IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Javier de la Rubia
12th August, 2025

Outline

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

Executive Summary

Our goal is to analyze and extract some conclusions about SpaceX Falcon 9 rocket launches.

We'll use data grabbed from SpaceX public API and by using web-scrapping from Wikipedia.

- Methodologies
  - Data collection via API calls and Web scraping
  - Clean and transform data to be consumed
  - Perform some Exploratory Data Analysis
  - Build charts and maps to make our findings more visible to stake holders
- Summary of all results

Introduction

- Project background and context
  - SpaceX achieved bringing the costs down when sending payloads into space
  - Cost is the main barrier to other competitors and SpaceX recovery method has been revolutionary
  - SpaceX “low” costs rely on recuperating the first stage of the rockets
- Problems you want to find answers
  - Define whether the first stage of a Falcon 9 rocket will land successfully
  - Determine which and how different parameters affect to this outcome
  - Determine if there are correlations between the launch sites and their success

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data is collected through API calls to the SpaceX public API
  - Some more data for Falcon 9 and falcon Heavy launches is scrapped from Wikipedia
- Perform data wrangling
  - Clean and keep only Falcon 9 data, filtering out the rest.
- Exploratory data analysis (EDA) using tables, plots and SQL queries
- Interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Tried Logistic regression, Support Vector Machine, Decision Tree & K-Nearest Neighbor models on test data
  - Extract models score and accuracy indexes

Data Collection

The following data is acquired by sending get requests to SpaceX’s official API:

- Rockets,
- Launch Sites,
- Payloads, and
- Cores

All this data is assembled into one data frame and save to disk for further analysis.

Information about launch dates, boosters, payload mass, orbits, launch sites, outcome, customers, etc. is grabbed from Wikipedia and filtered from the multiple tables on this site.

Data Collection – SpaceX API

Flowchart

1. API call to SpaceX endpoint
2. Save response to a pandas data frame
3. Normalize & complete data using the helper functions
4. Create a unified data frame
5. Save data frame to disk

[Github URL: 1. Data Collection Notebook]

Data Collection - Scraping

Flowchart

1. Create and make a get request to Wikipedia
2. Create a BeatifulSoup object and pass on the previously obtained content
3. Filter content tables to leave only the one with the right data
4. Extract column names
5. Save information to a dictionary and this to a pandas data frame
6. Save data frame to disk for future use

[Github URL: 3. Data Scraping notebook]

Data Wrangling

Flowchart

1. Select only Falcon 9 data
2. Normalize ‘payload_mass’ null values to its mean value
3. Save data to disk

[Github URL: 2. Data Wrangling Notebook]

EDA with Data Visualization

Show data tables for:

- Launch sites
- Outcome
- Orbit
- Success rate

Graph:

- Flight number vs Payload mass
- Flight number vs Launch site
- Flight number vs Orbit
- Relation between Orbit and Success rate
- Payload vs Orbit
- Launch success yearly trend
- Find & graph correlations between the different features

[Github URL: 4, EDA Basic Analysis Notebook]

[Github URL: 6. EDA F9 First Stage Landing Predictions]

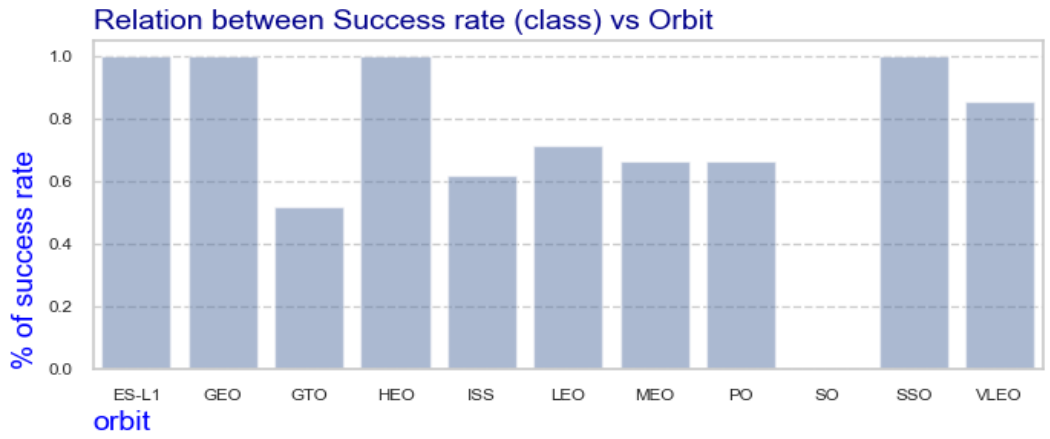
EDA with Data Visualization: examples

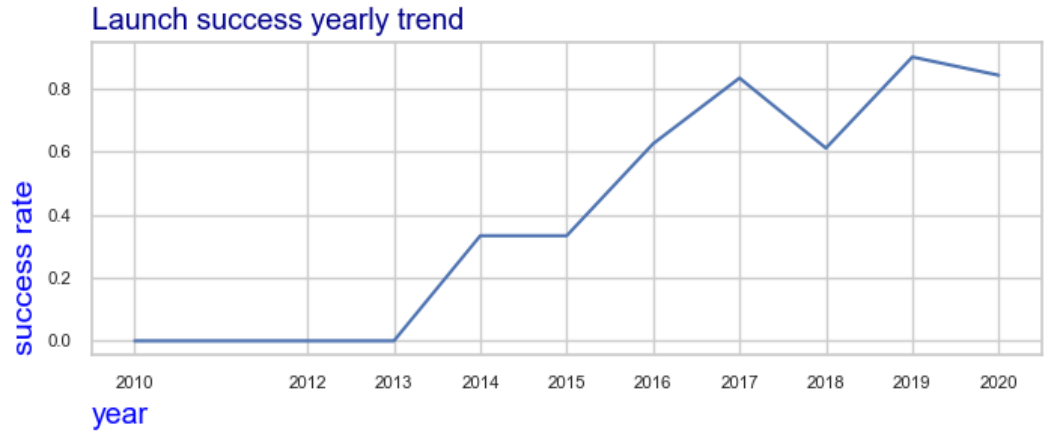
```
# number of launches to each orbit
df["orbit"].value_counts()
```

orbit	# count
GTO	27
ISS	21
VLEO	14
PO	9
LEO	7
SSO	5
MEO	3
HEO	1
ES-L1	1
SO	1
GEO	1

```
# itemized launch outcomes
df["outcome"].value_counts()
```

outcome	# count
True ASDS	41
None None	19
True RTLS	14
False ASDS	6
True Ocean	5
False Ocean	2
None ASDS	2
False RTLS	1





EDA with SQL

SQL Queries launched for our Exploratory Analysis:

- Unique launch sites
- Unique boosters
- Total payload mass carried by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- First successful landing on a ground pad
- Boosters successfully landed in dron ship and payload mass between 4000 and 6000
- Total number of successful and unsuccessful mission outcomes
- Boosters that have carried the maximum payload mass
- Month names, failure landing outcomes in drone ship, boosters, and launch site for the months in year 2015
- Count of landing outcomes between 2010-06-04 and 2017-03-20 in descending order

[Github URL: 5. EDA SQL Analysis]

EDA with SQL: examples

```
%%sql
-- # boosters that have carried the maximum payload mass
SELECT Booster_Version FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT max(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

```
 * sqlite:///data/spacex_data.sqlite3
Done.
```

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

Falcon 9 boosters

```
%%sql
-- # count of landing outcomes between 2010-06-04 and 2017-03-20 in descending order
SELECT Landing_Outcome, COUNT(Landing_Outcome) AS outcome_count FROM SPACEXTBL
GROUP BY Landing_Outcome
ORDER BY outcome_count DESC;
```

