IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Espen
26.12.2021

Executive Summary

Different approaches, methods and analysis have been used in this study to find the most important factors for the success rate for launches. API and web scraping were used to collect information about recent launches. Exploratory data analysis (EDA) were completed with Python and DB2.

Different machine learnings tools/algorithms were used to build models to predict outcomes (success/failure) of launches. Such models can be used to predict outcomes of future launches.

The analysis show that some of the most important factors for successful launches are:

- The Payload
- The orbit
- The launch site
- Experience (flight number)

Introduction

PROJECT BACKGROUND:

The last decade new possibilities for sending manned missions to space has increased in an incredible manner. This trend will probably continue the next year and results in affordable space trips for everyone. Still sending manned missions to Space is a risky and quite complicated process. Perhaps the most successful competitor today is SpaceX. This company has managed to reuse some of the parts of the rocket. This part is called “the first stage”. The first stage is doing most of the work to send the rocket into space and is expensive to manufacture. SpaceX have 50 % or less cost compared to their competitors. The first stage is quite large and expensive. A crucial factor for SpaceX ‘s success is that the first stage lan ds undamaged after a mission.

SpaceY want to compete with SpaceX. Like SpaceX we will reuse the first stage. To success we have to analyze and find which factors that affect the landings of the first stages. In this study we have collected different information about many of SpaceX’s rocket launches and if the landing of first stages where successful or not. Different machine learning algorithms (models) to predict the success of the first stages.

The main goal in this project is to find the most important factors that affect the landings of the first stages. SpaceY will strive to use this factor in their rocket launches. SpaceY will be a significant participant in the commercial space industri if they manage to reuse the first stages.

Section 1

Methodology

Data Collection

To different approaches were used to collect data for the analysis of the SpaceX launches and the success factors for the landing of first stages:

1) Requesting data from the SpaceX REST API (information about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome).

2) Webscraping Wiki pages with information about SpaceX's launches

Different Python libraries/packages were used to do the api calls, the webscraping and the following data wrangling

Data Collection – SpaceX API

The most important python libraries used to collect and prepare SpaceX API data:

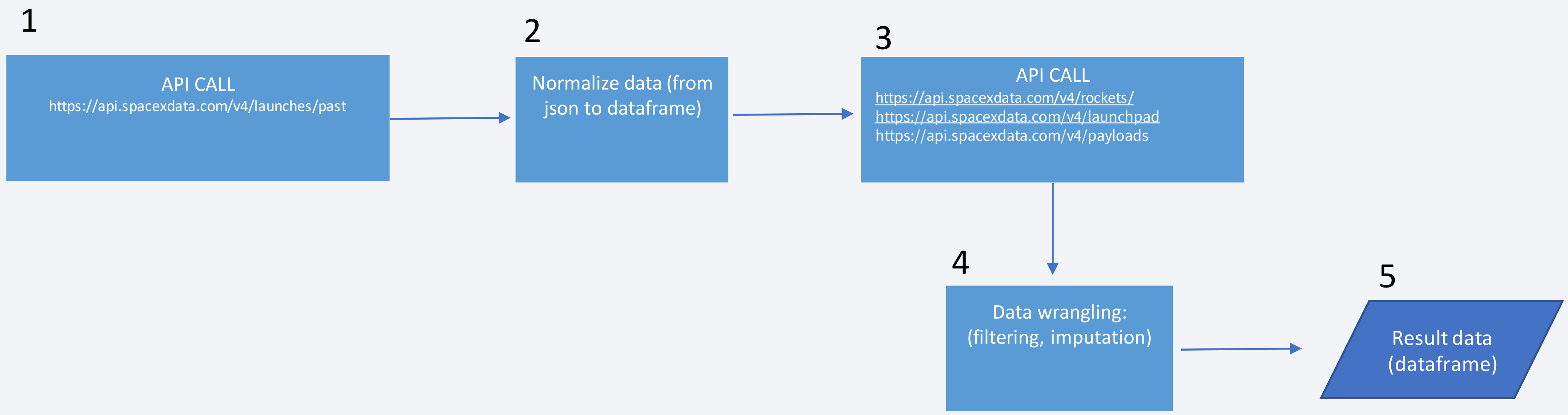
- requests
- pandas
- numpy
- beautifulsoup4

Addresses (urls) to the spaceX API:

- https://api.spacexdata.com/v4/launches/past
- https://api.spacexdata.com/v4/rockets/
- https://api.spacexdata.com/v4/launchpad
- https://api.spacexdata.com/v4/payloads

Data Collection – SpaceX API

Flowchart of the most important steps in the SpaceX API calls



1

API CALL
https://api.spacexdata.com/v4/launches/past

2

Normalize data (from json to dataframe)

3

API CALL
https://api.spacexdata.com/v4/rockets/
https://api.spacexdata.com/v4/launchpad
https://api.spacexdata.com/v4/payloads

4

Data wrangling:
(filtering, imputation)

5

Result data
(dataframe)

1) Requesting rocket launch data from SpaceX API
2) Normalize the respons data from 1
3) 3 new api-calls to get information about booster name, launch siste og payload
4) Filtering the data to only include Falcon 9 launches and replace missing payload values with mean value for payload)
5) The result data (will be used in the forthcoming analysis phase)

GitHub URL to the notebook for Data collection with API :notebook

Data Collection – Web scraping from Wikipedia

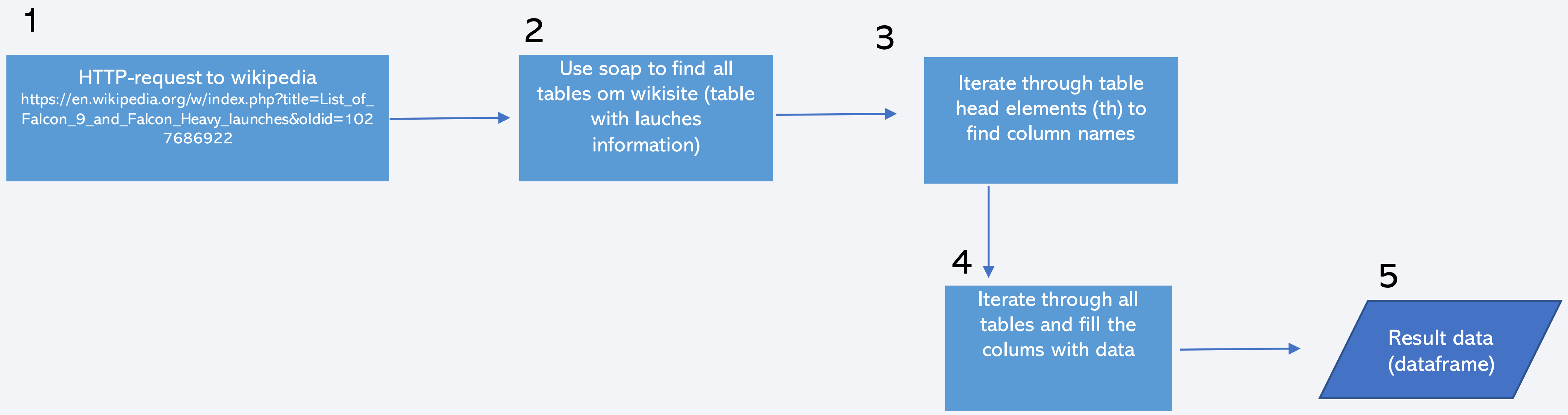
The most important python libraries used to collect and prepare data scraping from Wikipedia:

- requests
- pandas
- numpy
- beautifulsoup4

Addresses (urls) to the Wikipedia site with List of Falcon 9 and Falcon Heavy launches:
https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922

Data Collection – Web scraping from Wikipedia

Flowchart of the most important steps in the web scraping process


1
HTTP-request to wikipedia
https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
2
Use soap to find all tables om wikisite (table with lauches information)
3
Iterate through table head elements (th) to find column names
4
Iterate through all tables and fill the colums with data
5
Result data (dataframe)

1) HTTP-request to Wikipedia site
2) Use the wep scraping package SOAP to extra relevant information (tables) from Wikipedia site *(soupresponse = BeautifulSoup(response.content, "html.parser")*

3) Find the column names with launch information from the tables
4) Iterate through the columns and fill the extracted columns with data
5) The result data (will be used in the forthcoming analysis phase)

GitHub URL to the notebook for Data collection with Web scraping :notebook

Data Wrangling

The most important python libraries used in the Data Wranling phase :

- pandas
- numpy

Main goal with the Data Wrangling phase:

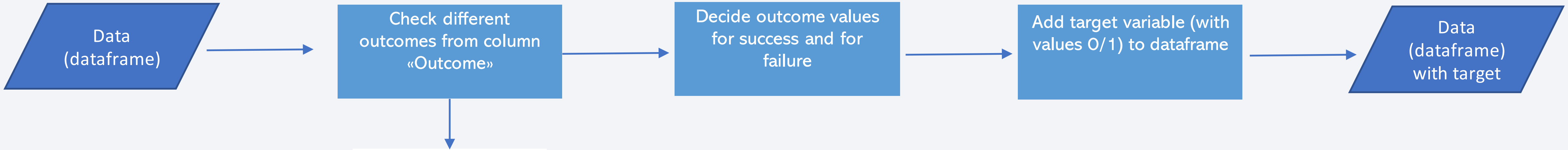
1) Explore occurrences of missing values
2) Check number of landing launches on different sites
3) Check occurrence on different orbits
4) Create target variable (landing outcome 1=success, 2=fail)

The tasks in the first three bullet points give us a good overview of the distribution and quality of the data. Different methods from the Pandas library are used here.

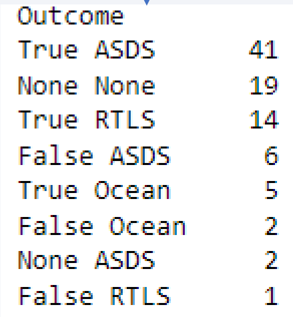
The column **outcome** is used to create labels for the target column.

Data Wrangling

Create a landing outcome label from Outcome column (the target column)



Data (dataframe)
Check different outcomes from column «Outcome»
Decide outcome values for success and for failure
Add target variable (with values 0/1) to dataframe
Data (dataframe) with target



```
Outcome
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
```

Different outcomes:

Success (1):
«True ASDS»
«True RTLS»
«True Ocean»

Failure (0):
«None None»
«False ASDS»
«False Ocean»
«None ASDS»
«False RTLS»

The success rate of the launces in the data: 0.6666

GitHub URL to the notebook for Data wrangling :notebook

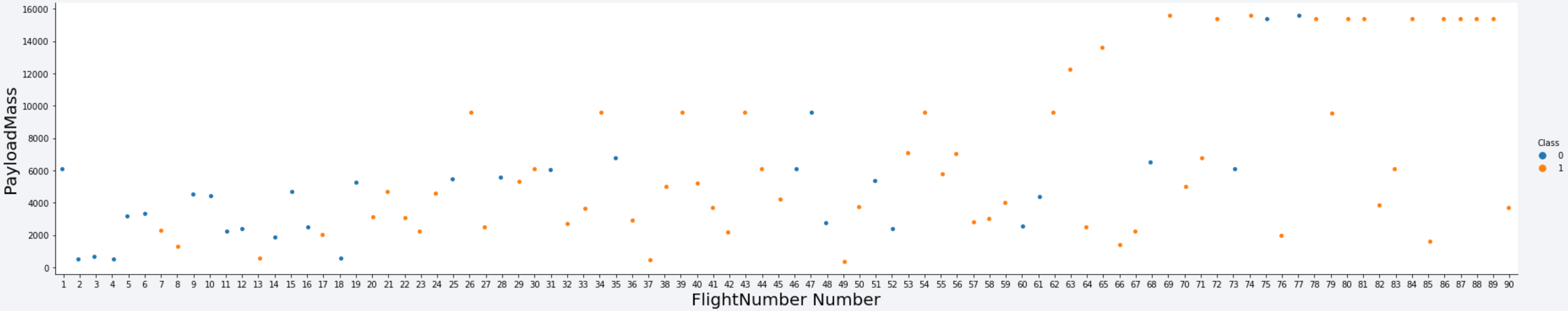
EDA with Data Visualization

The following graphs/charts show different exploratory data analysis task completed in the project:

1

Scatterplot: showing success/failure for different payloads

This graph can give an idea if success rates increase/decrease with increasing flight number time(experience) and payload

