Winning Space Race with Data Science

Bertha Walters
01/24/2023

Outline

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

Executive Summary

- This presentation will aim to build a classification model to predict whether a launch will land successfully or not
- Will explore relationships between outcome and several factors like orbit, payload, launch site
- 4 models were evaluated, resulting in Accuracy around 83%
- Out of all the models explored, Decision Tree had the best results on the training dataset. However, all models performed equally on the test dataset.

Introduction

- This project is aimed at determining the price of each launch for SpaceY by predicting whether if the first stage of the SpaceX Falcon 9 rocket will land successfully.
- If the rocket lands successfully, it can then be recovered and reused, significantly lowering the price of the launch.
- This will allow SpaceY to make a more informed bid against SpaceX for a rocket launch

Section 1

Methodology

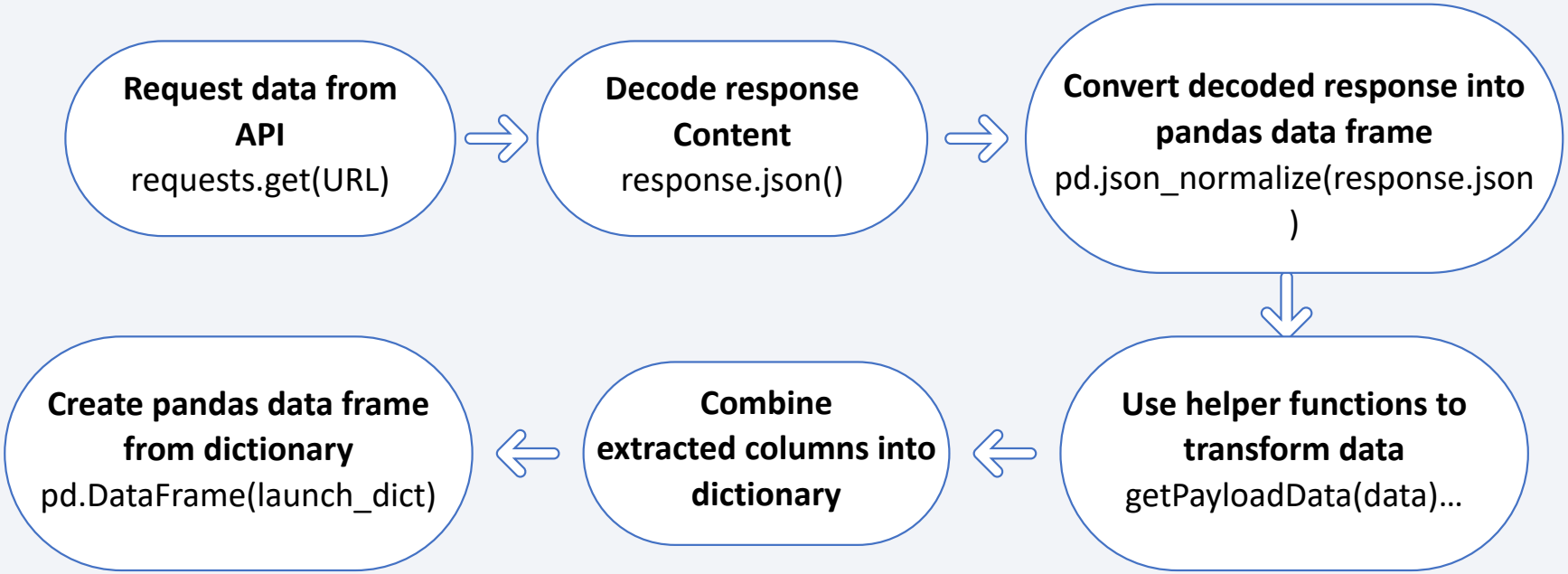
Methodology

Executive Summary

- Data collection methodology:
  - Collection using the SpaceX REST API.
  - Collection via webscraping wikipedia site
- Perform data wrangling
  - Data was evaluated for null values and these were replaced by the mean of the variable
  - Target variable was converted into binary by consolidating multiple categories into Pass/Fail criteria depending on whether the rocket landed successfully or not
- 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 – SpaceX API

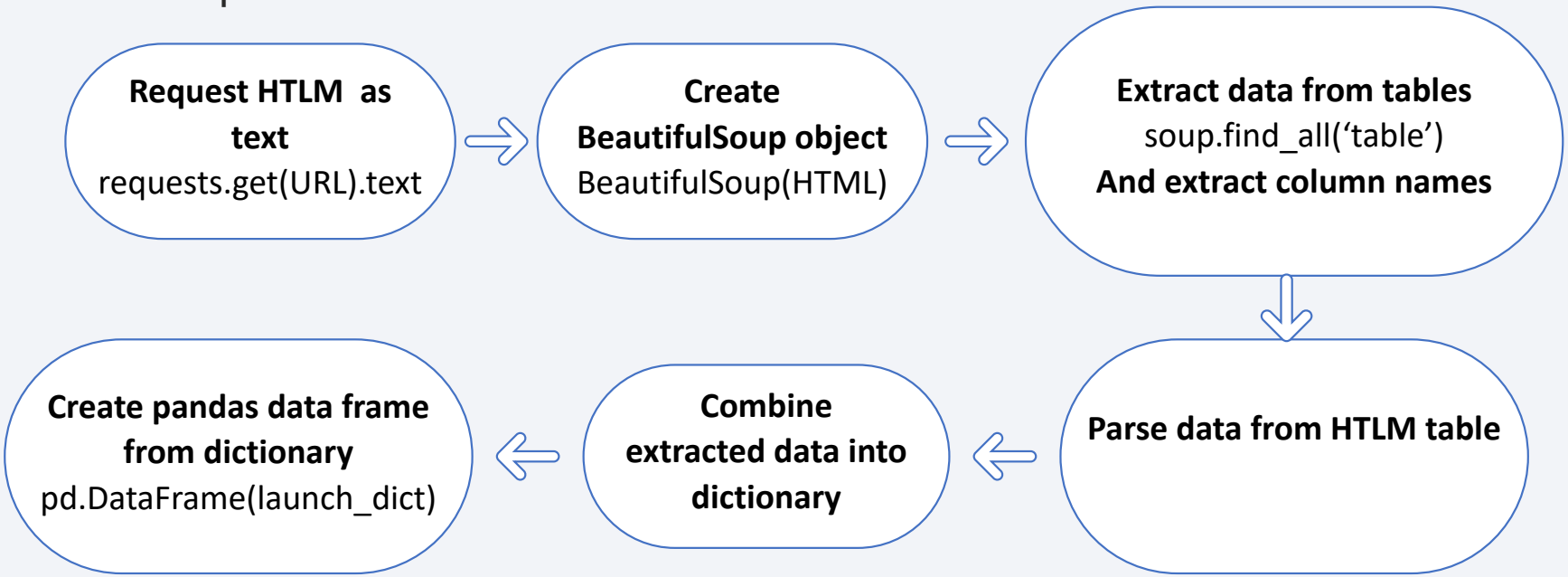
- Data Collection using SpaceX API and helper functions
  - Data extracted: booster name, launchpad name and coordinates, payload, outcome



