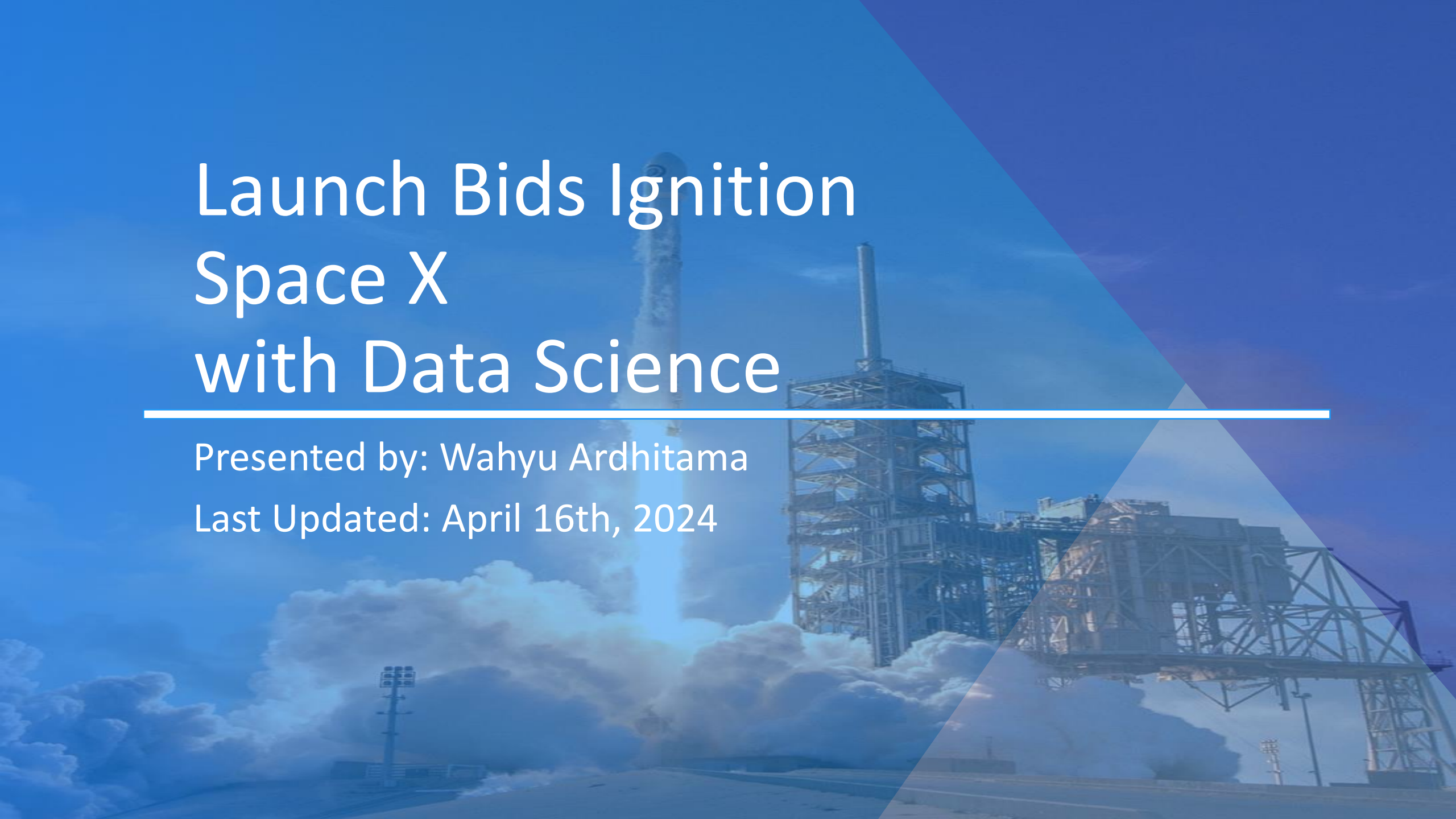


# Launch Bids Ignition Space X with Data Science

---

Presented by: Wahyu Ardhitama

Last Updated: April 16th, 2024







# Table of Contents

- **Executive Summary**
- **Introduction: Business Case and Objective**
- **Section 1. Methodology - PACE Framework**
- **Section 2. Insight Drawn from Exploratory Data Analysis**
- **Section 3. Map - Launch Sites Proximity Analysis**
- **Section 4. Dashboard - Build a Dashboard with Plotly Dash**
- **Section 5. Model – Predictive Analysis (Classification)**
- **Conclusion**
- **Appendix**

# Executive summary

## ➤ We use PACE framework

### Data Analysis PACE Steps:

1.  Plan/Prepare - import the relevant libraries and data  
Align project with business needs, requirements and constraints. Select an appropriate machine learning model based on the problem and business context. KDD: Selection, Data Wrangling (Pre-processing and Transformation).
2.  Analyze - Exploratory Data Analysis (EDA)  
Understanding data for accurate predictions, focus on the response variable (what the model predicts) and leverage exploratory data analysis to uncover patterns and address irregularities. KDD: Data Mining.
3.  Construct - model  
Construct and evaluate model. KDD: Evaluation.
4.  Execute - share  
Interpret model and share the story. KDD: Communicate to stakeholders.

## ➤ The recommended model

1. Random forest has the highest classification accuracy result
2. Next consider to explore other models

	Model	Train Score	Test Score	Accuracy
1	KNN	0.875000	0.944444	0.844444
3	SVM	0.861111	0.944444	0.822222
4	Logistic Regression	0.875000	0.944444	0.822222
2	Decision Tree	0.861111	0.888889	0.900000
0	RF	0.943576	0.861111	0.908333



# Introduction: Business case and Objective

# Business Case

Space X offers competitive price since its first successful landing in 2015 by reusing the rocket, more than half of its competitors' cost

- **Opportunity/Problem Statement:**
  - Space x has been operating since 2010 and the landing success rate continue to increase
  - Launch sites have an average success rate below 70%

# Goals

Improve the success rate of successful landings, currently averaging below 70%, to achieve a competitive bid proposal cost of \$62 million compared to our competitors

- **Project Objective:**  
**Increase Revenue from Rocket Launch**
- **KPI:**  
**Successful landing launch rate (%)**

## **Primary metric:**

- Improve **landing success rate** by 10% by identifying best launch site, orbit, pay load mass etc.

# Analytics Objective

Explore and compare **various machine learning models** and **find one** with **the best performance** to improve rockets landing

## Machine learning models:

- Predict the best successful landing site.
- The best option for customers launching on the landing site that offers the most value, considering factors such as from orbit, location etc.
- Determine the cost of the launch

## Hypothesis :

**Machine learning models delivers more incremental value of landing success relative to the current systems**



A photograph of a SpaceX Falcon Heavy rocket launching from the Kennedy Space Center. The rocket is ascending vertically, leaving a large, billowing plume of white smoke and fire at its base. To the right of the rocket, the massive, complex metal structure of the Mobile Launcher Platform (MLP) is visible, supported by a tall tower. The sky is a clear, deep blue. The overall scene captures the power and scale of heavy-lift space exploration.

# Section 1. Methodology

## PACE Framework





Plan



Align project with business needs, requirements and constraints. Select an appropriate machine learning model based on the problem and business context. KDD: Selection, Data Wrangling (Pre-processing and Transformation).

# Plan

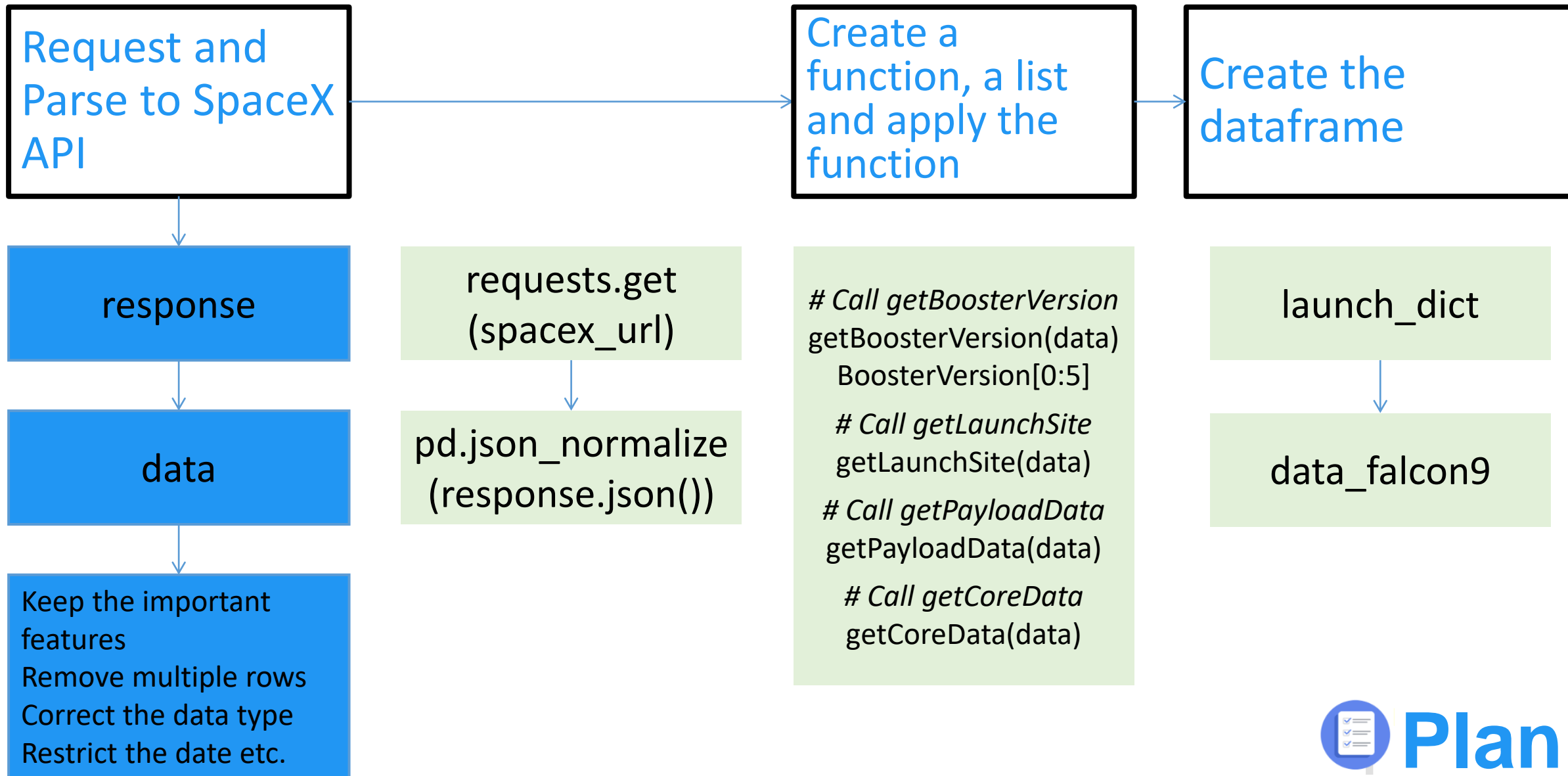
## Selection - data collection

- Takes the dataset and uses required features column to call the API and append the data to the list. <https://api.spacexdata.com/v4/launches/past>
- Request and parse the SpaceX launch data. [https://.../API\\_call\\_spacex\\_api.json](https://.../API_call_spacex_api.json)

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [01. jupyter-labs-spacex-data-collection-api.ipynb \(github.com\)](#)

# Data collection from API - Flowchart



# Data collection from API - Flowchart

data\_falcon9  
3 launch sites  
Flight No. 90

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs		LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False		None	1.0	0	B0003	-80.577366	28.561857
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False		None	1.0	0	B0005	-80.577366	28.561857
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False		None	1.0	0	B0007	-80.577366	28.561857
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False		None	1.0	0	B1003	-120.610829	34.632093
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False		None	1.0	0	B1004	-80.577366	28.561857
...	...	...	...	...	...	...	...	...	...	...	...		...	...	...	...	...	...
89	86	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	12	B1060	-80.603956	28.608058	
90	87	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	13	B1058	-80.603956	28.608058	
91	88	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	12	B1051	-80.603956	28.608058	
92	89	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3	True	True	True	5e9e3033383ecbb9e534e7cc	5.0	12	B1060	-80.577366	28.561857	
93	90	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1	True	False	True	5e9e3032383ecb6bb234e7ca	5.0	8	B1062	-80.577366	28.561857	

90 rows × 17 columns



Align project with business needs, requirements and constraints. Select an appropriate machine learning model based on the problem and business context. KDD: Selection, Data Wrangling (Pre-processing and Transformation).

