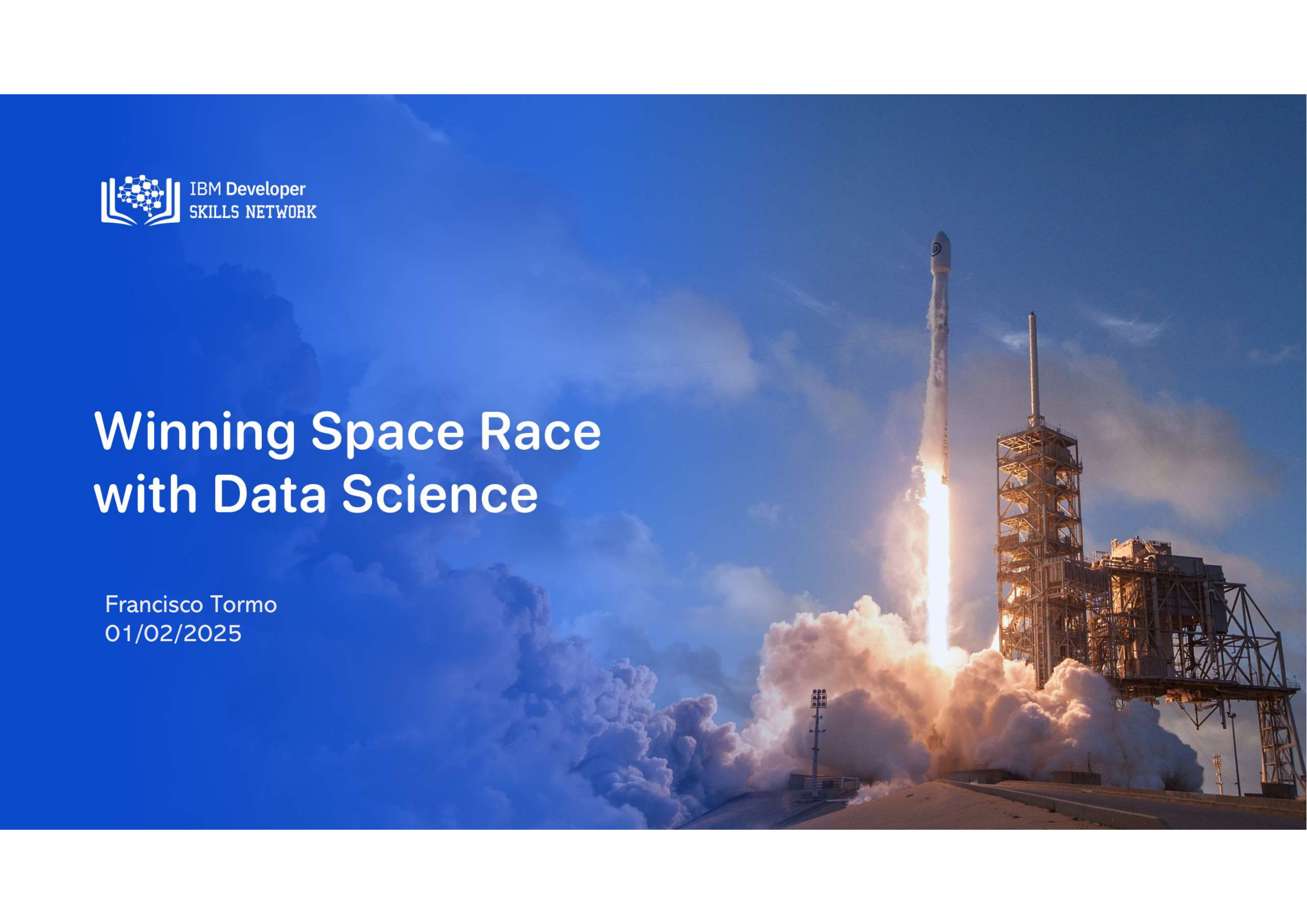




Winning Space Race with Data Science

Francisco Tormo
01/02/2025



Outline

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

Executive Summary

- Datos recopilados de la API pública de SpaceX y de Wikipedia. La etiqueta “Class” distingue los aterrizajes con éxito. Búsqueda de datos con SQL, visualización con mapas de folium y dashboards. Localización de columnas relevantes para usarlas como características. Conversión de variables categóricas a binarias. Estandarizar los datos y usar GridSearchCV, encontrar los mejores parámetros en los modelos de aprendizaje. Comprobar la precisión de los modelos.
- Cuatro modelos de aprendizaje: Logistic Regression, Support Vector Machine, Decision Tree Classifier and K Nearest Neighbors. En todos los modelos los resultados son similares, 83,33%. Todos predijeron aterrizajes con éxito con más precisión. Se requiere más datos para mejorar la precisión del modelo.

Introduction

Antecedentes

- Space X empresa competitiva, 62 millones frente a 165 de otras
- Space Y quiere competir en la Carrera Espacial
- Space Y quiere conocer factores de éxito, recuperación Stage 1

Problemas

- Space Y quiere un modelo automático para predecir la recuperación de Stage 1



Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
 - Datos combinados de la API pública de SpaceX y la página de Wikipedia de SpaceX
- Perform data wrangling
 - Clasificar los aterrizajes verdaderos como exitosos y no exitosos en caso contrario
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
 - Modelos ajustados mediante GridSearchCV

Data Collection

La recopilación de datos es una combinación de Peticiones a la API pública de Space X y datos extraídos de la web de Space X de Wikipedia.

En las siguientes slides vemos:

Diagrama de flujo, extracción datos API Space X

Diagrama de flujo, extraccion datos web Space X Wikipedia

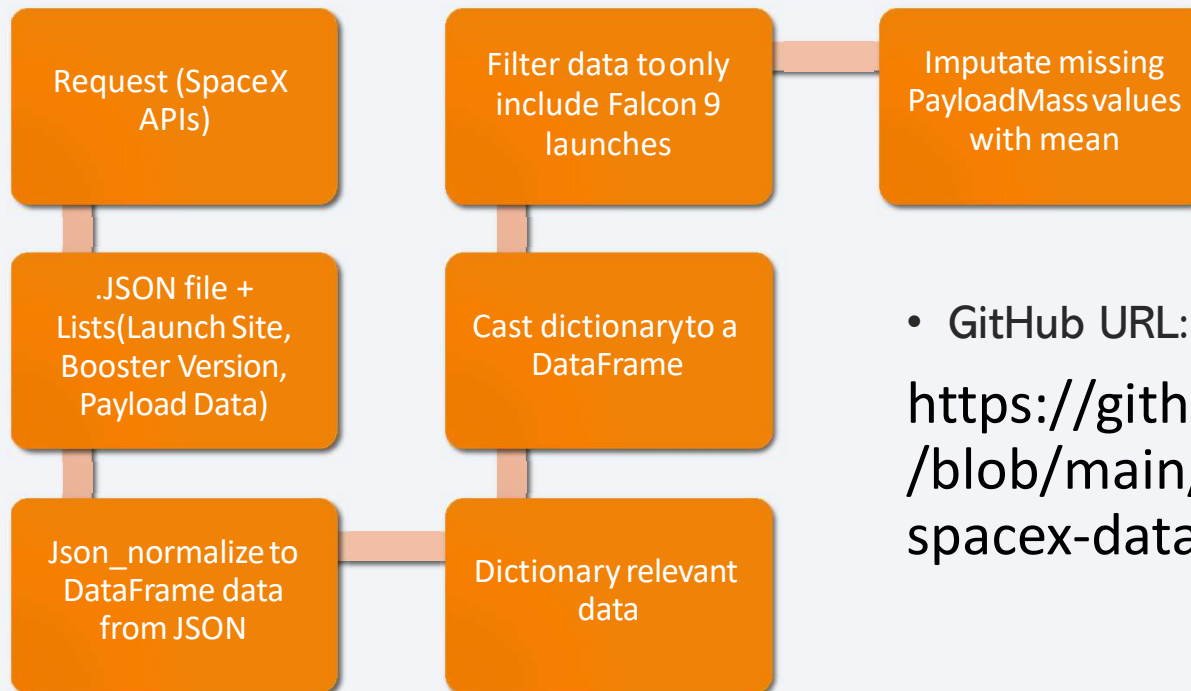
Columnas API Space X

FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Columnas Scraping Space X Wikipedia

Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

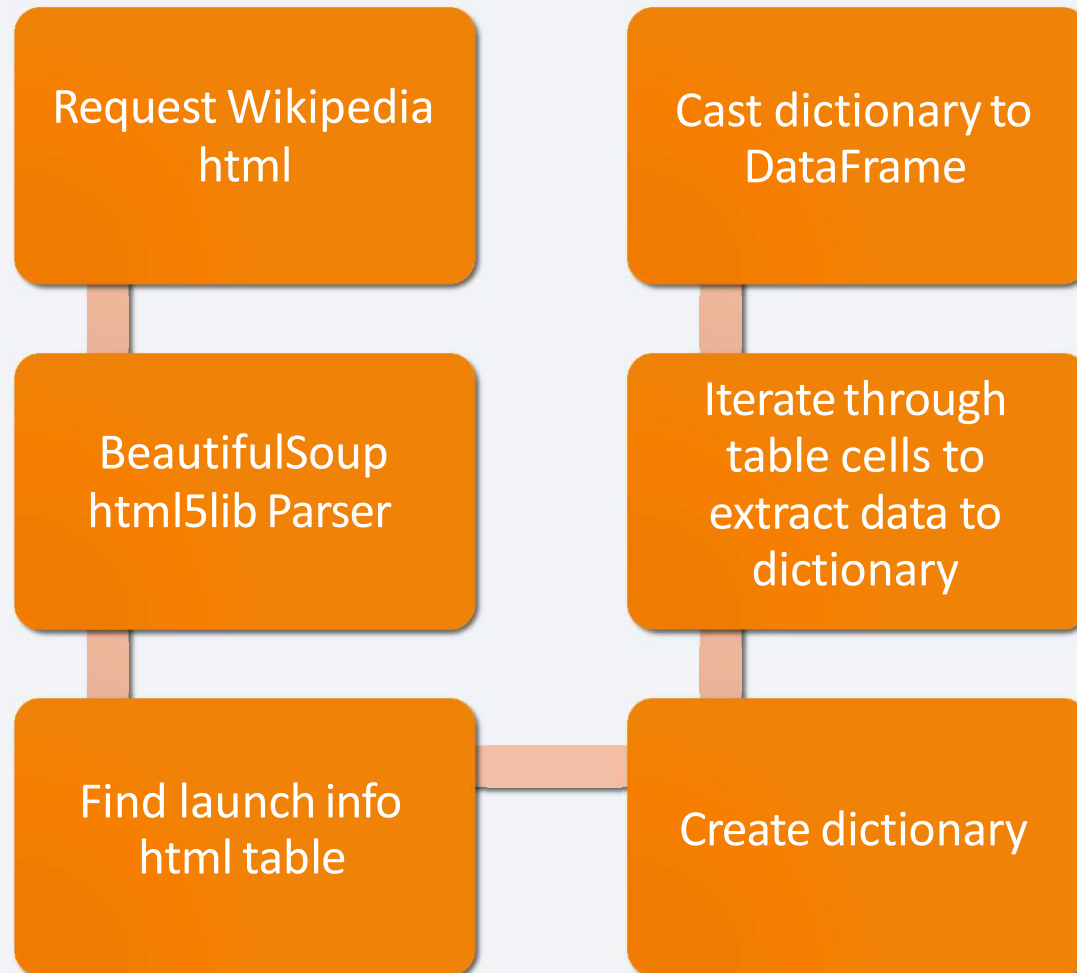
Data Collection – SpaceX API



- GitHub URL:

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)jupyter-labs-spacex-data-collection-api.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)jupyter-labs-spacex-data-collection-api.ipynb)

Data Collection - Scraping



GitHub URL:

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)jupyter-labs-webscraping.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)jupyter-labs-webscraping.ipynb)

Data Wrangling

Crear Etiqueta de Entrenamiento:

Exito = 1, Fracaso = 0

Columna de Resultado:

Resultado de la Misión

Ubicación de Aterrizaje

Nueva Columna "Class", según resultado de la mission 1 o 0

Asignación de Valores

True ASDS, True RTLS, True Ocean poner 1

None None, False ASDS, None ASDS, False Ocean, False RTLS poner 0

GitHub URL

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)labs-jupyter-spacex-Data%20wrangling.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)labs-jupyter-spacex-Data%20wrangling.ipynb)

EDA with Data Visualization

Análisis exploratorio de datos realizado en las variables Número de vuelo, Masa de la carga útil, Lugar de lanzamiento, Órbita, Clase y Año.

Gráficos:

- Flight Number – Payload Mass

- Flight Number – Launch Site

- Payload Mass – Launch Site

- Orbit – Success Rate

- Flight Number – Orbit

- Payload – Orbit

- Success Yearly Trend

GitHub URL:

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)jupyter-labs-eda-dataviz-v2.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)jupyter-labs-eda-dataviz-v2.ipynb)

EDA with SQL

Cargar los datos en sqlite3 “my_data1.db”

Realizar consultas con SQL Python

Comprender el conjunto de datos:

- Sitios de Lanzamiento

- Resultados de la Misión

- Tamaños de Carga Util

- Versiones de Propulsores

- Resultado de Aterrizaje

GitHub URL:

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)jupyter-labs-eda-sql-coursera_sqlite.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)jupyter-labs-eda-sql-coursera_sqlite.ipynb)

Build an Interactive Map with Folium

Los Mapas de Folium marcan lo siguiente:

- Lugares de Lanzamiento

- Aterrizajes Éxito o Fallido

- Proximidades Claves: Ferrocarril, autopista, costa y ciudad

Permite entender cuales son los lugares de lanzamiento con y donde está ubicados.
Visualizar los aterrizajes con éxito relacionado con la localización en el mapa.

