Winning Space Race with Data Science

Karol Kosyra
15 January 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - Exploratory Data Analysis with Data Visualization and SQL
  - Interactive Visual Analysis with Folium and Plotly Dash
  - Predictive analysis using Classification Models

- Summary of all results
  - Exploratory Data Analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

Introduction

- Project background and context

  SpaceX advertises Falcon 9 rocket launches on its website 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. Therefore if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch

- Problems you want to find answers
  - What affects the success of landing
  - Predict if the Falcon 9 first stage will land successfully

Section 1

Methodology

Methodology

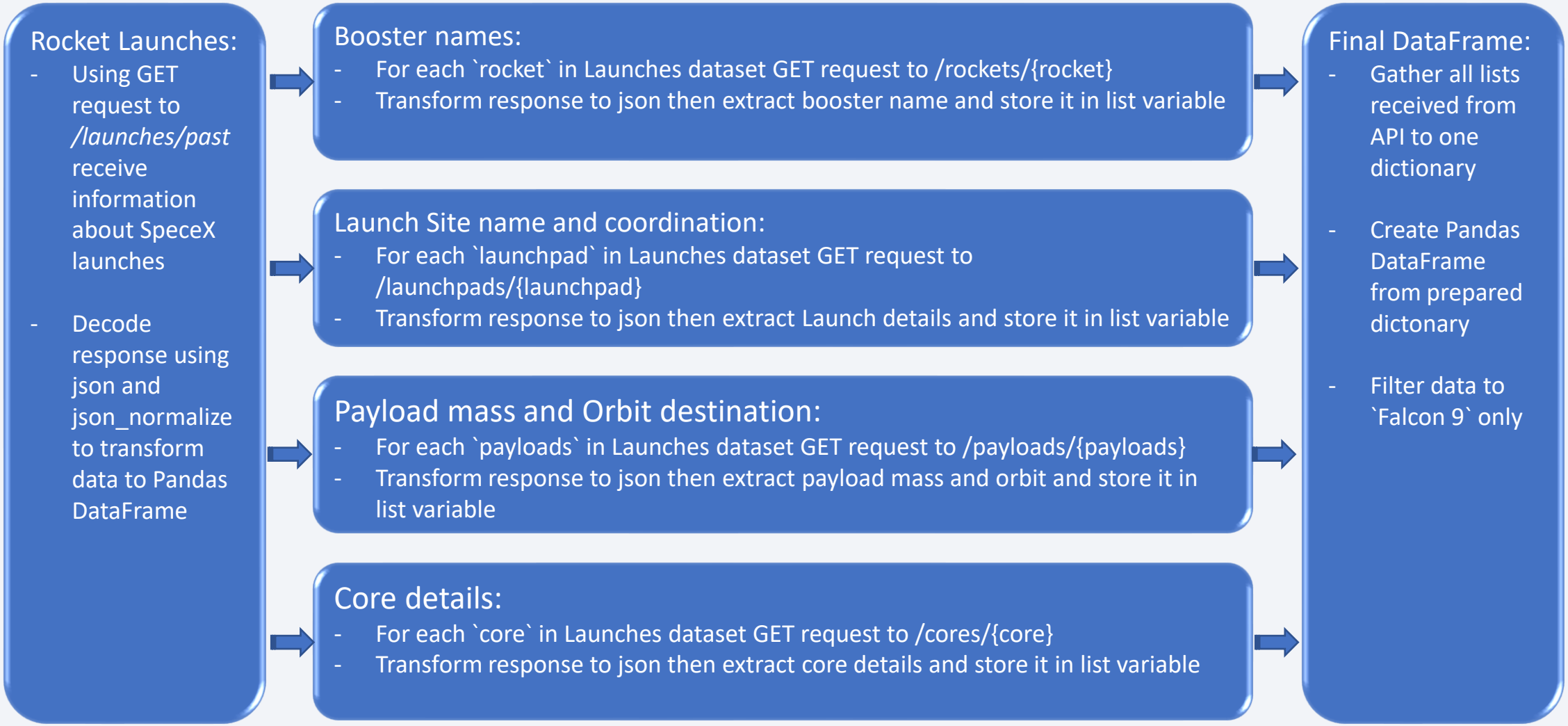
Executive Summary

- Data collection methodology:
  - SpaceX API and WebScrapping from Wikipedia
- Perform data wrangling
  - Dealing with Missing Values
  - Creating a landing outcome label
  - Applying OneHotEncoding and StandardScaler
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Splitting data to train and test datasets
  - Evaluating accuracy by finding best parameters using Grid Search and Cross Validation

Data Collection

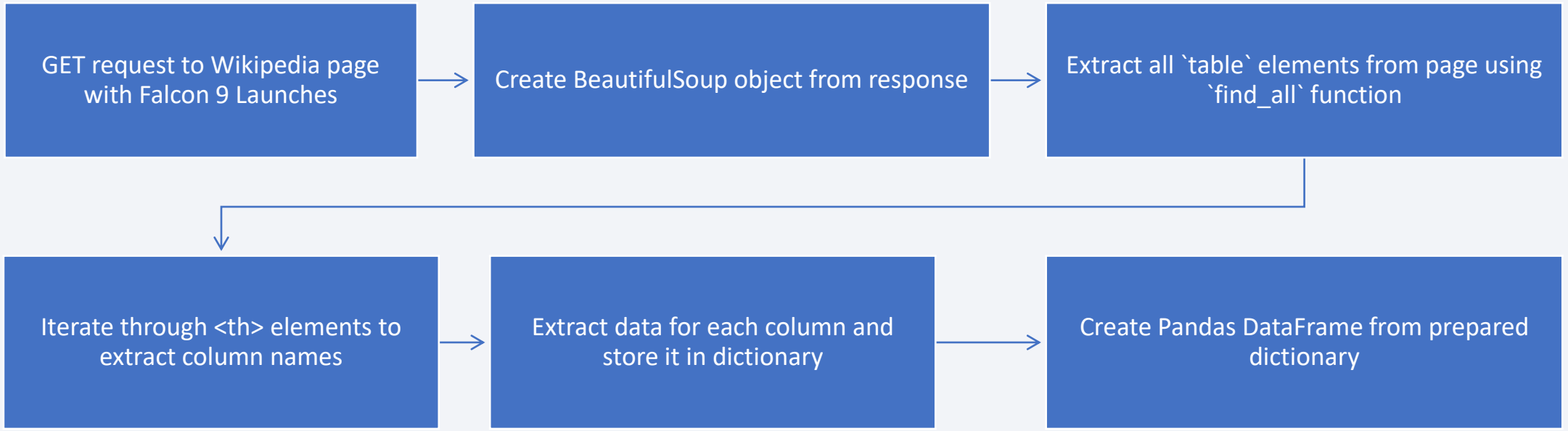
- Data was collected from two sources:
  - SpaceX API using GET request
    - ✓ Rocket launches from: https://api.spacexdata.com/v4/launches/past
    - ✓ Booster names from: https://api.spacexdata.com/v4/rockets
    - ✓ Launch Site name and coordinates from: https://api.spacexdata.com/v4/launchpads
    - ✓ Payload mass and Orbit destination from: https://api.spacexdata.com/v4/payloads
    - ✓ Details about core and flights from: https://api.spacexdata.com/v4/cores
  - Wikipedia page using WebScrapping with BeautifulSoup library
    - ✓ Web page: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
    - ✓ This web page contains data valid for date 9 June 2021

