IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Áron Mikes
October 2021

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

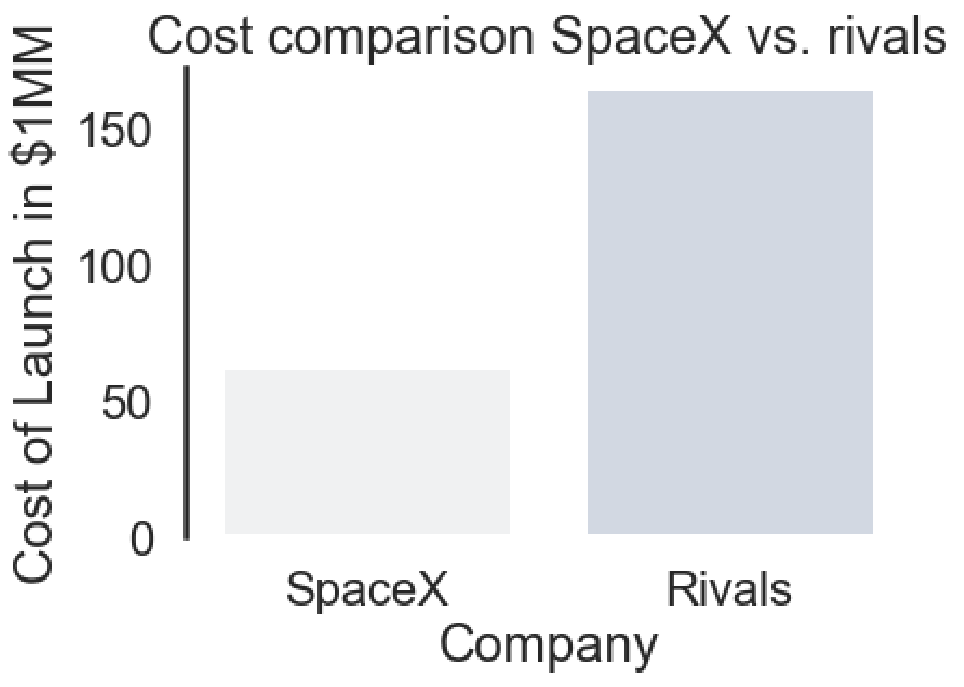
Executive Summary

- The orbit and payload mass are the most important factors to determine the success of a launch.
- SpaceX has gotten more succesful with time and this trend can be assumed to continue, making a model that can predict failures more valuable to competitors.
- As model that was developed contains no false negatives and is a powerful tool to predict when the first stage of a rocket will not land. This allows to bid against SpaceX with relatively little risk.

Introduction

SpaceX advertises Falcon 9 rocket launches with a 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 of the rocket.

This project will predict when the first stage can be reused and when not, enabling a rival company to offer competitive bids against SpaceX.



Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Using SpaceX' API and webscraping public data
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

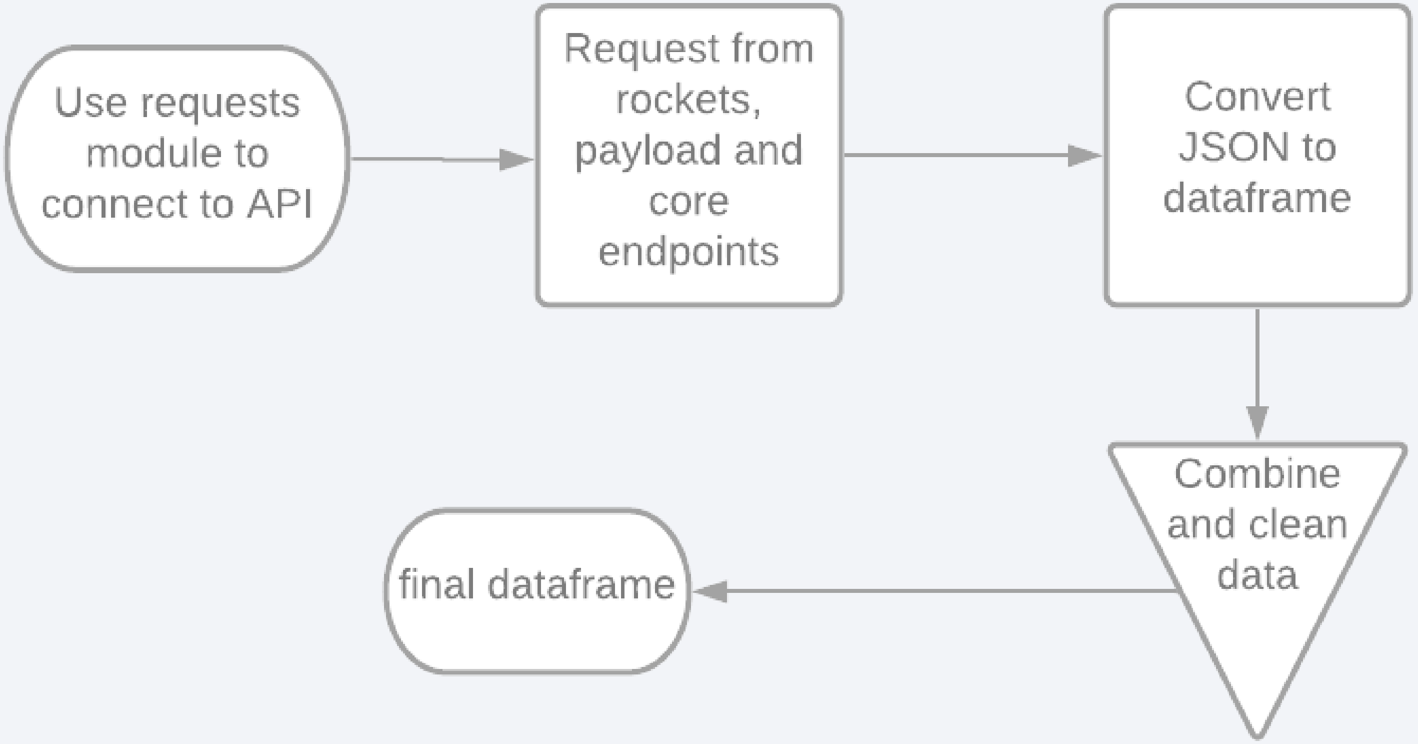
Data Collection

- Data was collected from two different sources
  - A public list of past launches of the Falcon 9 on Wikipedia
  - The SpaceX API (api.spacexdata.com)

Data Collection – SpaceX API

- Using the requests module I connect to the SpaceX API and then load the response object as a pandas dataframe.

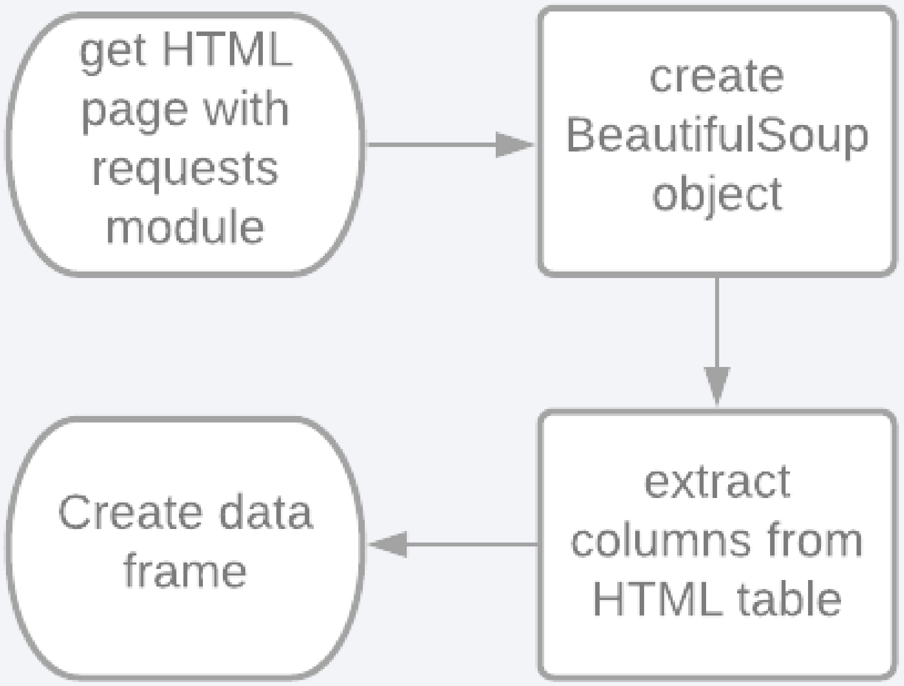
- Jupyter Notebook hosted on GitHub



Data Collection - Scraping

- After the entire page was retrieved the correct HTML table was selected. Finally the table was parsed and converted into a pandas dataframe.

