IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

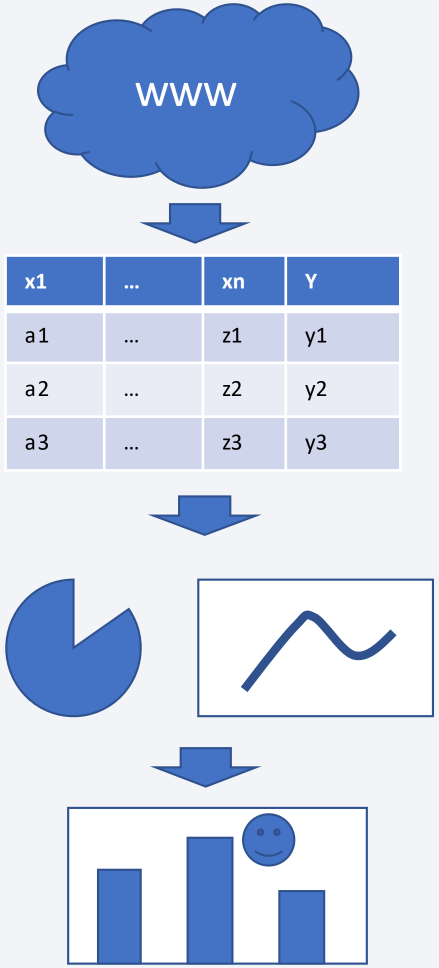
Bulat Gizatullin
29-10-2021

Outline

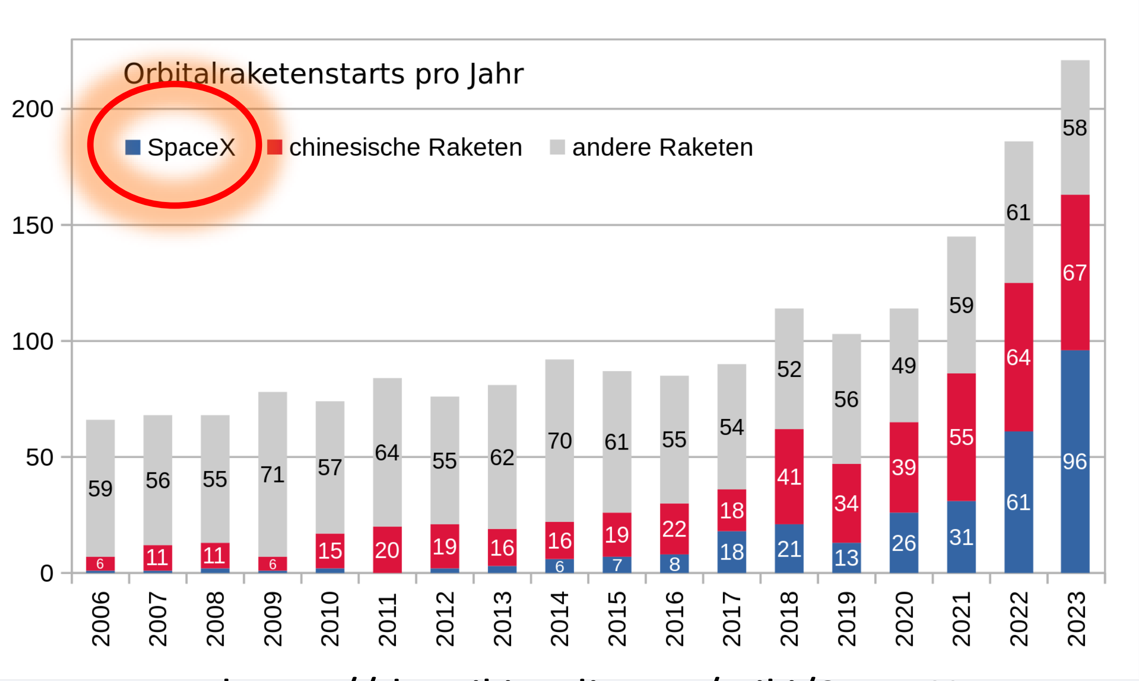
- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Methodologies
  - Data Collection using SpaceX API and web scraping,
  - Exploratory data analysis EDA using data wrangling, data visualization, interactive visual analytics (dashboard)
  - Machine learning prediction using logistic regression, KNN, decision tree, SVM
- Main results
  - The data obtained from above mentioned sources provide enough input for reliable prediction
  - Using EDA the features which correlates to launch success are obtained, such as payload, launch sites etc,
  - Different prediction metgods exhibit rather high accuracy on test data set, depending though on splitting parameters, which is related to relatively small dataset



Introduction



https://de.wikipedia.org/wiki/SpaceX

Competition of space launches

SpaceX is a leader with significantly lower launch costs of about 60 million USD compared to the competitors of approximately 160 million USD

The **first stage** saving and re-usage are the main reasons of low launch costs by SpaceX

Thus, **prediction** of first stage landing success can be used to define launch cost

- Problems:
  - ❑ Prediction of SpaceX launches success using dataset about previous launches
  - ❑ Finding parameters that can predict the launch outcome
  - ❑ The best method to predict launch outcome

Section 1

Methodology

Methodology

Executive Summary of Methodology

- **Data collection** methodology:
  - SpaceX API (Open Data base)
  - Web Scraping (Wikipedia)
- Perform **data wrangling**
  - Data were process by adding the outcome label depending on different data features
- Perform exploratory data analysis (**EDA**) using visualization and SQL
- Perform interactive visual analytics via **Dashboard** using Folium and Plotly Dash libraries
- Perform predictive analysis using **classification models**
  - Data needs to be first normalized and split into training and testing data sets
  - Further on fitting of training data set using different models, such as KNN, logistic regression, decision tree, support vector machine
  - The model is tested with test data set and analyzed in terms of accuracy metrics and confusion matrixes

GitHub repository of the project

Main folder:

https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024.git

GitHub URL of the completed **SpaceX API calls** notebook:
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-spacex-data-collection-api.ipynb

GitHub URL of the completed **web scraping** notebook :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-webscraping.ipynb

GitHub URL of the completed **data wrangling** related notebooks :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_labs-jupyter-spacex-Data_wrangling.ipynb

GitHub URL of the completed **EDA with data visualization** notebook :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

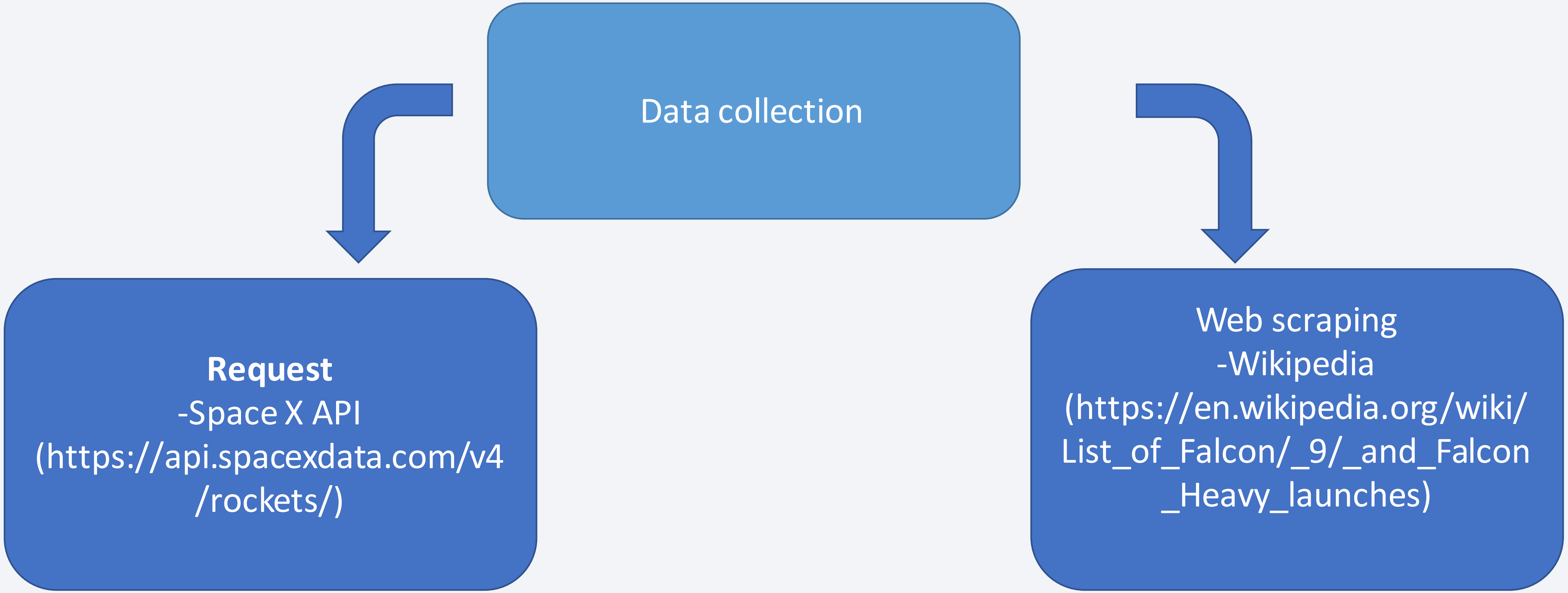
GitHub URL of the completed **EDA with SQL** notebook :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-eda-sql-coursera_sqllite.ipynb