Data Collection – SpaceX API



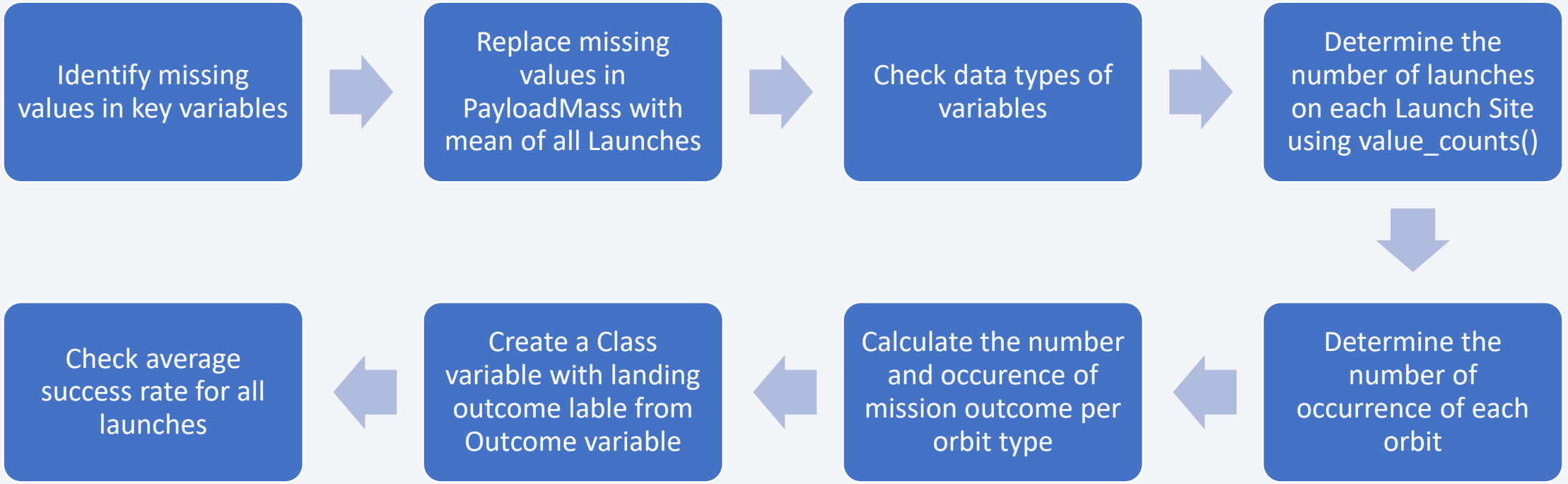
Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping



Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/jupyter-labs-webscraping.ipynb

Data Wrangling



Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/jupyter-labs-webscraping.ipynb

EDA with Data Visualization

- Scatter point charts using seaborn to see:
  - ✓How Flight Number and Payload mass affect launch outcome
  - ✓What is the distribution of Launch Site over Flight Number and with what outcome
  - ✓If there is any relationship between Payload Mass and Launch Site
  - ✓What is the distribution of Orbit type over Flight Number
  - ✓Relationship between payload and orbit type
- Bar chart to visualize success rate of each orbit type

✓Line chart to show success yearly trend

Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- SELECT:
  - ❑ unique launch sites in the space mission
  - ❑ 5 records where launch sites begin with the string 'CCA’
  - ❑ total payload mass carried by boosters launched by NASA (CRS)
  - ❑ average payload mass carried by booster version F9 v1.1
  - ❑ date when the first successful landing outcome in ground pad was achieved
  - ❑ names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - ❑ total number of successful and failure mission outcomes
  - ❑ names of the booster versions which have carried the maximum payload mass
  - ❑ failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
  - ❑ Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/jupyter-labs-eda-sql-coursera.ipynb

Build an Interactive Map with Folium

- Marker and Circle objects added to mark and highlight area of:
  - all launch site on the map
  - success and failed launches for each site on the map
- Line objects to estimate distance from closest:
  - Coastline
  - City
  - Railway
  - Highway

Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/lab_jupyter_launch_site_location.ipynb

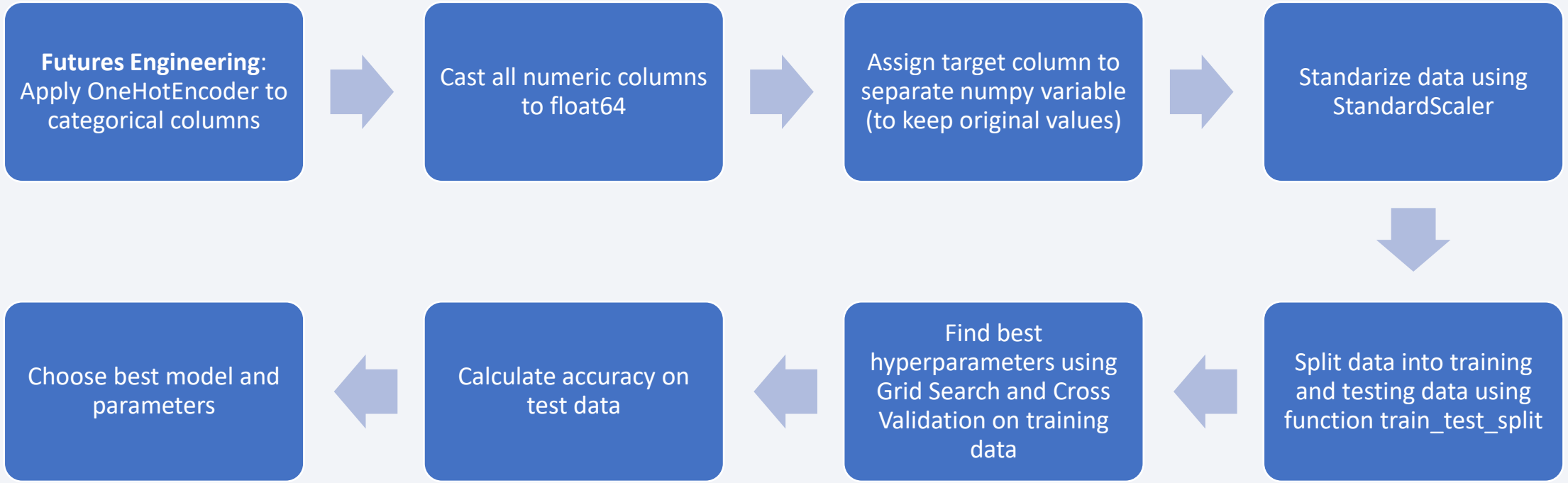
Build a Dashboard with Plotly Dash

- Dashboard created with Plotly Dash contains:
  - Pie chart with dropdown menu to select Launch site. If specific Launch Site is not selected then % of Total launches per site is showed. When Launch Site is selected then Successful launch rate is showed on graph.
  - Scatter point chart with Range Slider to filter Payload Mass. Graph shows relationship between Payload Mass and Launch success rate.

Link to notebook on GitHub: https://github.com/Karlos754/IBM-Data-Science/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

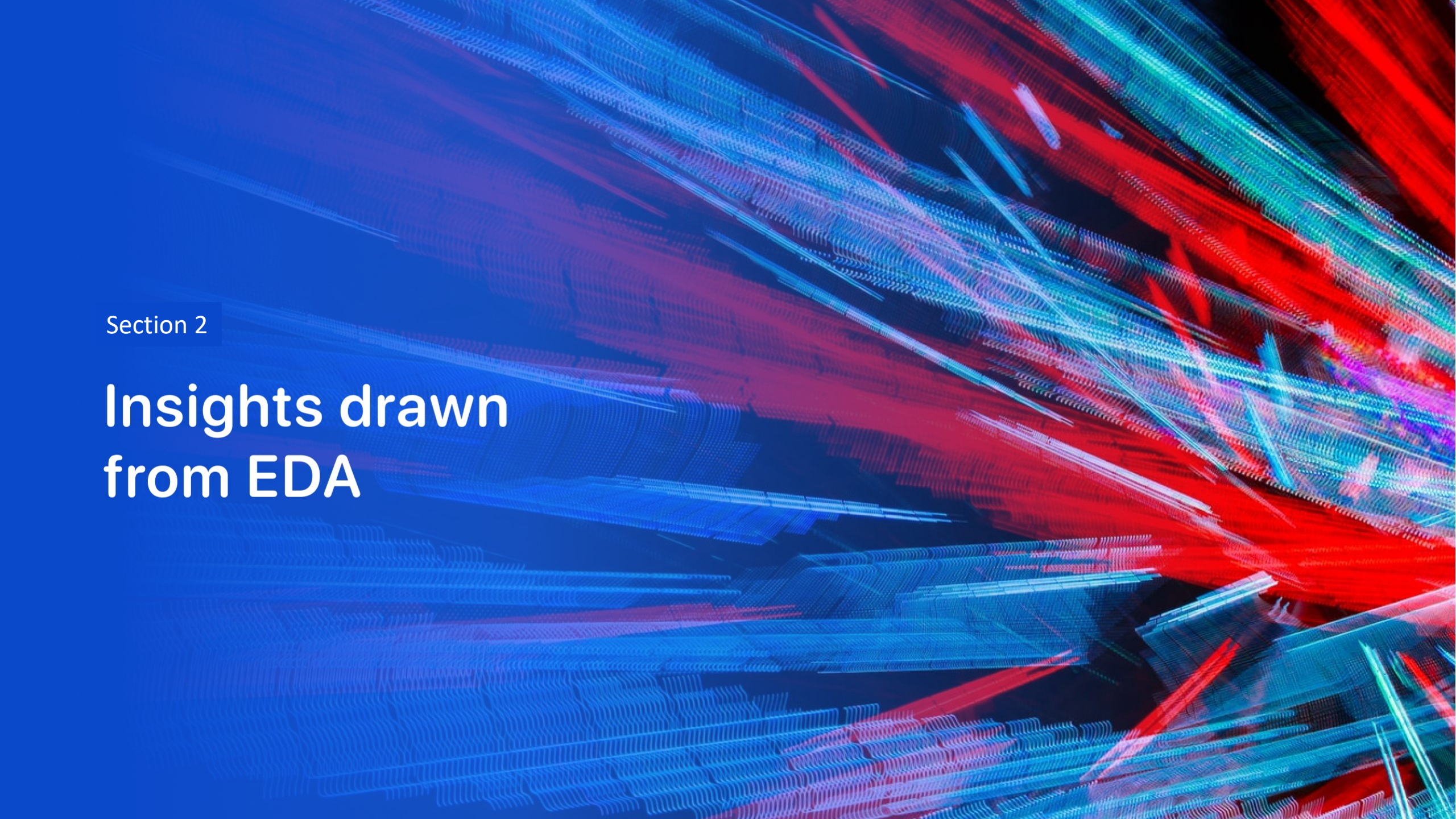
5 classification algorithms were used to predict landing result of Falcon 9:
Logistic Regression, Supported Vector Machine, Decision Tree, K-Nearest Neighbors.
Accuracy has been chosen as evaluation metric.
Following steps was applied to prepare classification model:



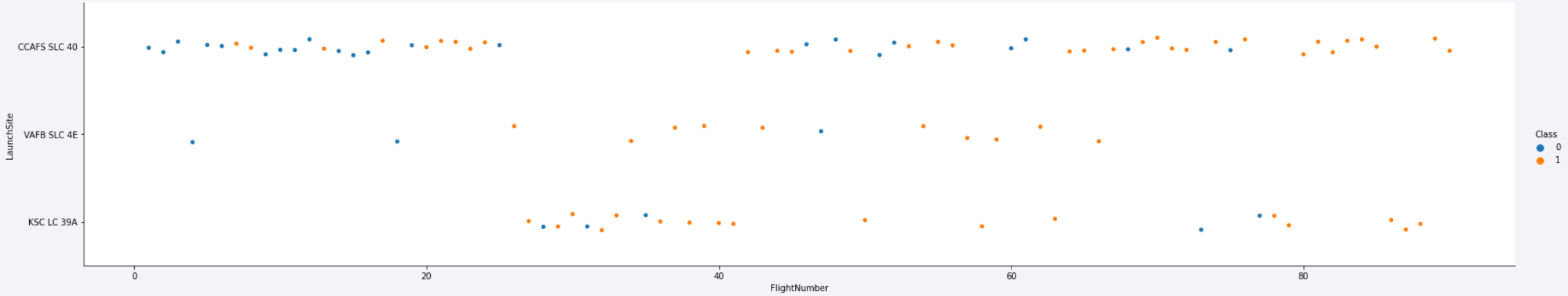
Link to notebook on GitHub: https://github.com/Karlos754/.../SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

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

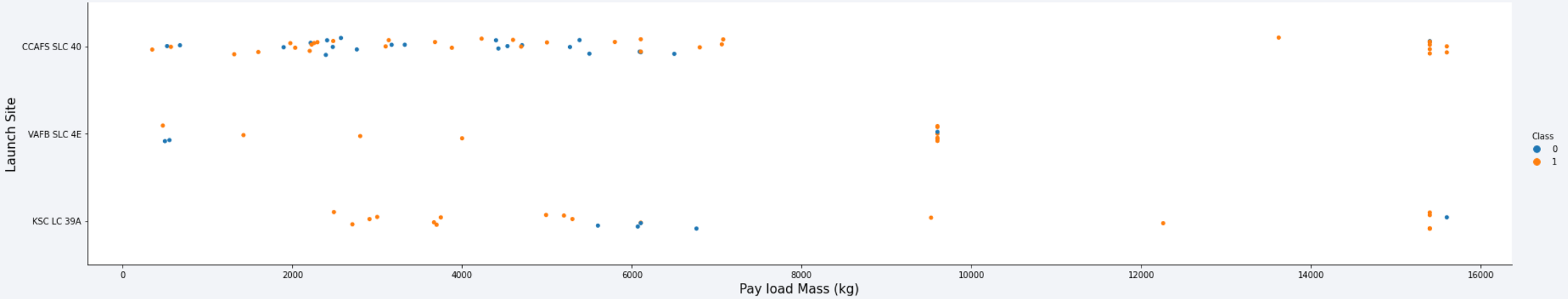

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



CCAFS SLC-40 is most likely Launch Site to use. However after about 23 flights they change it mainly to KSC LC-39A for like 20 flights and then they resume flights from CCAFS SLC-40 with higher landing success rate