https://github.com/berthawalters/ibmdscapstone/blob/master/SpaceX-API-Data-Collection.ipynb

Data Collection - Scraping

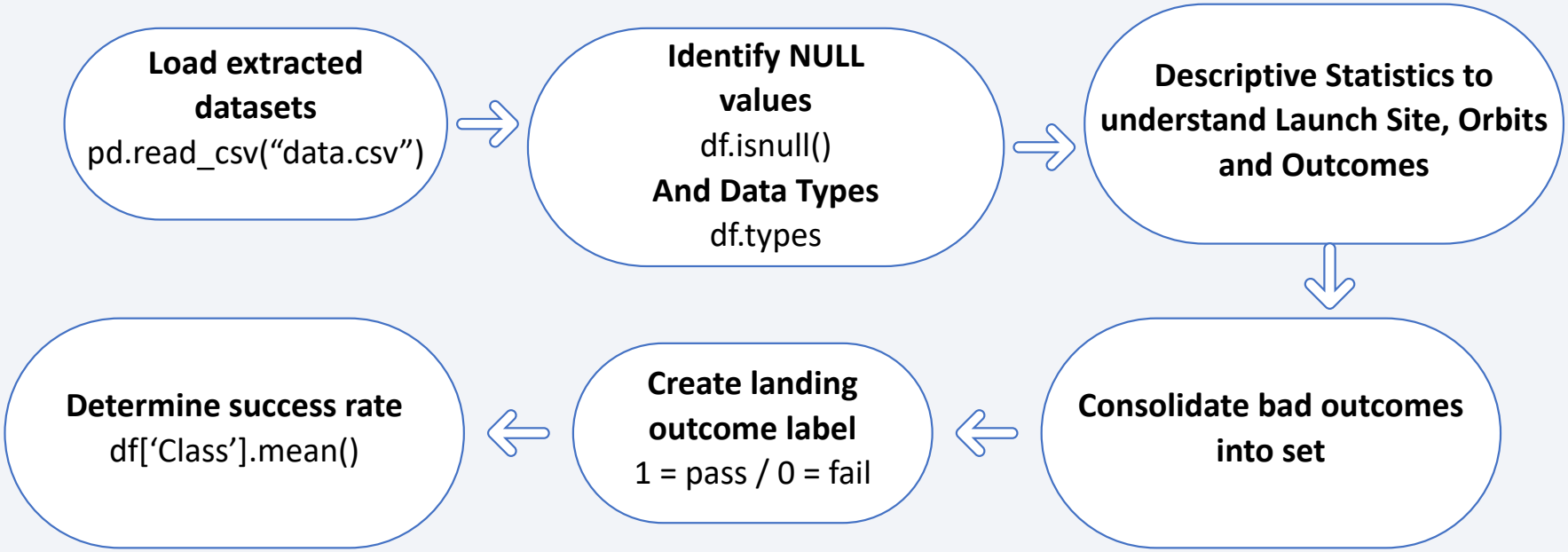
- Data Collection using webscraping to extract historical Falcon 9 launch records from Wikipedia HTML table



https://github.com/berthawalters/ibmdscapstone/blob/master/SpaceX-Webscraping-Data-Collection.ipynb

Data Wrangling

- In this step the dataset was evaluated for null values, data types were confirmed and the outcome was consolidated into a binary variable to be used for modeling



https://github.com/berthawalters/ibmdscapstone/blob/master/Data%20Wrangling.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose

EDA with SQL

- Display unique launch sites
- Display 5 records where launch site begins with CCA
- Display total payload carried by boosters launched by NASA (CRS)
- Display average payload carried by booster version F9 v1.1
- Display date of first successful in ground pad landing
- Display names of boosters which have success in drone ship and payload between 4000 and 6000
- Display total number of successful and failed mission outcomes
- Display booster_versions which have carried the maximum payload mass
- Display month, booster version and launch site for failed landing outcomes in 2015
- Rank the count of successful landing outcomes between 04/06/2010 and 20/03/2017 in descending order

