IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Lucas Houlmann
2022/02/16

Outline

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

Executive Summary

Methodologies Used	**Results**
Data collection methodology	A combination of Web scraping and API requests provided enough data to built a ML model
Data wrangling	Basic EDA on raw data and cleansing were used to get a final dataset to feed the ML model
Exploratory data analysis (EDA) using visualization and SQL	Gave insights such as average payload mass carried by booster, List of total number of successful and failure mission outcomes, the names of the unique launch sites in the space mission, etc ...
Interactive visual analytics using Folium and Plotly Dash	Gave an interactive way of looking at SpaceX historical launches datas via interactive map and dashboard and could answers keys insights in regards to pricing strategy such as preferred locations for launching sites and preferred rocket features
Predictive analysis using classification models	Determined that Decision tree was the best predictive model with an accuracy rate between 85 - 89% for predicting rocket launch outcomes

Introduction

- **Project background and context**

  Taking the role of a data scientist working for a new rocket company called Space Y that would like to compete with SpaceX, the purpose of the project is to gather rocket launched data from Space X, create a predictive model for SpaceX rocket landing outcome and interactive Dashboard that could be used and support the pricing strategy for Space Y.

- **Problems you want to find answers**

  Instead of using rocket science, can a machine learning model be trained to determine the probability of reuse of SpaceX first stage rockets which, as a result, could help determining the price of each launch for Space Y?

Section 1

Methodology

Methodology

- Data collection methodology:
  - Request and parse the SpaceX API launch data using GET request
  - Web scraping to collect Falcon 9 historical launch records from Wikipedia
- Perform data wrangling
  - Basic Exploratory Data Analysis (EDA)
  - Determine Training Labels
- 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

- Data collection methodology:
  - Request and parse the SpaceX API launch data using GET request

1. Import Libraries and Define Auxiliary Functions
2. Request and parse the SpaceX launch data
3. Turn JSON response into Pandas DataFrame
4. Some Data Wrangling
5. Apply functions to get needed features names and create a new Dataframe out of it

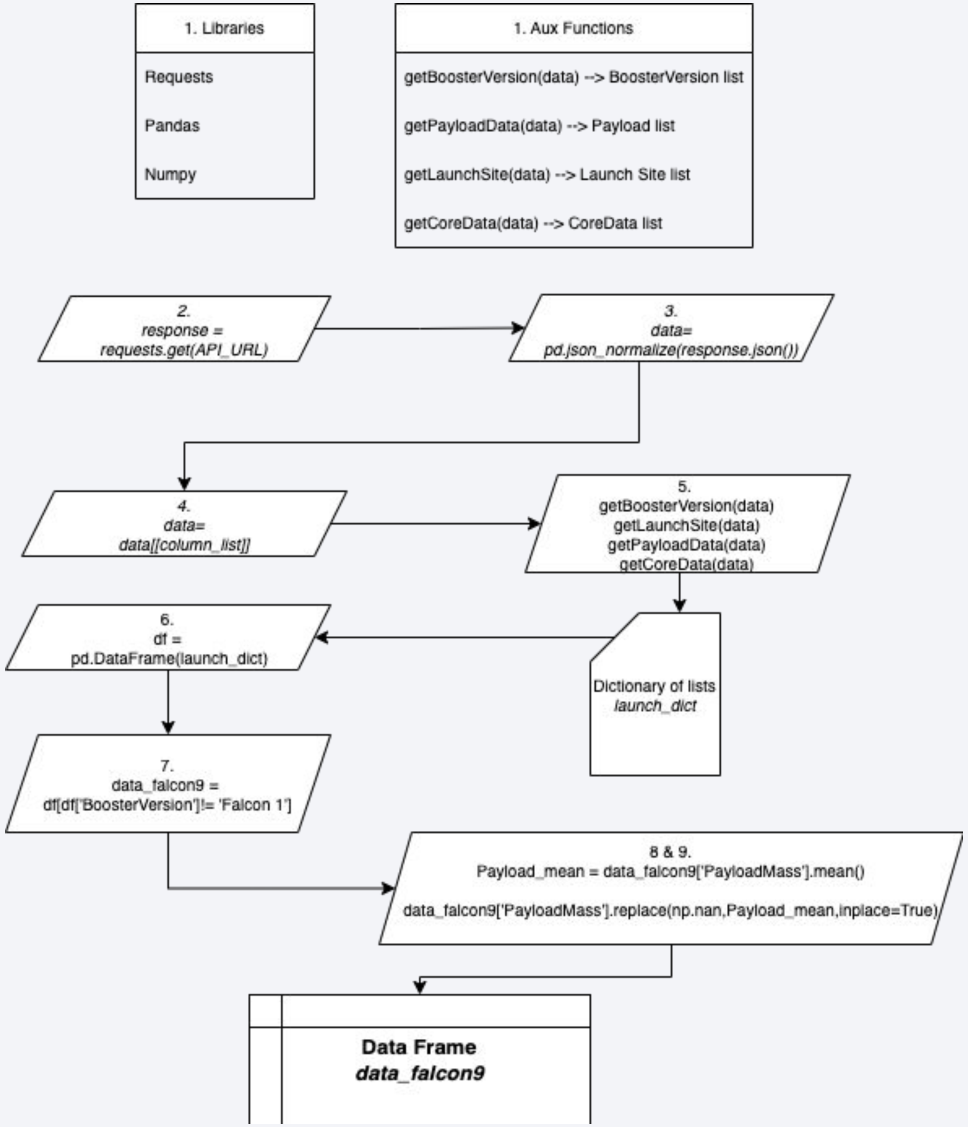
  - Web scraping to collect Falcon 9 historical launch records from Wikipedia

