Winning Space Race with Data Science

Nihanth Koka
05/18/2022

Outline

Executive Summary

Introduction

Methodology

Results

Conclusion

Summary of methodologies

Data Collection through API

Data Collection with Web Scraping

Data Wrangling

Exploratory Data Analysis with SQL

Exploratory Data Analysis with Data Visualization

Interactive Visual Analytics with Folium

Machine Learning Prediction

Summary of all results

Exploratory Data Analysis result

Interactive analytics in screenshots

Predictive Analytics result

Executive Summary

Introduction

Project background and context

Problems you want to find answers

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using Space X API and by web scraping the List of Falcon 9 and Falcon Heavy launches Wikipedia page.
- Perform data wrangling
  - One-hot encoding was applied, and irrelevant columns were dropped.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Logistic Regression, KNN, SVM and Decision tree models were built and trained with train and test data set to find the best model that fits our data.

Data Collection

Describe how data sets were collected.

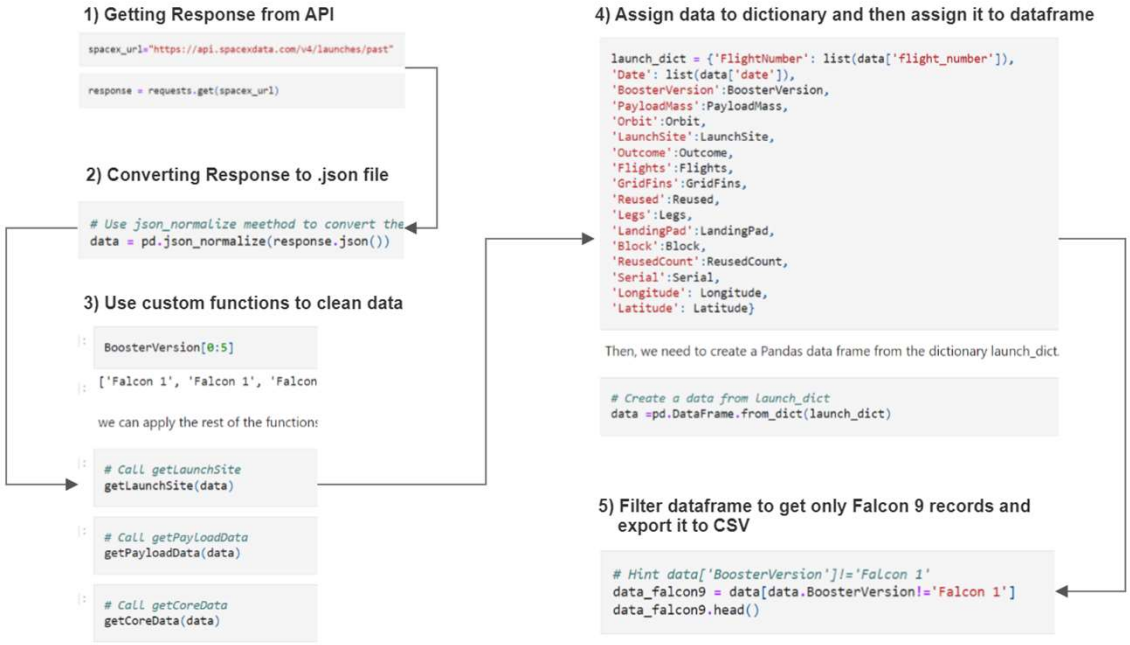
You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

We used the get request to the SpaceX API to collect data, clean the requested data and did some basic data wrangling and formatting and converted it to a data frame for further analysis.