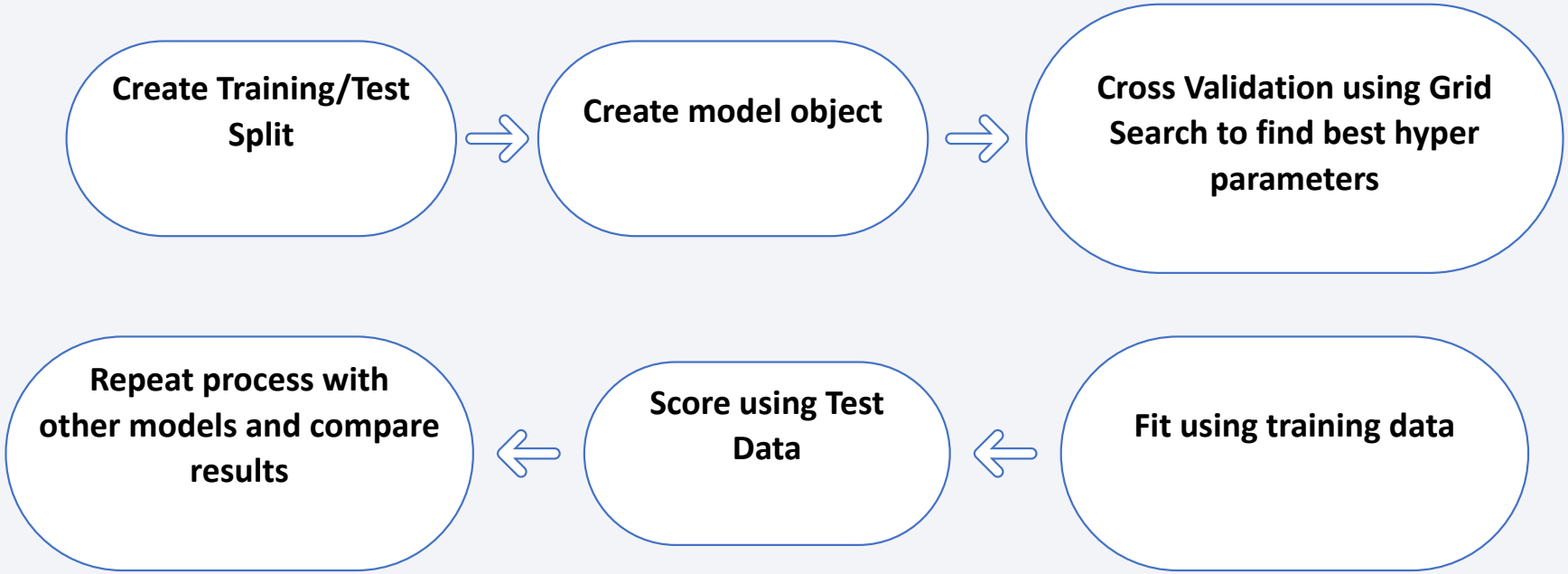
https://github.com/berthawalters/ibmdscapstone/blob/master/EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- Added the following features:
  - Red folium circle and name label at NASA Johnson Space Center's coordinates
  - Red folium circle and name labels for each launch site
  - Marker cluster to show different information for overlapping coordinates
  - Color coded markers to show landing outcome. Green = Successful, Red = Unsuccessful
  - Polyniline to show distance between launch site and key locations like a railway, highway, etc
- All the objects were added to understand the problem from a geospatial perspective and to provide insight into landing outcomes for each site.

https://github.com/berthawalters/ibmdscapstone/blob/master/Launch%20Site%20Folium.ipynb

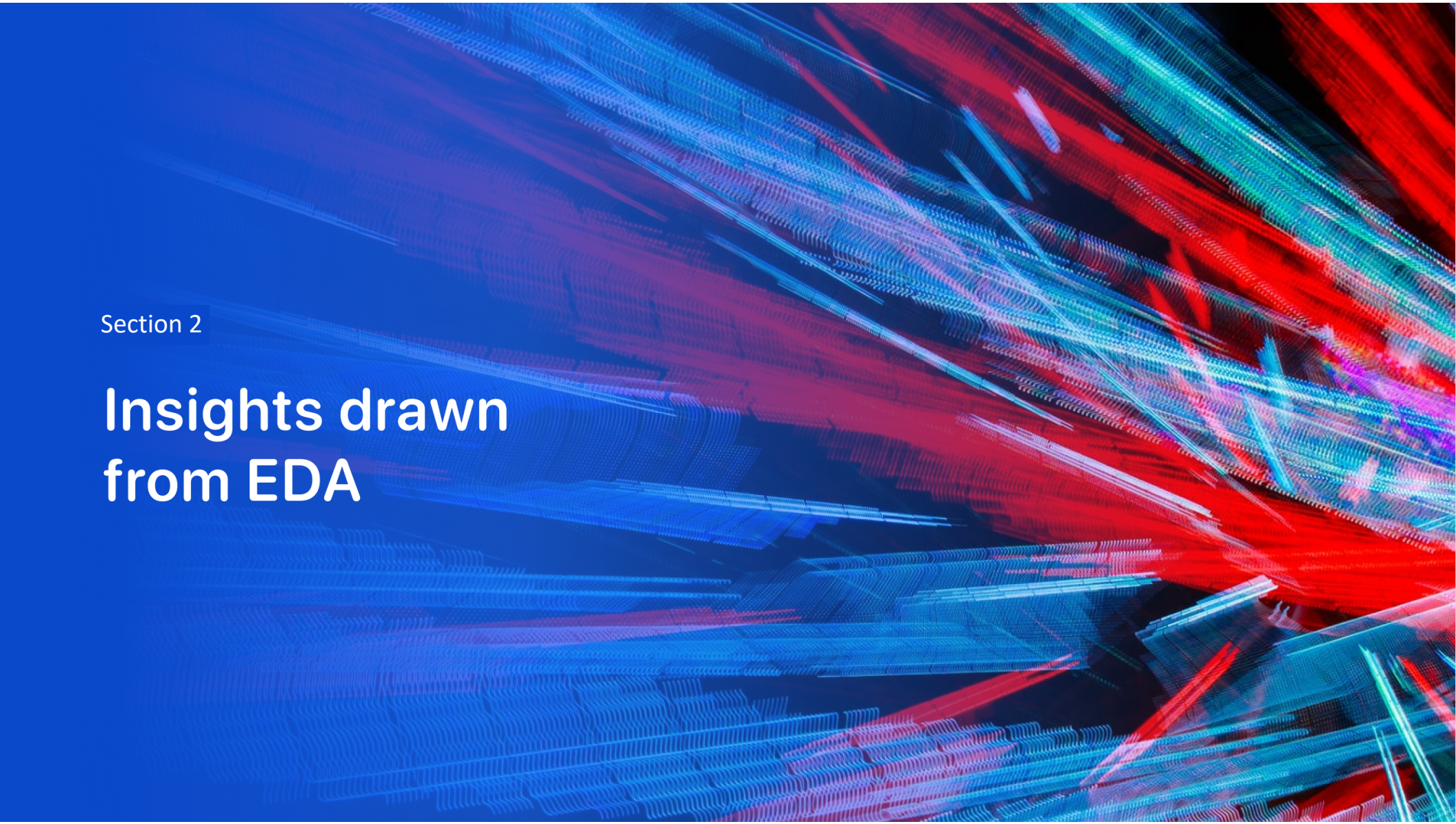
Predictive Analysis (Classification)