- Jupyter Notebook hosted on GitHub



Data Wrangling

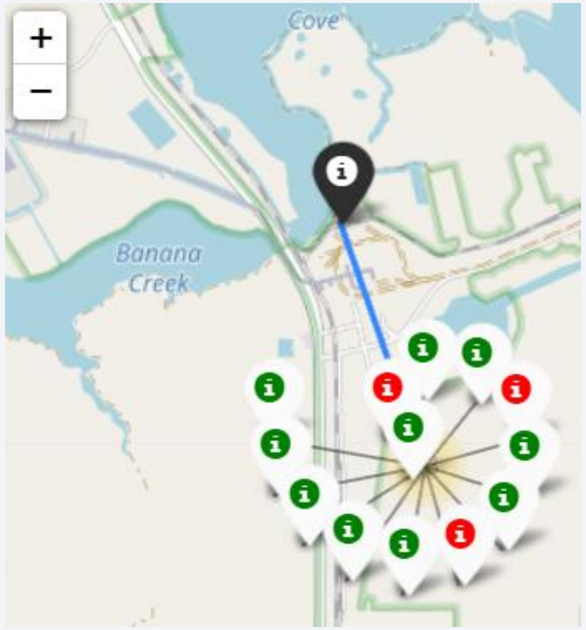
- Firstly the different launch sites were looked at, followed by the by different orbits that the rockets were shot into. This was then compared to the mission outcome. The various outcomes were then summarized in a new column 'Class' that merely contains 1 or 0 (failure or success to reuse the first stage of the rocket).

- Jupyter Notebook hosted on GitHub

EDA with Data Visualization

- To help explore the data a map of all launch sites was created and all successes/failures were plotted on the map.

- With another map the distance of launch sites to certain point of interests were visualized, in order to find correlations



- Jupyter Notebook hosted on GitHub.
  (run on nbviewer to see interactive maps)

Interactive Map with Folium

- For the maps shown on the last slide some features had to be added to increase the usability of the map:
  - Clusters were added, so that it is clearly visible how many launches were performed at each site
  - Red and green markers were added to clearly distinguish failend and successful missions
  - Interactive lines that show the disntace to points of interest

- Jupyter Notebook hosted on GitHub.
  (run on nbviewer to see interactive maps)

EDA with SQL

- Using the dataset hosted on DB2, the following queries were performed:
  - Selecting the names of the unique launch sites
  - Selecting 5 records where the launch sites starts with 'CCA'
  - Selecting total payload mass for launches performed for NASA
  - Selecting average payload mass of a certain booster version
  - Selecting the date of the first successful landing on a ground pad
  - Selecting successful landings by drone ship and payload mass between 4-6k kg.
  - Selecting total number of successful and failed outcomes
  - Selecting all booster versions which have carried the max payload mass
  - Selecting several columns for failed launches in 2015
  - Ranking the count of landing outcomes in specific time frame

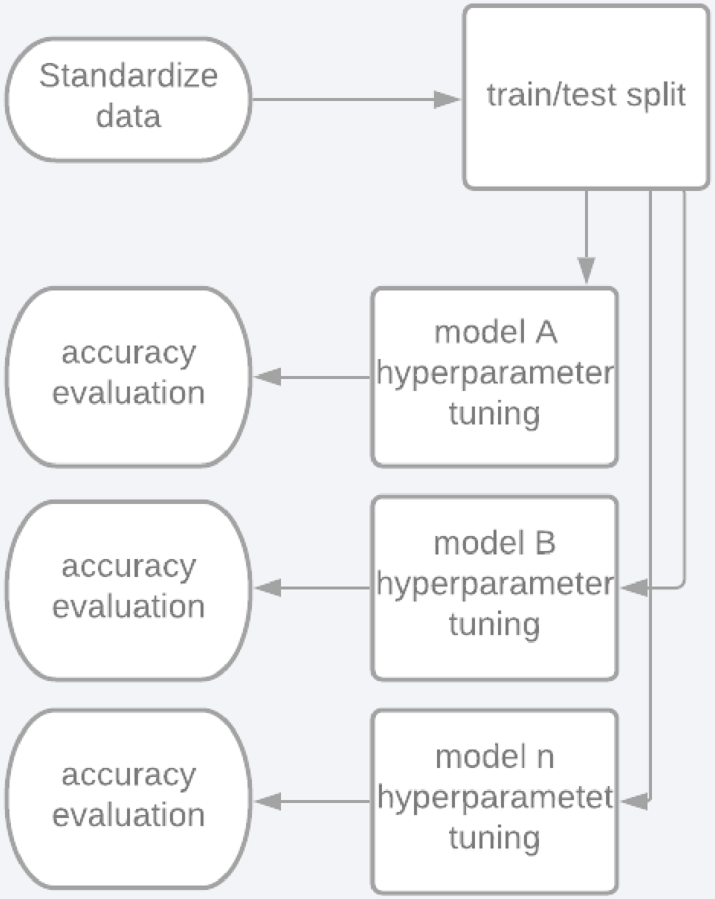
- Jupyter Notebook hosted on GitHub.

Build a Dashboard with Plotly Dash

- An interactive Plotly Dash App was buit to help visually explore the dataset
  - A dropdown list allows the user to select a launch site
  - A pie chart shows the successful to failed launches ratio of a selected site
  - A slider for the payload mass lets the user filter the results
  - A scatter plot shows the selection's booster versions and their class and payload mass
- Plotly Dash App hosted on GitHub

Predictive Analysis (Classification)

To find the best classification model the data was first standardized and then split into a train and test set. For each of the four classification models the optimal hyperparameters were selected using gridsearch and finally the best model was selected evaluating train- and test set accuracy as well as the confusion matrix.



- Jupyter Notebook hosted on GitHub.

