IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jose Savio Segundo
May 30, 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection: Using Space X API and Web scrapping from Wikipedia
  - Exploratory Data Analysis: Data Wrangling, creating a database and analysing it using SQL
  - Visualization of the Data: Interactive Visual Analytics and Dashboard
  - Predictive Analysis with Machine Learning
- Summary of all results
  - The data collection resulted in a sizeble amount of data providing useful information
  - The data analysis resulted in good unsdertanding of the valueble parameters in the data to help the business decision
  - The Predictive analysis resulted in multiple models with good accuracy (83%) to predict the landing of the first stage. More data in a future work can help to increase the accuracy.

Introduction

- Project background and context
  - The goal of this work is to analyze the successful landing of the first stage of the rockets from our main competitor, Space X. This fact is considered the key point to reduce the cost of launching.
- Problems you want to find answers
  - The main question is: For each launch, will the first stage successfully land or not?
  - This main question can be derived into several other questions like: The successfully landing is a function of several factors, how does each of those factors like pay load, launch site, etc affect the landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - From Space X API (Using requests library):
    - https://api.spacexdata.com/v4/rockets/
    - https://api.spacexdata.com/v4/launchpads/
    - https://api.spacexdata.com/v4/payloads/
    - https://api.spacexdata.com/v4/cores/
    - https://api.spacexdata.com/v4/launches/past
  - Web scraping (Using BeautifulSoup library) from:
    - https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Methodology

Executive Summary

- Performed data wrangling
  - Assigned the mean pay load mass when data was not available
  - Analyzed the data to summarize the outcome as successful lending or not, from multiple launch sites, multiple orbit types and multiple landing scenarios (Ocean, ground pad and drone ship)
- Performed exploratory data analysis (EDA) using visualization and SQL
- Performed interactive visual analytics using Folium and Plotly Dash

Methodology

Executive Summary

- Performed predictive analysis using classification models
  - After all previous steps the data was standardized, split into training and test data for the machine learning step. In this step four modelling techniques were applied (Logistic Regression, Support Vector Machine, Tree Classifier, K Nearest Neighbors). All four models achieved a good prediction accuracy(83%)

Data Collection

- The data sets were collected from Space X API (Using requests library):
  - https://api.spacexdata.com/v4/rockets/
  - https://api.spacexdata.com/v4/launchpads/
  - https://api.spacexdata.com/v4/payloads/
  - https://api.spacexdata.com/v4/cores/
  - https://api.spacexdata.com/v4/launches/past
- Additional data was collected web scrapping the Wikipedia (Using BeautifulSoup library) from:
  - https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Data Collection

- SpaceX provides public data sets in their API. Using the Request library we can get the data regarding the rockets, launch site, payload and outcome of the landing.

Requested and parsed the SpaceX launch data using the GET request → Filtered the dataframe to only include Falcon 9 launches → Replaced missing values for Payload Mass with the mean of Payload Mass

- The following Wikipedia page provides detailed historical information about the SpaceX Falcon 9 launches: [List of Falcon 9 and Falcon Heavy launches](). Using the BeautifulSoup library we can get the data.

Requested the Falcon 9 Launch Wiki page from its URL → Extracted all column/variable names from the HTML table header → Created a data frame by parsing the launch HTML tables

Data Collection – SpaceX API

- [Link to Data Collection API.ipynb]()

Requested json

```
# Use json_normalize meethod to convert the json result into a dataframe
data=pd.json_normalize(response.json())
# Get the head of the dataframe
data.head()
# Create a data from launch_dict
data2=pd.DataFrame(launch_dict)
# Show the head of the dataframe
data2.head()
```

```
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = data2[data2["BoosterVersion"]!="Falcon 1"]
data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
data_falcon9
```

```
# Calculate the mean value of PayloadMass column
payload_mean = data_falcon9["PayloadMass"].mean()
# Replace the np.nan values with its mean value
data_falcon9["PayloadMass"]=data_falcon9['PayloadMass'].replace(np.nan, payload_mean)
data_falcon9["PayloadMass"]
```

Data Collection – Scraping

- [Link to Data Collection with Web Scraping.ipynb]()

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

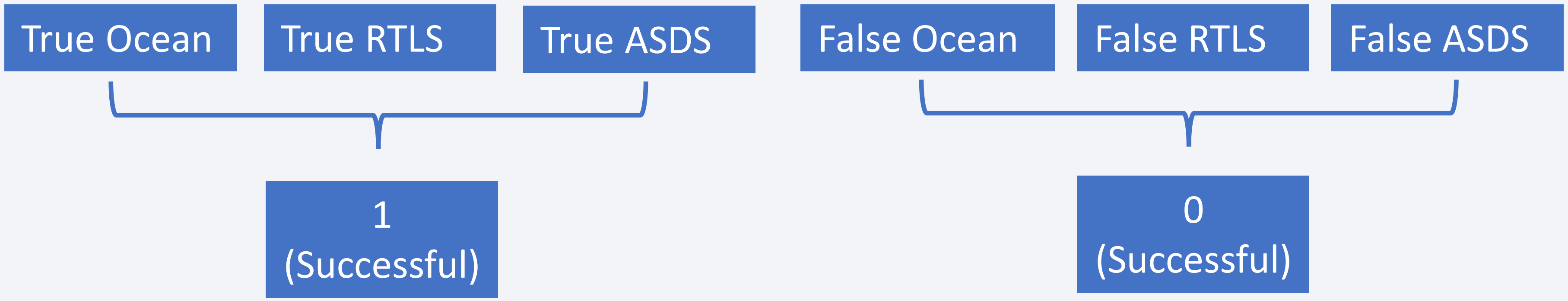
```
# use requests.get() method with the provided static_url
# assign the response to a object
response=requests.get(static_url)
# Use BeautifulSoup() to create a BeautifulSoup object
from a response text content
soup=BeautifulSoup(response.text,'html')
# Use soup.title attribute
soup.title
```

```
# Use the find_all function in the BeautifulSoup object, with element
type `table`
# Assign the result to a list called `html_tables`
html_tables=soup.find_all('table')
# Let's print the third table and check its content
first_launch_table = html_tables[2]
print(first_launch_table)
column_names = []
# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided
extract_column_from_header() to get a column name
# Append the Non-empty column name (`if name is not None and
len(name) > 0`) into a list called column_names
tc=first_launch_table.find_all('th')
for th in tc:
    name = extract_column_from_header(th)
    if name is not None and len(name) > 0:
        column_names.append(name)
print(column_names)
```

```
#Append each property to the dictionary
df=pd.DataFrame(launch_dict)
```