GitHub URL of the completed **interactive map** with Folium map (please, use https://nbviewer.org/ to load the maps; Github does not support maps view) :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_lab_jupyter_launch_site_location.jupyterlite.ipynb

GitHub URL of the completed **Plotly Dash** lab :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/spacex_dash_app_BG1.py

GitHub URL of the completed **predictive analysis** lab :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_SpaceX_Machine_Learning_Prediction_Part_5.ipynb

Data Collection



GitHub URL of the completed **SpaceX API calls** notebook:
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-spacex-data-collection-api.ipynb

GitHub URL of the completed **web scraping** notebook :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-webscraping.ipynb

Data Collection - SpaceX API

Key points:

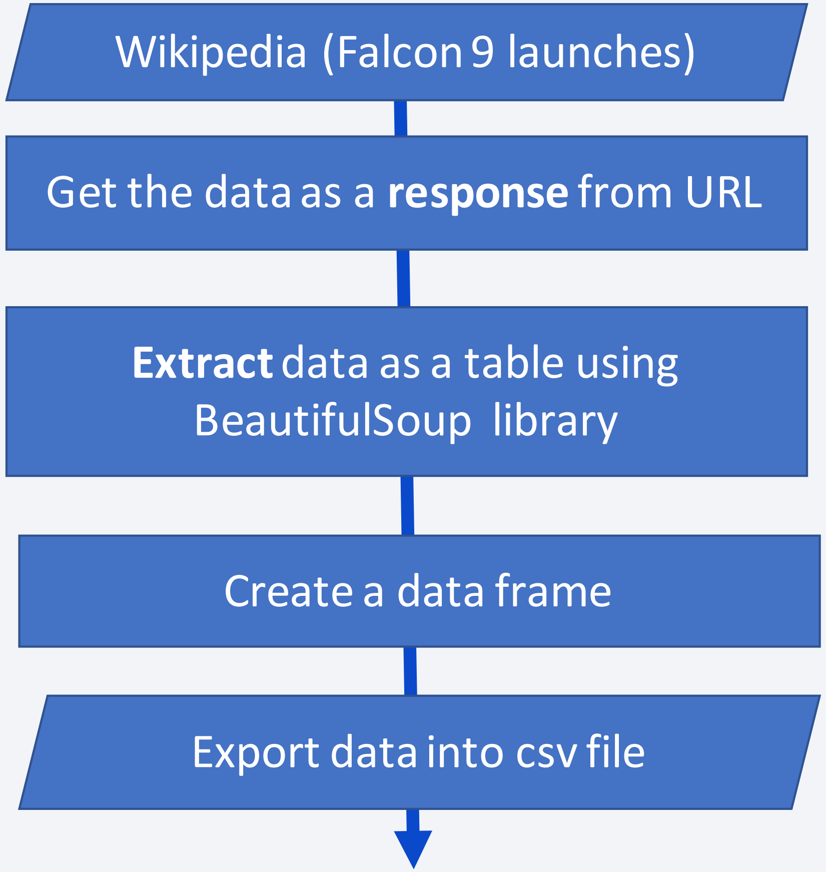
- Define helper functions
- Convert json to dataframe
- Construct new dataset using helper functions
- Confirm that all features has no missing values

GitHub URL of the completed SpaceX API calls notebook:
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

Key points:

- Define helper functions
- Verify BeatifulSoup object
- Separate Falcon 9 data
- Create dataframe from parsed dataset



GitHub URL of the completed **web scraping** notebook :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-webscraping.ipynb

Data Wrangling

Key points:

- Identify the features and their unique values
- Separate bad and successful outcome related features
- Create label for mission outcome for further classification analysis

GitHub URL of the completed **data wrangling** related notebooks :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_labs-jupyter-spacex-Data_wrangling.ipynb

EDA with Data Visualization

Tasks:
- Upload a dataset
- Find the features correlating with mission outcome

Charts for inspection:

- Payload v.s. Flight number (to see that success rate increases, bigger payload mass increase success rate)

- Launch sites v.s. Flight number (some launch sites are less risky)

- Launch site v.s. Payload (some launch sites are suitable for lighter payloads)

- Success rate v.s. Orbit type (which orbit launch are defined as more successful)

- Orbit type v.s. Flight Number (how success rate is changed overtime for particular orbit)

- Orbit v.s. Payload (similar to previous, higher payload increase successful outcome rate)

- Success rate v.s. date (positive trend overtime)

GitHub URL of the completed **EDA with data visualization** notebook :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

The following SQL queries were performed:

- First SQL query to remove blanks rows from table was performed
- Names of the unique launch sites in the space mission;
- Top 5 launch sites whose name 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 between 4000 and 6000 kg;
- 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; and
- Rank of the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20.

GitHub URL of the completed **EDA with SQL** notebook:
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Folium map objects

Markers
-indicate point on the map, such as launch sites

Circles
- Indicate highlighted area around specific coordinates, e.g. NASA JSC

Marker clusters
- group events in each place, such as successful or bas launches on a launch site

Icons
- Show the properties of an object, e.g. value of distance in km

Lines
- Indicate distance between two point on the map, e.g. between launch site and railroad

GitHub URL of the completed **interactive map** with Folium map (please, use https://nbviewer.org/ to load the maps; Github does not support maps view) :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

Dashboard

Dropdown menu
- Select the launch site
- Otherwise show the parameters for all launch sites

Pie chart
- Shows the total successful launches count for all sites
- Shows the success rate for selected launch site

Callback function 1
- Group the results for particular launch site is the latter is selected

Slider
- Select the range of payload mass

Scatter chart
- Shows the correlation between payload and launch sites

Callback function 2
- Group the results for particular launch site and chosen range of payloads