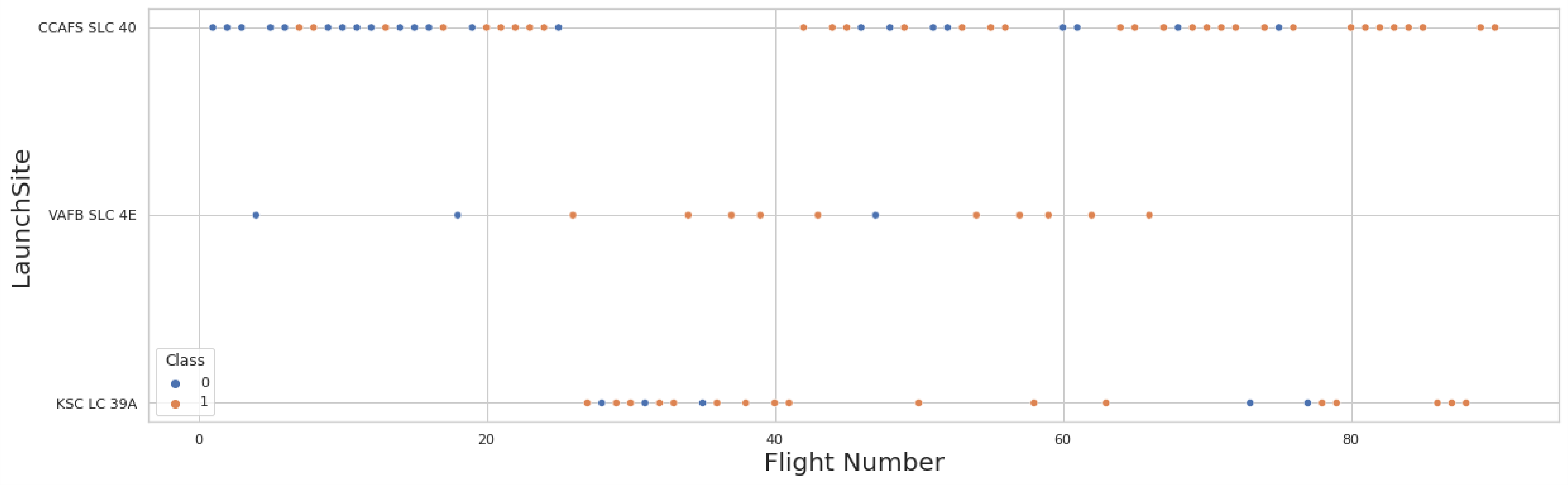


EDA with Data Visualization

2

Scatterplot: showing success/failure for the different flight numbers/launch site

This graph can give an idea if success rates increase with flight number (increased time(experience) for the different launch sites

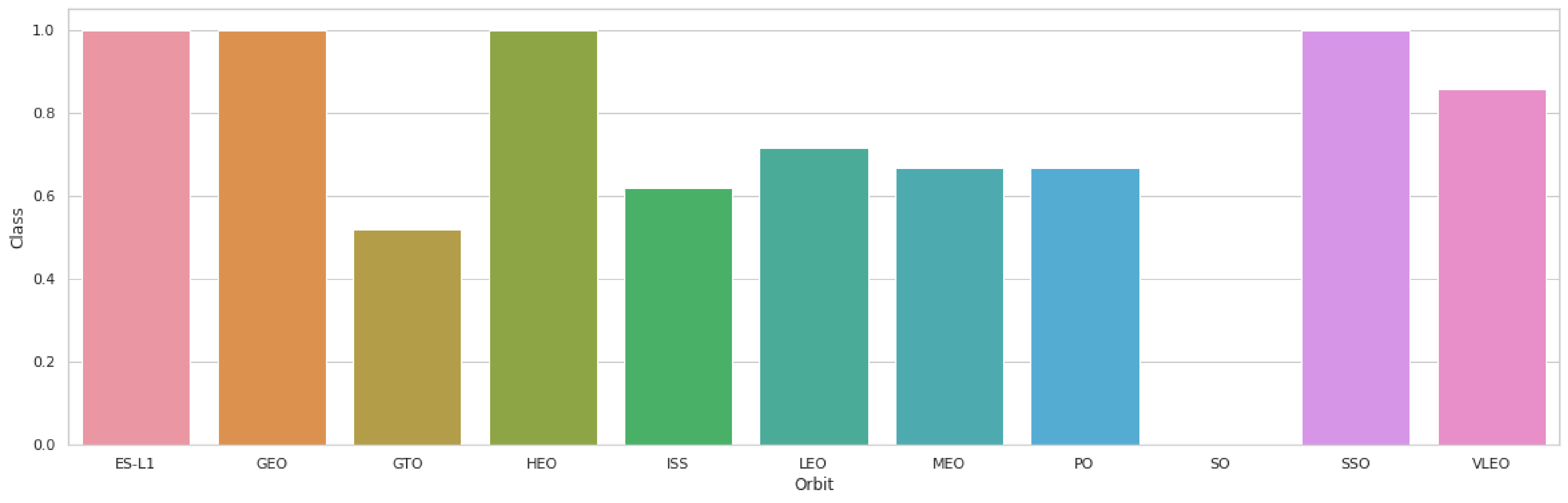


EDA with Data Visualization

3

Bar chart: showing successrate for the different orbits

This graph can give an idea of success rates for different orbits

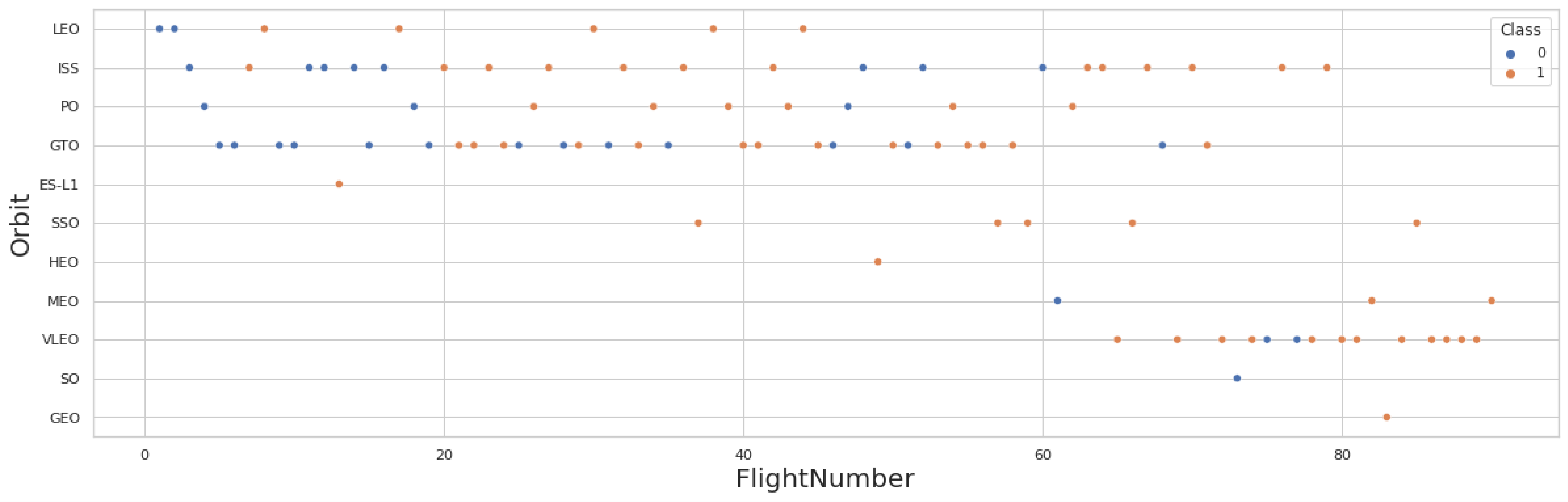


EDA with Data Visualization

4

Scatter plot: showing FlightNumber, different orbits and success rate

This graph can give an idea of use of different orbits over time and their success rate

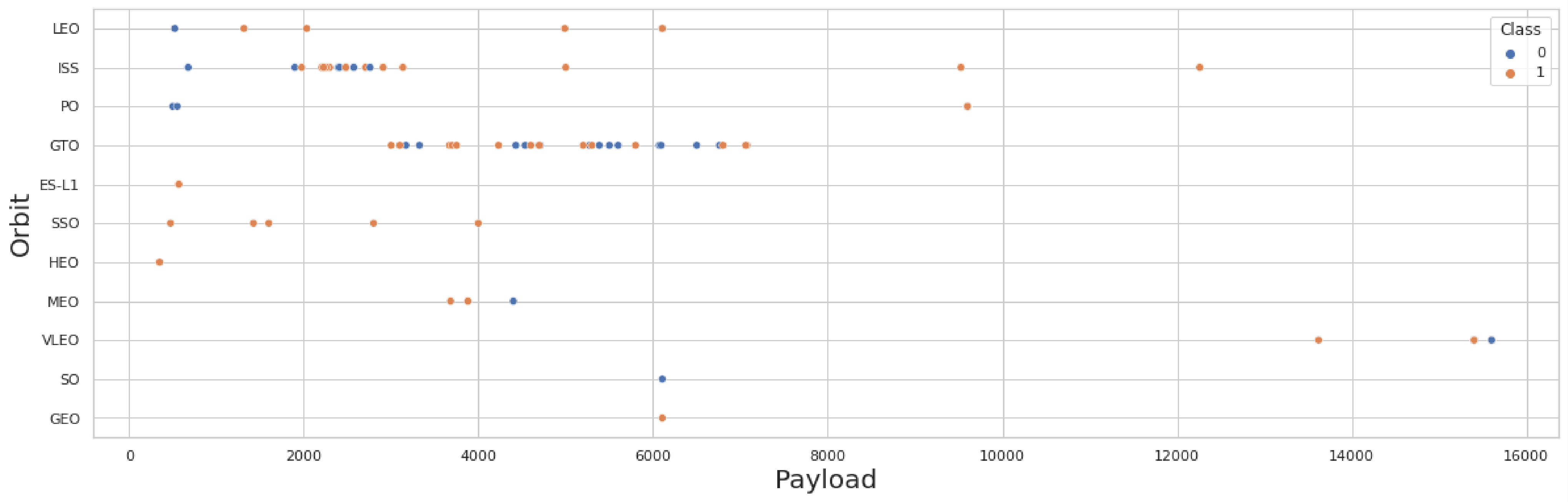


EDA with Data Visualization

5

Scatter plot: showing payload for different orbits and success rate

This graph can give an idea if payload has an impact on success rate for the different orbits

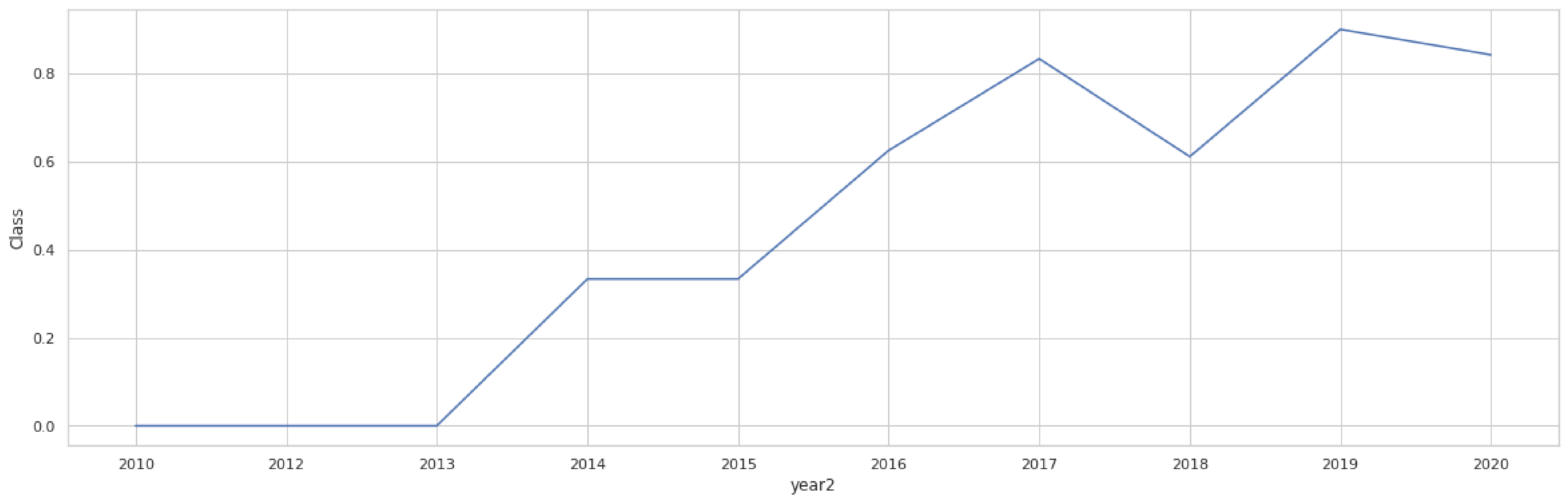


EDA with Data Visualization

6

Line plot: showing success rate with increasing year:

This graph can give an idea if the total success rate is increasing with exerience (years)



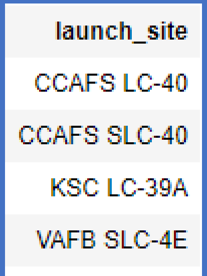
GitHub URL to the notebook for Data wrangling :notebook

EDA with SQL

Several analysis were done by querying data base tables in a Db2 database. The following slides shows different queries and the results:

QUERY 1 : Show the different launch sites

```
%sql select distinct(LAUNCH_SITE) from SPACEXDATASET
```

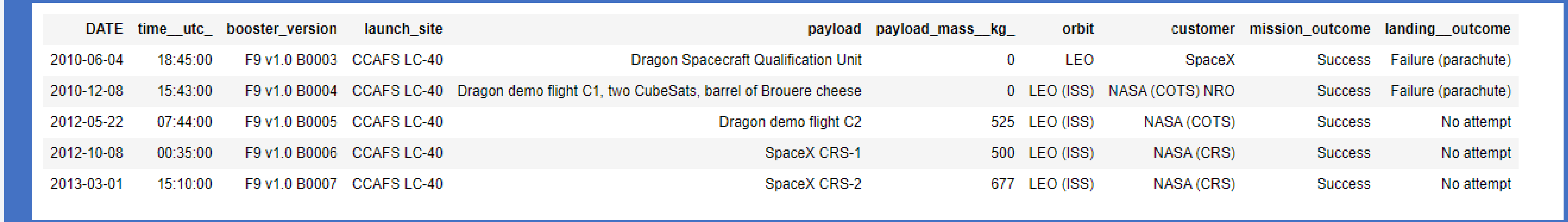


launch_site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E

EDA with SQL

QUERY 2 : Show 5 records for launch sites beginning with the string ‘CCA’

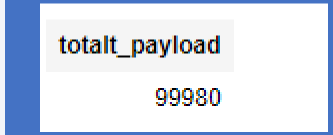
*%sql select * from SPACEXDATASET where SUBSTR(LAUNCH_SITE,1,3)='CCA' FETCH FIRST 5 ROWS ONLY*



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

QUERY 3 : Show total payload mass carried by boosters launched by NASA (CRS)

%sql select sum(PAYLOAD_MASS__KG_) as totalt_payload from SPACEXDATASET where substr(customer,1,4)='NASA'

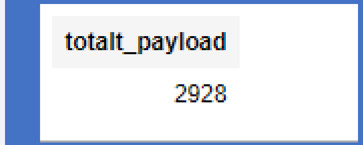


totalt_payload
99980

EDA with SQL

QUERY 4 : Show average payload mass carried by booster version F9 v1.1

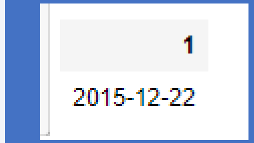
%sql select avg(PAYLOAD_MASS__KG_) as totalt_payload from SPACEXDATASET where BOOSTER_VERSION='F9 v1.1'



totalt_payload
2928

QUERY 5 : List the date when the first successful landing outcome in ground pad was achieved

%sql select min(DATE) from SPACEXDATASET where LANDING__OUTCOME='Success (ground pad)'

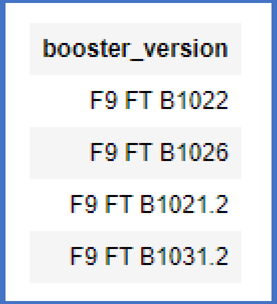


1
2015-12-22

EDA with SQL

QUERY 6 : Show the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

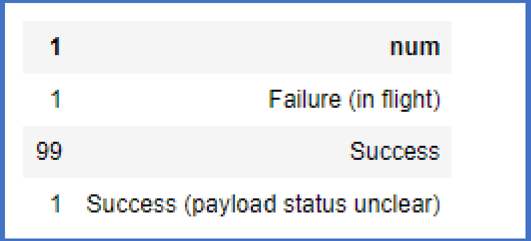
%sql select BOOSTER_VERSION from SPACEXDATASET where LANDING__OUTCOME='Success (drone ship)' and PAYLOAD_MASS__KG_>4000 and PAYLOAD_MASS__KG_<6000



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

QUERY 7: Show the total number of successful and failure mission outcomes

%sql select count(*),MISSION_OUTCOME as num from SPACEXDATASET group by MISSION_OUTCOME

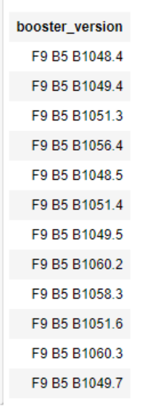


1	num
1	Failure (in flight)
99	Success
1	Success (payload status unclear)

EDA with SQL

QUERY 8 : Show the names of the booster_versions which have carried the maximum payload mass. Use a subquery

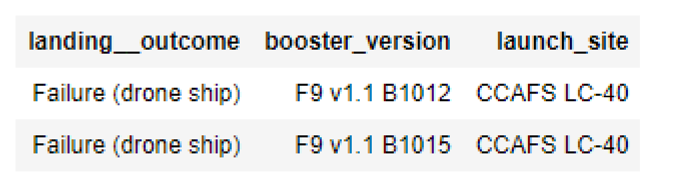
```
%sql select BOOSTER_VERSION from SPACEXDATASET where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXDATASET)
```



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

QUERY 9: Show the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%sql select LANDING__OUTCOME, BOOSTER_VERSION,LAUNCH_SITE from SPACEXDATASET where LANDING__OUTCOME='Failure (drone ship)' and
year(DATE)=2015
```

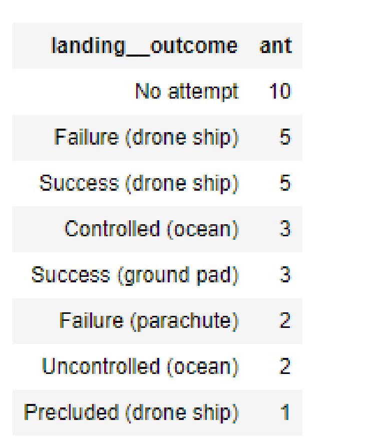


landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

EDA with SQL

QUERY 10 : Show the rank of 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