1. Import Libraries and Define Auxiliary Functions
2. Request and parse the SpaceX launch data
3. Turn HTML response into BeautifulSoup Object
4. Collect all relevant column names from the HTML table header
5. Create a data frame by parsing the launch HTML tables

Data Collection – SpaceX API

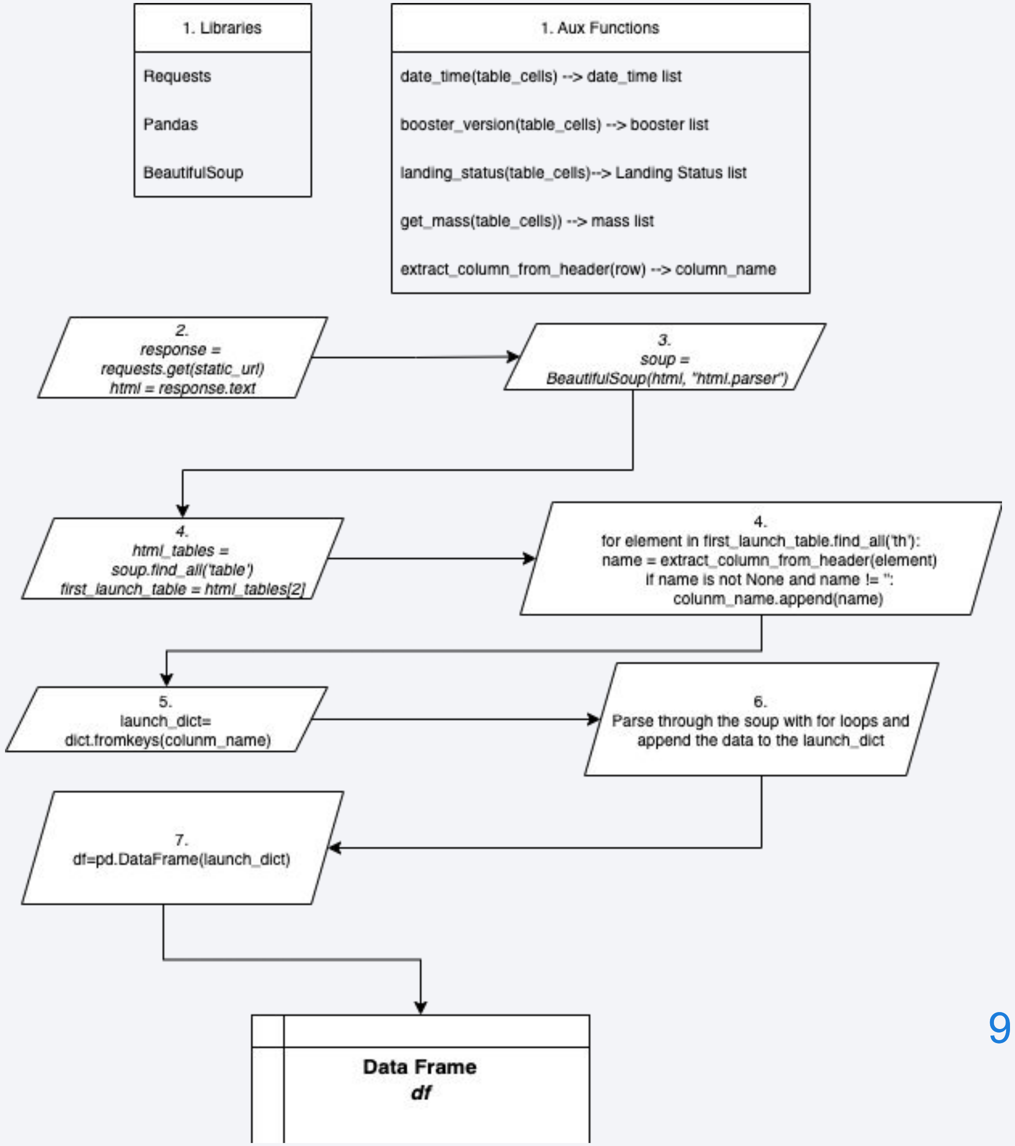
1. Import Libraries and Define Auxiliary Functions
2. Requesting rocket launch data from SpaceX API
3. Decode the response content as Json
4. Filter data to *rocket, payloads, launchpad, and cores* using the IDs given for each launch
5. Apply *Aux Functions* to get the name for each features stored in lists
6. Create a dataframe from launch_dict
7. Filter the dataframe to only include Falcon 9 launches
8. Deal with empty datas:
    - Calculate the mean value of *PayloadMass* column
    - Replace the *np.nan* values with its mean value