Results

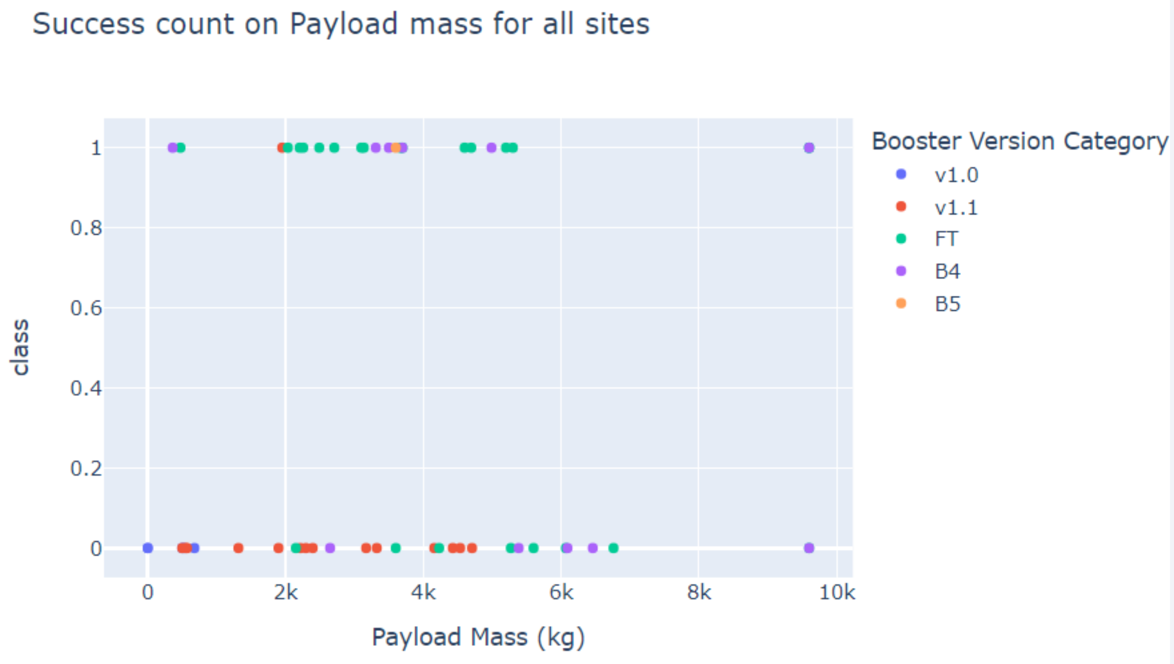
- Exploratory data analysis shows that payload mass and orbit are important fators for the success rate, as well as launch site and booster version used. Launch site proximities and customers are less important

- Interactive analytics have proven the most useful to visualize and uncover the above correlations

- Predictive analysis shows that

a decision tree classifier can

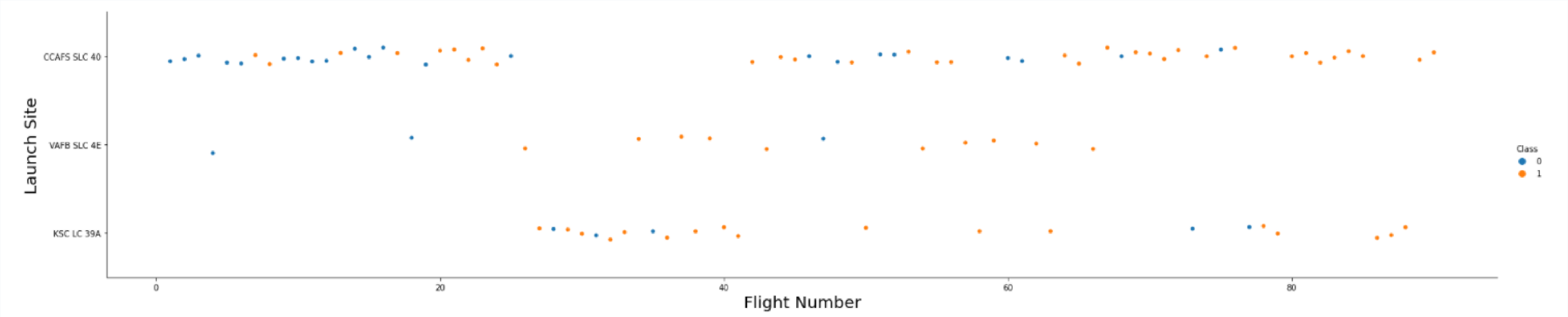
optimally predict a launch outcome



Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

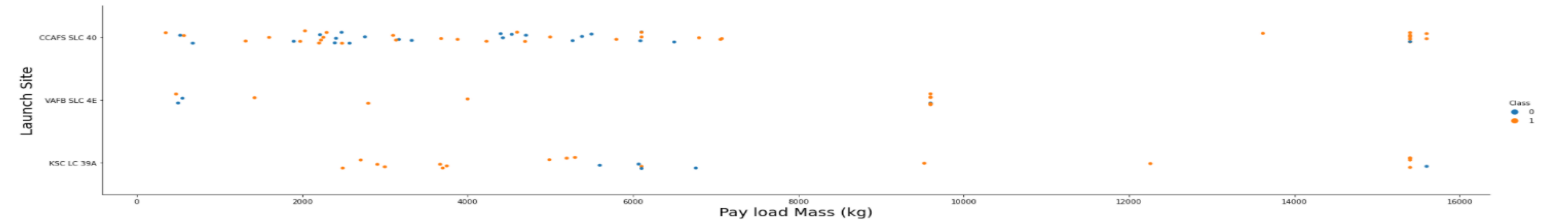


Above plot shows the successes (orange) and failures (blue) for launches at each site.

Observations:

- Results improve over time
- Site VAFB is rarely used but shows stellar success rates

Payload vs. Launch Site



Above plot shows the success of launches for each launch site and payload mass.

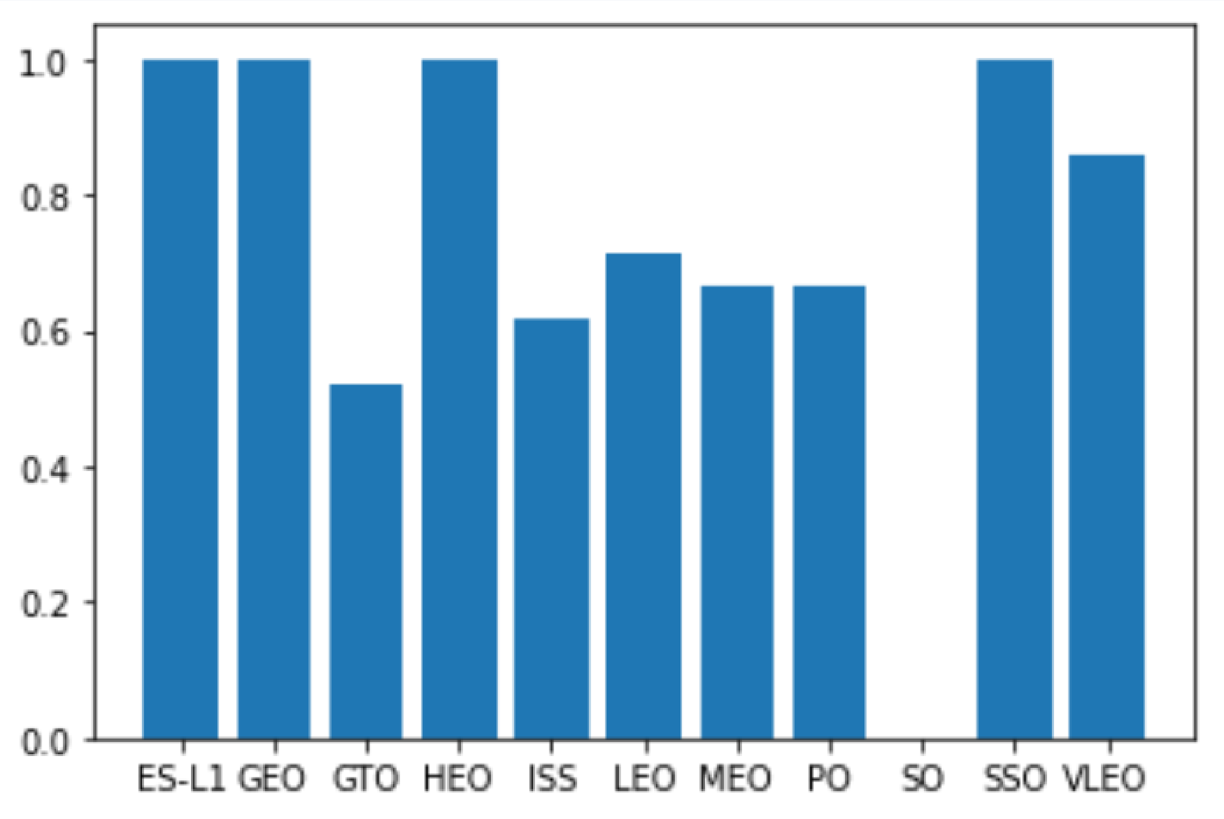
Observations:

- Low pay load mass has high success rates at the KSC launch site
- A higher launch site coincides with a better success rate

Success Rate vs. Orbit Type

The success rates of different orbit types are shown to the right. The higher the bar, the better the rate.