```
 * sqlite:///data/spacex_data.sqlite3
Done.
```

Landing_Outcome	outcome_count
Success	38
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Failure	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1
No attempt	1

Landing outcomes between 2010-06-04 and 2017-03-20

Build an Interactive Map with Folium

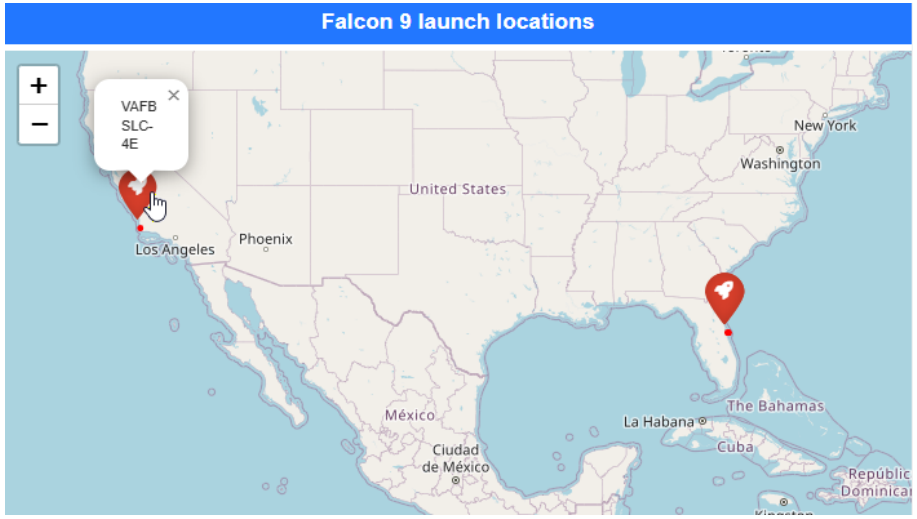
Given all the Falcon 9 launch locations data, plot interactive maps to:

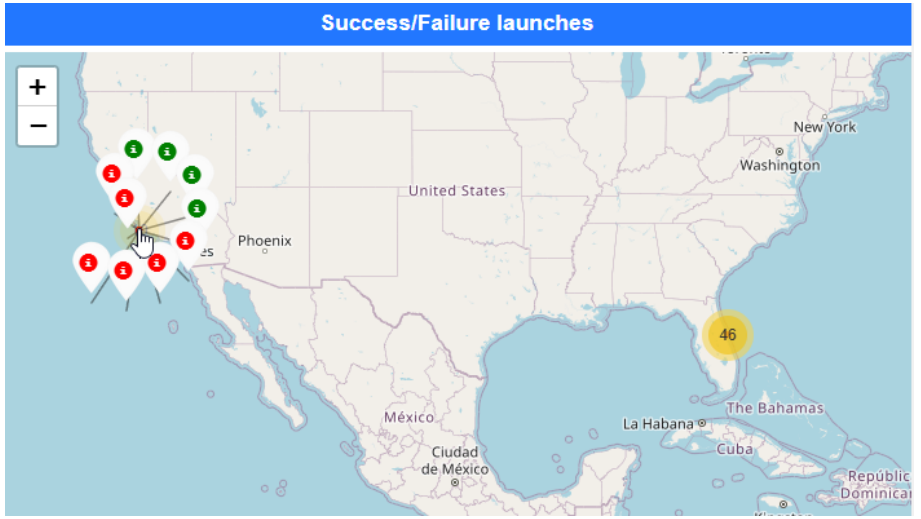
- Show Falcon 9 launch locations on a map
- Show Success/Failure rate on each location
- Show launch sites and distance to their closest coastline

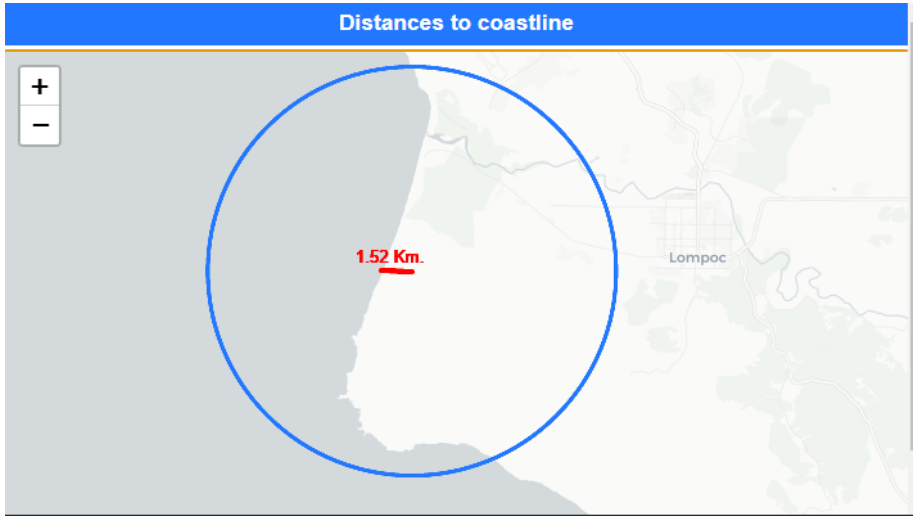
Showing distances to closest cities will follow a similar process

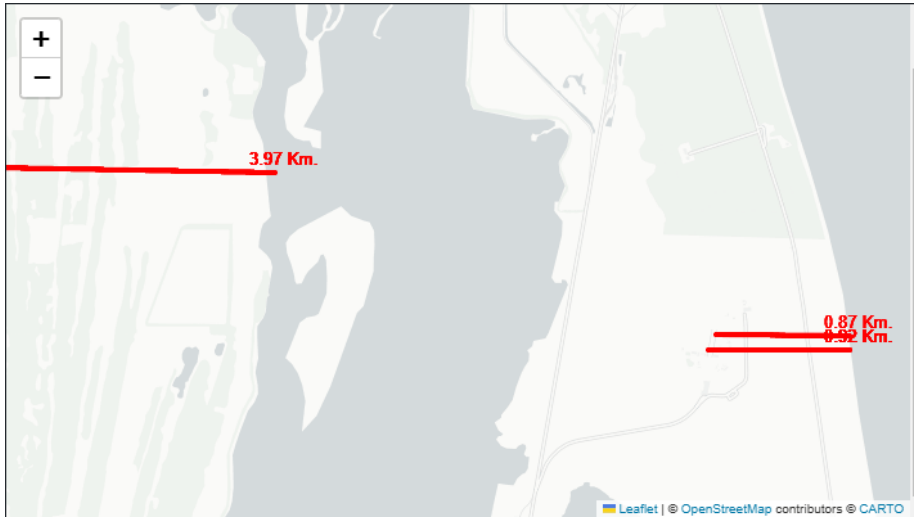
[Github URL: 7. EDA Folium Maps Analysis]

Interactive Map with Folium: examples









Build a Dashboard with Plotly Dash

Dash application that allows us to see a pie chart containing the number of succeeded launches from all sites or from each individual site.

Also, we can choose the range of payloads that we want to see.

Along with the dash application, there is a notebook which answers on a table format the same following questions.

1. Which sites show the highest number of successful and failed launches?
    - KSC LC-39A with 10 successful launches off 13, 76.92% success ratio
    - CCAFS LC-40 with 19 failed launches off 26, 73.08% failure ratio
2. Which payload range has the highest number of successful and failed launches?
    - 2,000 – 5,000 kg, with 21 combined succeeded launches, a 62% success ratio
    - 0 – 2,000 kg with 77%, and 5,000 – 7,000 kg with 78%, show the highest failure rates
3. Which Falcon 9 booster version has the highest launch success rate (v1.0, v1.1, FT, B4, or B5)?
    - FT with a 66.67%

