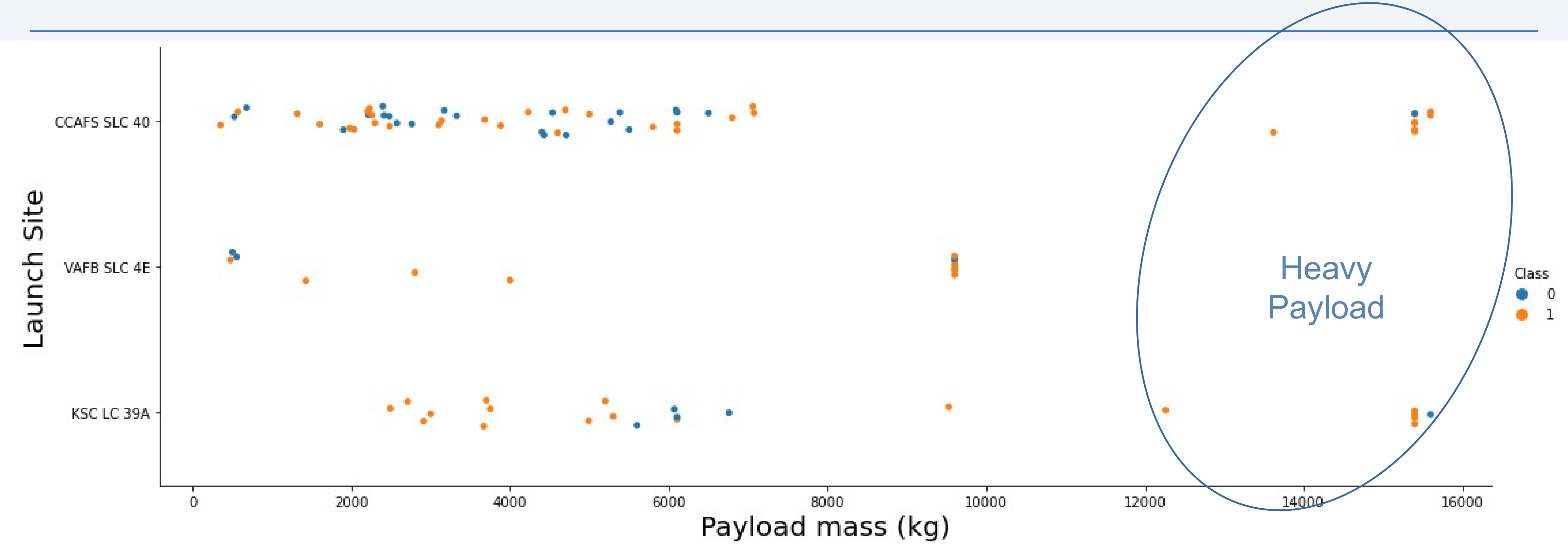
# Flight Number vs. Launch Site




Observations:

- CCAFS SLC 40 was the first Launch Site used (low Flight Numbers, Zone A)
  - In the first flights the success rate is quite low
- For flight numbers between ~27 and ~41 the KSC LC 39A site was used almost exclusively (Zone B)
- From the flight number ~42 on the CCAFS SLC 40 was used again very often (Zone C)
- The VAFB SLC 4E site has been used sporadically over time
- In general, the success rate of the landing seems to increase with the flight number (i.e. with time) for all launch sites

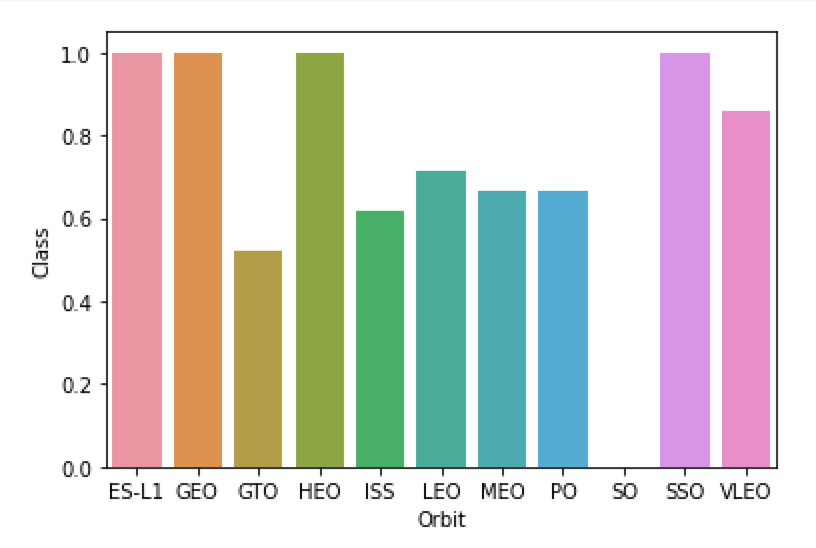
# Payload vs. Launch Site




Observations:

- There is no correlation between PayloadMass and landing success
- No heavy Payload (>10000 kg) rockets have been launched from the site VAFB SLC 4E
- Most of the launched rockets have Payload under 8000 kg

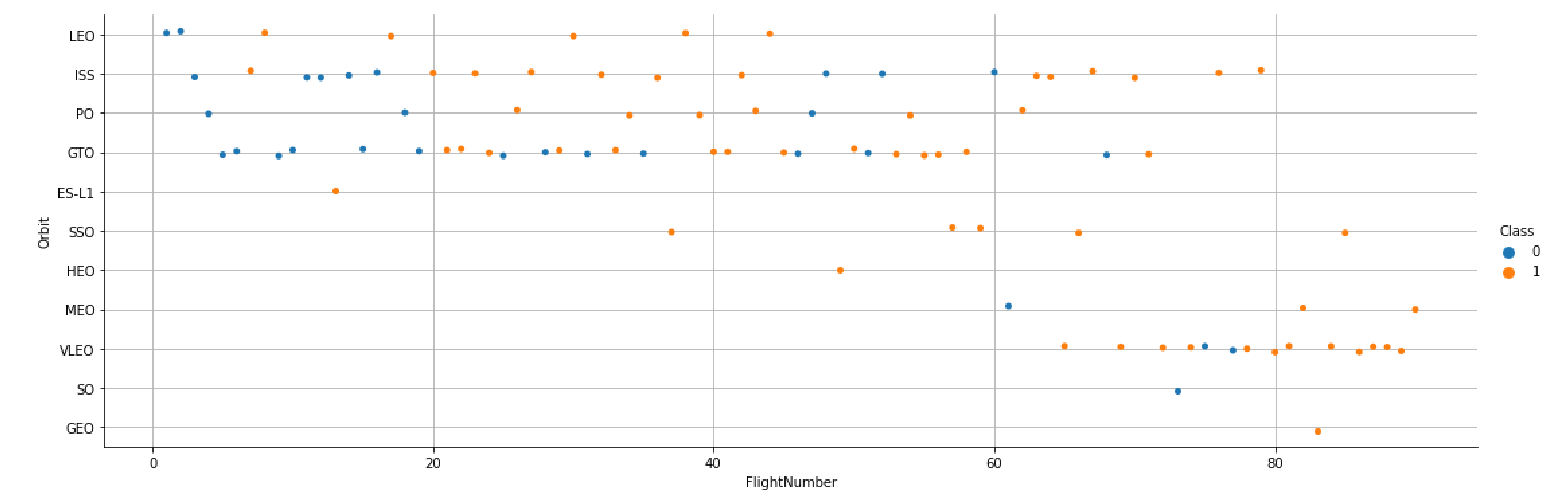
# Success Rate vs. Orbit Type




Observations:

- The orbits with the highest success rate (class mean value = 1) are: ES-L1, GEO, HEO and SSO
- As shown in the next slide, ES-L1, HEO and GEO have only one datapoint each, so the mean value has no sense here
- The SSO orbit has 5 datapoints, all with a successful landing

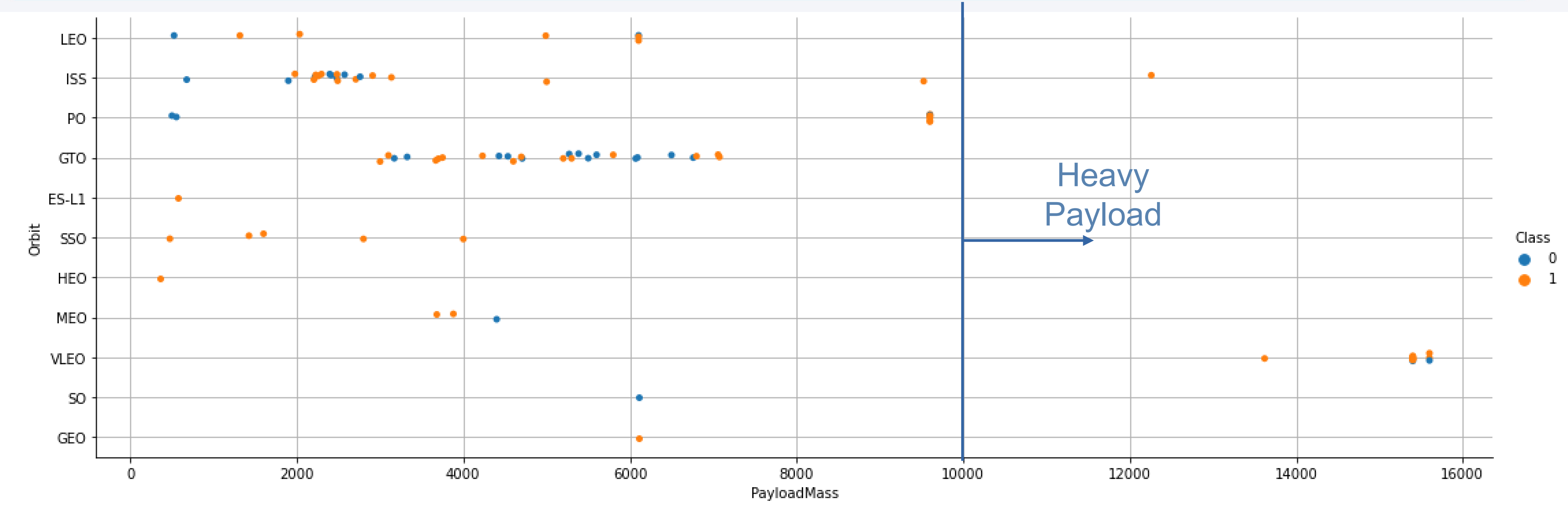
# Flight Number vs. Orbit Type




Observations:

- Some of the orbits have only few datapoints (ES-L1, HEO, MEO, SO, GEO) and therefore no conclusions can be drawn from them
- For the LEO orbit, the landing success seems to be related to the Flight number (higher Flight numbers are successful while few low flight numbers are not)
- For other orbits like ISS, PO, VLEO or GTO there is no relationship
- The SSO shows only successful landings (also no relationship with the flight number)

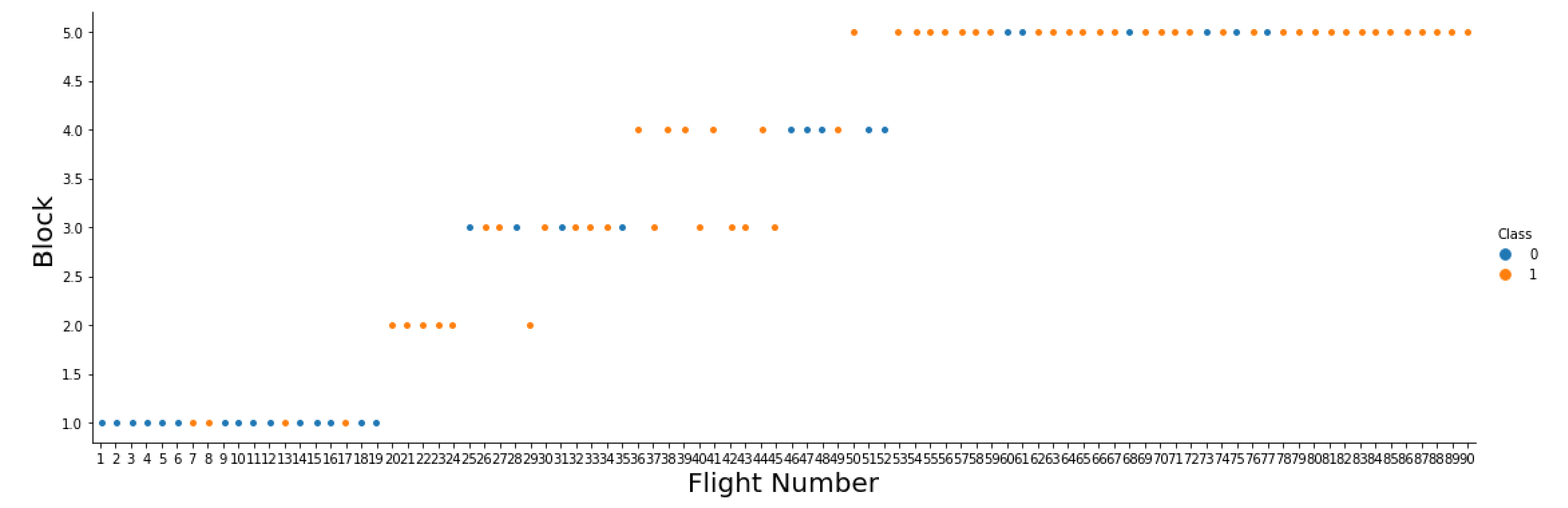
# Payload vs. Orbit Type




Observations:

- Some of the orbits have only few datapoints (ES-L1, HEO, MEO, SO, GEO) and therefore no conclusions can be drawn
- from them
- Heavy Payload rockets (> 10000 kg) have been sent only to orbits ISS and VLEO
- For orbits LEO, PO and ISS heavier payloads have more chances of being successful than lower payloads
- For GTO there is no correlation between payload and success rate