GitHub URL:

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)lab_jupyter_launch_site_location.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)lab_jupyter_launch_site_location.ipynb)

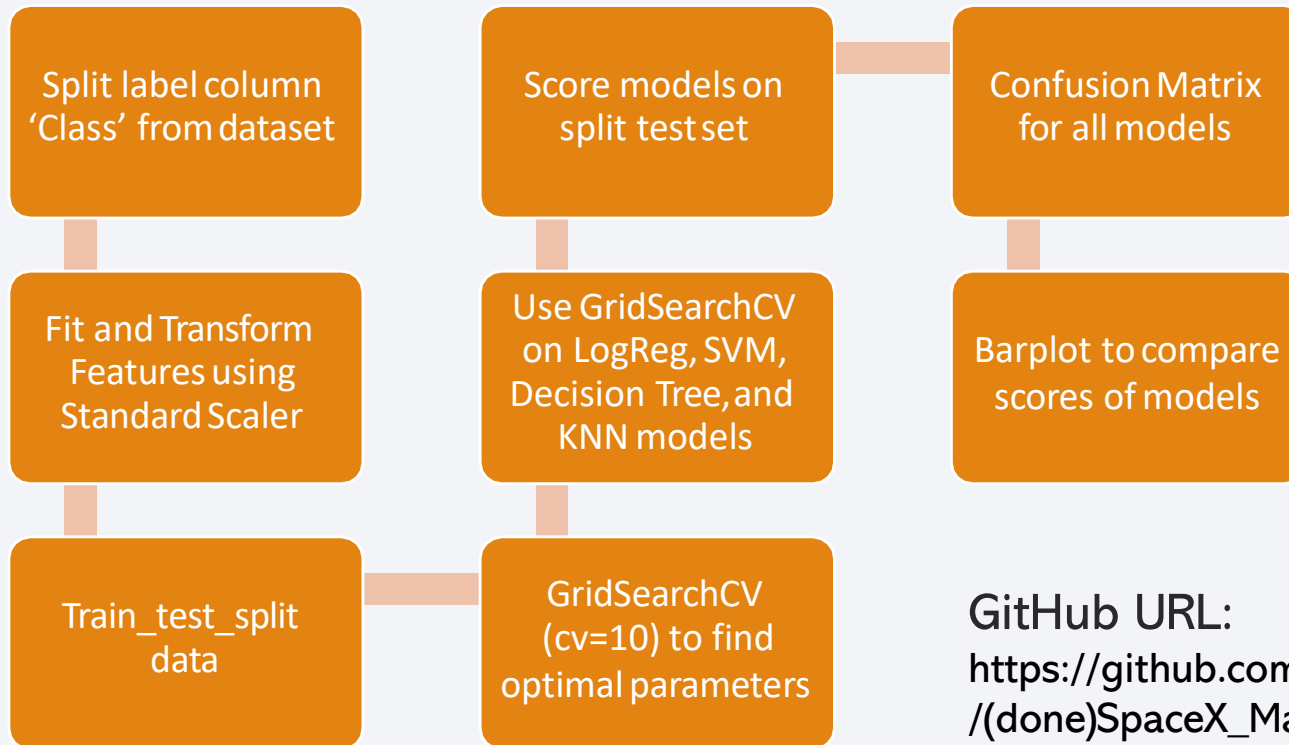
Build a Dashboard with Plotly Dash

- Dashboard contiene:
 - Gráfico Circular (Pie Chart)
 - Diagrama de Dispersión (Scatter Plot)
- Gráfico Circular, permite selección:
 - Mostrar distribución aterrizajes con éxito, sitios lanzamiento
 - Mostrar tasas de éxito sitios lanzamiento individuales
 - Visualiza la tasa de éxito del lugar de lanzamiento
- Gráfico Dispersión, permite selección:
 - Todos los sitios o uno
 - Masa de Carga Útil
 - Ayuda a ver como varia éxito de lanzamiento, según lugar, masa carga útil y version propulsor

GitHub URL:

https://github.com/Ftormogil/spaceY/blob/main/spacex_dash_app.py

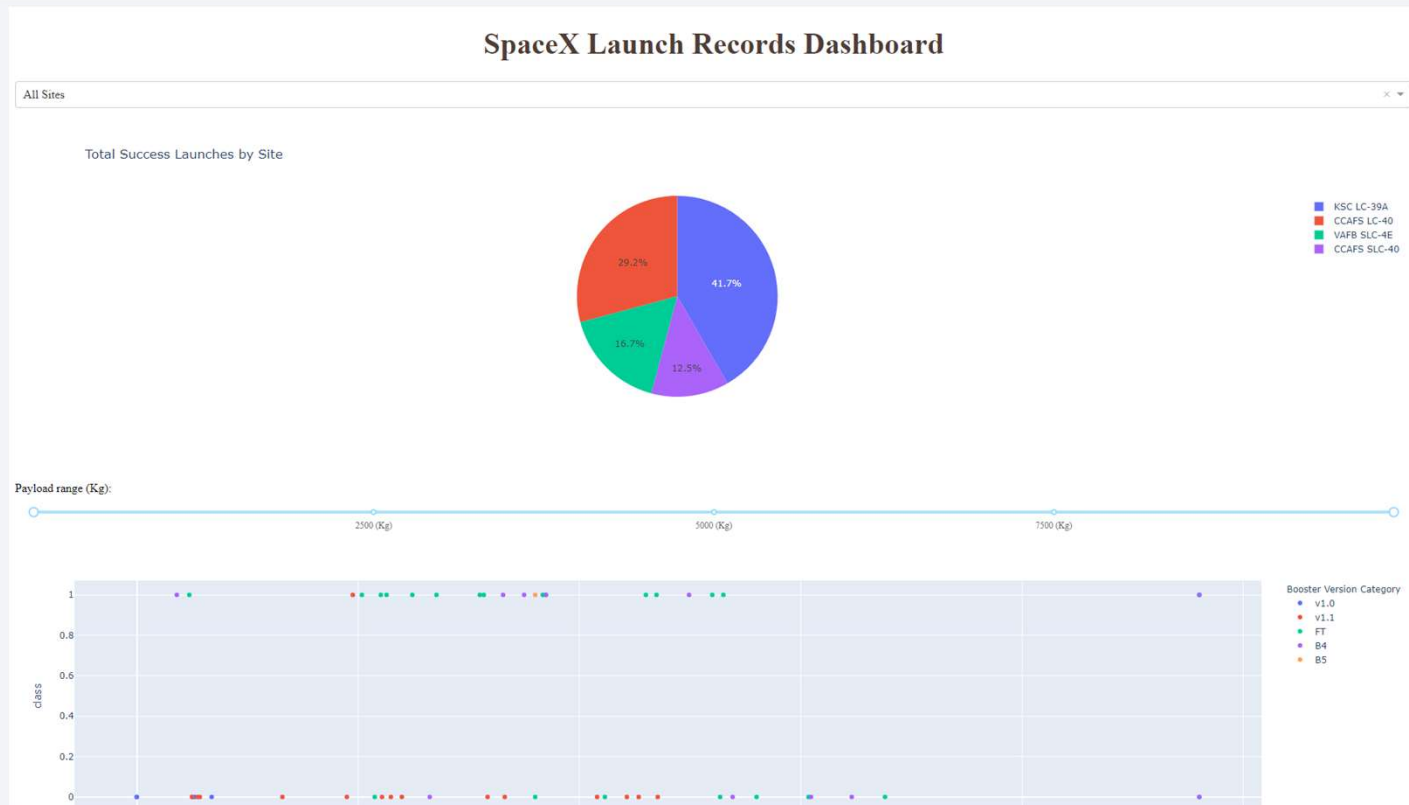
Predictive Analysis (Classification)



GitHub URL:

[https://github.com/Ftormogil/spaceY/blob/main/\(done\)SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb](https://github.com/Ftormogil/spaceY/blob/main/(done)SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb)