# Plan

## Data Scrapping

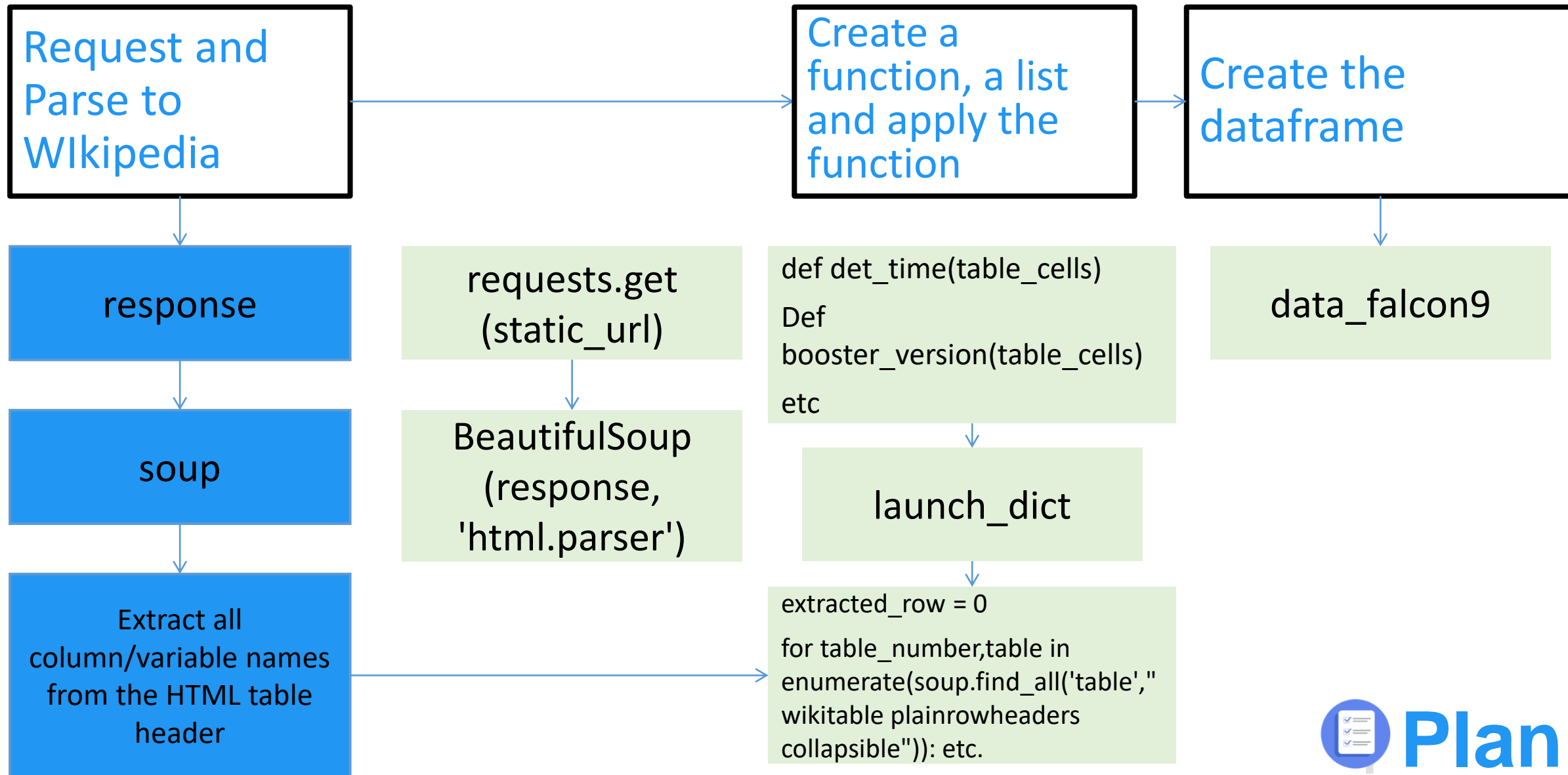
- Collect Falcon 9 historical launch records from wikipedia.

[https://en.wikipedia.org/wiki/List\\_of\\_Falcon\\_9\\_and\\_Falcon\\_Heavy\\_launches](https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [02. jupyter-labs-webscraping.ipynb](#)

# Scrapping from Wikipedia- Flowchart



# Data collection from Wikipedia- Flowchart

data\_falcon9  
5 launch sites  
Flight No. 121

Data from the API differs from that on Wikipedia. We are adhering to the data outlined in the IBM capstone assignment.

Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10
Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
116	117	CCSFS Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1051.10	Success	9 May 2021	06:42
117	118	KSC Starlink	~14,000 kg	LEO	SpaceX	Success\n	F9 B5B1058.8	Success	15 May 2021	22:56
118	119	CCSFS Starlink	15,600 kg	LEO	SpaceX	Success\n	F9 B5B1063.2	Success	26 May 2021	18:59
119	120	KSC SpaceX CRS-22	3,328 kg	LEO	NASA	Success\n	F9 B5B1067.1	Success	3 June 2021	17:29
120	121	CCSFS SXM-8	7,000 kg	GTO	Sirius XM	Success\n	F9 B5	Success	6 June 2021	04:26





Align project with business needs, requirements and constraints. Select an appropriate machine learning model based on the problem and business context. KDD: Selection, Data Wrangling (Pre-processing and Transformation).

# Plan

## Data Wrangling (pre-processing and transformation)

- Convert outcomes training labels with 1 means the booster successfully landed, 0 means it failed

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [03. labs-jupyter-spacex-Data wrangling.ipynb \(github.com\)](#)

# Wrangling- Flowchart

df  
3 location sites  
90 flight numbers

Calculate the  
number of  
launches on  
each site

Calculate the  
number and  
occurrence of  
each orbit

Calculate the  
number and  
occurrence of  
mission **outcomes** of  
the orbits

Assign to the  
variable  
landing\_class

Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857	0
LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857	0
ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857	0
PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093	0
GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857	0



Analyze

Exploratory Data Analysis



Understanding data for accurate predictions, focus on the response variable (what the model predicts) and leverage exploratory data analysis to uncover patterns and address irregularities. KDD: Data Mining.

# Analyze

## SQL

- In this SQL data, 4 launch sites. **Unfortunately it differ from data in wrangling phase.**
- Lunch sites begin with the string 'CCA'
- There are 53 Customers
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- The first successful landing outcome 22th Dec 2015
- List the total number of successful and failure mission outcomes
- 12 booster\_versions which have carried the maximum payload mass.
- Rank the count of landing outcomes between the date 2010-06-04 and 2017-03-20
- List the failed landing\_outcomes in drone ship, their booster versions, and launch site names for in year 2015

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [04. jupyter-labs-eda-sql-coursera sqlite.ipynb \(github.com\)](#)

df\_1

4 location sites

101 flight numbers



Understanding data for accurate predictions, focus on the response variable (what the model predicts) and leverage exploratory data analysis to uncover patterns and address irregularities. KDD: Data Mining.

# Analyze

## EDA Data Visualization

- Scatterplot to observe relationship between variables.
- Bar plot to visualize the success rate of each orbit.
- Line chart to track the trend of the success rate over each year.
- Pie chart to analyze the success rate of each launch site.

**df**

3 location sites  
90 flight numbers

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [05. edadataviz.ipynb \(github.com\)](#)



Understanding data for accurate predictions, focus on the response variable (what the model predicts) and leverage exploratory data analysis to uncover patterns and address irregularities. KDD: Data Mining.

# Analyze

**df**  
4 location sites  
56 flight numbers

## Interactive Map with Folium

- First, create a Folium Map object with an initial center location set to NASA Johnson Space Center in Houston, Texas
- Use folium.Circle to add a highlighted circle area
- Add marker using folium.Marker
- Mark each site using MarkerCluster()
- Add a MousePosition on the map to get coordinate for a mouse over a point on the map
- Calculate the distance and add distance marker
- Create lines using folium.Polyline

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [06. lab jupyter launch site location.ipynb \(github.com\)](#)



Understanding data for accurate predictions, focus on the response variable (what the model predicts) and leverage exploratory data analysis to uncover patterns and address irregularities. KDD: Data Mining.

# Analyze

**df**  
4 location sites  
56 flight numbers

## Dashboard with Plotly Dash

- Add a dropdown list to enable Launch Site selection
- Create pie chart to show the total successful launches count for all sites
- Add a slider to select payload range
- Create a scatter chart to show the correlation between payload and launch success
- Add a callback function for `site-dropdown` as input, `success-pie-chart` as output
- Add a callback function for `site-dropdown` and `payload-slider` as inputs, `success-payload-scatter-chart` as output

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [spacex\\_dash\\_app.py \(github.com\)](https://github.com/spacex_dash_app.py)





# Construct

## Models and Findings



# Construct

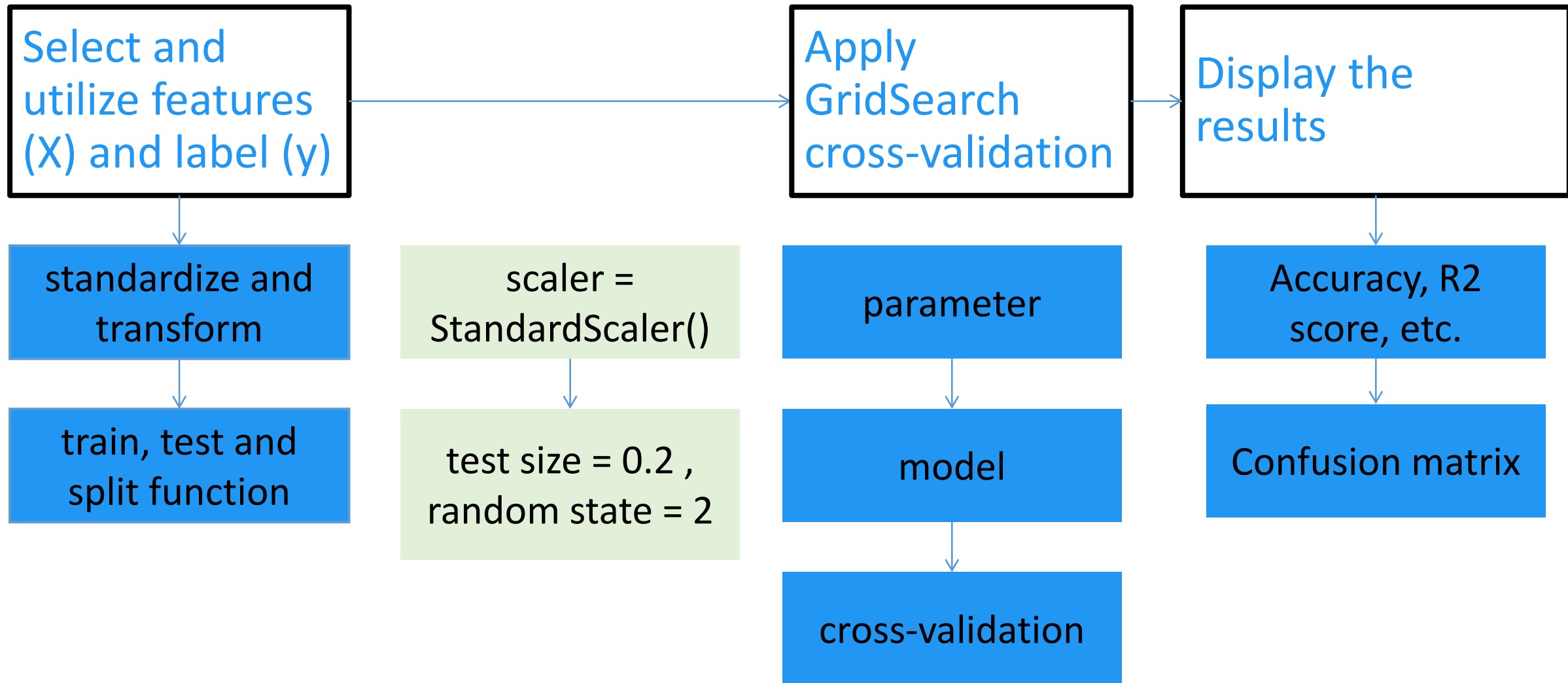
## Predictive Analysis (Classification)

- Choose the relevant features
- Perform One hot encoding for the object or string values data
- Select the features (X) and label (y)
- Standardize the data in X and then reassign it to the variable X
- Utilize the function `train_test_split` to split in the data X and Y into training and test data. Set the parameter `test_size` to 0.2 and `random_state` to 2.
- Determine the parameter and apply the training and test datasets to logistic regression, SVM, Decision tree and KNN model
- Employ GridSearch with the model, cross-validation with a value of 10 and the parameter
- Display the results including accuracy,  $R^2$  and confusion matrix

## Link

- Task008-P01-DS-SpaceX-Falcon-20240409 in Kaggle (The complete version with outcome and code cell)
- [07. SpaceX Machine Learning Prediction Part 5.ipynb \(github.com\)](#)