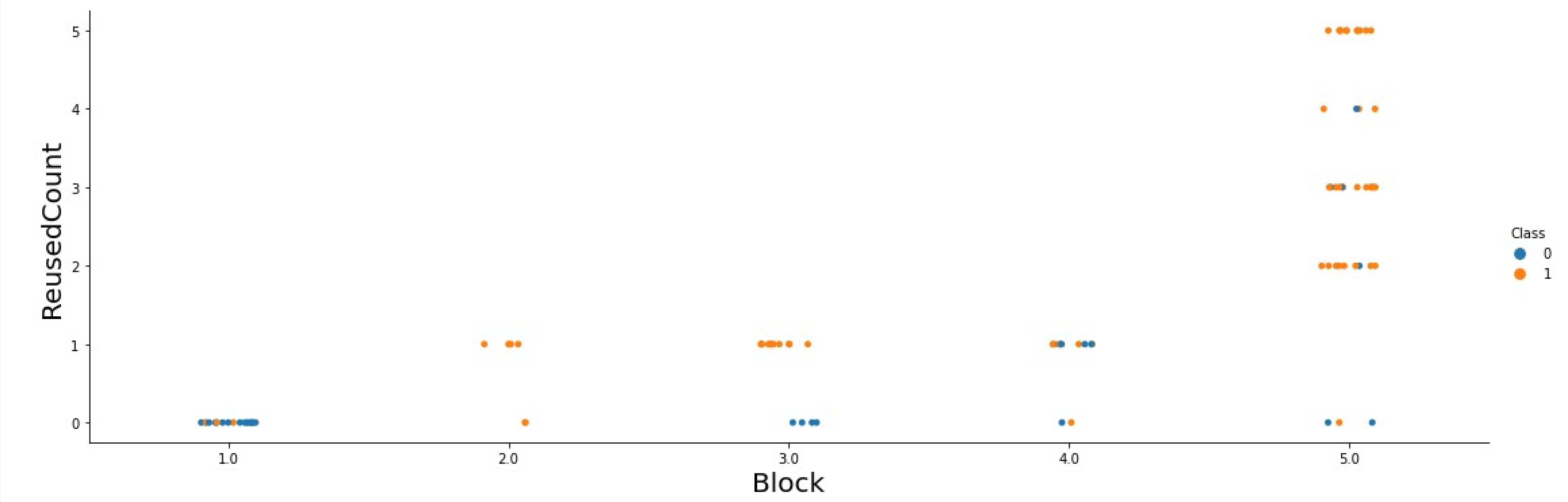
# Block vs Flight Number




Observations:

- There have been several development phases for the Falcon 9 Booster (Block numbers)
- Here we can see how the different phases are distributed with the flight number (i.e. with time)
- While the first phase Block 1 had a low landing success rate, Block 2 landings were always successful and for Block 3, Block 4 and Block 5 the success rate is high

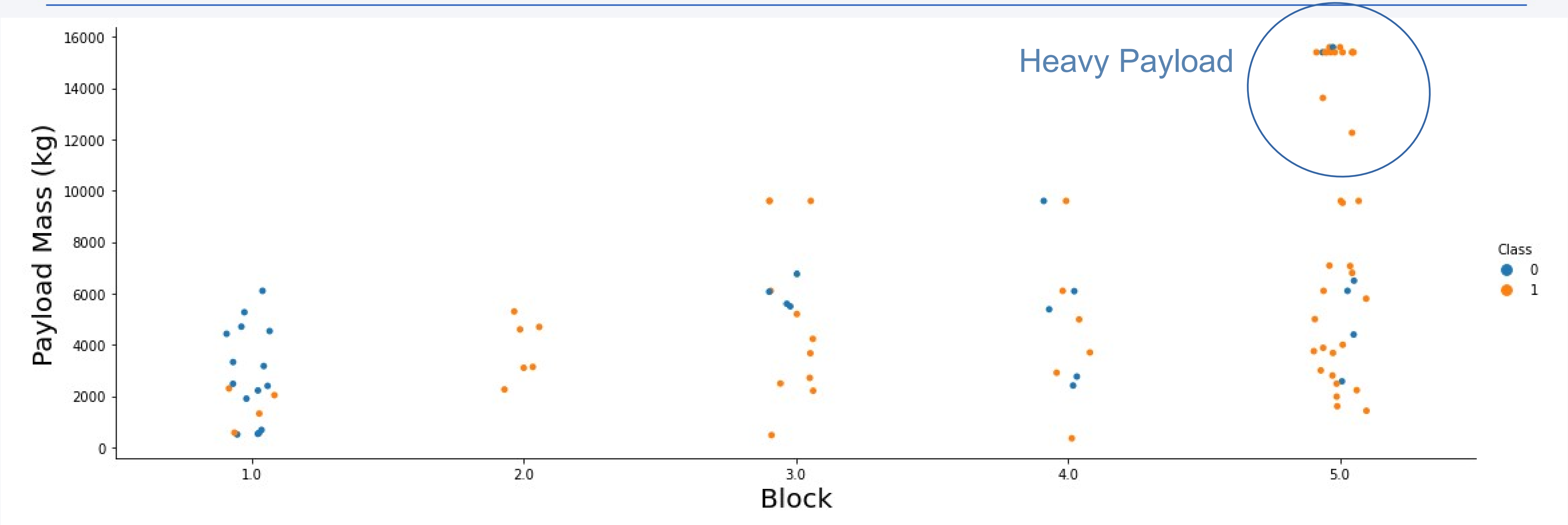
# Block vs ReusedCount




Observations:

- While the first development phase of the Falcon 9 Booster (Block1) lead to mostly unsuccessful landings, the next versions (Block 2, Block 3 and Block 4) allow for reusability of the first stage.
- Block 5 rockets have been reused up to five times

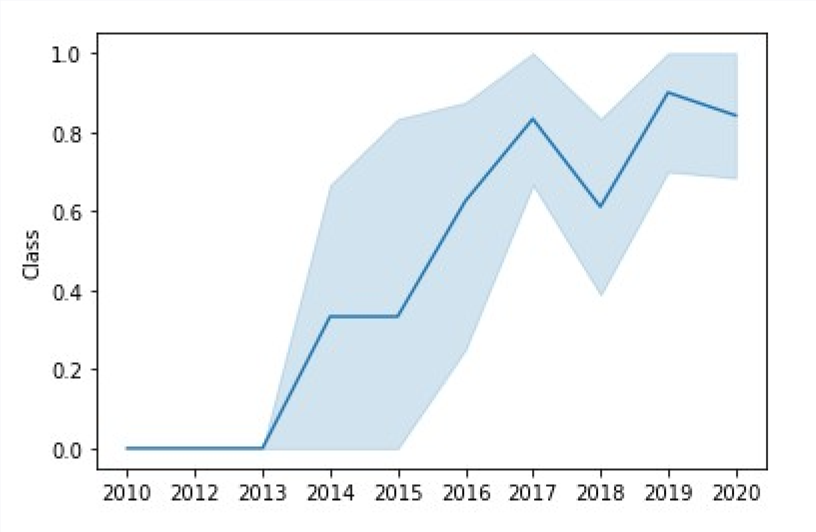
# Block vs Payload Mass




Observations:

- The first Block versions had low payload
- The payload has progressively increased in later Block versions
- The heaviest rockets belong all to the Block 5 version

# Launch Success Yearly Trend with 95% confidence interval




Observations:

- Since 2013, the success rate of the rocket landings has increased steadily and is currently at ~80%

# All Launch Site Names

Find the names of the unique launch sites

%sql select distinct LAUNCH_SITE from SPACEXDATASET;

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

- The command DISTINCT delivers the unique values of a column
- The Launch Site names are: CCAFS LC-40, CCAFS SLC-40, KSC LC 39A, VAFB SLC-4E

