Winning Space Race with Data Science

MM-de
03/2023

Outline

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

Executive Summary

- This capstone project will analyse, if the SpaceX Falcon 9 first stage will land successfully using data science methods and classification algorithms.
- The analysis is divided into the following main steps:
  - Data collection, wrangling, and formatting
  - Exploratory data analysis
  - Interactive data visualization
  - Machine learning prediction
- The analysis shows, that correlations between certain features of the rocket launches and the outcome of the rocket launches are visible.
- The most suitable machine learning algorithm to to predict if the Falcon 9 first stage landing is the decision tree model.

Introduction

- This capstone project has the target to predict, if the Falcon 9 first stage will land successfully and which criteria's have an influence on the successful landing.
- The customer is a competitor of SpaceX, who advertises that a Falcon 9 rocket launches with a cost of 62 million dollars. Other providers cost up to 165 million dollars per start. The price saving of SpaceX result of the reuse of the first rocket stage.
- Most unsuccessful landings are planned. Sometimes, SpaceX will perform a controlled landing in the ocean.
- The main question that we are trying to answer is, for a given set of features about a Falcon 9 rocket launch which include among others the payload mass, orbit type, launch site, will the first stage of the rocket land successfully.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology and data wrangling:
    - Data collection, wrangling, formatting with SpaceX API and Web scraping
- Perform exploratory data analysis (EDA) using visualization and SQL
    - Visualization using scatter plots / bar graphs and analysis using SQL queries
- Perform interactive visual analytics using Folium and Plotly Dash
    - Visualization of payload and successful launches
- Perform predictive analysis using classification models
    - Machine learning with logistic regression, SVM, decision trees and KNN

Data Collection

- Data collection from SpaceX REST API about rocket launches
  https://api.spacexdata.com/v4
  Data collected:
    - FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, landingPad, Block, ReusedCount, Serial, Longitude, Latitude
- Data collection from Wikipedia via Web Scraping
  https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- Data collected:
    - Flight No,, Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection from SpaceX API → Decoding and tranformation into pandas dataframe → Request required data → Store data in dictionary and transformation to dataframe → Filtering for Falcon 9 launches → Replacing missing values of PayloadMass with mean → Exporting data to .csv

Data Collection – SpaceX API

Data collection based on SpaceX API to get data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications and landing outcome.

1: - Make a GET response to the SpaceX REST API
   - Convert the response to a .json file then to a Pandas DataFrame

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
data = pd.json_normalize(response.json())
```

2: - Data cleaning
   - Create and define lists for the data storage
   - Extract data and write data to predefined lists
   - Setup a dictionary and create a dataset

```
#Global variables
BoosterVersion = []
PayloadMass = []
Orbit = []
LaunchSite = []
Outcome = []
Flights = []
GridFins = []
Reused = []
Legs = []
LandingPad = []
Block = []
ReusedCount = []
Serial = []
Longitude = []
Latitude = []
```

```
# Call getBoosterVersion
getBoosterVersion(data)

# Call getLaunchSite
getLaunchSite(data)

# Call getPayloadData
getPayloadData(data)

# Call getCoreData
getCoreData(data)
```

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

3: - Transform dataset to a Panday Dataframe for further analysis

```
# Create a data from launch_dict
df = pd.DataFrame.from_dict(launch_dict)
```

4: - Filter the DataFrame for Falcon 9 launches
   - Reset FlightNumber column in dataset
   - Replace the NaN values of PayloadMass with the mean values

```
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = df[df["BoosterVersion"] != "Falcon 1"]
data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
data_falcon9
```

5: - Result: We end up with 90 rows and 17 columns. Extract of the data:

```
# Calculate the mean value of PayloadMass column
data_falcon9 = data_falcon9.fillna(value={"PayloadMass" : data_falcon9["PayloadMass"].mean()})
```

Link - Github

Data Collection - Scraping

1: - Get response from HTML and create Soup object

```
# assign the response to a object
response = requests.get(static_url)
```

```
soup = BeautifulSoup(data)
```

2: - Find tables

```
html_tables = soup.find_all("table")
```

3: - Extract column names

```
column_names = []

# Apply find_all() function with `th` element o
# Iterate each th element and apply the provide
# Append the Non-empty column name (`if name is

for row in first_launch_table.find_all("th"):
    name = extract_column_from_header(row)
    if(name != None and len(name) > 0):
        column_names.append(name)
```

4: - Create launch data dictionary

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each value to be an empty lis
```

5: - Extract table data to dictionary

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
   # get table row
    for rows in table.find_all("tr"):
```

6: - Create dataframe from dictionary