# Predictive Analysis- Flowchart



PERFECTING  
PROPULSIVE  
LANDING

## Section 2. Insight Drawn from EDA





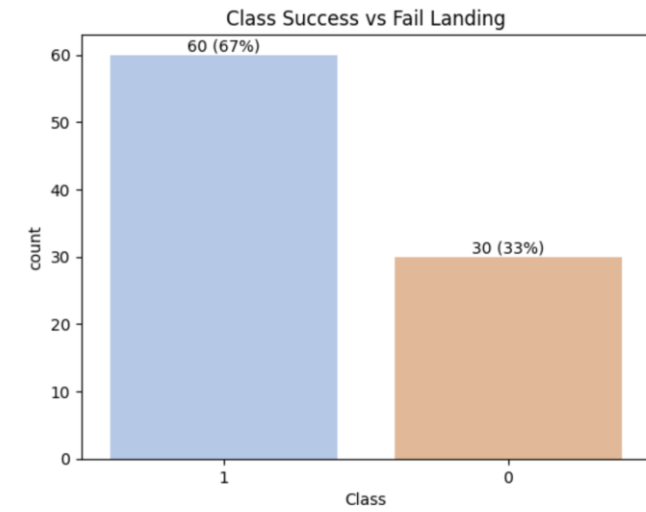
# Analyze

## EDA - Data Visualization

# The first landing

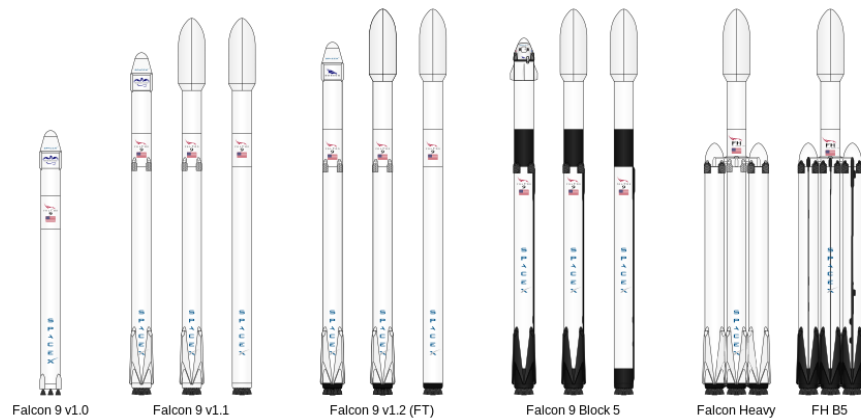


**Success  
(Class = 1)**

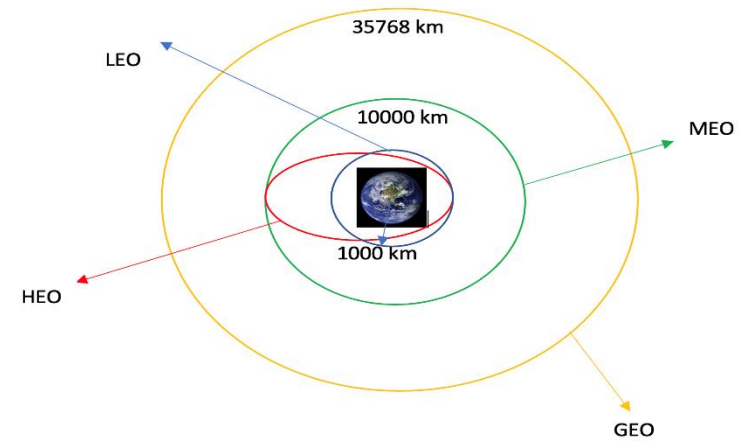


**Failure  
(Class = 0)**

# The rockets and orbits



Heavy  
Launch



Orbit

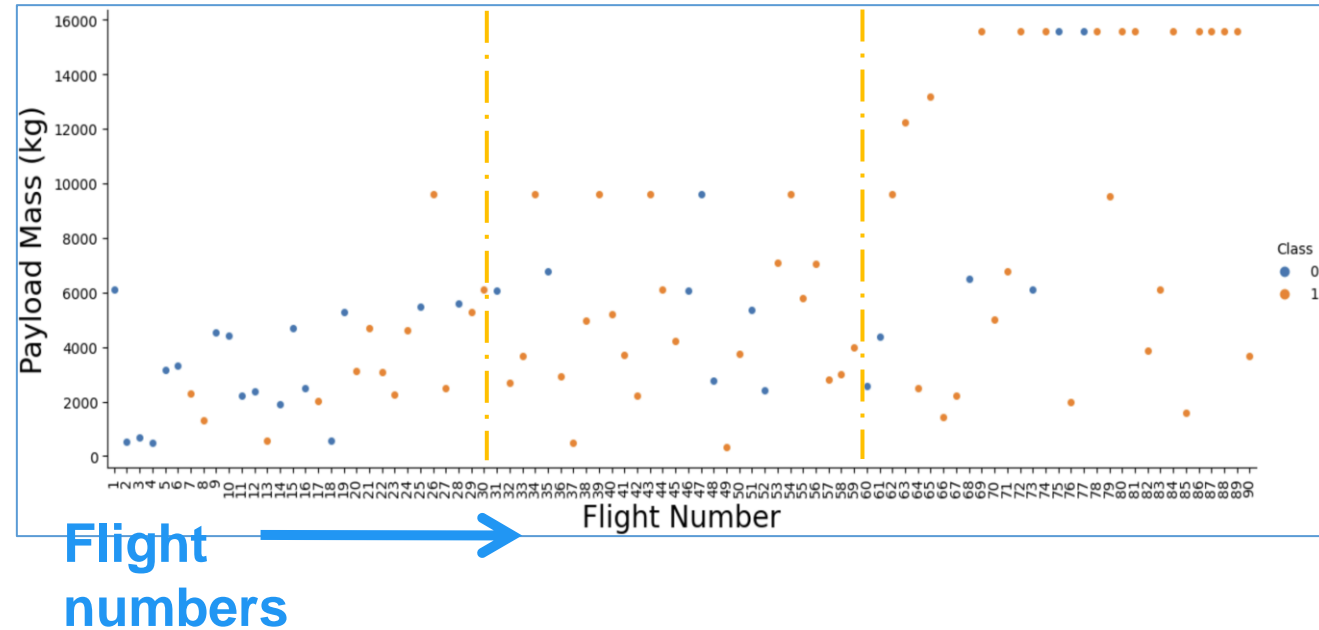


# Flight number vs Payload mass

From 2010 to 2020, we have had 90 flight numbers

- Flight numbers increases, the first stage more likely land successfully

Payload  
Mass



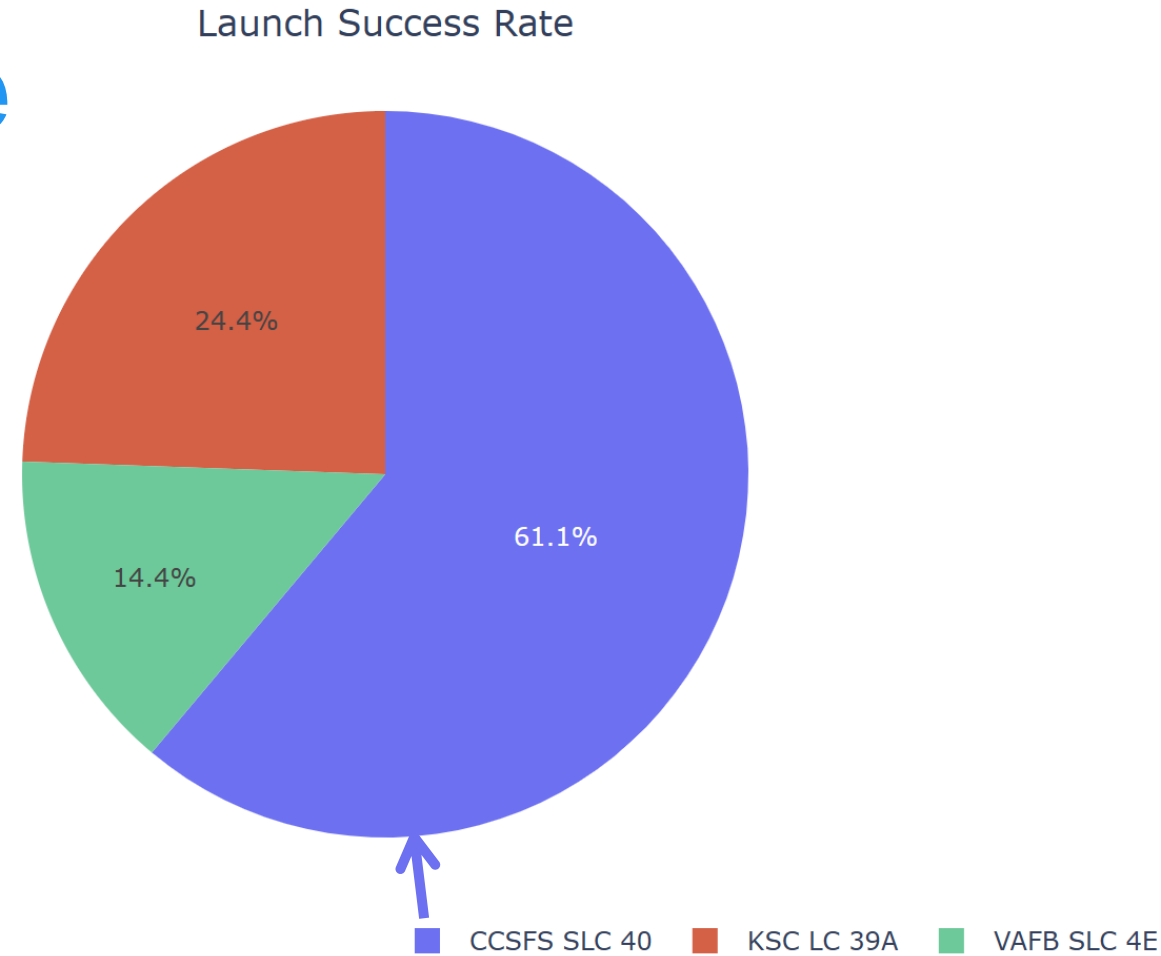
df

3 location sites  
90 flight numbers

# Launch Success Rate Contribution

**2010 to 2020 we have to 90 flight numbers**

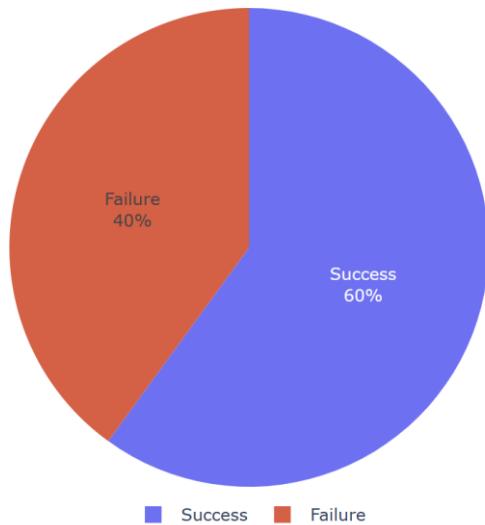
- `CCSFS SLC 40` contributed the highest launching success, more than 60% of total (its first launched in 2010)
- `KSC LC 39A` followed second place with nearly 25% of total. (launching in 2013)
- The youngest launch site in 2017, VAFB SLC 4E has less than 15% of the total launches.



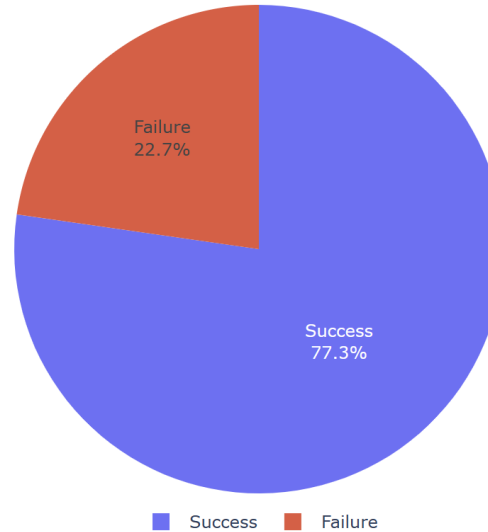
# Launch Success Rate

df  
3 location sites  
90 flight numbers

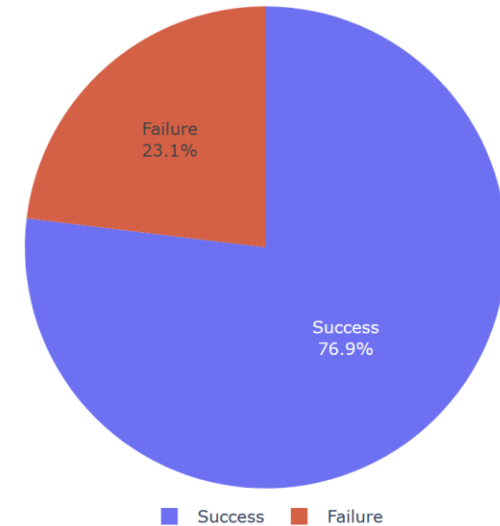
Launch Success Rate for CCSFS SLC 40



Launch Success Rate for KSC LC 39A



Launch Success Rate for VAFB SLC 4E



## 2010 to 2020 we have to 90 flight numbers

- `KSC LC 39A` and `VAFB SLC 4E` has success rates of more than 75%
- `CCAFS SLC 40` has a success rate of 60% (the oldest launch site)