# A ) Launch Site Names Begin with 'KSC'

Find 5 records where launch sites begin with `KSC`

%sql select * from SPACEXDATASET WHERE LAUNCH_SITE LIKE 'KSC%' LIMIT 5;

| 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) |
| 2017-03-16 | 06:00:00 | F9 FT B1030 | KSC LC-39A | EchoStar 23 | 5600 | GTO | EchoStar | Success | No attempt |
| 2017-03-30 | 22:27:00 | F9 FT B1021.2 | KSC LC-39A | SES-10 | 5300 | GTO | SES | Success | Success (drone ship) |
| 2017-05-01 | 11:15:00 | F9 FT B1032.1 | KSC LC-39A | NROL-76 | 5300 | LEO | NRO | Success | Success (ground pad) |
| 2017-05-15 | 23:21:00 | F9 FT B1034 | KSC LC-39A | Inmarsat-5 F4 | 6070 | GTO | Inmarsat | Success | No attempt |

- The LIKE command allows to search for substrings
- The command LIMIT gives the maximum amount of rows to show

# B ) Launch Site Names Begin with 'CCA'

Find 5 records where launch sites begin with `CCA`

%sql select * from SPACEXDATASET WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;

| 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 |

- The LIKE command allows to search for substrings
- The command LIMIT gives the maximum amount of rows to show

# Total Payload Mass

Calculate the total payload carried by boosters from NASA

%sql select SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS_KG from SPACEXDATASET WHERE CUSTOMER LIKE '%NASA%';

| total_payload_mass_kg |
|---|
| 107010 |

- The command SUM provides the sum of values of a column
- The total payload mass for NASA boosters is 107010 kg
- If filtering by customer “NASA (CRS)” the total mass is 48213 kg

# Average Payload Mass by F9 v1.1

Calculate the average payload mass carried by booster version F9 v1.1

%sql select AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS_KG from
SPACEXDATASET WHERE BOOSTER_VERSION LIKE '%F9 v1.1%';

| average_payload_mass_kg |
| --- |
| 2534 |

- The command AVG delivers the average of the values of a column
- The average payload mass for Booster version F9 v1.1 is 2534 kg

# First Successful Ground Landing Date

Find the dates of the first successful landing outcome on ground pad

%sql select MIN(DATE) AS MIN_DATE from SPACEXDATASET WHERE LCASE(LANDING__OUTCOME) LIKE '%success (ground%';

| min_date |
|---|
| 2015-12-22 |

- The MIN command provides the minimal value of a column
- The first successful landing on ground pad was on 2015-12-22
- The first successful landing on drone ship was on 2016-04-08

# 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

%sql select BOOSTER_VERSION from SPACEXDATASET WHERE LANDING__OUTCOME LIKE '%Success (drone ship)%' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;

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

- BETWEEN allows to filter values within a numerical range
- The Booster versions for landing on drone ship are: F9 FT B1022, F9 FT B1026, F9 FT B1021.2, F9 FT B1031.2

# Total Number of Successful and Failure Mission Outcomes

Calculate the total number of successful and failure mission outcomes

%sql SELECT SUBSTRING(MISSION_OUTCOME,1,7) AS MISSION_OUTCOME_, COUNT(SUBSTRING(MISSION_OUTCOME,1,7)) AS VALUE_COUNT FROM SPACEXDATASET GROUP BY SUBSTRING(MISSION_OUTCOME,1,7);

| mission_outcome_ | value_count |
|---|---|
| Failure | 1 |
| Success | 100 |

- The COUNT command counts the number of values in a column or group
- GROUPBY allows to divide the rows into groups according to a given value
- SUBSTRING is used in order to get only two groups here (Success and Failure), since there are two different values for successful landings in the MISSION_OUTCOME column
- There are 100 successful mission outcomes and one failed mission

# Boosters that Carried Maximum Payload

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

%sql SELECT BOOSTER_VERSION, PAYLOAD_MASS__KG_ FROM SPACEXDATASET WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET);

| booster_version | payload_mass__kg_ |
|---|---|
| F9 B5 B1048.4 | 15600 |
| F9 B5 B1049.4 | 15600 |
| F9 B5 B1051.3 | 15600 |
| F9 B5 B1056.4 | 15600 |
| F9 B5 B1048.5 | 15600 |
| F9 B5 B1051.4 | 15600 |
| F9 B5 B1049.5 | 15600 |
| F9 B5 B1060.2 | 15600 |
| F9 B5 B1058.3 | 15600 |
| F9 B5 B1051.6 | 15600 |
| F9 B5 B1060.3 | 15600 |
| F9 B5 B1049.7 | 15600 |

- A subquery is used to define the value of the payload as the maximum payload value and filter the table accordingly
- List of Booster Versions: 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

# 2017 Launch Records

List the records which will display the month names, successful landing_outcomes in ground pad, booster versions, launch_site for the months in year 2017

%sql SELECT TO_CHAR(TO_DATE(MONTH(DATE), 'MM'), 'MONTH') AS MONTH, LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXDATASET WHERE (DATE LIKE '2017%' AND LANDING__OUTCOME LIKE '%Success (ground pad)%');

| MONTH | landing__outcome | booster_version | launch_site |
|---|---|---|---|
| FEBRUARY | Success (ground pad) | F9 FT B1031.1 | KSC LC-39A |
| MAY | Success (ground pad) | F9 FT B1032.1 | KSC LC-39A |
| JUNE | Success (ground pad) | F9 FT B1035.1 | KSC LC-39A |
| AUGUST | Success (ground pad) | F9 B4 B1039.1 | KSC LC-39A |
| SEPTEMBER | Success (ground pad) | F9 B4 B1040.1 | KSC LC-39A |
| DECEMBER | Success (ground pad) | F9 FT B1035.2 | CCAFS SLC-40 |

- The date and the LANDING_OUTCOME are filtered by similarity of the string with the command LIKE
- In order to show the month names, first the integer value of the month is extracted from the date with MONTH(DATE), then this value is given to the function TO_DATE and converted to a string with TO_CHAR

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

Rank the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order

%sql SELECT YEAR(DATE) AS YEAR, COUNT(LANDING__OUTCOME) AS SUCCESS_LANDING_COUNT FROM SPACEXDATASET WHERE (LANDING__OUTCOME LIKE '%Success%' AND DATE BETWEEN '2010-06-04' AND '2017-03-20') GROUP BY YEAR(DATE) ORDER BY LANDING_COUNT DESC;

