IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Christos Koronios

August 14, 2022

Outline

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

Executive Summary

Summary of methodologies

- Data Collection via API & Web Scraping
- Data Wrangling
- Exploratory Data Analysis with SQL
- Exploratory Data Analysis with Data Visualization
- Interactive Visual Analytics with Folium
- Interactive dashboard (Plotly Dash)
- Machine Learning Predictive analysis (Classification)

Summary of all results

- Exploratory Data Analysis outcome
- Interactive analysis
- Predictive analysis

Introduction

Project background and context

Space X 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 Space X 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 space X for a rocket launch. This goal of the project is to create a machine learning pipeline to predict if the first stage will land successfully.

Problems you want to find answers

- Which factors determine the success of the first stage's rocket launch?
- The interconnection among the various parameters of the rocket launch
- What operating conditions needs to be in place to ensure a successful landing program.

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
  - SpaceX Rest API
  - Web Scraping from Wikipedia
- Perform data wrangling
  - One-hot encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Classification models LR, KNN, SVM and DT were built and evaluated

Data Collection

- Data collection through ***get*** request to the SpaceX API.
- Data collection through Web scraping and ***BeautifulSoup*** from Wikipedia
- The response date was stored as a **Json**, **normalized** and restored as pandas' **dataframe**
- Data **wrangling technics** such as replacing null values with their respective means

Data Collection – SpaceX API

```
#df.replace(to_replace = np.nan, value =-99999)
data_falcon9.isnull().sum()
```

```
FlightNumber       0
Date               0
BoosterVersion     0
PayloadMass        0
Orbit              0
LaunchSite         0
Outcome            0
Flights            0
GridFins           0
Reused             0
Legs               0
LandingPad        26
Block              0
ReusedCount        0
Serial             0
Longitude          0
Latitude           0
dtype: int64
```

```
# Use json_normalize meethod to convert the json result into a dataframe
response.json()
data = pd.json_normalize(response.json())
```

Get

Clean

Normlize

Show

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

```
# Calculate the mean value of PayloadMass column
data_falcon9.PayloadMass.mean()

# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(to_replace = np.nan, value = 6123.547647, inplace=True)

#df.replace(to_replace = np.nan, value =-99999)
data_falcon9.isnull().sum()
```

LINK

Data Collection - Scraping

```
# use requests.get() method with the provided static_url
# assign the response to a object
data  = requests.get(static_url).text
```

...

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, 'html.parser')
```

```
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"):
        #check to see if first table heading is as number corresponding to launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
        #if it is number save cells in a dictonary
        if flag:
            extracted_row += 1
            # Flight Number value
            # TODO: Append the flight_number into launch_dict with key `Flight No.`
            #print(flight_number)
            datatimelist=date_time(row[0])
```

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

Response from HTML

Create Dictionary

Convert to DF

BeautifulSoup

Append

DF to CSV

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

```
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 list
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

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

LINK

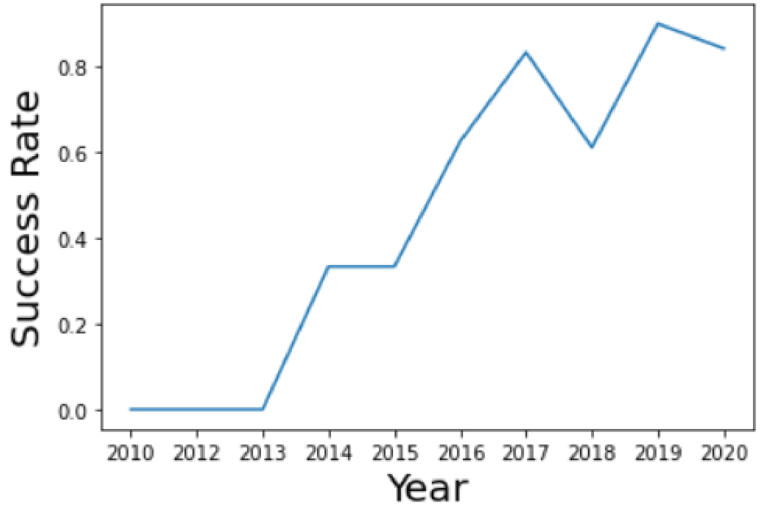
Data Wrangling

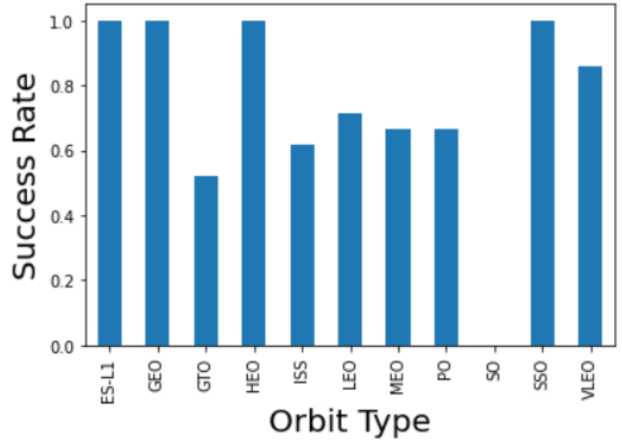
Data Wrangling Process

- Identify null values
- Identify how many lunches occurred from each launch site
- Identify the occurrence of each orbit
- Identify the occurrence of each orbit per orbit type
- Create landing outcome label from outcome column and export to csv

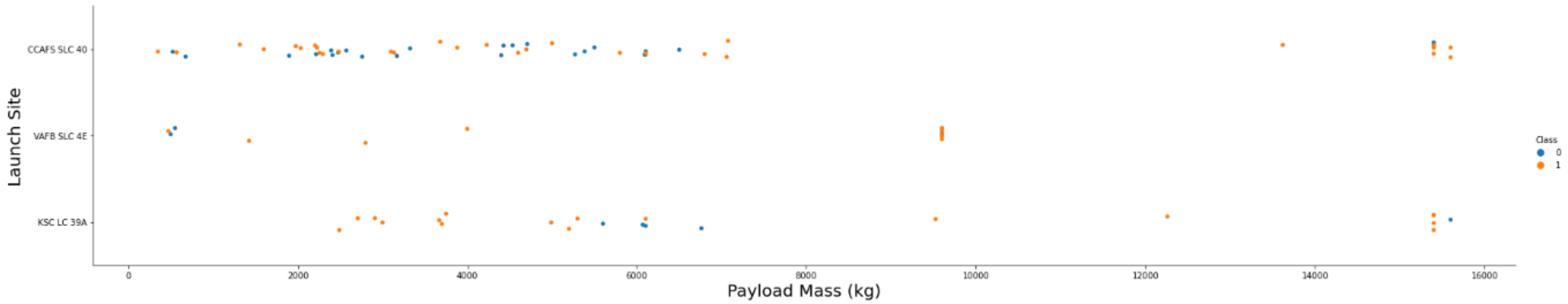
[LINK to the notebook]