\*It should be CCAFS instead of CCSFS

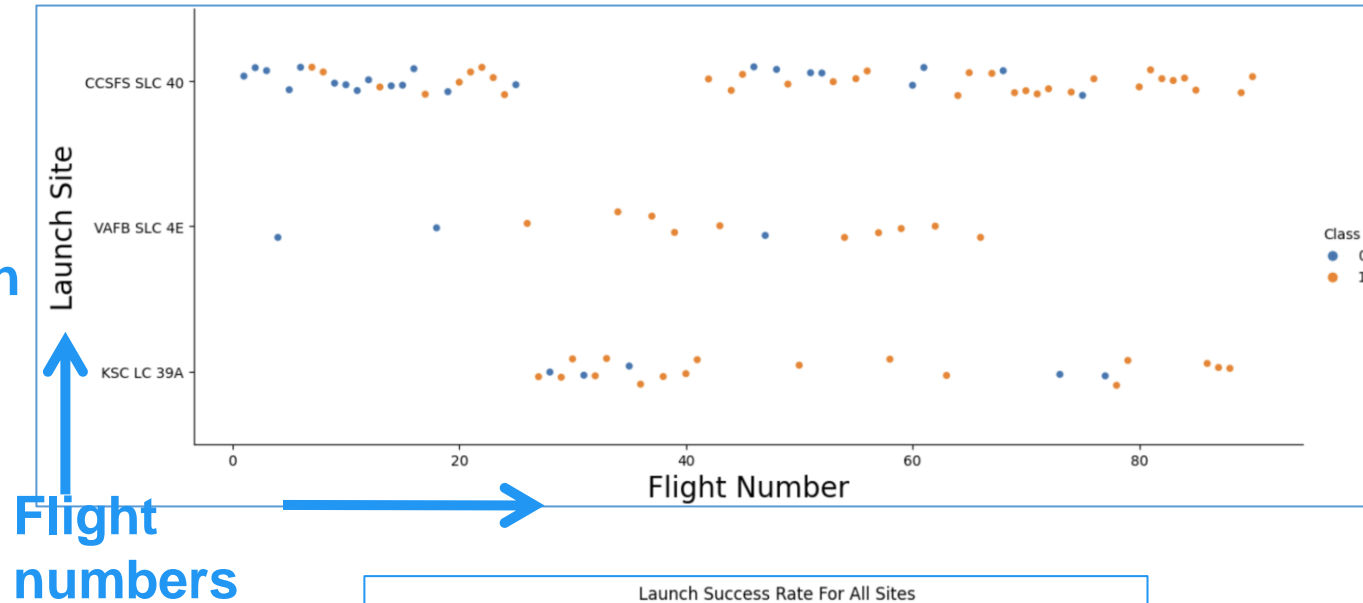
# Flight number vs Launch site



2010 to 2020 we have to 90 flight numbers

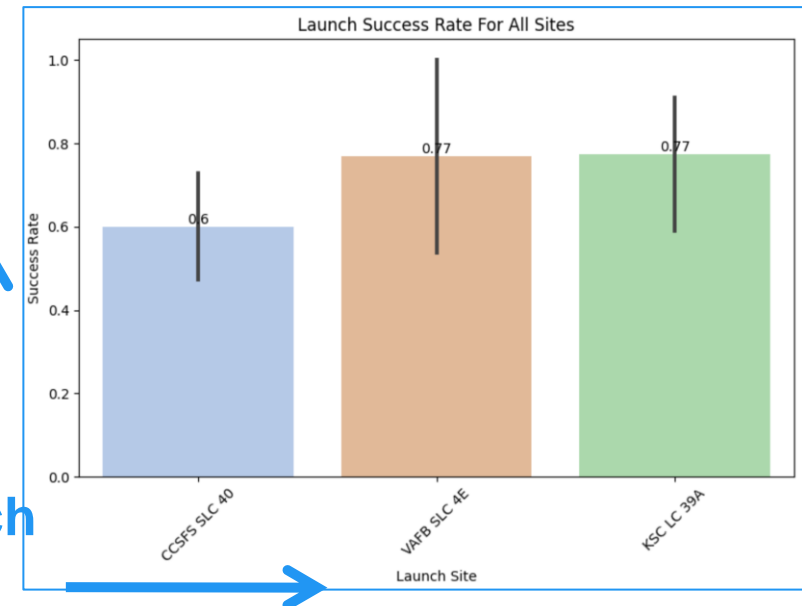
- As the flight number increases, the success rate of the first landing improves across all launch sites

Launch Site



Success Rate

Launch Site

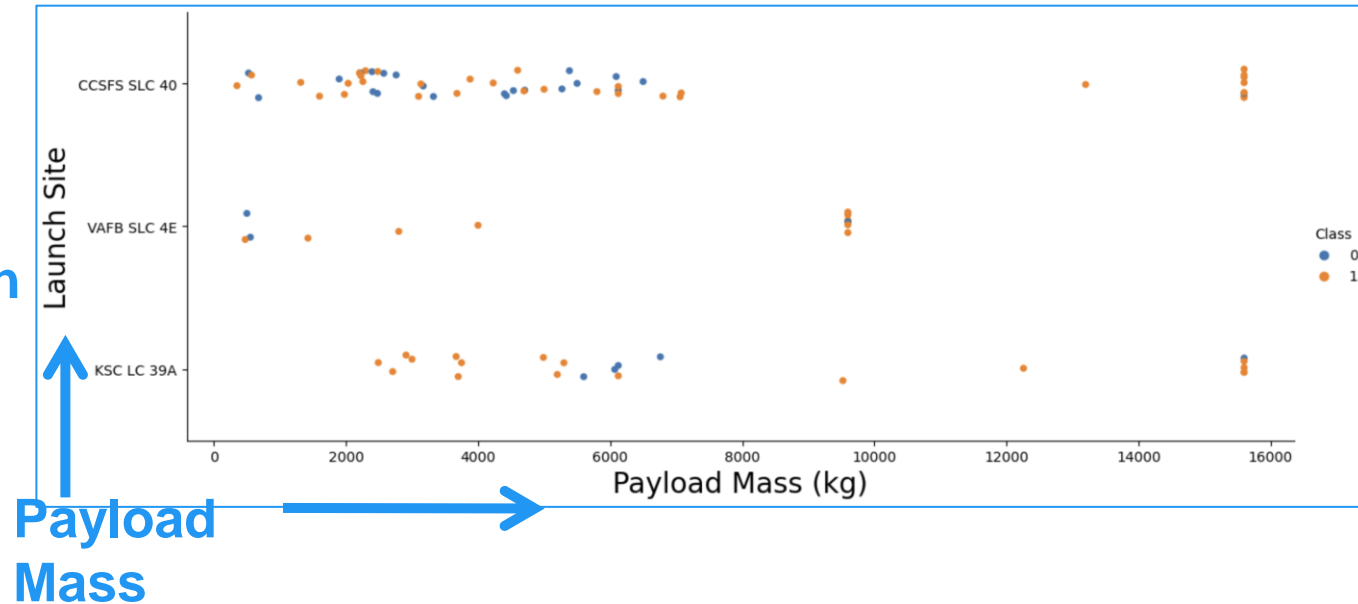


# Payload Mass vs Launch site

2010 to 2020 we have to 90 flight numbers

- As payload increases in different launch site, the success landing increase
- Payload mass below 8K in CCSFS SLC 40 has also several failure landing

Launch Site

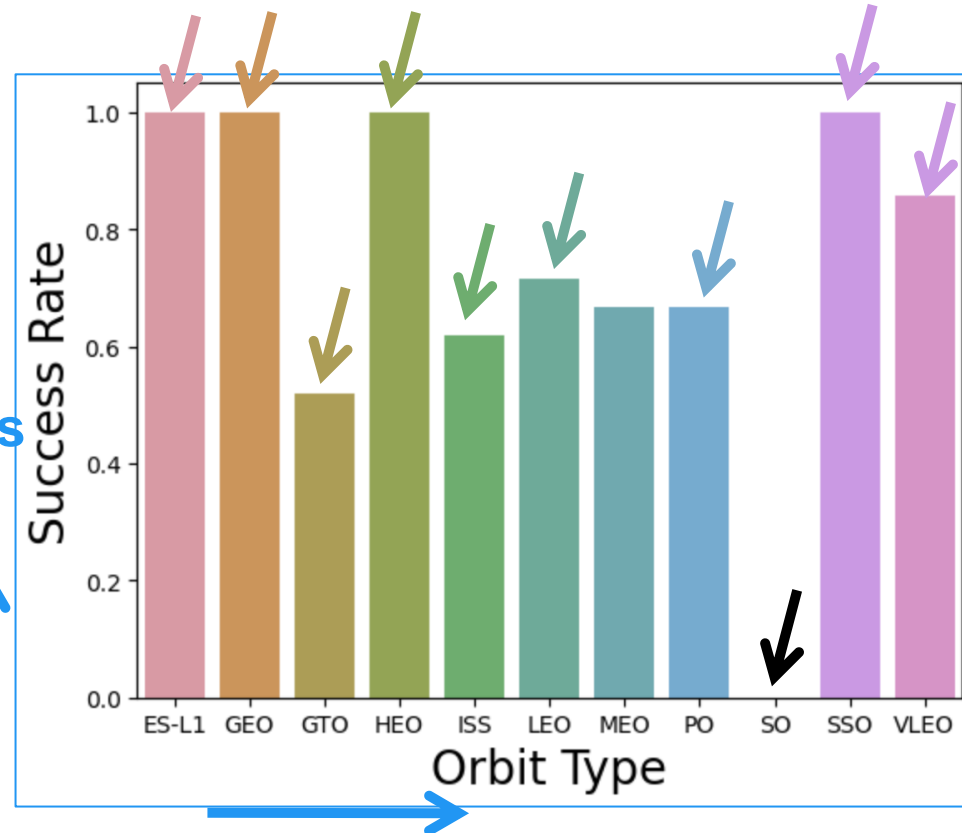


# Success Rate vs Orbit Type

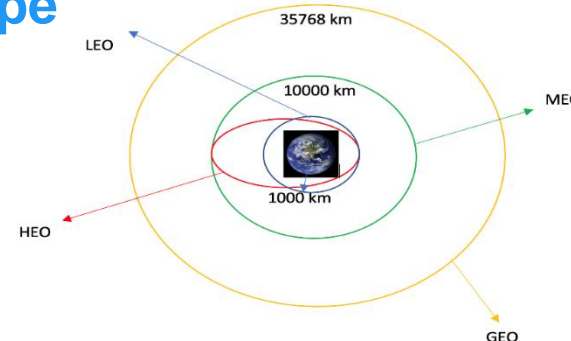
2010 to 2020 we have to 90 flight numbers

- The orbit which launches before 2015, with more than 2 launches have a success rate below 80%
- The orbit that has a 100% has only been launched once except for SO, which failed in the first launch
- SSO has 5 launches with 100% and VLEO has above 80% success rate

Success Rate



Orbit Type



Orbit	Min_Date
LEO	2010-06-04
ISS	2013-03-01
PO	2013-09-29
GTO	2013-12-03
ES-L1	2015-02-11
SSO	2017-08-24
HEO	2018-04-18
MEO	2018-12-23
VLEO	2019-05-24
SO	2020-01-19
GEO	2020-07-20

Orbit	count
GTO	27
ISS	21
VLEO	14
PO	9
LEO	7
SSO	5
MEO	3
ES-L1	1
HEO	1
SO	1
GEO	1



int64  
**Analyze**

# Flight Number vs Orbit Type

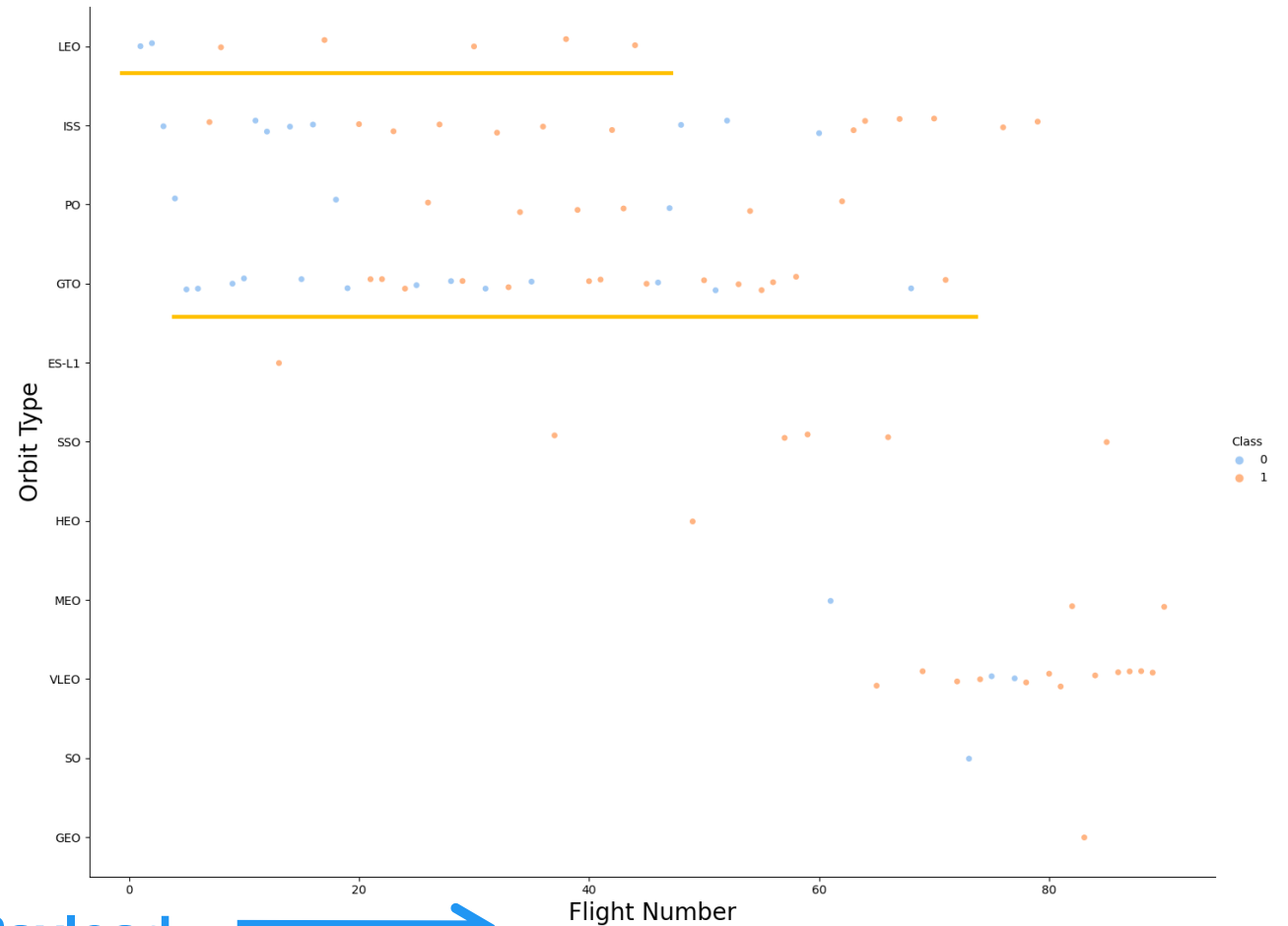
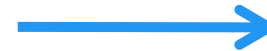
2010 to 2020 we have to 90 flight numbers