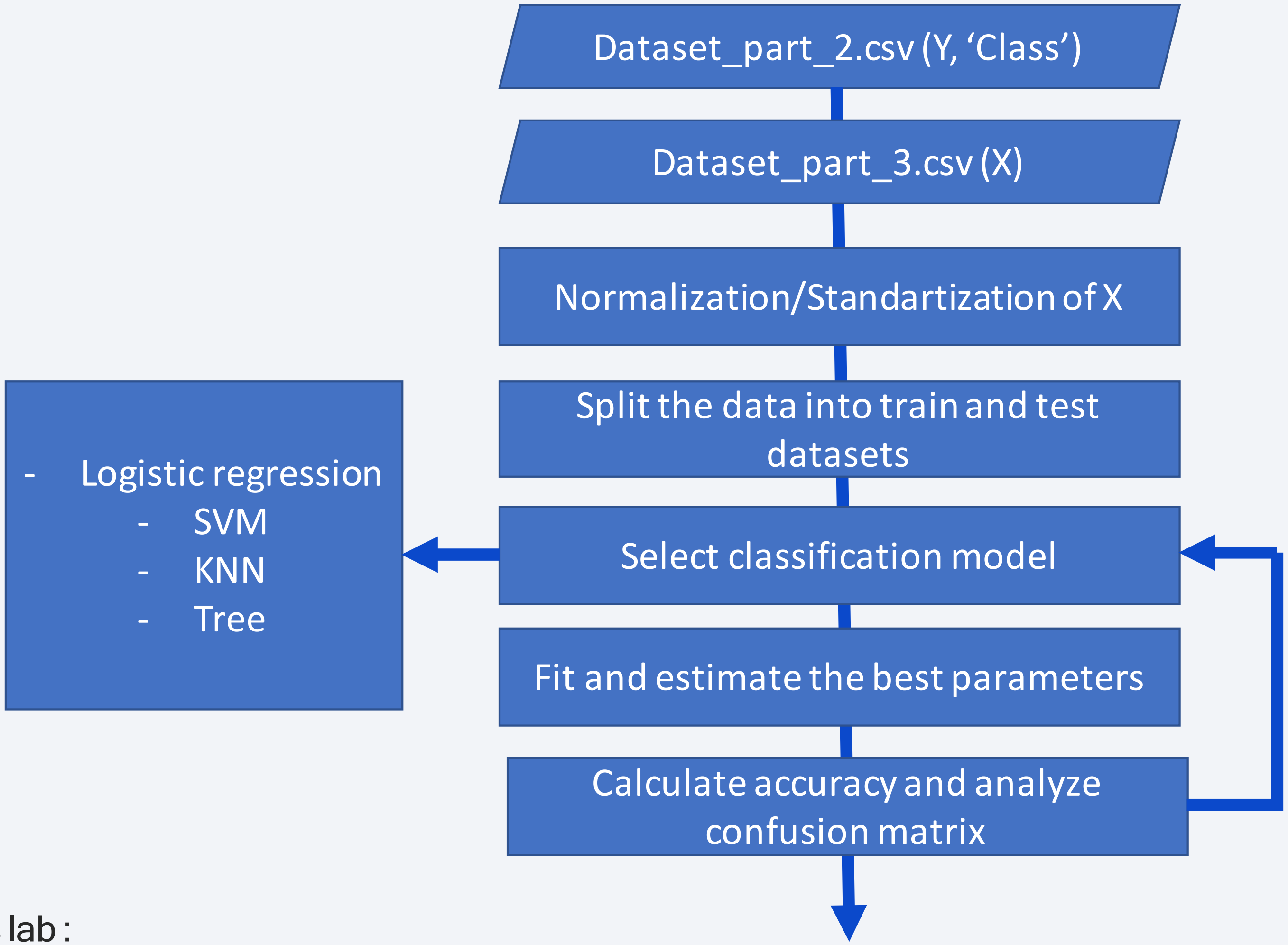
The built dashboard allows identifying the best place and payloads range for success launch

GitHub URL of the completed **Plotly Dash** lab :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/spacex_dash_app_BG1.py

Predictive Analysis (Classification)

Key points:

- Define confusion matrix plotting function
- Compare test and train accuracies
- Inspect effect of splitting parameters on the classification methods performance



GitHub URL of the completed **predictive analysis** lab :
https://github.com/bgizatul/Applied-Data-Science-Capstone-BulatG2024/blob/main/BG_SpaceX_Machine_Learning_Prediction_Part_5.ipynb

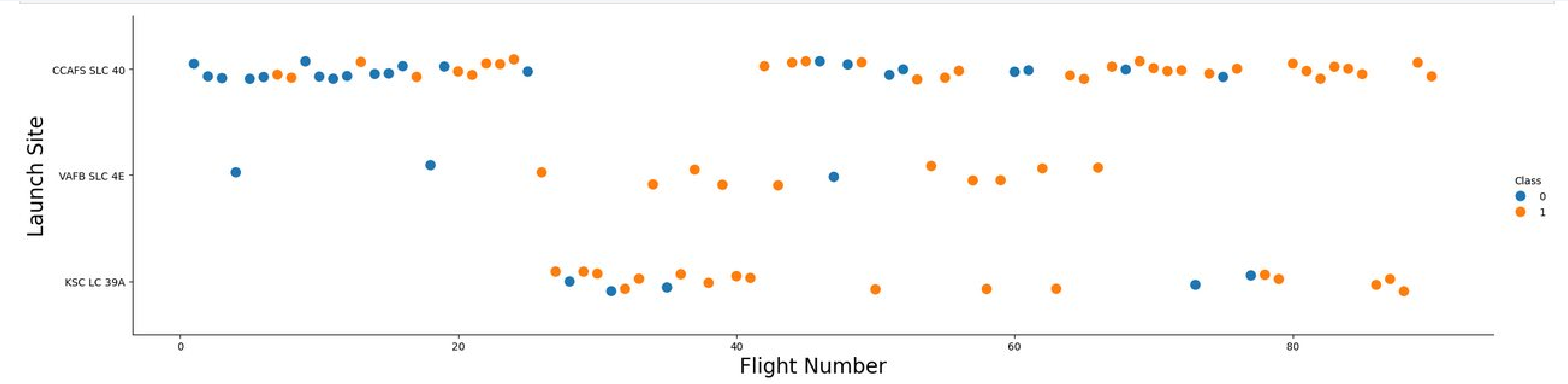
Results

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

Section 2

Insights drawn from EDA

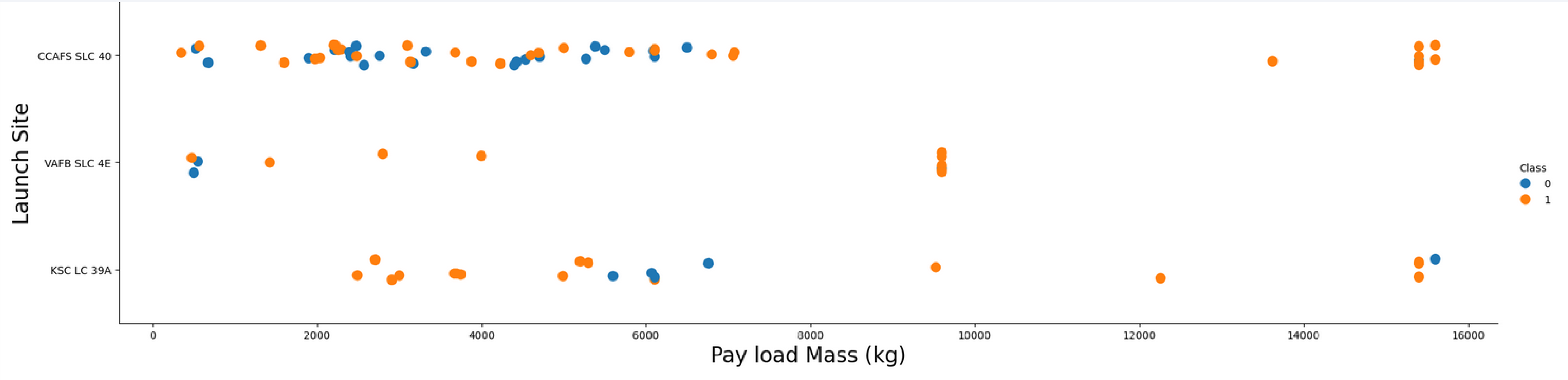
Flight Number vs. Launch Site



Correlation of launch site and flight number for successful and bad launches

- The chart shows :
  - successful rate over time for different launch sites
  - Number of flights for a particular launch site
- Preliminary conclusions:
  - successful rate increases over time
  - **CCAFS SLC 40** was used at the beginning of launches and shows the lowest success rate