Results



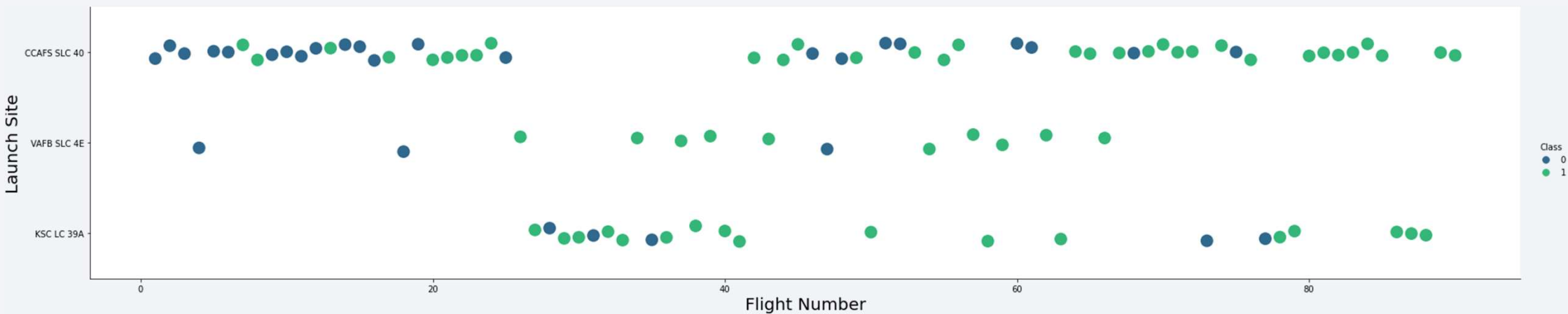
Vista previa Panel Plotly.

The background of the slide is an abstract composition. It features a solid blue area on the left side, which transitions into a complex pattern of diagonal streaks in shades of blue, red, and cyan on the right. These streaks are layered over a fine, grid-like texture, creating a sense of depth and movement.

Section 2

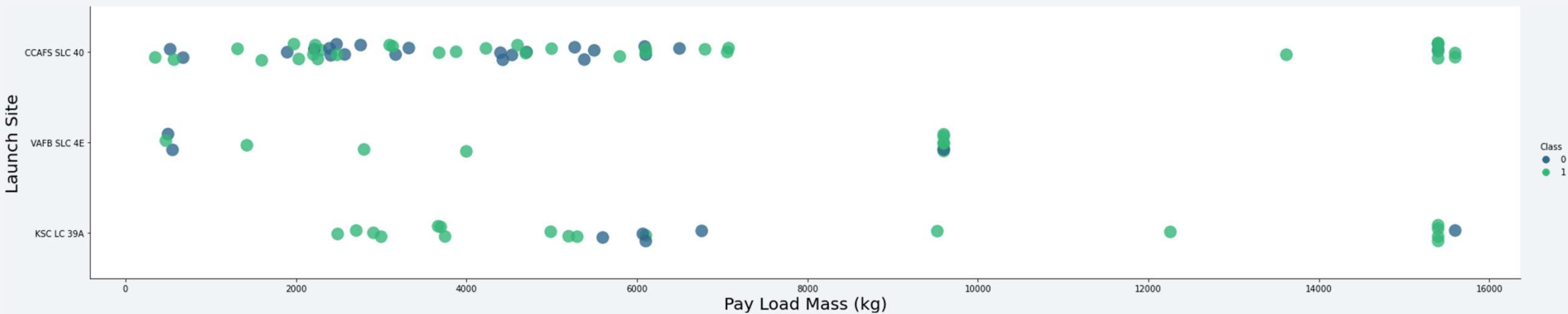
Insights drawn from EDA

Flight Number vs. Launch Site



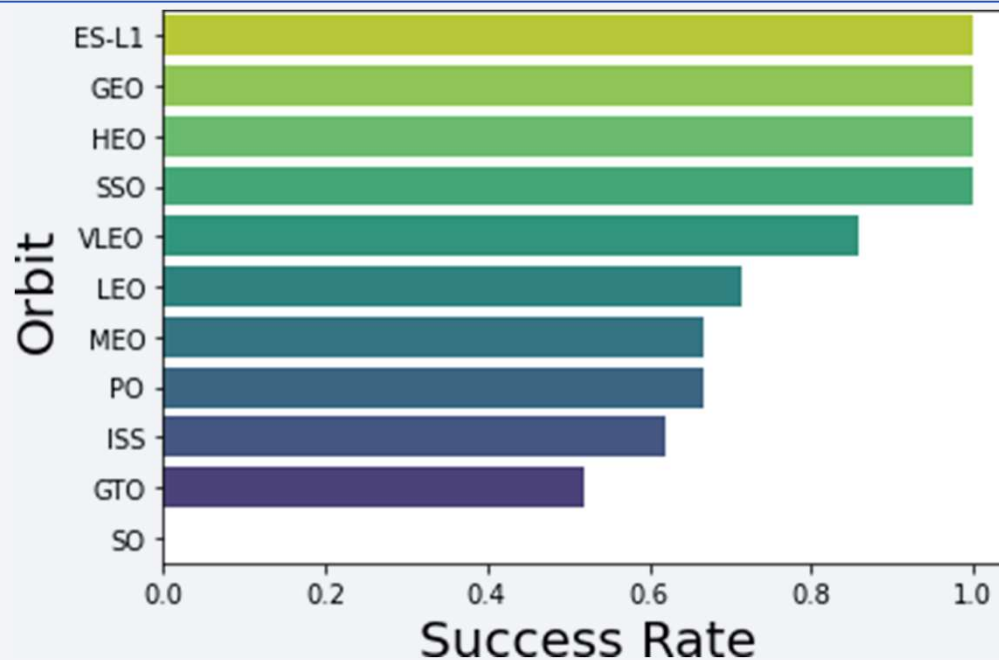
- Vemos aumento de éxito con el tiempo (Flight Number).
- Gran avance en torno al vuelo 20.
- CCAFS, sitio principal de lanzamiento, mayor volumen

Payload vs. Launch Site



- Carga útil oscila entre 0 y 6000 kg.
- Lugares de lanzamiento, utilizan diferentes masas de carga útil.

Success Rate vs. Orbit Type



ES-L1(1),GEO(1),HEO(1)

100%

SSO(5)

100%

VLEO(14)

>80%

SO(1)

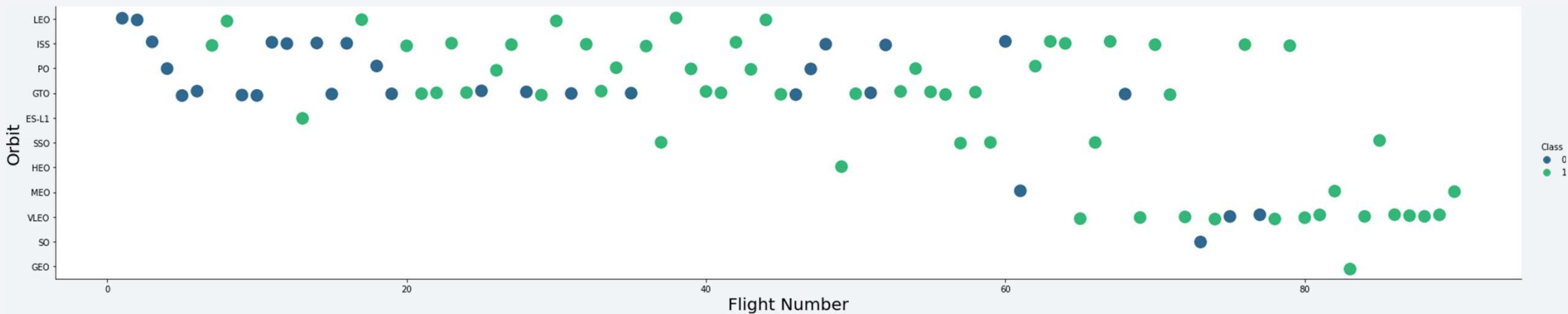
0%

GTO(27)

50% aprox.