- the `LEO` orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in `GTO` orbit

Orbit Type



Payload Mass

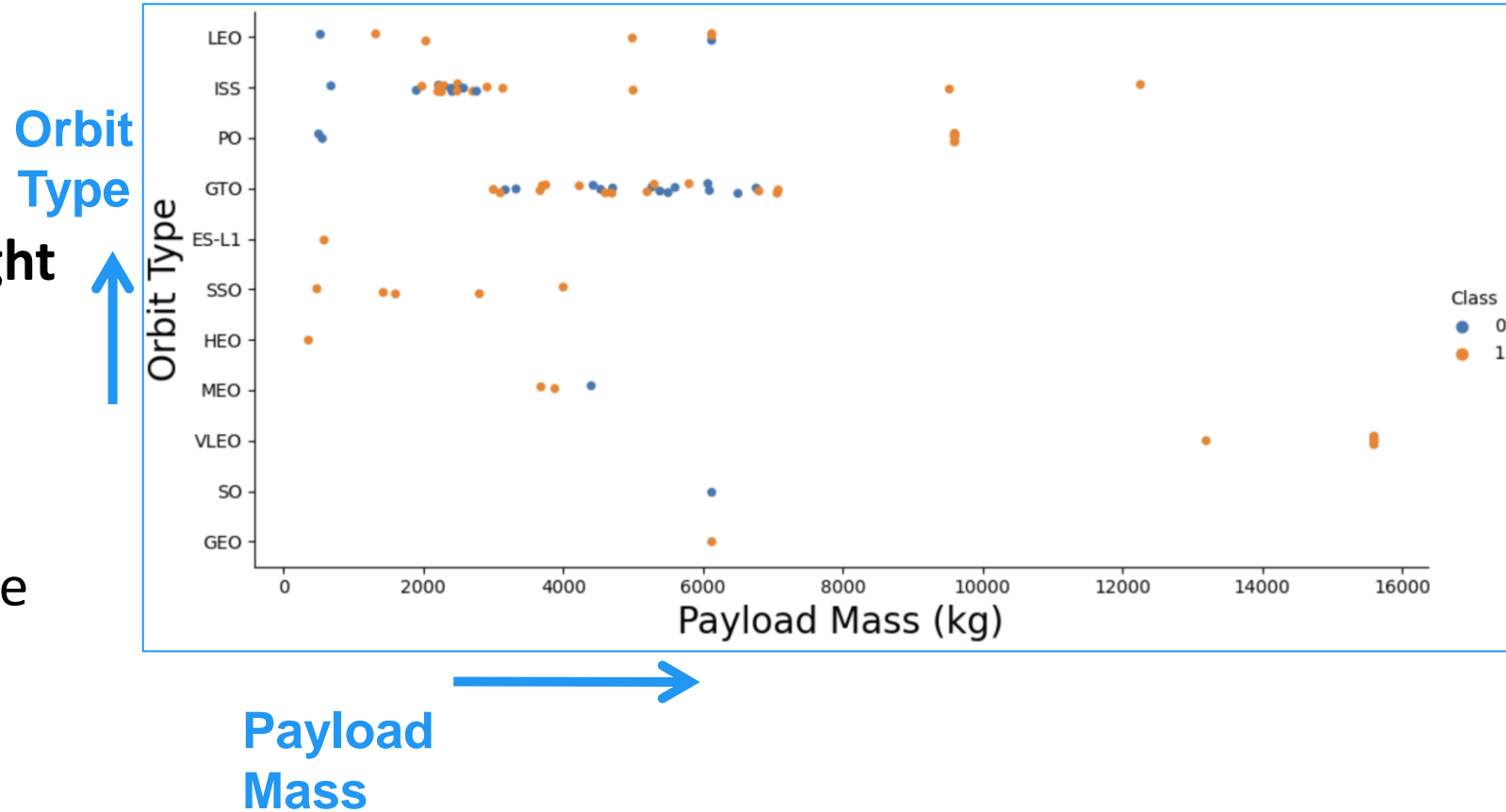




# Payload vs Orbit Type

2010 to 2020 we have to 90 flight numbers

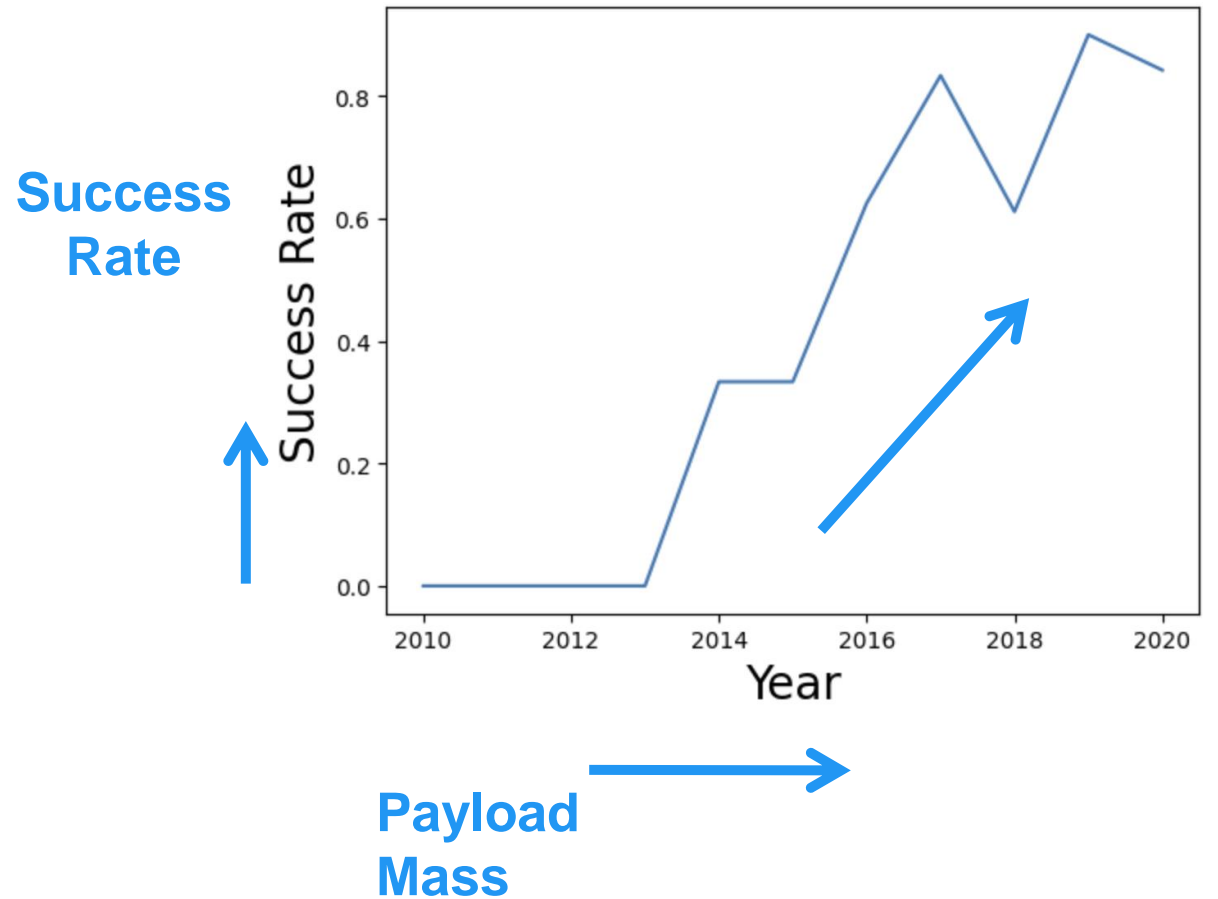
- With heavy payloads the successful landing or positive landing rate are more for `Polar`, `LEO` and `ISS`
- `GTO` we cannot distinguish this well as success and failure landing



# Launch Success Yearly Trend

**2010 to 2020 we have to 90 flight numbers**

- The success rate have been increasing since 2013 to 2020





# Analyze

## Analysis with SQL

# All Launch Site Names

**df\_1**  
4 location sites  
101 flight numbers

	Launch_Site
0	CCAFS LC-40
1	VAFB SLC-4E
2	KSC LC-39A
3	CCAFS SLC-40

Kaggle

```
df_q = pd.read_sql('select distinct Launch_Site from spacexdata', conn)
df_q
```

SQL

```
%sql select distinct(LAUNCH_SITE) from SPACEXTBL
* sqlite:///my_data1.db
```

- Using SQL Data, we acquire 4 Launch Sites from spaceX data 2010 to 2020
- The data in SQL differ from data during wrangling phase. (following the assignment instructions)

# Launch Site Names Begin with 'CCA'

SQL

Kaggle

```
%sql select * from SPACEXTBL where LAUNCH_SITE like 'CCA%' limit 5
```

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

```
df_q = pd.read_sql("select * from spacexdata where Launch_Site like 'CCA%' limit 5", conn)
df_q
```

	index	Date	Time_(UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS_KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	0	2010-06-04 00:00:00	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	1	2010-12-08 00:00:00	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)
2	2	2012-05-22 00:00:00	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
3	3	2012-10-08 00:00:00	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	4	2013-03-01 00:00:00	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- The 5 rockets launched from CCSFS LC-40 began their launches in June 2010. Two rockets were launched in 2010, and two were launched in 2012. The displayed data only shows 5 launches
- All 5 launches had a Low Earth Orbit (LEO) trajectory, and all of them were successful



# Total Payload Mass

Kaggle

	Customer	Total_Payload_Mass
0	NASA (CRS)	45596

```
df_q = pd.read_sql("select Customer, sum(PAYLOAD_MASS__KG_) as Total_Payload_Mass from spacexdata where Customer='NASA (CRS)' group by Customer", conn)
df_q
```

SQL

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where CUSTOMER = 'NASA (CRS)'
```

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

- For NASA (CRS), the total payload was 45,596 kg

# Average Payload Mass F9 v.1.1

Kaggle

	Booster_Version	Avg_Payload_Mass
0	F9 v1.1	2928.4

```
df_q = pd.read_sql("select Booster_Version, avg(PAYLOAD_MASS_KG_) as Avg_Payload_Mass from spacexdata where Booster_Version='F9 v1.1' group by Booster_Version", df_q)
```

SQL

```
%sql select avg(PAYLOAD_MASS_KG_) from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1'
```

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

- The average payload mass served by F9 v.1.1 Booster was 2,928.4 kg

# First Successful Ground Landing Date

**min(DATE)**

2015-12-22

Kaggle

```
df_q = pd.read_sql("select *, min(Date) as min_date from spacexdata where Landing_Outcome='Success (ground pad)'", conn)
df_q
```

SQL

```
%sql select min(DATE) from SPACEXTBL where Landing_Outcome = 'Success (ground pad)'
```

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

- The first successful ground landing occurred in December 2015



# Successful Drone Ship Landing with Payload between 4000 and 6000

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

Kaggle

```
df_q = pd.read_sql("select distinct Booster_Version from spacexdata where Landing_Outcome='Success (drone ship)' and PAYLOAD_MASS_KG_ between 4000 and 6000", df_q)
```

SQL

```
%sql select BOOSTER_VERSION from SPACEXTBL where Landing_Outcome = 'Success (drone ship)' and PAYLOAD_MASS_KG_ > 4000 and  
* sqlite:///my_data1.db
```

- For payload masses between 4,000 and 6,000 kg, the Falcon 9 FT Booster with version B10 includes series of 22, 26, 21.2, and 31.2

# Total Number of Successful and Failure Mission Outcomes

	Mission_Outcome	count(*)
0	Failure	1
1	Success	100

Kaggle

```
df_q = pd.read_sql("select substr(Mission_Outcome,1,7) as Mission_Outcome, count(*) from spacexdata group by 1", conn)
df_q
```

SQL

```
%sql select case when Mission_Outcome like 'Success%' then 'Success' else Mission_Outcome end as outcome,COUNT(*) as total_count from S
* sqlite:///my_data1.db
```

- Total Successful mission 100 and 1 is a failure mission

# Booster Carried Maximum Payload

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

Kaggle

```
df_q = pd.read_sql("select distinct Booster_Version from spacexdata where PAYLOAD_MASS_KG_ = (select max(PAYLOAD_MASS_KG_) from spacexdata)", conn)
df_q
```

SQL