Payload vs. Launch Site

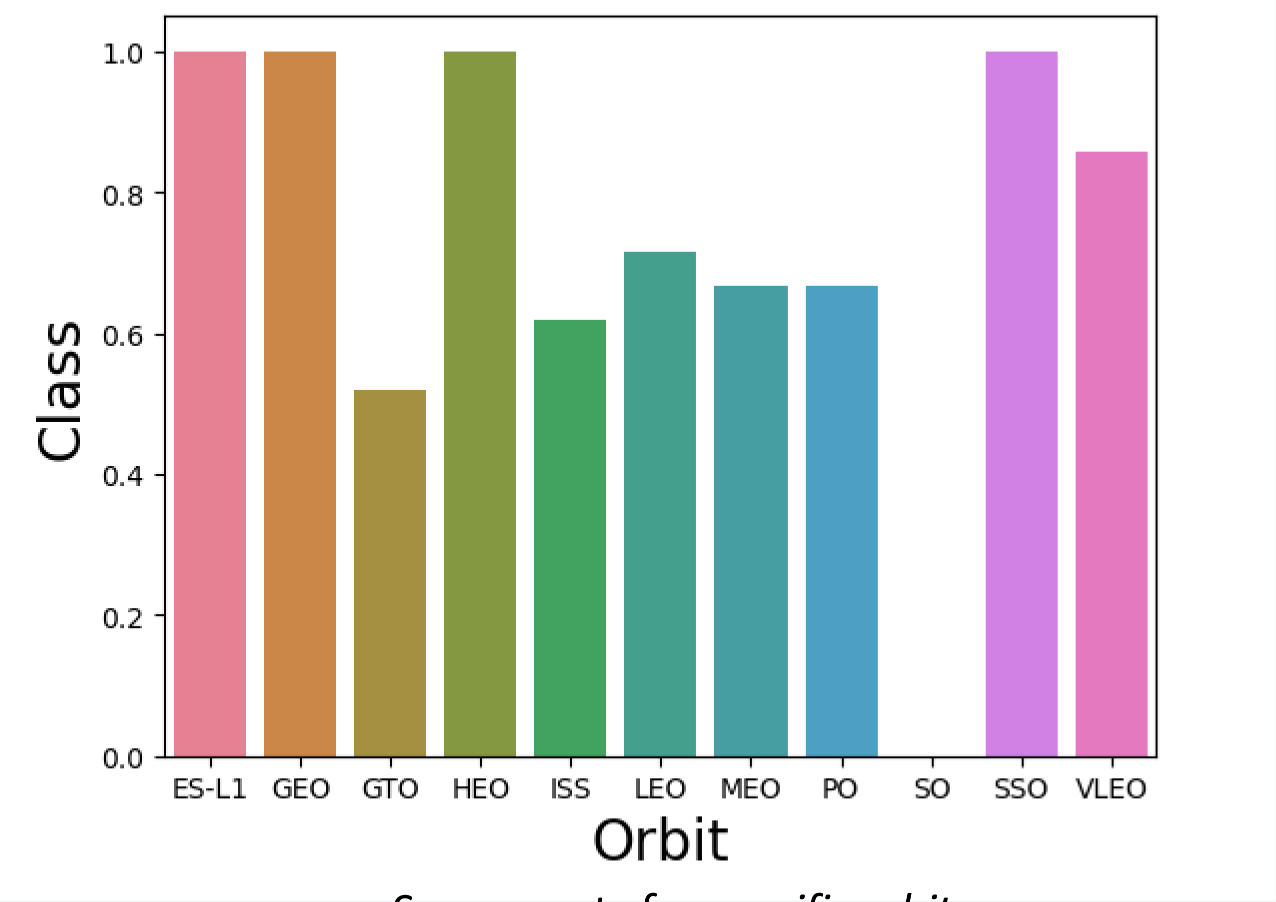


Correlation of launch site and payload mass for successful and bad launches

- The chart shows :
  - the successful rate for different launch sites depending on payload mass
- Preliminary conclusions:
  - successful rate increases for heavy, >7000 kg, payloads
  - If we consider that payload mass increases over time as well as success rate, the reason for less risky heavy launches is mostly related to launch technology development
  - More information about orbits, launch sites, payload, and their correlation is required

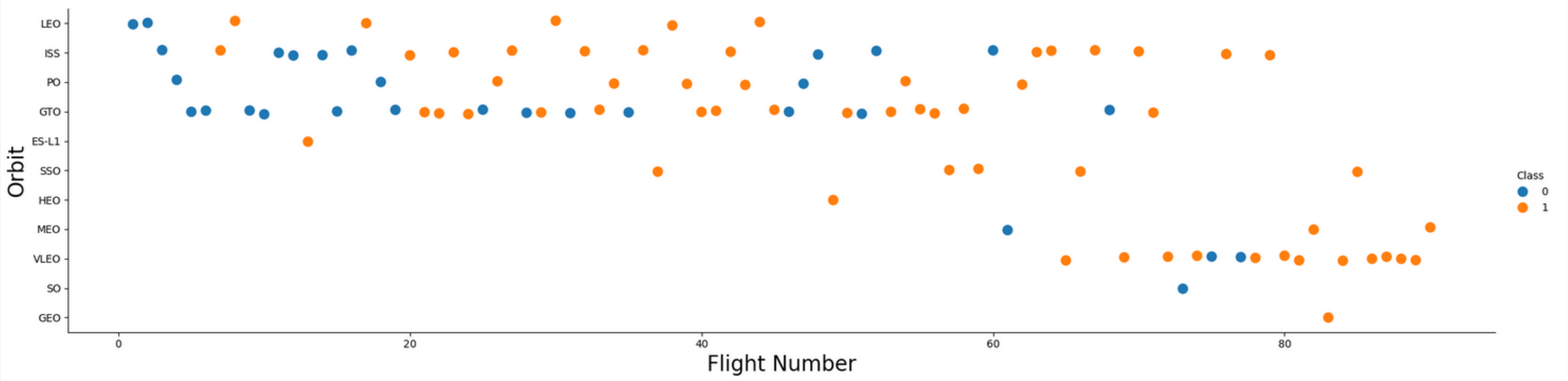
Success Rate vs. Orbit Type

- The chart shows :
  - the successful rate for different orbits
  - It should be analyzed in combination with the chart **Flight numbers v.s. Orbit type** (see the next slide), since for some orbits the number of flights is too low to be considered statistically relevant
- Preliminary conclusions:
  - The **GTO** has the lowest success rate with one of the highest flight number in orbit
  - The orbits with the **low flight** number must be ignored (see next slide)



Success rate for specific orbit

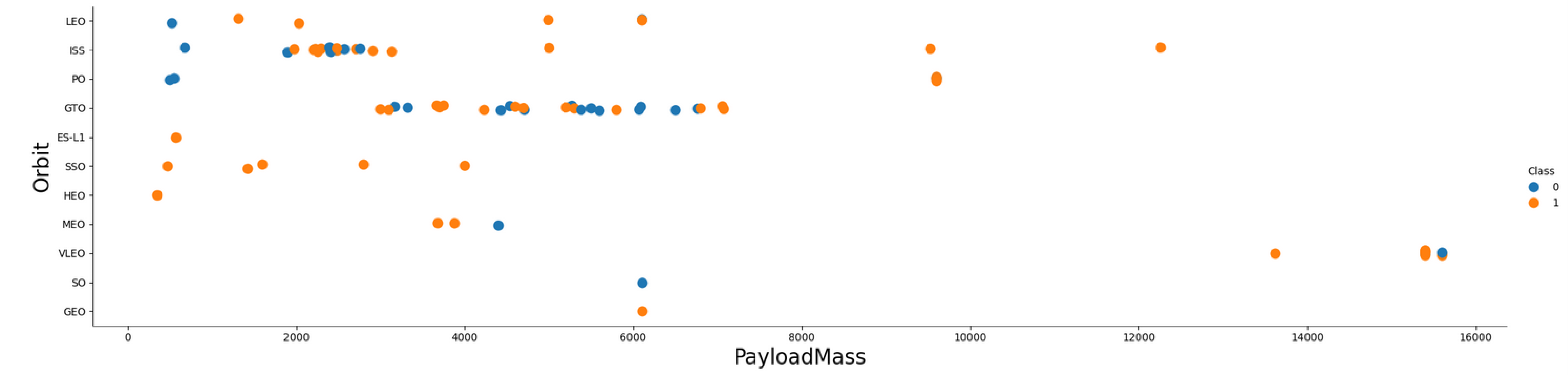
Flight Number vs. Orbit Type



Correlation of flight number and orbits for successful and bad launches

- The chart shows :
  - the successful rate for different orbits depending on flight number
- Preliminary conclusions:
  - The **GTO** has the lowest success rate with one of the highest flight numbers in orbit
  - The orbits GEO, SO, MEO, HEO, and ESL-1 have the lowest number of flights
  - **SSO** orbit with 5 flights has 100 % of success