https://github.com/berthawalters/ibmdscapstone/blob/master/Machine%20Learning%20Model.ipynb

Results

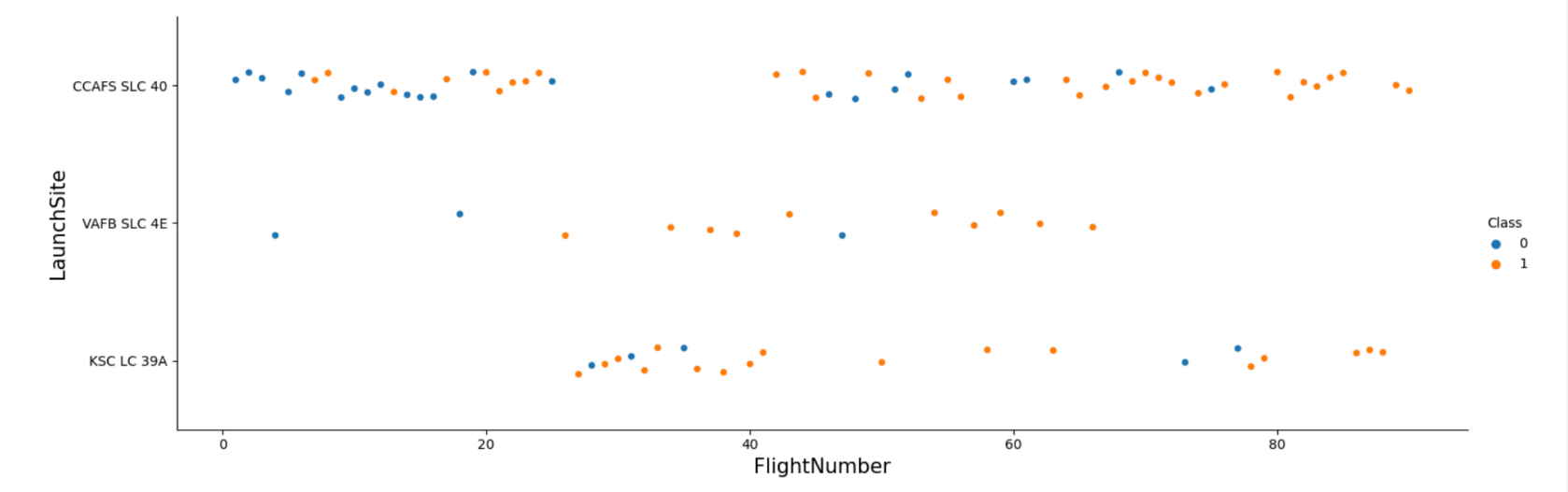
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