```
%sql select BOOSTER_VERSION from SPACEXTBL where PAYLOAD_MASS_KG_ = (select max(PAYLOAD_MASS_KG_) from SPACEXTBL)
```

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

- The Falcon 9 B5 Booster which carried maximum payload version B10 includes series of 48 to 60

# 2015 Launch Records

	Month	Landing_Outcome	Booster_Version	Launch_Site
0	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
1	04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Kaggle

```
df_q = pd.read_sql("select distinct Landing_Outcome, Booster_Version, Launch_Site from spacexdata where Landing_Outcome='Failure (drone ship)'", conn)
df_q
```

SQL

```
%sql select substr(Date, 6, 2) as Month, substr(Date, 0, 5) as Year, Landing_Outcome, Booster_Version, Launch_Site from SPACEX
* sqlite:///my_data1.db
```

- In 2015 there was two failure (drone ship) launches, both in CCAFS LC-40 with booster version v1.1 B10 serie 12 and 15

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

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

Kaggle

```
df_q = pd.read_sql("select Landing_Outcome, count(*) from spacexdata where Date between '2011-06-04' and '2017-03-20' group by Landing_Outcome order by 2 desc", df_q)
```

SQL

```
%sql select Landing_Outcome, count(*) from SPACEXTBL where Date between '2011-06-04' and '2017-03-20' group by Landing_Outcome order by 2 desc
```

- The most landing outcome occurrence of 10 due to no attempt
- Drone ship has success and failure tied in 5 times
- Success landed in ground pad also draw to Controlled in ocean with 3 times

A satellite view of Earth from space, showing the curvature of the planet and the transition from a deep blue sky to a lighter blue atmosphere. The surface of the Earth is visible, with dark blue oceans and patches of white clouds. In the lower right, there are bright, yellowish-orange lights representing city lights or urban areas.

## Section 3. Map

### Launch Sites Proximity Analysis



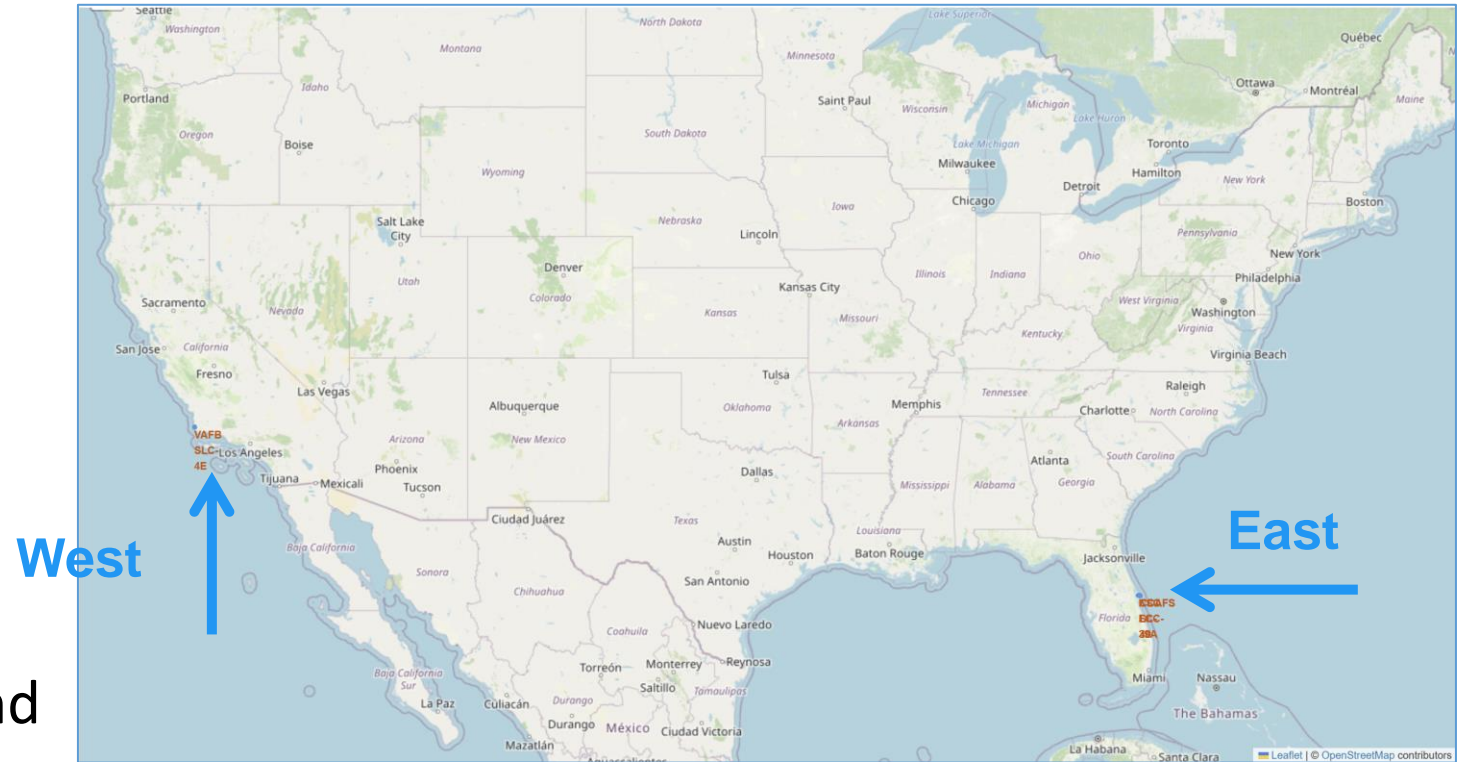
Analyze

Map Visualization

# Launch Site Map

## We identify 2 Launch Area

- West, VAFB SLC-4E on Vanderberg State
- East, KSC LC-39A on Kennedy and the CCAFS SLC-40 and CCAFS LC-40 on Cape Canaveral



df

4 location sites

56 flight numbers

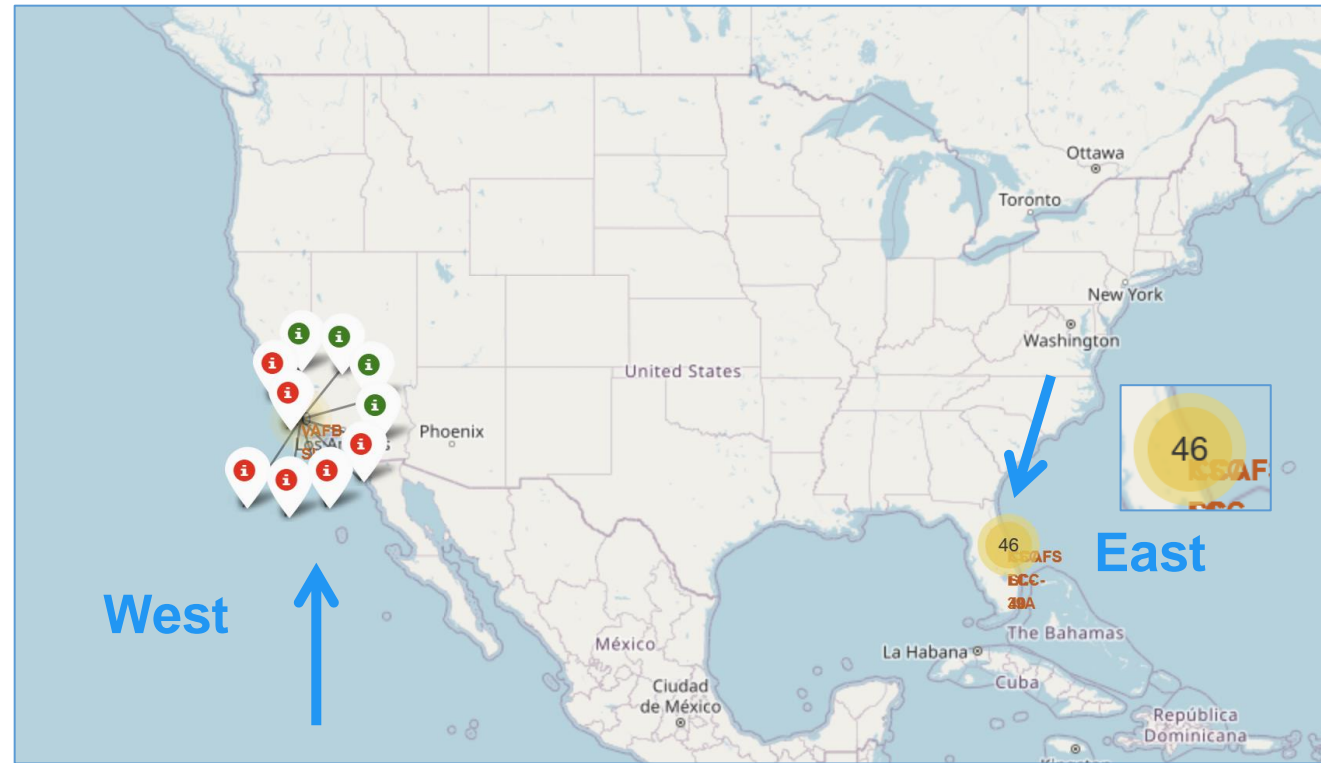


# West Launch Site Map

## In the West Area (VAFB SLC-4E )

- The map shows there is 10 launches