Payload vs. Orbit Type

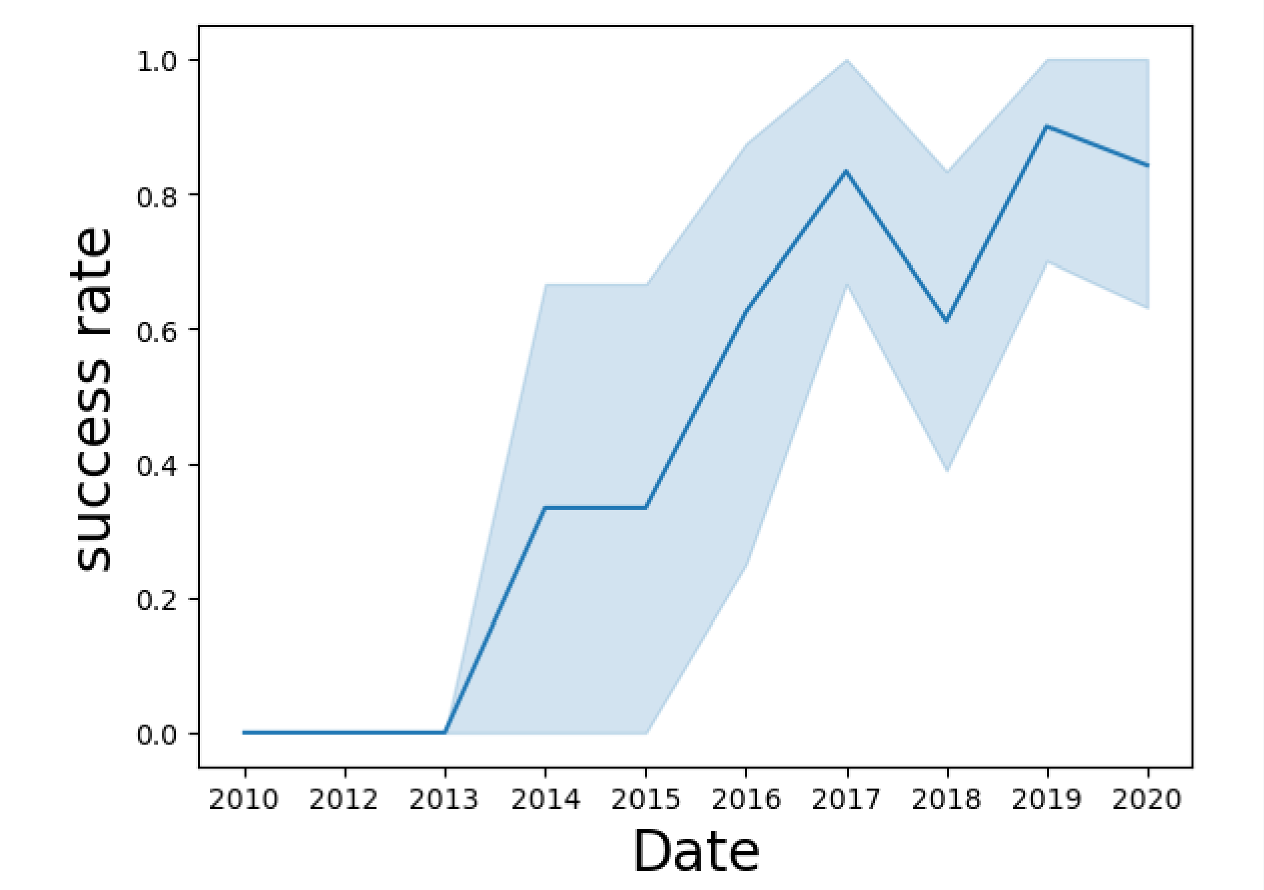


Correlation of payload mass and orbits for successful and bad launches

- The chart shows :
  - successful rate for different orbits depending on payload mass
- Preliminary conclusions:
  - The **GTO** has the lowest success rate showing no increase in time and the absence of a payload effect
  - Oppositely, **ISS** orbit shows an increase in success rate for higher payload mass as well as a positive tendency in time
  - **SSO** orbit with 5 flights has 100 % success and is better for low payload mass launches

Launch Success Yearly Trend

- The chart shows :
  - chronological changes in success rate
- Preliminary conclusions:
  - The Falcon 9 launch success rate has improved over time



Time dependency of success rate

All Launch Site Names

- According to the query, there are 4 distinct launch sites
  - CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, and CCAFS SLC-40

```
%sql SELECT DISTINCT Launch_Site FROM
```

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

Launch Site Names Begin with 'CCA'

- First 5 launches from CCAFS LC-40

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

Total Payload Mass

- Below is the total payload mass launched for NASA (CRS)

```
%sql SELECT SUM("Payload_mass__kg_") FROM SPACEXTABLE where "Customer" = "NASA (CRS)"
```

SUM("Payload_mass__kg_")
45596

Average Payload Mass by F9 v1.1

- The average payload mass launched using booster F9 v1.1
- Considering previous results the average payload mass for F9 v1.1 is lower than the approximate boarder above which the launches show a higher success rate

```
%sql SELECT AVG("Payload_mass__kg_") FROM SPACEXTABLE where "Booster_Version" = "F9 v1.1"
```

AVG("Payload_mass__kg_")
2928.4

First Successful Ground Landing Date

- The first successful landing on a ground pad was carried out approximately 5 years after the first launches

```
%sql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE  "%Success%ground%pad%"
```

MIN("Date")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- The FT series of boosters shows the higher success rate of landing on drone ships when the payload is between 4000 and 6000 k

```
%sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE  "%Success%drone%ship%" AND "Payload_mass__kg_" > 4000 AND "Payload_mass__kg_" < 6000
```

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

Total Number of Successful and Failure Mission Outcomes

- In general, we have around 100 mission outcomes, which enough for statistical analysis

```
%sql SELECT COUNT("Mission_Outcome") from SPACEXTABLE where "Mission_Outcome" LIKE "%Success%" or "Mission_Outcome" LIKE "%Failure%"
```

COUNT("Mission_Outcome")
101

Boosters Carried Maximum Payload

- The booster of F9 B5 version carried the maximum payloads
- According to previous results, F9 B5 booster can show the higher success rate

```
%%sql SELECT "Booster_Version"
FROM SPACEXTABLE
WHERE "Payload_mass__kg_" = (SELECT MAX("Payload_mass__kg_") from SPACEXTABLE)
```

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

2015 Launch Records

- In 2015 there was a couple of failed landings on drone ship with F9 v1.1 booster

```
%%sql
SELECT substr("Date", 6,2) AS "Month",
"Landing_Outcome",
"Booster_Version",
"Launch_site"

FROM SPACEXTABLE
WHERE
substr(Date,0,5)='2015'
AND
"Landing_Outcome" LIKE "Failure%drone%ship%"
```

Month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- This time range is for a stable increase in the success rate of launches (see Slide 23)
- Despite of growing success rate, the success and failed landings outcome is about 50/50

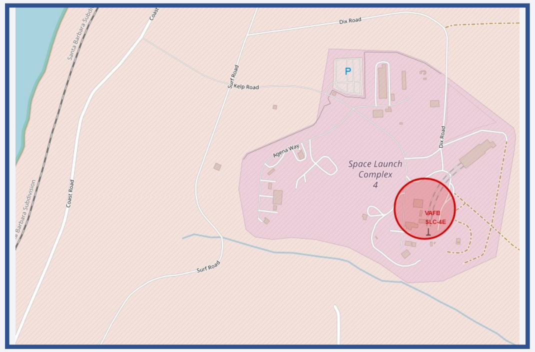
```
%%sql
SELECT "Landing_Outcome",
COUNT (*) AS lo_count
FROM SPACEXTABLE
WHERE "Date" BETWEEN "2010-06-04" AND "2017-03-20"
GROUP BY "Landing_Outcome"
ORDER BY lo_count DESC
```