Section 2

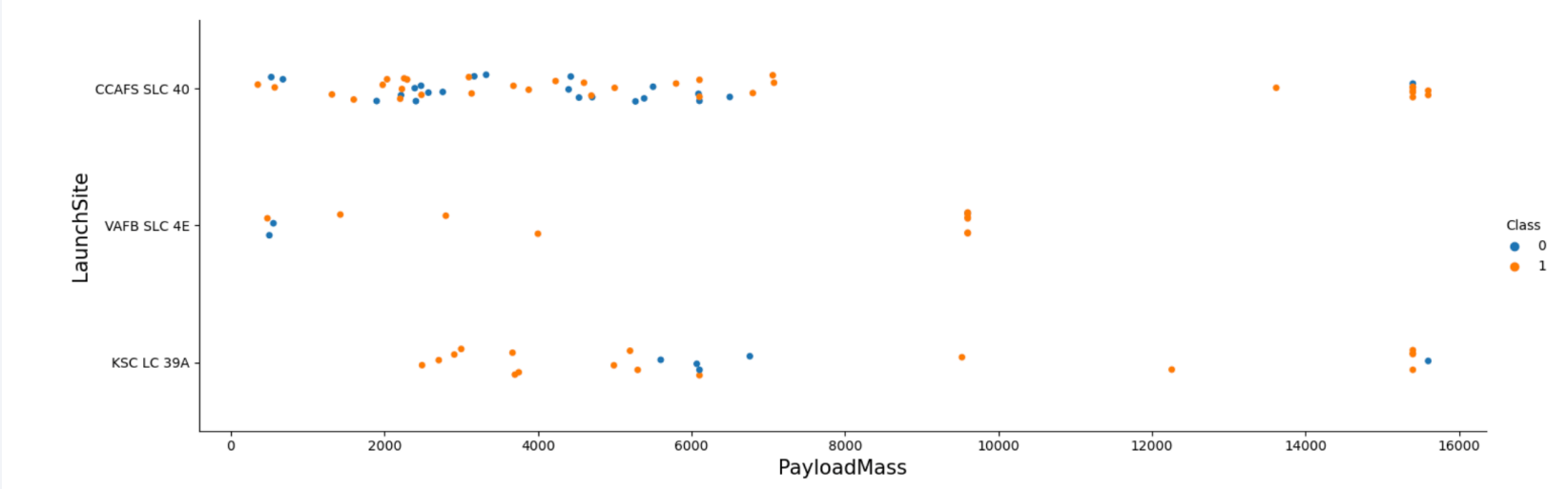
Insights drawn from EDA

Flight Number vs. Launch Site



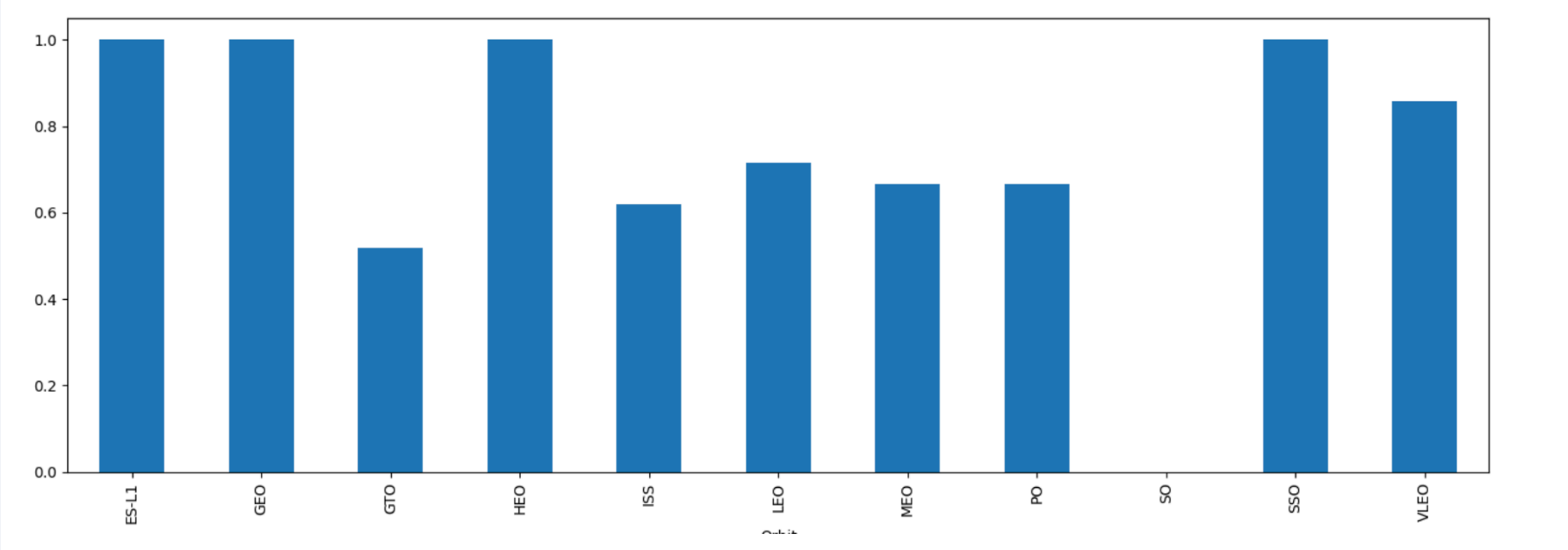
- Outcome improves with increasing number of flights

Payload vs. Launch Site



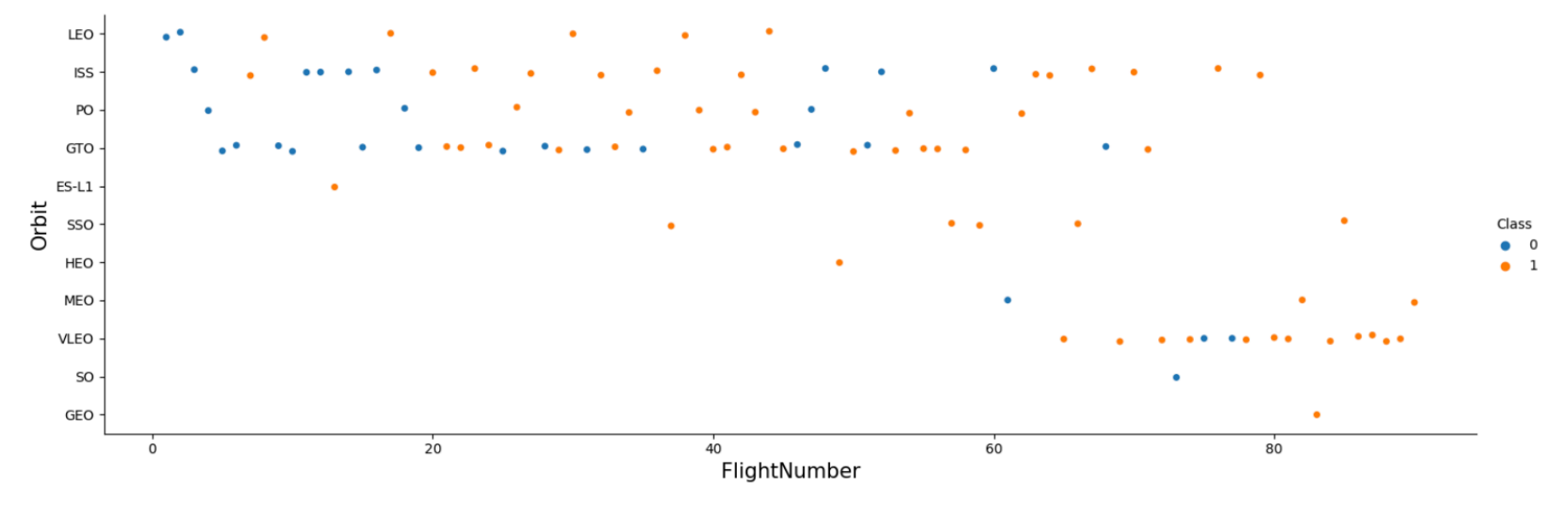
- For the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type