EDA with Data Visualization





[LINK to workbook]



EDA with SQL

We gained further insight in the dataset via SQL queries such as

- ✓ The names of unique launch sites in the space mission.
- ✓ The total payload mass carried by boosters launched by NASA (CRS)
- ✓ The average payload mass carried by booster version F9 v1.1
- ✓ The total number of successful and failure mission outcomes
- ✓ The names of the boosters with success in ground pad given specific payload mass
- ✓ The failed landing outcomes in drone ship, their booster version and launch site names

LINK to the notebook

Build an Interactive Map with Folium

The followed process can be summarized into five steps

- Create objects on the map such as markers and circles to depict the launch sites
- Quantity 'Success Rate' using binary system, 0 for failure and 1 for success
- Mark them on the map
- Identify the key locations in the proximities (I.e. Railway and Coastline)
- Calculate their distance from them

Build a Dashboard with Plotly Dash

The Dashboard consists of

- A dropdown Filtering option for the Lauch Sites
- A Pie Chart for illustration
- A Scatter Chart for illustration
- A Rangeslider for the Payload mass a second Filtering mechanism

Link to the workbook

Predictive Analysis (Classification)

The followed process can be summarized into three main steps

- Build the ML model
  - Create a new column
  - Standardize the data
  - Split the data
  - Build the model and fit the training data
- Model Evaluation
  - Select the correct Evaluation method
  - Depict the Confusion matrices
- Select the correct ML model
  - Select the model with the highest accuracy

[LINK to the workbook]

Results

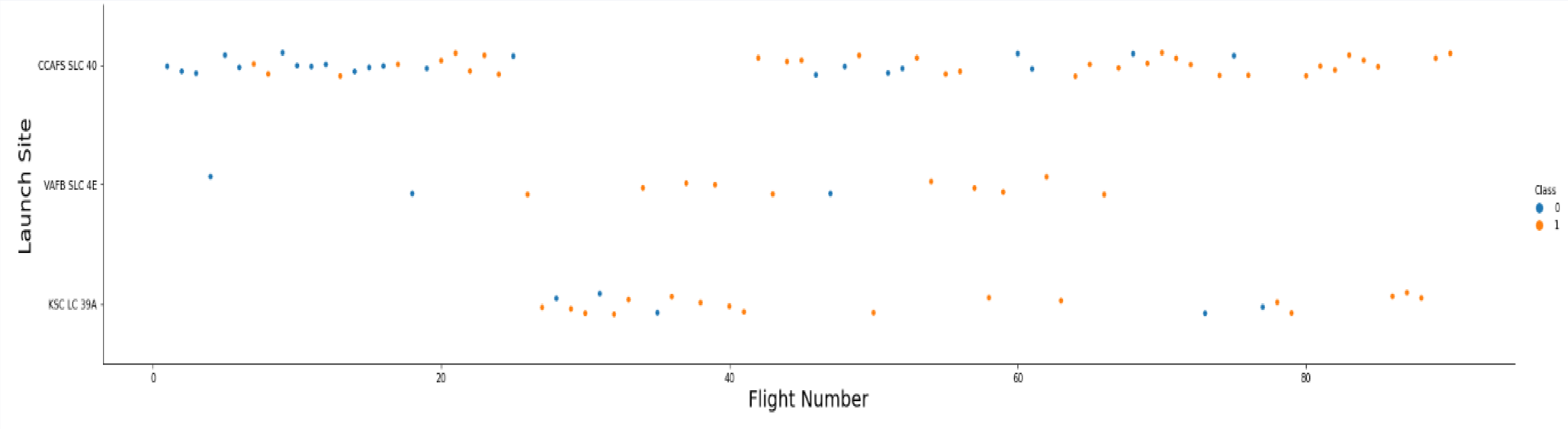
- Lower weight payloads are mostly connected to higher Success rates
- Machine Learning Classification models SVM, KNN and LR showed a surprisingly identical Evaluation figure
- KSC LC had the most successful launches
- Between 2013 and 2017 the Success Rate of all sites showed an upward trend
- The Launch Sites are particularly close to the coastline