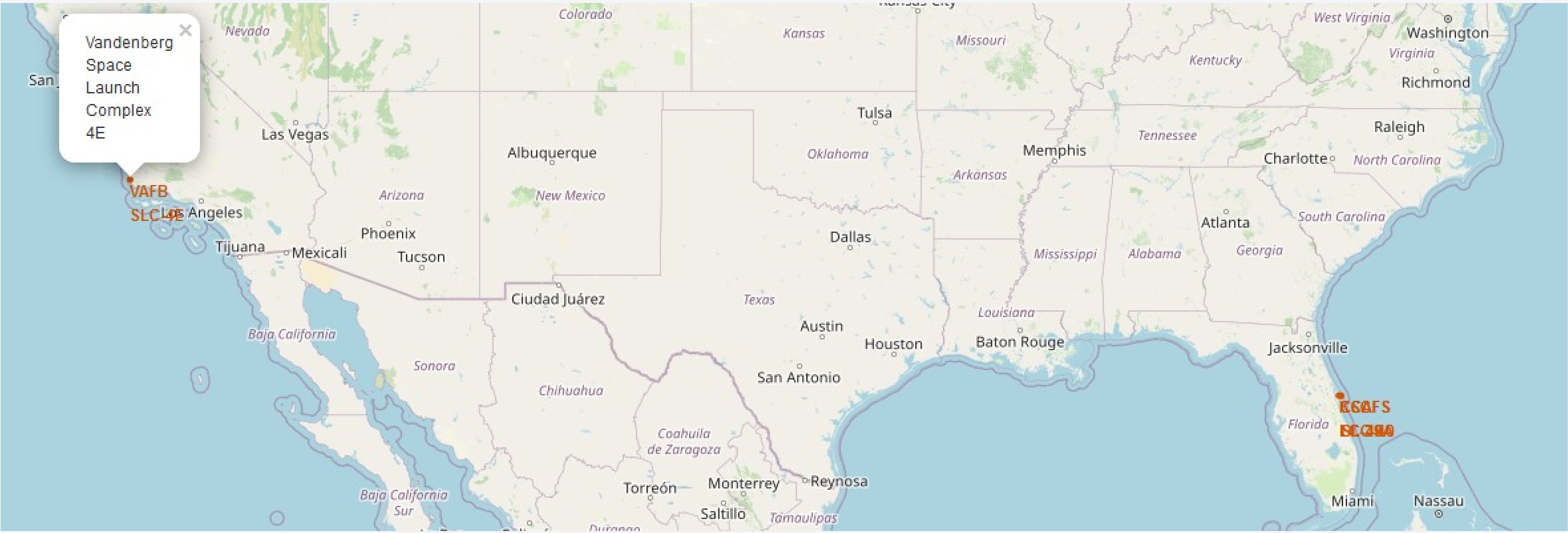
| YEAR | success_landing_count |
|---|---|
| 2016 | 5 |
| 2017 | 2 |
| 2015 | 1 |

- The successful landing outcomes were highest in 2016

Section 4

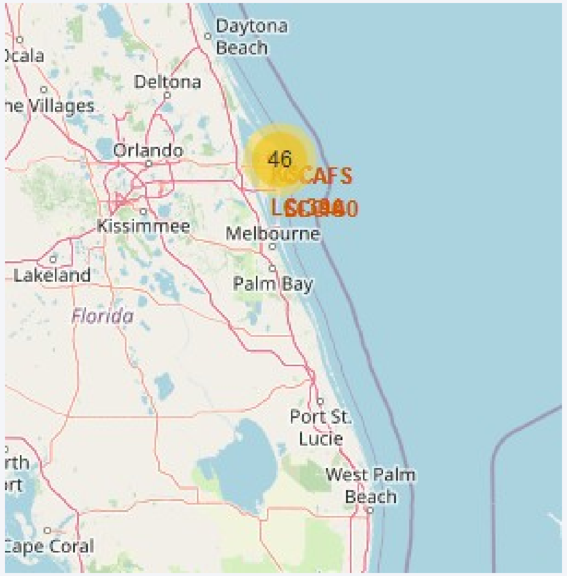
# Launch Sites Proximities Analysis

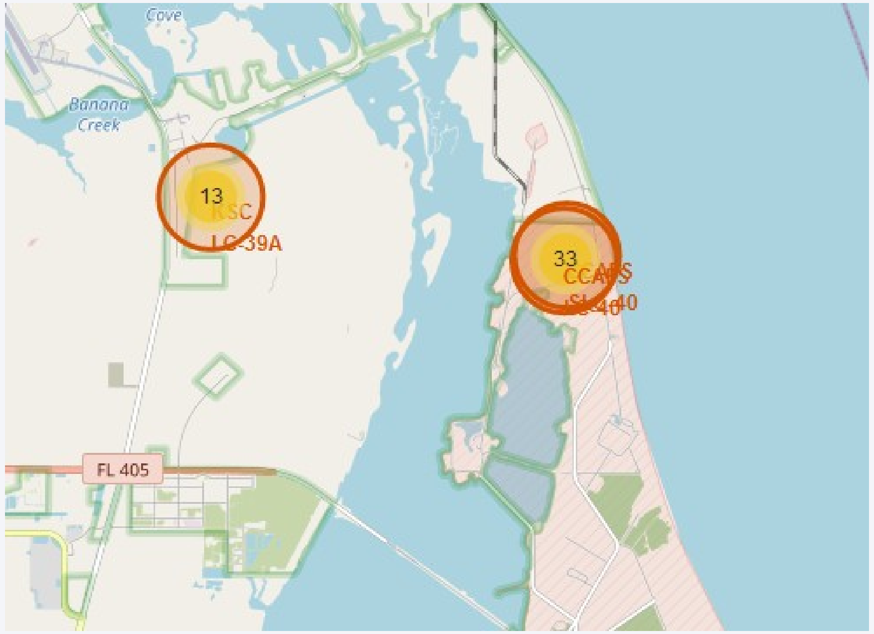
# View of all launch sites with Folium

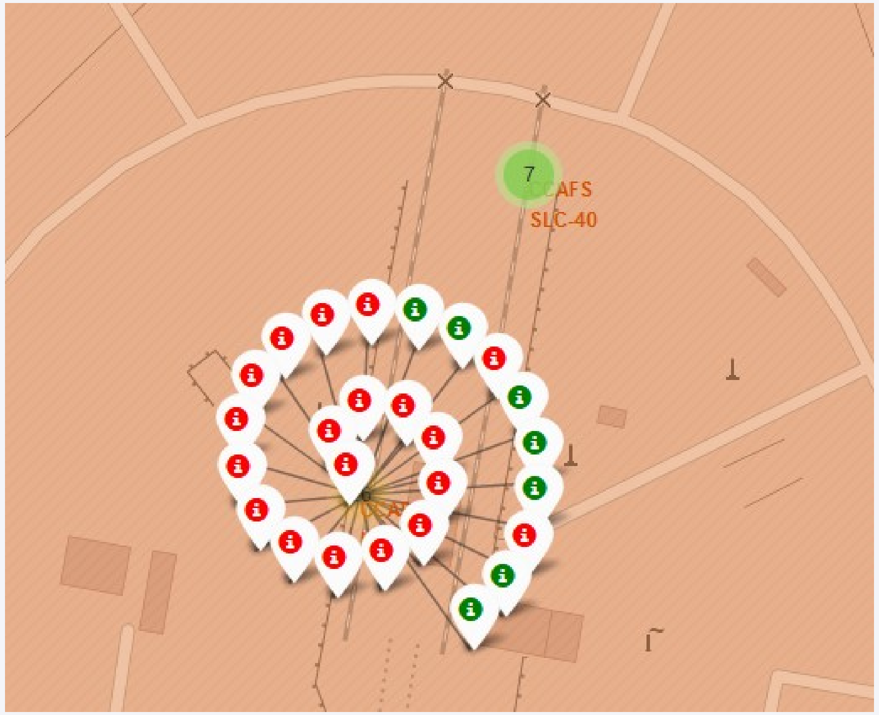



- The launch sites are situated in two opposite points of the USA: the VAFB site is on the Pacific side (west coast) while the CCAF and KSC sites are on the Atlantic side (east coast)
- All sites are very close to the coast
- The sites on the state of Florida are closer to the equator line than the VAFB site