Data Wrangling

- The goal was convert the outcome of the launch into training labels. 1 for all successful landings and 0 for all unsuccessful landings.



Data Wrangling

- [Link to Data Wrangling.ipynb]()

```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
# landing_outcomes = values on Outcome column
landing_outcomes=df['Outcome'].value_counts()
for i,outcome in enumerate(landing_outcomes.keys()):
    print(i,outcome)
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
bad_outcomes
```

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
landing_class =
df['Outcome'].map(lambda x: 0 if x in
bad_outcomes else 1)
df['Class']=landing_class
df[['Class']].head(8)
df.head(5)
df["Class"].mean()
```

EDA with Data Visualization

- Scatter plots were plotted to visualize the relationship between the following parameters:
  - Pay load Mass x Flight Number
  - Launch Site x Flight Number
  - Launch Site x Pay load Mass
  - Orbit x Flight Number
  - Orbit x Pay load Mass
- A bar chart was plotted to visualize the Success rate for each Orbit.
- A line chart was plotted to visualize the average success rate per year
- Link to EDA with Visualization Lab.ipynb

EDA with SQL

- SQL Queries:
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'CCA'
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first successful landing outcome in ground pad was acheived.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
  - List the 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 EDA With SQL.ipynb

Build an Interactive Map with Folium

- Folium map objects:
  - Markers indicate the location of the launch sites
  - Circles highlight surrounding areas to the launch sites
  - Marker clusters indicates successful or unsuccessful launches on each launch site
  - Lines are used to indicate distances between launch sites and other points of interest (highway, railroad, city)
- Link to Interactive Visual Analytics with Folium lab.ipynb

Build a Dashboard with Plotly Dash

- The SpaceX Launch Records Dashboard has a dropdown listing all the launch sites
- After selecting on of the launch sites we can see the stats regarding the successfulness of the launches
- A rangeslider for the payload range allows the user to select the desired range for the visualization