```
df=pd.DataFrame(launch_dict)
```

7: - Export to .csv

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

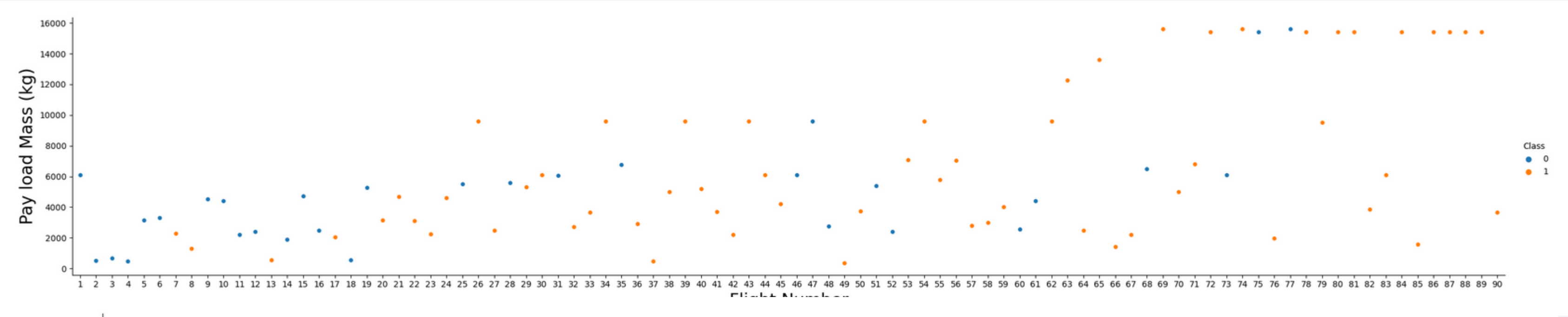
Data collection about Falcon 9 launches from Wikipedia → Creating BeautifulSoup object → Extracting column headers from HTML table → Collecting required data → Create dictionary with required data → Transformation to dataframe → Exporting to .csv

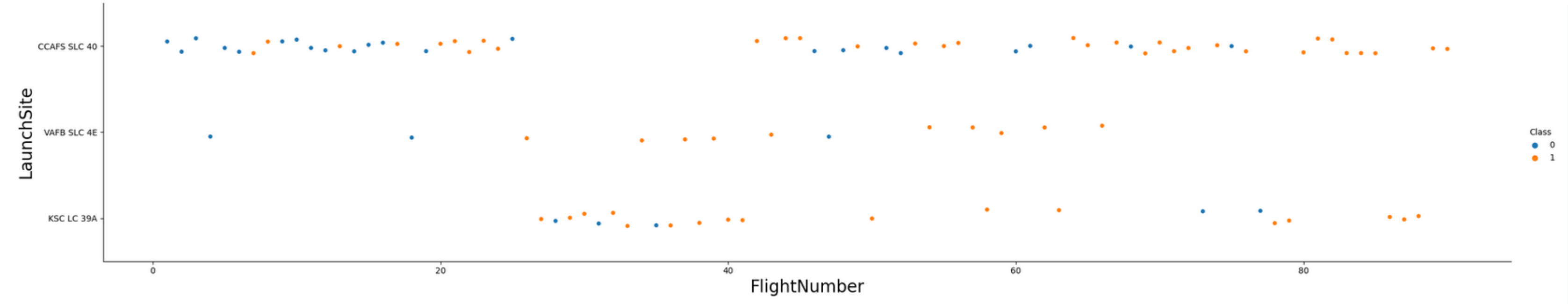
Data Wrangling

- Analysis was performed on the data.
- Success rates for different launch sites, orbits, payloads and throughout the years.
- Finally, a result column was created that held the information about the success and failure of the launch.

EDA with Data Visualization

- Exploration was performed using bar plots, scatter plots for understanding the relations between pairs of features.
- The pairs of features were Launch Site and Payload, success rate and Orbit Types, Orbit Types and Flight Numbers etc. (Examples shown below)





EDA with SQL

- Exploration on the data was also done using SQL. Following queries were performed:
- The names of the unique launch sites.
- 5 launch sites that 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 of the first successful landing outcome in ground pad.
- Successful boosters in drone ship that have payload mass between 4000 and 6000.
- Total number of successful and failure mission outcomes.
- Names of the booster versions which have carried the maximum payload mass.
- Failure for drone ship ,booster versions, launch site and months for the months in year 2015.
- Rank the count of successful landing outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

```
%sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

```
%sql SELECT UNIQUE(LAUNCH_SITE) FROM SPACEXTBL;
```

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL \
    WHERE CUSTOMER = 'NASA (CRS)';
```

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

TOTAL_PAYLOAD_MASS
45596

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS FROM SPACEXTBL \
    WHERE BOOSTER_VERSION = 'F9 v1.1';
```

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

AVERAGE_PAYLOAD_MASS
2928.4

Build an Interactive Map with Folium

- Folium maps with Circles, Markers, MarkerClusters, MousePositions were generated.
- Circles are used to highlight areas surrounding the launch sites like Cape Canaveral Space Launch Complex 40 (CCAFS LC-40).
- Markers are used to mark the co-ordinates of the launch sites.
- Mouse Positions were used to calculate the co-ordinates of the location the mouse is pointing to on the map.
- Lines were used to display the distance between the launch sites and other locations such as railways, coastlines and cities etc.

Build a Dashboard with Plotly Dash

- Graphs were displayed on an Interactive UI to visualize the data.
- Pie chart to display the success and failure rate of a selected Launch Site.
- Payload range slider to select the launches in the specified range of payloads to analyze.

- The dashboard allows effortless analysis of the relation between payload ranges, launch sites and their success and failure rates.

Predictive Analysis (Classification)

- Built ML models to train on the data for prediction of launch success or failure.
- Decision Tree
- Logistic Regression.
- Support Vector Machines
- K Nearest Neighbors.