[Github URL: 9. Analysis Notebook] & [Github URL: 9 server code]

Predictive Analysis (Classification)

Workflow

1. Load data
2. Standardize predictions data
3. Split into train and test sets
4. Try the following models:
    1. Logistic Regression
    2. Support Vector Machine
    3. Decision Tree
    4. K-Nearest Neighbor
5. Show results

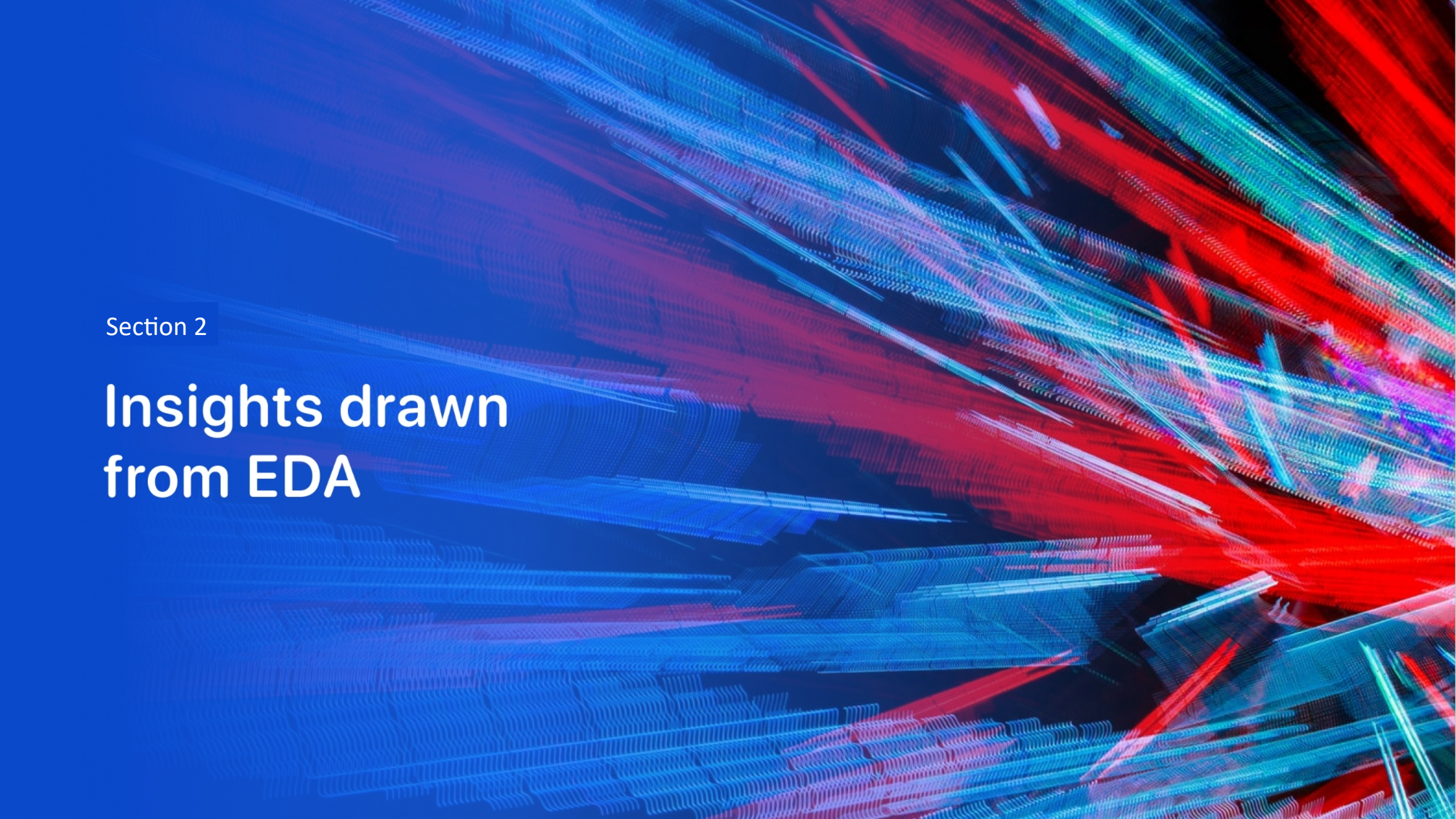
    Support Vector Machine algorithm performs best on test data and train data combined. See table below.

[Github URL: 8. ML: Predictive Analysis Notebook]

algorithm	# test_data_score_pct	# train_data_accuracy
LR	83.3333	82.1429
SVM	83.3333	84.8214
Decision Tree	66.6667	88.5714
KNN	66.6667	83.3929

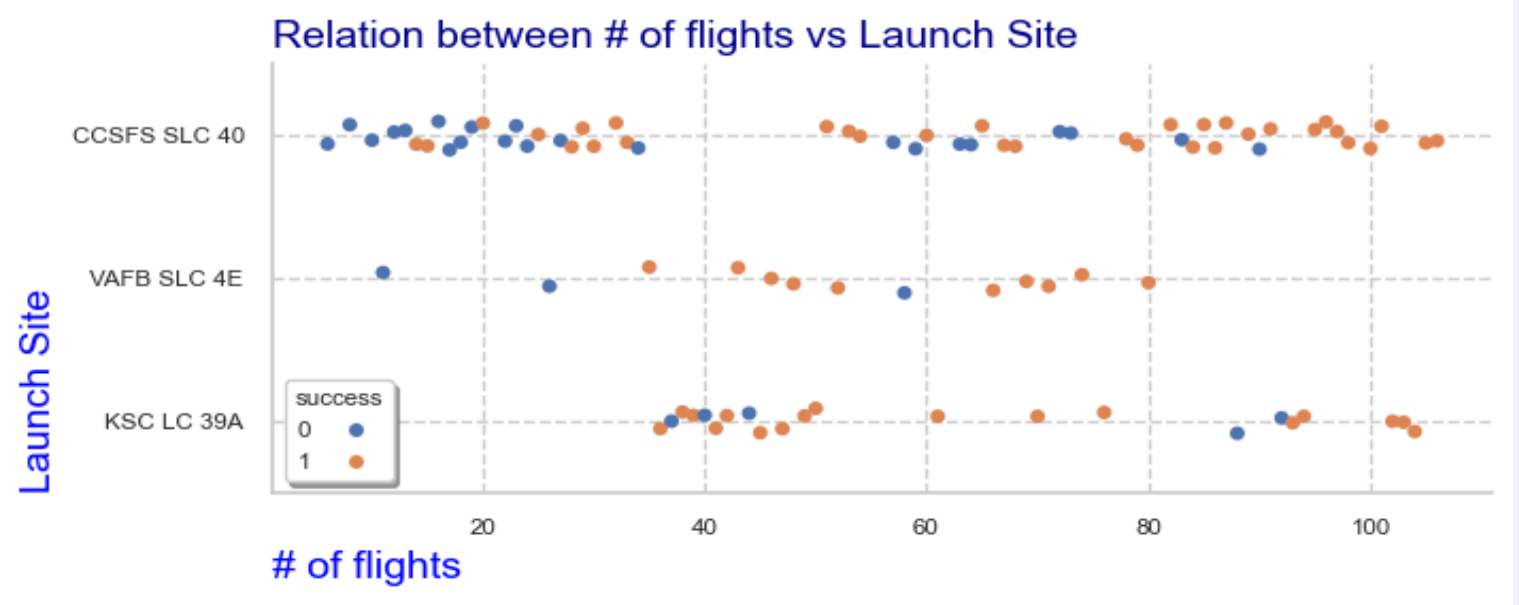
Results

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


Section 2
Insights drawn
from EDA

Flight Number vs. Launch Site

Scatter plot of number of lights, column flight_number, versus launch site
0 = failed launch
1 = successful launch

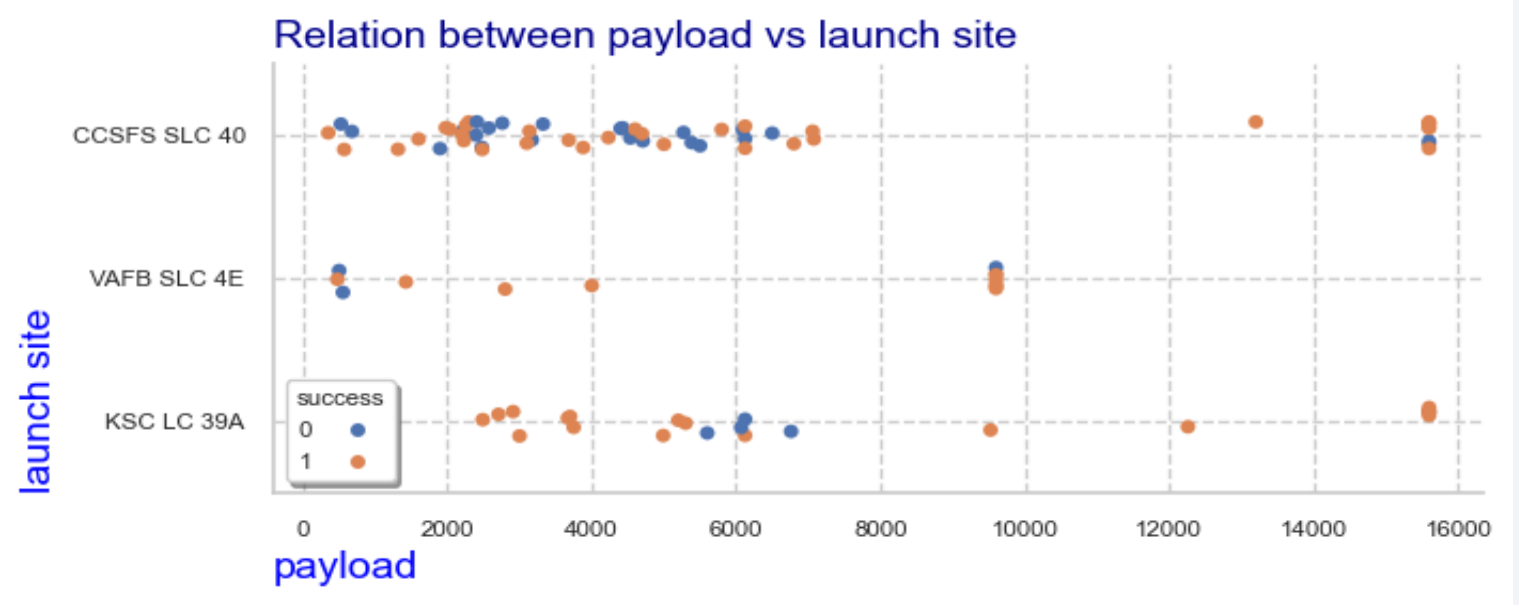


Insights:
The number of succesful launches increases with the number of launches.
VAFB show the best ratio of succeeded launches.
KSC doesn't show information below almost 40 launches.

[Github URL: 5. SQL: EDA Notebook]

Payload vs. Launch Site

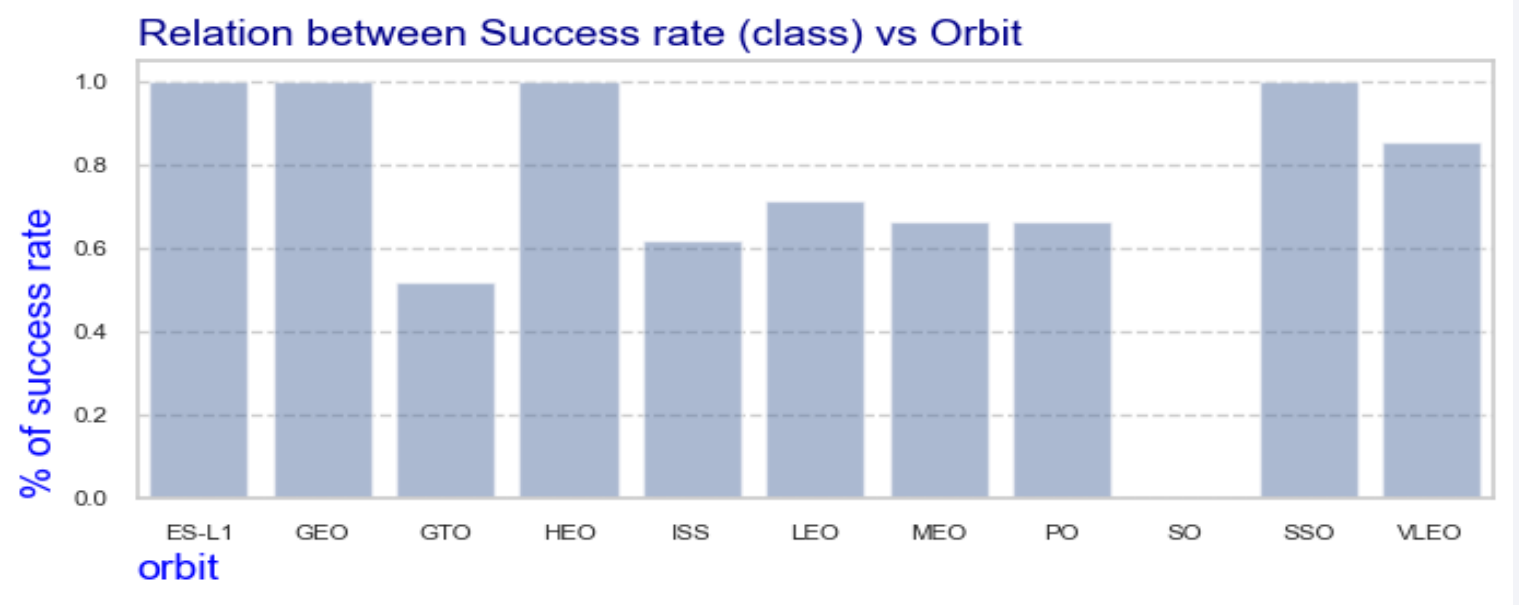
Scatter plot of payload mass versus launch site
0 = failed launch
1 = successful launch



Insights:
The number of failed launches happen with smaller payloads.
CCSFS and KSC sites concentrate on launching smaller payloads.
VAFB doesn't show payloads above the 10,000 kg mark.

Success Rate vs. Orbit Type

orbit	succeeded_launches_pct
ES-L1	1.0
GEO	1.0
HEO	1.0
SSO	1.0
VLEO	0.86
LEO	0.71
PO	0.67
MEO	0.67
ISS	0.62
GTO	0.52
SO	0.0

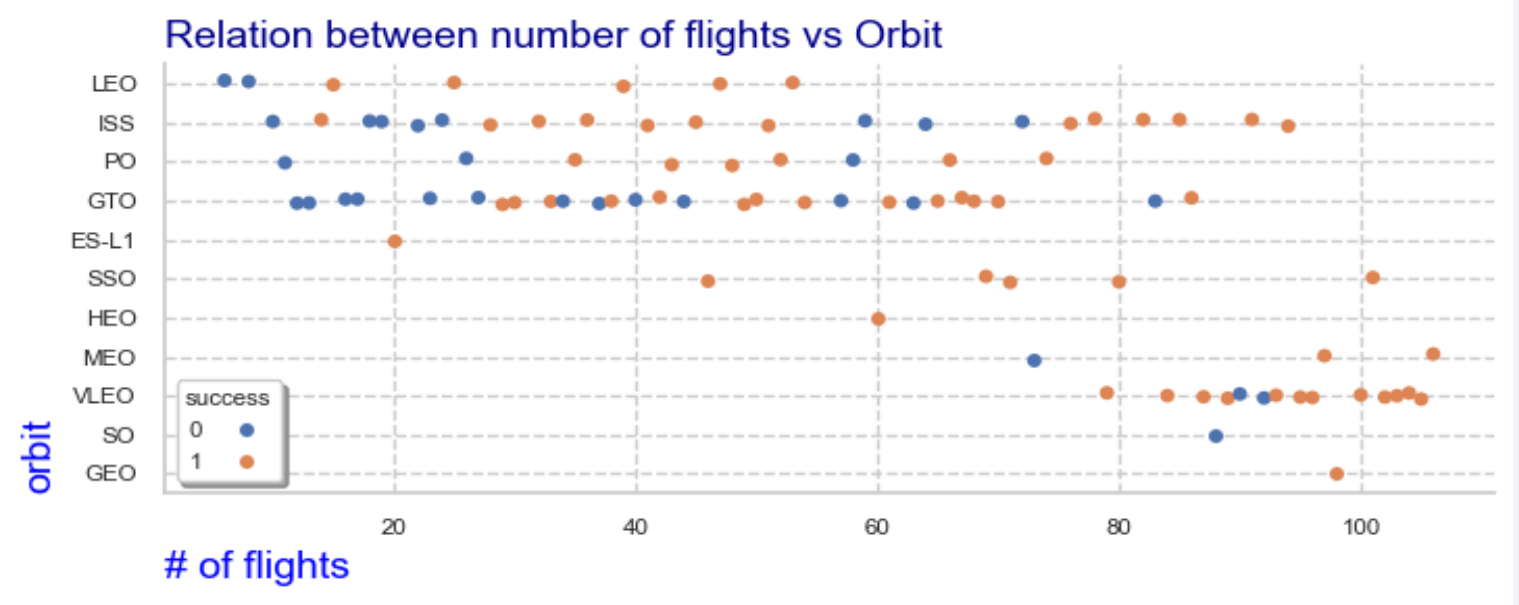


Insights:
SO orbit show no launches.