Section 2

Insights drawn from EDA

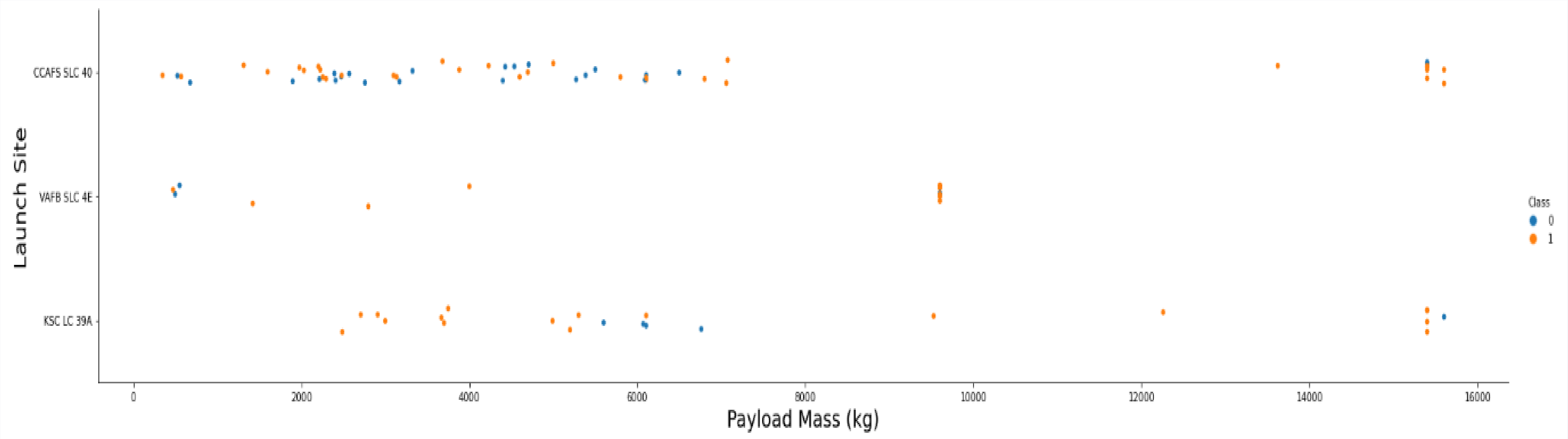
Flight Number vs. Launch Site

Launch Site *CCASF SLC 40* has the most launches



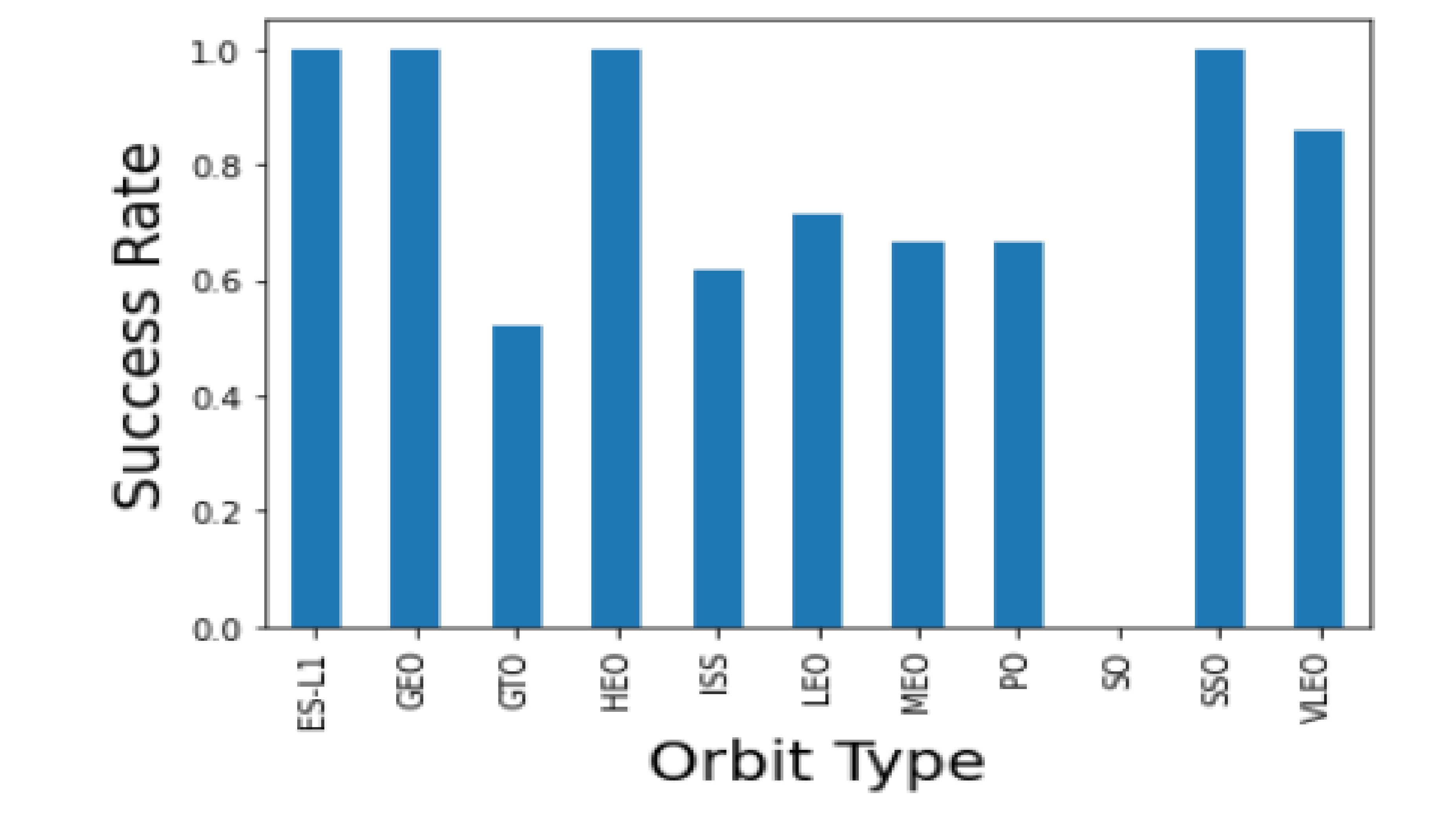
Payload vs. Launch Site

The scatterplot illustrates a Positive trend between the increase of the Payload Mass and the Success Rate (The higher the PM the higher the SR)