[GitHub link]



Data Collection - Scraping

1. Import Libraries and Define Auxiliary Functions
2. Request the Falcon9 Launch Wiki page from its URL
3. Create a BeautifulSoup object from the HTML response
4. Extract all column/variable names from the HTML table header
5. 6. & 7. Create a data frame by parsing the launch HTML tables



[GitHub link]

Data Wrangling

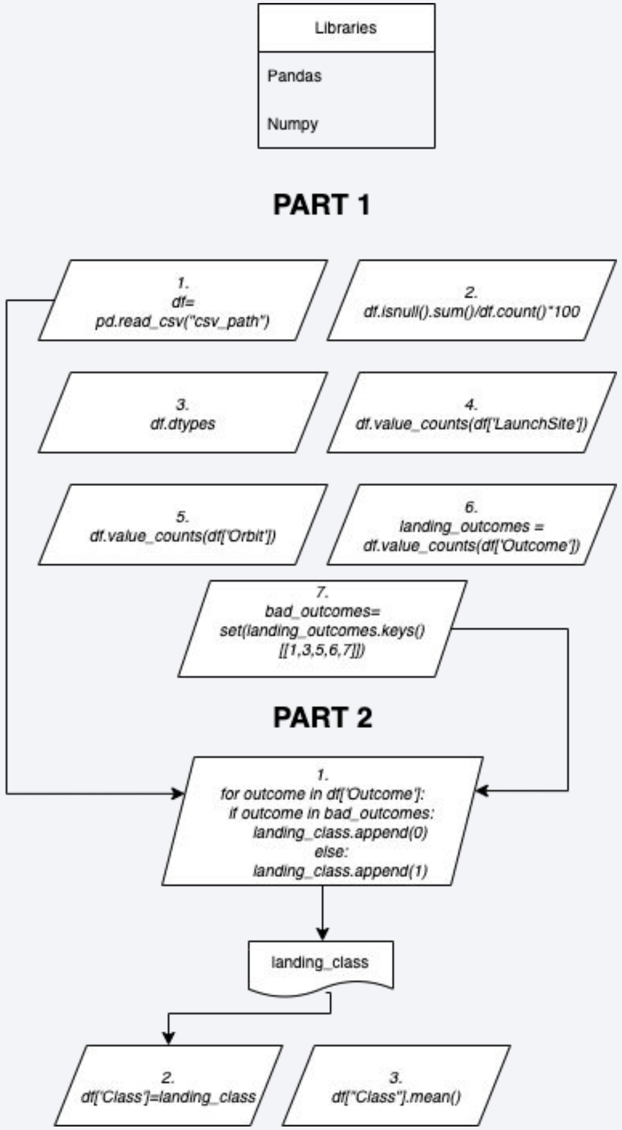
PART 1: Basic Exploratory Data Analysis (EDA):

1. Load Space X dataset, from last section
2. Identify and calculate the percentage of the missing values in each attribute
3. Identify which columns are numerical and categorical
4. Calculate the number of launches on each site
5. Calculate the number and occurrence of each orbit
6. Calculate the number and occurence of mission outcome per orbit type
7. Create a set of outcomes where the second stage did not land successfully

PART 2: Determine Training Labels

1. Create a landing outcome label from Outcome column with 1 (Success) and 0 (Fail)
2. Append the created column to the dataframe
3. Determine the success rate

[GitHub link]



EDA with Data Visualization

Chart Used	Purposes
Scatter plot	• Visualize the relationship between *Flight Number* and *Launch Site* • Visualize the relationship between *Flight Number* and *Payload mass* • Visualize the relationship between *Payload mass* and *Launch Site* • Visualize the relationship between *Flight Number* and *Orbit type* • Visualize the relationship between *Payload mass* and *Orbit type*
Bar chart	• Visualize the relationship between *success rate* of each *orbit type*
Line plot	• Visualize the launch *success* yearly trend

GitHub link

EDA with SQL 1/2

SQL Queries Used	Purpose
SELECT DISTINCT launch_site FROM SPACEXTBL	Display the names of the unique launch sites in the space mission
SELECT launch_site FROM SPACEXTBL where launch_site LIKE 'CCA%' LIMIT 5	Display 5 records where launch sites begin with the string 'CCA'
SELECT SUM(payload_mass__kg_) as total_payload_mass FROM SPACEXTBL where customer LIKE 'NASA (CRS)%'	Display the total payload mass carried by boosters launched by NASA (CRS)
SELECT AVG(payload_mass__kg_) as AVG_payload_mass FROM SPACEXTBL where booster_version LIKE '%F9 v1.1%'	Display average payload mass carried by booster version F9 v1.1
SELECT MIN(DATE) as date FROM SPACEXTBL WHERE landing__outcome LIKE '%Success (ground pad)%'	List the date when the first successful landing outcome in ground pad was acheived.
SELECT booster_version, landing__outcome FROM SPACEXTBL WHERE landing__outcome LIKE '%Success (drone ship)%' AND payload_mass__kg_ 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
SELECT mission_outcome, COUNT(mission_outcome)as count FROM SPACEXTBL GROUP BY mission_outcome	List the total number of successful and failure mission outcomes