(X) tamaño muestra

Flight Number vs. Orbit Type



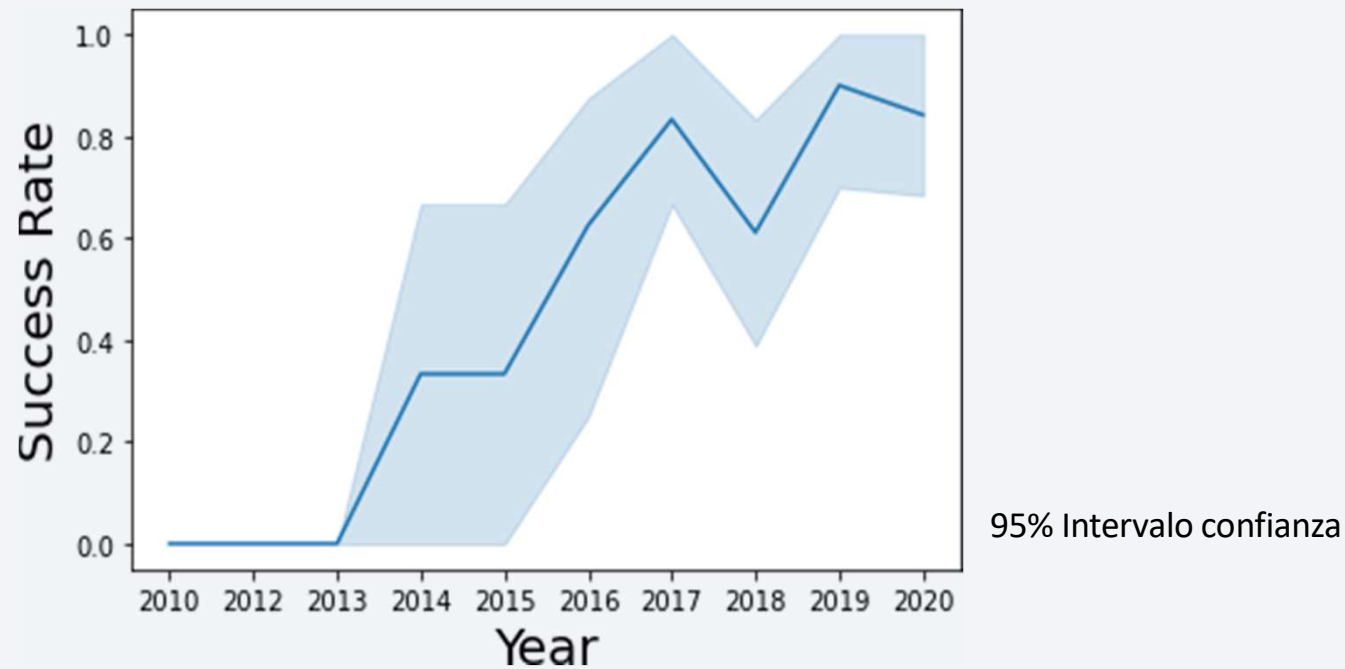
- Preferencia de órbita cambiaron según el número de vuelo. Relacionado con el resultado
- Comenzó con órbitas LEO, con éxito moderado. Volvió a VLEO más tarde.
- Mejor con órbitas más bajas o heliosincrónicas.

Payload vs. Orbit Type



- Masa de carga útil se relaciona con la órbita
- LEO y SSO, masa de carga útil baja
- VLEO, órbita con más éxito, masa carga útil extremo superior del rango

Launch Success Yearly Trend



- Éxito aumenta desde 2013, ligera caída en 2018
- Éxito 80%, 2019 y 2020

All Launch Site Names

```
[9]: %sql select distinct LAUNCH_SITE from SPACEXTABLE
* sqlite:///my_data1.db
Done.
[9]: Launch_Site
-----
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
```

- CCAFS SLC-40 y CCAFSSLC-40, probablemente mismo lugar
- Es possible que solo existan 3 lugares distintos, CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E

Launch Site Names Begin with 'CCA'

```
[11]: %sql select * from SPACEXTABLE where launch_site like 'CCA%' limit 5
```

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

```
[11]:
```

	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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
	2012-10-08	0:35:00	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

- Cinco primeros resultados donde campo LAUNCH_SITE empieza por CCA

Total Payload Mass

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

```
[12]: %sql select sum(PAYLOAD_MASS__KG_) as total from SPACE_TABLE where customer == 'NASA (CRS)'
* sqlite:///my_data1.db
Done.
[12]: total
-----
45596
```

Esta consulta calcula el total del campo PAYLOAD_MASS__KG_ para el cliente NASA (CRS)

Average Payload Mass by F9 v1.1

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

```
[13]: %sql select avg(PAYLOAD_MASS__KG_) as media from SPACEXTABLE where Booster_Version like 'F9 v1.1%'
* sqlite:///my_data1.db
Done.
```

	media
[13]:	2534.6666666666665

Esta consulta muestra la media del campo PAYLOAD_MASS__KG_ donde el campo Booster_Version es igual a F9 v1.1%

First Successful Ground Landing Date

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

Hint: Use min function

```
[14]: %sql select min(date) as MinDate from SPACEXTABLE where mission_outcome == 'Success'
```

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

```
Done.
```

```
[14]: MinDate
```

```
2010-06-04
```

La consulta devuelve el primer aterrizaje con éxito.

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
[15]: %sql select booster_version from SPACEXTABLE where mission_outcome == 'Success' and Landing_Outcome == 'Success (drone ship)' and (PAYLOAD_MASS_KG_ between 4000 and 6000)
* sqlite:///my_data1.db
Done.
```

```
[15]: Booster_Version
```

```
F9 FT B1022
```

```
F9 FT B1026
```

```
F9 FT B1021.2
```

```
F9 FT B1031.2
```

Esta consulta devuelve el valor del campo BOOSTER_VERSION para aterrizajes con éxito y carga entre 4000 y 6000 kg

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
[17]: %sql select mission_outcome, count(*) as Number from SPACEXTABLE group by mission_outcome order by mission_outcome
```

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

```
Done.
```

```
[17]:
```

Mission_Outcome	Number
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

La consulta devuelve recuento de cada mission. El 98% son éxito

Boosters Carried Maximum Payload

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

```
[20]: %sql select booster_version from SPACEXTABLE where payload_mass__kg_=(select max(payload_mass__kg_) from SPACEXTABLE)
* sqlite:///my_data1.db
Done.
```

```
[20]: 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

La consulta devuelve el campo BOOSTER_VERSION, todos son F9 B5 B10XX.XX
Esto indica que la carga útil se relaciona con BOOSTER_VERSION

2015 Launch Records

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
[23]: %sql select substr(Date, 6,2) as Mes, Landing_Outcome, booster_version, launch_site from SPACEXTABLE where substr(Date,0,5)='2015' and Landing_Outcome == 'Failure (drone ship)'
* sqlite:///my_data1.db
Done.
```

```
[23]:
```

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

Para los datos seleccionados en la consulta, vemos que devuelve dos registros, enero y abril