- 4 successful first landing or 40% success rate

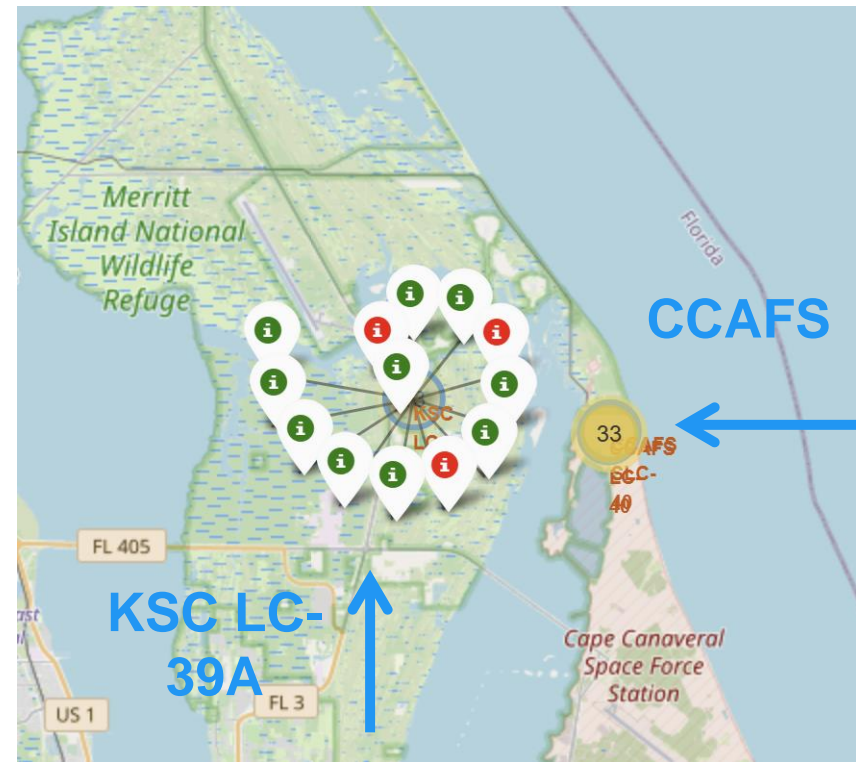
It shows 46 launches occurred in East Area



# East Launch Site Map

## In the East Area (KSC LC-39A )

- 10 successful first landing or more than 70% success rate



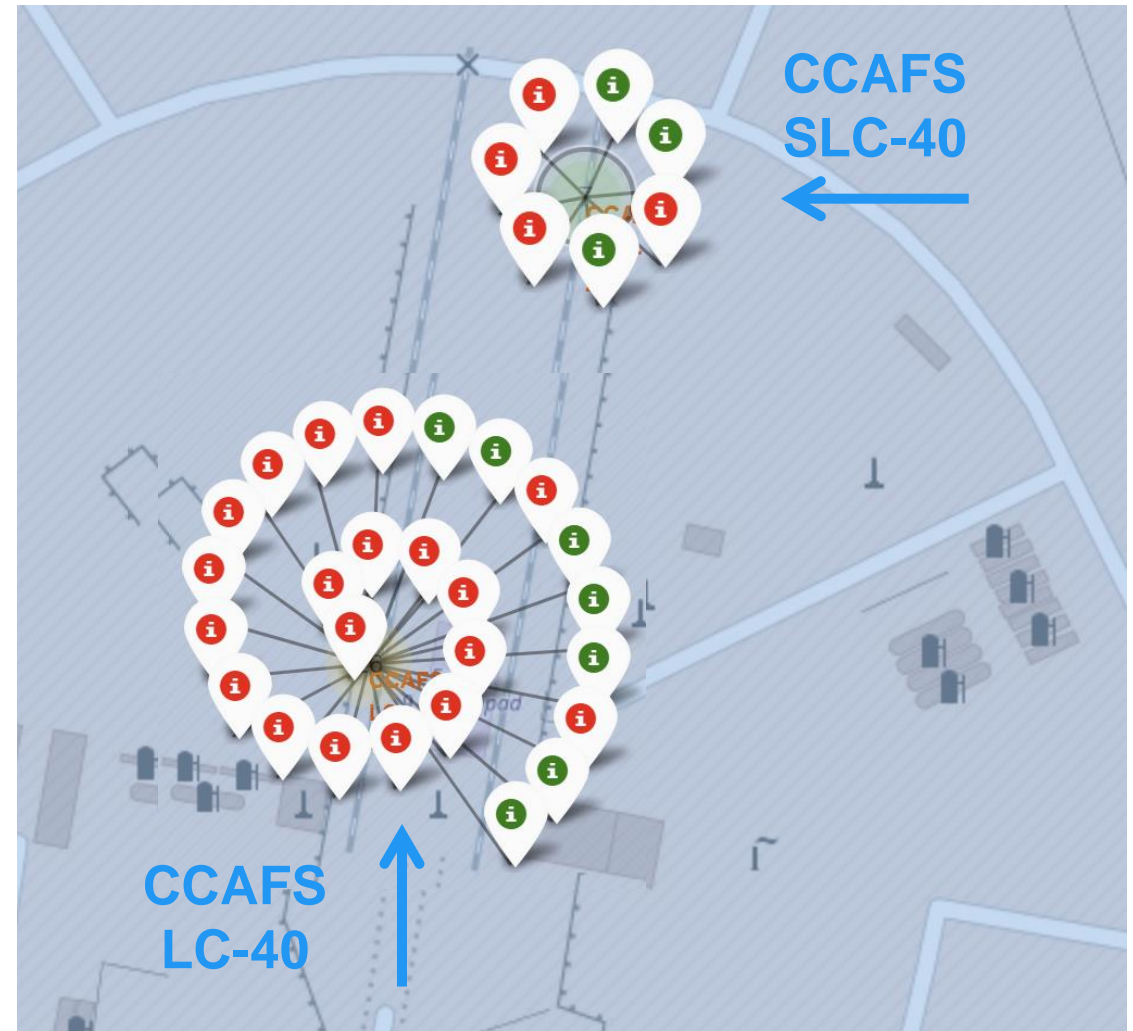
# East Launch Site Map

## In the East Area (CAFS LC-40 )

- 7 successful first landing or less than 30% success rate

## In the East Area (CAFS SLC-40 )

- 3 successful first landing or less than 50% success rate



# East Launch Site Map

## In the East Area (KSC LC-39A )

- The distance 7,8 km to the CAFS SLC-40 nearest Coastline

## In the East Area (CAFS SLC-40 )

- The distance 0,87 km to its nearest coastline



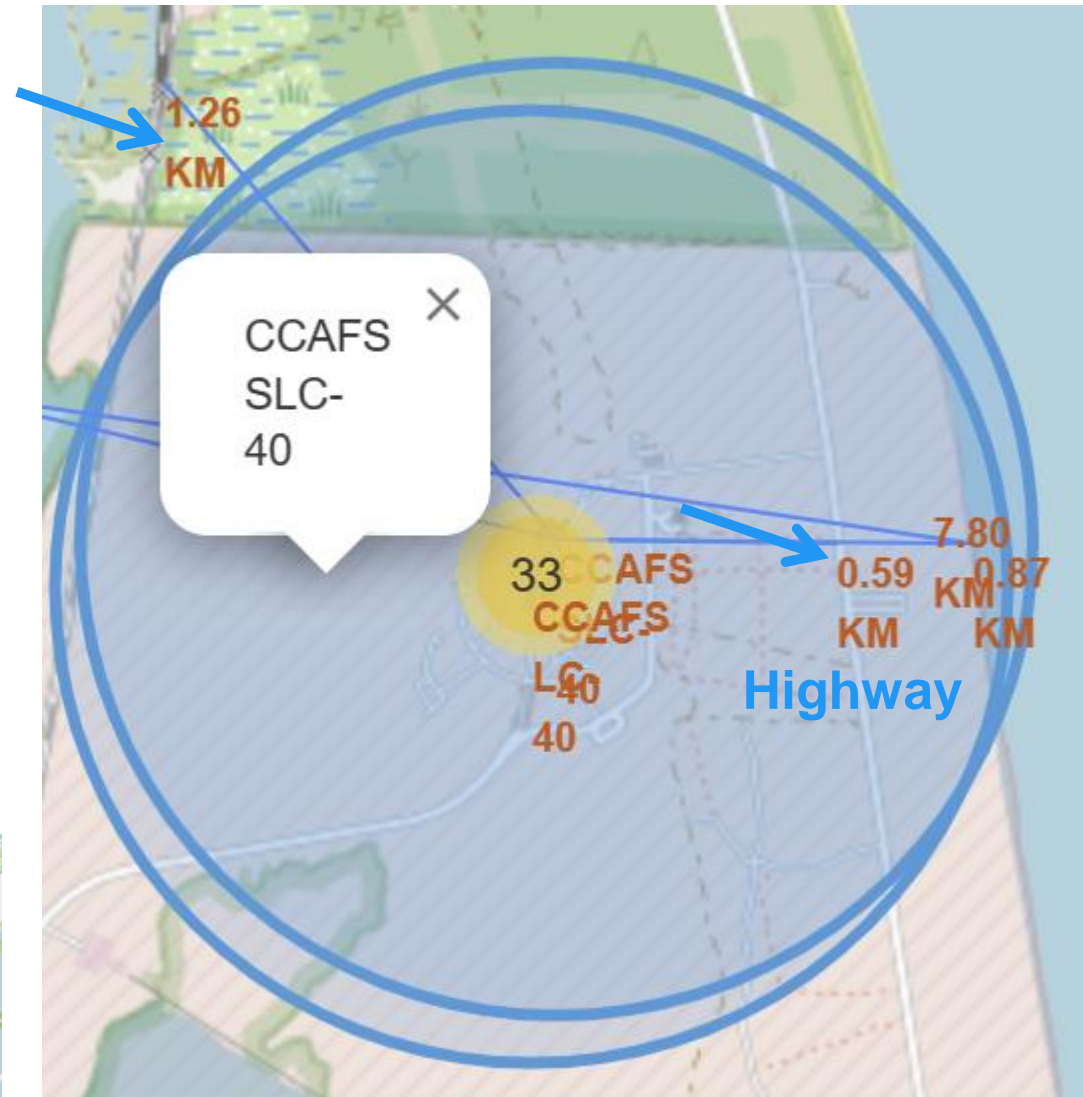


# CCAFS LC-40 Launch Site Map

## In the East Area (CAFS SLC-40 )

- The distance 1,26 km to its nearest **railway**,  
0,59km to its nearest **highway** and 23,47 km  
to its nearest **city**

Railway





# Section 4. Dashboard

## Build a Dashboard with Plotly Dash



# Analyze

## Interactive Dashboard

# All Site Success Launches

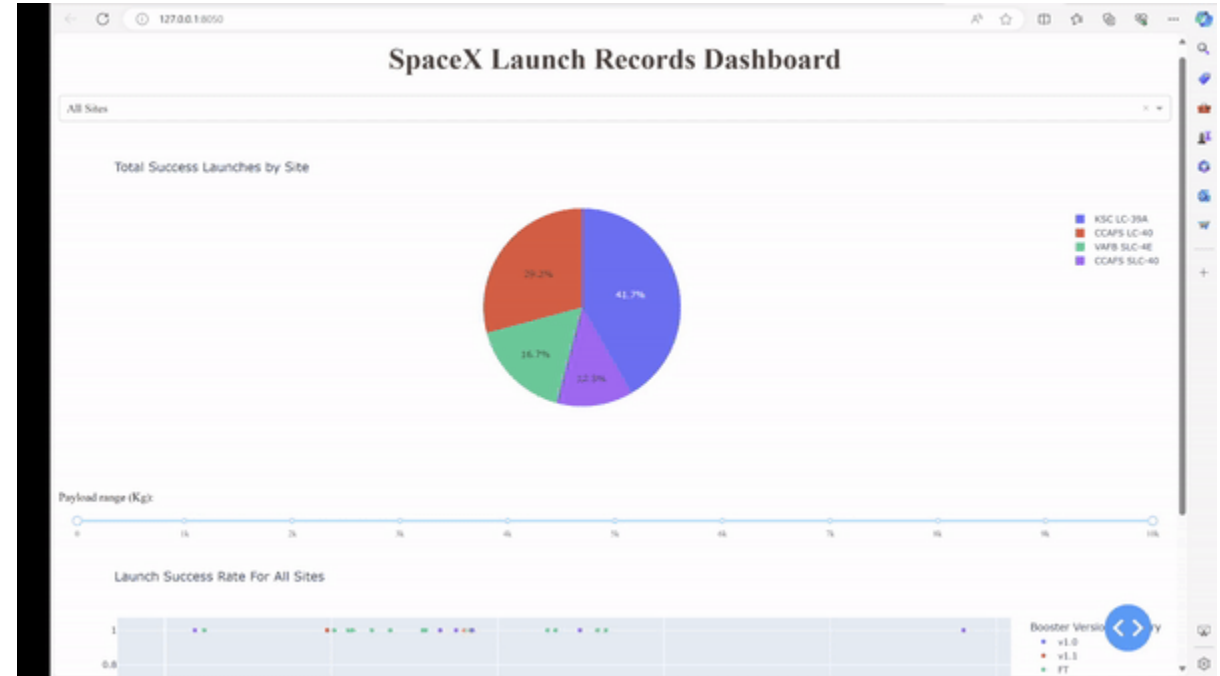
## Total Success Launches by Site

- KSC LC-39A has the highest success 40% of total launches compare the others

df

4 location sites

56 flight numbers

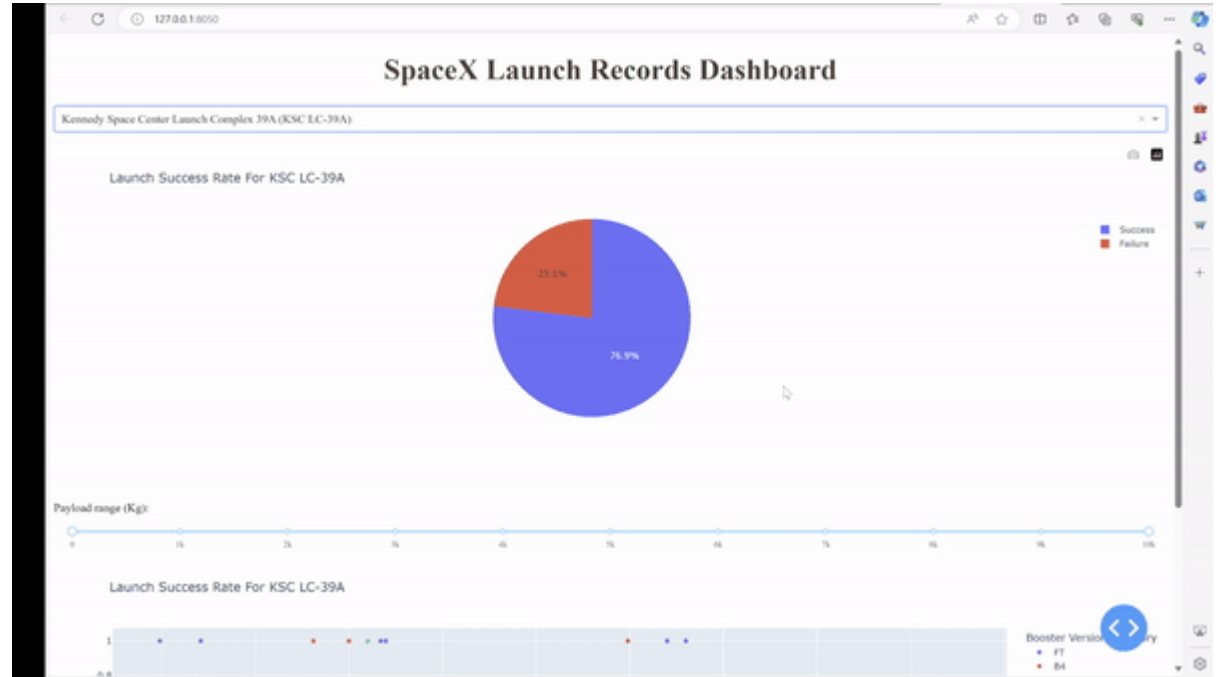




# The Highest Success Launches

## Total Success Launches by Site

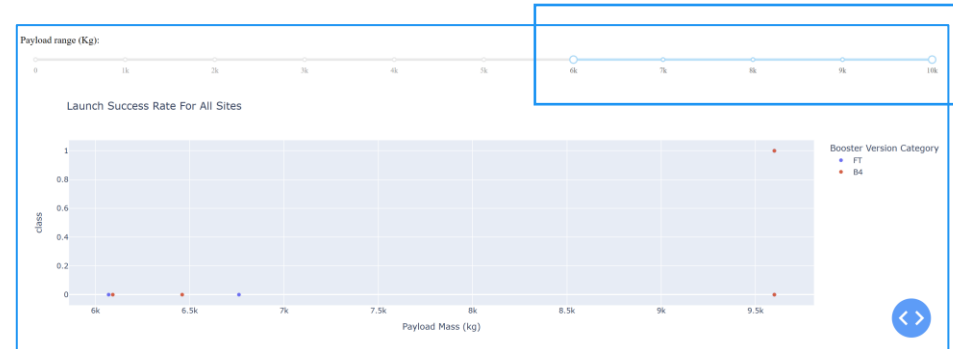
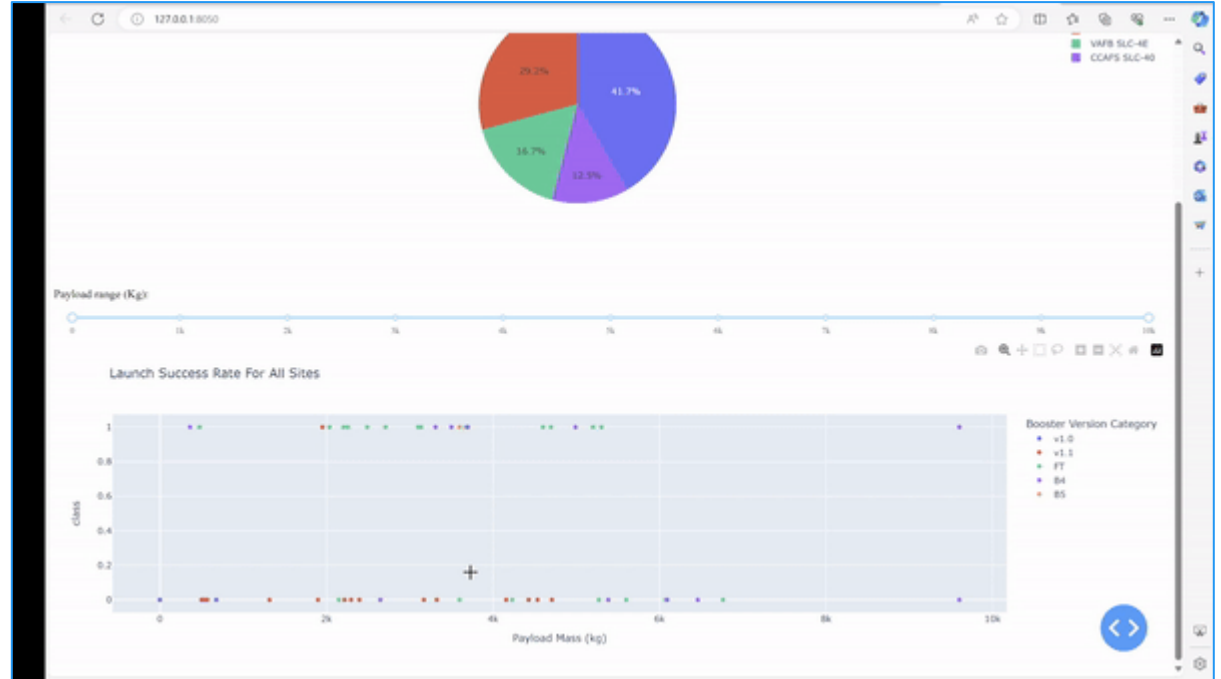
- KSC LC-39A has the highest success rate of 70%



# Payload vs Launch Outcome

## In regards to payload mass

- FT and B4 have more successful landing
- Range from 0 to 6K and above 8K, the number of successful landing and failure is almost equal.
- Between 6k to 8k, 3 fail launches.
- No correlation to landing success





# Section 5. Model

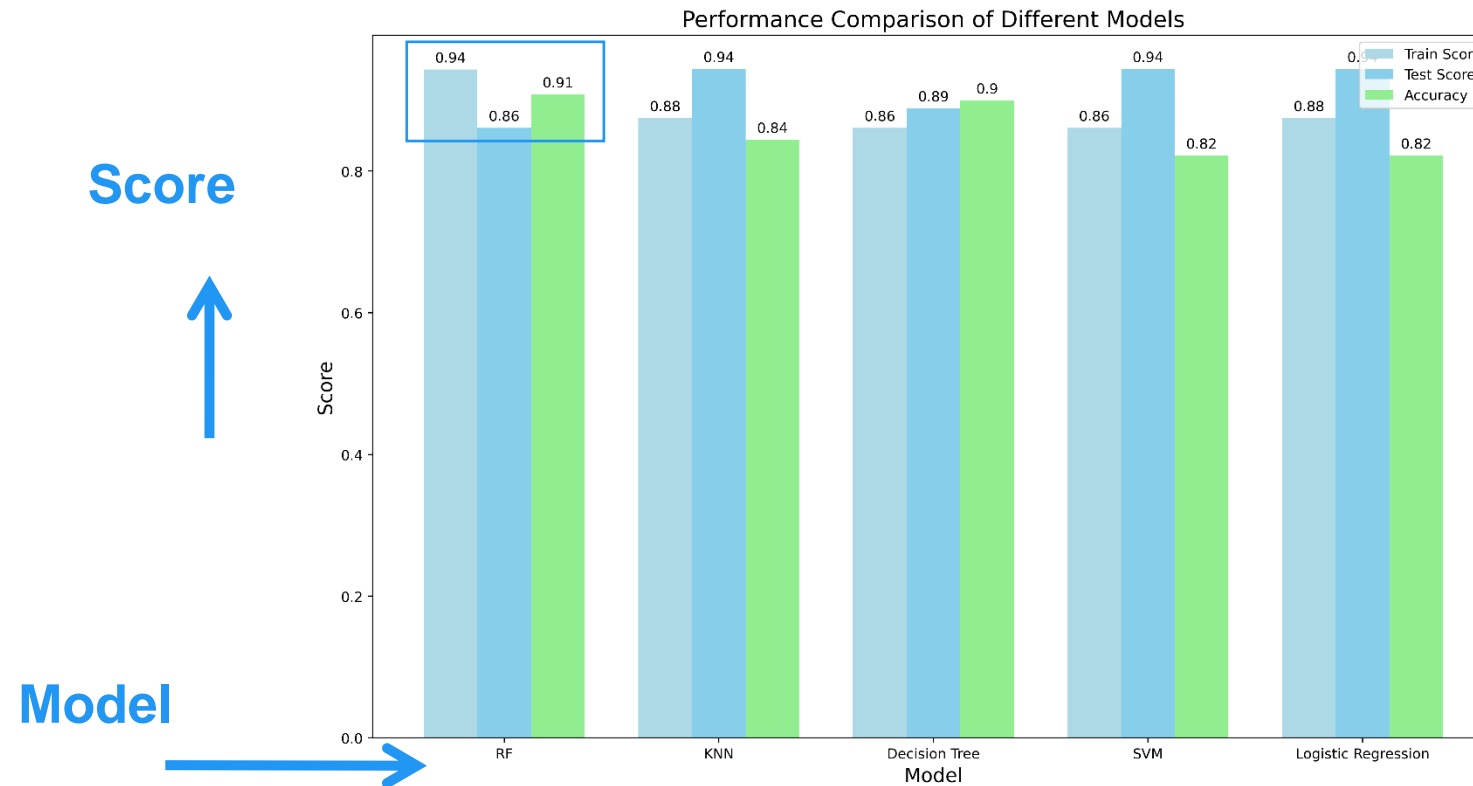
## Predictive Analysis (Classification)



# Models and Findings

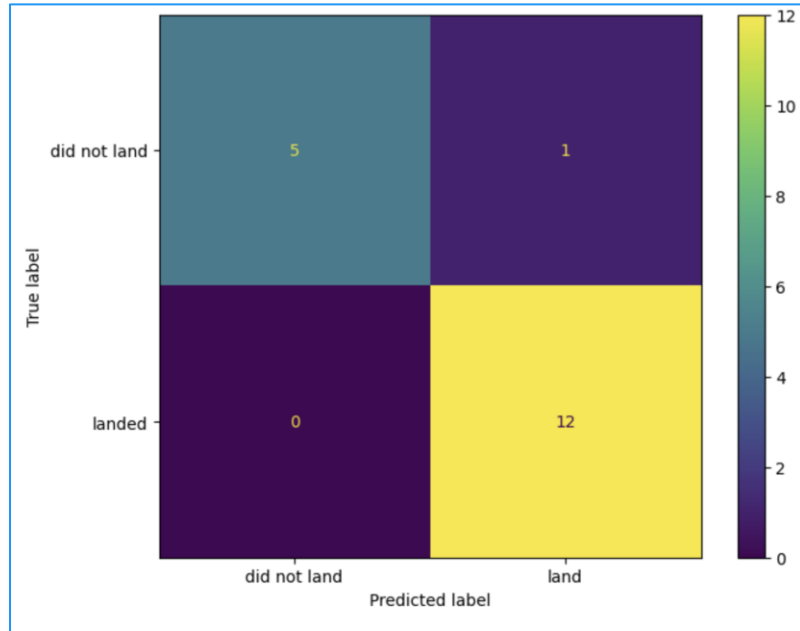
# Classification Accuracy

df  
3 location sites  
90 flight numbers



1. KNN, SVM and Logistic Regression performs reasonably well on the test set with a high score. However, the accuracy is slightly lower compared to some other models.
2. KNN, SVM and Logistic Regression have high test scores but may be overfitting.
3. Decision Tree and Random Forest have balanced performance.