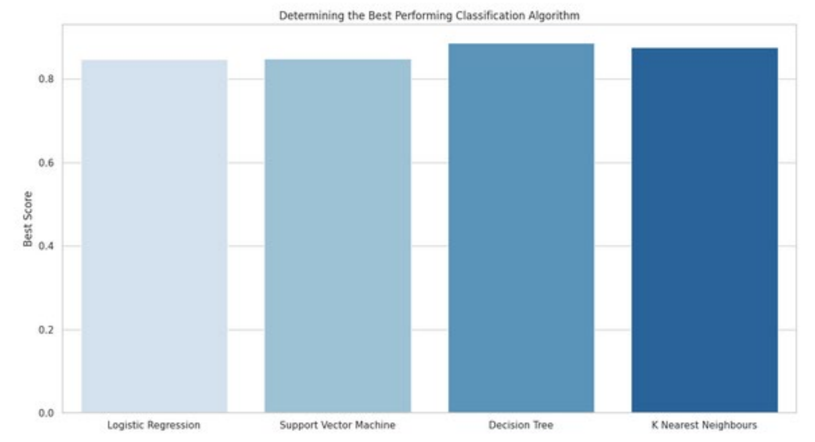
- The data was standardized and split into training and testing sets.

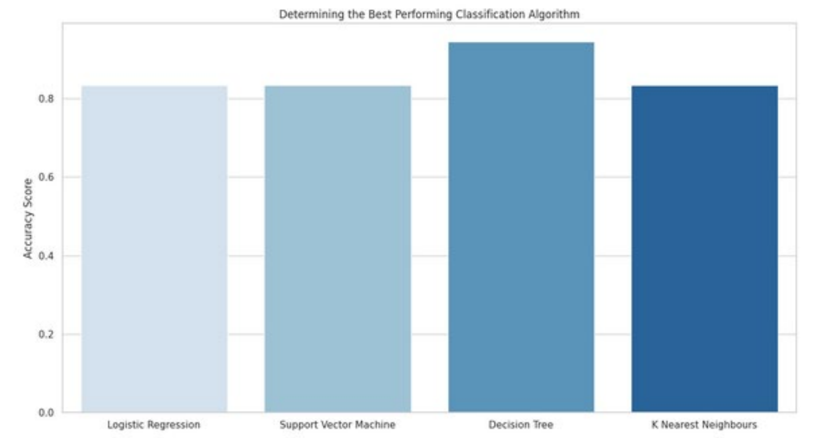
- Hyper-parameter optimization was done on the models to find the best parameters for the models.

- The accuracy scores of the models were compared to select the best one.

Results

- SpaceX uses 4 different launch sites. CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40.
- The average payload of F9 v1.1 booster is 2,928.4 kg.
- The total payload of NASA Boosters is 45,596 kg.
- The first successful ground pad landing was done on 1st May 2017.
- Only 1 in-flight launch resulted in failure. Rest all were a success.
- Only 2 drone ship failures were reported in the F9 v1.1 B1012 and F9 v1.1 B1015 boosters.
- The success rate for the launches have increased over the years after the year 2013.
- Interactive folium maps showed that most of the successful launches were near the coastlines away from cities in safety locations.
- These locations also have sophisticated infrastructure such as railways.
- The predictive analysis revealed that the Decision Tree is the best model for the predictions as it had the highest accuracy score.





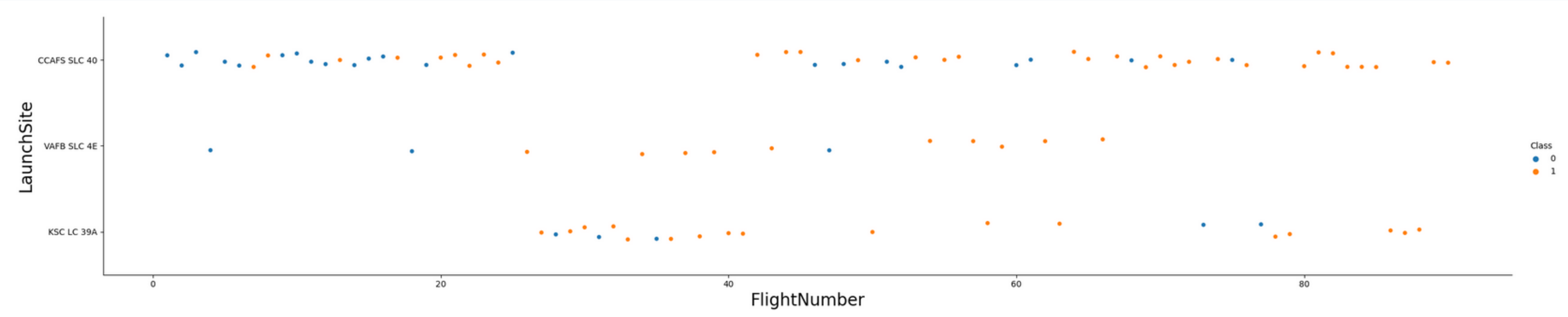
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