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

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

```
[24]: %sql select Landing_Outcome, count(*) as Total from SPACEXTABLE where date >='2010-06-04' and date <='2017-03-20' group by Landing_Outcome order by Total Desc
* sqlite:///my_data1.db
Done.
```

```
[24]:
```

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

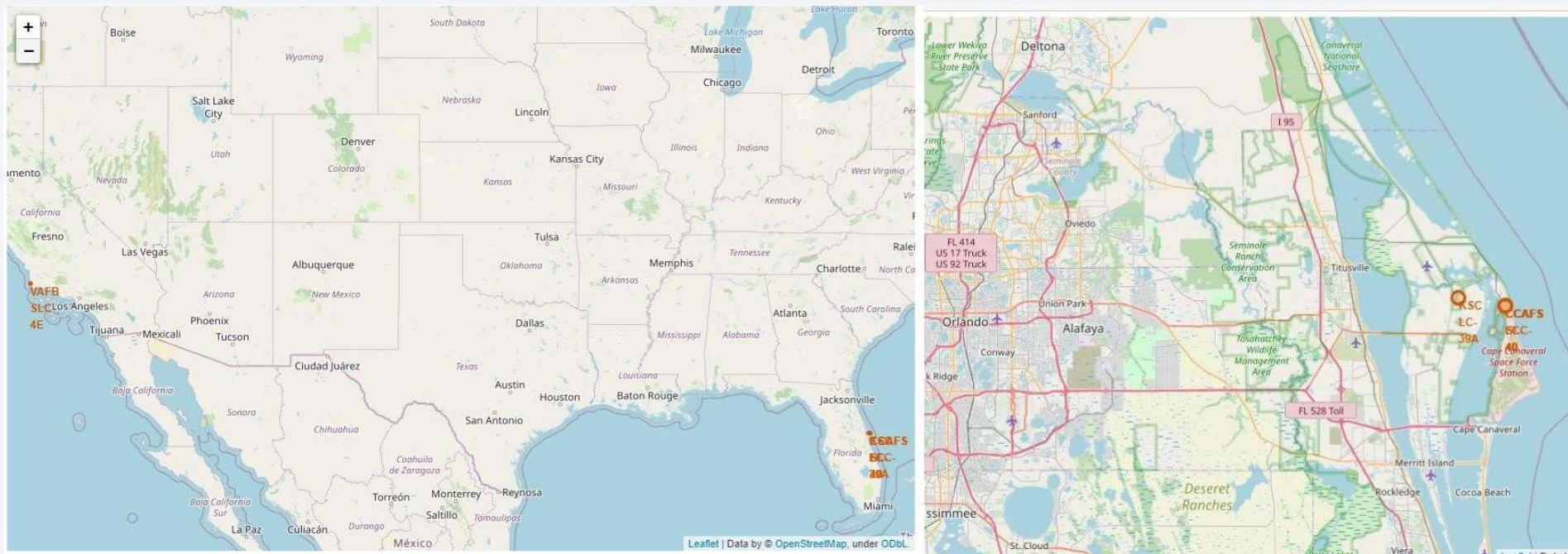
La consulta devuelve total del campo LANDING_OUTCOME entre las fecha seleccionadas.

A satellite view of Earth from space, showing the curvature of the planet and the glowing lights of cities at night. The background is a deep blue gradient.

Section 3

Launch Sites Proximities Analysis

Launch Site Locations

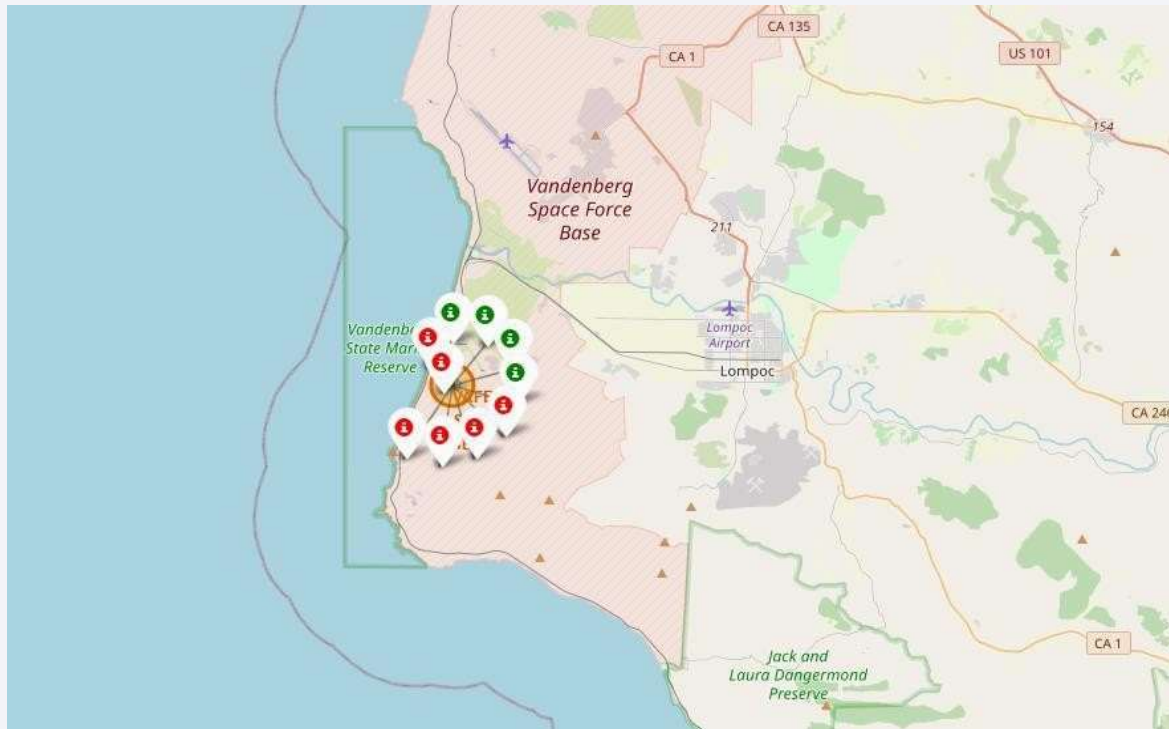


Mapa Izquierdo, todos los lugares de lanzamiento.