Both orbits with perfect scores can be seen as well as orbits without successes.

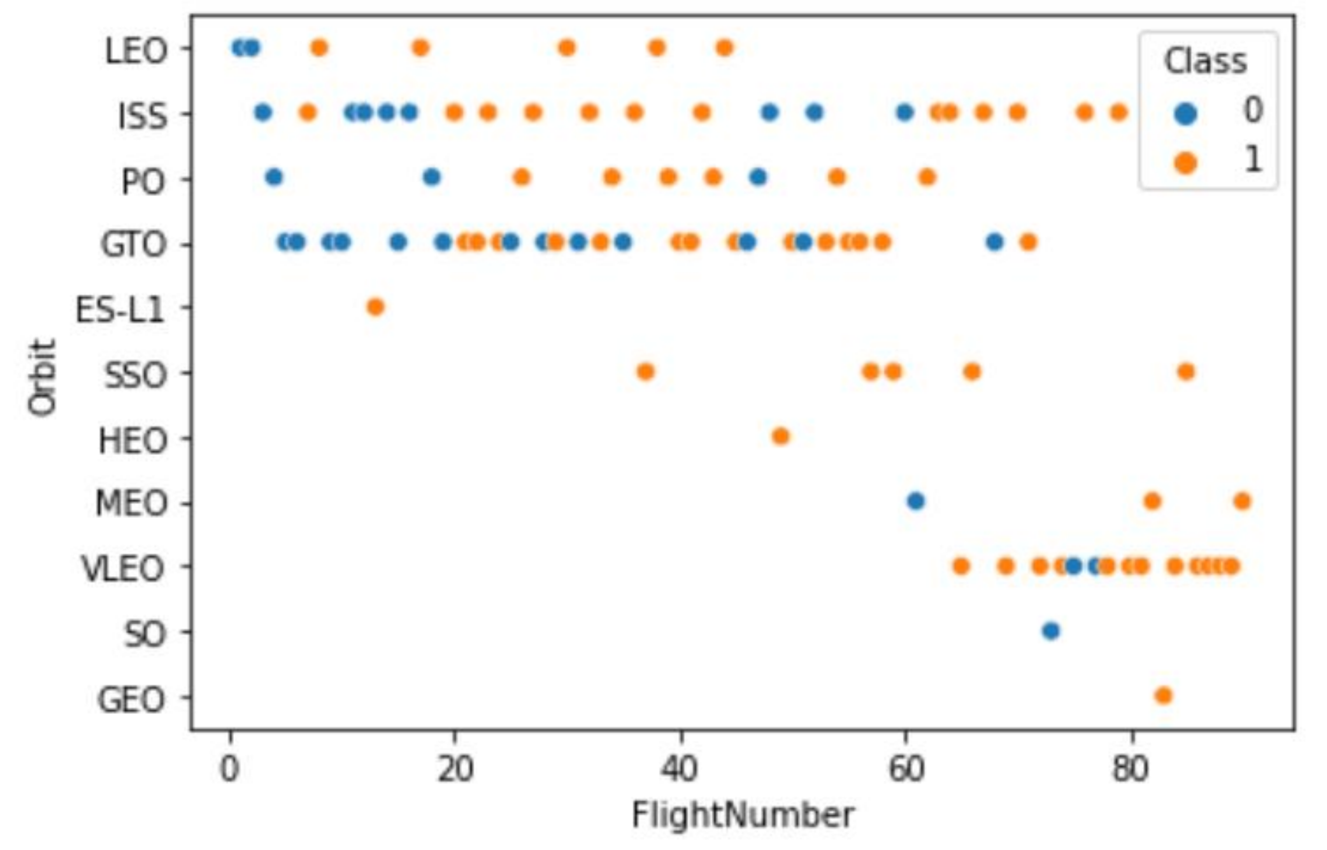


Flight Number vs. Orbit Type

A comparison of flight numbers and orbits shows that gradual shifts have occurred and the VLEO orbit has replaced the LEO orbit.

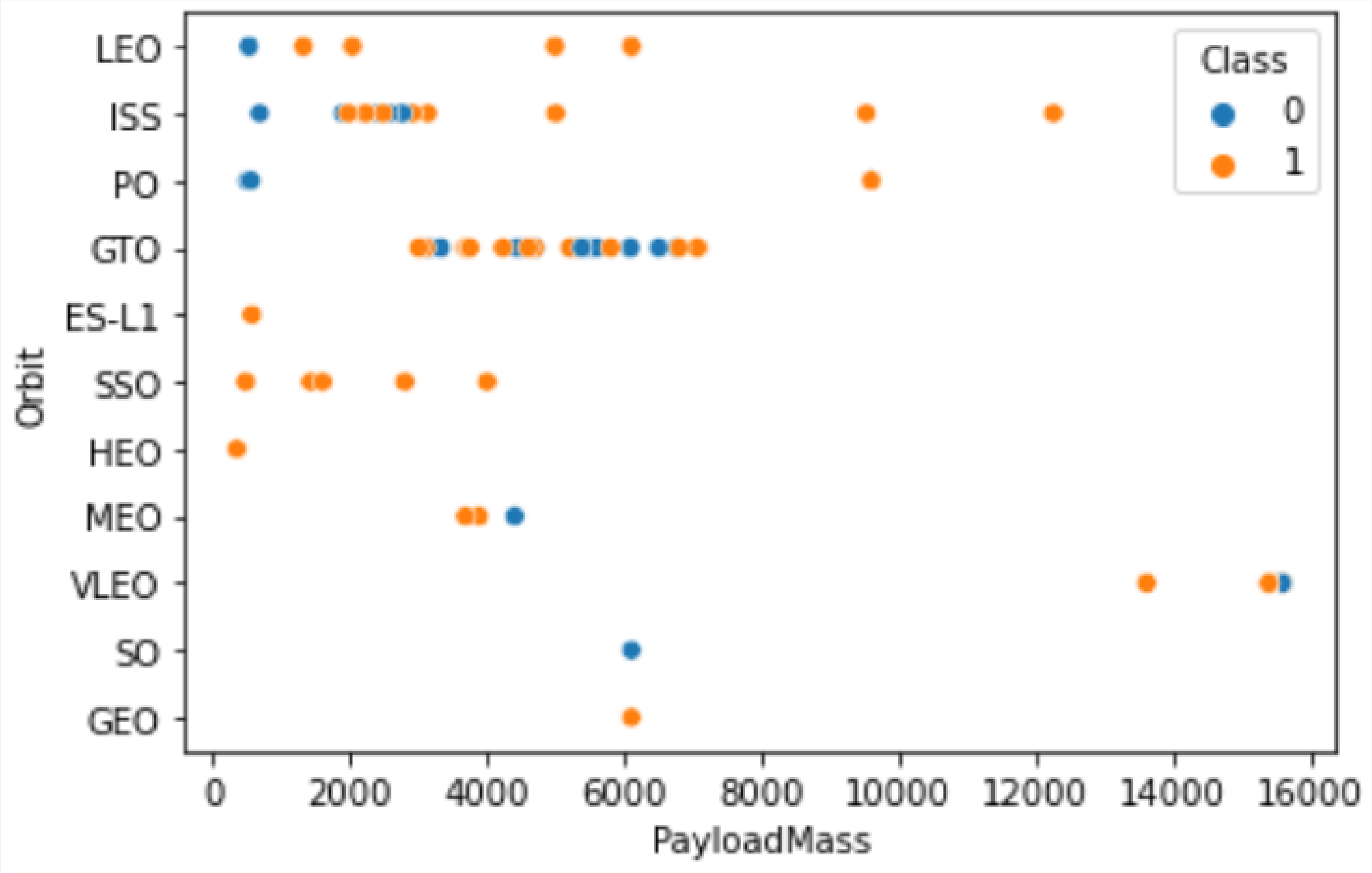
Yet LEO could improve the success rate over time.