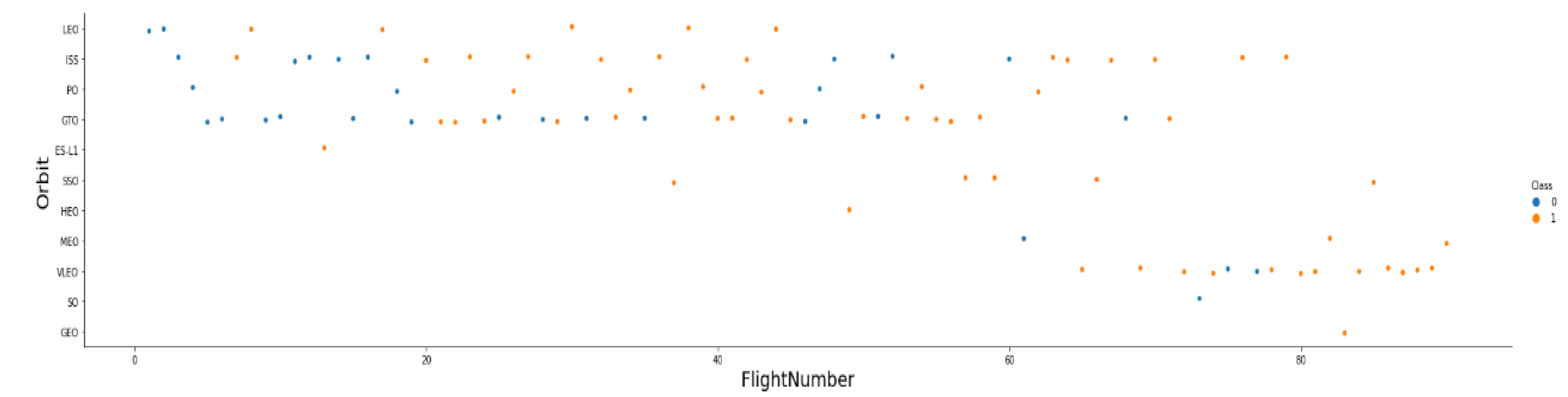
Success Rate vs. Orbit Type

The highest Success Rate occurs for the Orbit Types ES-L1, GEO, HEO and SSO



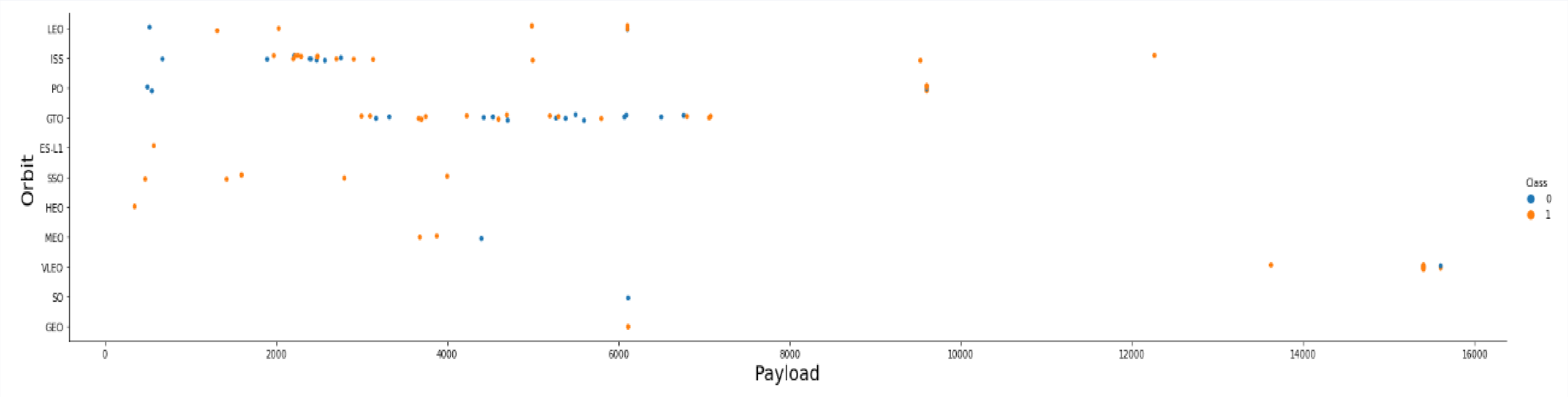
Flight Number vs. Orbit Type

The plow shows that there is a slightly positive relationship between the Flight Number and each Orbit Type although it is not a strong one



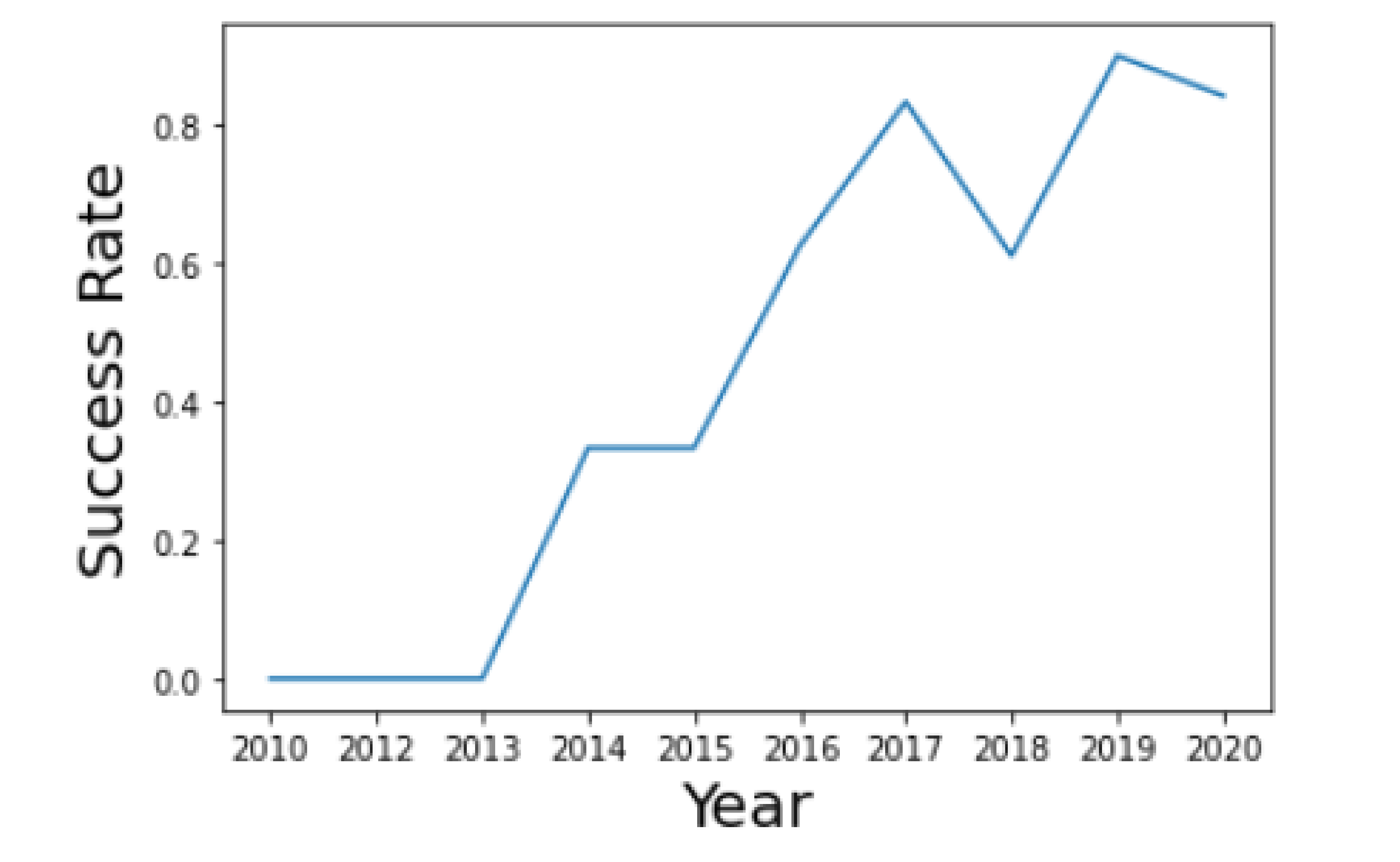
Payload vs. Orbit Type

Heavy Payloads mean higher Successful Rate for Polar, LEO and ISS



Launch Success Yearly Trend

There was a sharp upwards trend between 2013 and 2017



All Launch Site Names

We use the *'distinct'* statement to fetch one distinct values of the Launch Sites

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

```
]: %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'

We use the *'Limit'* statement to limit our search to five rows

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

```
%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

We use the *'sum'* function to get the sum of an attribute

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer = 'NASA (CRS)'
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

We use the *'AVG'* function to get the average payload

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

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.0%'
```

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

AVG(PAYLOAD_MASS__KG_)
340.4

First Successful Ground Landing Date

We use the 'MIN' function to receive the first successful landing date

The code is *%sql SELECT MIN(Date) FROM SPACEXTBL WHERE LandingOutcome = 'Success (ground pad)'*

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
%sql SELECT MIN(Date) FROM SPACEXTBL WHERE LandingOutcome = 'Success (ground pad)'
```

The outcome was December 22nd, 2015

	firstsuccessfull_landing_date
0	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Code

%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success (drone ship)' AND 4000 < PAYLOAD_MASS__KG_ < 6000