Mapa Derecho, dos lugares lanzamiento de Florida. Cerca uno de otro

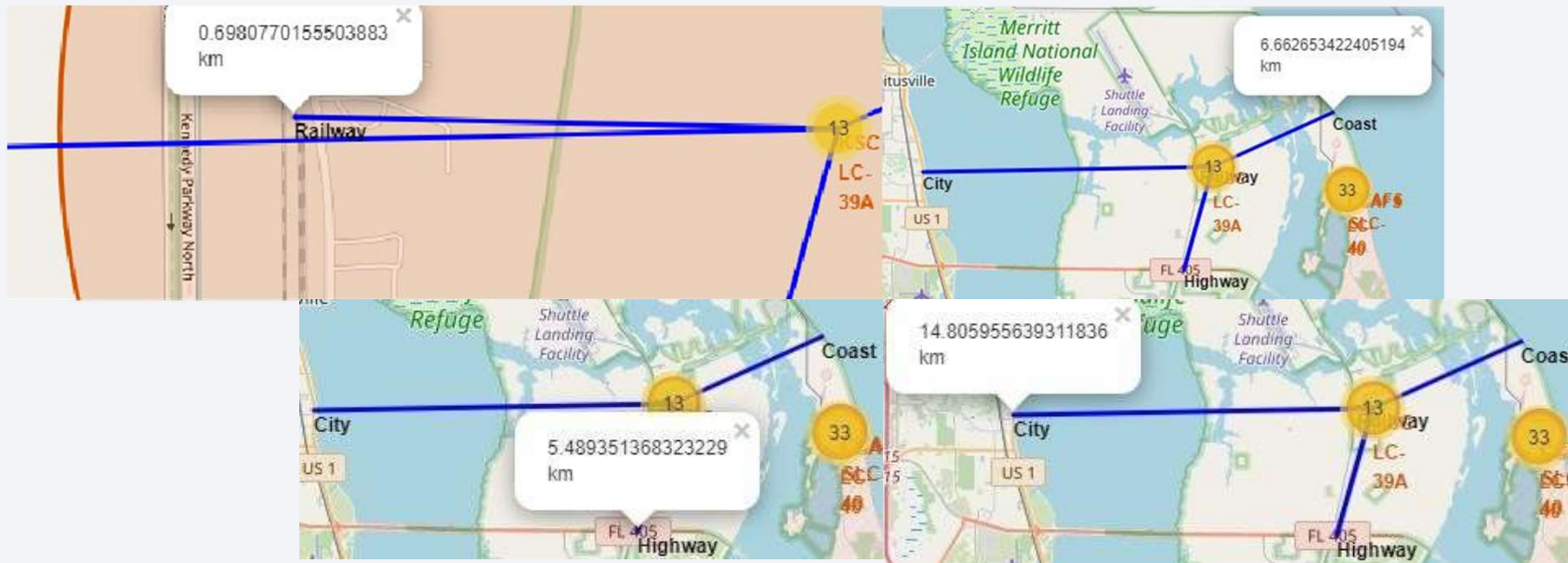
Todos están cerca del Oceano.

Color-Labeled Launch Markers



Podemos seleccionar los grupos del Mapa. Vemos aterrizaje con éxito (verde) o fallido (rojo). Para VAFB SLV-4E 4 con éxito y 6 fallidos.

Key Location Proximities



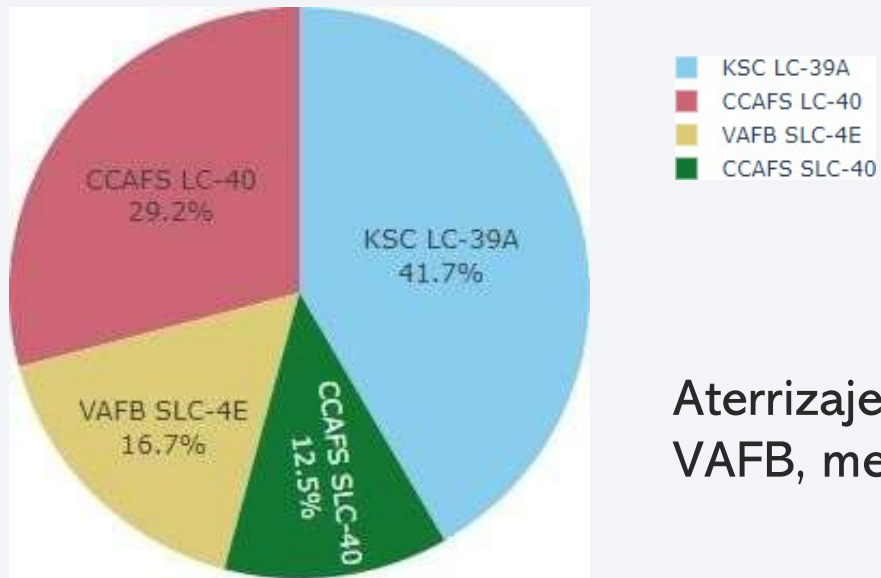
KSC LC-30-A como ejemplo. Están muy cerca del vías ferreas (suministros), carreteras (personas y suministros), cerca de costas y lejos de ciudades(fallidos al mar).



Section 4

Build a Dashboard with Plotly Dash

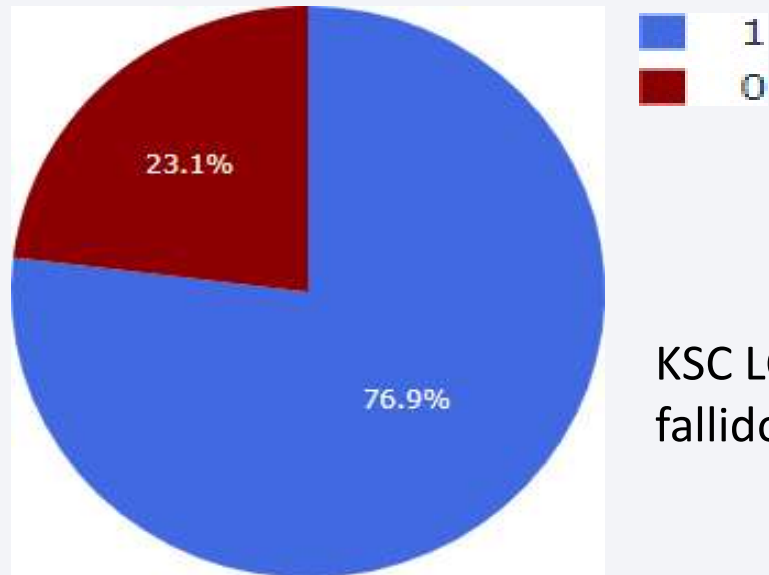
Successful Launches Across Launch Sites



Aterrizajes con éxito en la lugares de lanzamiento VAFB, menor porcentaje de éxito.

Highest Success Rate Launch Site

KSC LC-39A Success Rate (blue=success)



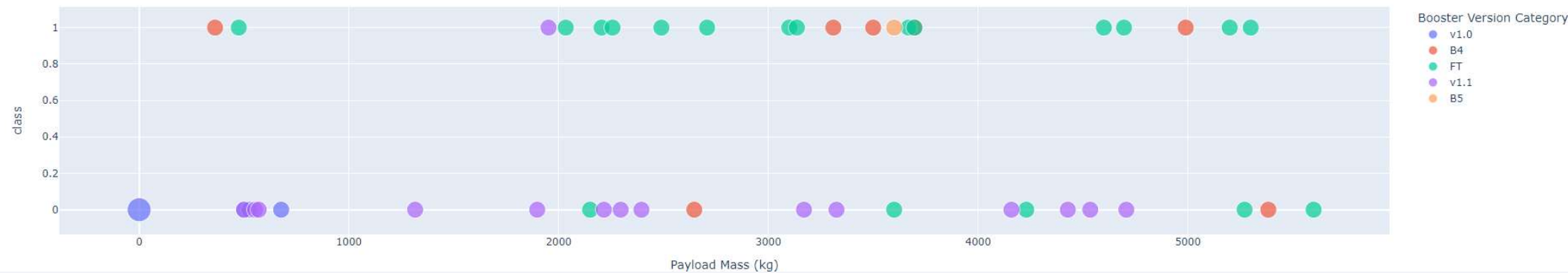
KSC LC-39A, tasa éxito más alta con 10 existosos y 3 fallidos

Payload Mass vs. Success vs. Booster V. Category

Payload range (Kg):