Payload vs. Launch Site

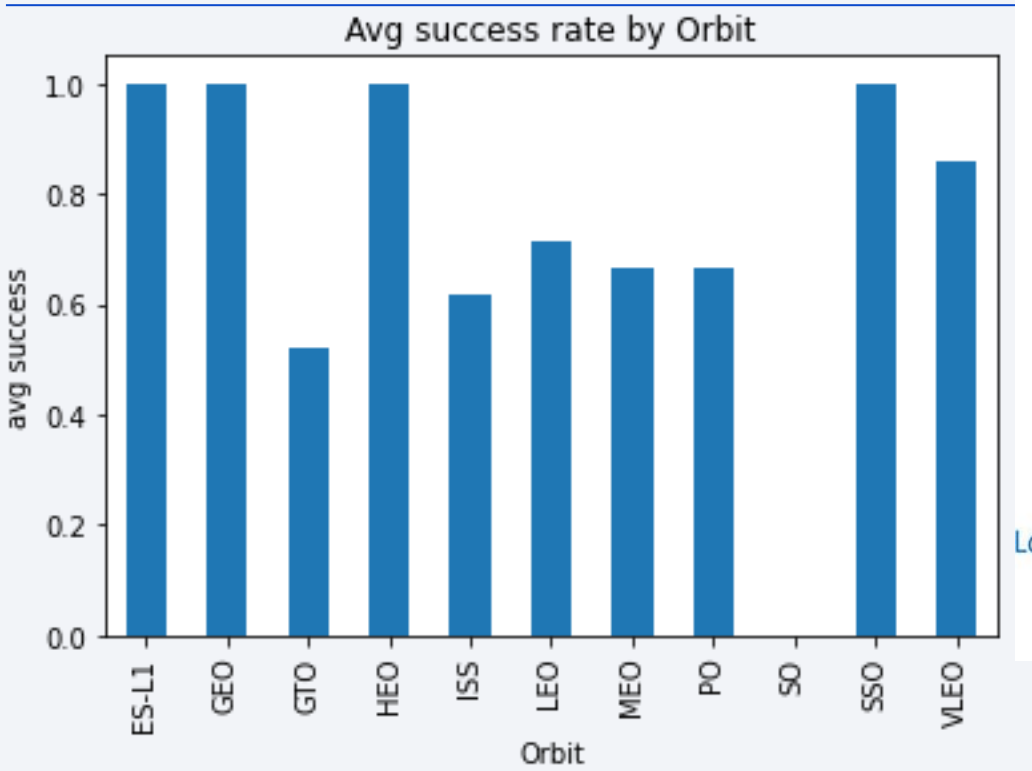


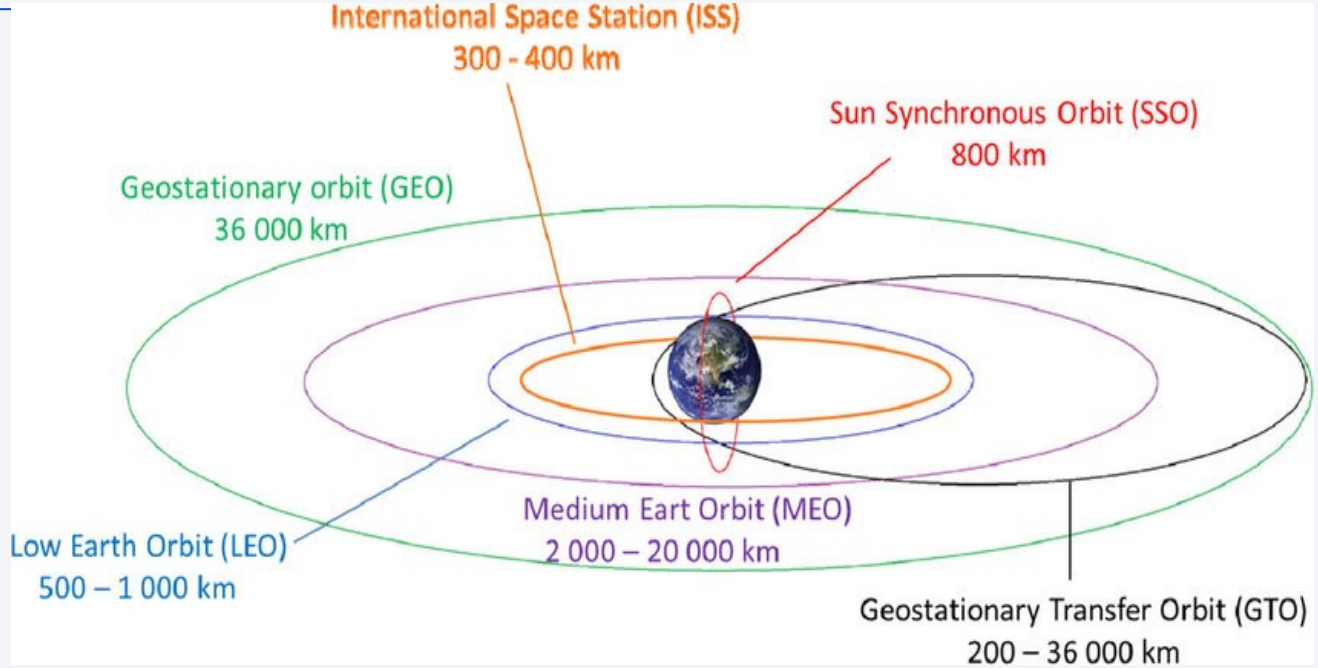
CCAFS SLC-40 Launch Site is very successful for heavy payload mass (> 13k kg)

VAFB SLC-4E has no launches for heavy payloads

KSC LC-39A is successful for light payloads (< 5 800 kg)

Success Rate vs. Orbit Type

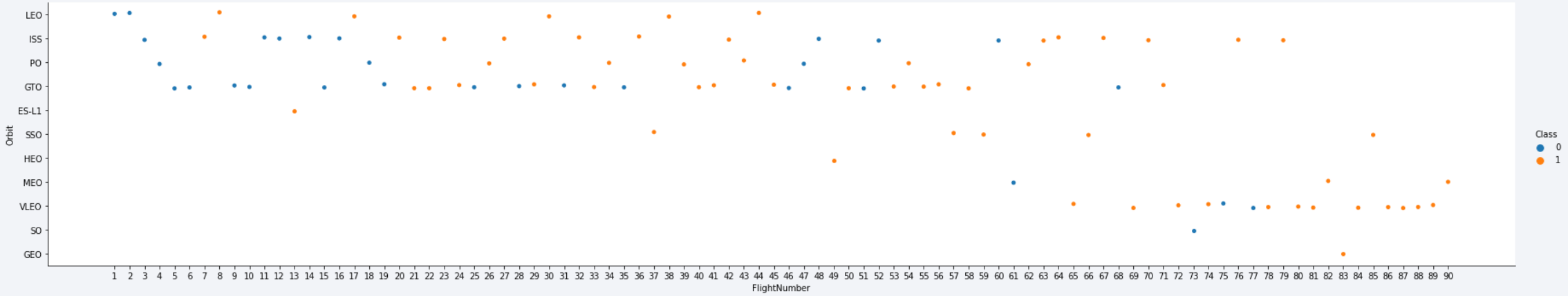




Launches success rate for GTO and ISS orbits is below average however this destination was mostly used.

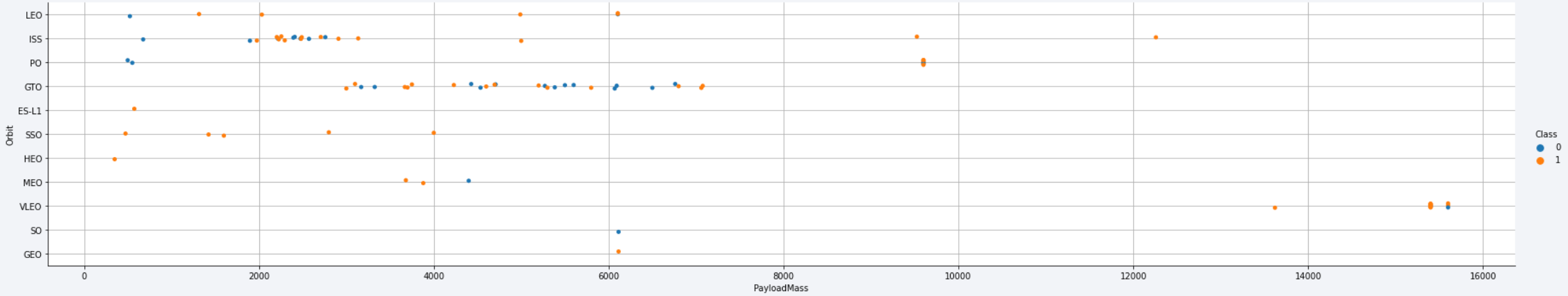
Some Orbits have high success rate but there was just a few launches on this orbits.

Flight Number vs. Orbit Type



- LEO Orbit has successful flights just after 3 launches on this Orbit
- VLEO Orbit is mostly chosen after 70th flight

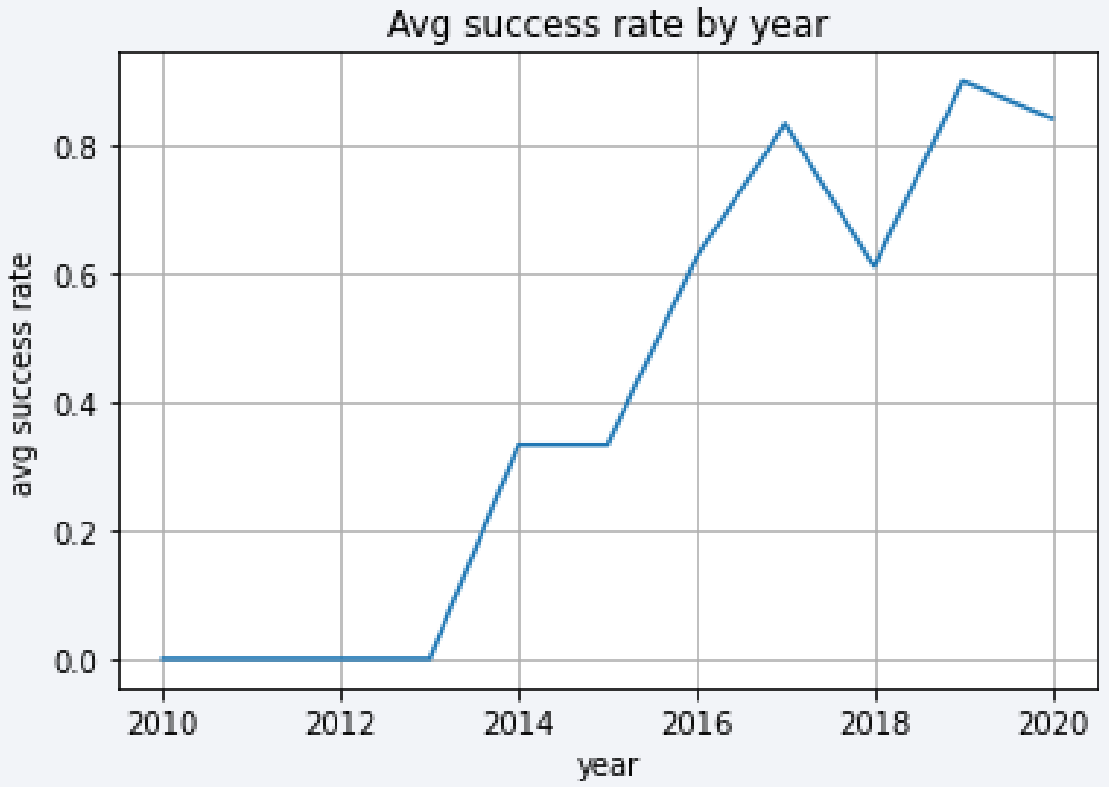
Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