```
%sql select LANDING__OUTCOME,count(*) as ant from SPACEXDATASET where DATE>=TO_DATE('2010-06-04', 'YYYY-MM-DD') and
DATE<=TO_DATE('2017-03-20', 'YYYY-MM-DD')  group by LANDING__OUTCOME order by ant desc
```



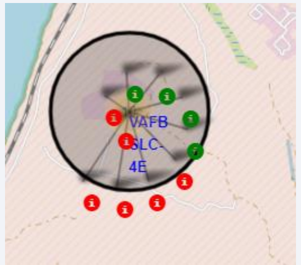
landing__outcome	ant
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

GitHub URL to the notebook for EDA with SQL :notebook

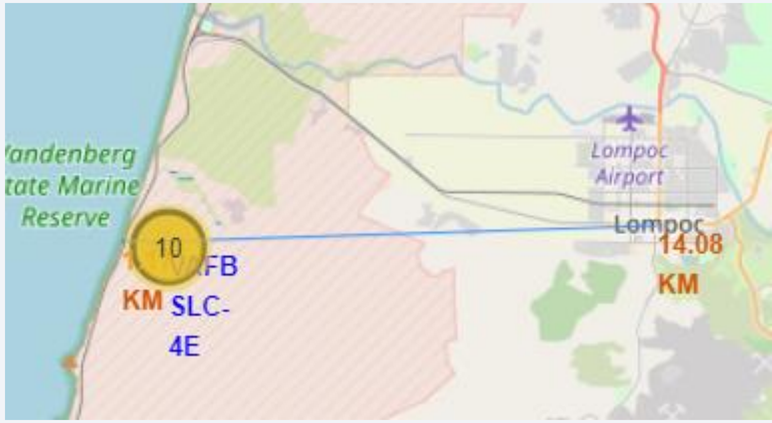
Build an Interactive Map with Folium

The Python package Folium was used for analyzing geographical and spatial data to check if the location of the launch site has impact on the success rate

The different launch sites were marked with a circle on the map and then icons with different color codes were included to give an indication of the distribution of success/failure launches at each sites



Distance to nearest railway, coastline, city and highway was calculated and marked on the map with polylines



Build an Interactive Map with Folium

The geo spatial analysis give a good foundation for SpaceY to choose locations for their lauches.

Some important questions here :

- Are launch sites in close proximity to railways?
- Are launch sites in close proximity to highways?
- Are launch sites in close proximity to coastline?
- Do launch sites keep certain distance away from cities?

GitHub URL to the notebook for Interactive maps with Folium :notebook

Build a Dashboard with Plotly Dash

The dashboard (created with Dash) gives insight in the success rate total for all historical launches or for each individual launch site.

Correlation between payload and success is also available information in the dashboard.

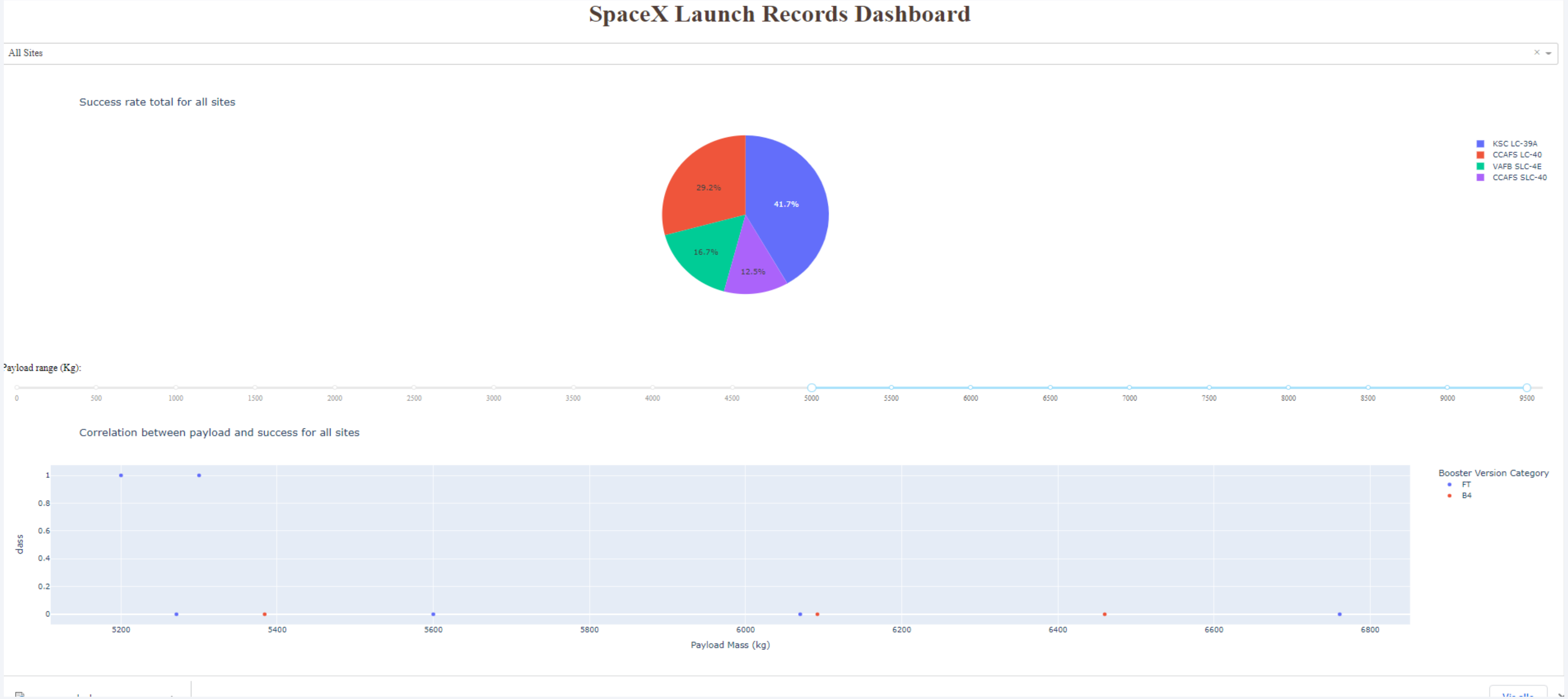
The user can filter on site and payload range

GitHub URL to pdf with more screenshots from dashboard :pdf

GitHub URL to python program for creating Dash application: python.py

Build a Dashboard with Plotly Dash

Screenshot of dashboard (all sites selected and payload >=5000



SpaceX Launch Records Dashboard

All Sites

Success rate total for all sites

Payload range (Kg):

Correlation between payload and success for all sites

Predictive Analysis (Classification)

Different Python packages were used to prepare, build, fit, evaluate and compare different machine learning models to predict the outcome of the target variable (class =success/failure of launches). The most important package is sklearn.

4 different types/algorithms fitted and compared to find the best predictor for success/failure:

#	Modell type/name
1	Support vector machine
2	Nearest neighbors
3	Logistic regression
4	Decision Tree classifier