[GitHub link]

EDA with SQL 2/2

SQL Queries Used	Purpose
SELECT booster_version FROM SPACEXTBL where payload_mass__kg_ = (SELECT MAX(payload_mass__kg_) FROM SPACEXTBL)	List the names of the booster_versions which have carried the maximum payload mass
SELECT landing__outcome, booster_version, launch_site FROM SPACEXTBL WHERE landing__outcome LIKE '%Failure (drone ship)%' AND DATE BETWEEN '2015-01-01' AND '2015-12-31'	List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
SELECT landing__outcome, COUNT(landing__outcome) as count FROM SPACEXTBL WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY landing__outcome ORDER BY count desc	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

[GitHub link]

Build an Interactive Map with Folium

Map Object Used	**Purpose**
folium.Map(location_coor))	Create a basic map centered on location coordinates provided as argument
folium.Circle	Add a highlighted circle area with a text label on a specific coordinate
folium.Marker	• Create a circle at launchsite coordinate with a icon showing its name • Create and add a marker within clusters to the site map • Display the distance between coastline point and launch site using the icon property
folium.Popup	Add a text label on a specific coordinate
MarkerCluster()	Create a cluster mark for the success/failed launches for each site on the map
MousePosition()	Add Mouse Position to get the coordinate (Lat, Long) for a mouse over on the map
PolyLine()	Draw a line between a launch site to the selected coastline point

[GitHub link]

Build a Dashboard with Plotly Dash

Plots/graphs and interactions Used	Purpose
Dropdown menu	Select Launch Site to be be used for the analysis and visuals
Call back function on get_pie_chart function with Success rate pie chart as Output and Launch Site Dropdown selection as Input	Render success-pie-chart based on selected site dropdown
Range slider	Select Payload range to be used for the analysis and visuals
Call back function on get_scatter_chart function with Success rate by payload scatter plot chart as Output and Launch Site Dropdown selection and Payload range as Input	Render the success-payload-scatter-chart scatter plot

[GitHub link]

Predictive Analysis (Classification)

Development process	Description
Building methods	1. Create a NumPy array for the column to be the target (i.e: *Class)* and store it in variable Y 2. Standardize all the parameters to fit the model and store them in a variable X 3. Split the data into training and testing data
Evaluation methods	1. Create a object for each selected model *(i.e : Logistic Reg, SVM, Decision Tree, KNN)* 2. Fit the trained datas in the model 3. Calculate the accuracy of the model with the test datas (*i.e: best_score_ attribute*) 4. Plot the confusion matrix using the predicted datas and test datas
Improvements methods	Search and select for the best parameters for each model using for loops on possible parameters options
Best performance model finding methods	Calculate Jaccard, F1, LogLoss for each model and select the model with the highest accuracy average

GitHub link

Results - Exploratory data analysis with SQL 1/2

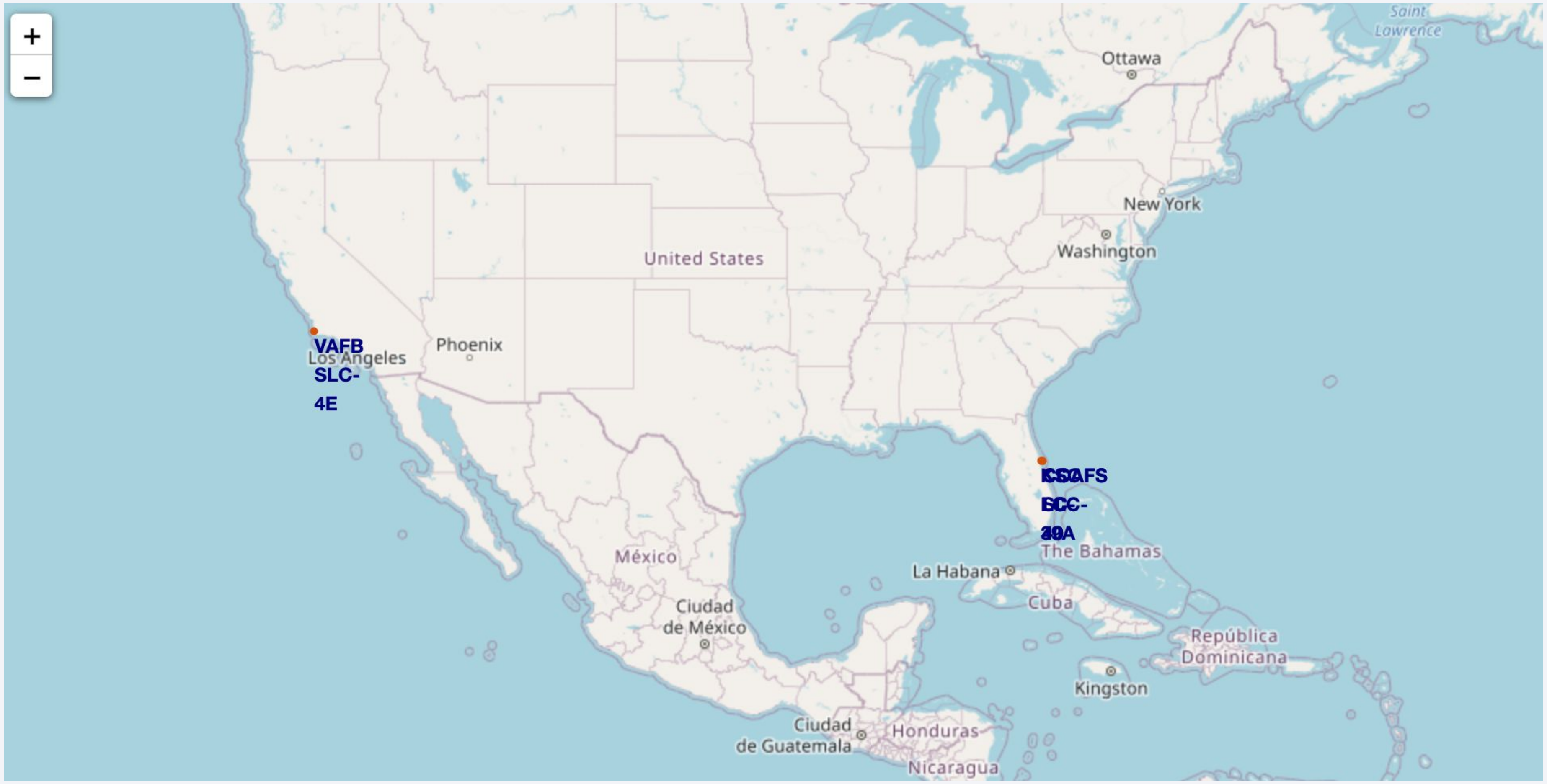
Items	Results
Display the names of the unique launch sites in the space mission	CCAFS LC-40, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E
Display 5 records where launch sites begin with the string 'CCA'	CCAFS LC-40, CCAFS LC-40, CCAFS LC-40, CCAFS LC-40, CCAFS LC-40
Display the total payload mass carried by boosters launched by NASA (CRS)	48213
Display average payload mass carried by booster version F9 v1.1	2534
List the date when the first successful landing outcome in ground pad was achieved.	2015-12-22