The ISS orbit remains popular throughout.



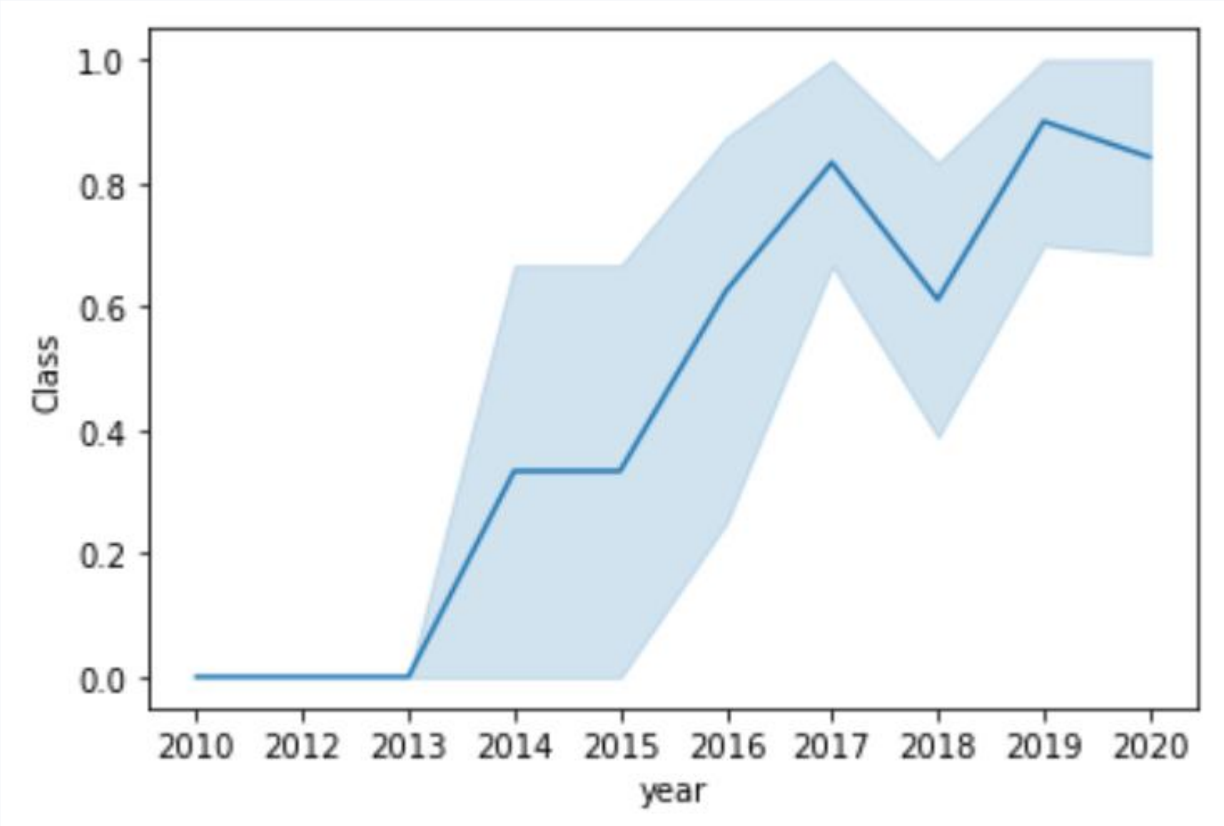
Payload vs. Orbit Type

The comparion of payload mass and orbit type confirms the intuition that most orbits are used for certain ranges of payload only.



Launch Success Yearly Trend

We can see how the first successes could be celebrated in 2014 and since then have overall improved greatly.



All Launch Site Names

The 'distinct' keyword returns the unique values in the launch_site column

```
%%sql

SELECT distinct launch_site
FROM SPACEXDATASET;
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

Launch Site Names Begin with 'CCA'

The 'like' keyword lets us filter by characters and the 'limit' keyword defines the number of returned results.

```
%%sql

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

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b8751
8.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

DATE	time__utc_	booster_version	launch_site	payload	payload_ma
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677

Total Payload Mass

The 'SUM' keyword adds up the individual values,
while the 'LIKE' keywords allows us to filter by customer (here: NASA)

```
%%sql

SELECT SUM (payload_mass__kg_)
FROM SPACEXDATASET
WHERE customer LIKE ('NASA (CRS)')
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
45596

Average Payload Mass by F9 v1.1

Working with the 'AVG' and like keywords

```
%%sql

SELECT AVG(payload_mass__kg_)
FROM SPACEXDATASET
WHERE booster_version LIKE ('F9 v1.1%')
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b8751
8.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
2534

First Successful Ground Landing Date

Using the 'MIN' keyword to find the earliest date

```
%%sql

SELECT MIN(Date)
FROM SPACEXDATASET
WHERE landing__outcome LIKE ('Success (ground pad)')
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b8751
8.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Using two 'AND' operators to define a range

```
%%sql

SELECT distinct booster_version
FROM SPACEXDATASET
WHERE landing__outcome LIKE ('Success (drone ship)')
AND payload_mass__kg_ > 4000
AND payload_mass__kg_ < 6000;
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

Using 'GROUP BY' to get total numbers:

```
%%sql

SELECT mission_outcome, COUNT(mission_outcome)
FROM SPACEXDATASET
GROUP BY (mission_outcome)
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b8751
8.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

mission_outcome	2
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

Using a subquery to show max payload mass

Carried by each booster version

```
%%sql

SELECT booster_version, payload_mass__kg_
FROM SPACEXDATASET
WHERE payload_mass__kg_ = (
    SELECT MAX(payload_mass__kg_)
    FROM SPACEXDATASET)
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b8751
8.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

2015 Launch Records

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

```
%%sql
SELECT landing__outcome, booster_version, launch_site, date
FROM SPACEXDATASET
WHERE landing__outcome LIKE ('Failure (drone ship)')
AND year(Date) = 2015;
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b8751
8.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

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

Ranking 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

```
%%sql

SELECT landing__outcome, COUNT(landing__outcome)
FROM SPACEXDATASET
WHERE date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY (landing__outcome)
ORDER BY COUNT(landing__outcome) DESC
```