- Successful flights started in 2013
- Between 2015 and 2017 was significant increase of successful flights
- Little decrease was in 2018 but since then its increasing again



All Launch Site Names

Display the names of the unique launch sites in the space mission

+ Code + Markdown

```
%%sql
select distinct
    launch_site
from spacex
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

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

`distinct` statement was used to select unique launch_site names

Launch Site Names Begin with 'CCA'

Display 5 records where launch sites begin with the string 'CCA'

```
%%sql
select
    *
from spacex
where launch_site like 'CCA%'
limit 5
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

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

To select records `where` clause was used with `like` statement to select only this records where launch_site name begins with CCA. `%` in like statement string is a wildcard and it means that there can be some text after this phrase.
`limit 5` statement was used to select only 5 records

Total Payload Mass

Display the total payload mass carried by boosters launched by

```
%%sql
select
    sum(payload_mass__kg_) as total_payload_mass
from spacex
where customer = 'NASA (CRS)'
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

total_payload_mass
45596

To display total payload mass for ‘NASA (CRS)’ customer can use sum() function and where clause

Average Payload Mass by F9 v1.1

Display average payload mass carried by booster version F9 v1.1

```
%%sql
select
    avg(payload_mass__kg_) as avg_payload_mass
from spacex
where booster_version = 'F9 v1.1'
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

avg_payload_mass
2928

To select average payload mass we can use avg function with `where` clause to filter only ‘F9 v1.1’ booster

First Successful Ground Landing Date

List the date when the first successful landing outcome in ground

Hint:Use min function

```
%%sql
select
    min(date) as first_success_landing
from spacex
where "Landing _Outcome" = 'Success (ground pad)'
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

first_success_landing
2015-12-22

To display first date for successful ground pad landing we can use min() function and `where` clause to filter data

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship a
4000 but less than 6000

```
%%sql
select
    distinct booster_version
from spacex
where "Landing _Outcome" = 'Success (drone ship)'
  and payload_mass__kg_ > 4000 and payload_mass__kg_ < 6000
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

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

There are 4 boosters which successfully landed on drone ship with payload mass between 4000 and 6000 kg

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%%sql
select
    mission_outcome
   ,count(*) as total_missions
from spacex
group by mission_outcome
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

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

To calculate total number of mission_outcome we can use count() function with `group by` clause to indicate for which column we want to count records

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the m

```
%%sql
select distinct
    booster_version
from spacex
where payload_mass__kg_ = (
    select max(payload_mass__kg_)
    from spacex
)
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

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

To select unique booster_versions which have carried max payload mass we can use `distinct` statement to remove duplicated names in output and subquery in `where` clause to filter only this records with highest payload mass using max function in subquery.

2015 Launch Records

List the failed landing_outcomes in drone ship, their booster versio
2015

+ Code + Markdown

```
%%sql
select
    "Landing _Outcome" as landing_outcome
    ,booster_version
    ,launch_site
from spacex
where extract(year from date) = 2015
  and "Landing _Outcome" = 'Failure (drone ship)'
```

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

landing_outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

To select records only for missions in 2015 we can use extract() function in `where` clasue

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

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

```
%%sql
select
        "Landing _Outcome"
       ,count(*) as landing_count
from spacex
where date between '2010-06-04' and '2017-03-20'
group by "Landing _Outcome"
order by 2 desc
```

✓ 0.1s

```
 * db2+ibm_db://pys11030:***@824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

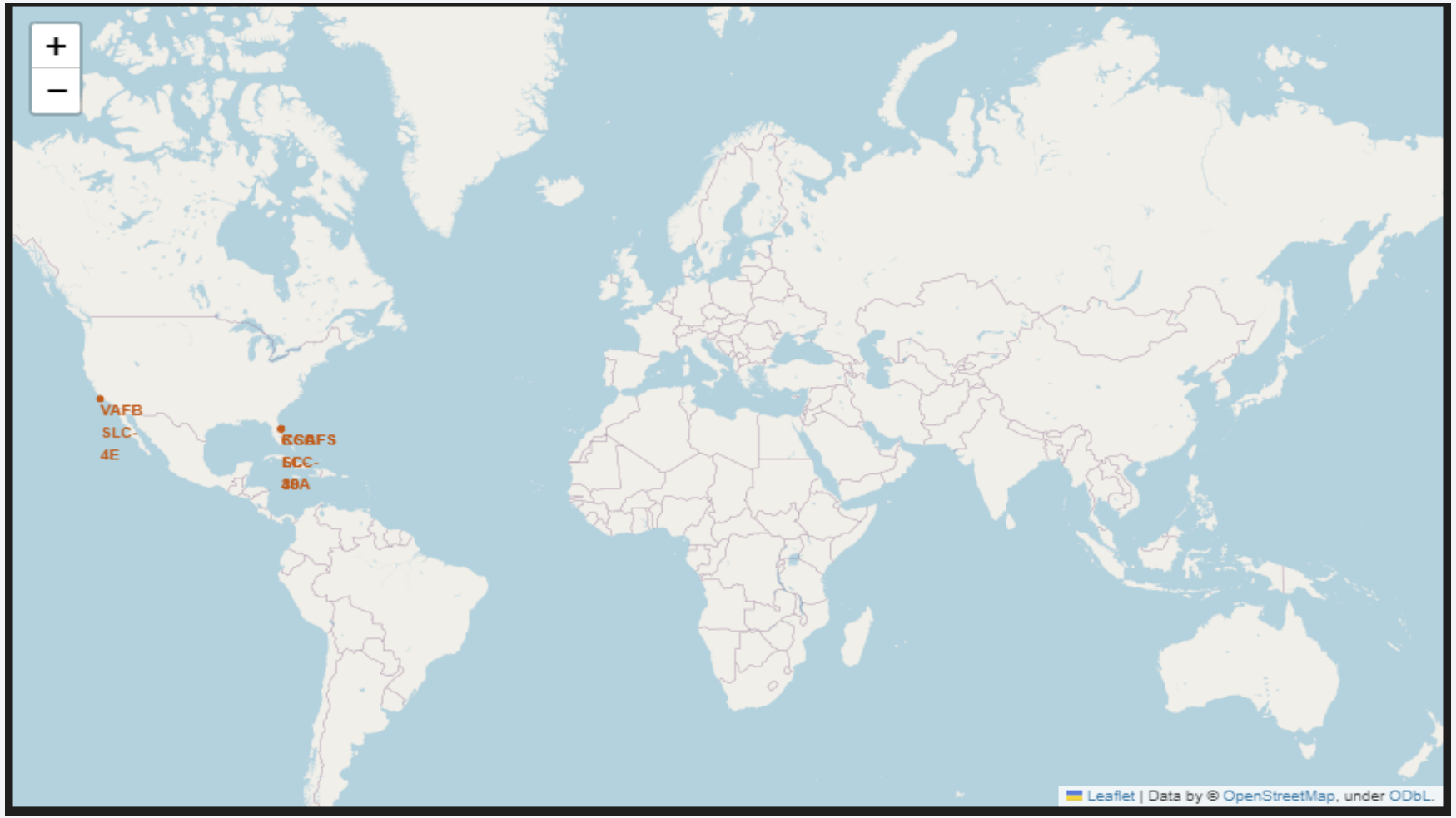
Landing _Outcome	landing_count
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

To rank the count of landing outcomes we have to `group by` result by landing_outcome, use count() function and sort output using `order by` clause (by 2nd output column – landing_count) descending (from highest to smallest)