Results - Exploratory data analysis with SQL 2/2

Items	Results
List the names of the booster_versions which have carried the maximum payload mass	F9 FT B1022 Success (drone ship) F9 FT B1026 Success (drone ship) F9 FT B1021.2 Success (drone ship) F9 FT B1031.2 Success (drone ship)
List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015	Failure (drone ship) F9 v1.1 B1012CCAFS LC-40 Failure (drone ship) F9 v1.1 B1015CCAFS LC-40
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	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
List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000	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
List the total number of successful and failure mission outcomes	Failure (in flight) 1 Success 99 Success (payload status unclear) 1

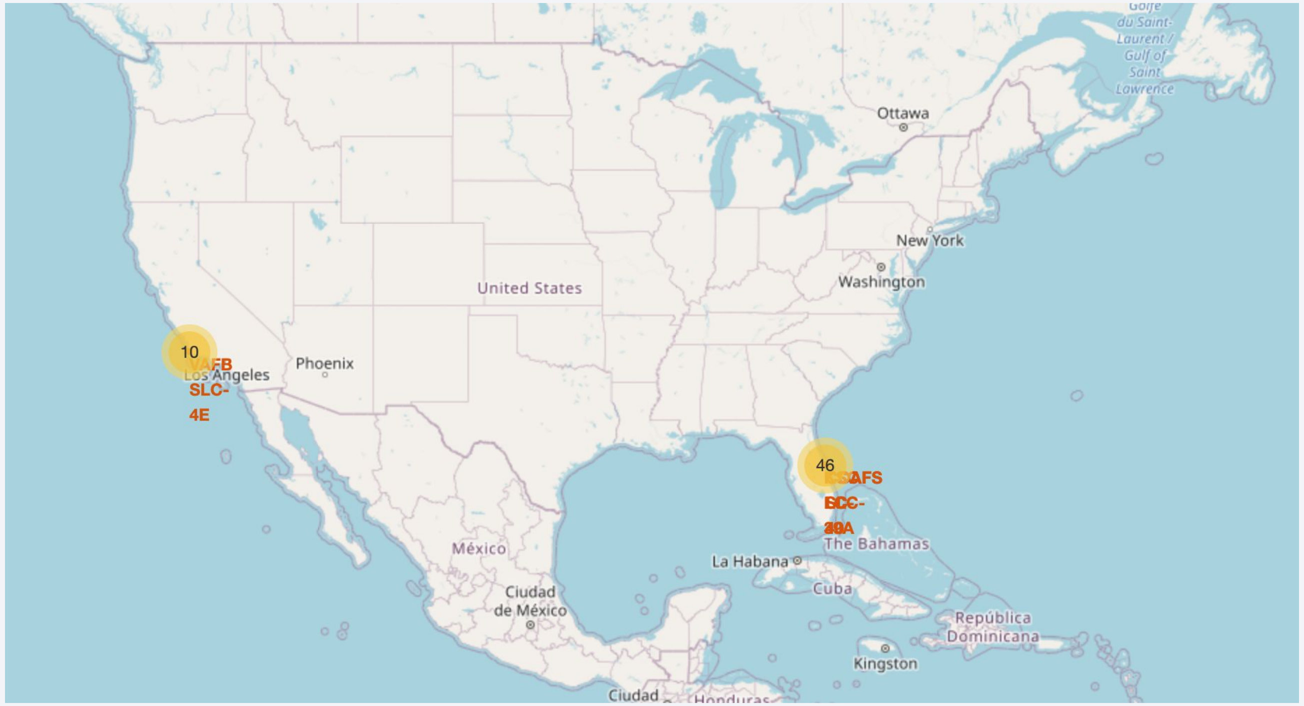
Results - Interactive analytics map in screenshots