ES-L1, GEO, HEO and SSO orbits show a 100% of success launch rate, while VLEO is above the 80% mark
GTO orbit has the lowest success rate.

[Github URL: 5. SQL: EDA Notebook]

Flight Number vs. Orbit Type

orbit	successful_launches	# of_flights	success_ratio
ES-L1	1	1	1.0
GEO	1	1	1.0
HEO	1	1	1.0
SSO	5	5	1.0
VLEO	12	14	0.86
LEO	5	7	0.71
PO	6	9	0.67
MEO	2	3	0.67
ISS	13	21	0.62
GTO	14	27	0.52
SO	0	1	0.0

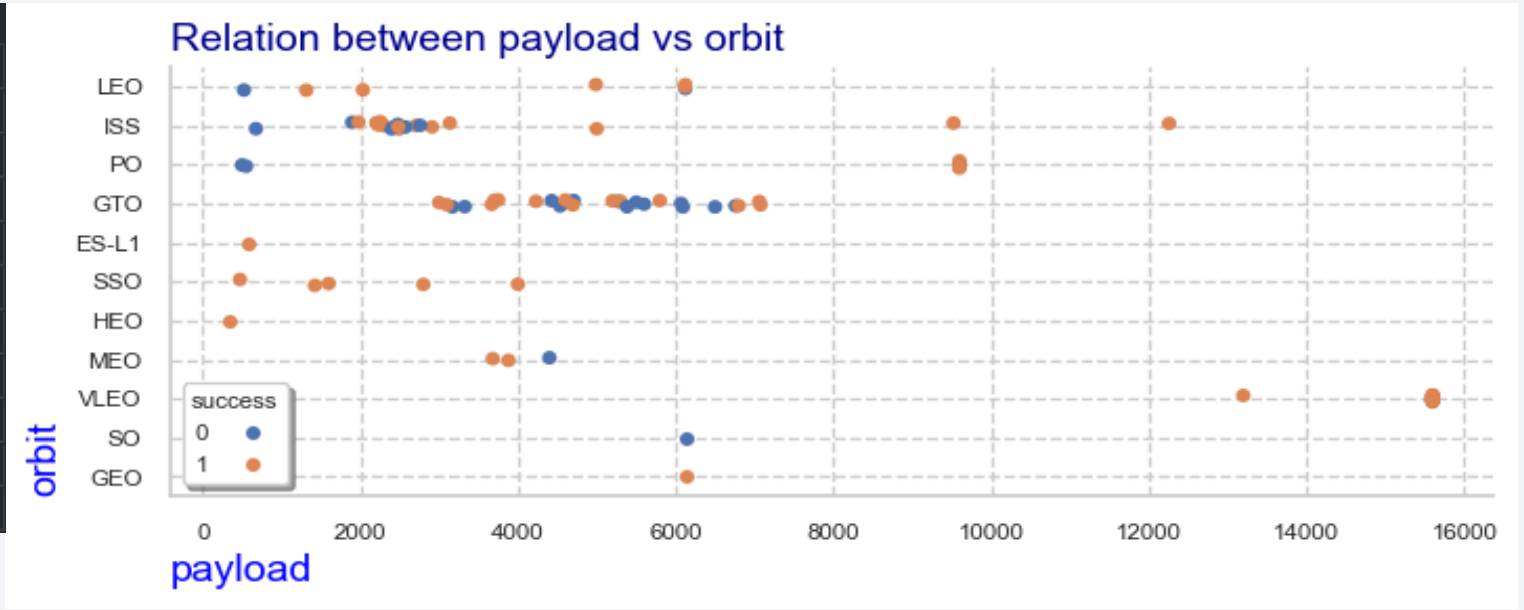


Insights:
SO orbit show no launches.
ES-L1, GEO, HEO and SSO orbits show a 100% of success launch rate. However, the number of flights is minimum. VLEO is above the 80% mark with a much higher number of launches.
GTO orbit has the lowest success rate of all, but with the highest number of attempted launches.

[Github URL: 5. SQL: EDA Notebook]

Payload vs. Orbit Type

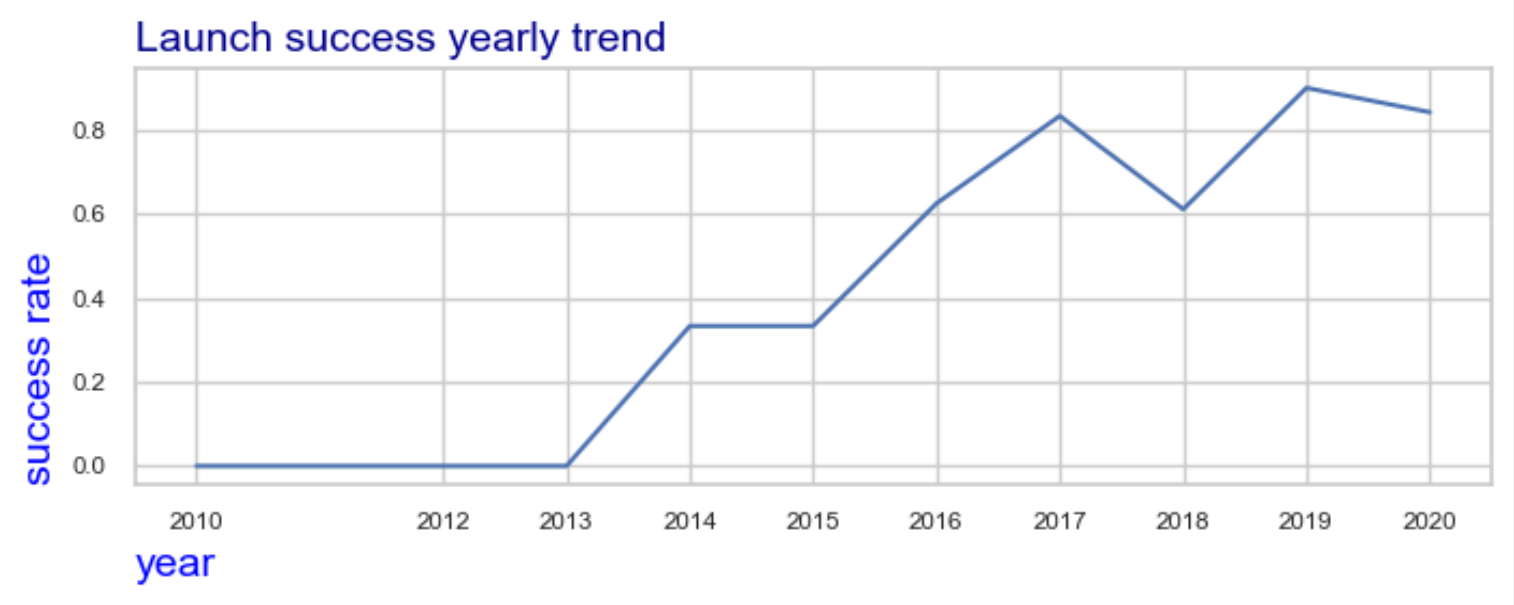
orbit	# payload	# launches	# successful_launches	# success_ratio
ES-L1	570.0	1	1	1.0
GEO	6123.55	1	1	1.0
HEO	350.0	1	1	1.0
SSO	10300.0	5	5	1.0
VLEO	216000.0	14	12	0.86
LEO	27235.64	7	5	0.71
PO	68253.0	9	6	0.67
MEO	11961.0	3	2	0.67
ISS	68878.7	21	13	0.62
GTO	135323.85	27	14	0.52
SO	6123.55	1	0	0.0



Insights:
There is no clear relation between payload mass and a successful launch. However, we can state that orbits with a higher payload show results above a 50%.
Orbits with fewer launches, and smaller payloads, show, in general but not in all cases, success of a 100%.
This is more visible in the tabular version of the data.

[Github URL: 5. SQL: EDA Notebook]

Launch Success Yearly Trend



Insights:
During the first 3 years of the analysis the number of attempts keeps the line flat on a 0% success rate.
The trend starts to go upwards from 2014 on.
It is between 2016 and 2020 that the ratio of success stabilizes within an acceptable 60 to 90%

[Github URL: 5. SQL: EDA Notebook]

All Launch Site Names