Section 3

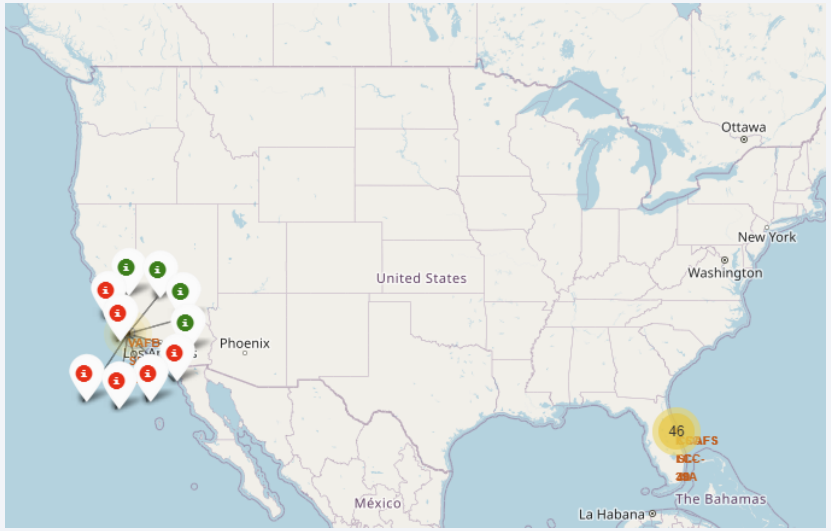
Launch Sites Proximities Analysis

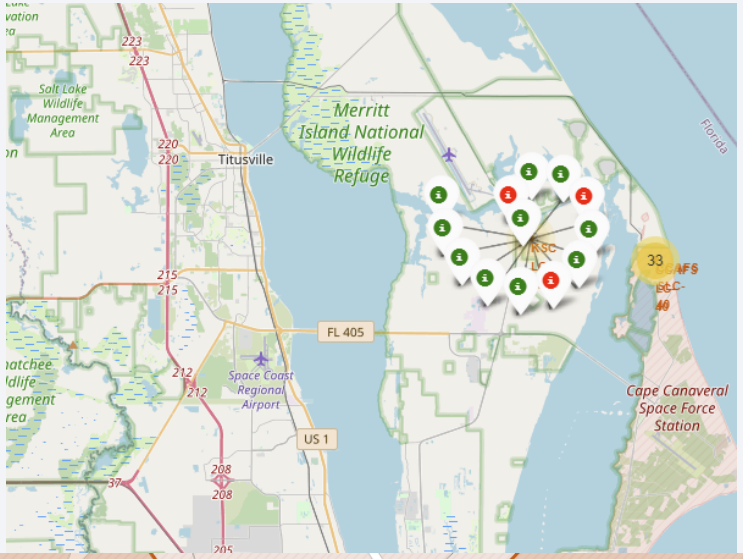
Launch Sites locations



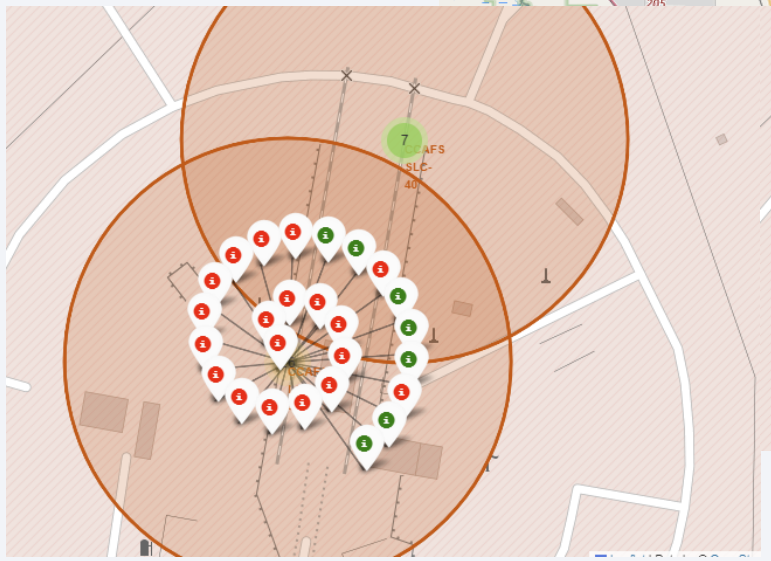
Launch Sites are located in USA (California and Florida)

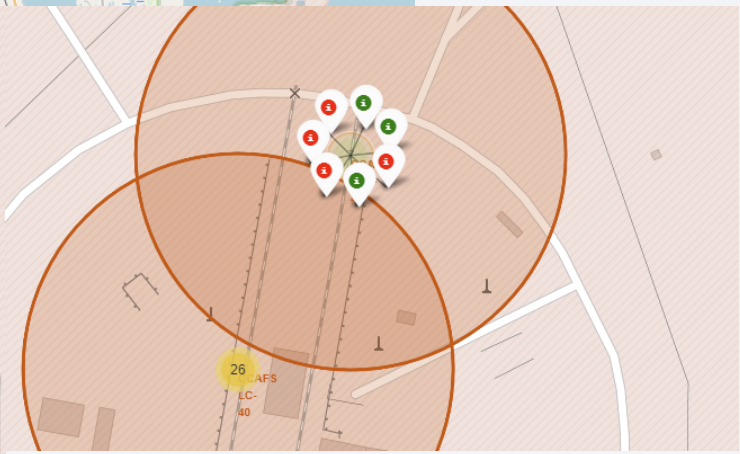
Success and failed launches





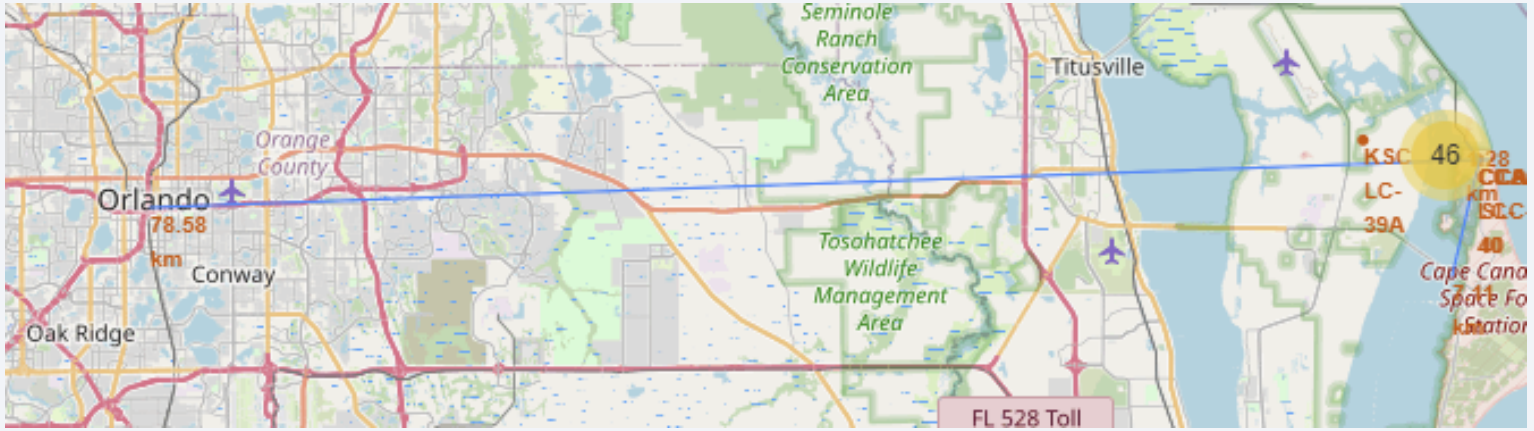
Green markers represent successful launches
Red markers represent failed launches