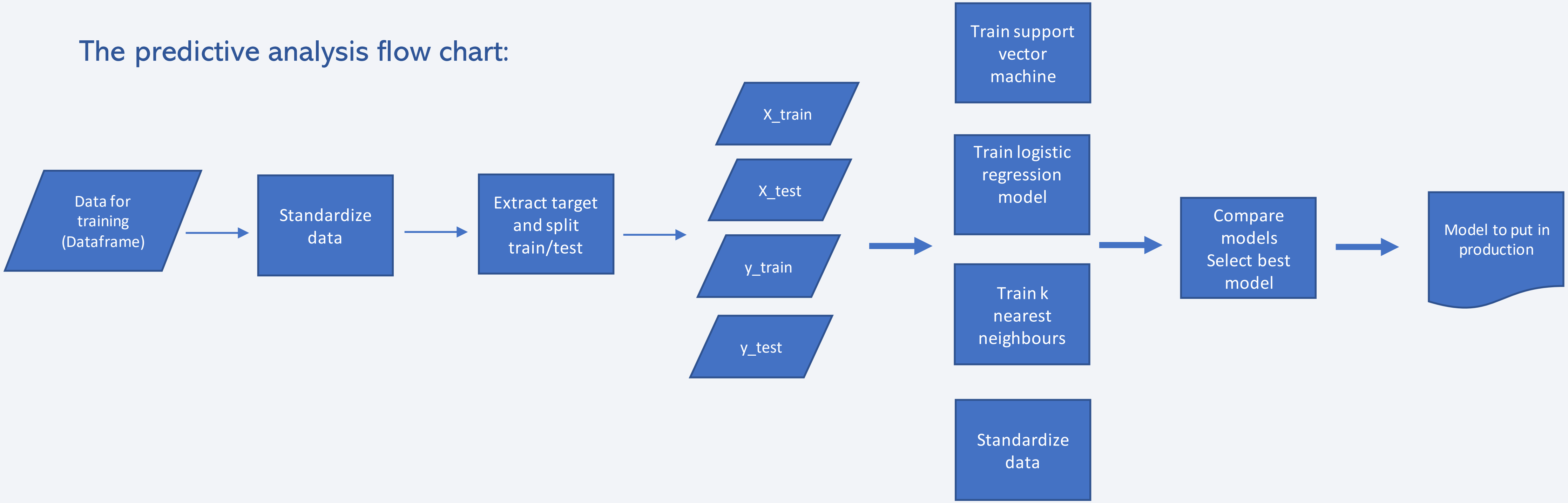
Predictive Analysis (Classification)

The predictive analysis included the following steps:

1. Extract and create the target variable from the dataframe
2. Standardize the data (using the StandarScaler in sklearn)
3. Splitting data in train and test data (both the independent features X and the target variable y)
4. Build a logistic regression model and find the best combination of hyperparameters (GridSearchView). Calculate the accuracy to evaluate the model performance
5. Build a support vector machine model and find the best combination of hyperparameters (GridSearchView). Calculate the accuracy to evaluate the model performance
6. Build a decision tree classifier model and find the best combination of hyperparameters (GridSearchView). Calculate the accuracy to evaluate the model performance
7. Build a k nearest neighbors model and find the best combination of hyperparameters (GridSearchView). Calculate the accuracy to evaluate the model performance
8. Compare the model to find the model that performs best

Predictive Analysis (Classification)

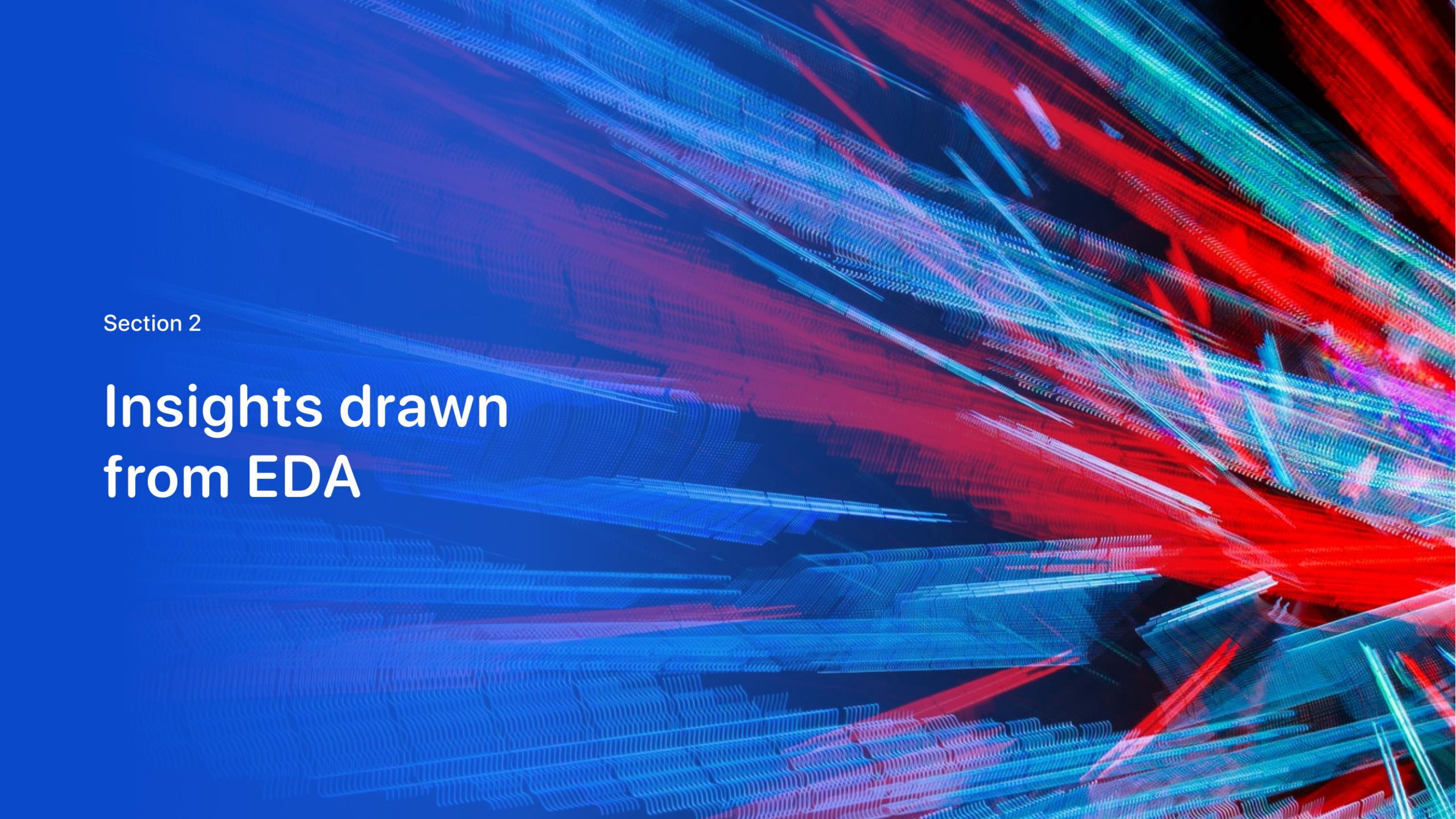
The predictive analysis flow chart:


Data for training (Dataframe)
Standardize data
Extract target and split train/test
X_train
X_test
y_train
y_test
Train support vector machine
Train logistic regression model
Train k nearest neighbours
Standardize data
Compare models Select best model
Model to put in production

GitHub URL to notebook with model training and evaluation : [notebook]

Results

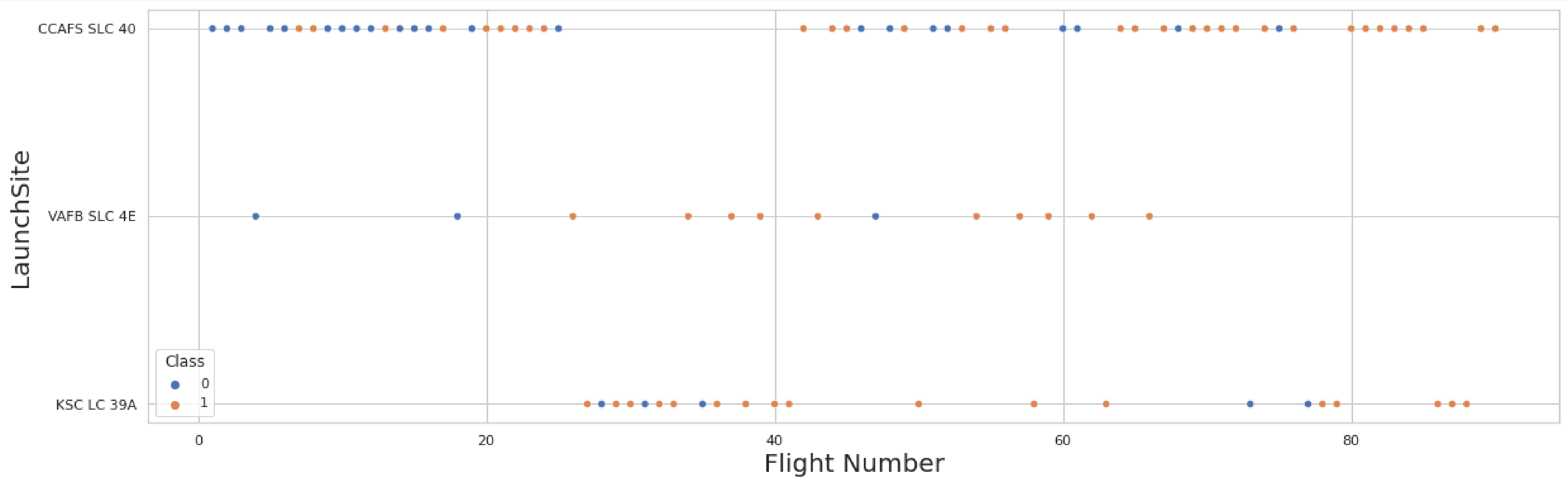
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



Section 2

Insights drawn from EDA

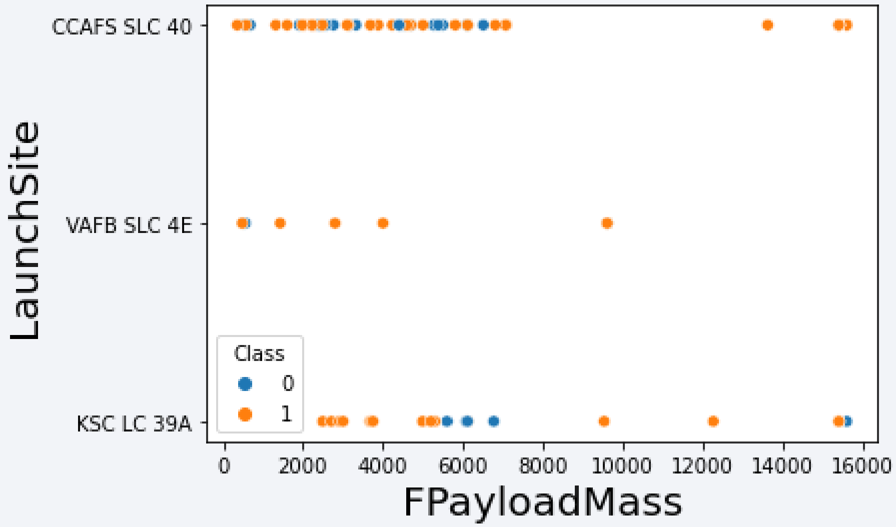
Flight Number vs. Launch Site



The scatter plot shows launches from the different launch sites over time.

The plot shows that the last five launches have been successful on all the sites. The most recent launch has been at CCAFS SLC-40

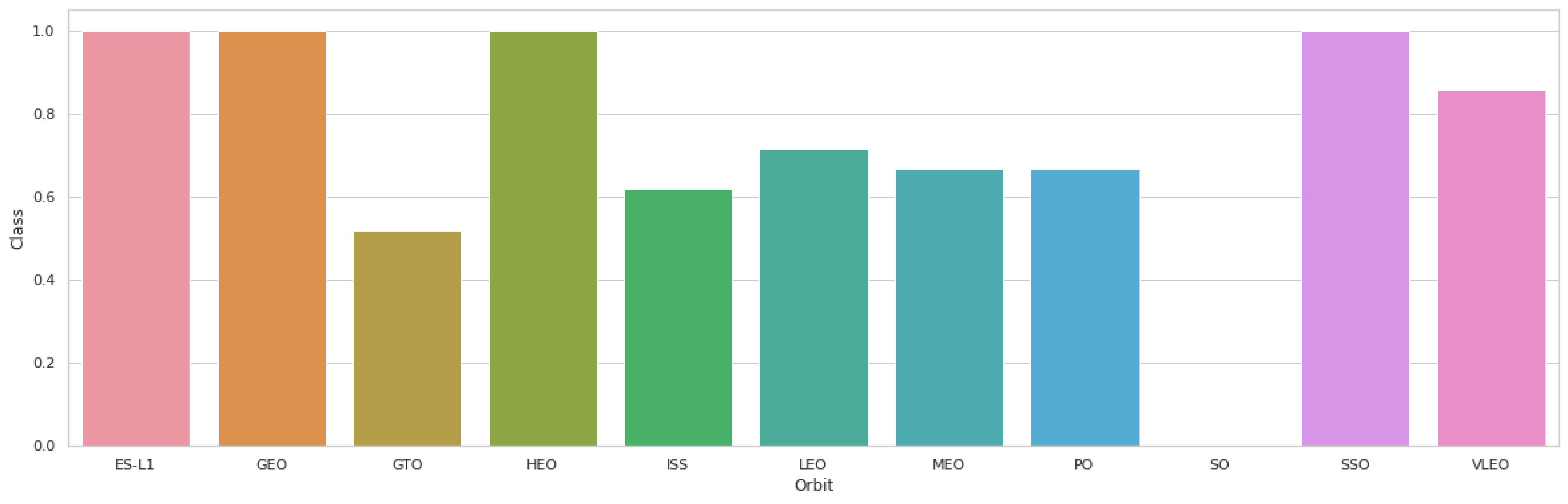
Payload vs. Launch Site



The scatter plot shows Payload for different launches

The plot shows that most of the launches with high payload have been successful and the launch sites with highest payloads are CCAFS SLC 40 anf KSC LC 39A

Success Rate vs. Orbit Type

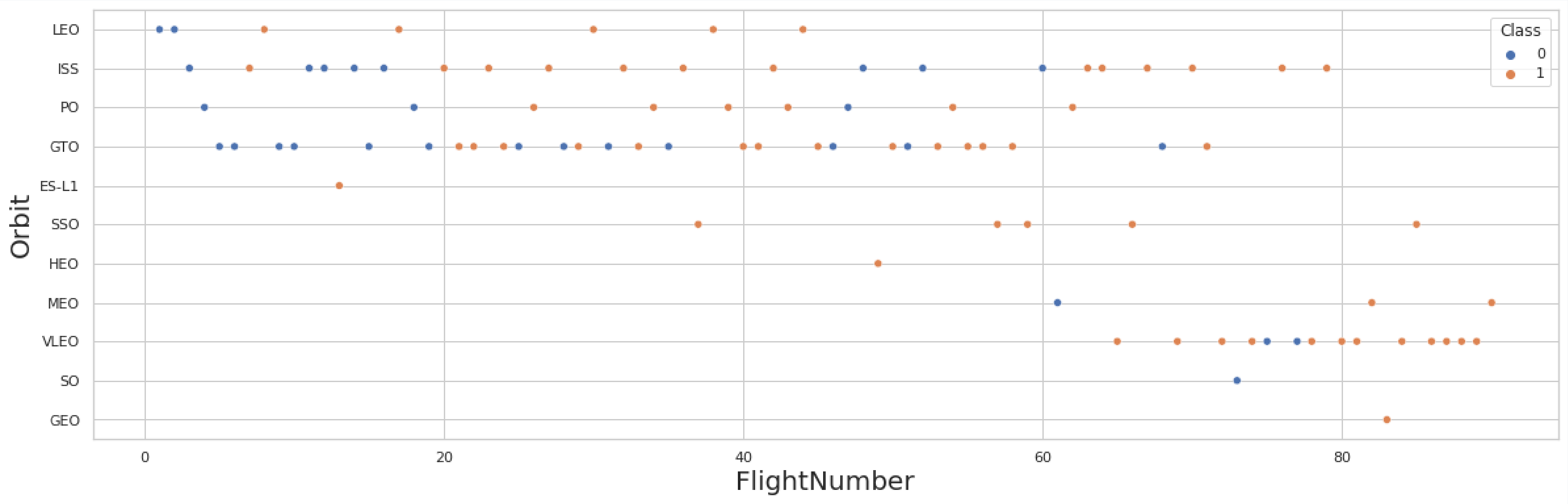


The bar plot shows success rate for different orbits

3 of the orbits with 100% success rate have only one launch

The orbit SSO has 5 launches and all with success

Flight Number vs. Orbit Type



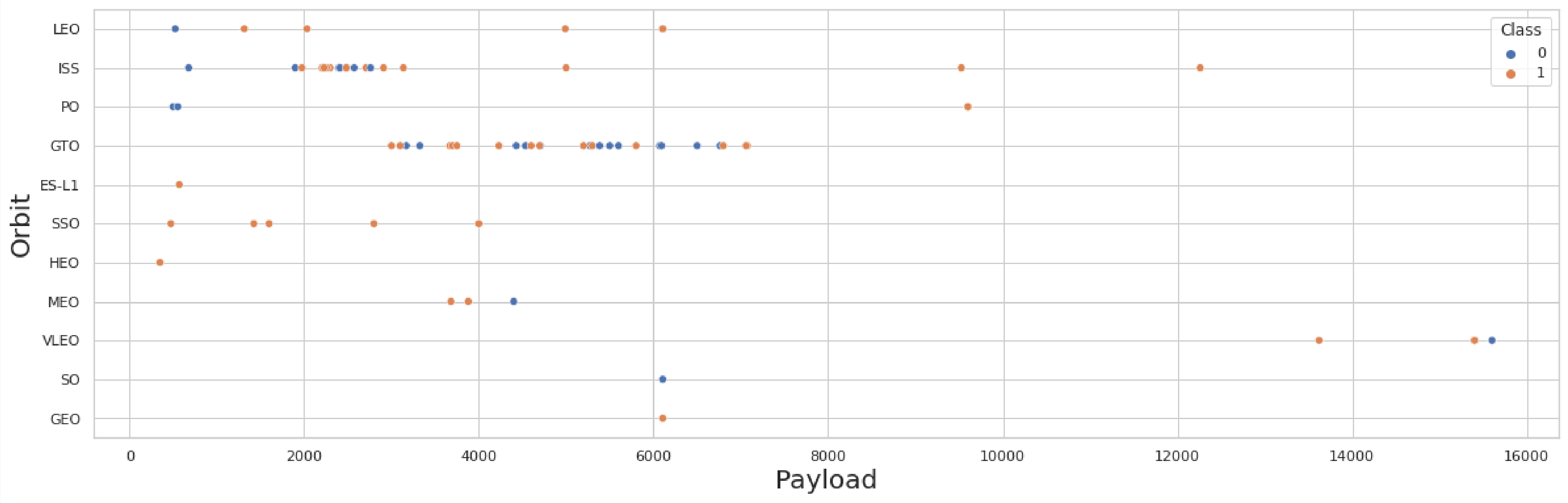
The scatter plot shows success rate for different orbits over time (FlightNumber)