```
# Unique launch sites
%sql SELECT DISTINCT Launch_Site AS site FROM SPACEXTBL;
```

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

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

By including the 'DISTINCT' keyword in our query, it will return only the unique values of the column chosen, 'Launch_Site' in this instance.

[Github URL: 5. SQL: EDA Notebook]

Launch Site Names Begin with 'CCA'

```
%%sql
--# Launch sites that begin with CCA
SELECT Date, Booster_Version, Launch_Site, Orbit, Customer, Mission_Outcome, Landing_Outcome
    FROM SPACEXTBL
    WHERE Launch_Site LIKE 'CCA%'
    LIMIT 5;
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

Date	Booster_Version	Launch_Site	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	F9 v1.0 B0003	CCAFS LC-40	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	F9 v1.0 B0004	CCAFS LC-40	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	F9 v1.0 B0005	CCAFS LC-40	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	F9 v1.0 B0006	CCAFS LC-40	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	F9 v1.0 B0007	CCAFS LC-40	LEO (ISS)	NASA (CRS)	Success	No attempt

- LIKE helps finding the precise string that we're looking form, "CCA*", the the Launch_Site column.
- LIMIT 5, stops after finding the first 5 results

[Github URL: 5. SQL: EDA Notebook]

Total Payload Mass

```
%%sql
--# Payload mass carried by boosters launched by NASA (CRS)
SELECT sum(PAYLOAD_MASS__KG_) AS nasa_payload_mass
    FROM SPACEXTBL
    WHERE customer LIKE 'NASA (CRS)%';
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

nasa_payload_mass
48213

Total payload mass carried by NASA missions.

To find these missions, we need to query the ‘Customer’ column and find all those that start by ‘NASA (CRS)’

[Github URL: 5. SQL: EDA Notebook]

Average Payload Mass by F9 v1.1

```
%%sql
--# Average payload mass carried by booster version F9 v1.1
SELECT round(avg(PAYLOAD_MASS__KG_), 2) AS avg_f9_payload_mass
    FROM SPACEXTBL
    WHERE Booster_Version LIKE 'F9 v1.1%';
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

avg_f9_payload_mass
2534.67

Average payload carried by Falcon 9 boosters, version 1.1

The result is averaged and rounded to 2 decimal places.

[Github URL: 5. SQL: EDA Notebook]

First Successful Ground Landing Date

```
%%sql
--# first successful landing on a ground pad
SELECT min(Date) AS first_successful_landing
    FROM SPACEXTBL
    WHERE Landing_Outcome = 'Success (ground pad)';
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

first_successful_landing
2015-12-22

We need to find the minimum value of a date, ‘min(Date)’ when the landing outcome is equal to ‘Success (ground pad)’.

[Github URL: 5. SQL: EDA Notebook]

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql
--# boosters successfully landed in dron ship and payload mass between 4000 and 6000
SELECT DISTINCT Booster_Version FROM SPACEXTBL
    WHERE 4000 < PAYLOAD_MASS__KG_ < 6000
    AND Landing_Outcome = 'Success (drone ship)';
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

Booster_Version
F9 FT B1021.1
F9 FT B1022
F9 FT B1023.1
F9 FT B1026
F9 FT B1029.1
F9 FT B1021.2
F9 FT B1029.2
F9 FT B1036.1
F9 FT B1038.1
F9 B4 B1041.1
F9 FT B1031.2
F9 B4 B1042.1
F9 B4 B1045.1
F9 B5 B1046.1

In this query, we need to combine 2 conditions with a ‘AND’ clause to find all payloads greater 2000 and less than 4000. in this way, we define a range.

The discriminant is finding all booster versions which landed successfully on a drone ship.

[Github URL: 5. SQL: EDA Notebook]

Total Number of Successful and Failure Mission Outcomes

```
%%sql
SELECT DISTINCT Mission_Outcome AS outcome, count(Mission_Outcome) AS count
    FROM SPACEXTBL
    GROUP BY Mission_Outcome;
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

outcome	count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

We GROUP results BY the mission outcome and count the results of each one of them.

In my example, “Success” seems to be repeated, being the total number of successful outcomes 99. This seems to be an issue with the encoding since it doesn’t recognize both strings as the same.

[Github URL: 5. SQL: EDA Notebook]

Boosters Carried Maximum Payload

```
%%sql
-- # boosters that have carried the maximum payload mass
SELECT Booster_Version FROM SPACEXTBL
    WHERE PAYLOAD_MASS__KG_ = (SELECT max(PAYLOAD_MASS__KG_)
    FROM SPACEXTBL);
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

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

Find the name of the booster which have carried the maximum payload possible.

To find them, we need to know what is the maximum payload using a subquery.

Then, select the boosters which payload_mass column match that subquery.

[Github URL: 5. SQL: EDA Notebook]

2015 Launch Records

```
%%sql
SELECT substr(Date, 0, 5) AS year,
Booster_Version AS booster,
Launch_Site AS launch_site,
Landing_Outcome AS outcome
    FROM SPACEXTBL
    WHERE year = '2015' AND outcome = 'Failure (drone ship)';
```

✓ 0.0s

```
 * sqlite:///data/05_spacex_data.sqlite3
Done.
```

year	booster	launch_site	outcome
2015	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
2015	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

Year can be extracted from the Date column by using the sqlite substring method.

Then, there only to combine the 2 conditions:

- year = 2015

- outcome = ‘Failure (drone ship)’

[Github URL: 5. SQL: EDA Notebook]

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