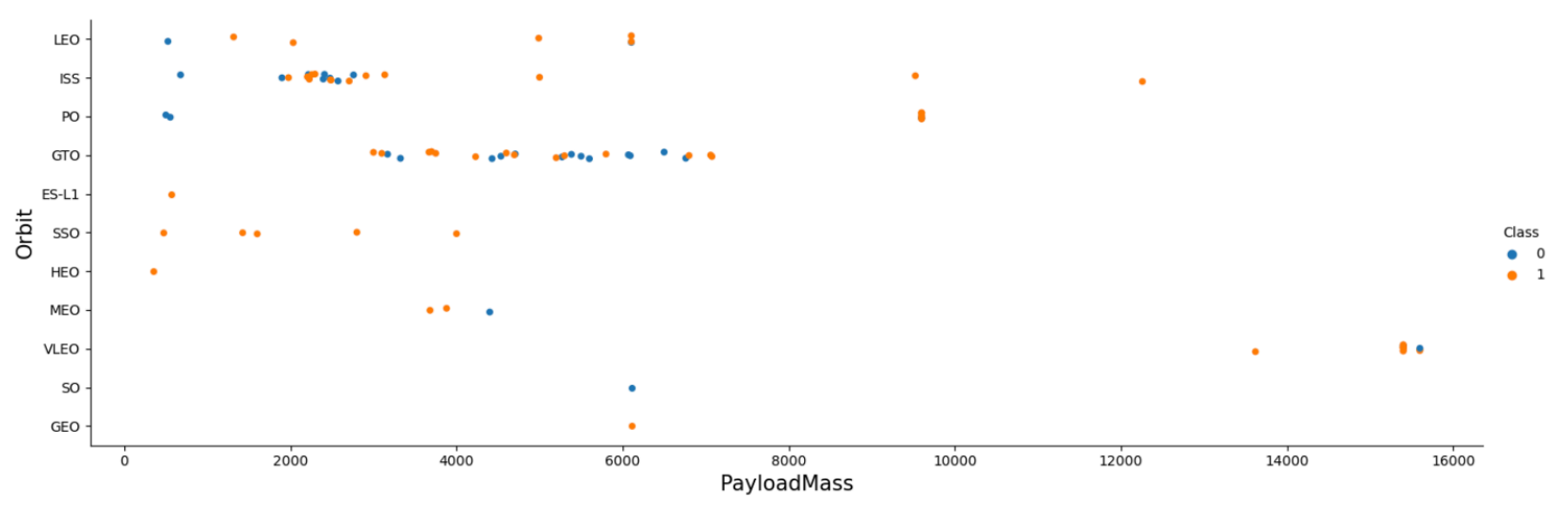
- ES-L1,GEO, HEO and SSO have highest success rates

Flight Number vs. Orbit Type



- The success rate increases for LEO. No clear relationship for the other orbits

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.

All Launch Site Names

- Grab Launch_Site from table
- Distinct command selects unique values only

```
%sql  select distinct Launch_Site from SPACEXTBL ;
```

```
 * sqlite:///my_data1.db
Done.
```

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

Launch Site Names Begin with 'CCA'

- Like command does partial string matching
- % serves as wildcard
- Limit 5 selects up to 5 records only

```
%sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5;
```

```
 * sqlite:///my_data1.db
Done.
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Sum to aggregate
- Where for filtering

```
%sql select sum(PAYLOAD_MASS__KG_) as total_payload from SPACEXTBL where Customer = 'NASA (CRS)';
```

```
 * sqlite:///my_data1.db
Done.
```

total_payload
45596

Average Payload Mass by F9 v1.1

- Avg to calculate mean payload

```
%sql select avg(PAYLOAD_MASS__KG_) as avg_payload from SPACEXTBL where Booster_Version = 'F9 v1.1';
```

```
 * sqlite:///my_data1.db
Done.
```

avg_payload
2928.4

First Successful Ground Landing Date

- Min function to select first date
- Where for filtering

```
%sql select min(Date) as first_success_date from SPACEXTBL where "Landing _Outcome" = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

first_success_date
01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- Where for filtering
- And to add multiple filter constraints

```
%sql select Booster_Version from SPACEXTBL where "Landing _Outcome" = 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS
```

```
 * sqlite:///my_data1.db
Done.
```

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1029.1
F9 FT B1021.2
F9 FT B1036.1
F9 B4 B1041.1
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- Use partial string match to aggregate into Success and Failure Categories

```
%%sql select (select count(Mission_Outcome) from SPACEXTBL
              where Mission_Outcome like '%Success%') as Success,
              (select count(Mission_Outcome) from SPACEXTBL
              where Mission_Outcome like '%Failure%') as Failure;
```

```
 * sqlite:///my_data1.db
Done.
```

Success	Failure
100	1

Boosters Carried Maximum Payload

- Subquery to select maximum payload and use for filtering

```
%%sql select Booster_Version from SPACEXTBL
where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL);
```

```
 * sqlite:///my_data1.db
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

2015 Launch Records

- substr to extract month and year

```
%%sql select substr(Date, 4, 2) AS month, Booster_Version, Launch_Site from SPACEXTBL
where "Landing _Outcome" = 'Failure (drone ship)' and substr(Date,7,4) = '2015';
```

```
 * sqlite:///my_data1.db
Done.
```

month	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

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

- 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