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

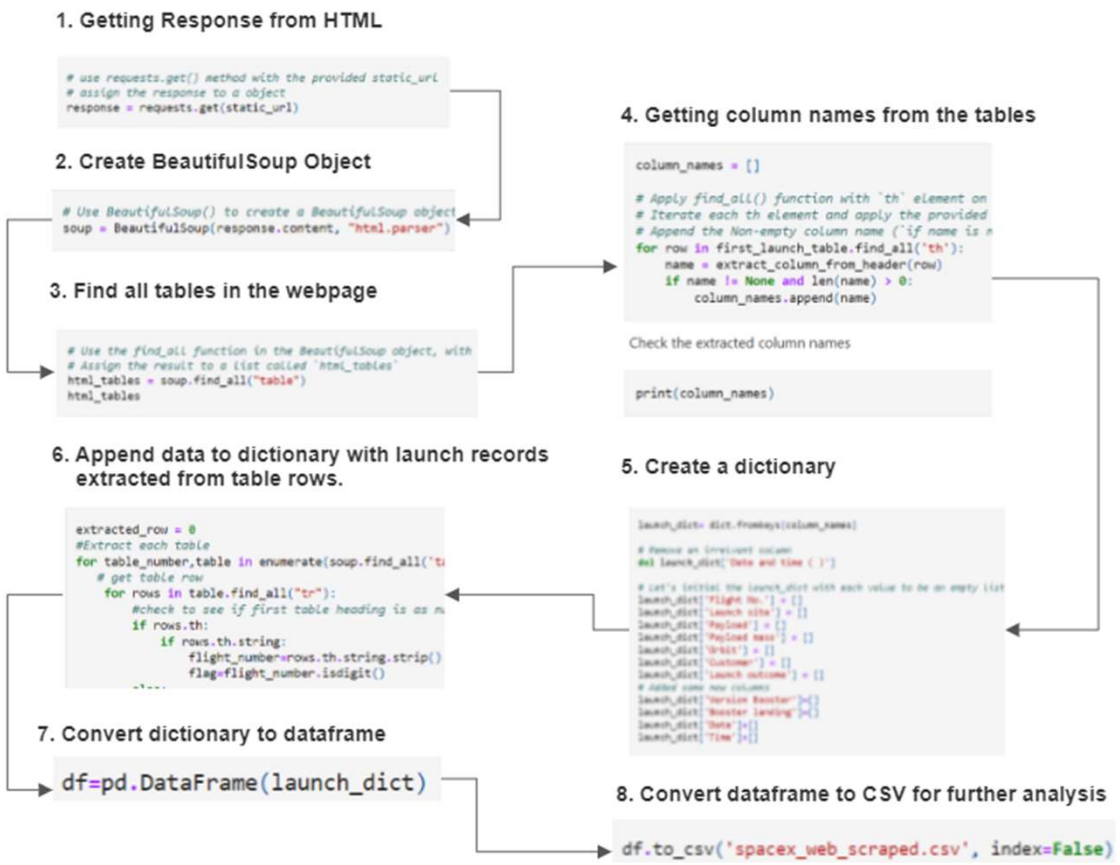


Data Collection – Scraping

We applied web scrapping to get Falcon 9 launch records from its Wikipedia page with BeautifulSoup. We parsed the table and converted it into a pandas dataframe.

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling

1. Perform Exploratory Analysis on the data set.
2. Calculate the number of launches on each site.
3. Calculate the number of occurrences on each orbit.
4. Create a landing outcome label from Outcome column
5. Calculate the success rate by calculating the mean of landing outcome column.
6. Export the dataset to CSV.

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

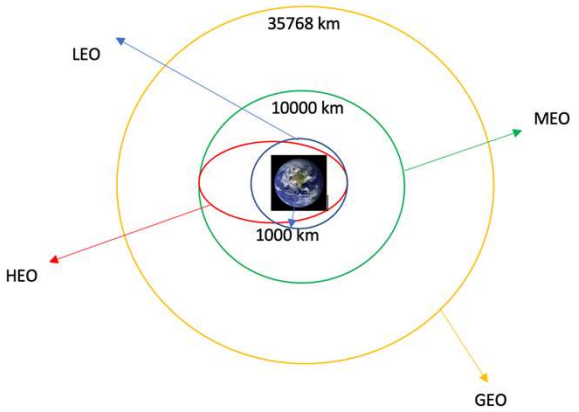


Figure showing different orbits SpaceX used to launch rockets.

EDA with Data Visualization

Scatter Graph

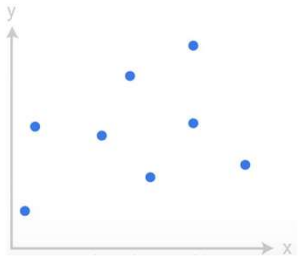
Flight Number vs Payload Mass

Flight Number vs Launch Site

Payload Mass vs Launch Site

Flight Number vs Orbit

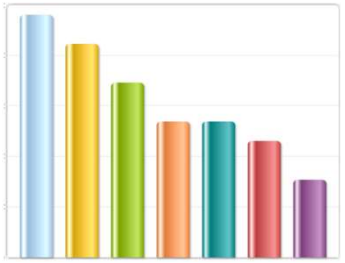
Payload Mass vs Orbit



A scatter plot is a type of plot or mathematical diagram using Cartesian coordinates to display values for typically two variables for a set of data.

Bar Graph

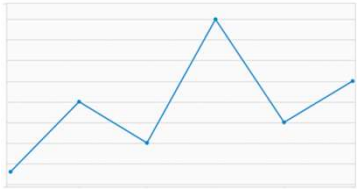
Orbit vs Success Rate



A bar chart or bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent.

Line Graph

Year vs Success Rate



A line chart is a type of chart which displays information as a series of data points called 'markers' connected by straight line segments.

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

First the excel file is loaded in DB2 database in IBM Watson Cloud and is connected from Jupyter notebook to do analysis using SQL.

The following analysis were done from the data by writing SQL queries:

- Getting the names of the unique launch sites in the space mission.
- Getting 5 records where launch sites begin with the string 'CCA'.
- Calculate the total payload mass carried by boosters launched by NASA (CRS)
- Getting the date when the first successful landing outcome in ground pad was achieved.
- Getting the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
- Getting the total number of successful and failure mission outcomes.
- Getting the names of the booster versions which have carried the maximum payload mass.
- Getting the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Ranking 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.

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera.ipynb

Build an Interactive Map with Folium

- We marked all launch sites, and added map objects such as markers, circles, lines to mark the success or failure of launches for each site on the folium map.
- For a successful launch we placed a green marker and for unsuccessful launch we placed a red marker around the launch site.
- We calculated the distance from launch site to various landmarks like coastline, city, highways and railways. We concluded the below from that

Are launch sites in close proximity to railways? No
Are launch sites in close proximity to highways? No
Are launch sites in close proximity to coastline? Yes
Do launch sites keep certain distance away from cities? Yes

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

A dashboard was built using Plotly Dash and the user is given option to select any Launch Site and Payload range and plot the results.

Graphs
Pie Chart:

A pie chart is plotted showing success rate of all launch or a particular launch site based in user selection.

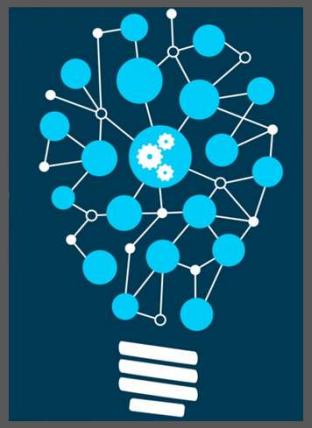
Scatter Chart:

A scatter plot is plotted for Payload Mass(kg) vs class(success or failure of a launch). The user can select their desired payload range and view the results.

Github Notebook Link: https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/Space%20X%20Plotly%20dash.ipynb

Github Python Code Link: https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/plotlydash.py

Predictive Analysis (Classification)



Numpy and Pandas were used to transform the data and the data is split into training and testing data set.