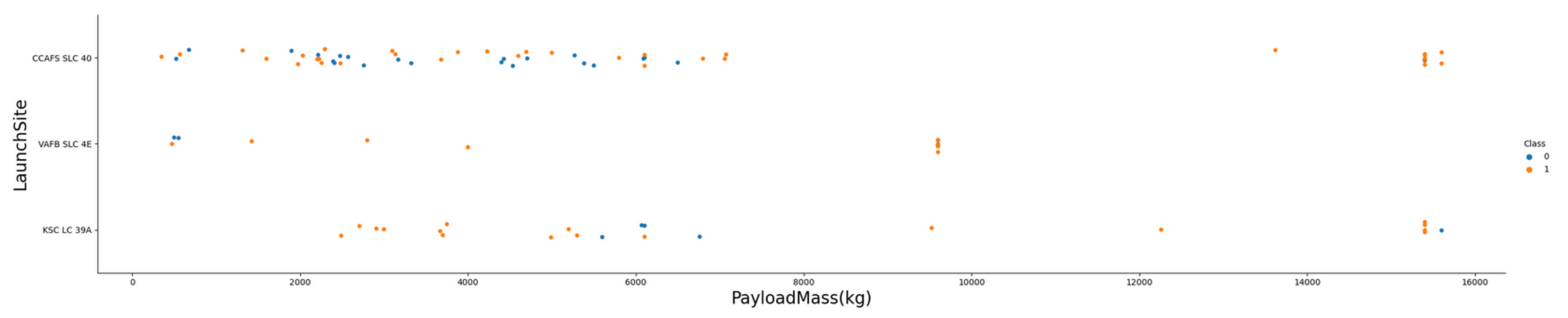
- The CCAFS SLC 40 has the highest success rate.

- The success rates improved with more launches performed.



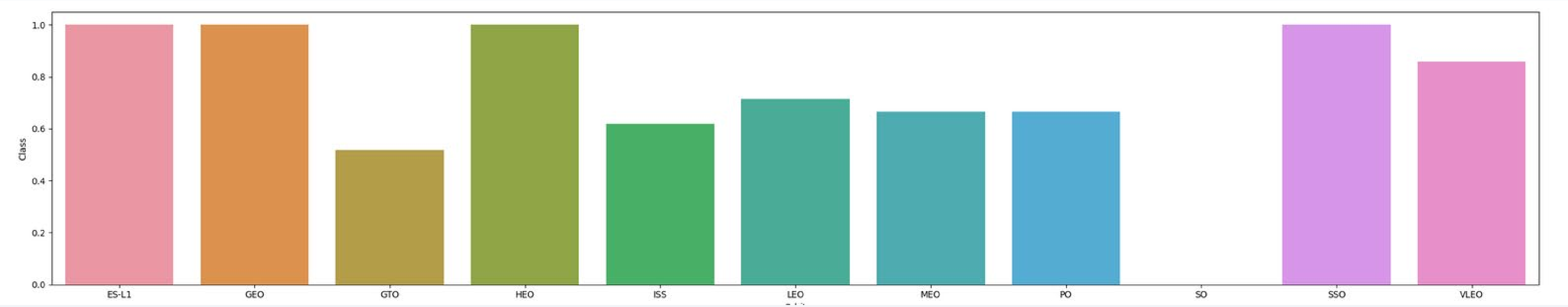
Payload vs. Launch Site

- Higher Payload launches have a higher success rate.
- Most Launches have Payload Masses below 6000 kg



Success Rate vs. Orbit Type

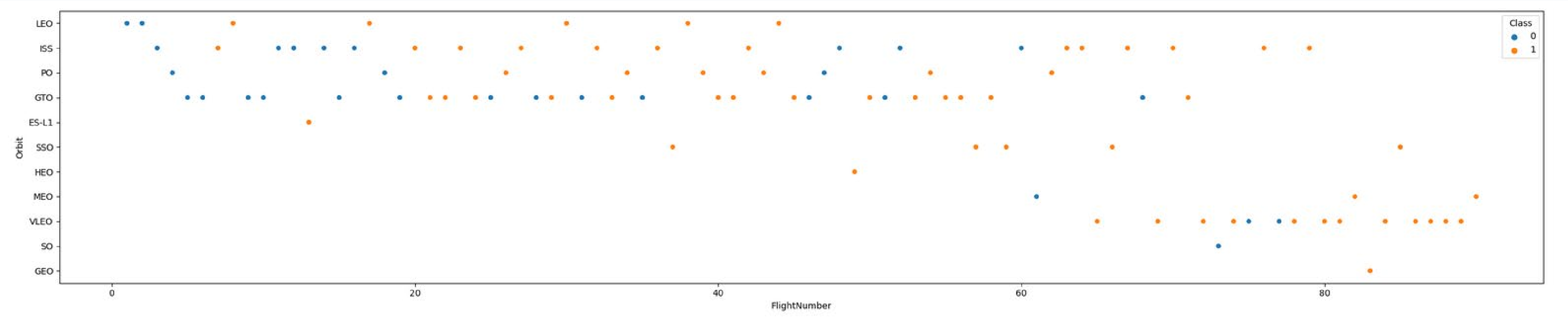
- ES-L1, GEO, HEO and SSO have the highest success rate for launches
- SO orbit has 0 success rate.