Outcome

	boosterversion
0	F9 FT B1022
1	F9 FT B1026
2	F9 FT B1021.2
3	F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

We use the *'COUNT'* function and we group the data with the '*GROUP BY*' clause

List the total number of successful and failure mission outcomes

```
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME
```

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

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

Boosters Carried Maximum Payload

We used the *'MAX'* function and a subquery

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
%sql SELECT DISTINCT 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

Code

```
%sql SELECT LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM
SPACEXTBL WHERE Landing__Outcome = 'Failure (drone ship)' AND YEAR(DATE) =
2015;
```

Outcome

	boosterversion	launchsite	landingoutcome
0	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
1	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

Code

```
%%sql
SELECT LANDINGOUTCOME, COUNT(LANDINGOUTCOME)
FROM SPACEXTBL
WHERE DATE BETWEEN '2010-06-04'  AND '2017-03-20'
GROUP BY LANDING__OUTCOME
ORDER BY TOTAL_NUMBER DESC
```

Outcome ---->

	landingoutcome	count
0	No attempt	10
1	Success (drone ship)	6
2	Failure (drone ship)	5
3	Success (ground pad)	5
4	Controlled (ocean)	3
5	Uncontrolled (ocean)	2
6	Precluded (drone ship)	1
7	Failure (parachute)	1

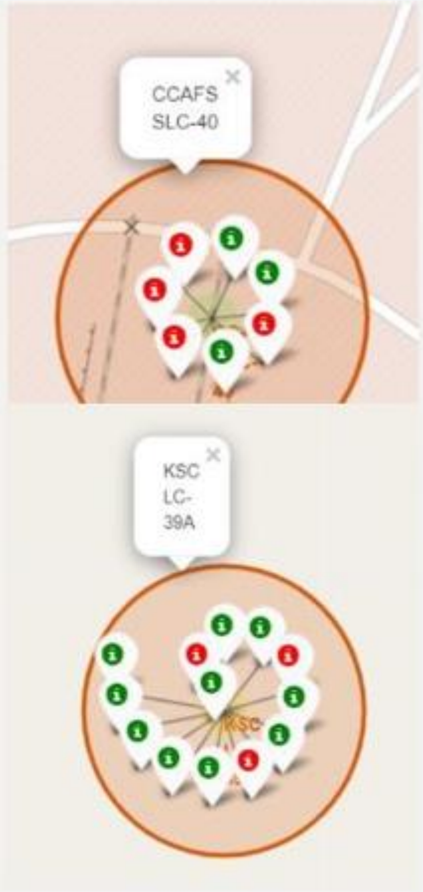
Section 3

Launch Sites Proximities Analysis

Launch Site Marked on the USA map

The Launch Sites are located in USA and specifically the coast lines near California and Florida