# Confusion Matrix



	precision	recall	f1-score	support
did not land	1.00	0.50	0.67	6
land	0.80	1.00	0.89	12
accuracy			0.83	18
macro avg	0.90	0.75	0.78	18
weighted avg	0.87	0.83	0.81	18

[[ 3  3]
[ 0 12]]

Accuracy Score: 0.83  
Mean Cross-Validation Accuracy: 0.82  
ROC AUC: 0.87

**df**  
3 location sites  
90 flight numbers

## Confusion Matrix:

- In this case, there are 3 instances of "did not land" correctly classified as such, 3 instances incorrectly classified as "land,"
- 12 instances of "land" correctly classified as such, with 0 instances incorrectly classified as "did not land."

Overall, a good performance with high precision and recall for the "land" class and slightly lower performance for the did not "land" class

## Evaluation Metrics:

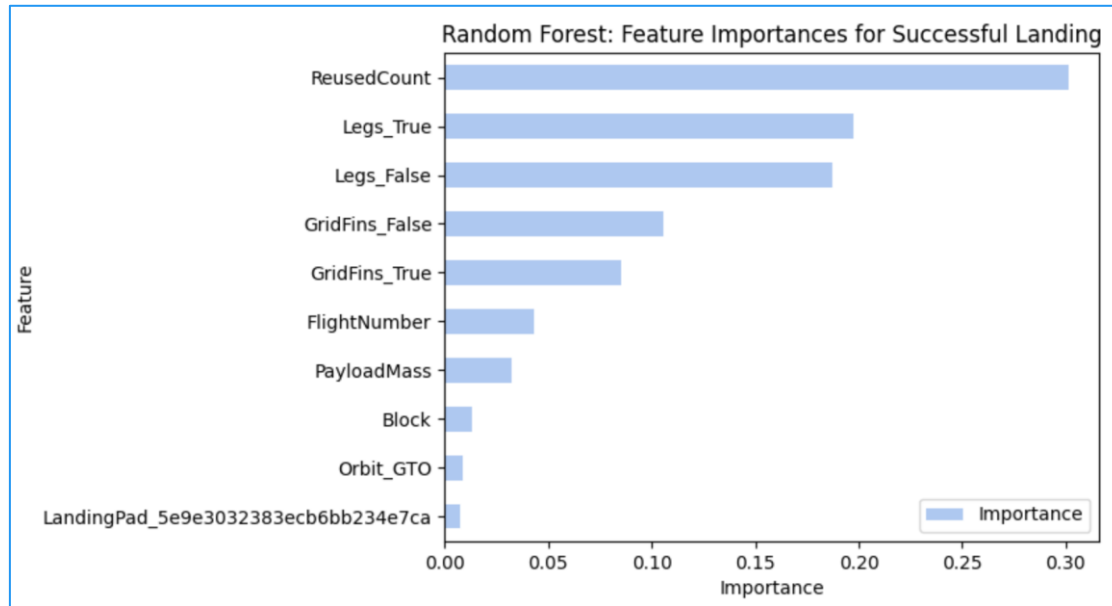
1. The accuracy 83% of instances were correctly classified
2. 82% suggesting consistent performance across subset data
3. Good discrimination between positive and negative classes to 87%

The model performs well overall and generalizes effectively to unseen data.

df

3 location sites  
90 flight numbers

# Feature Importance



The feature importance of the Random forrest as follows:

- 1.Reused count
- 2.Legs
- 3.Grid fins
- 4.Flight number
- 5.Payload mass
- 6.Block
- 7.Orbit GTO
- 8.Landing pad with last 3 numbers or letters 7ca



# Conclusion





# Conclusion

---

1. We used different data across the assignments. The number of successful landings increase with the flight number increases
2. The data in SQL we acquired 4 launch sites. Space X start operating in 2010 and first successful landing in 2015.
3. The highest success rate is KSC LC-39A
4. We decided to choose **Random forest** which performs well on the test set 94% with the highest accuracy 91% score among all models. The feature importance of the Random forest as follows:
  - Reused count
  - Legs
  - Grid fins
  - Flight number
  - Payload mass
  - Block
  - Orbit GTO
  - Landing pad with last 3 numbers or letters 7ca

## Next Steps

- Considering XGBoost and Kmeans
- Analyze the feature importance from random forest further
- Review the hyperparameter tuning



# Appendix

# Appendix

## Documents:

- <https://www.kaggle.com/code/wahyuardhitama/Task008-P01-DS-SpaceX-Falcon-20240409>
- <https://github.com/whyzie/Task008-P01-DS-SpaceX-Falcon-20240409>