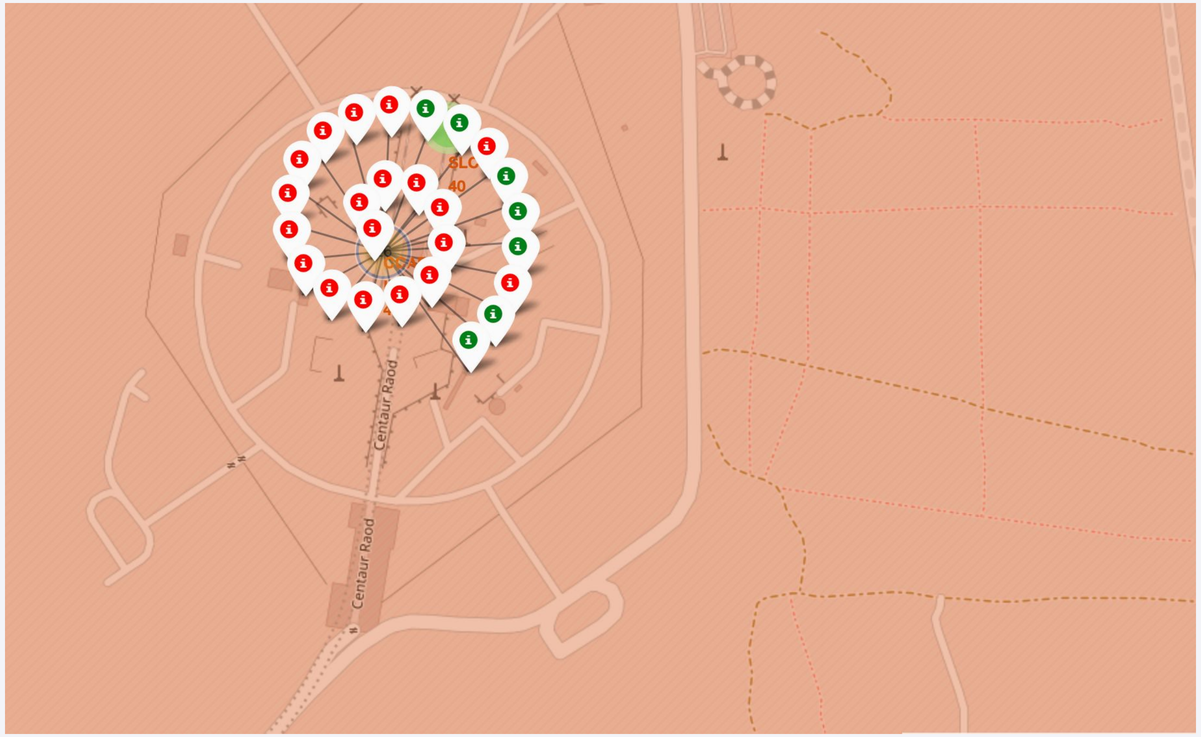
Screenshots of the Interactive Map with Markers for each Launch Site