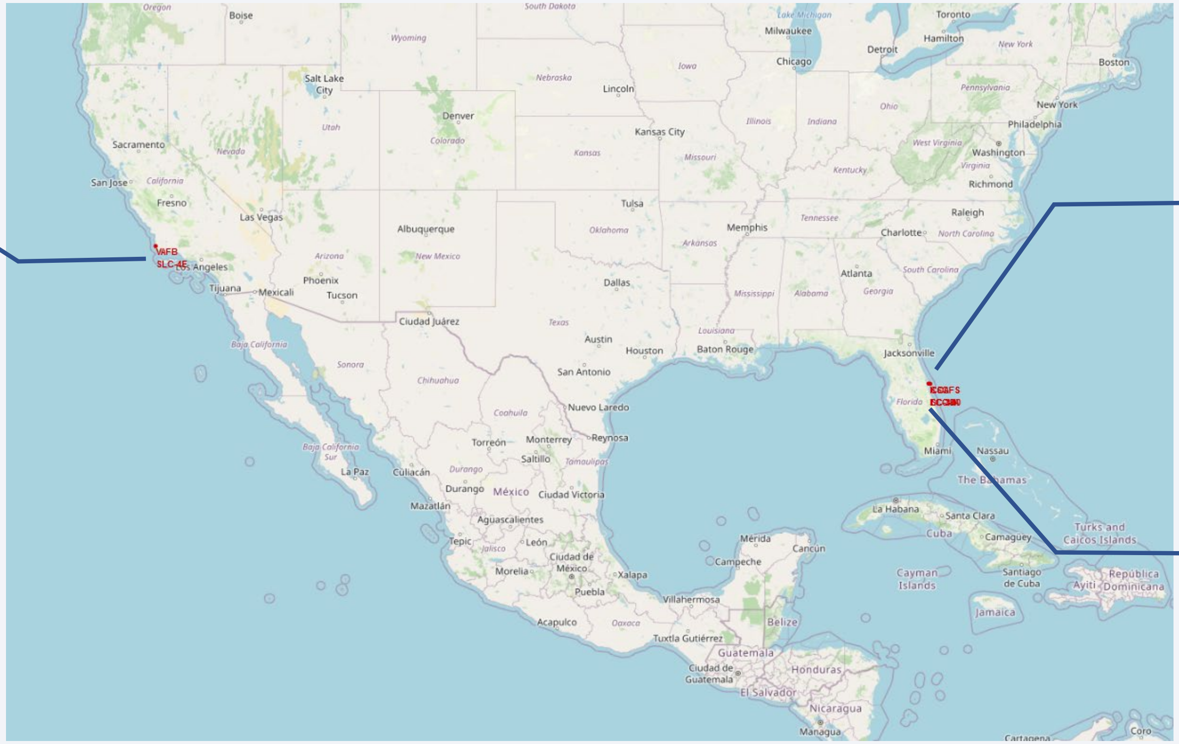
Landing_Outcome	lo_count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

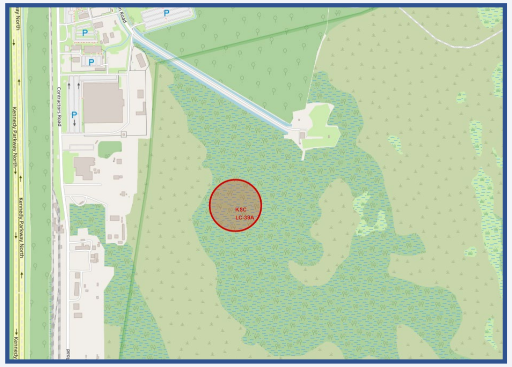
Section 3

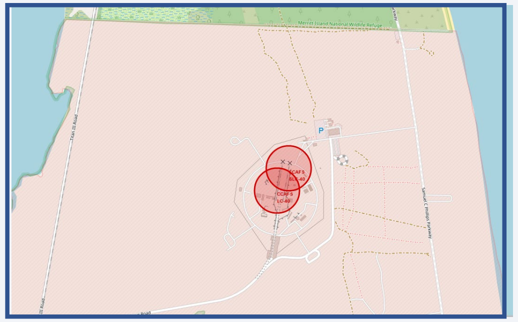
Launch Sites Proximities Analysis

Launches sites of SpaceX missions





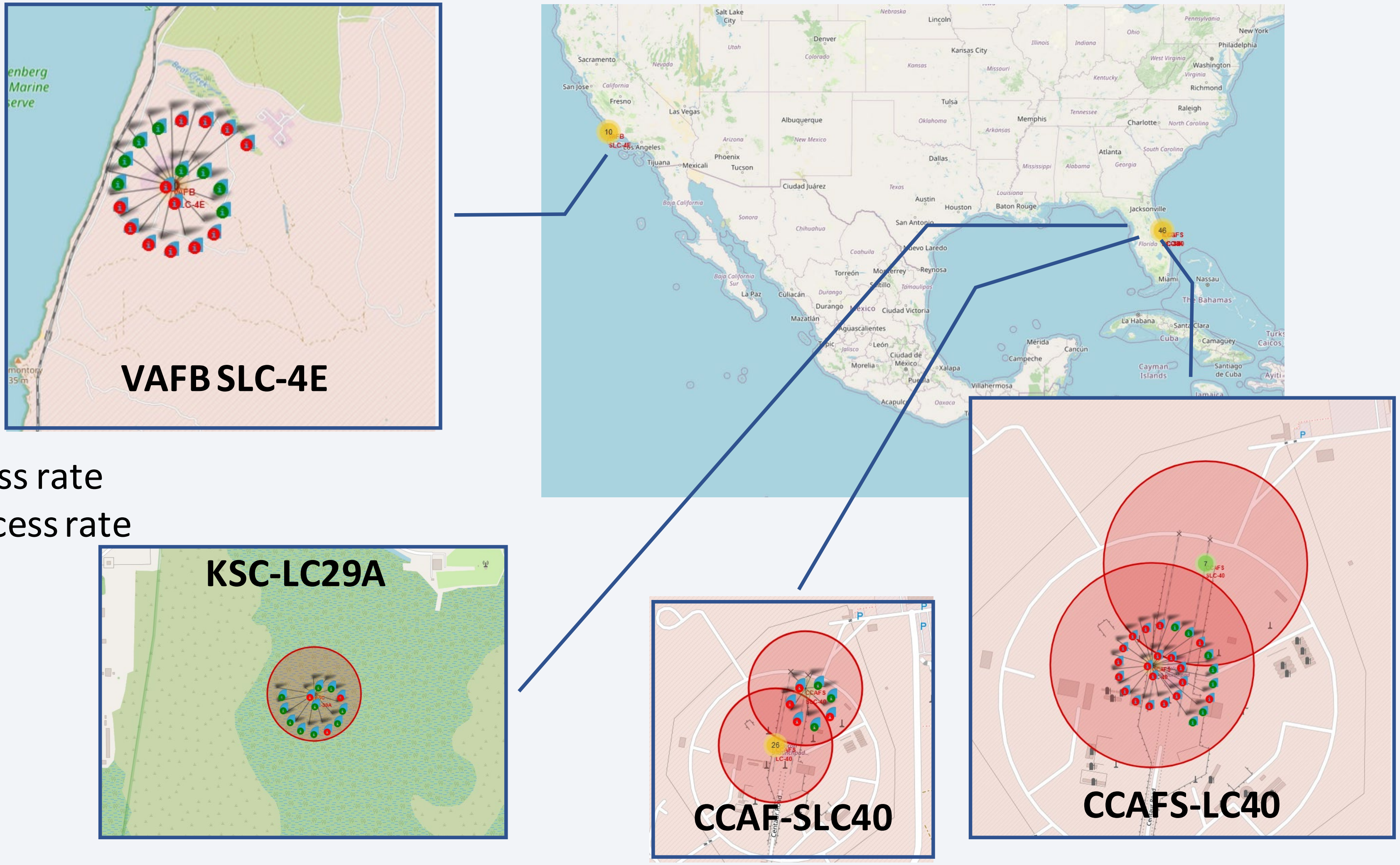




- **VAFB SLC-4E: Vandenberg Space Launch Complex 4 (CA)**

- **KSC-LC29A: Kennedy Space Center - Merritt Island (FL)**
- **CCAFS-LC40: Cape Canaveral Launch Complex 40 (FL)**
- **CCAF-SLC40: Cape Canaveral Space Launch Complex 40(FL)**

Success/failed launches for each site



- **KSC-LC29A** shows the highest success rate
- **CCAFS-LC40** exhibits the lowest success rate