```
 * ibm_db_sa://ydh91868:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

landing__outcome	2
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

Launch site locations

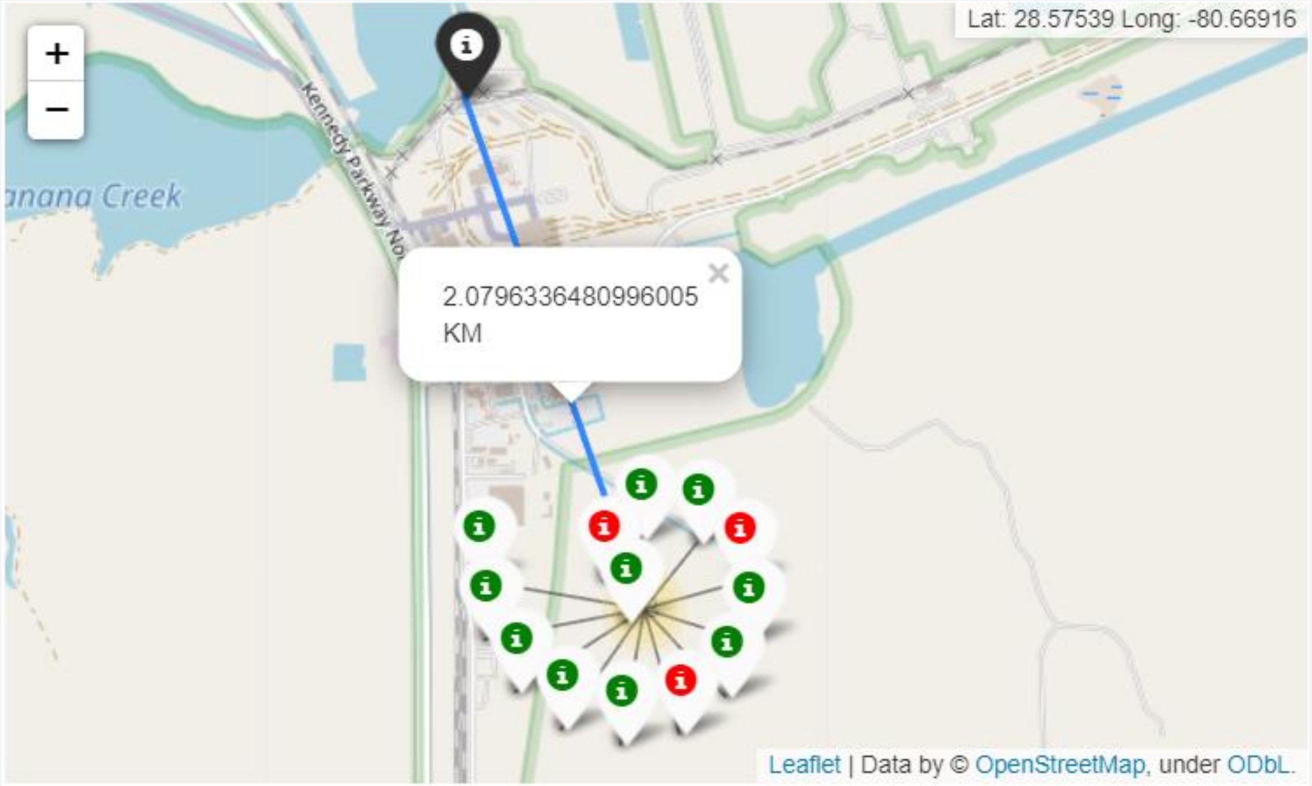
Most launches took place at Cape Canaveral in Florida while other took place on the US West Coast:

Visualizing successful launches per site

By coloring successful/failed launches and grouping them by launch site we can quickly visualize how successful a certain launch site is:

Uncovering proximities to points of interest

Calculating the distance to various points of interest (here: a railway station) can uncover correlations between success rate and launch site proximities:

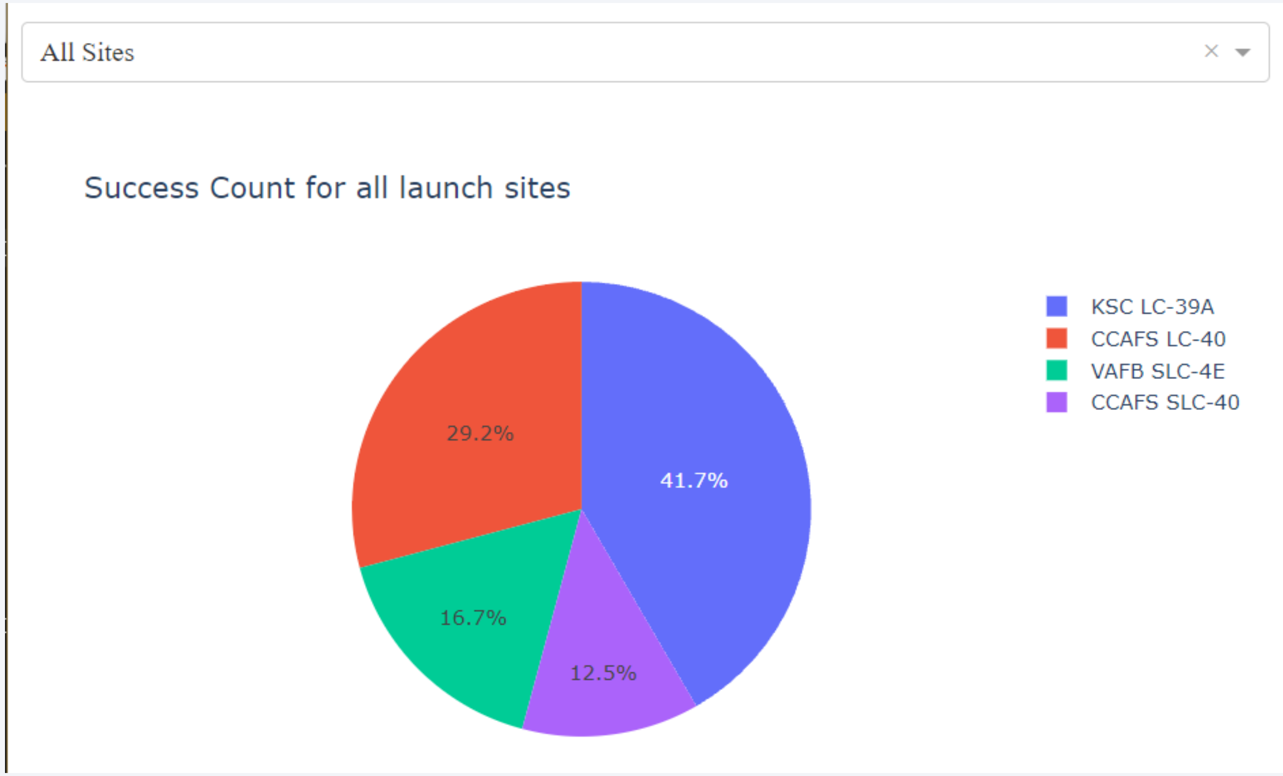


Section 5

Build a Dashboard with Plotly Dash

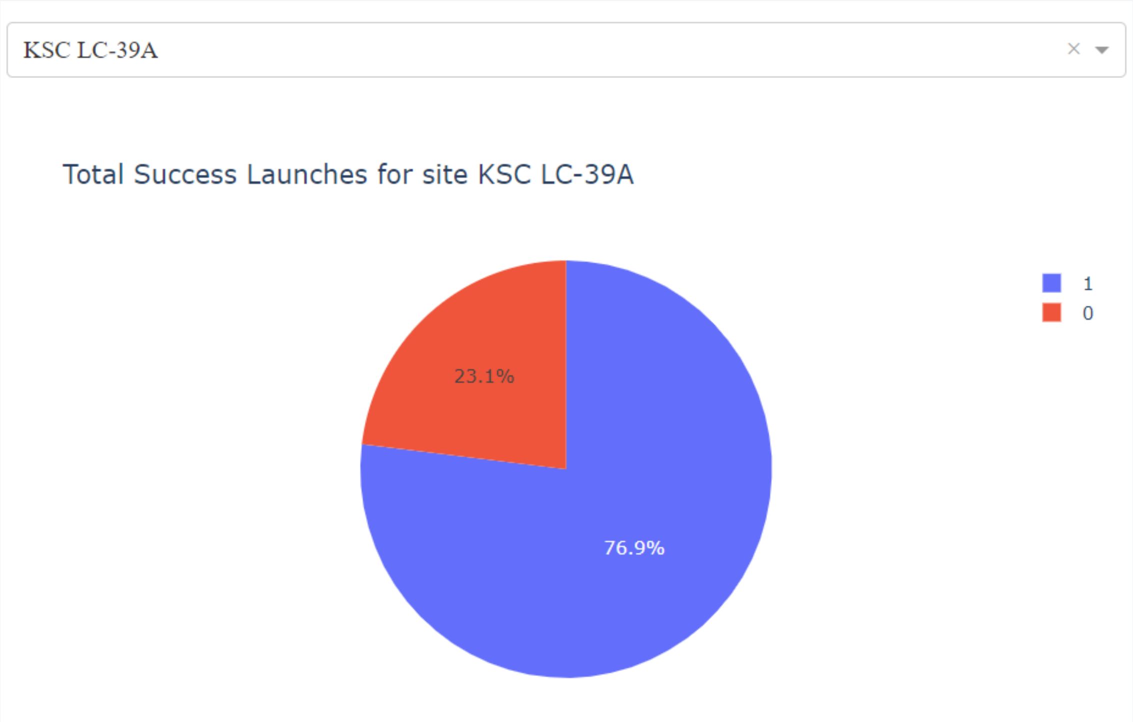
Which launch sites are the most successful?