The following algorithms were used on training dataset

1) Logistic Regression

2) Support Vector Machines

3) Decision Tree

4) K Nearest Neighbors

GridSearchCV is used to tune different hyper parameters for the above 4 models to find the best parameters.

Accuracy score is calculated for all 4 models and the model with the highest score is used.

Github Notebook Link:

https://github.com/nihanth123/Applied-Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

Exploratory data analysis results

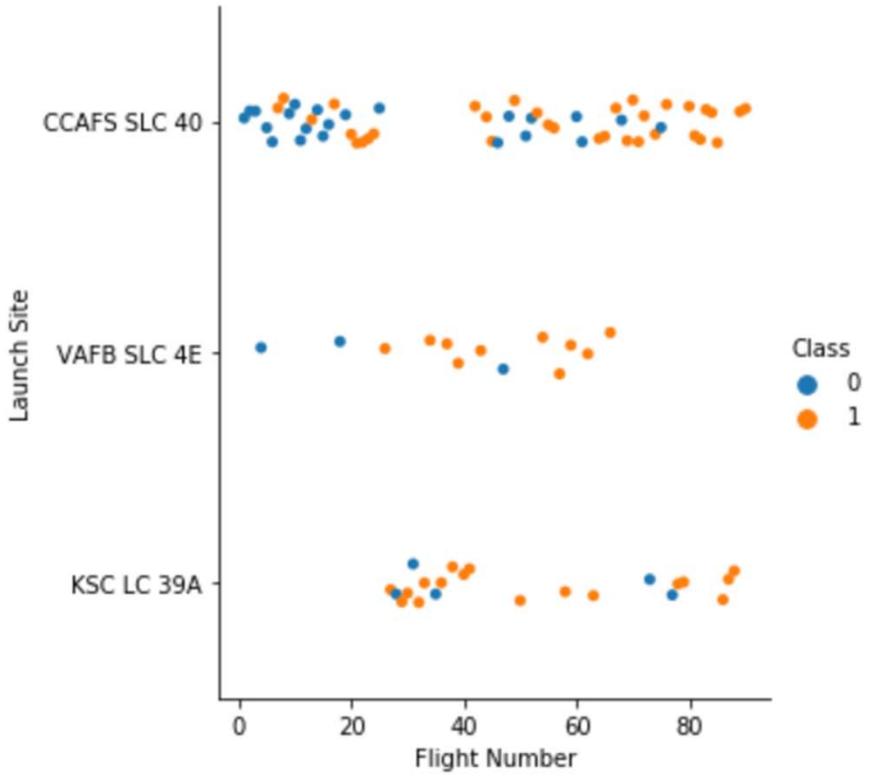
Interactive analytics demo in screenshots

Predictive analysis results

Section 2

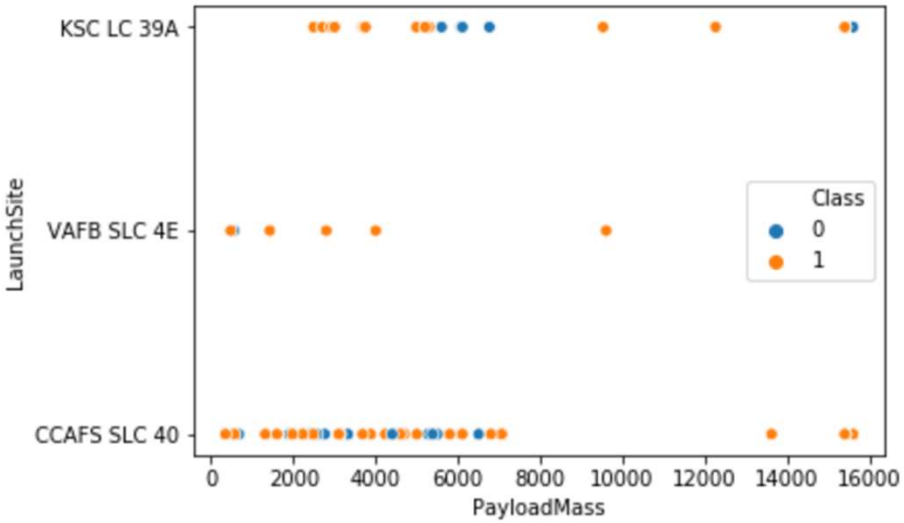
Insights drawn from EDA

FLIGHT NUMBER VS. LAUNCH SITE



From the plot we can conclude that the larger the flight number at the launch site the greater the success rate.

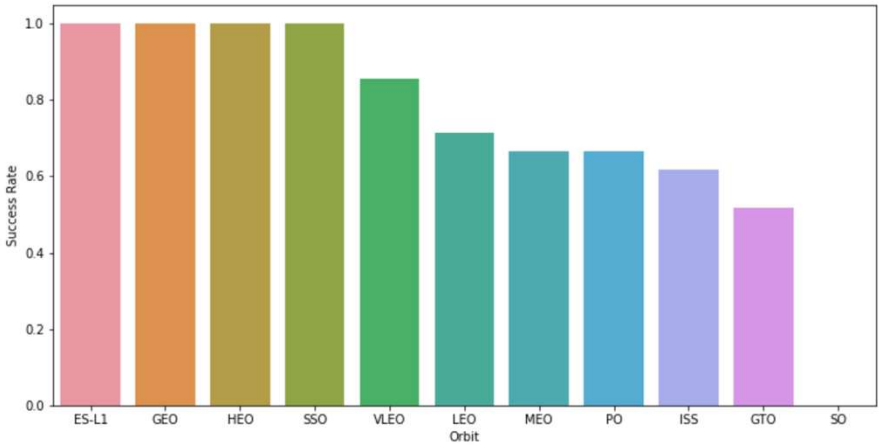
Payload vs. Launch Site



From the plot we can say that the higher the payload mass for launch site CCAFS SLC 40 the higher the success rate of the rocket.