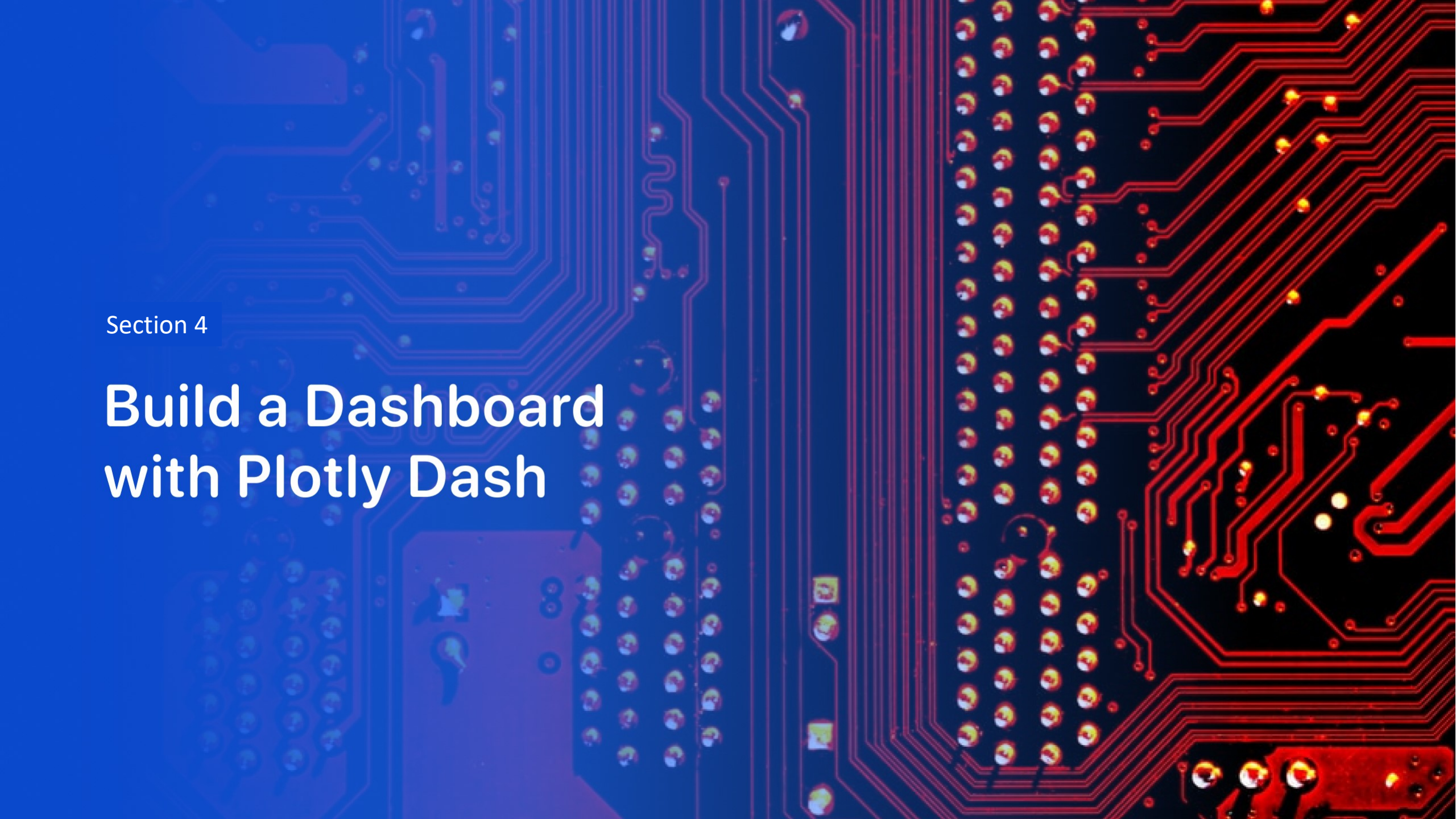




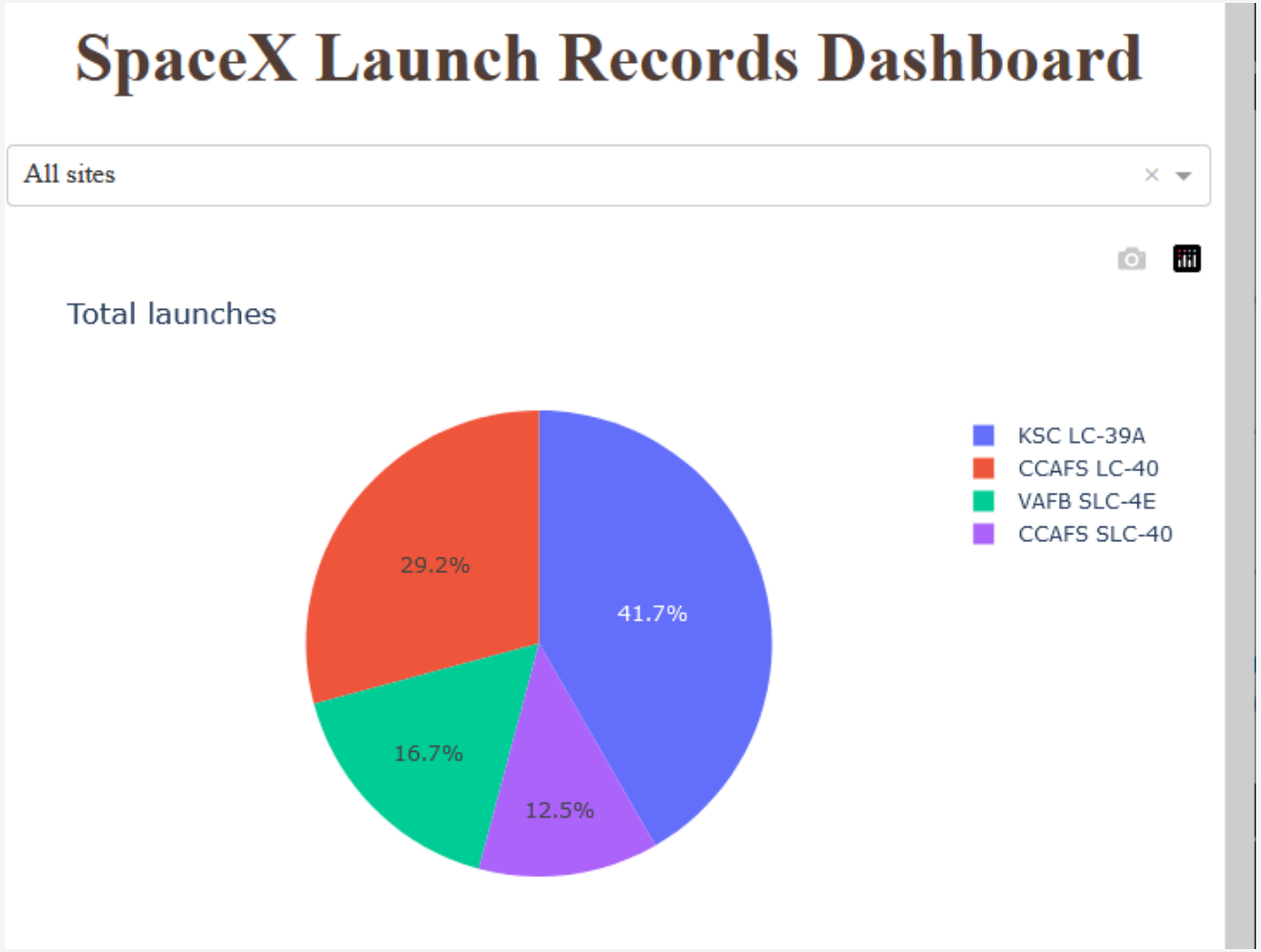
Launch Site distance to some landmarks

Launch Site in Florida are located near coastline (~0.86km) and railway (~1.28km) but far from city (nearest Orland is ~78km away)