- A scatter plot show the correlation between Payload and Success for all sites
- Link to SpaceX Dashboard.py

Predictive Analysis (Classification)

- Data preparation
  - Load the dataframe
  - Standardize the data
  - Split data X and Y into training and test data.
- Create Logistic regression object
  - Calculate accuracy of test data
  - Generate Confusion matrix
- Create a Support Vector Machine object
  - Calculate accuracy of test data
  - Generate Confusion matrix
- Create a Tree Classifier object
  - Calculate accuracy of test data
  - Generate Confusion matrix
- Create a K nearest neighbors object
  - Calculate accuracy of test data
  - Generate Confusion matrix
- Compare accuracy to fins best model (All four models have similar accuracy (around 83%)
- Link to Machine Learning Prediction.ipynb

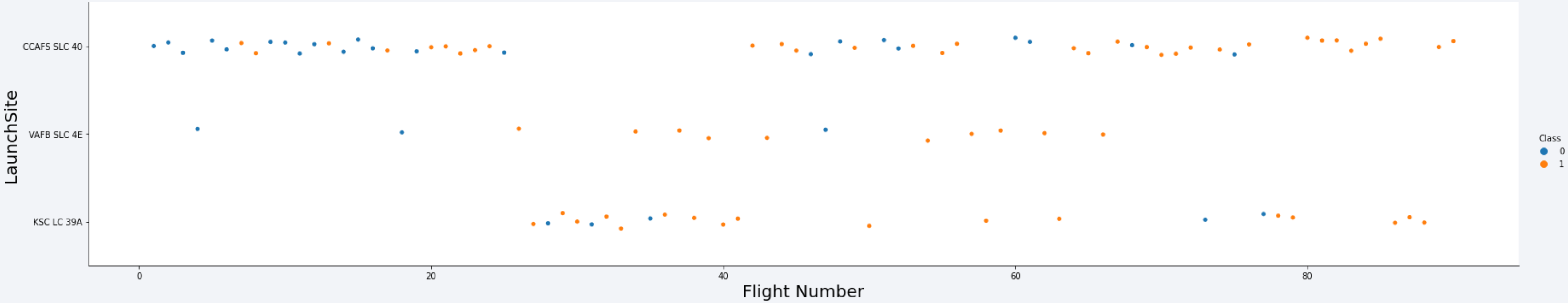
Results

- After an exploratory data analysis we can see around 67% of the launches have a successful landing.
- Since 2013 the success rate has been increasing, with an average over 80% after 2019

Section 2

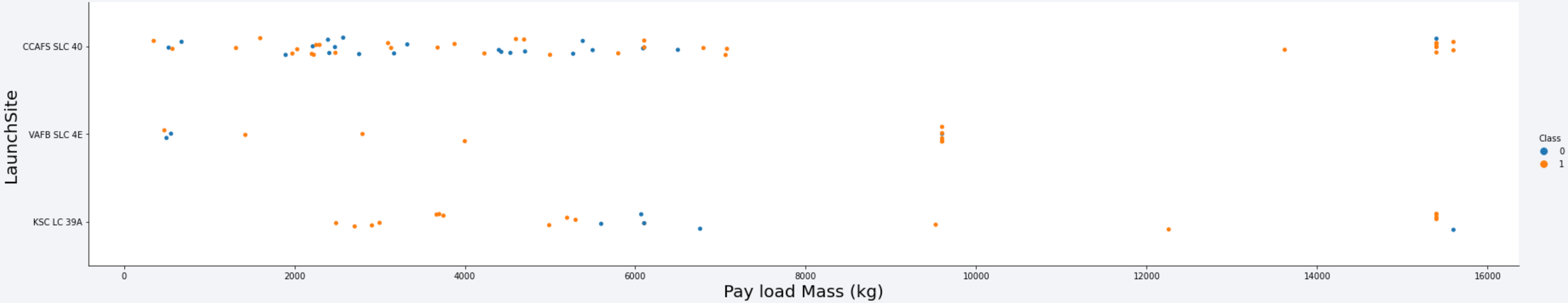
Insights drawn from EDA

Flight Number vs. Launch Site



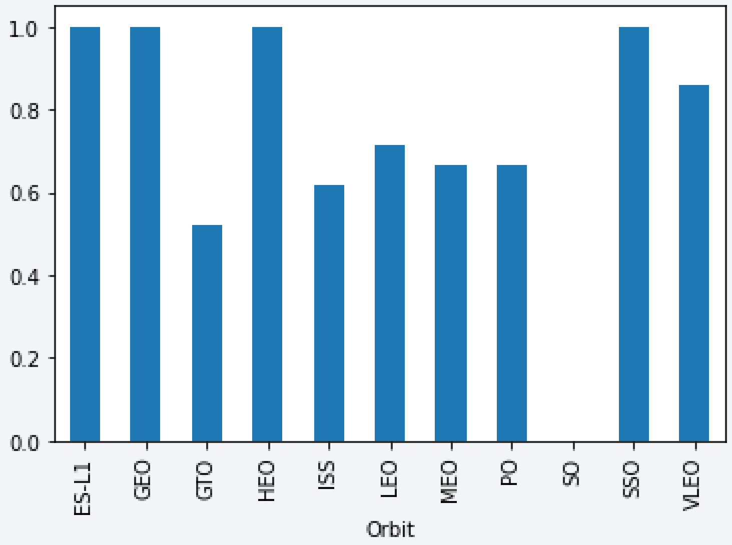
- The number of successful landings (Class=1) has been increasing with the flight number for all three launch sites

Payload vs. Launch Site



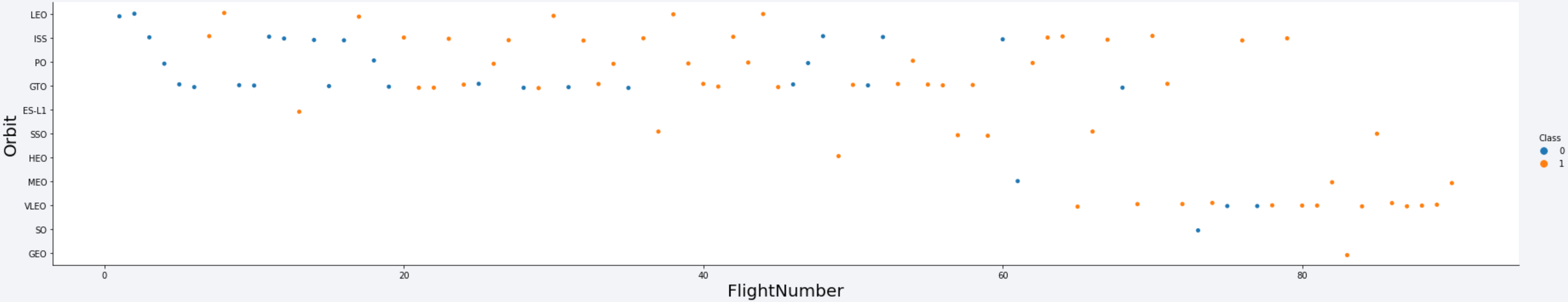
- The percentage of successful landings decreases with the payload, over 8000 kg the majority of the launches have failed.

Success Rate vs. Orbit Type



- The orbits: ES-L1, GEO, HEO AND SSO have almost 100% of success rate

Flight Number vs. Orbit Type

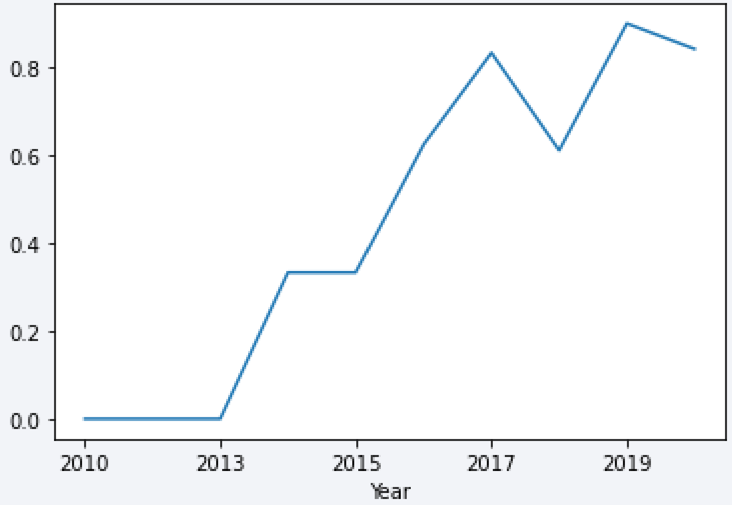


- We can se that the first launches were concentrated in the orbits: LEO, ISS, PO and GTO. After the flight number 60 most of the launches happened in the VLEO orbit

Payload vs. Orbit Type

- The majority of the launches have 'lighter' payload, less than 8000 kg. Some orbits as GTO have a wide range for the payload in the launches.

Launch Success Yearly Trend



- There is a clear increase in the success rate for the landing. From 2016 the success rate has been over 60% and after 2019 it has been over 80%.

All Launch Site Names

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

In [7]:

```
%sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL3 ORDER BY 1;
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[7]:

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

Launch Site Names Begin with 'CCA'

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

In [8]:
```
%sql SELECT * FROM SPACEXTBL3 WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[8]:

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

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

In [9]:

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SPACEXTBL3 WHERE PAYLOAD  LIKE '%CRS%';
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[9]:

total_payload
111268

Average Payload Mass by F9 v1.1

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

In [10]:
```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD FROM SPACEXTBL3 WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[10]:

avg_payload
2928

First Successful Ground Landing Date

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

Hint:Use min function

In [11]:

```
%sql SELECT MIN(DATE) AS FIRST_SUCCESS_GP FROM SPACEXTBL3 WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[11]:

first_success_gp
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [12]:

```
%sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL3 WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND LANDING__OUTCOME = 'Success (drone ship)';
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[12]:

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

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

In [13]:
```
%sql SELECT MISSION_OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL3 GROUP BY MISSION_OUTCOME ORDER BY MISSION_OUTCOME;
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[13]:

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

Boosters Carried Maximum Payload

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

In [12]:
```
%sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL3 WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL3) ORDER BY BOOSTER_VERSIO
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[12]:

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

2015 Launch Records

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

In [13]:
```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL3 WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND DATE_PART('YEAR', DATE) = 2015;
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

Out[13]:

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
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

In [14]:

```
%sql SELECT LANDING__OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL3 WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY LANDING__OUTCOME ORDER BY QTY
```