```
%%sql select "Landing _Outcome", count("Landing _Outcome") from SPACEXTBL
where Date >= '04-06-2010' and Date <= '20-03-2017' and "Landing _Outcome" like '%Success%'
group by "Landing _Outcome"
order by count("Landing _Outcome") desc ;
```

```
 * sqlite:///my_data1.db
Done.
```

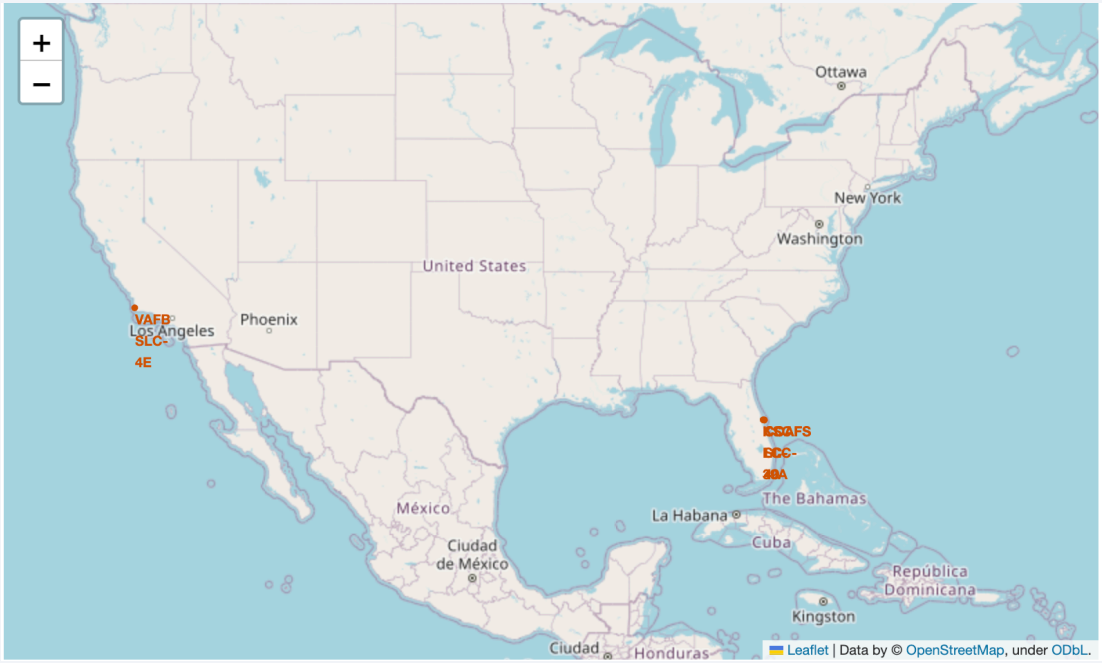
Landing _Outcome	count("Landing _Outcome")
Success	20
Success (drone ship)	8
Success (ground pad)	6

Section 3

Launch Sites Proximities Analysis

Launch Site Map

- Folium map showing locations go all launch sites in the dataset

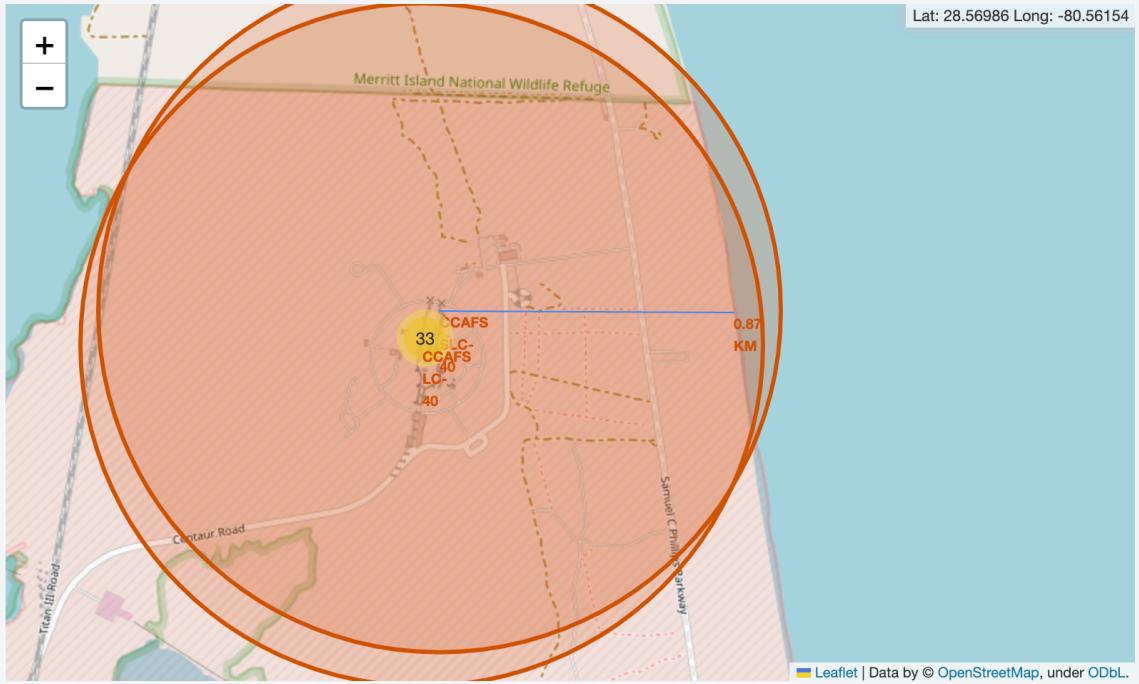


Success/Failed Launches Folium Map

- Map showing a marker for each site on the map. The number of launches is aggregated (ex. 26 bottom circle). A detailed view of each launch available upon clicking
  - Red = Failed Launch
  - Green = Successful Launch

Proximity Map

- Map showing a line with a calculated distance from launch site to coastline