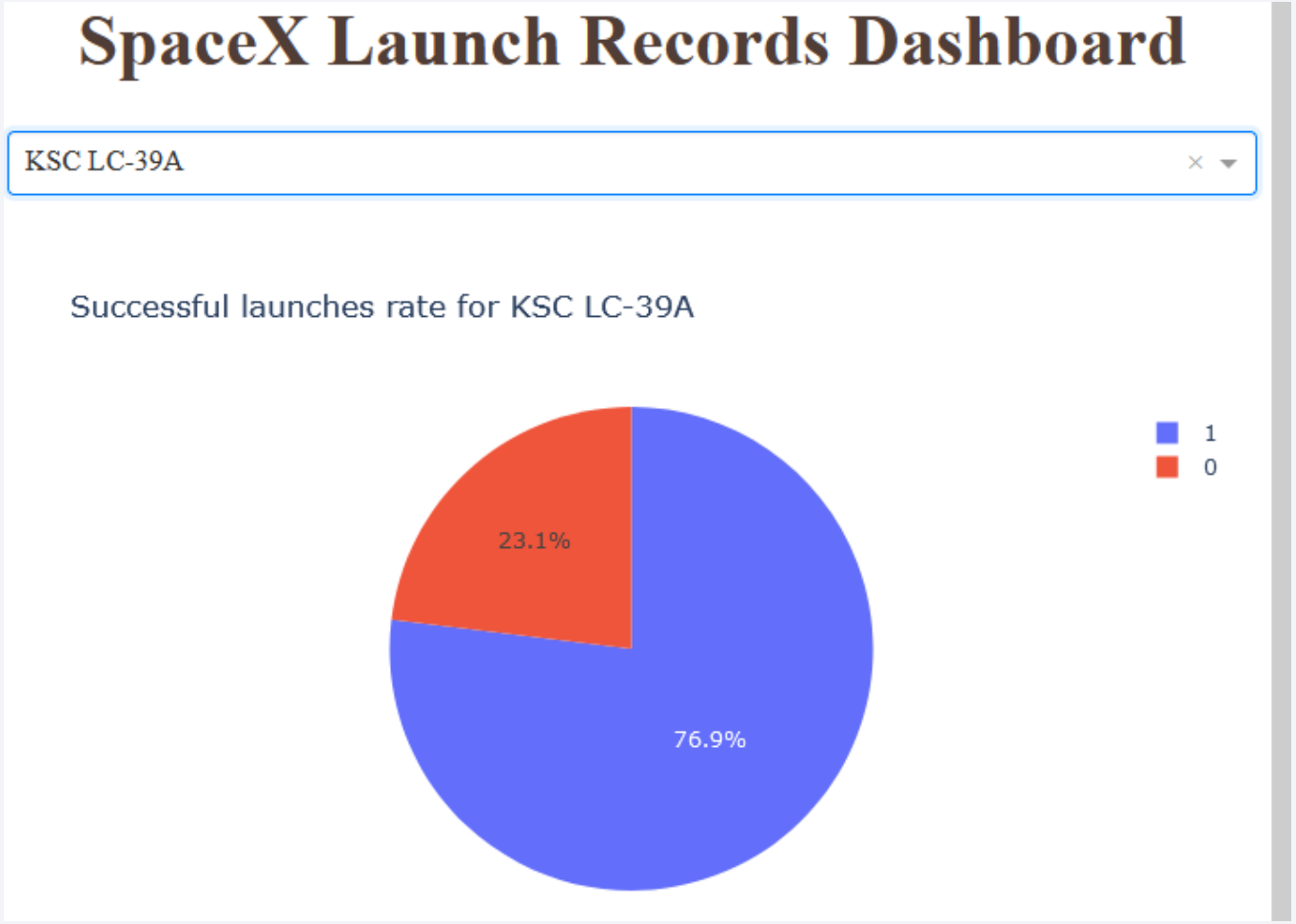

Section 4
Build a Dashboard with Plotly Dash

Success launch rate for all Sites



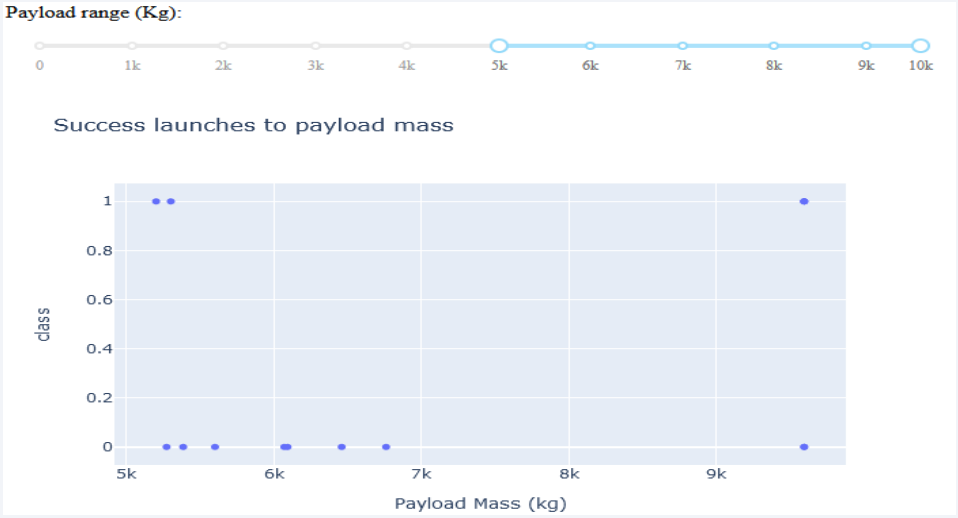
- KSC LC-39A is Site with most total launches (41.7%)
- CCAFS SLC-40 has the least launches of all Sites (12.5% of total)

<Dashboard Screenshot 2>



- KSC LC-39A as most used Launch Site has high success rate – 76.9%

Influence of Payload mass to launch outcome



- There is much more launches with payload mass < 5 000 kg
- Launch success rate is much more higher for payload mass < 5 000kg

Section 5

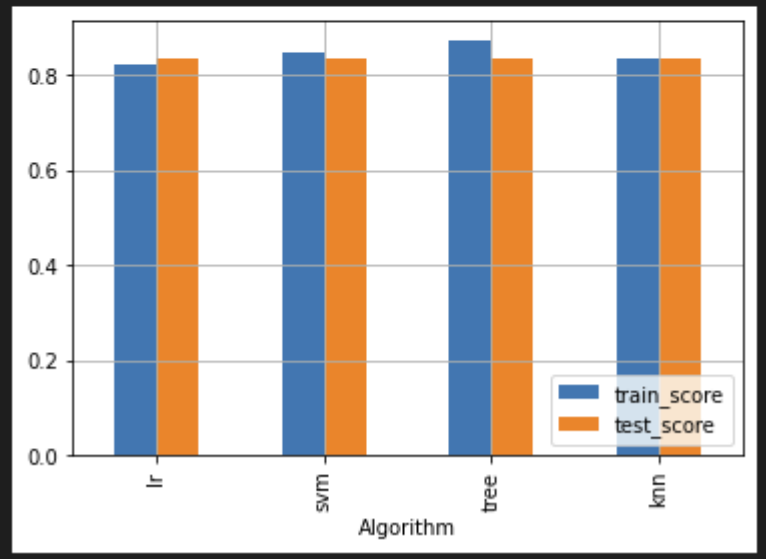
Predictive Analysis (Classification)

Classification Accuracy

```
scores_df.plot(
    x='Algorithm',
    y=['train_score', 'test_score'],
    kind='bar',
    grid=True
).legend(loc='lower right')
```

✓ 0.1s

```
<matplotlib.legend.Legend at 0x1fe5279f4c0>
```

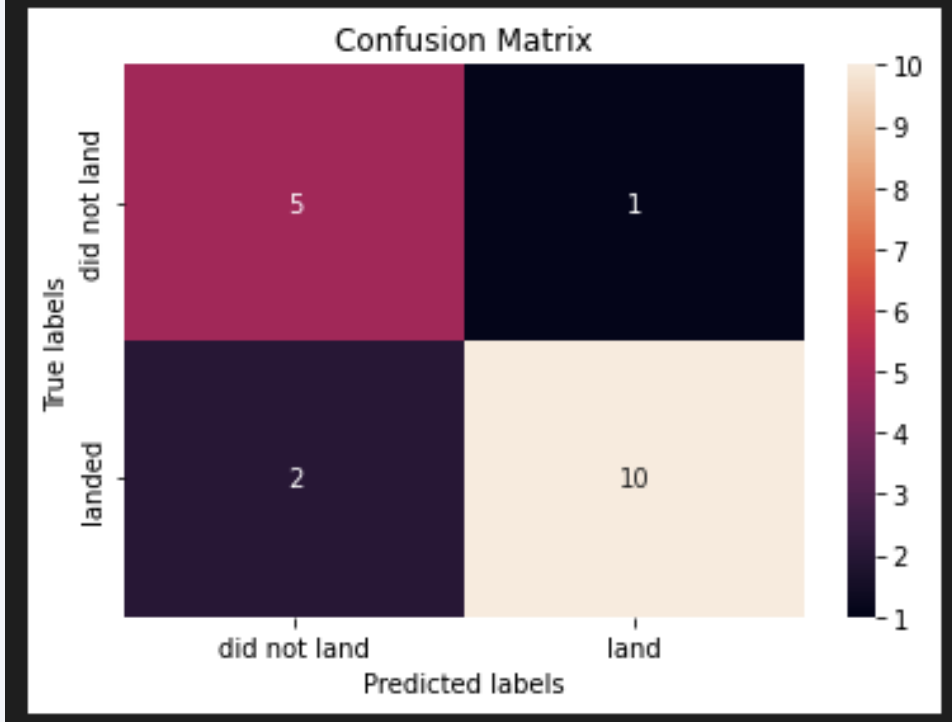


All 4 models have the same accuracy on test data (83%) however Decision Tree has better result on train data (87%).

Confusion Matrix

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```

✓ 0.1s



Decision Tree model correctly predicted 5 failed landings (True Negative) and 10 success landings (True Positive).

It incorrectly predicted success landing when it was failure in 2 records (False Positive) and one failure when it was success (False Negative)

Conclusions

- SpaceX landing success rate increasing every year
- Heavy payload mass used in late flights (above 70^{th}) have high success landing rate
- For this small dataset to predict landing succession all 4 models have similar accuracy
- Taking into account accuracy on training and testing dataset we can indicate Decision Tree as model to use.

Appendix

- None

Thank you!