```
 * ibm_db_sa://yxr30668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
Done.
```

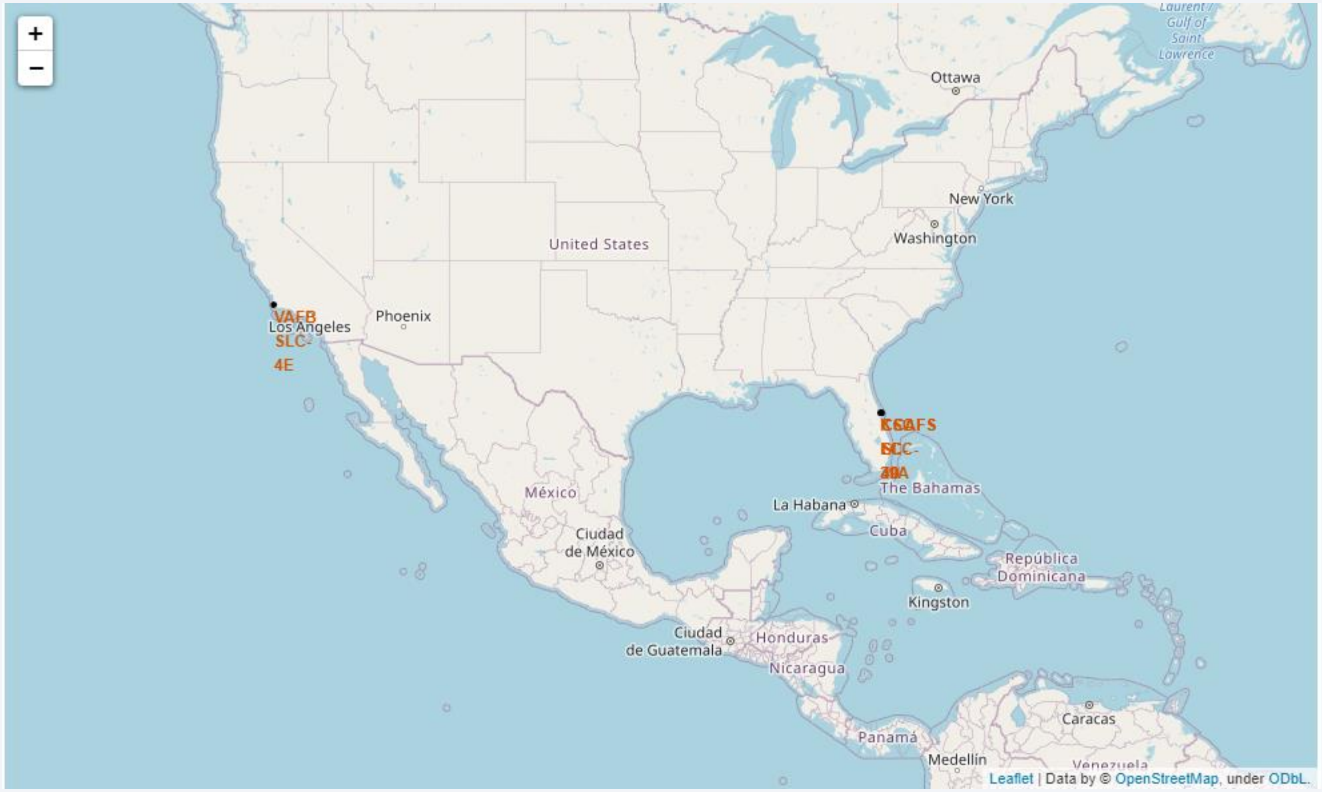
Out[14]:

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

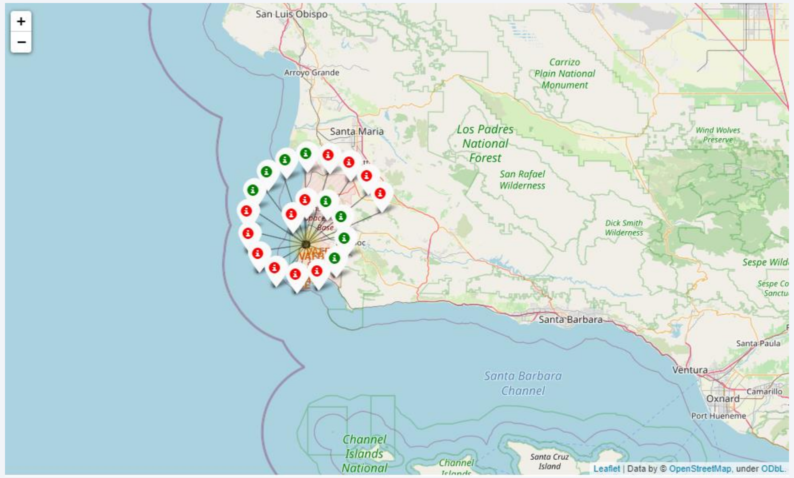
Launch Sites Proximities Analysis

Launch Sites on Map

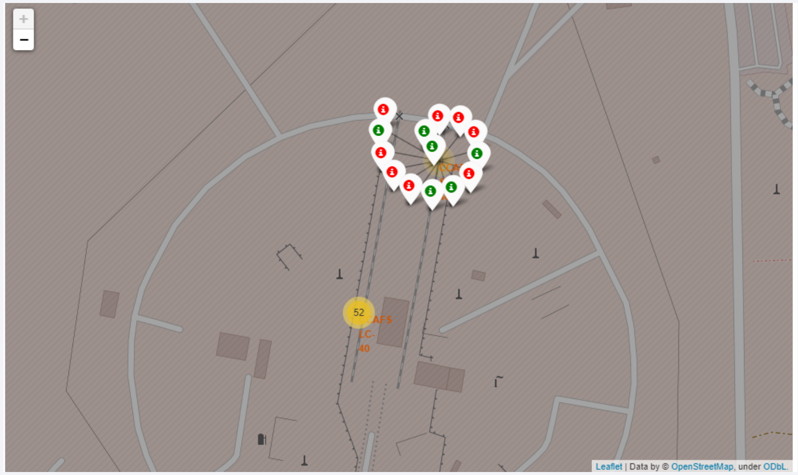


- The launch sites from Space X are in south of US as they are closer to the equator line, in order to maximize Earth's rotational speed. They are also close to the ocean as in the case something goes wrong it is more likely to fall on the ocean.

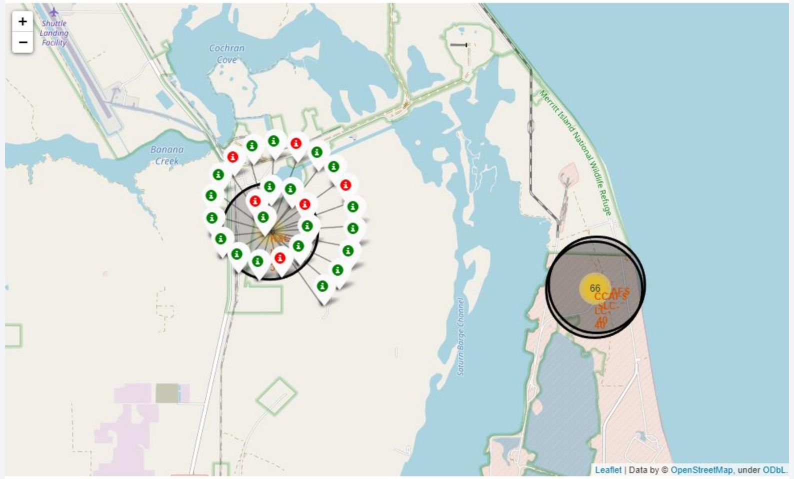
Outcome per launch site



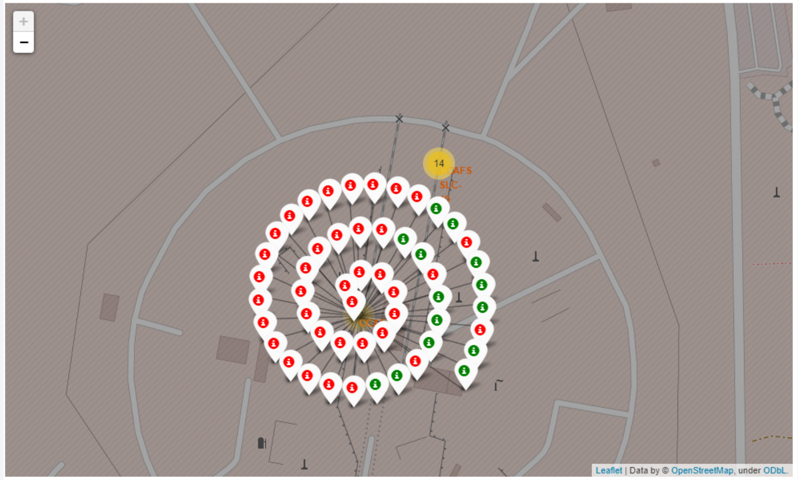
VAFB-SLC-4E



CCAFS-SLC-40