# Visualizing Launch Success per Launch Site



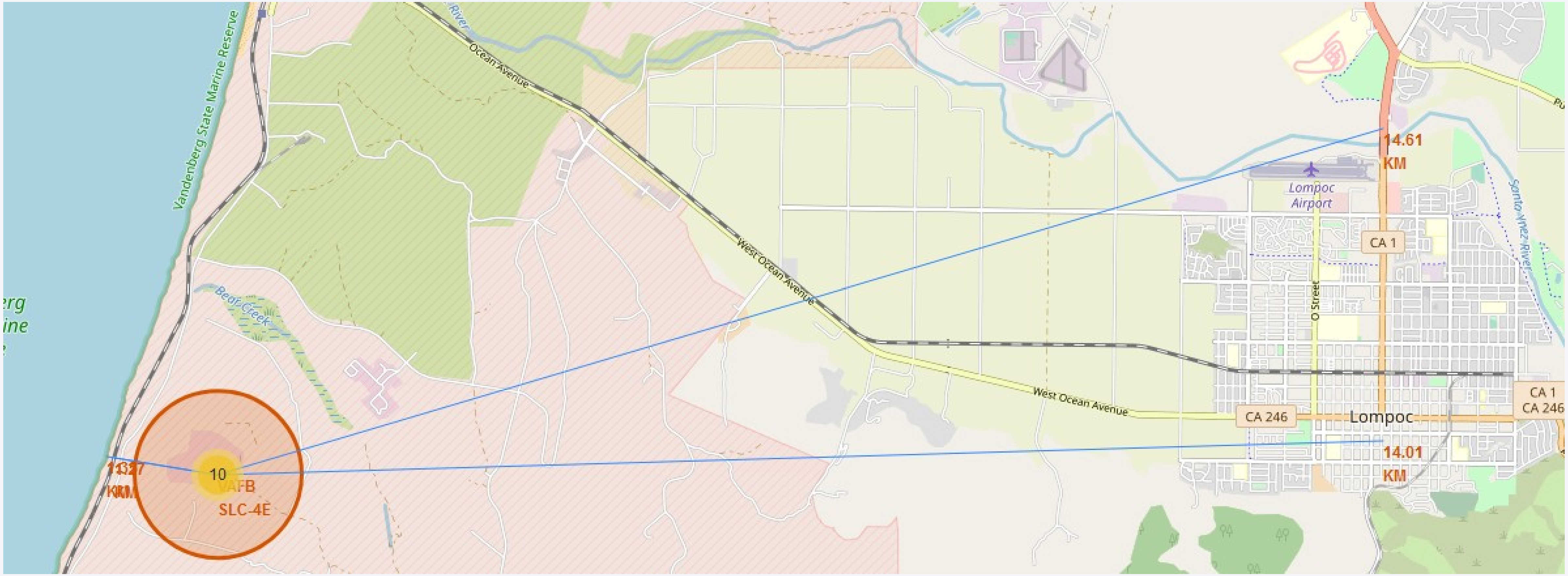




**Zoom**

Successful launches are marked by green icons while unsuccessful ones are depicted by red icons. This allows to easily recognize the predominant case for each launch site: for CCAFS LC-40 and VAFB sites most of the launches did not success; for CCAFS SLC-40 the distribution is 50/50. The KSC site has predominantly successful launches

# VAFB Site and distances to relevant places




The VAFB site is 14 km away from the next city Lompoc, as well as from the next highway. The next railway however is very close: 1.27 km and the coast is only 1.35 km away.

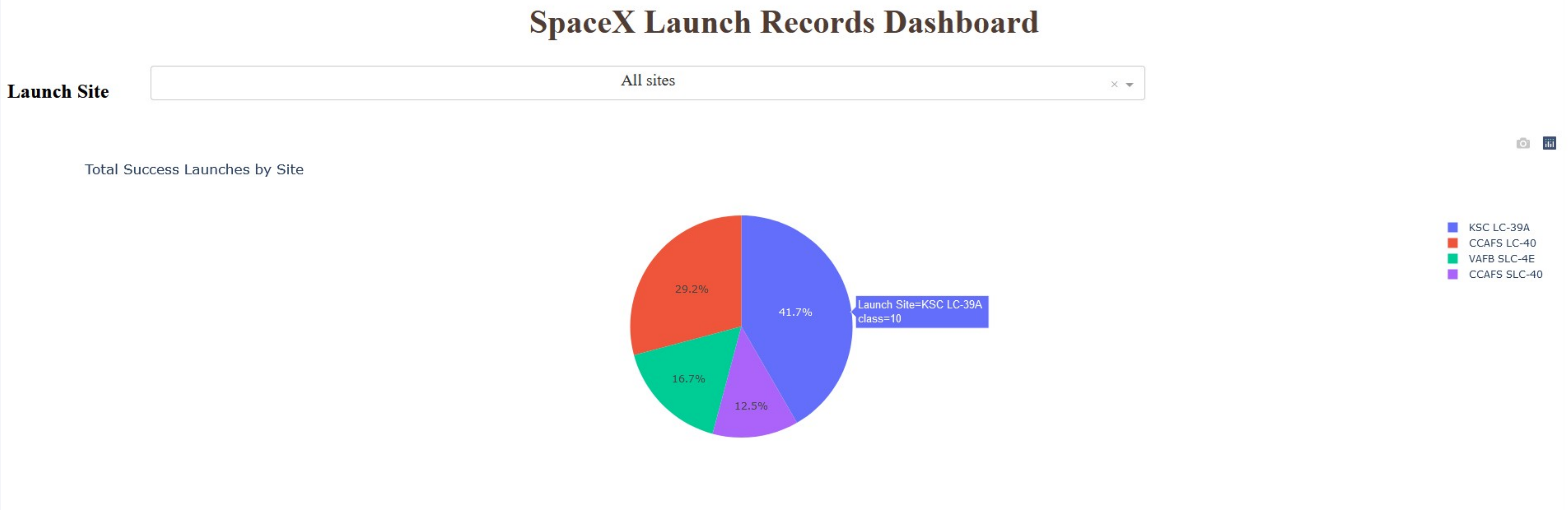
In general, cities are several kilometers apart from every launch site, which seems to be a safety measure. All sites are very close to the coast probably to reduce accidents on the ground.

Railways are quite close to all sites (1-2 km) as might be useful to transport material to the sites. Highways are not very close, but smaller roads allow reaching the sites by car or truck.