```
%%sql
--# count of landing outcomes between 2010-06-04 and 2017-03-20 in descending order
SELECT Landing_Outcome AS outcome, COUNT(Landing_Outcome) AS outcome_count
    FROM SPACEXTBL
    WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
    GROUP BY outcome
    ORDER BY outcome_count DESC;
```

✓ 0.0s

* sqlite:///data/05_spacex_data.sqlite3
Done.

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

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

We need to use GROUP BY

[Github URL: 5. SQL: EDA Notebook]

Section 3

Launch Sites Proximities Analysis

SpaceX Falcon 9 Launch Sites map

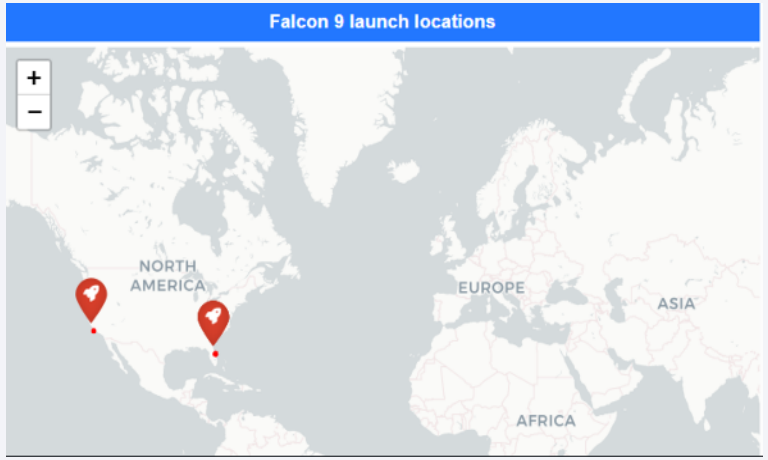


Fig 1. World map

Maps that show launch locations for the Falcon 9 missions.

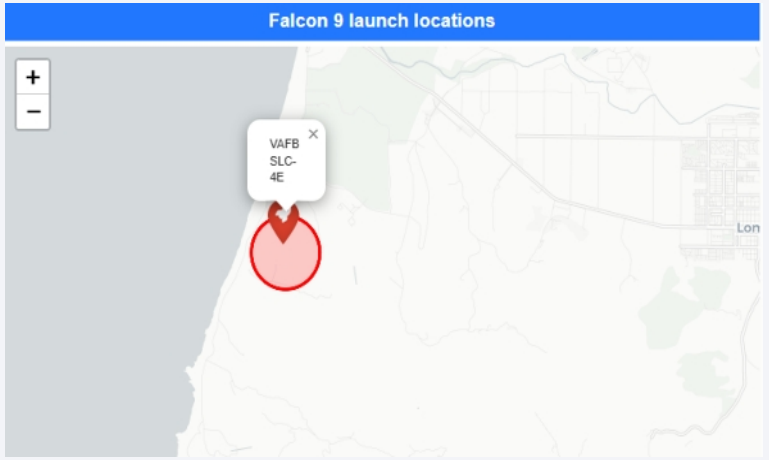


Fig 2. VAFB SLC-4E (Los Angeles)

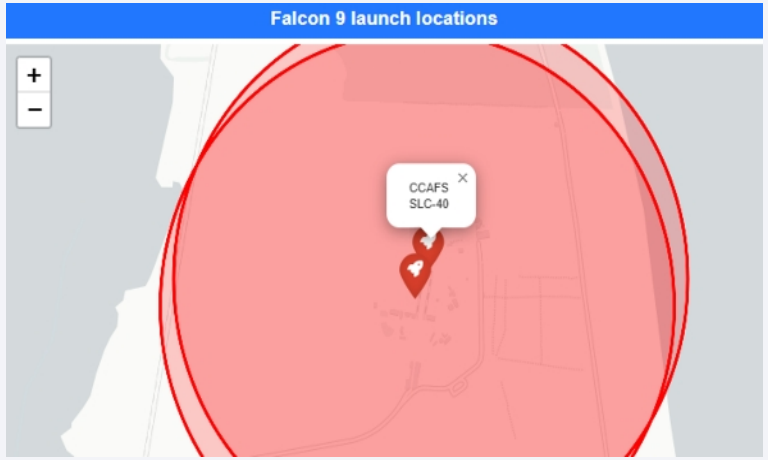


Fig 3.CCAFS SLC-40 & CCAFS LC-40 (Florida)

[Github URL: 7. EDA Folium Notebook]

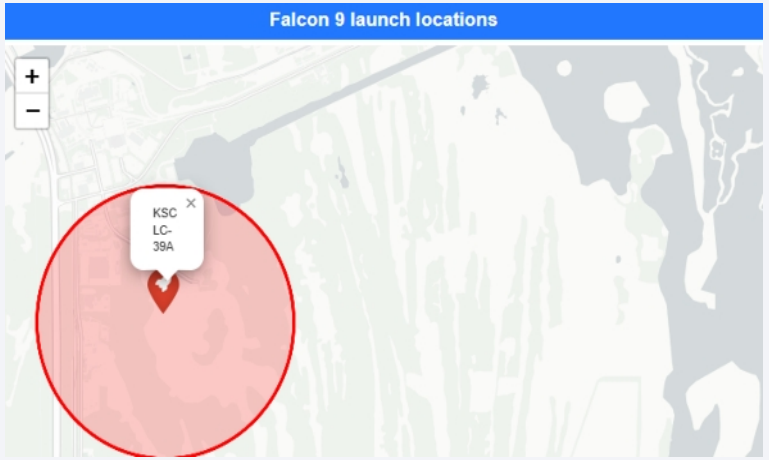


Fig 4. KSC LC-39A (Florida)

SpaceX Falcon 9 launch success/failure rate

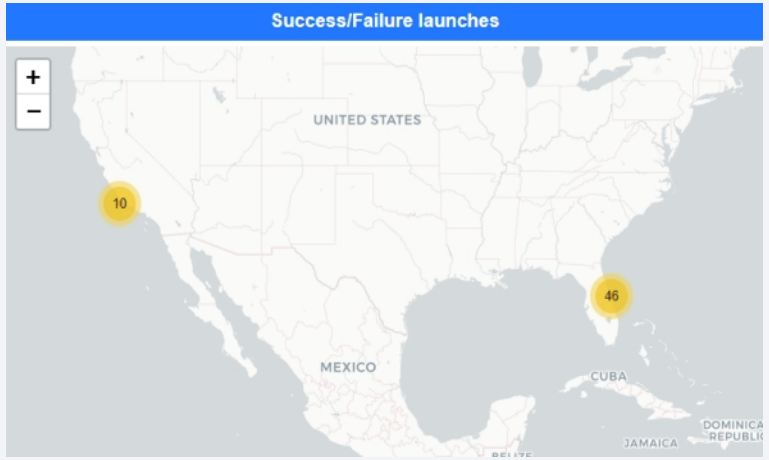


Fig 1. World map

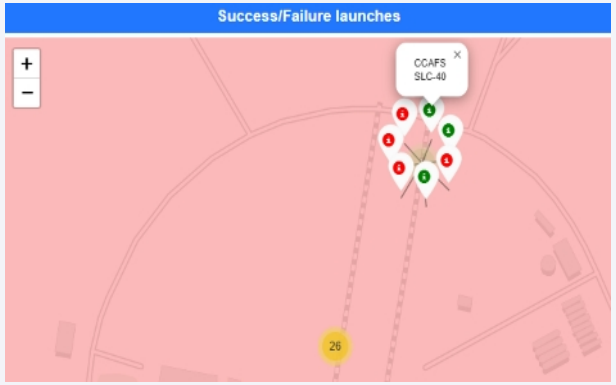


Fig 2. CCAFS SLC-40 Florida)

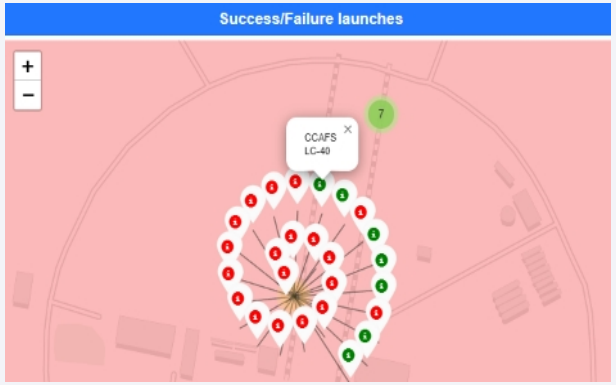


Fig 1. CCAFS LC-40 (Florida)

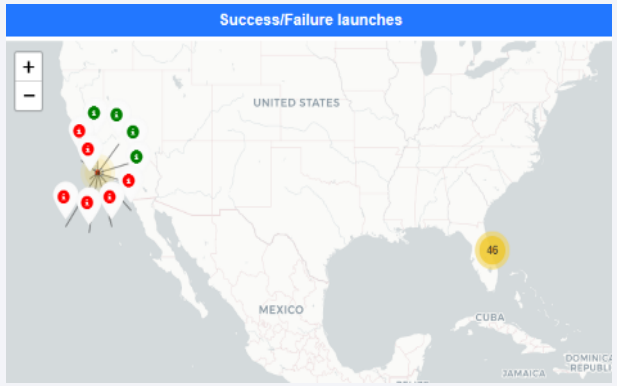


Fig 2. CAFB (Los Angeles)

Fig 2. KSC (Florida)

Show number of launches from each launch site.

Zooming-in, you see the detail of success and failures on each site (as seen on the smaller screenshots).

[Github URL: 7. EDA Folium Notebook]

SpaceX Falcon 9 Proximity to closest cities

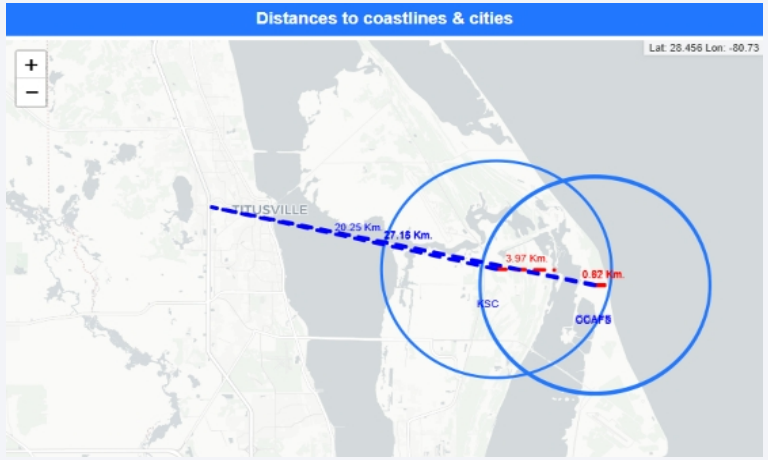


Fig 1. Launch sites in Florida

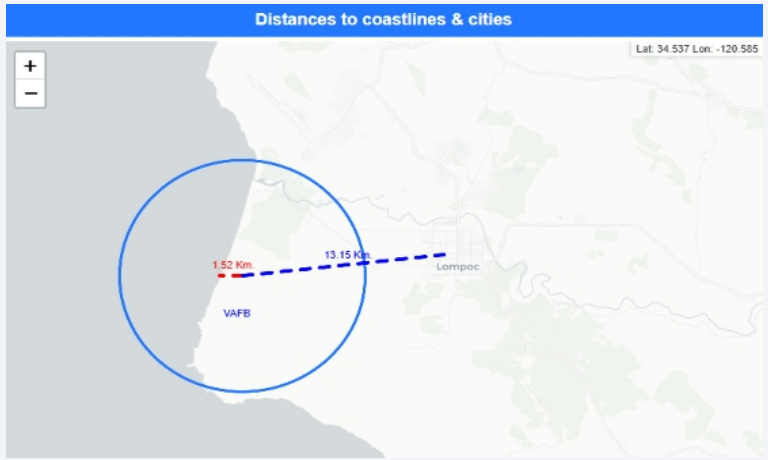


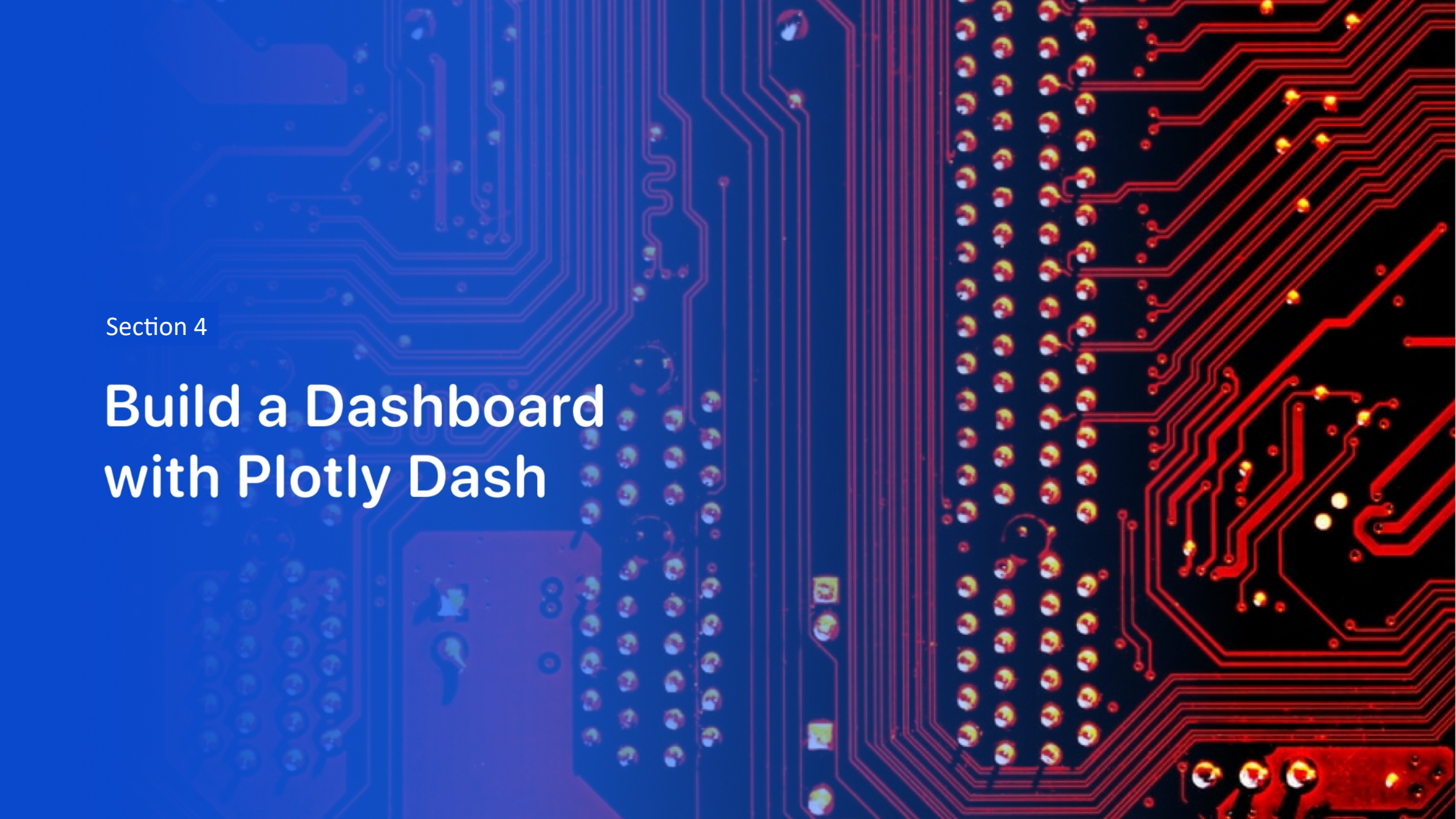
Fig 2. Launch site in Los Angeles

Pictures show distances between coordinates from the launch site and the closest cities (in blue) and the coastline (red).

Insights:

Launches take place relatively afar from cities but close to the coast railroads and some roads (not plotted here).

[Github URL: 7. EDA Folium Notebook]

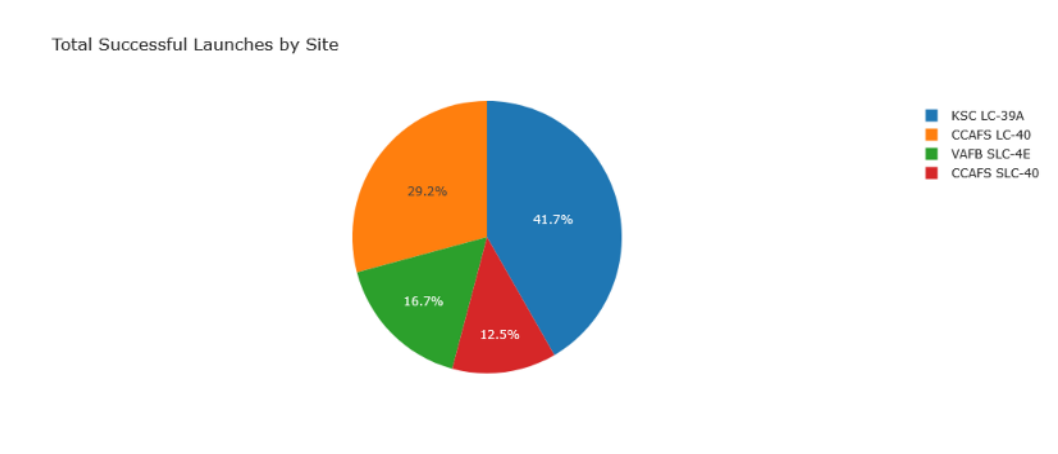

Section 4