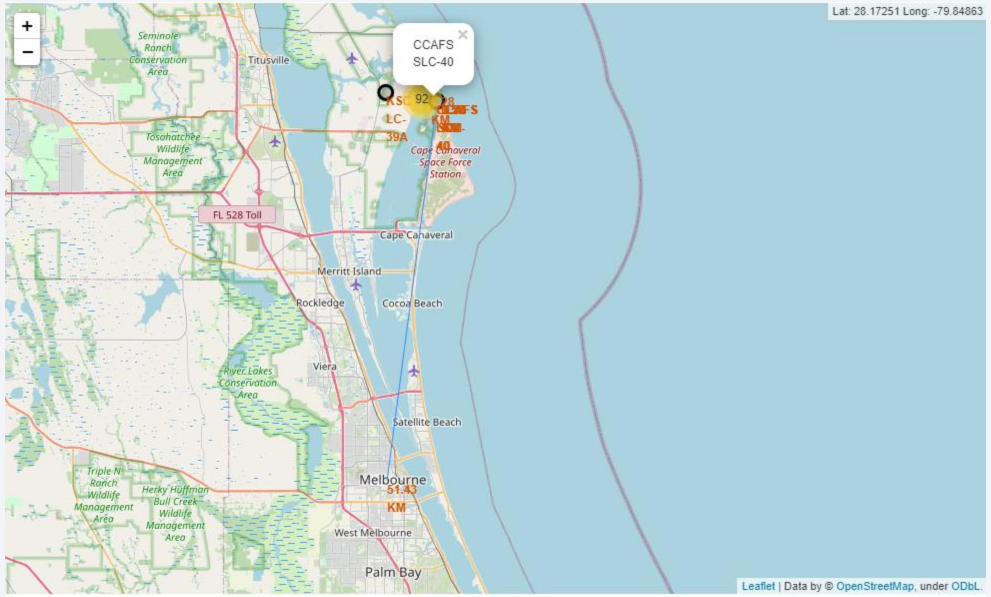


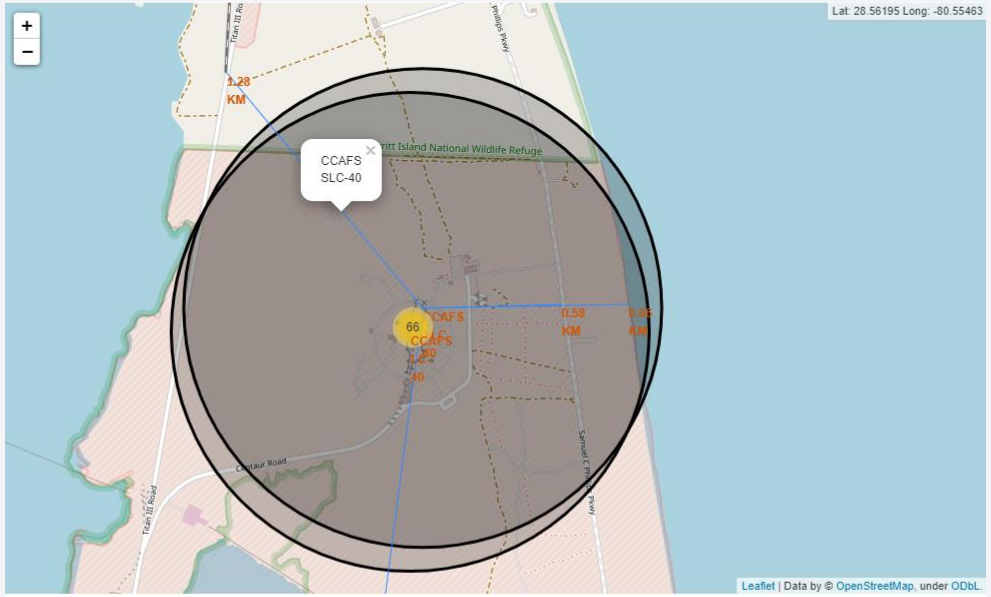
KSC-LC-39A



CCAFS-LC-40

Distance from CCAFS-SLC-40 to Railway, Coast line and Closest city

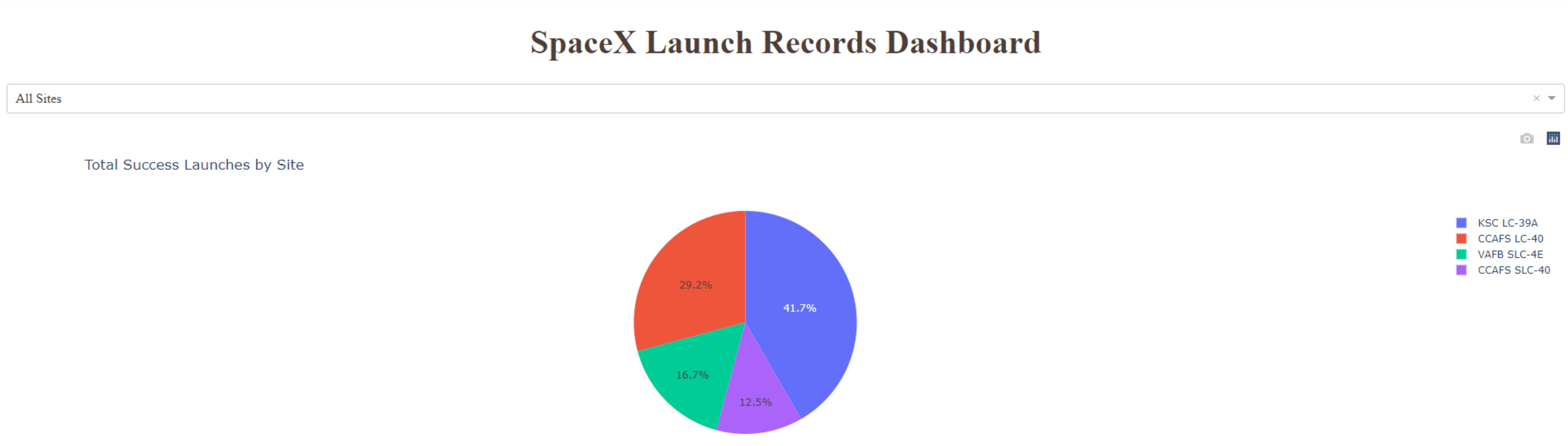




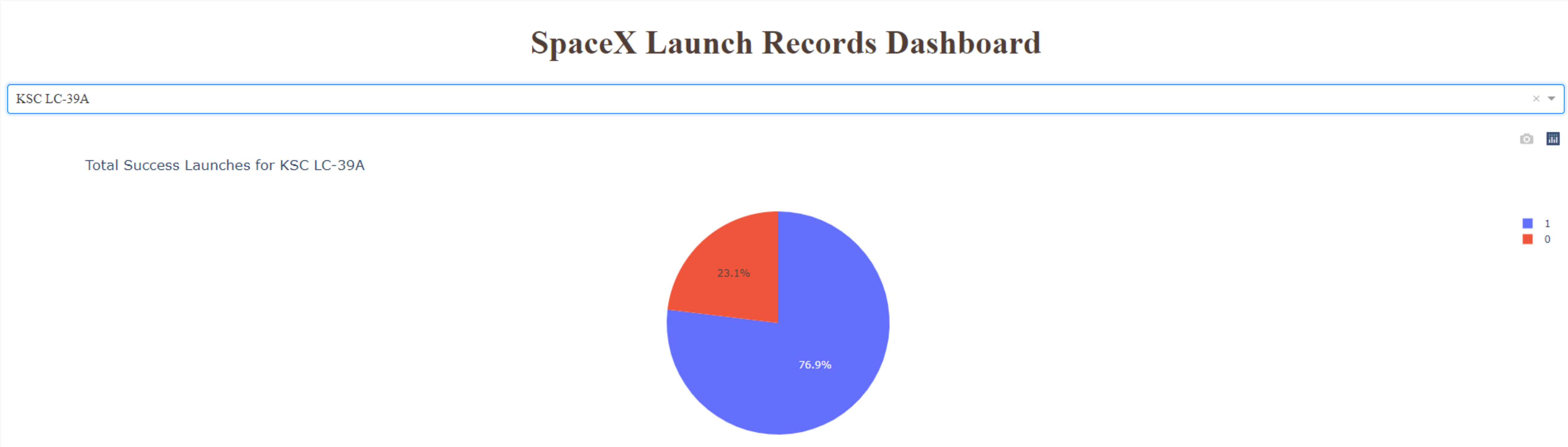
Section 4

Build a Dashboard with Plotly Dash

SpaceX Launch Records Dashboard – Launch Success for all sites

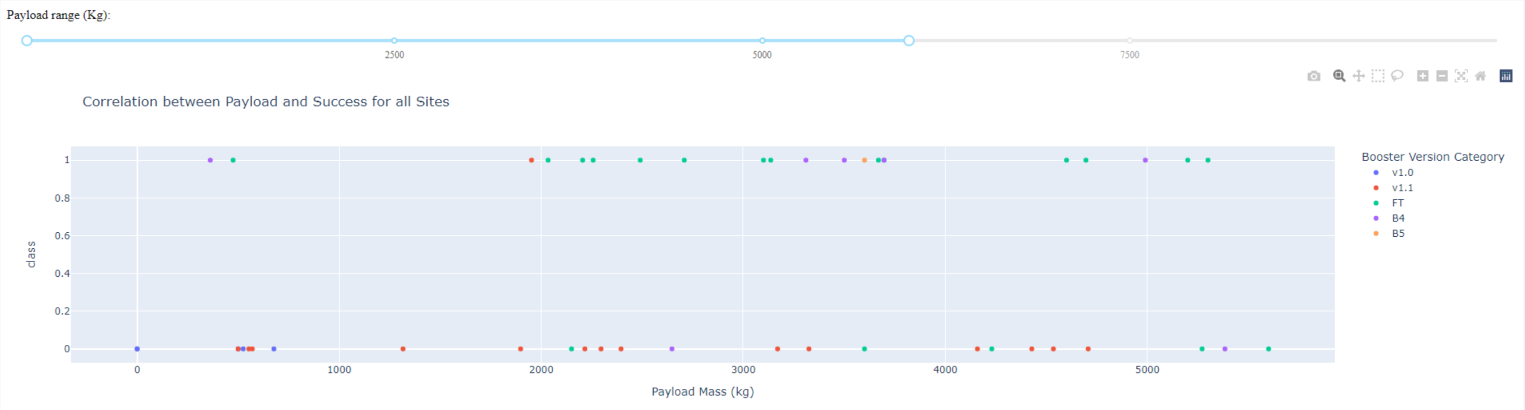


SpaceX Launch Records Dashboard – Highest launch success ratio site (KSC LC-39A)



SpaceX Launch Records Dashboard – Correlation between Payload and Success Rate

- Lighter Payload (up to 6000 kg): Higher success rate, specially for the FT booster



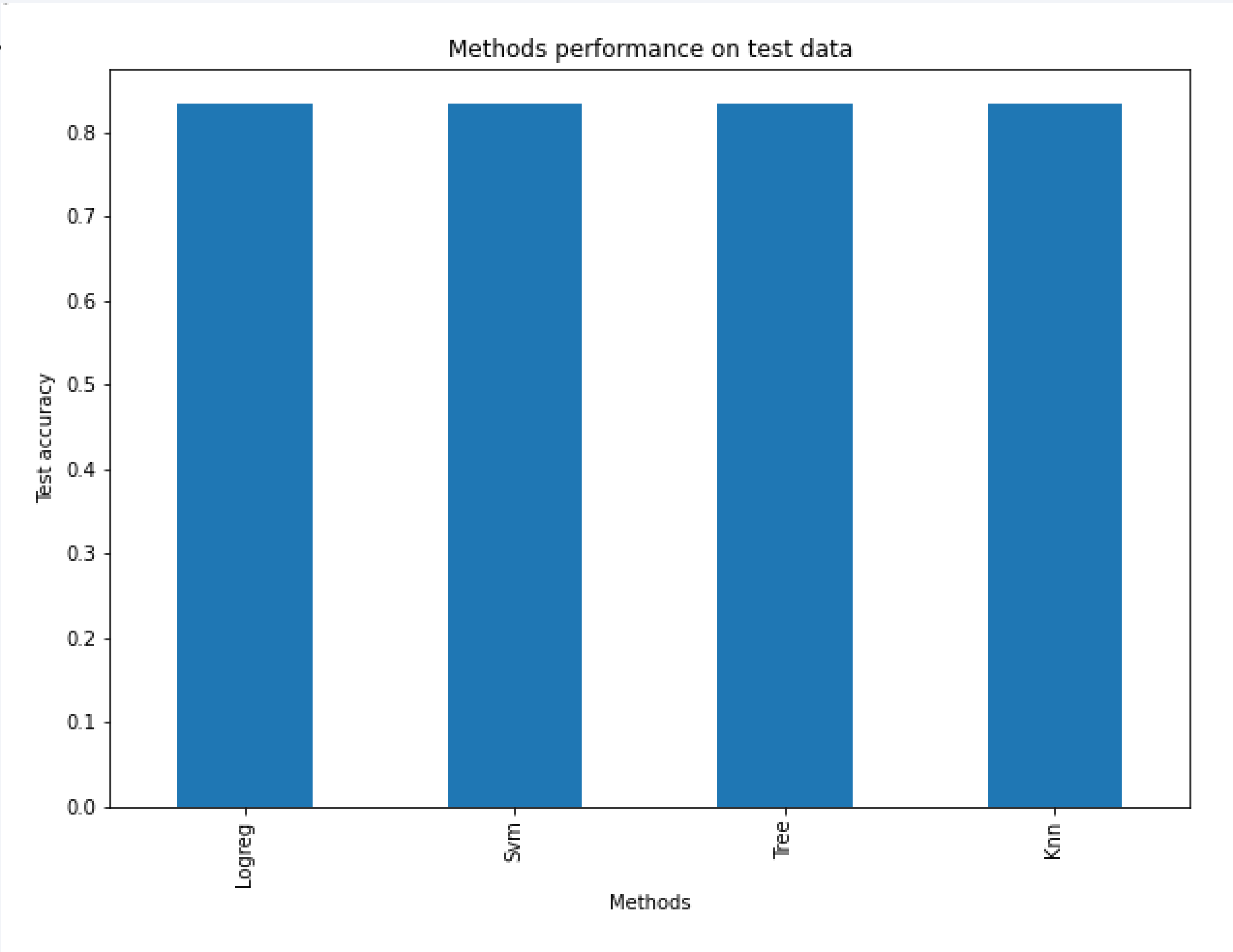
- Heavier Payload (above 6000 kg): Lower number of launches with lower success rate