Results - Interactive analytics map in screenshots

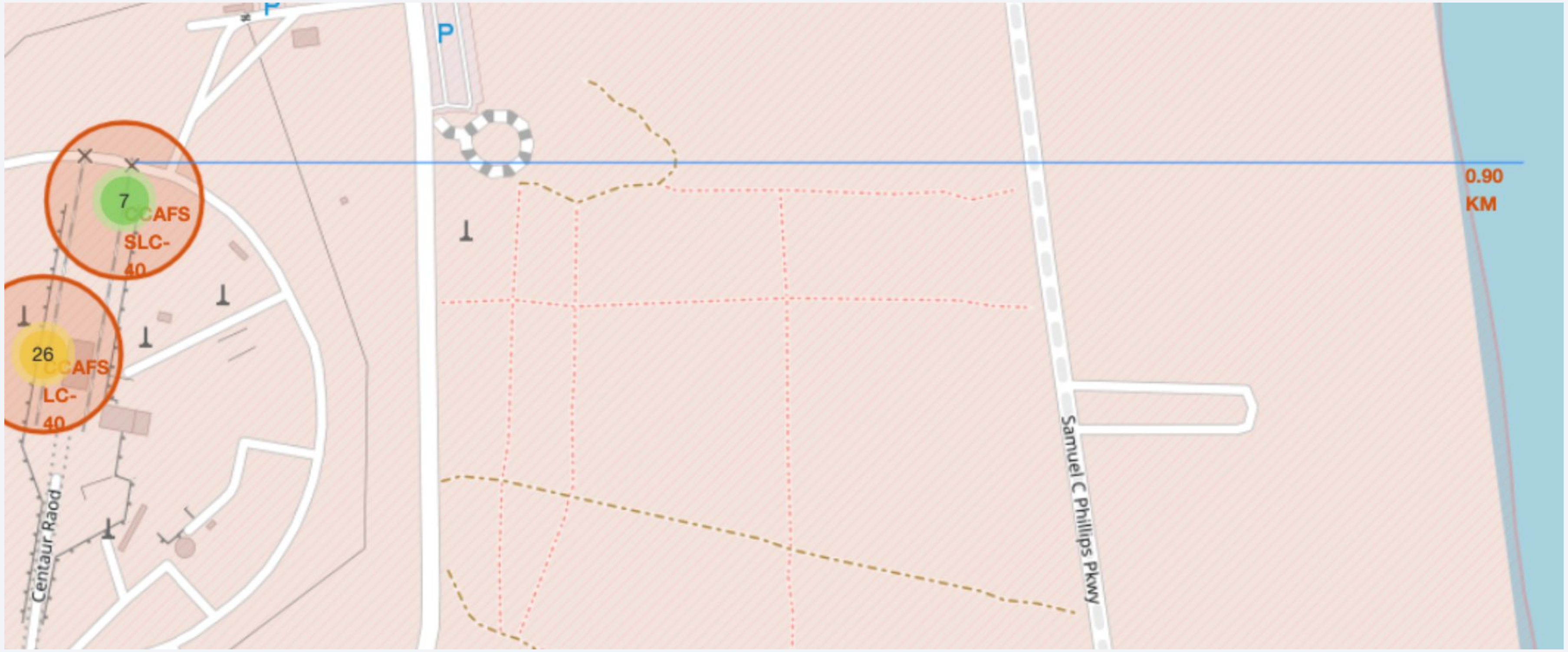
Screenshots of the Interactive Map with Clusters for each Launch Site and colored markers for each launch outcome at each Launch Site





Results - Interactive analytics map in screenshots

Screenshots of the Interactive Map with Clusters for each Launch Site, colored markers for each launch outcome and line showing the distances from launch site to proximity (here: coastline)



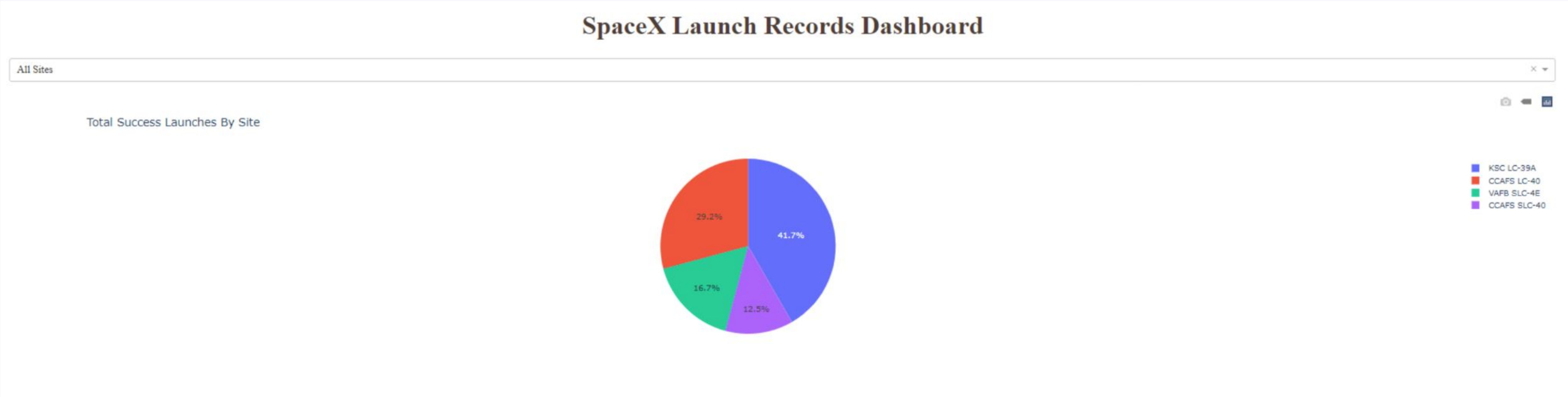
Results - Interactive analytics map

- For Logistic purposes and ease access, all launch sites are close to highways, raintracks and coastline and far from major cities and the Equator Line.