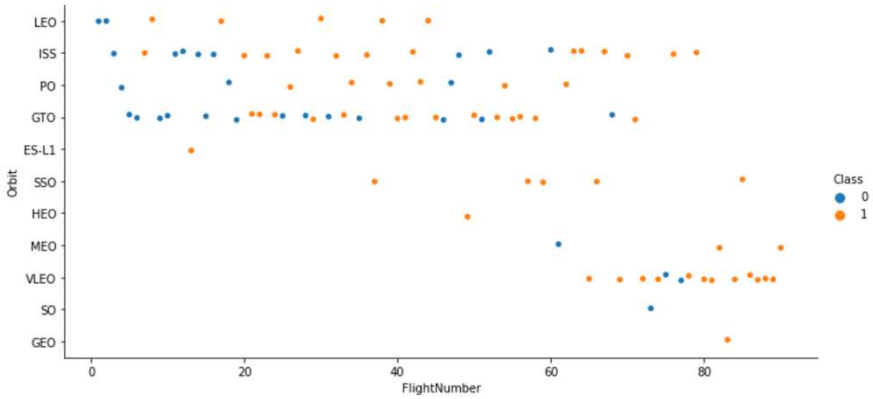
We cannot come to any conclusions for the other two launch sites based on the visualization.

Success Rate vs. Orbit Type



From the bar plot we can say that orbits ES-L1, GEO, HEO and SSO have highest success rate compared to other orbits

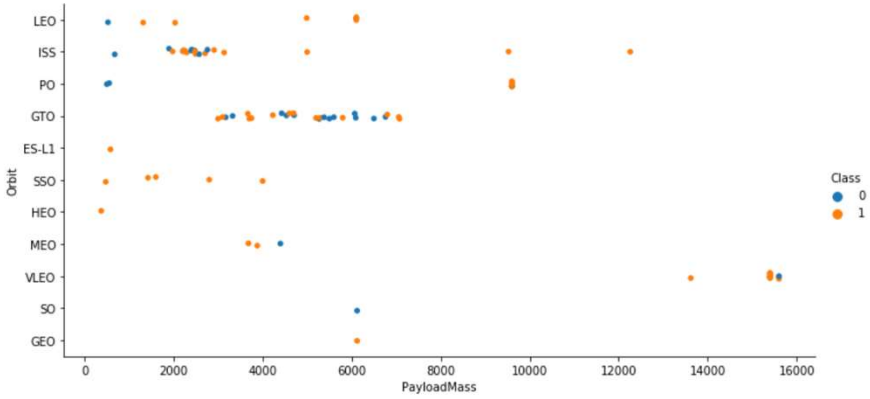
Flight Number vs. Orbit Type



From the scatter plot we can say that for LEO orbit the success rate increases with number of flights.

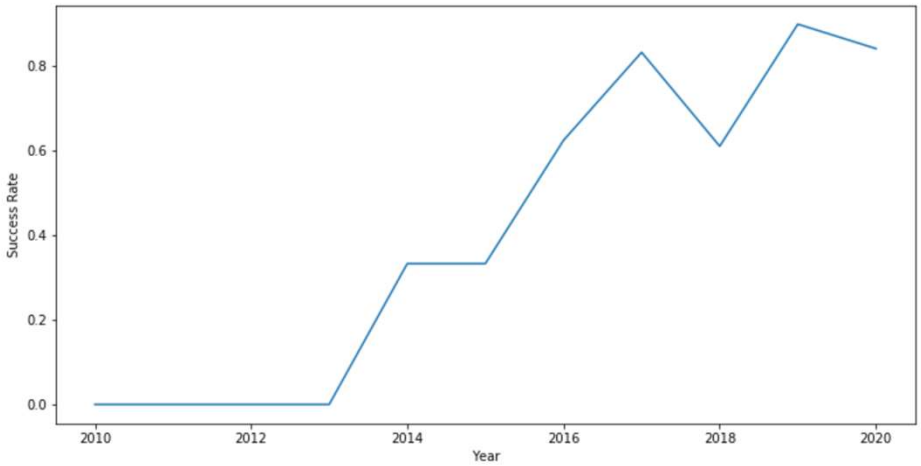
For GTO orbit there is no relationship between flight number and the success of the launch.

Payload vs. Orbit Type



From the scatter plot we can say that the higher the Payload Mass for orbits LEO, ISS and PO the more the success rate.

Launch Success Yearly Trend



From the line plot we can say that the success rate started to increase from year 2013 and kept on increasing till 2020.

All Launch Sites Names

We used distinct statement on the SPACEX table to get the unique names of launch sites.

```
%%sql
select distinct launch_site from bvj73427.SpaceX
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776
Done.
```

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

Launch Site Names Begin with 'CCA'

We use like SQL statement in where clause to get all the launch sites that begin with CCA and then used limit to limit the results to 5.

```
%%sql
select * from bvj73427.SpaceX
where launch_site like 'CCA%'
limit 5
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/bludb
Done.
```

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

We use sum function to get the total payload mass and filter it by NASA (CRS) to get the total payload mass for just that customer.

```
%%sql
select sum(payload_mass__kg_) as "Total Payload Mass" from bvj73427.SpaceX
where customer = 'NASA (CRS)'
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nq
Done.
```

Total Payload Mass
45596

Average Payload Mass by F9 v1.1

We use avg function and filter the booster_version by F9 v1.1 to get the avg payload mass for that particular booster version.

```
%%sql
select avg(payload_mass__kg_) as "Avg Payload Mass"
from bvj73427.SpaceX
where booster_version = 'F9 v1.1'
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa
Done.
```

Avg Payload Mass
2928

First Successful Ground Landing Date

```
%%sql
select min(DATE) as "First Successful ground landing"
from bvj73427.SpaceX
where  landing__outcome = 'Success (ground pad)'
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776
Done.
```

First Successful ground landing
2015-12-22

By using min function we can find the date of first successful landing.