Section 5

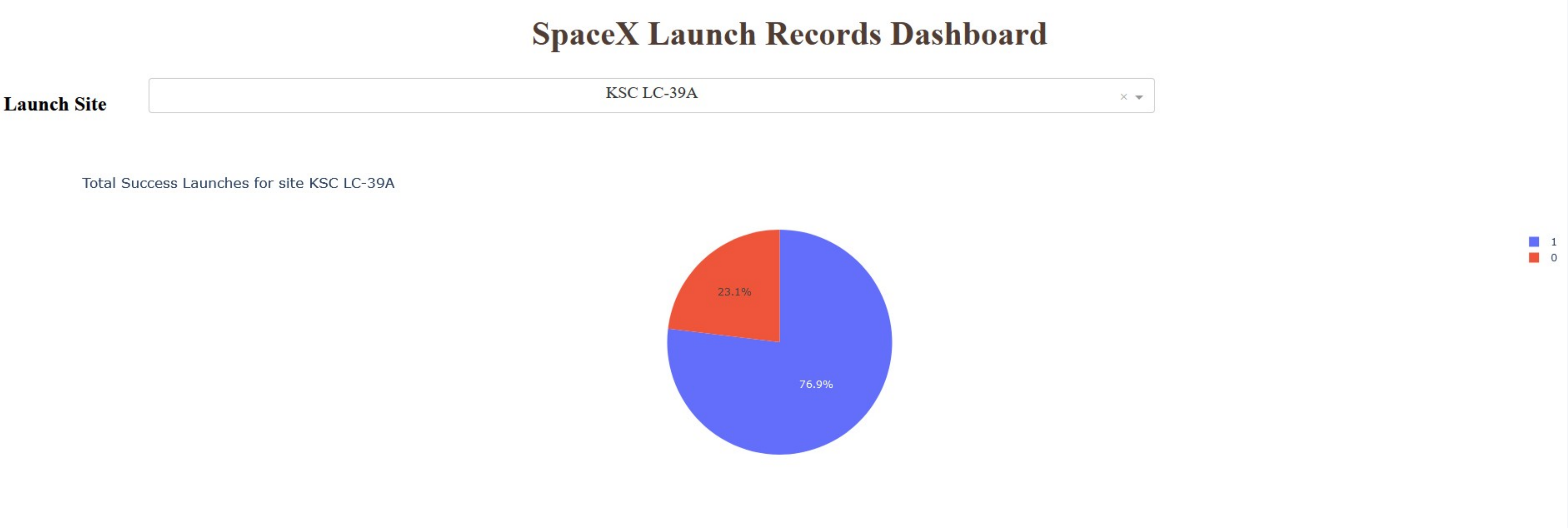
# Build a Dashboard with Plotly Dash

# Distribution of successful landings per Launch Site




- The pie chart shows the total number of successful landings per Launch site (the number is shown when hovering over with the mouse) and the percentage of successful landings per site with respect to the total number of successful landings
- KSC is the Launch Site with the highest number of successful landings

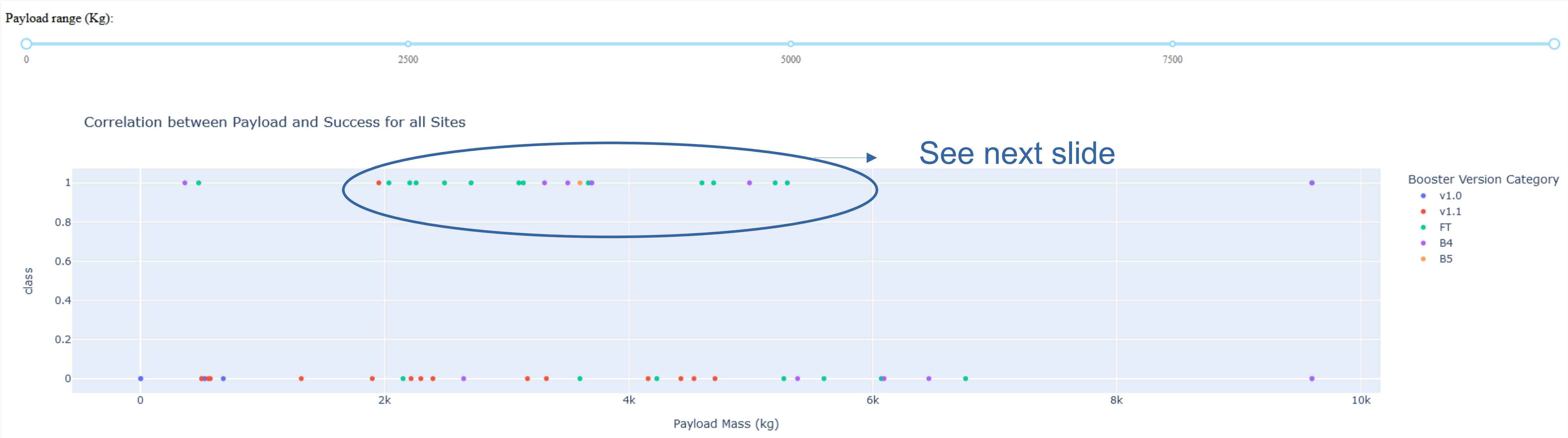
# Percentage of successful landings in KSC Site




- The drop-down menu allows to choose a specific Launch Site, in this case the KSC LC-39A.
- The pie chart shows the distribution of successful and non successful landings for the Launch Site KSC.
- Approximately three quarters of the launches could recover the first stage for this site.

# Payload vs Success for all Sites

- Screenshot: Payload vs Success for all Sites and a Payload range from 0 to 10000 kg
- The largest success rate is for payloads between 2000 and 6000 kg

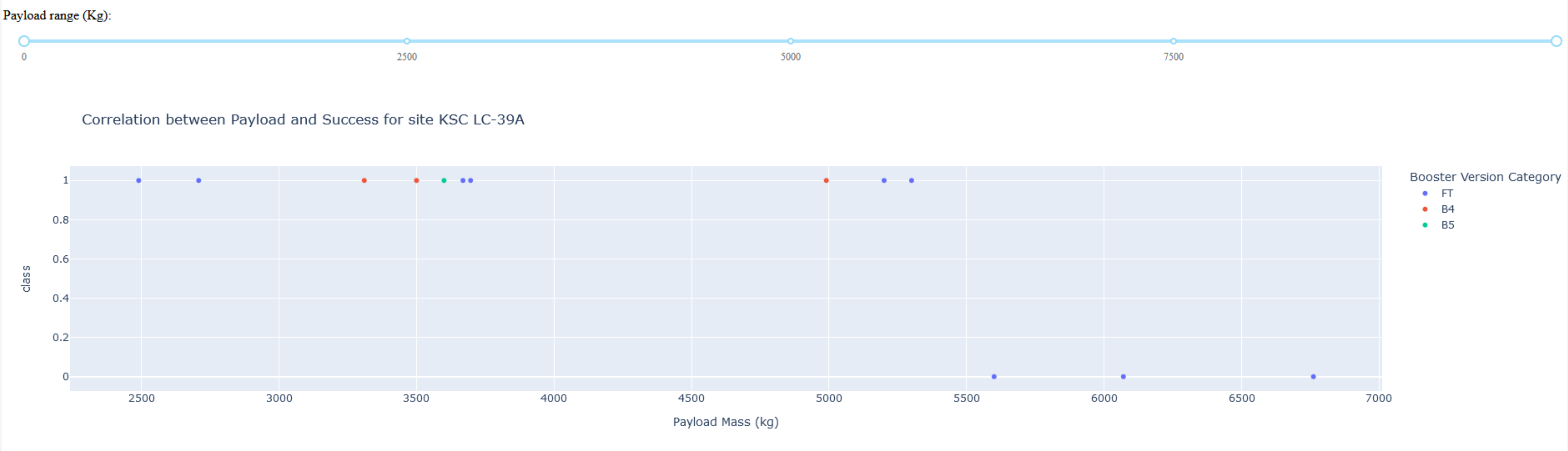


# Payload vs Success for all Sites

- Screenshot: Payload vs Success for all Sites and a Payload range from 0 to 6000 kg
- In this range, the Booster Versions B4 and FT show the highest success rate (65% for FT and 71% for B4), while version v1.1 shows poor success rate

# Payload vs Success for KSC Site

- For launches of site KSC the payload seems to be an important parameter: all launches with payloads below 5500 kg were successful, while all launches with higher payloads were not.
- For the sites CCAFS LC-40, CCAFS SLC-40 and VAFB there is no relationship between payload and success rate