Successful Lauches per Launch Location

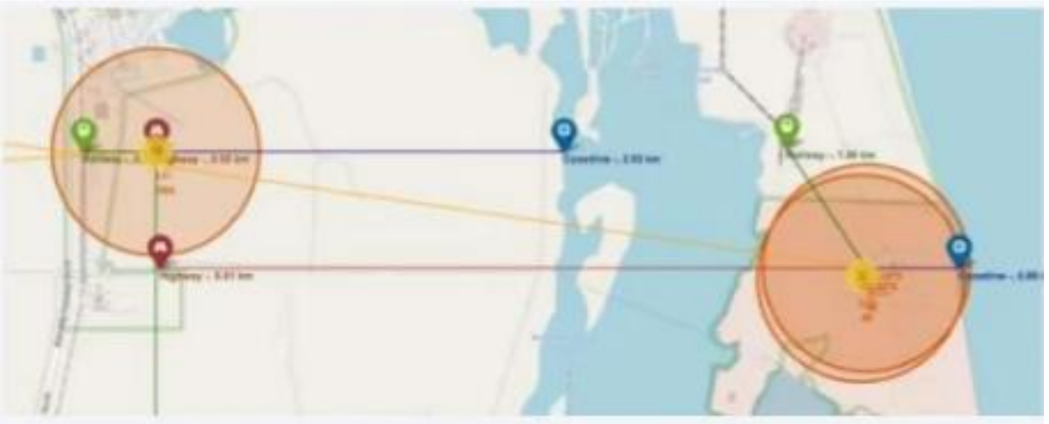


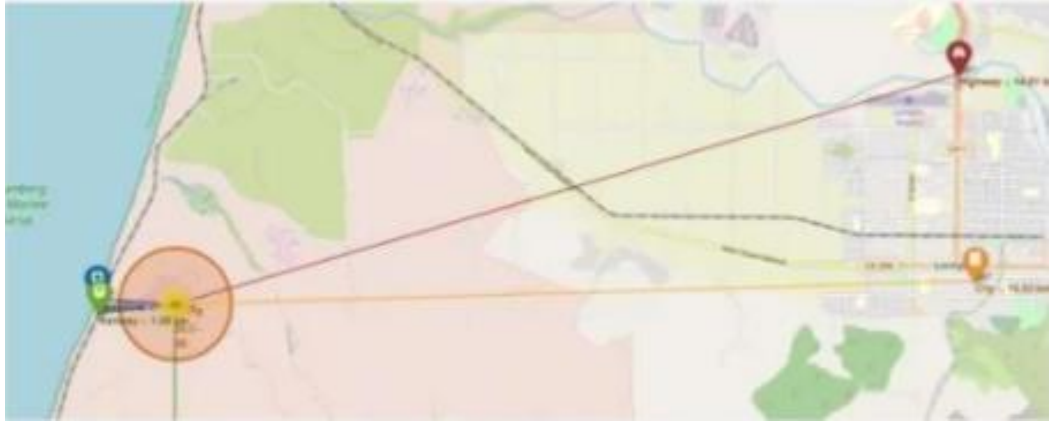
Florida Launch Sites

Green Marker shows successful Launches and Red Marker shows Failures

California Launch Site

Launch Site distance to landmarks

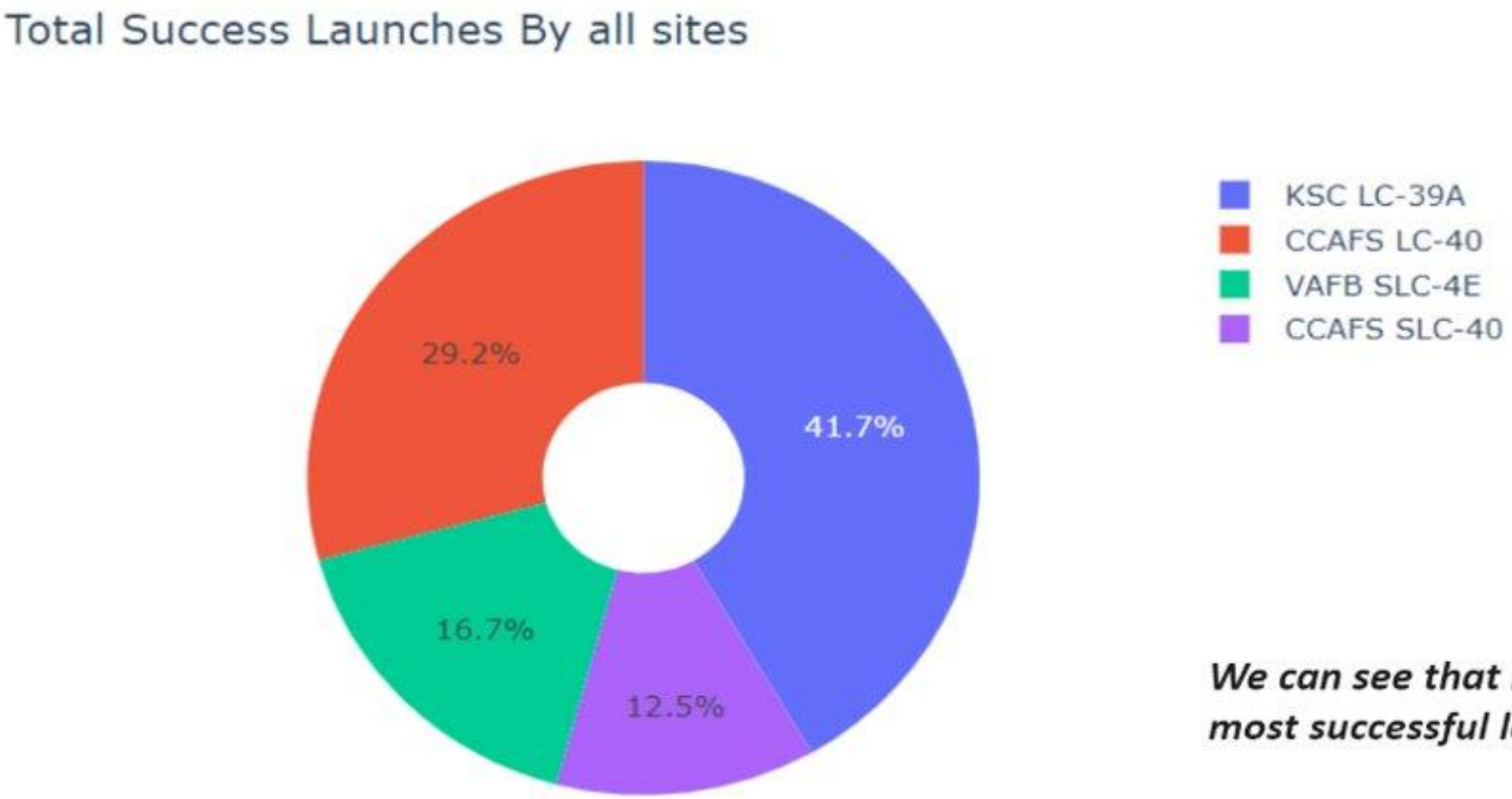