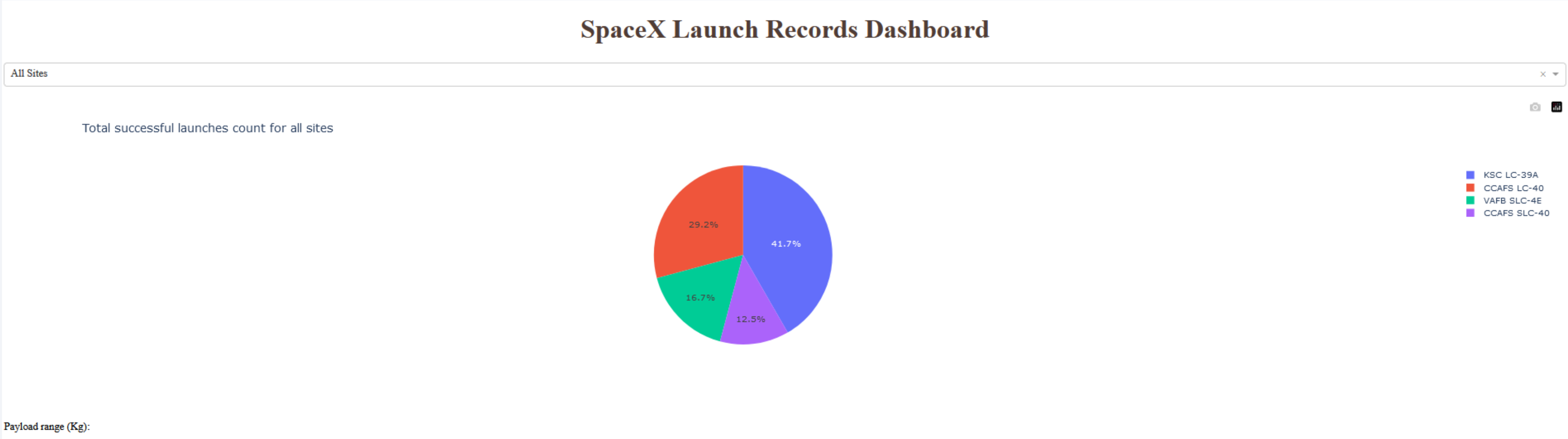
Distances from launch sites to its proximities

- Launch sites are close to coasts for safety purposes
- Launch sites are rather far from populated areas for protecting the population from dangerous accidents
- Launch sites are close to railroad to decrease transportation costs

Section 4

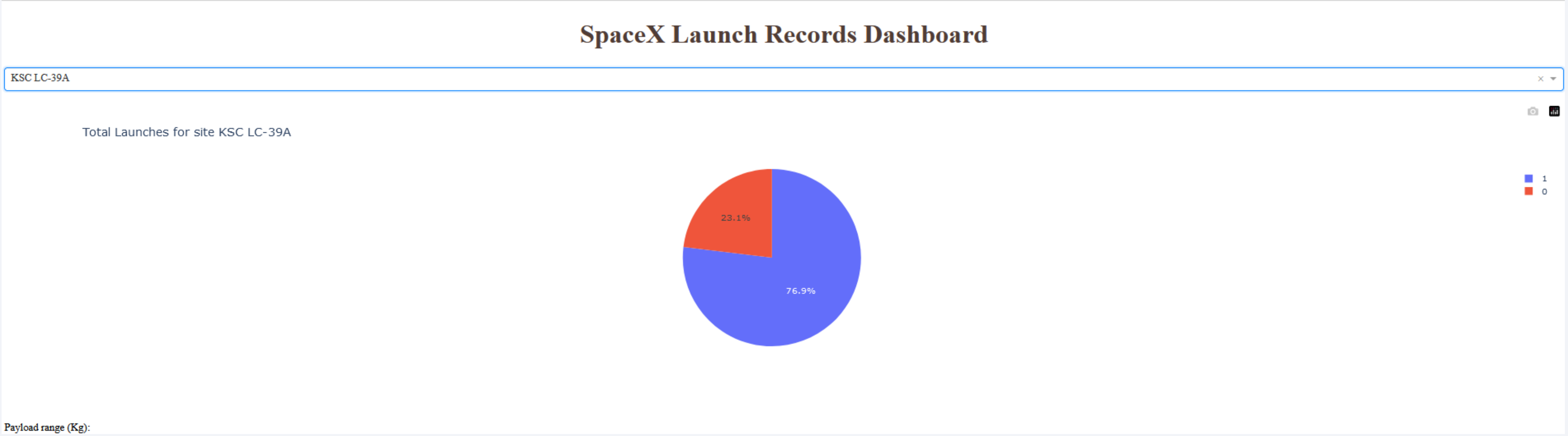
Build a Dashboard with Plotly Dash

Total successful launches for all sites



KSC LC-39A launch site has the highest number of successful launches of about 42% of the total 101 launches

Success rate of launches for KSC LC-39A



KSC LC-39A is characterized by high success rate of about 77%

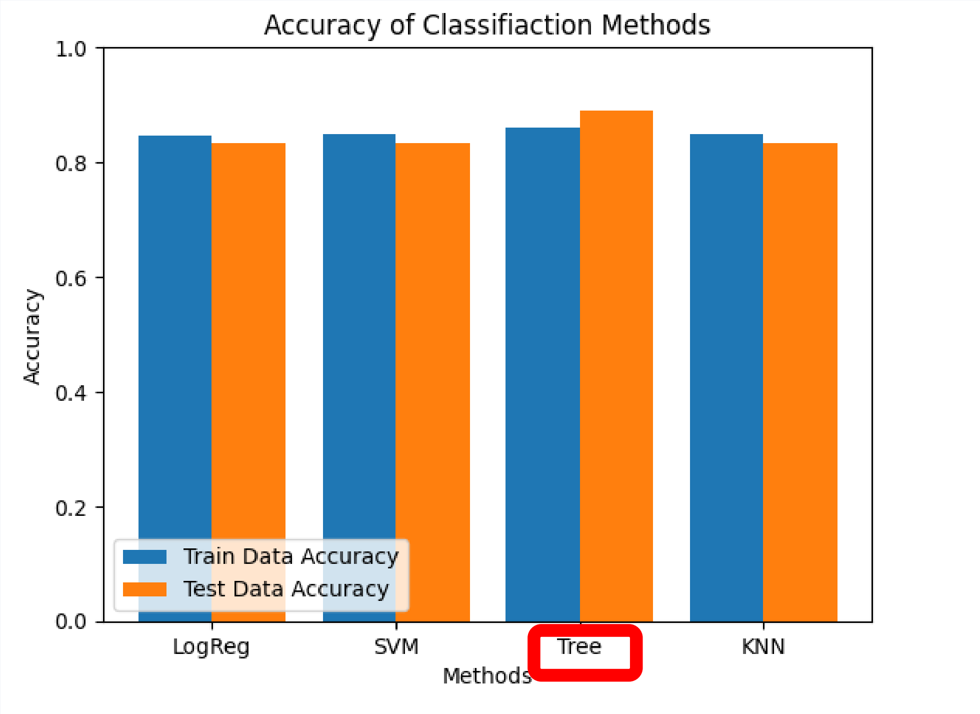
Effect of Payload mass on success rate

- Most of the successful launches were done with payloads **less than ~5500 kg** and using **FT** Version of the Booster, which is a more developed and advanced booster version
- Oppositely, **V1.0 and V1.1** show low success rates as perhaps a first version of the booster with low reliability

Section 5

Predictive Analysis (Classification)

Classification Accuracy



Comparison of accuracy of prediction classification methods

Methods	Train Accuracy	Test Accuracy
LogReg	0.846	0.833
SVM	0.848	0.833
Tree	**0.861**	**0.889**
KNN	0.848	0.833