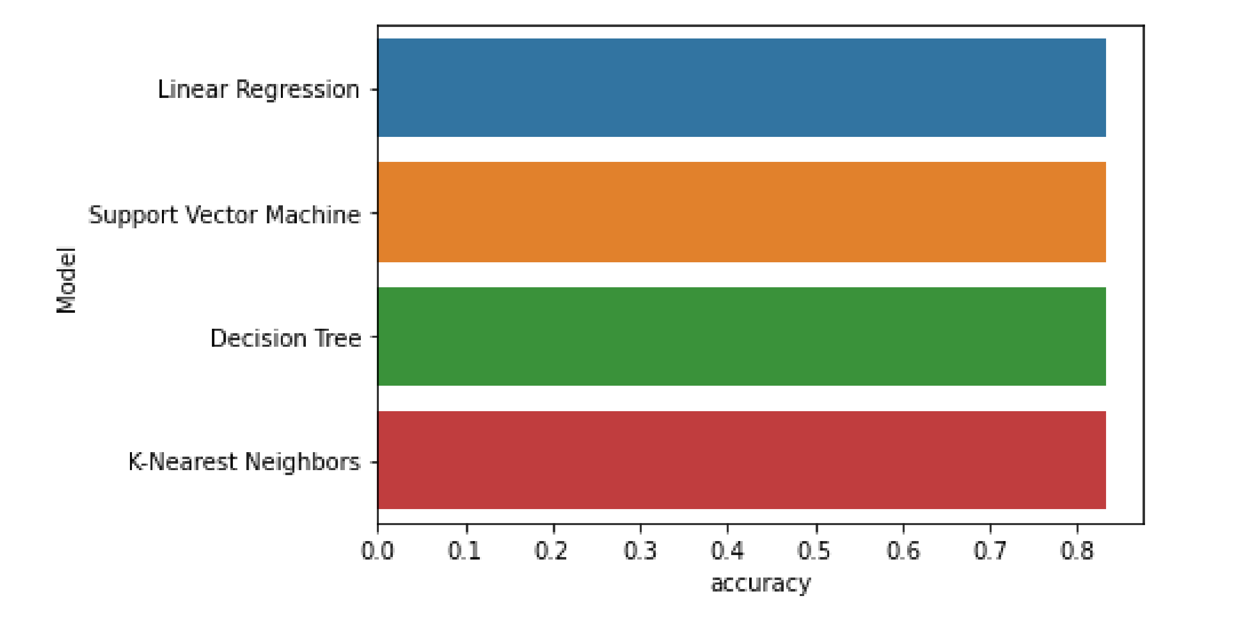
Section 6

# Predictive Analysis (Classification)

# Classification Accuracy

- Bar Chart: comparison of models by accuracy

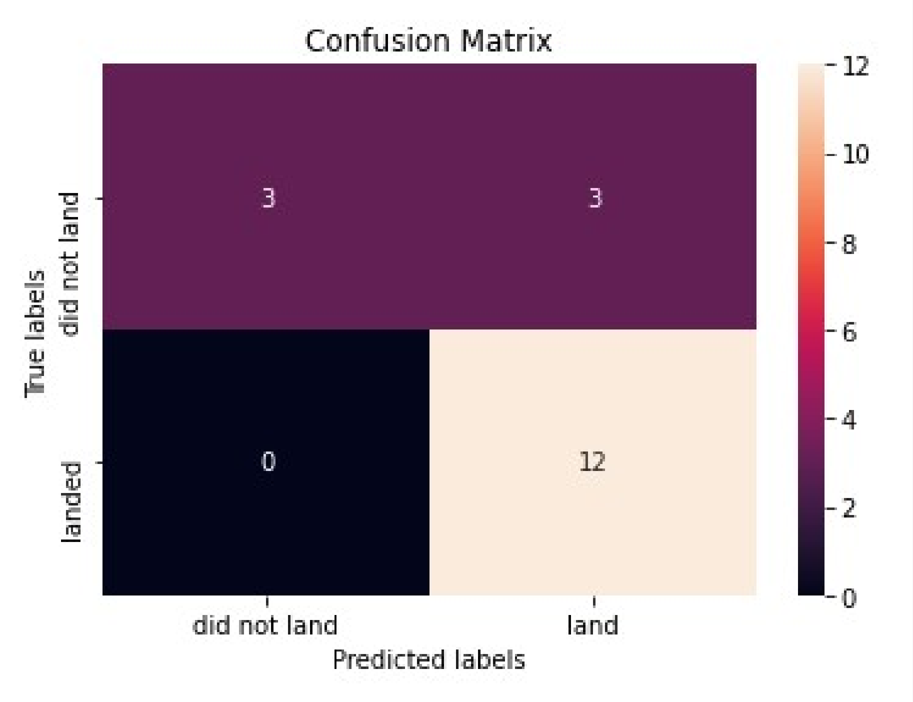
- All models give the same accuracy on the test data.
- The Decision Tree model varies between accuracies of 0.77 and 0.83, depending on the run, due to it's random forest nature
- KNN has the lowest logloss (lowest error)




| | Model | accuracy | f1_weighted | jaccard | logloss |
|---|---|---|---|---|---|
| 0 | Linear Regression | 0.833333 | 0.814815 | 0.8 | 0.478667 |
| 1 | Support Vector Machine | 0.833333 | 0.814815 | 0.8 | NaN |
| 2 | Decision Tree | 0.833333 | 0.814815 | 0.8 | 0.371943 |
| 3 | K-Nearest Neighbors | 0.833333 | 0.814815 | 0.8 | 0.366219 |

# Confusion Matrix

- All models give the same confusion matrix:
- The models can correctly predict unsuccessful flights: all flights labeled as unsuccessful matched reality (no false negatives)
- However, there are false positives: 3 flights were predicted as successfully landing, but did not land
- The evaluation of correctly and incorrectly predicted datapoints is given by the f1 score. The f1_weighted score for all models is 81%



# Conclusions

Variables such as Flight number, Payload, Launch Site, orbit, Booster Version have been analyzed in order to detect their influence on the first stage landing success

- The Kennedy Space Complex shows the highest number of successful first stage recovery (3 out of 4 launches here succeeded). For this specific site, the payload seems to play a role in landing success: only rockets with payloads under 5500 kg could recover the first stage. This relationship could not be established for other sites.
- The development phase of the Falcon 5 rockets play an important role: the success of the first stage recovery increased with the development phase (Block number)
- In general, the highest first stage recovery success rate is given for payloads between 2000 kg and 6000 kg. Within this range, the Booster Versions B4 and FT have the highest success rates of 71% and 65%, respectively
- The orbit to which the rocket is aimed does not play a major role in the first stage landing success

SpaceX launches have continuously increased the success of the first stage recovery since 2013; current success is about 80%

Several classification machine learning models have been trained for predicting the first stage recovery success with an accuracy of 83%

# Conclusions

Launch price estimation

- Data shows that the current  first stage recovery success rate is ~80% (year 2020). Thus, cost for the first stage fabrication can be reduced by up to 80%
- For a new company that would like to start offering space flights today, the total starting cost would include additional factors such as investment costs, fixed costs and other variable costs such as fuel and second stage cost
- As no breakdown of cost factors is available, total cost savings can not be quantified
- Furthermore added costs due to necessary repairs, landing pads etc. are unknown
- Assuming comparable margins, the approximately 60% lower price point of SpaceX (62 million dollars vs. 165 million dollars for other companies) indicates that first stage cost is a major cost factor
- With the given simplifications and assuming all other things being equal, the first stage makes up approx. 80% of total starting cost in conventional companies, which Space X reduced by approx. 80%

# Thank you!