Section 4

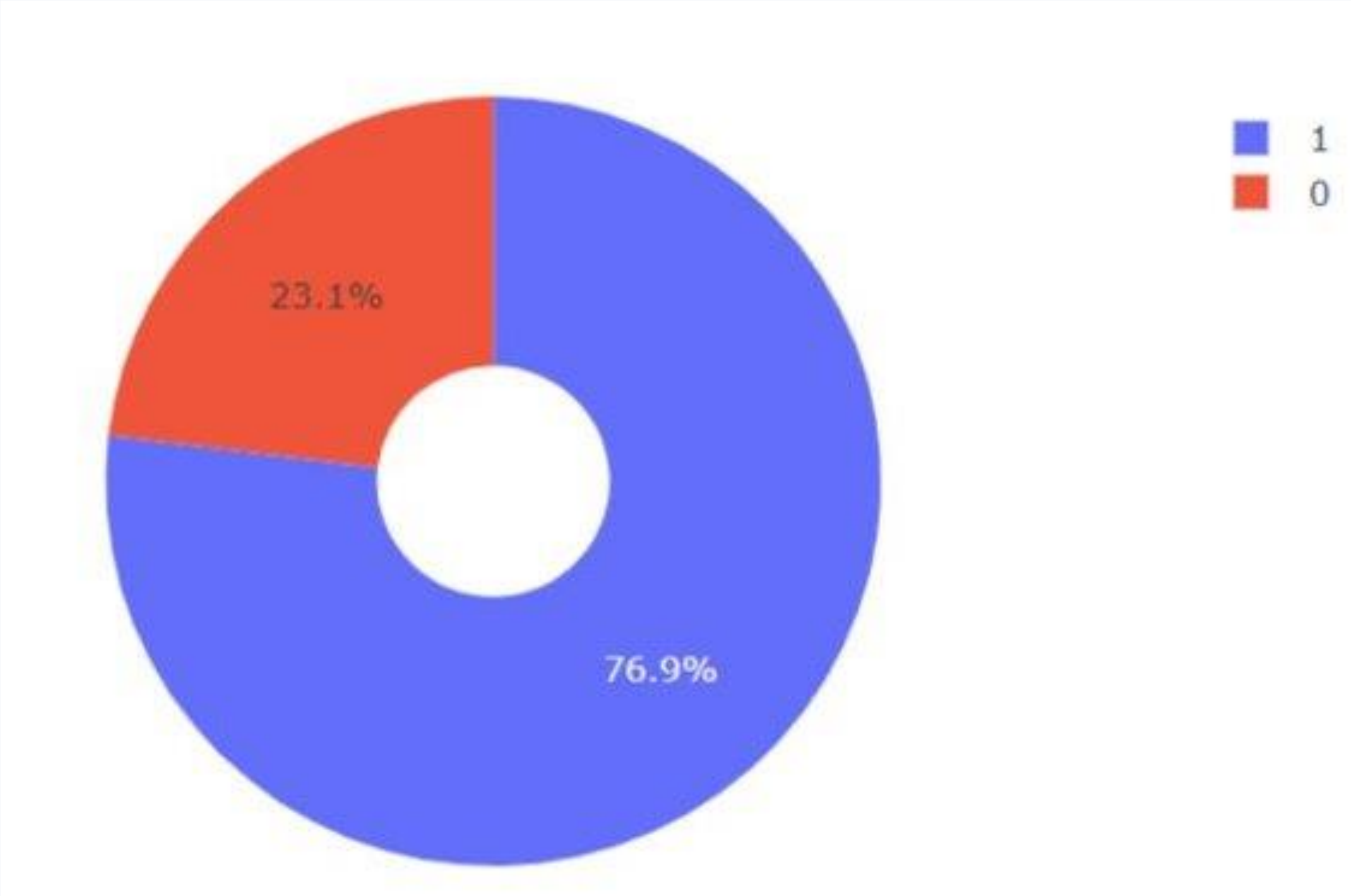
Build a Dashboard with Plotly Dash

Success Rate per Launch Site – Pie Chart



We can see that KSC LC-39A had the most successful launches from all the sites

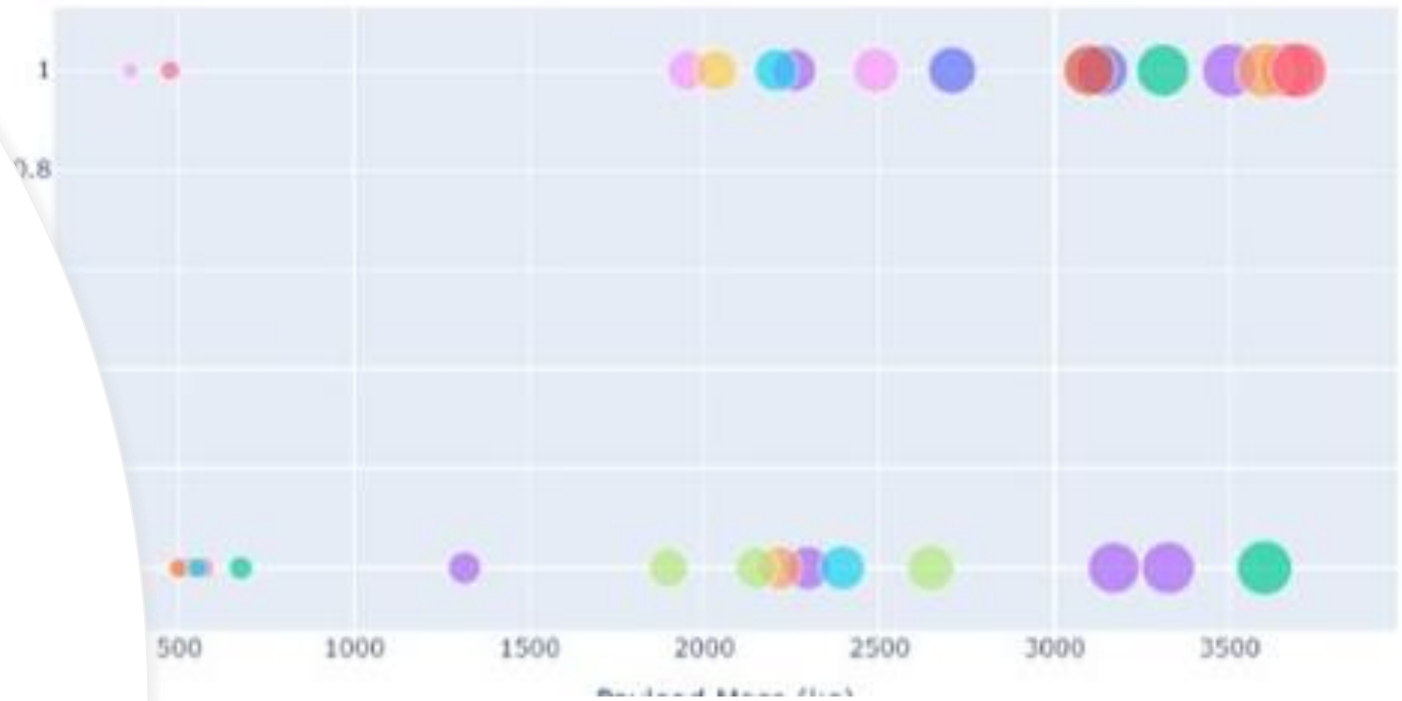
Launch Site with the highest success rate – Pie Chart