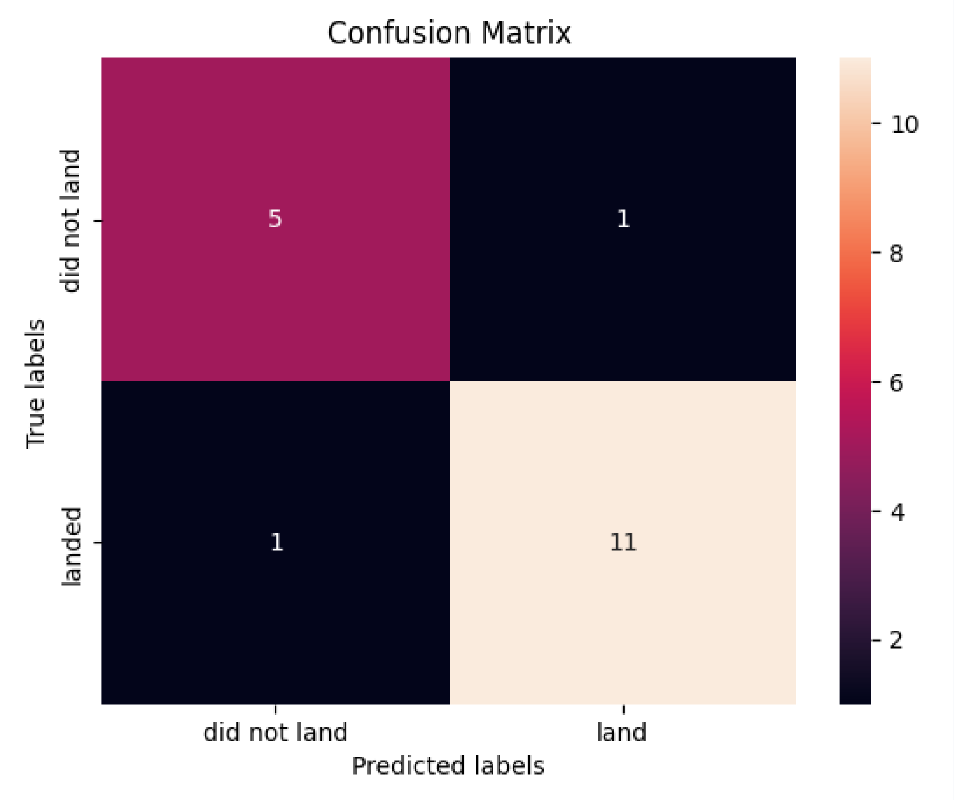
Decision Tree classification model exhibits the highest accuracy for both train and test data sets

The accuracy of prediction is above **88 %** which allows rather trustworthy prediction of successful launch

The metrics of others models are also at reliable level, showing though **similar** values* of accuracy

* see the Appendix

Confusion Matrix



Confusion matrix of Decision Tree classifier

Confusion matrix of **Decision Tree** classification model shows large* numbers of true positive and true negative predictions

False positive and negative with only 1 results provides reliable accuracy of predictions

For comparison, other models shows no false positive predictions, while equal number for negative prediction provides **poor quality** of those models

* see the Appendix

Conclusions

- The data analysis was successfully performed using data sets obtained from Wikipedia and open database
- The preliminary analysis shows the correlation of launch outcome with parameters such as launch site, payloads, and orbits, while the rate of successful launches increases over time, showing a positive trend in technology development
- Preliminary insight from EDA shows that
  - The best place for launches is KSC LC-39A
  - The payloads above ~ 7000 kg are more successful launches
  - The latest versions of boosters are more successful in landing and further re-usage
  - GTO orbit launches are more risky

- The trained classification model decision tree classifier exhibits an accuracy is about 88 %
- The test accuracy of other ML classification algorithms strongly depends on data set preparation and splitting (e.g. using random_state), showing accuracy of prediction up to 94 %, which can be the subject of further analysis

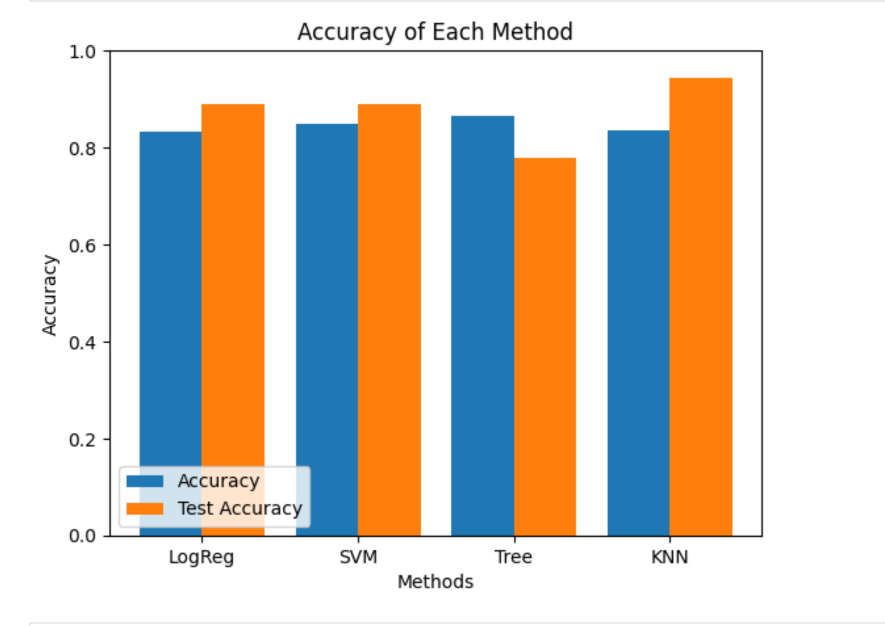
Appendix

Classification Accuracy

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y,  test_size = 0.2, random_state = 2)
```

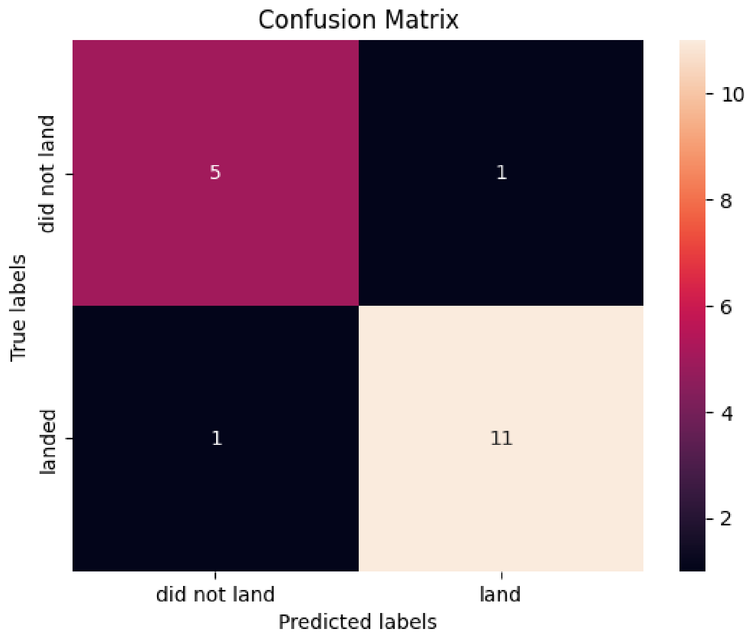
```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y,  test_size = 0.2, random_state = 3)
```

With random state 3
All confusion matrix are different
- KNN is the best methods with 94 % accuracy

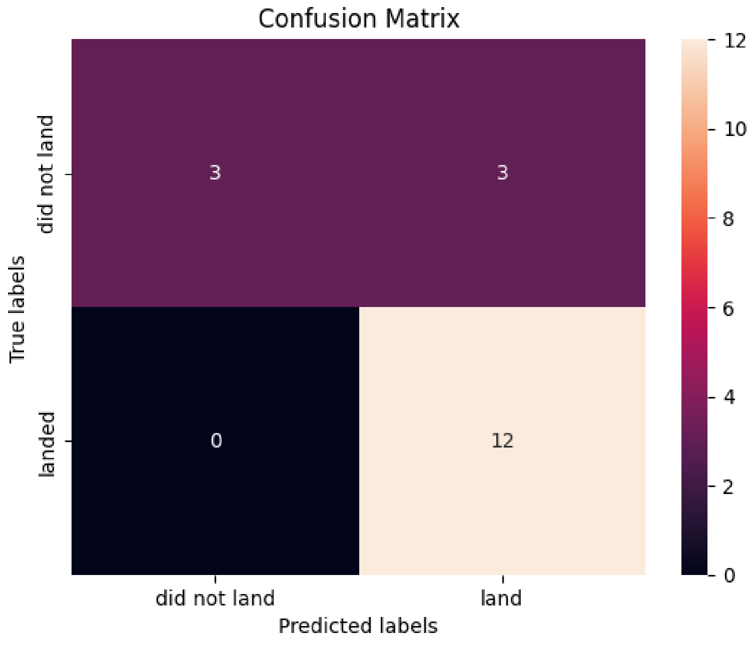


The accuracy of Decision tree model is always higher, but accuracy using test data varies a lot randomly, while the metrics of others model are stable

Confusion Matrix of Tree vs. Other models



Confusion matrix of Decision Tree classifier



Confusion matrix of Logistic Regression, SVM, and KNN models

Thank you!