The first successful ground landing occurred on December 22nd, 2015.

Successful Drone Ship Landing with Payload between 4000 and 6000

We filter the data based on landing outcome and payload mass kg to get the desired results.

```
%%sql
select booster_version from bvj73427.SpaceX
where  landing__outcome = 'Success (drone ship)'
and payload_mass__kg_ between 4000 and 6000
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

We use two different queries one for Successful missions and one for Failure missions. The union of those 2 queries will give us the required results.

```
%%sql
select 'Success' as "Outcome", count(mission_outcome) as "No of Outcomes" from bvj73427.SpaceX
where mission_outcome like 'Succ%'
union
select 'Failure' as "Outcome", count(mission_outcome) as "No of Outcomes" from bvj73427.SpaceX
where mission_outcome like 'Fail%'
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.a
Done.
```

Outcome	No of Outcomes
Failure	1
Success	100

```
%%sql
select distinct booster_version from bvj73427.SpaceX
where payload_mass__kg_ in (
    select max(payload_mass__kg_) from bvj73427.SpaceX)
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa
Done.
```

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

Boosters Carried Maximum Payload

We use subquery in where clause to filter payload mass kg to select maximum value for each booster version.

2015 Launch Records

We filter the data on landing outcome and year to get the 2015 launch records which have failed.

```
%%sql
select landing__outcome , booster_version, launch_site from bvj73427.SpaceX
where landing__outcome = 'Failure (drone ship)'
and year(DATE) =2015
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnr
Done.
```

landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

```
%%sql
select landing__outcome, count(landing__outcome) as "Count" from bvj73427.SpaceX
where DATE between '2010-06-04' and '2017-03-20'
group by landing__outcome
order by  count(landing__outcome) desc
```