- However, specific close proximities doesn’t seem to play a major role in defining the success rate of a launch. The result showed no clear pattern between distances of proximities and success rates for launches

Results - Interactive analytics DashBoard in screenshots

Dashboard with launch sites dropdown menu and Success rate Pie Chart



Results - Interactive analytics DashBoard in screenshots

Dashboard with range selector and Correlation scatter plot

Results - Interactive analytics DashBoard

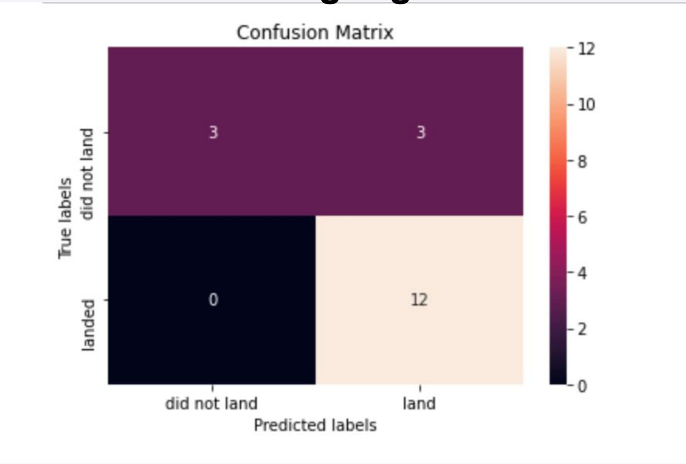
- KSC LC-39A is the site with the most successful launches and highest success rate
- Payload between 2490 - 5300 kg has the highest launch success rate
- Payload between 5600 - 700 kg has the lowest launch success rate
- Booster version FT has the highest launch success rate

Results - Predictive Analysis

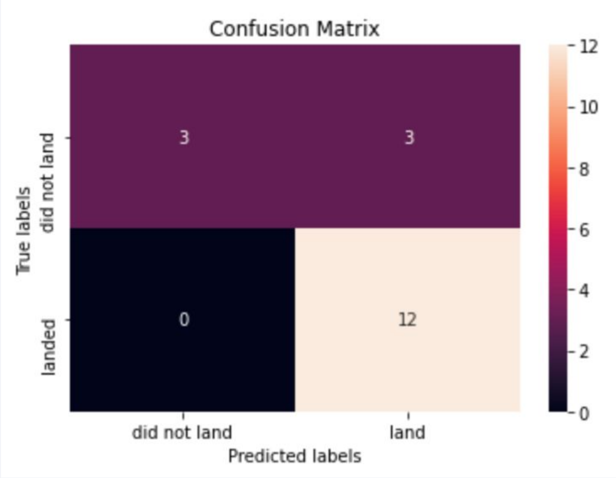
- **Both Confusion Matrix and accuracy tests showed that Decision Tree is the best predictive model for the project**

	Jaccard	F1	LogLoss
Logistic Regression	0.800000	0.814815	0.478667
SVM	0.800000	0.814815	NA
Decision Tree	0.846154	0.888889	NA
KNN	0.800000	0.814815	NA

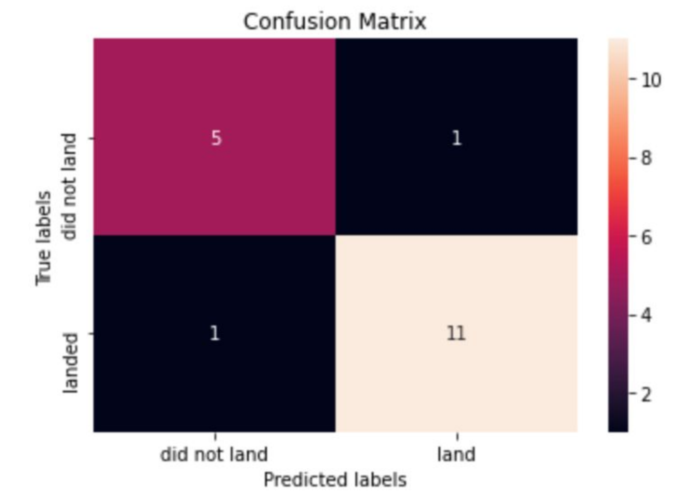
LogReg



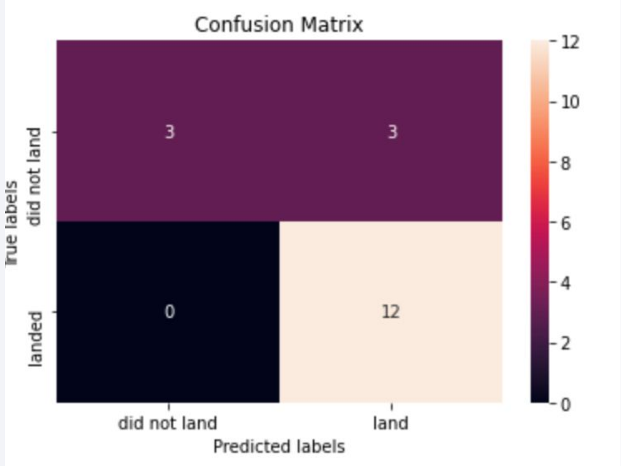
SVM



Decision Tree



KNN



Conclusions

If SpaceY is looking to compete with SpaceX, the following could be considered when determining their pricing strategy:

- For logistic purpose and easy access, the company should build their launch site(s) close to highways, raintracks and coastline and far from major cities and the Equator Line.
- The success launch rates based on SpaceX data suggest that Pacific Coast should be preferred
- Rocket payload between 2490 - 5300 kg should be preferred
- Rocket booster version FT should be preferred
- Decision Tree model should be used for predicting launching success rate

Thank you!