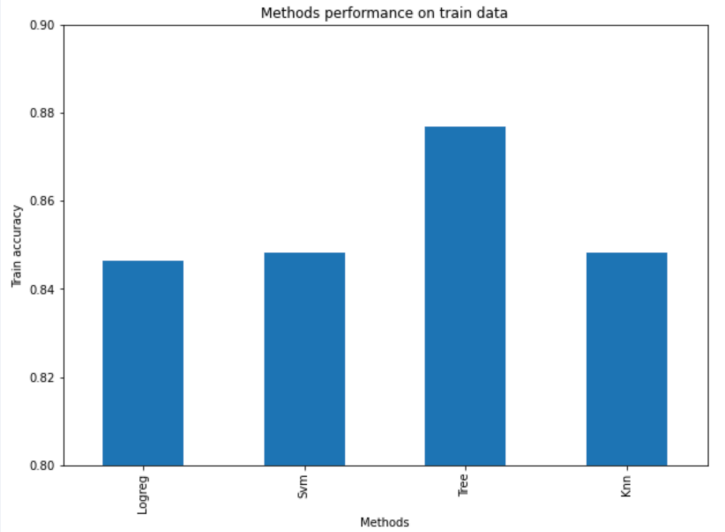


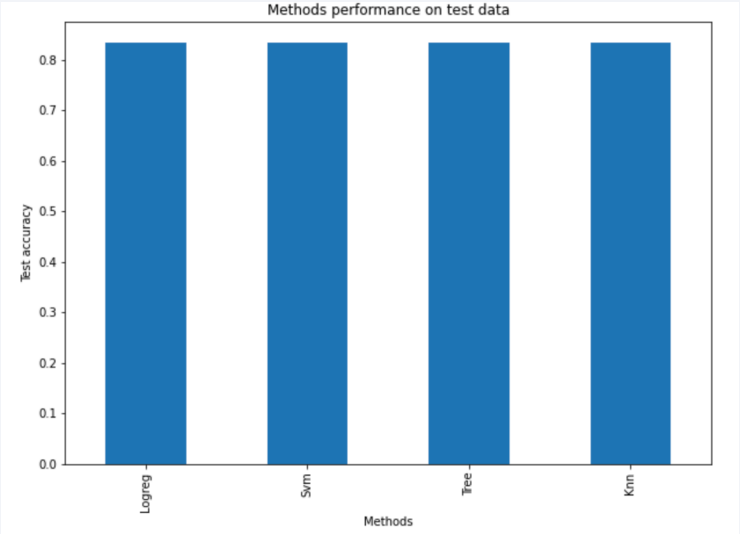
Section 5

Predictive Analysis (Classification)

Classification Accuracy

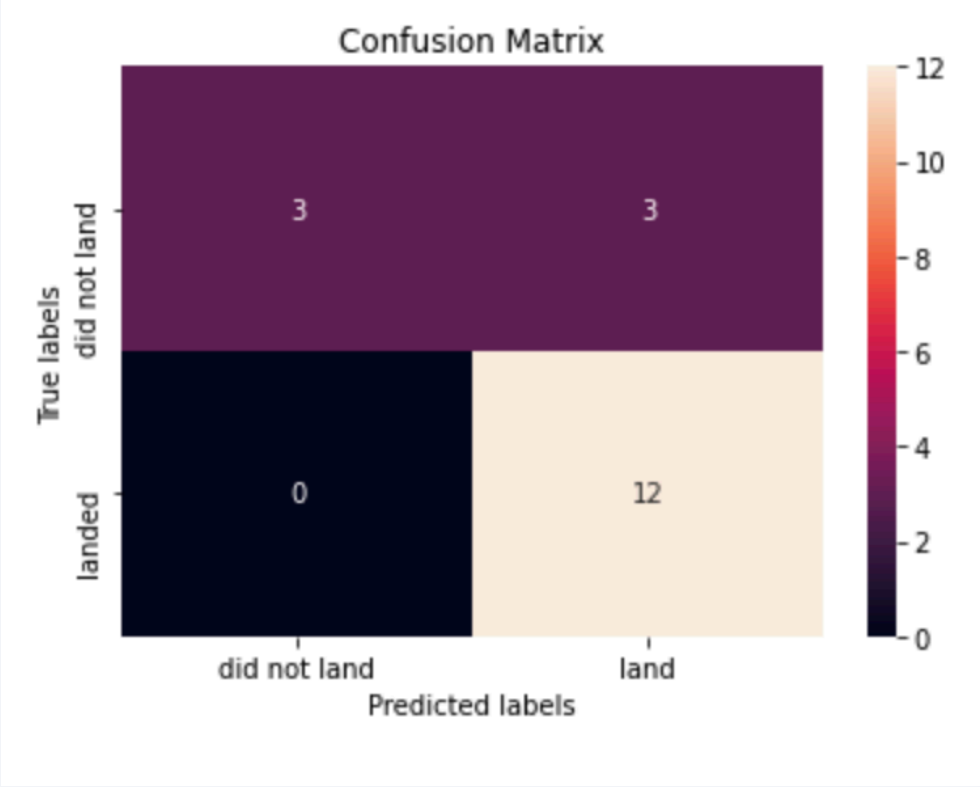
- The decision tree model has the highest accuracy in training sample
- However, accuracy on the test set is equal for all models evaluated





Confusion Matrix

- All models show the same confusion matrix as their accuracy is the same.
- TP = 12 (Predicted and reality successful)
- TN = 3
- FP = 0
- FN = 3 (Predicted as unsuccessful landing but in reality it landed successfully)



Conclusions

- Whether a launch is successful or not can be predicted using several factors like orbit, payload, launch site
- Accuracy is estimated to be around 83%
- Low weighted payloads perform better than heavy weighted
- Out of all the models explored, Decision Tree had the best results on the training dataset
- However, all models performed equally on the test dataset.

Thank you!