Flight Number vs. Orbit Type

- An increased success rate increases success rate is observed for all the orbits.

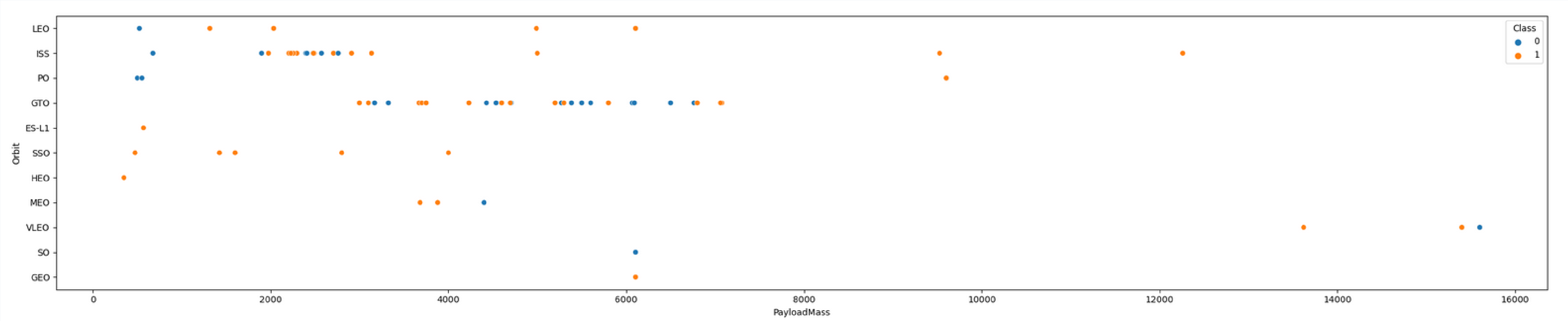
- SSO and VLEO orbits are used for later launches and have a promising success rate.



Payload vs. Orbit Type

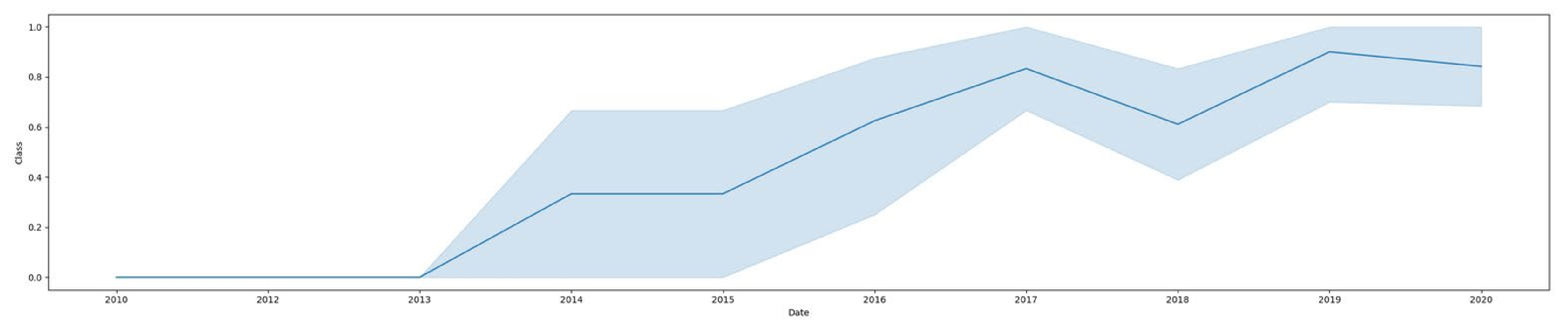
- With heavy payloads the success rate are more for PO, LEO and ISS orbits.

- For GTO the successful and unsuccessful launches do not show a significant pattern.



Launch Success Yearly Trend

- The success rate clearly has improved over the years after the year 2013.
- The success rate increased to approx. 80%.



All Launch Site Names

- The unique launch sites are.

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

```
[7]: sql select distinct Launch_Site from spacextbl
```

Launch Site Names Begin with 'CCA'

- The launch missions for which the launch site name begins with "CCA".

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

```
[14]: sql select * from spacextbl where Launch_Site like 'CCA%' limit 5
```

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

[14]:

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

- Total payload carried by boosters launched by NASA.

Display the total payload mass carried by boosters launched by NASA (CRS)

```
[17]: sql select sum(payload_mass__kg_) from spacextbl where customer = 'NASA (CRS)';

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

[17]:

sum(payload_mass__kg_)
45596

Average Payload Mass by F9 v1.1

- Average payload carried by boosters version F9 v1.1.

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