Build a Dashboard with Plotly Dash

Success count for all sites

Insights:

KSC LC-39A, in Florida, show the highest launch success ratio.

On the other hand, CCAFS SLC-40, also in Florida, has the lowest success ratio.



site	# launches	# success	# failure	# success_ratio	# failure_ratio
CCAFS LC-40	26	7	19	0.2692	0.7308
VAFB SLC-4E	10	4	6	0.4	0.6
KSC LC-39A	13	10	3	0.7692	0.2308
CCAFS SLC-40	7	3	4	0.4286	0.5714

[Github URL: 7. Dash Analysis Notebook]

Highest launch success ratio

Insights:

KSC LC-39A, in Florida, show the highest launch success ratio.

As we can see in the previous table:

- Its launch success rate is around a 76%.
- Its failure launch rate is around 23%.

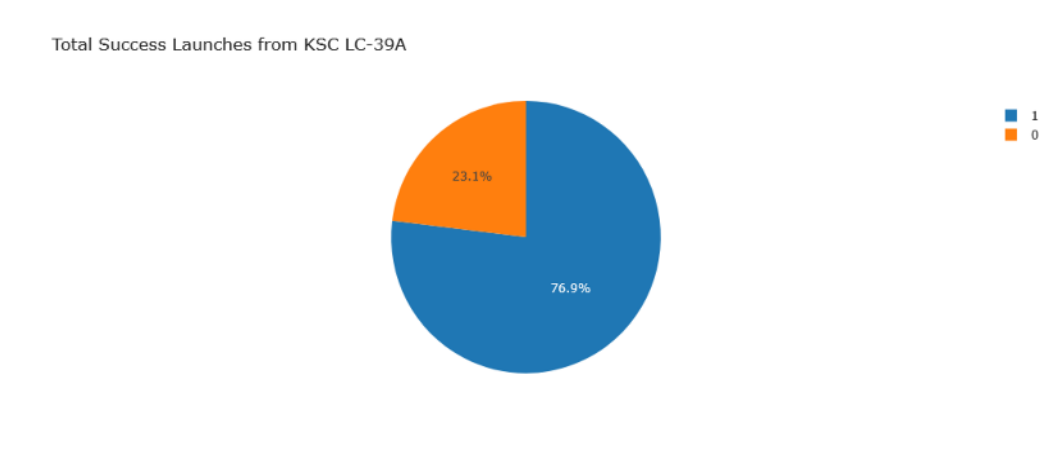


Fig 1. KSC launch success/failure ratio

[Github URL: 7. Dash Analysis Notebook]

Payload vs launch Outcome

Fig 1 show the total range of payloads for ALL launch sites.

Fig 2 contain the information for a payload range from 2000 to 4000 kg

Fig 3 contain payloads from 2k to 6k kg

Finally, fig 4, contain the information in tabular format for contrast.

[Github URL: 7. Dash Analysis Notebook]

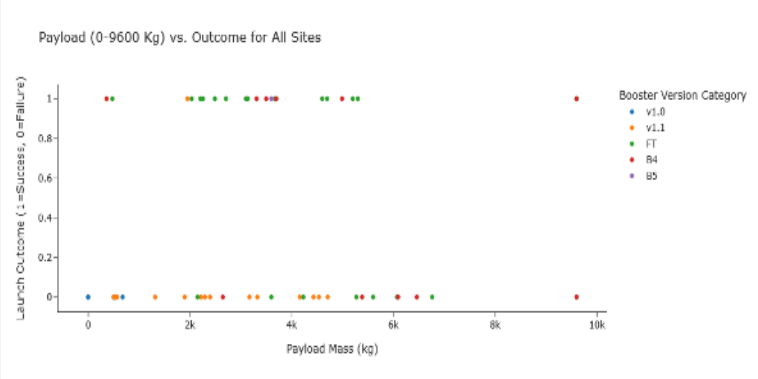


Fig 1. Launch outcome by payload range

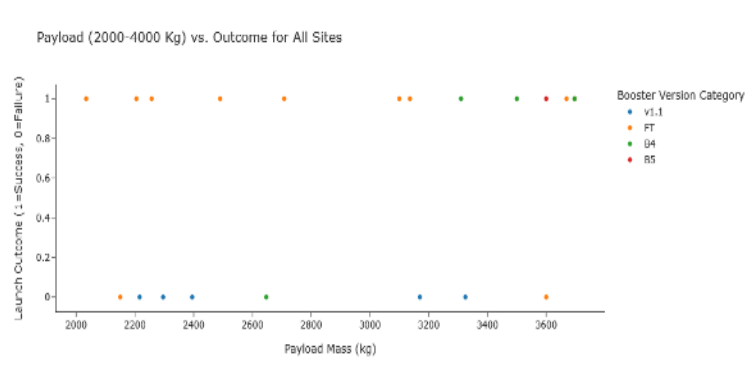


Fig 2. Launch outcome by payload from 2k-4k

# min	# max	# launches	# success	# failure	# success_ratio	# failure_ratio ···
0	2000	13	3	10	0.2308	0.7692
2000	4000	21	13	8	0.619	0.381
4000	6000	13	5	8	0.3846	0.6154
6000	8000	4	0	4	0.0	1.0
8000	10000	5	3	2	0.6	0.4

Fig 4. Success ratio by payload range

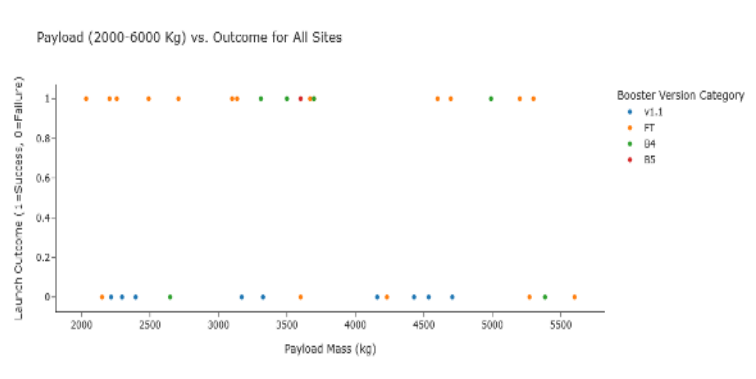


Fig 3. Launch outcome by payload from 2k-6k

Most successful booster

Insights:

Although B5 booster show a 100% success rate, it accounts only one launch so we will not count this.

The second best is FT with nearly a 66,67% success rate.

booster	launches	success	failure	success_ratio	failure_ratio
B5	1	1	0	1.0	0.0
FT	24	16	8	0.6667	0.3333
B4	11	6	5	0.5455	0.4545
v1.1	15	1	14	0.0667	0.9333
v1.0	5	0	5	0.0	1.0

[Github URL: 7. Dash Analysis Notebook]

Section 5

Predictive Analysis (Classification)

Classification Accuracy

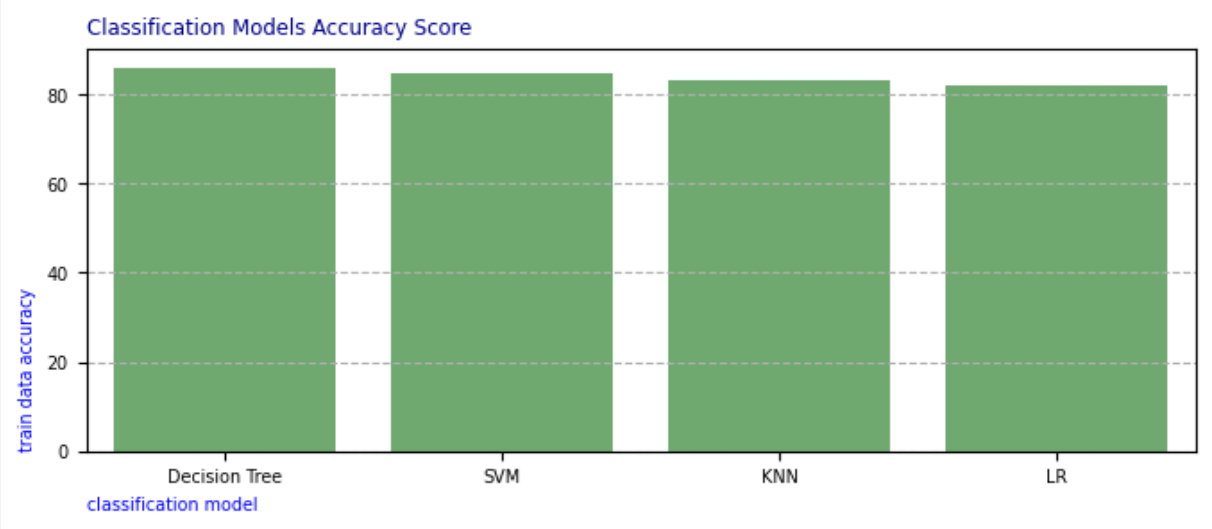