Almost 40% of successful launches were made from site KSC:



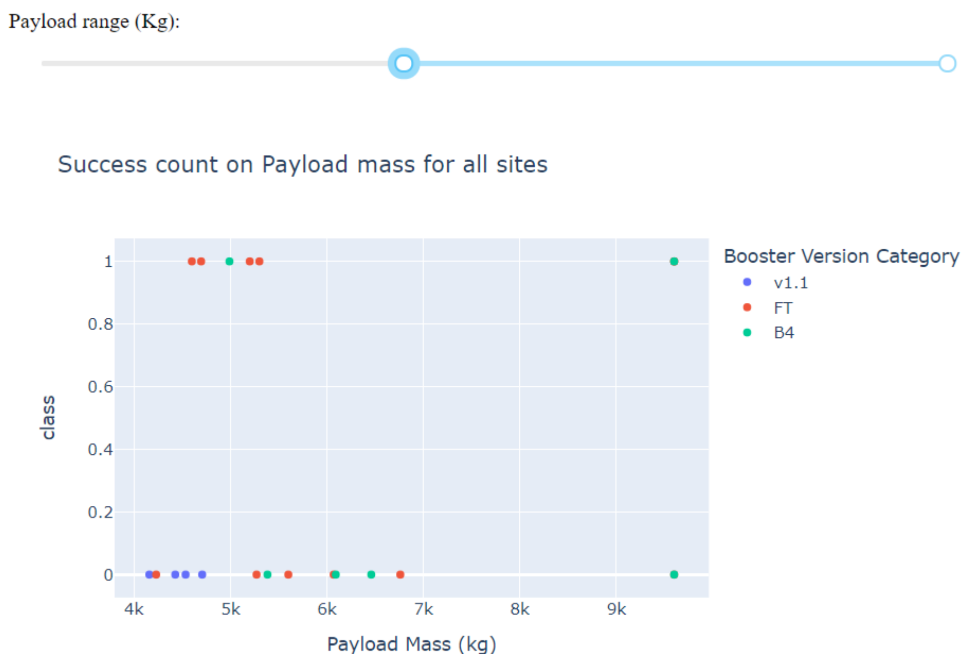
Closer look at launch site KSC

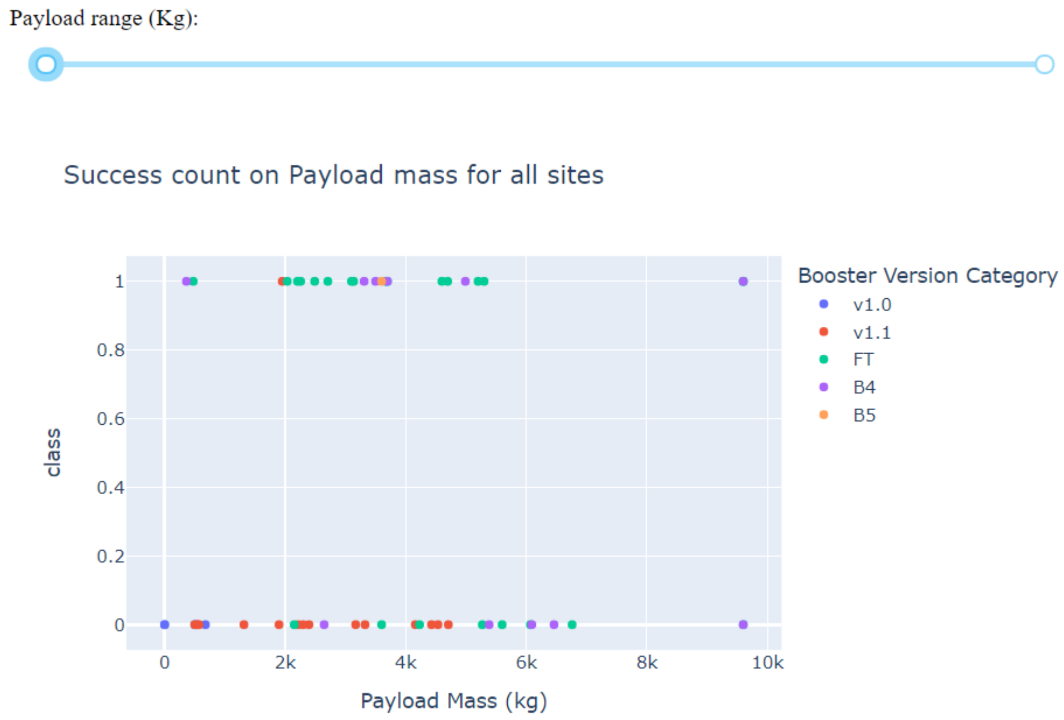
Site KSC has the highest success rate of all launch sites as less than one quarter of launches failed to return the first stage of the rocket.