```
[18]: sql select avg(payload_mass__kg_) from spacextbl where Booster_Version = 'F9 v1.1';
```

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

[18]:

avg(payload_mass__kg_)
2928.4

First Successful Ground Landing Date

- Date of first successful launch for ground pad.
- 01.05.2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- Names of boosters successful in drone ship with payload mass between 4000 and 6000 kg.

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

Total Number of Successful and Failure Mission Outcomes

- Total number of successful and failed launches.

List the total number of successful and failure mission outcomes

[26]:
```
sql select Mission_Outcome, count(*) from spacextbl group by Mission_Outcome;
```

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

[26]:

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

Boosters Carried Maximum Payload

- Booster versions with highest payload mass.

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

- Failed landing outcomes in drone ship in year 2015.

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

- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

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

Section 3

Launch Sites Proximities Analysis

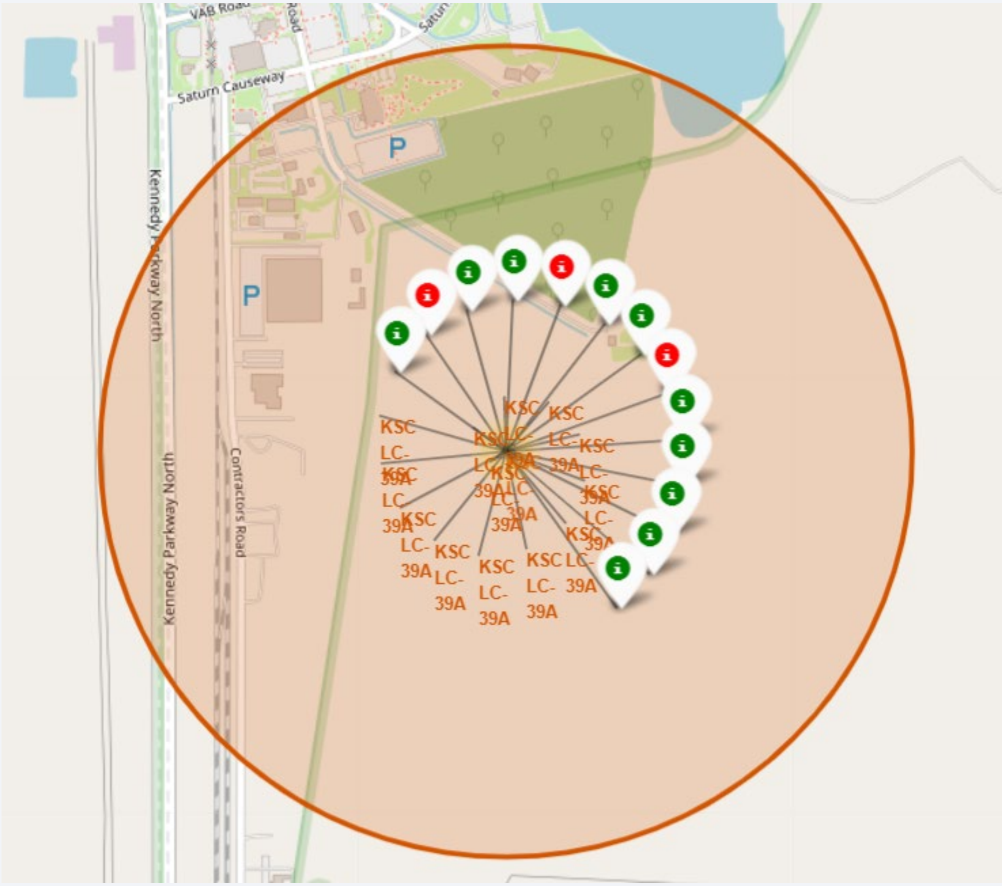
All Launch Sites on the Map

- All the launch sites are away from the populated areas (cities) near the coastlines but close enough to sophisticated infrastructure like railways.

Launch Outcomes for each Site

- This shows the KSC LC-39A launch site.

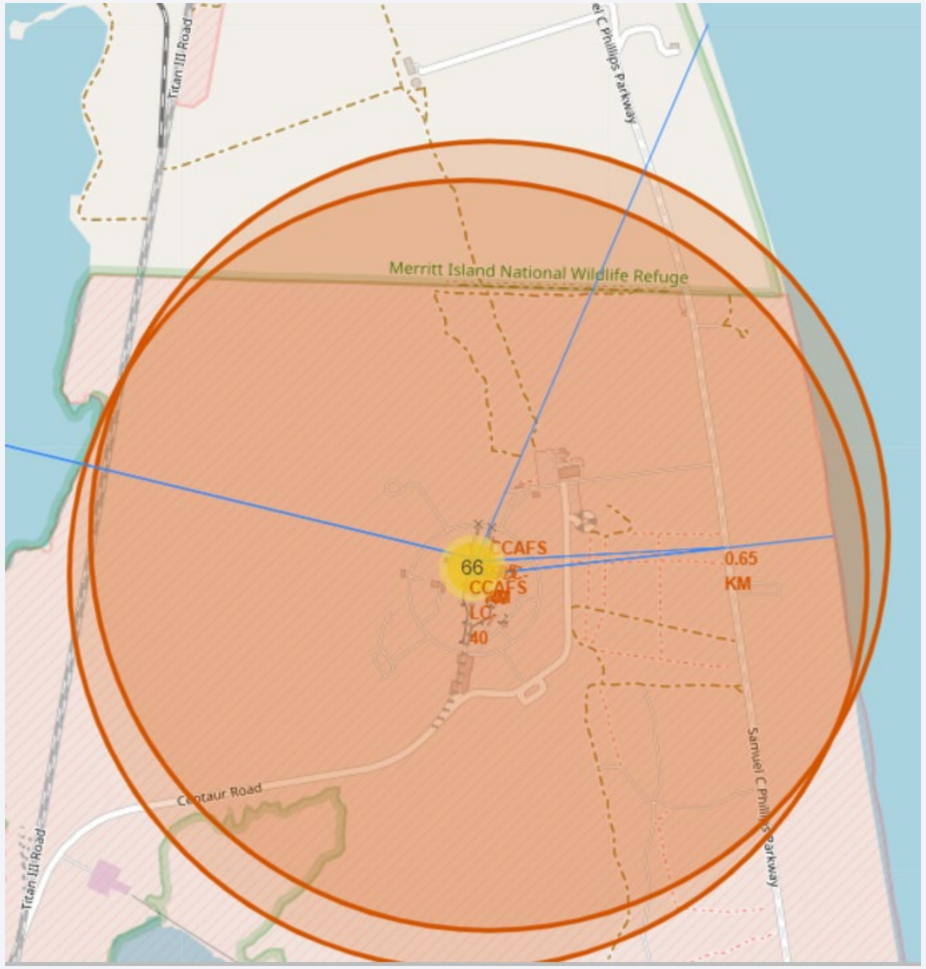
- The green and red markers denote the successful and failed launch missions respectively.



Infrastructure and Safety

- Launch sites have a sophisticated infrastructure as they have good railways and roads in the vicinity.

- The sites are also far away from the populated cities thus ensuring safety.

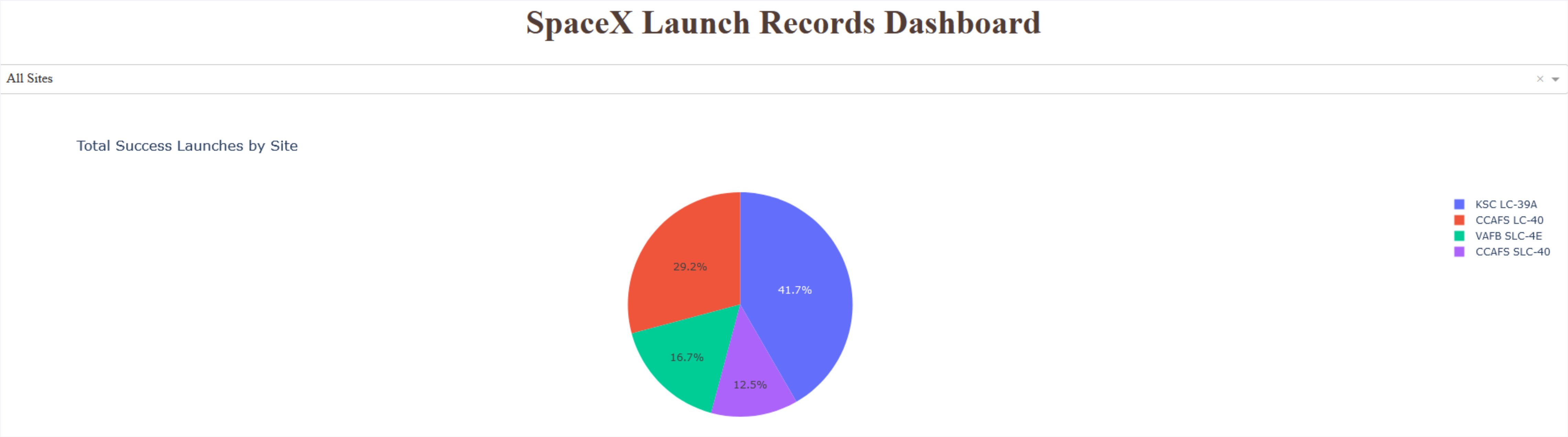


Section 4

Build a Dashboard with Plotly Dash

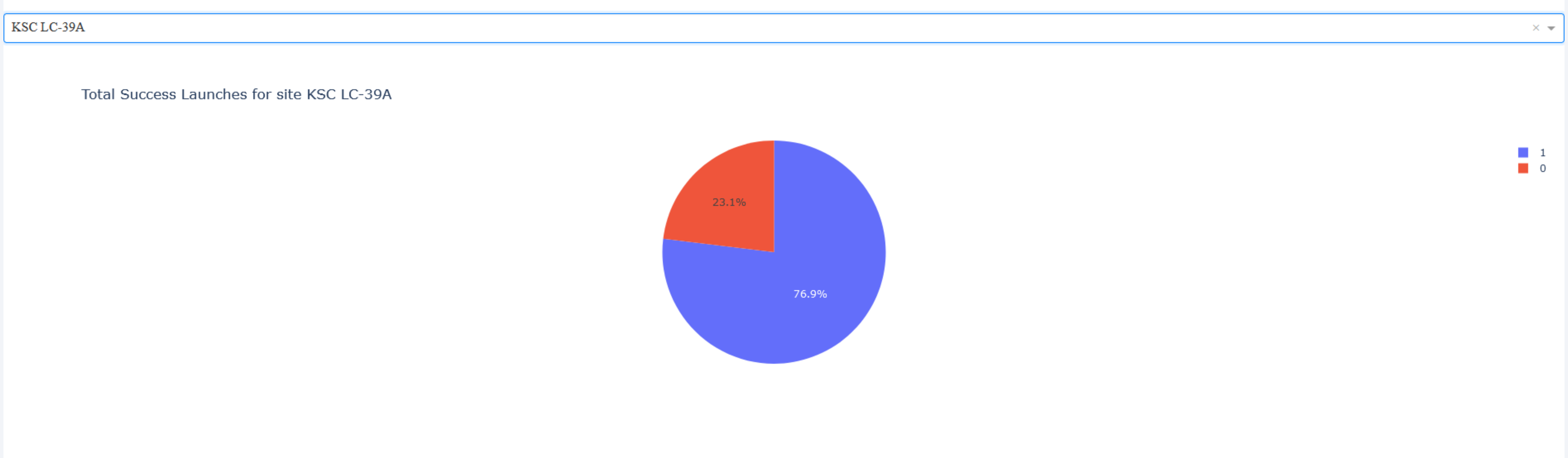
Successful Launches by Launch Site

- The launch sites is an important factor affecting the success of the launch mission.



Success Rrate for KSC LC 39A

- KSC LC 39A site reports 76,9 % success in all of its mission launches.



Payload and Launch Outcome Relation

- Payloads in the range of 3000 to 6000 kg for v1.1 booster result in failures for all launch missions.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