Section 5

Predictive Analysis (Classification)

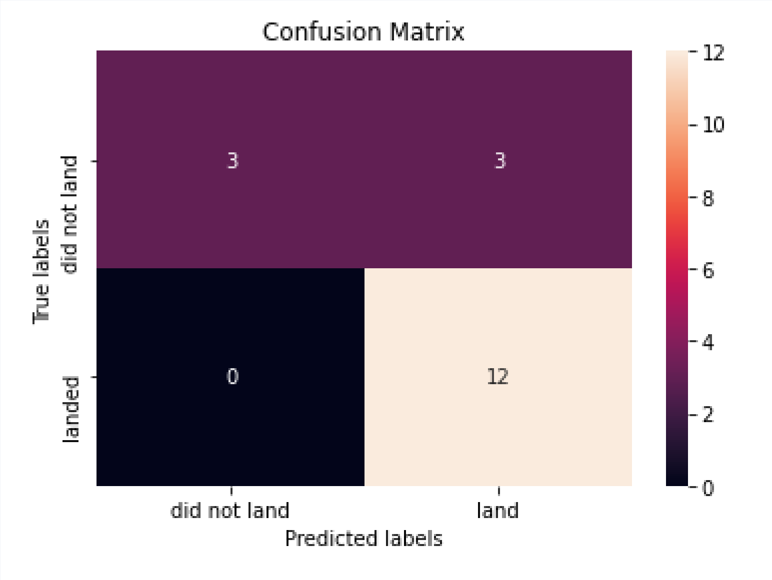
Classification Accuracy

- All four modeling techniques: Logistic Regression, Support Vector Machine, Tree Classifier, K Nearest Neighbors achieved a similar good prediction accuracy(83%).



Confusion Matrix

- All four modeling techniques: Logistic Regression, Support Vector Machine, Tree Classifier, K Nearest Neighbors achieved a similar Confusion Matrix
- All modeling techniques had false positives, which is not ideal as it is overestimating the successful landings.



Conclusions

- The data confirms that SpaceX has been increasing the success rate of the landing for the first stage of its rockets. In the most recent years this success rate has been over 80%.
- The payload seems to be one of the key factors affecting the success rate. With heavier loads making harder to have a success landing. Other key factors in the success are the orbit and the launch site.
- The Predictive analysis can predict the landing of the first stage with good accuracy (83%).
- The predictive analysis has overpredicted the successful landing (False positive) which is something to be improved, as this will underestimate the overall cost as it is assuming a higher recovery of the first stage.
- More data in a future work and a segmentation odf the data as function of key parameters as payload, orbit, launch site, etc can help to increase the accuracy.

Thank you!