3 of the orbits with 100% success rate have only one launch

The orbit SSO has 5 launches and all with success

All launches with flight number above 80 are successful (independent of orbit)

Payload vs. Orbit Type

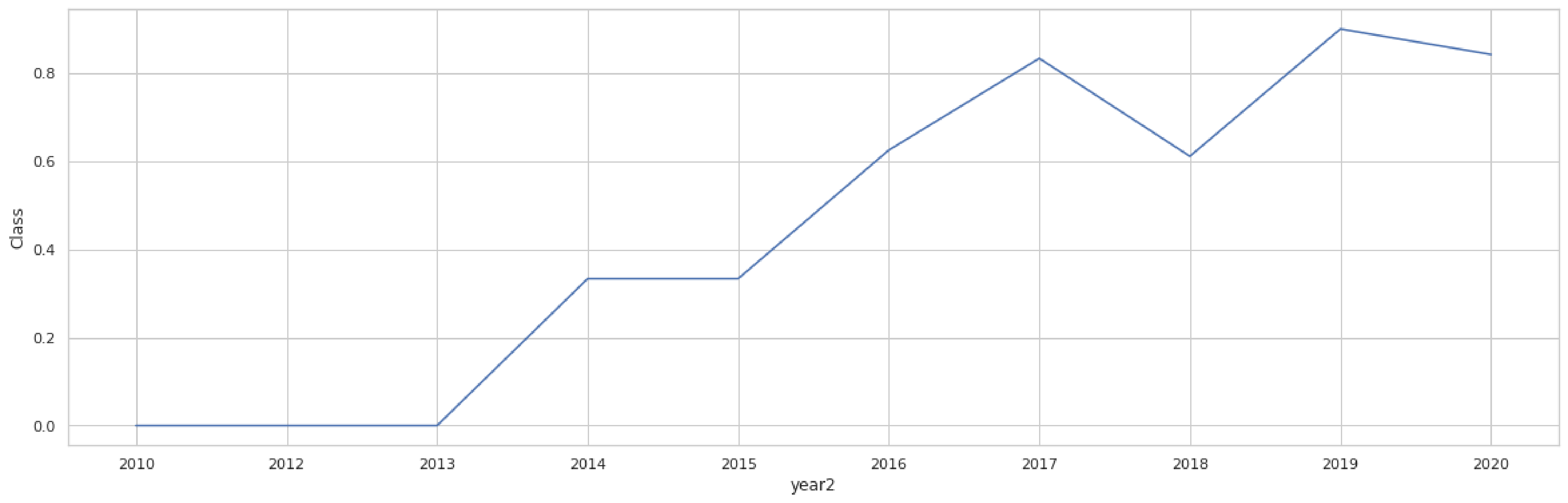


The scatter plot shows the outcomes for different orbits with increasing payload

The launches with highest payload are successful (except for one)

The orbits with highest payload launches are ISS, PO and VLEO

Launch Success Yearly Trend



The line chart shows success rate by year

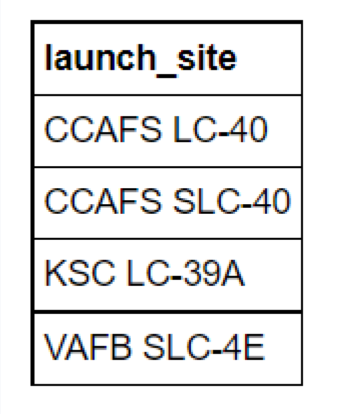
The succes rate has increased since 2013 (with a drop in 2018 and 2020)

All Launch Site Names

Querying the DB2 database table SPACEXDATASET – “Find the names of the unique launch sites”

Query: ***select distinct(LAUNCH_SITE) from SPACEXDATASET***

Result:



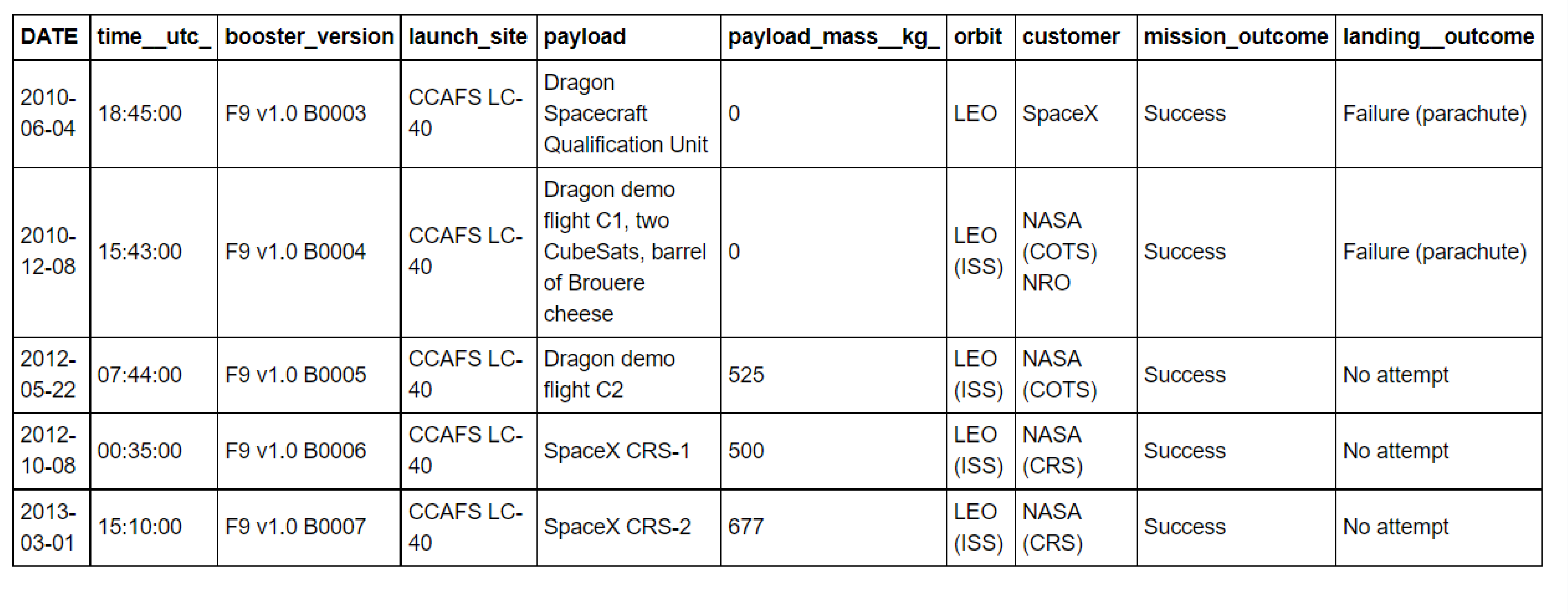
launch_site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E

Launch Site Names Begin with 'CCA'

Querying the DB2 database table SPACEXDATASET – “Find 5 records where launch sites begin with `CCA`”

Query: select * from SPACEXDATASET where SUBSTR(LAUNCH_SITE,1,3)='CCA' FETCH FIRST 5 ROWS ONLY

Result:



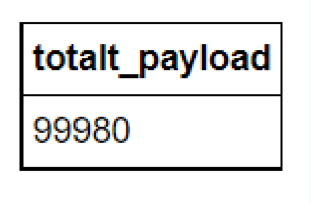
DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight 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

Querying the DB2 database table SPACEXDATASET – "Calculate the total payload carried by boosters from NASA"

Query:**select sum(PAYLOAD_MASS__KG_) as totalt_payload from SPACEXDATASET where substr(customer,1,4)='NASA'**

Result:



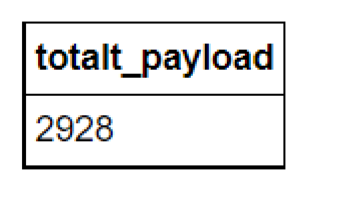
totalt_payload
99980

Average Payload Mass by F9 v1.1

Querying the DB2 database table SPACEXDATASET – “Calculate the average payload mass carried by booster version F9 v1.1”

Query: **select avg(PAYLOAD_MASS__KG_) as totalt_payload from SPACEXDATASET where BOOSTER_VERSION='F9 v1.1'**

Result:



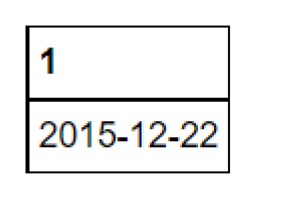
totalt_payload
2928

First Successful Ground Landing Date

Querying the DB2 database table SPACEXDATASET – “Find the dates of the first successful landing outcome on ground pad”

Query: **select min(DATE) from SPACEXDATASET where LANDING__OUTCOME='Success (ground pad)'**

Result:



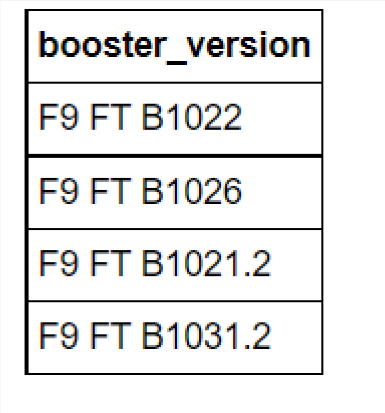
1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Querying the DB2 database table SPACEXDATASET – “List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000”

Query: **select BOOSTER_VERSION from SPACEXDATASET where LANDING__OUTCOME='Success (drone ship)' and PAYLOAD_MASS__KG_>4000 and PAYLOAD_MASS__KG_<6000**