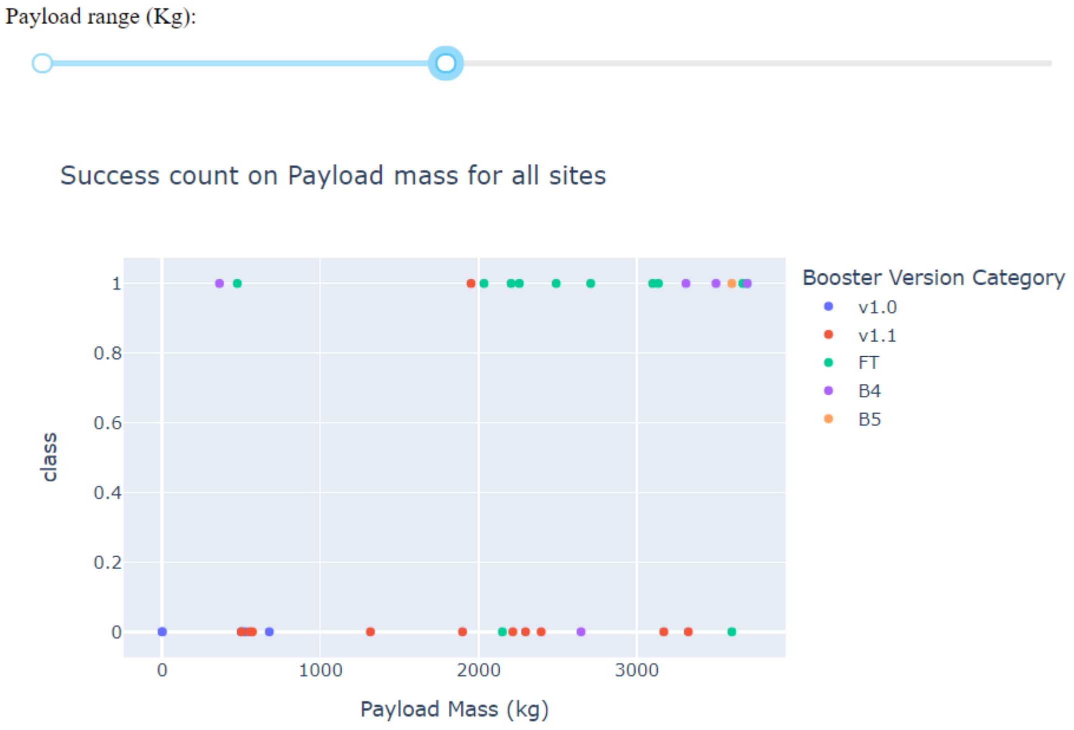


Payload mass as a factor of success

Payloads on either end of the scale show lower success rates





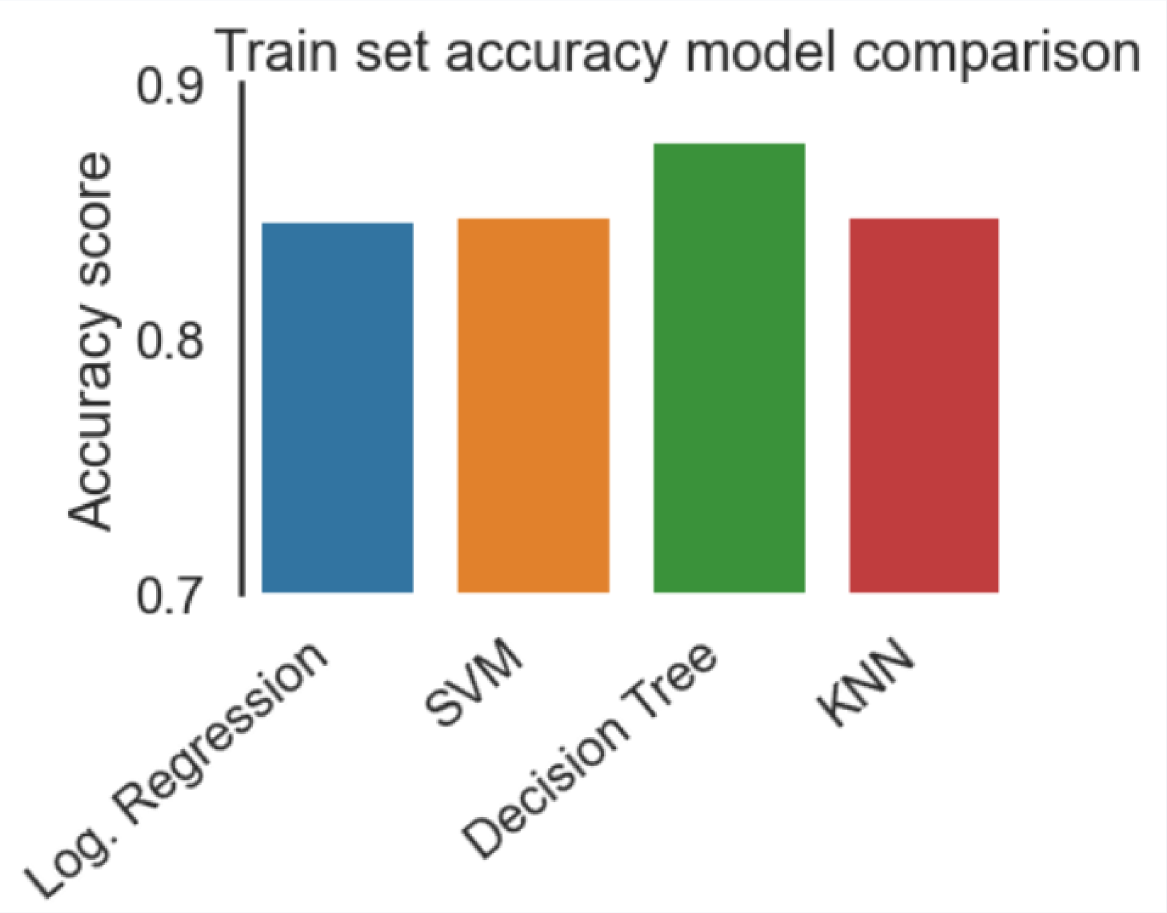


Section 6

Predictive Analysis (Classification)

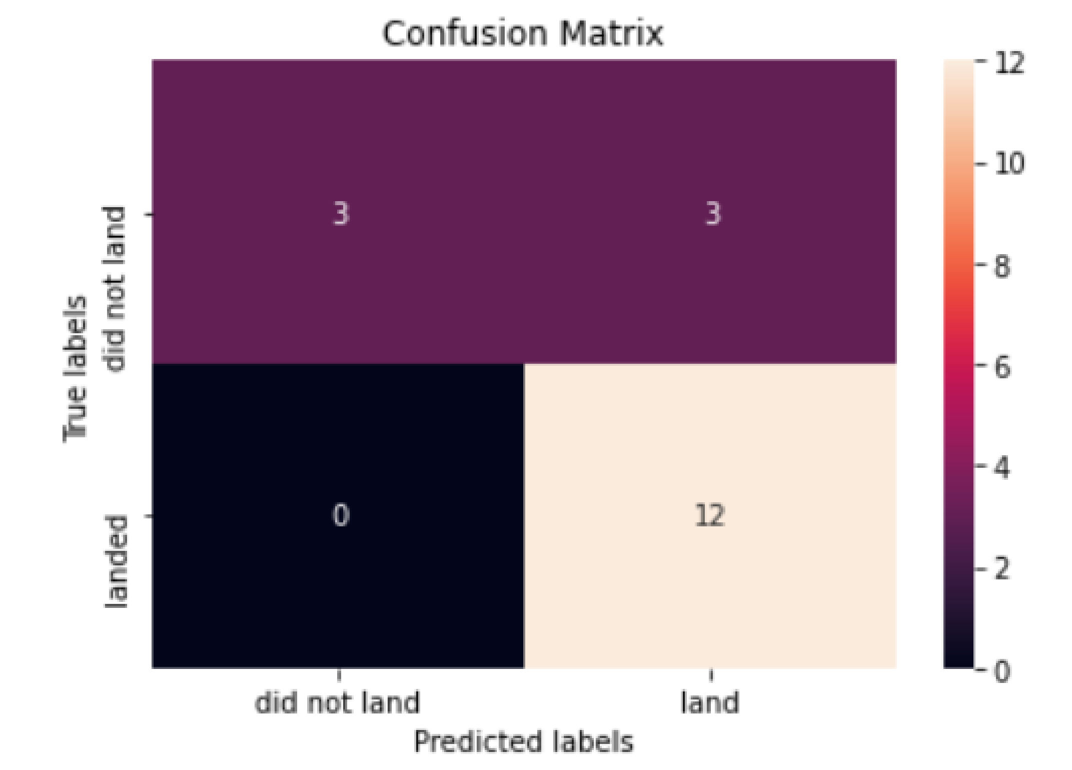
Classification Accuracy

After testing four different machine learning models and tuning the hyperparameters, the decision tree classifier came up with the best train set accuracy.



Confusion Matrix

The confusion matrix shows that out of 18 observed launches 15 could be predicted correctly. There are no false negatives but there are 3 false positives.



Conclusions

- The orbit and payload mass are the most important factors to determine the success of a launch
- SpaceX has gotten more succesful with time and this trend can be assumed to continue, making a model that can predict failures more valuable to competitors
- As our model contains no false negatives it is a powerful tool to predict when the first stage of a rocket will not land. This allows to bid against SpaceX with relatively little risk.


Thank you!