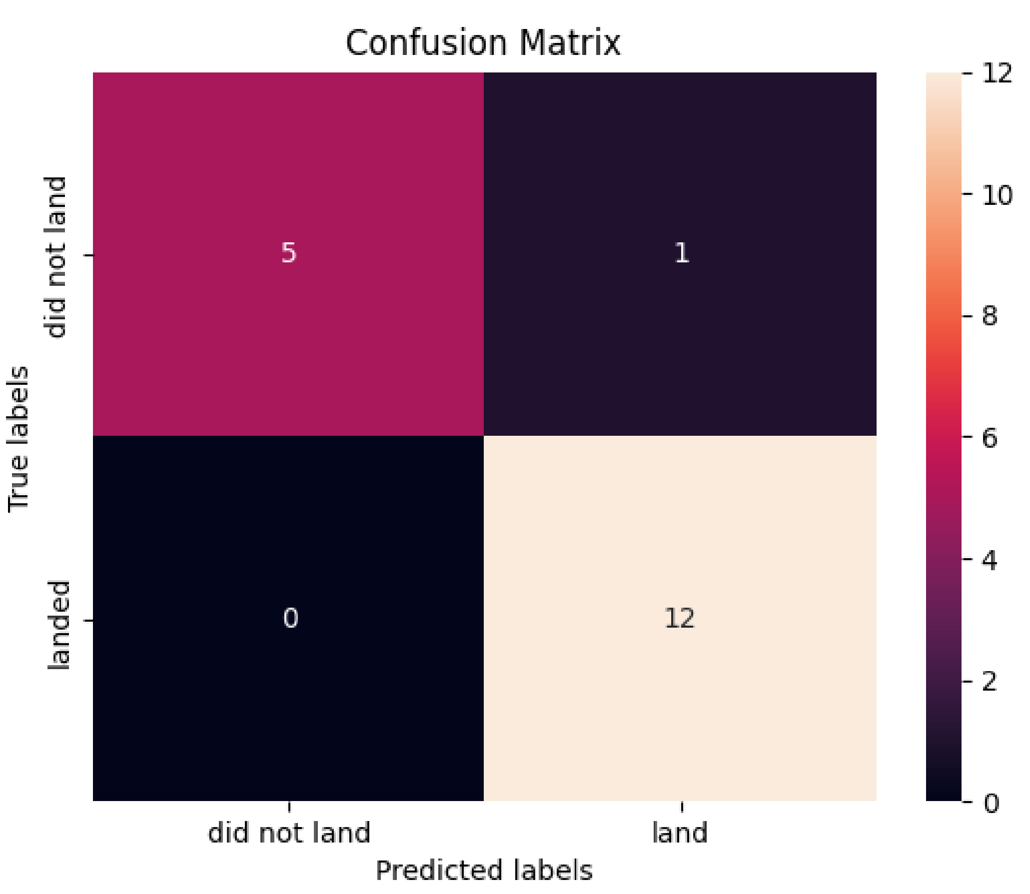
	Algorithm	Accuracy Score	Best Score
0	Logistic Regression	0.833333	0.846429
1	Support Vector Machine	0.833333	0.848214
2	Decision Tree	0.944444	0.885714
3	K Nearest Neighbours	0.833333	0.875000

- Four classification models were used.
- Logistic Regression
- Support Vector Machines
- Decision Trees.
- K Nearest Neighbors

- The validation accuracy of the Decision Tree model was the highest with 94%

Confusion Matrix

- The confusion matrix of the Decision Tree shows that the model performs more accurate in classification, than the other models.
- The model correctly predicted 5 of 6 unsuccessful launches and 12 of 12 successful launches correct.



Conclusions

- Most of the launch sites are located close to the coastline to prevent damage of critical infrastructure.
- The launch site KSC LC-39A is the launch site with the highest success rate for launches.
- The probability of successful launches has been improved since 2013.
- The "Decision Tree Classifier Model" was the best model in predicting the success and failure of the launches.

Appendix

- All the notebooks have been uploaded on GitHub – Results
- Link

Thank you!