```
 * ibm_db_sa://bvj73427:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39
Done.
```

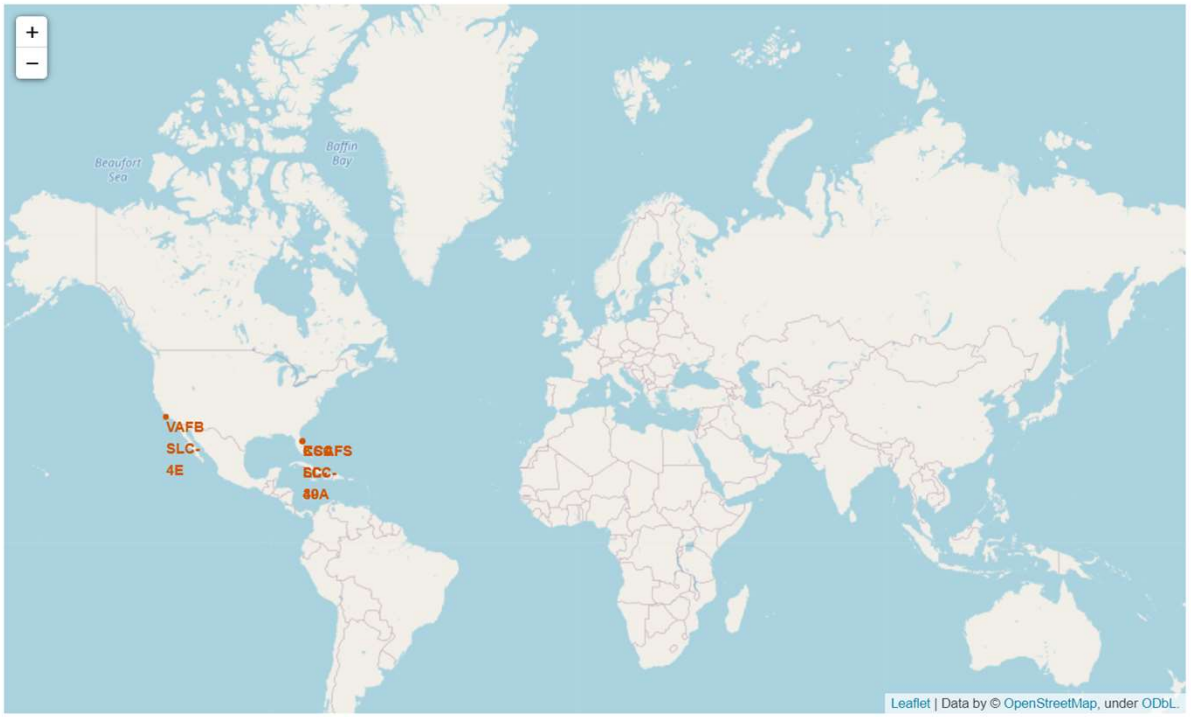
landing__outcome	Count
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

We count all the landing outcomes and filter it by date between 2010-06-04 and 2017-03-20. Then group by landing outcomes and use order by to get the results in descending order.

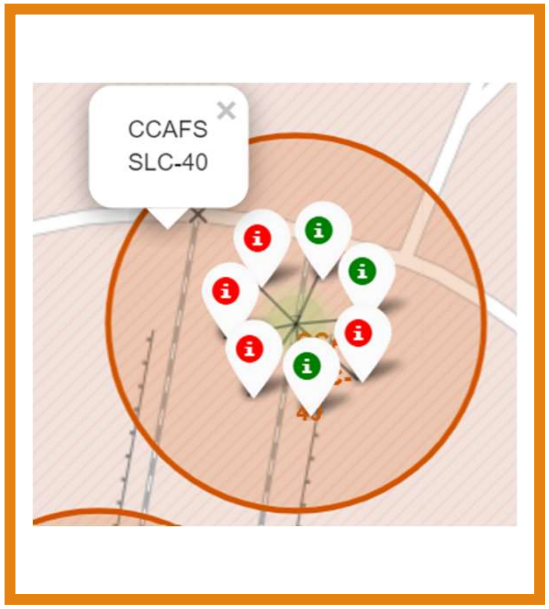
Section 3

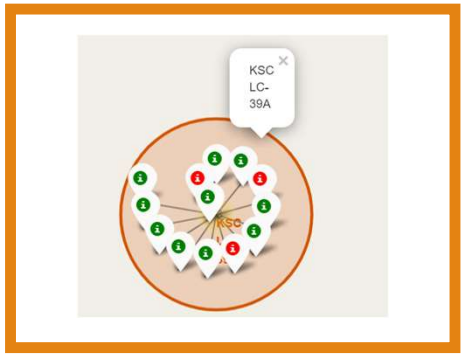
Launch Sites Proximities Analysis

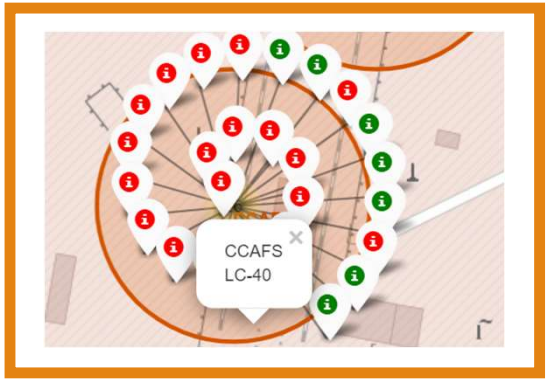
All Global Launch Sites Markers



We can say that launch sites are close to coast in Florida and California in the U.S







Markers showing Success/Failure of Launch

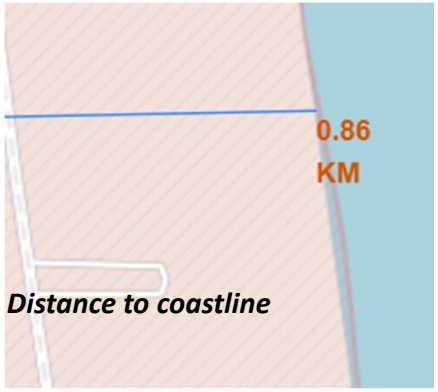
Green Marker shows successful launches and Red marker shows an unsuccessful launch.

CCAFS SLC-40, CCAFS LC-40 and KSC LC-39A are Florida launch sites.

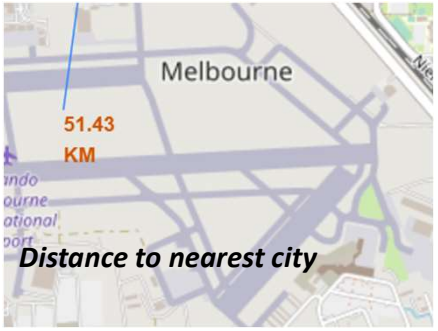
VAFB SLC-4E is California launch site.

Launch Sites Distance to Landmarks

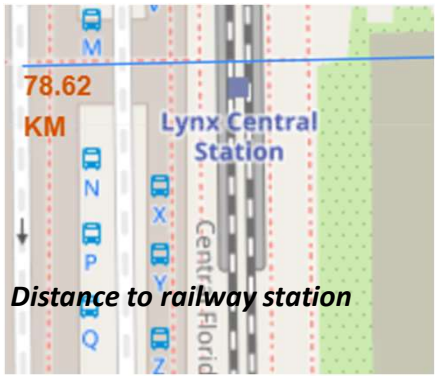
Distance to highway



Distance to coastline

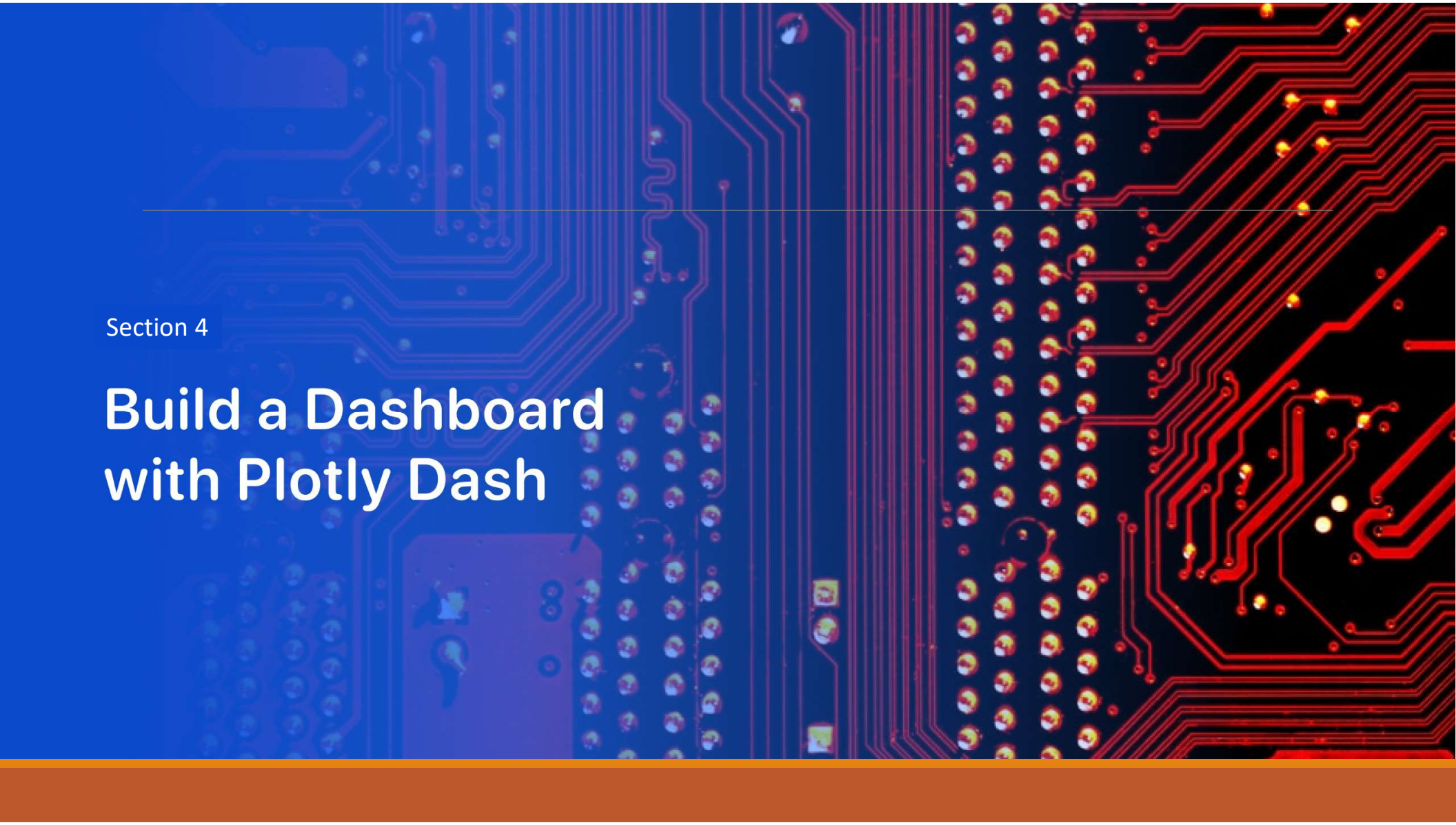


Distance to nearest city



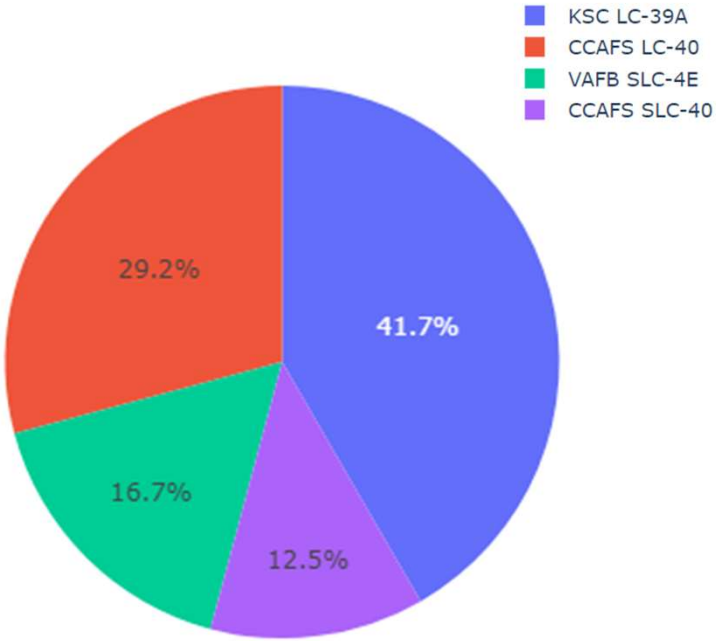
Distance to railway station

From our analysis we can say that launch sites are close to coastline and far away from highways, cities and railways.