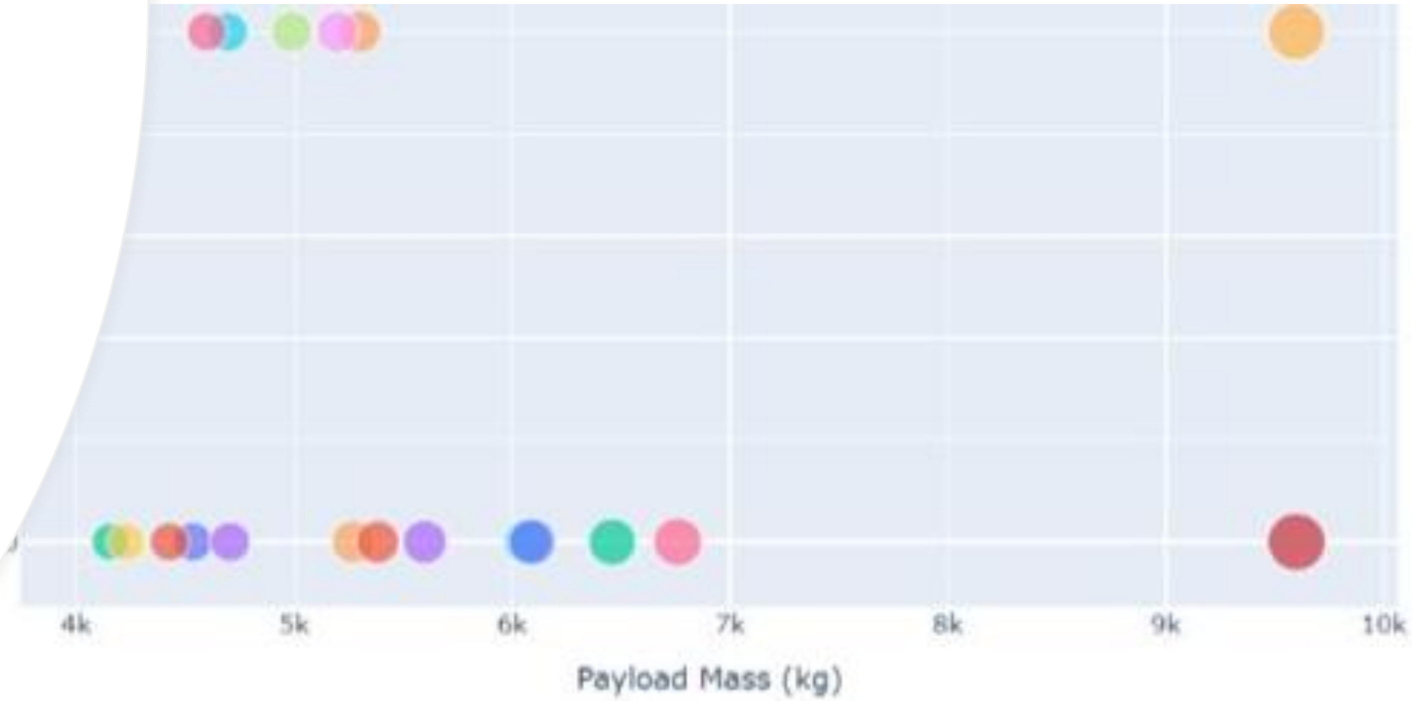


KSC LC-39A has the highest Success ratio of 76.9%

Payload to Success Rate Scatter plot

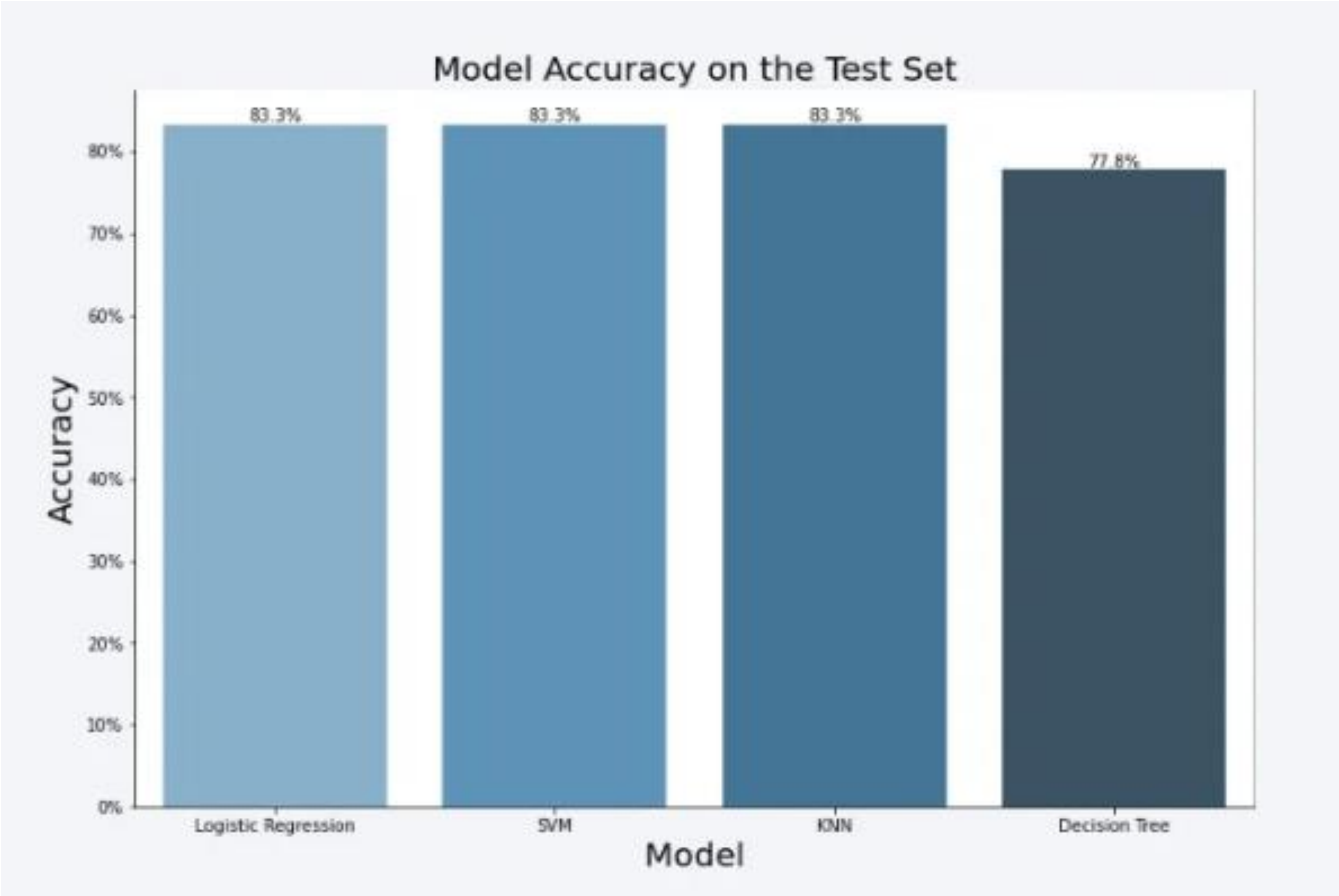
In the upper the chart refers to a Payload up to 4tons whereas the second one to a Payload beyond 4tons, but less than 10tons. The illustrations show that higher Payloads result in less successful launches





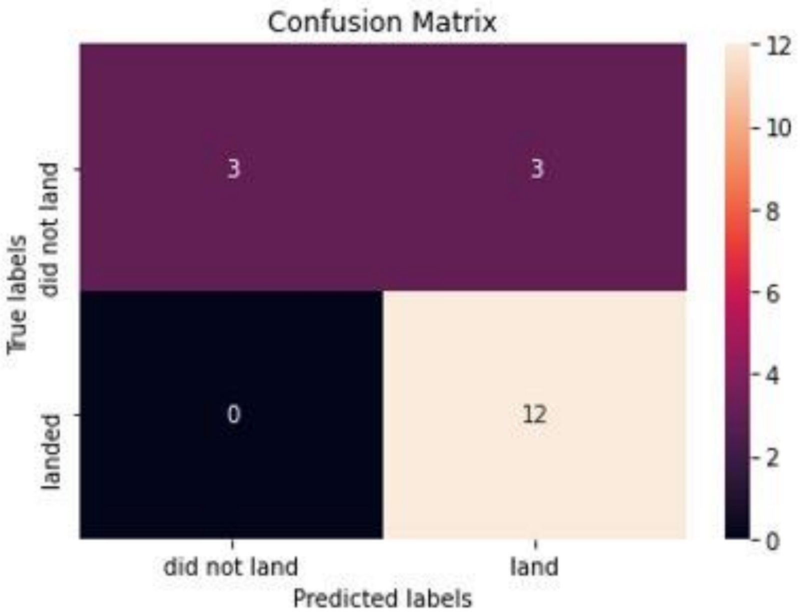
Section 5

Predictive Analysis (Classification)



Classification Accuracy

Confusion Matrix



The Matrix refers to the Decision Tree classifier

The model is not a good measurement for the 'False Negatives' since there were twelve cases where the model predicted an incorrect value

Conclusions

- ❖ Lower weight payloads are mostly connected to higher Success rates
- ❖ Machine Learning Classification models SVM, KNN and LR showed a surprisingly identical Evaluation figure
- ❖ KSC LC had the most successful launches
- ❖ Between 2013 and 2017 the Success Rate of all sites showed an upward trend
- ❖ The Launch Sites are particularly close to the coastline

Thank you!