Fig 1 show tabular data of how each classification models perform on test data (on the left) and on training data (on the right)

Fig 2 show the accuracy ratio on the training data.

algorithm	test_data_score_pct_score	train_data_accuracy_score
Decision Tree	77.7778	86.0714
SVM	83.3333	84.8214
KNN	77.7778	83.3929
LR	83.3333	82.1429

Given this information, we can conclude that Decision Trees algorithm performs best on training data, but not so much over the test data.

Likewise, Logistic Regression, and Support Vector Machine do better with the test data set.



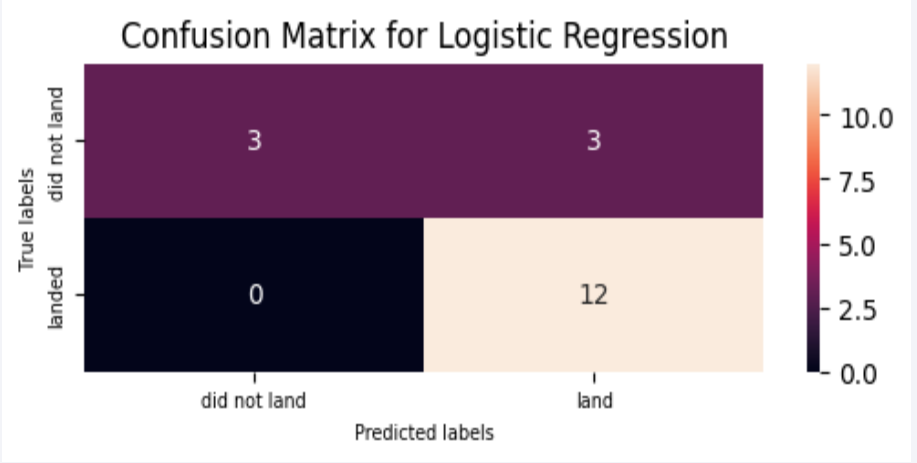
[Github URL: 8. ML Predictions Notebook]

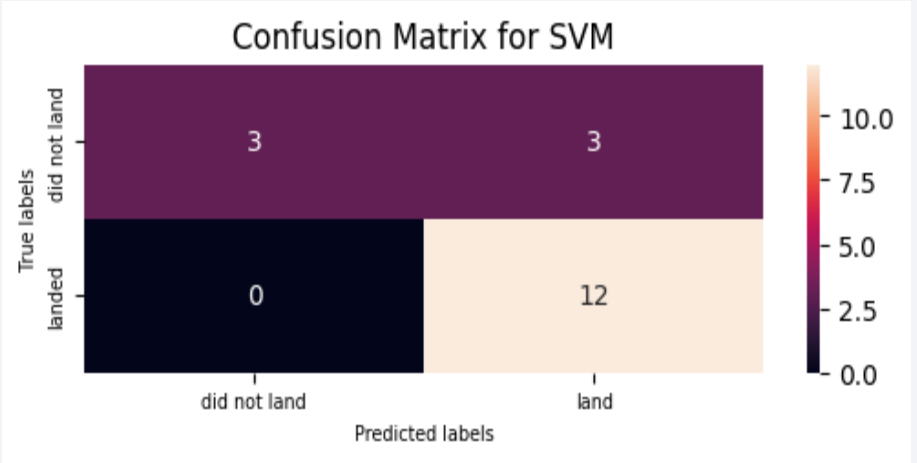
Confusion Matrix

Fig 1 show the confusion matrix for the Logistic Regression classification model, which is the best performing algorithm on training data.

Fig 2, however, show the confusion matrix for the SVM model, which perform best on test data.

[Github URL: 8. ML Predictions Notebook]





Conclusions

Number of successful launches increases has increased over the last years, significantly from 2015. The success trend seems to stabilize at a 60 - 90% ratio from 2016 onwards.

The number of launches has also increased, so the successes have gone up. VAFB show the best ratio of succeeded launches. KSC doesn’t show information below almost 40 launches.

Orbits such as ES-L1, GEO, HEO and SSO show a 100% success launch rate. On the contrary, GTO reaches only a ~50% success rate.

Launch sites KSC LC-39A has a higher success launch rate (~77%) while CCAFS LC40 has the worst ratio with a ~73% of failures.

Launch sites are located relatively far from cities, while they do locate close to the shore and facilities as railroads.

Payloads between 2000 and 5000 kg have a higher success rate (~62%). On the other hand, the ranges 0 - 2000 kg (77%), 5000 - 7000 kg (78%) show the highest failure rates.

Falcon 9 FT boosters have a ~67% success rate.

In our train/test split data examples, Logistic Regression perform best on training data (~86%), while SVM and LR (83% on both cases) do on test data.

Appendix

Thank you!