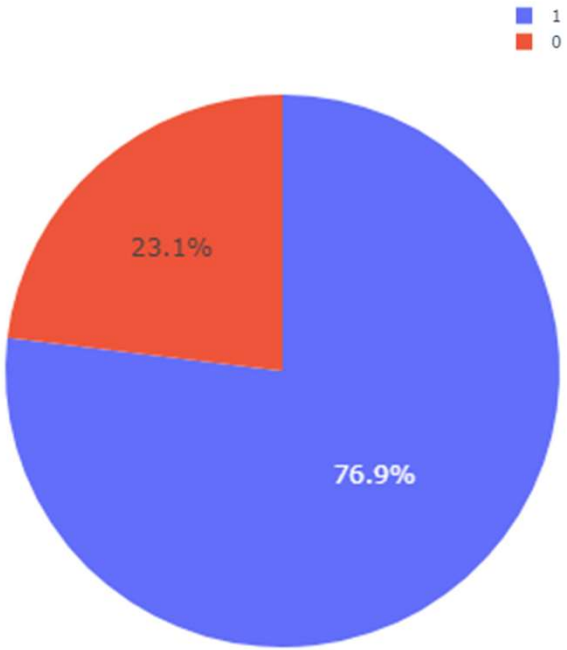
Section 4

Build a Dashboard with Plotly Dash



Pie Chart showing success percentage of each launch site

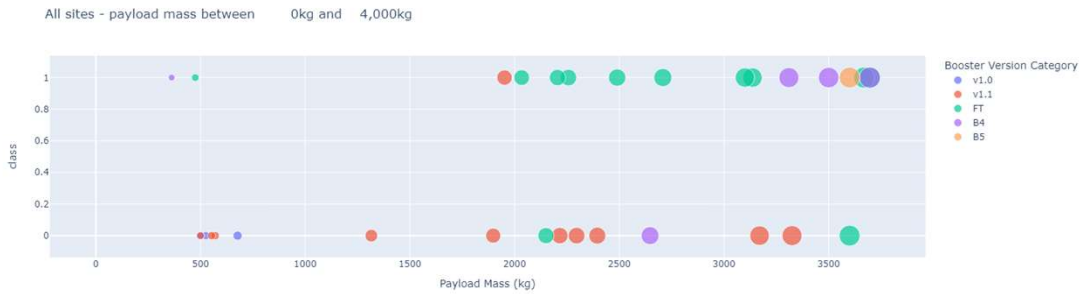
We can say that KSC LC-39A has higher success percentage compared to other three launch sites.



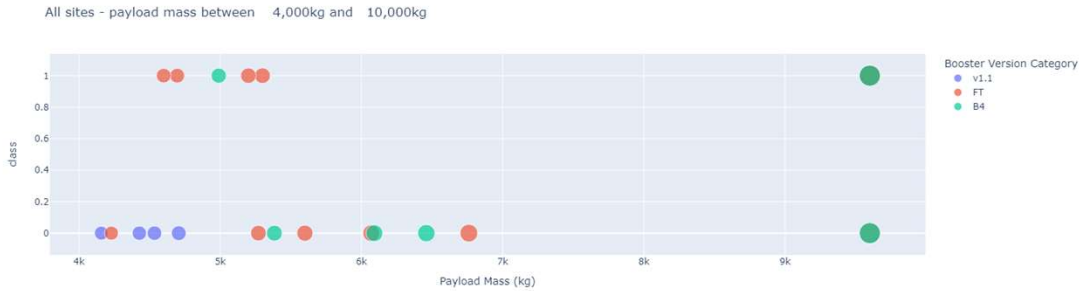
Pie Chart to find launch site with highest success ratio

From our analysis we can say that KSC LC-39A has highest success rate of 76.9%.

Scatter plot of launch outcomes with relation to Payload



Payload vs launch outcome for lower payloads (<4000kg)

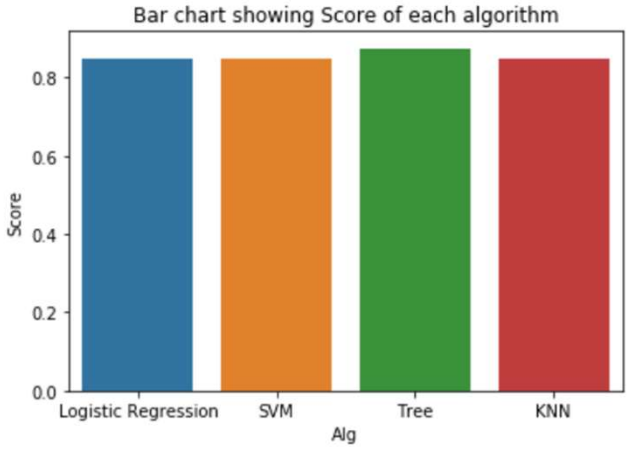


Payload vs launch outcome for higher payloads (>4000kg)

From the visualizations we can say that success rate is higher for lower payloads when compared with higher payloads.

Section 5

Predictive Analysis (Classification)



Classification Accuracy using Training Data

We can conclude that Decision Tree classifier has the best score among the 4 models with a score of 0.875 for the training data.

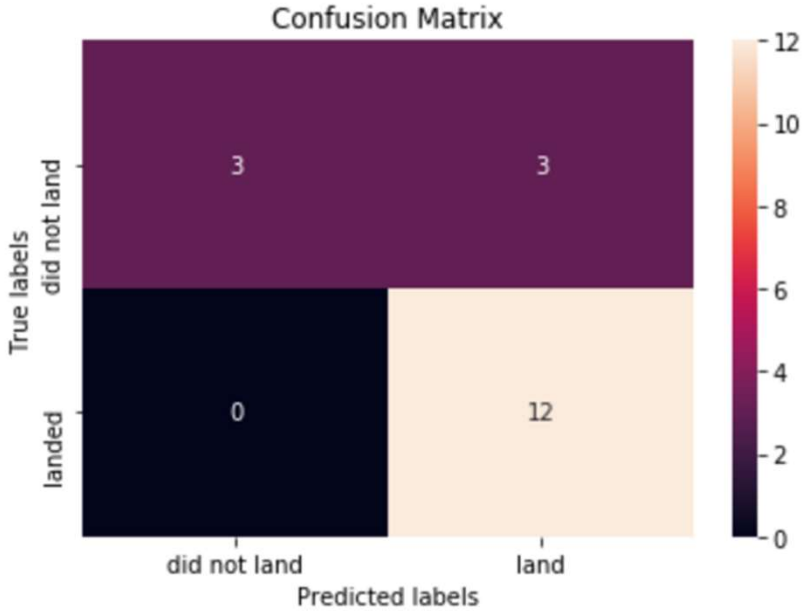
```
]: model = {'Logistic Regression':logreg_cv.best_score_,
           'SVM':svm_cv.best_score_,
           'Decision Tree':tree_cv.best_score_,
           'KNN':knn_cv.best_score_}
bestmodel = max(model, key=model.get)
print('Best Model is: ', bestmodel,' with a score of', model[bestmodel])