Result:



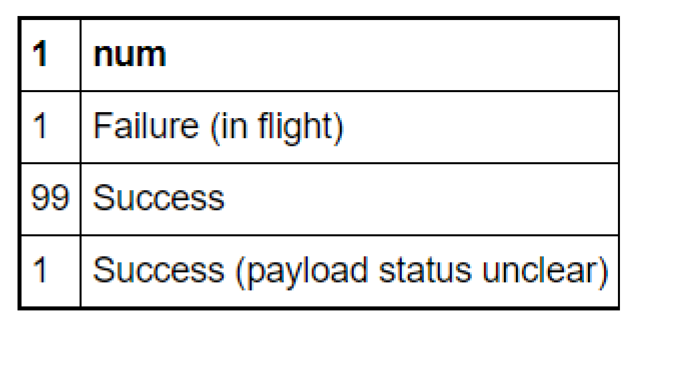
booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

Querying the DB2 database table SPACEXDATASET – “Calculate the total number of successful and failure mission outcomes”

Query: **select count(*),MISSION_OUTCOME as num from SPACEXDATASET group by MISSION_OUTCOME**

Result:



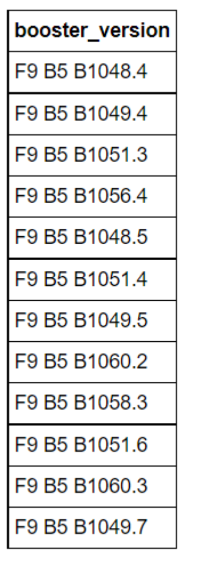
1	num
1	Failure (in flight)
99	Success
1	Success (payload status unclear)

Boosters Carried Maximum Payload

Querying the DB2 database table SPACEXDATASET – “List the names of the booster which have carried the maximum payload mass”

Query: **select BOOSTER_VERSION from SPACEXDATASET where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXDATASET)**

Result:



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

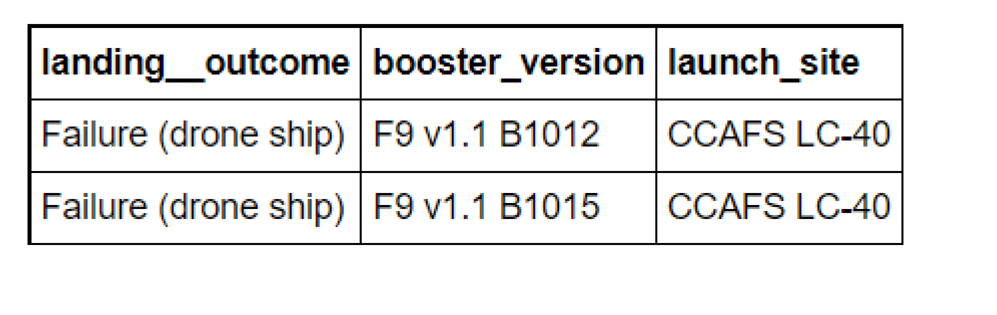
The result contains name of all booster version with maximun payload

2015 Launch Records

Querying the DB2 database table SPACEXDATASET – “List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015”

Query: **select LANDING__OUTCOME, BOOSTER_VERSION,LAUNCH_SITE from SPACEXDATASET where LANDING__OUTCOME='Failure (drone ship)' and year(DATE)=2015**

Result:



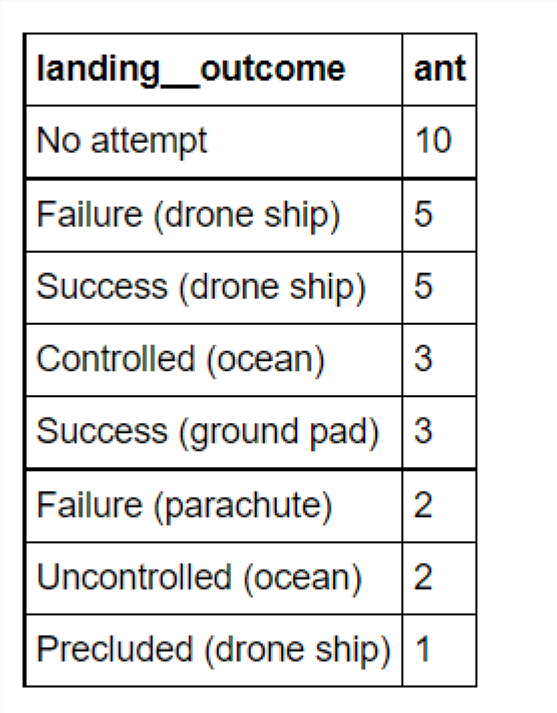
landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

Querying the DB2 database table SPACEXDATASET – "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"

Query: **select LANDING__OUTCOME,count(*) as ant from SPACEXDATASET where DATE>=TO_DATE('2010-06-04', 'YYYY-MM-DD') and DATE<=TO_DATE('2017-03-20', 'YYYY-MM-DD') group by LANDING__OUTCOME order by ant desc**

Result:



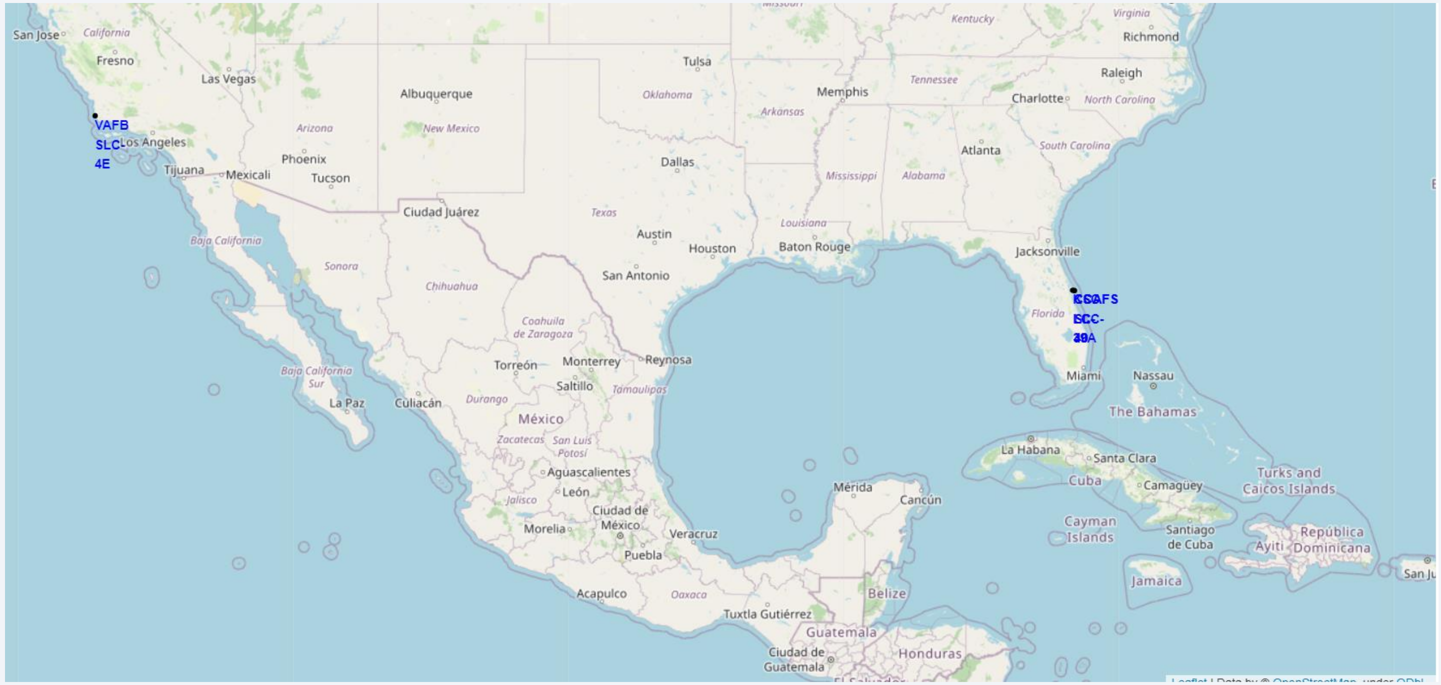
landing__outcome	ant
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 4

Launch Sites Proximities Analysis

Location for the launch sites

This map shows the location of the launch sites:

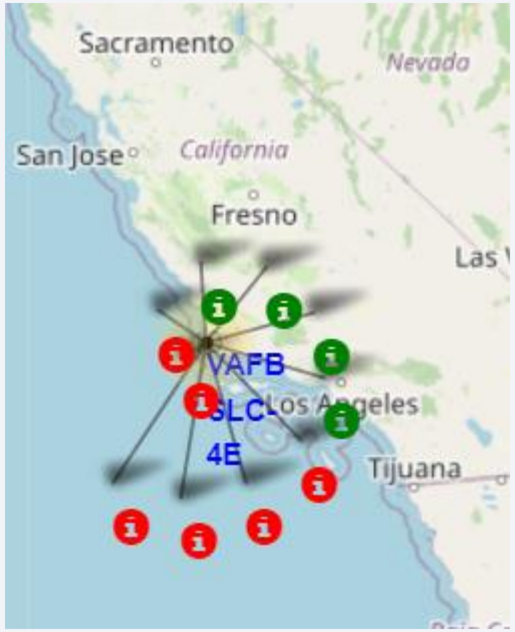


- The sites are located in California and Florida
- The sites are located close to the coast (ocean)
- The site are located approximately at same latitude (28° to 35 °)

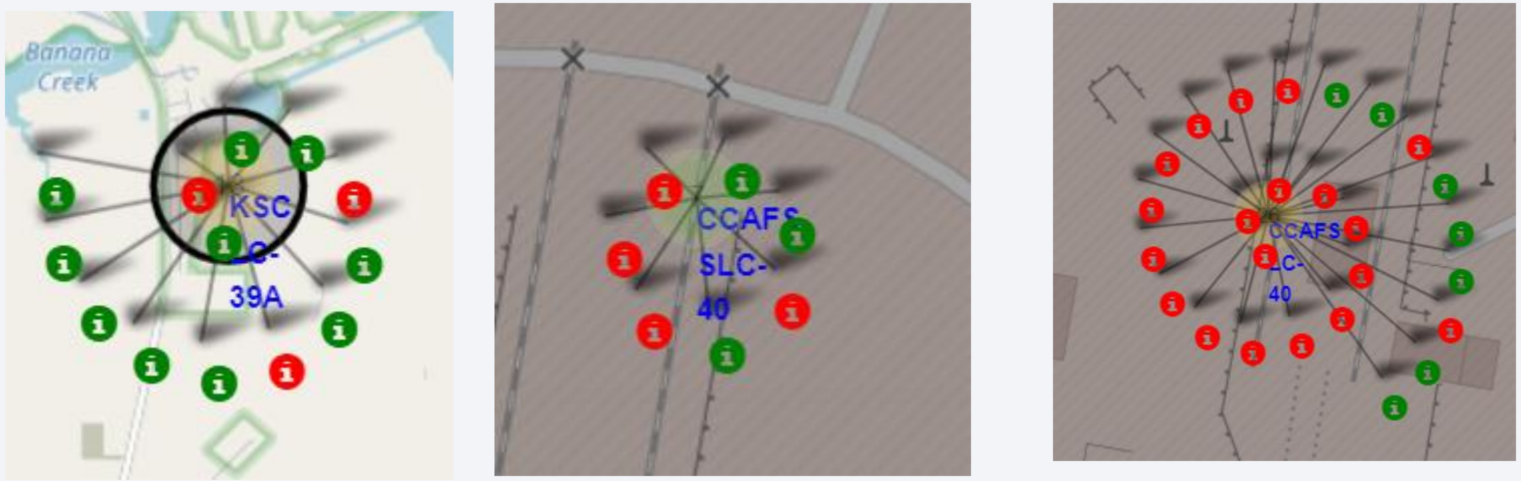
Success/failed launches for each site

This map successed and failed launches for the sites in California and Florida

California:

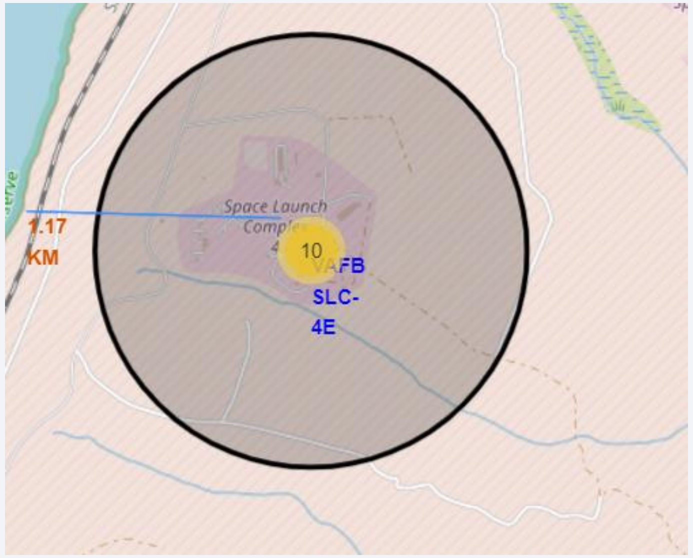


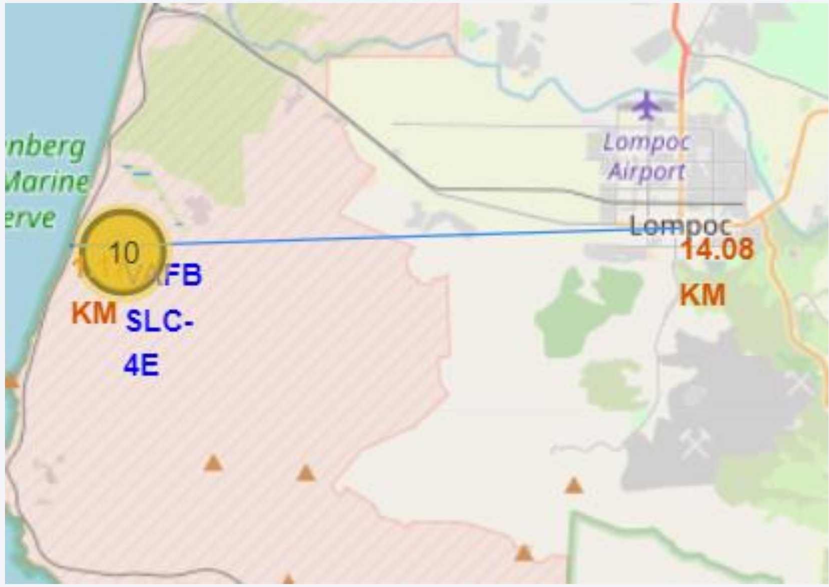
Florida



- The site with highest succeed rate is KSC LC.39A (10/13=0.77)

Distances between launch sites and important infra structure



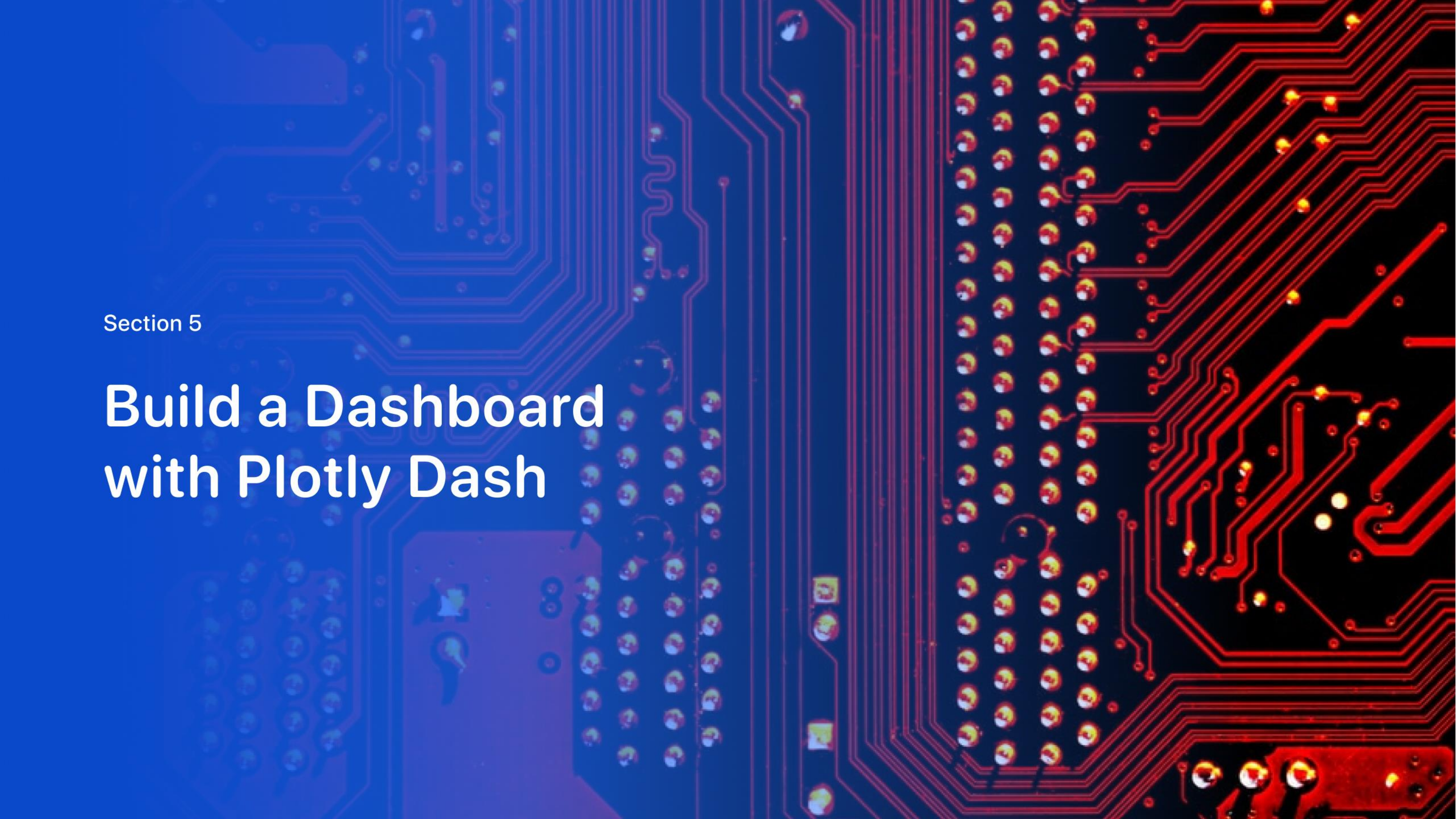


The map shows that the distance from the launch site VAFB SLC-4E to:
the coast is 1.17 km
Railway less than 1.17 km

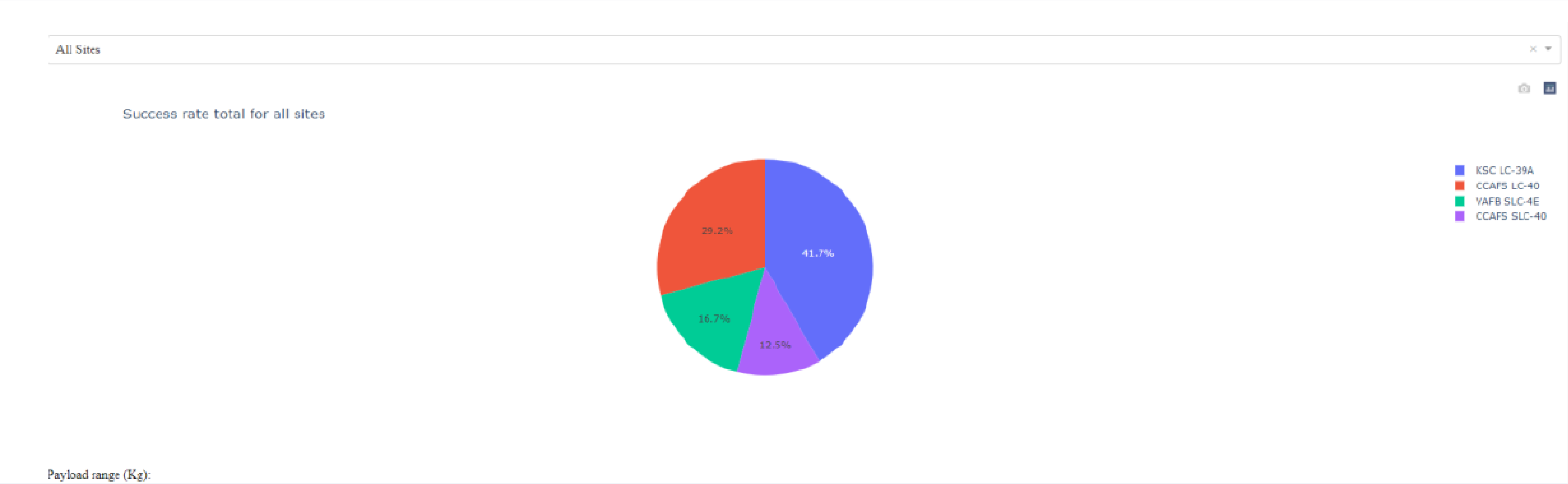
The map shows that the distance from the launch site VAFB SLC-4E to:
The nearest city/village (Lompoc):14.08 km

Section 5

Build a Dashboard with Plotly Dash



Launch success rate for all sites



All Sites

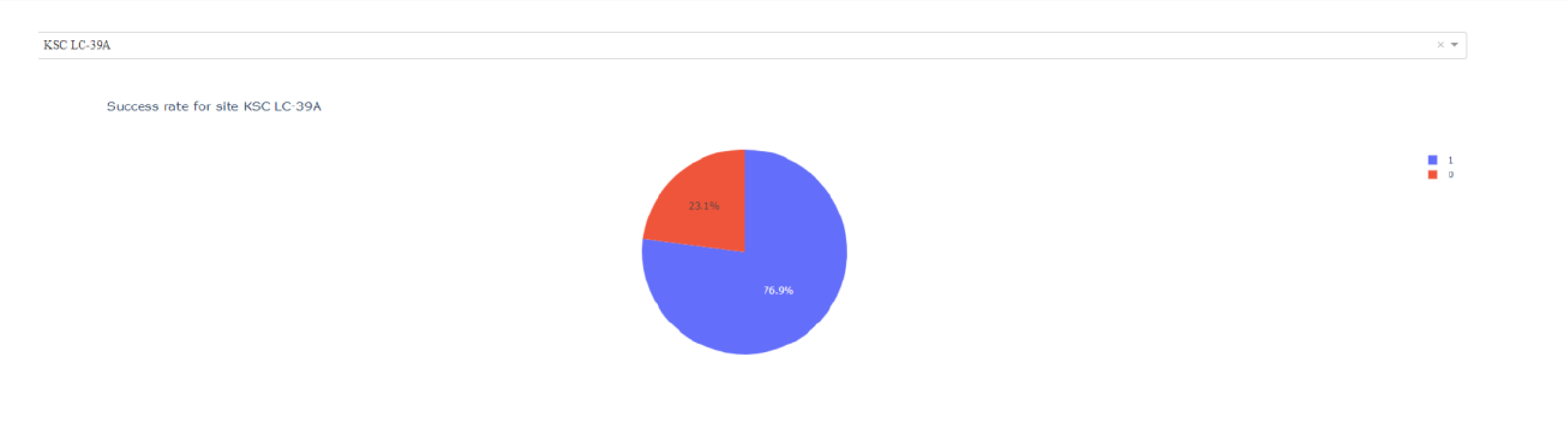
Success rate total for all sites

Payload range (Kg):

The Piechart shows the success rate for the different launch sites

KSC LC-39 A has 41.7 % of the total number of successfully launches

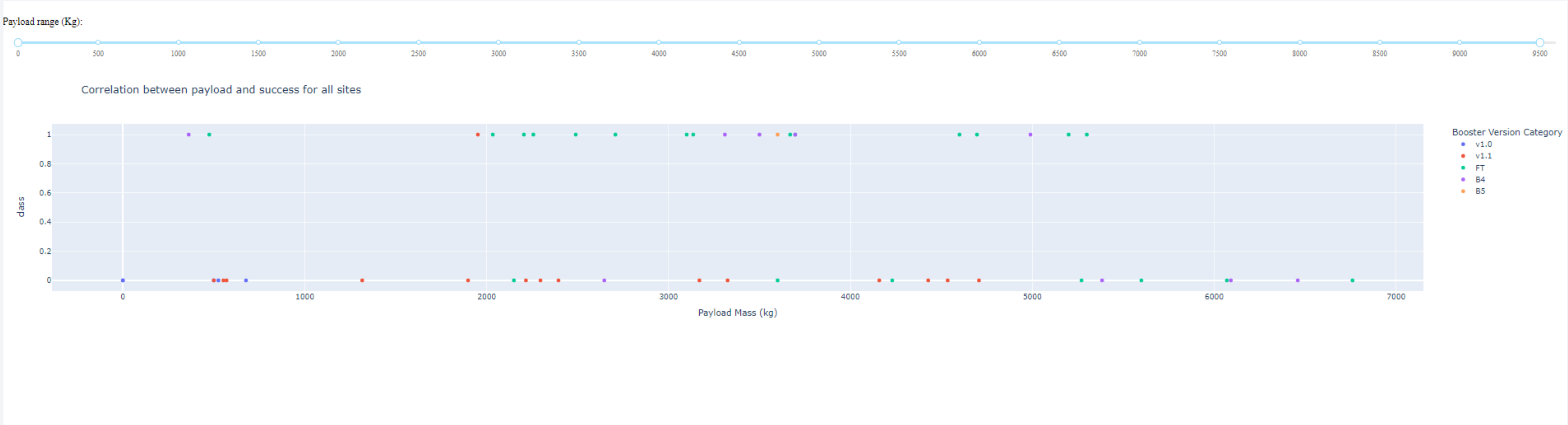
Success rate for the launch site KSX LC-39A



KSC LC-39A

The success rate for the site with highest success rate is: 76.9 %

Payload vs. Launch Outcome



Most of the successfully launches have payload in the range from 2000 to 5000 kg.

The booster versions categories for the launches with highest payload are “FT” and “B4”

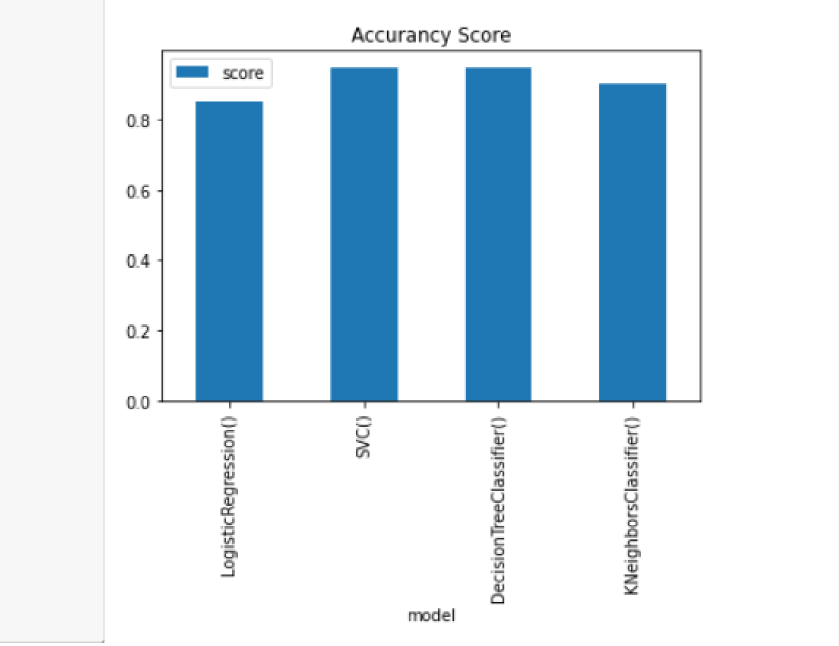
Section 6

Predictive Analysis (Classification)

Classification Accuracy

The bar chart show accuracy score for the 4 different machine learning models:

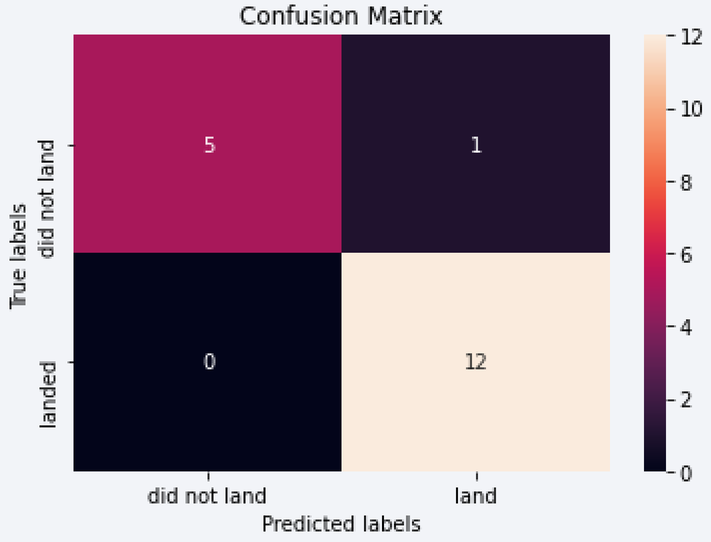
All of the models have quite high accuracy scores, but SVC and Decision Tree Classifier have the highest with 0.95.



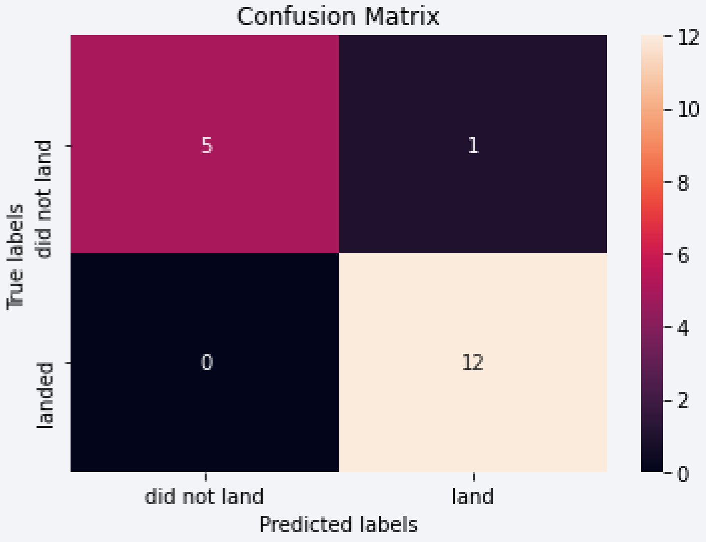
Confusion Matrix

Under are the confusion matrices for the 2 models with highest accuracy score:

SVM:



Decision Tree classifier



The confusion matrices show that the models predict correctely 17 of 18 test observations. 12 true positive (TP) and 5 true negative (TN) and 1 false postive (FP). 17/18=0.94 (accurancy score)

Conclusions

- Analysis indicate that the success of launches depends on payload, orbit and the site.
- Increasing success with experience/time (flight number)
- Launches with average or high payload (in the range 2000 to 5000) are most successful
- Successrate depends on orbit. The orbit with most successes are SSO.
- Geographical analyses show that the launch sites should be placed close to the coast and nearby important infrastructure as highways and railways. The sites should no be to close to cities or other crowed places (towns/villages etc.).
- Machine learning models can be good tools to find the most important factors to predict outcomes of launches. A dataset with information about payload, block, reused count, orbit, launch sites were used to build different machine learning models to predict success of launches. The most promising types of models/algorithms were svc and decision tree classifier (Accuracy score 0.95 on test data.

Appendix

Thank you!