Payload Mass vs. Success vs. Booster Version Category



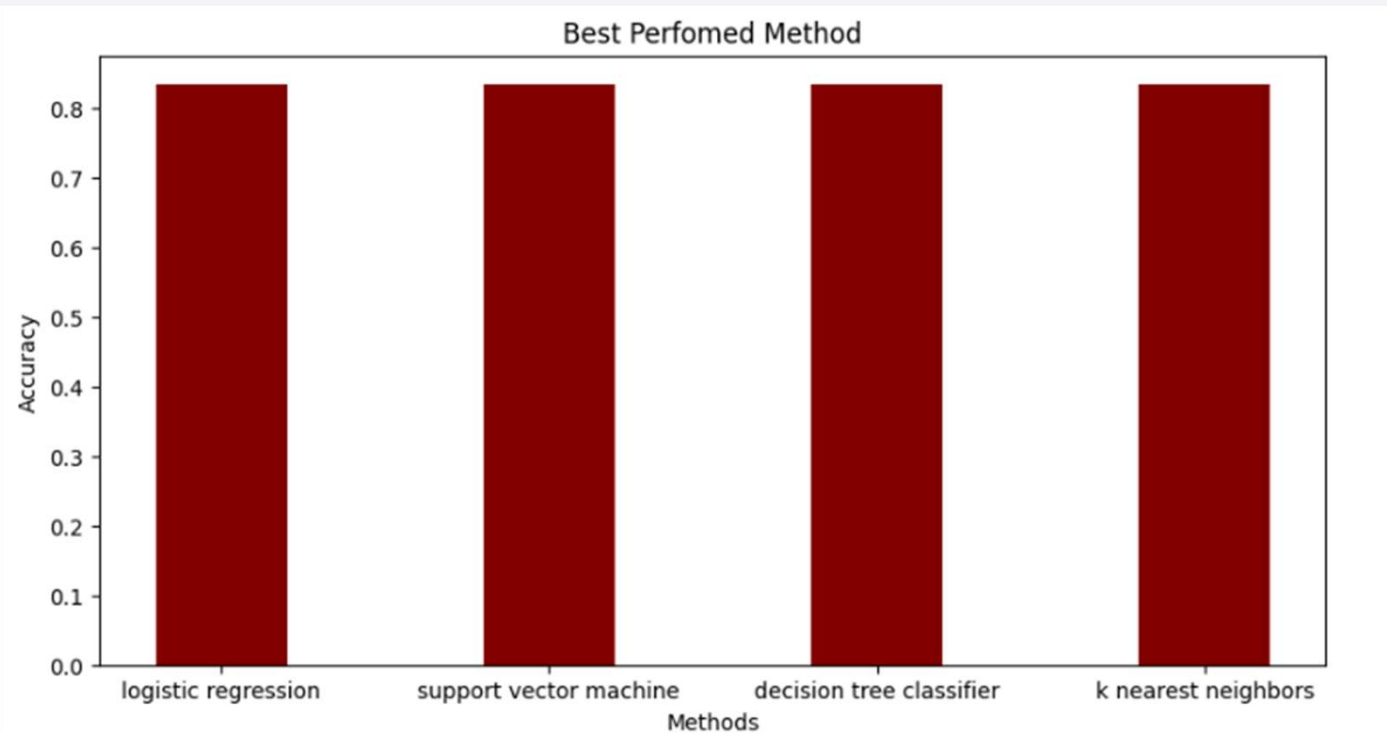
Class 1, indica éxito. Class 0, indica fracas. El diagrama tiene en cuenta la version del propulsor y la cantidad de lanzamientos según el tamaño del punto.



Section 5

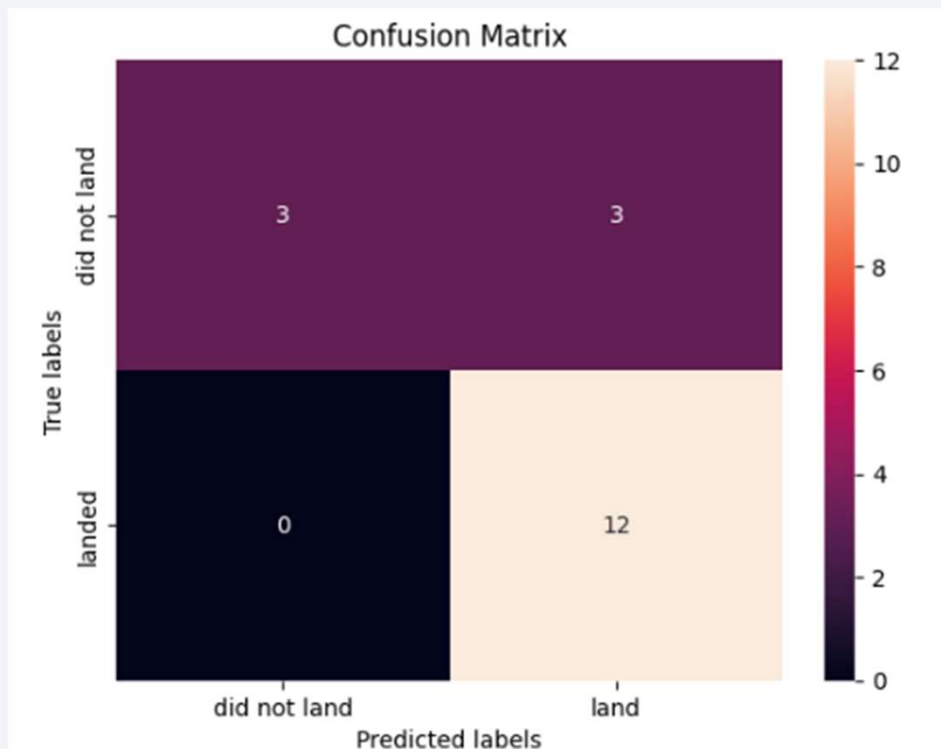
Predictive Analysis (Classification)

Classification Accuracy



- Todos los modelos tienen misma precision 83,33%. El tamaño de datos de la prueba es pequeño (18). Provoca variaciones en el modelo en ejecuciones repetidas. Necesitamos más datos para determinar un mejor modelo.

Confusion Matrix



La matrix es la misma para todos los modelos. Predijeron 12 aterrizajes con éxito cuando la etiqueta verdadera es aterrizaje éxito. Predijeron 3 aterrizajes con fallo cuando la etiqueta verdadera es aterrizaje fallido. Predijeron 3 aterrizajes con éxito cuando la verdadera es aterrizaje fallido(falso positivos) Estos modelos sobrepredicen aterrizajes con éxito

Conclusions

- La tarea ha sido desarrollar un modelo de aprendizaje para Space Y, competir con la empresa SpaceX
- El objeto es predecir con éxito el aterrizaje de la etapa 1 para ahorrar dinero
- Usamos los datos de la API de SpaceX y de la web SpaceX de Wikipedia
- Realizamos consulta SQL usando los datos anteriores y creamos etiquetas de datos
- Creamos un Dashboard para la visualización
- Creamos modelos de aprendizaje con una precisión del 83.33%
- Space Y puede utilizar el modelo para predecir si el aterrizaje de Etapa 1 tiene éxito o no
- Se necesitan más datos para determinar un mejor modelo y aumentar la precisión

Appendix

GitHub URL:

<https://github.com/Ftormogil/spaceY>

Gracias a Coursera e instructores:

<https://www.coursera.org/>

Rav Ahuja, Alex Aklson, Aije Egwaikhide, Svetlana Levitan, Romeo Kienzler, Polong Lin, Joseph Santarcangelo, Azim Hirjani, Hima Vasudevan, Saishruthi Swaminathan, Saeed Aghabozorgi, Yan Luo

Thank you!