if bestmodel == 'Logistic Regression':
    print('Best params is:', logreg_cv.best_params_,)
if bestmodel == 'SVM':
    print('Best params is:', svm_cv.best_params_,)
if bestmodel == 'Decision Tree':
    print('Best params is:', tree_cv.best_params_,)
if bestmodel == 'KNN':
    print('Best params is:', knn_cv.best_params_,)
```

```
Best Model is:  Decision Tree  with a score of 0.875
Best params is: {'criterion': 'gini', 'max_depth': 14, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 10,
'splitter': 'random'}
```

The decision tree classifier has a score of 0.833 for the testing data.

```
tree_cv.score(X_test,Y_test)
```

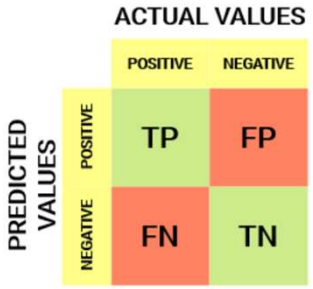
```
0.8333333333333334
```



Confusion Matrix

The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes.

The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.



Conclusions

The decision tree classifier is the best Machine learning algorithm for the data provided as it has the higher train data score than other classifiers.

From the analysis we can say that Launch sites are near coastlines rather than cities and highways.

Lower payloads have higher success rates when compared to heavier payloads.

KSC LC-39A launch site has higher success rate when compared to other launch sites.

Launch success rate started to increase from year 2013 till 2020.

Thank you!