

Proyecto Ciencia de datos: Descubrimiento de insights de peleas de la UFC

Contenido

- Introducción 3
- Extracción de datos 4
 - Versiones del proyecto 5
- Visualización Efectiva..... 8
- Análisis Exploratorio de Datos (EDA)..... 9
 - Hallazgos con el EDA de la Visualización Efectiva de Datos..... 10
 - Hallazgos con el EDA generado con YDATA 14
 - Variables con alta correlación 16
- Modelado de Datos 17
- Hallazgos y conclusiones 18

Introducción

Este proyecto enfocado a las artes marciales mixtas (MMA) de la UFC, busca identificar los principales insights como lo son golpes en la cabeza, cuerpo y piernas con esto se pretende detectar posibles traumatismos. De igual forma se busca proponer un modelo de Machine Learning que identifica si un peleador va a ganar o no su próximo combate. La principal fuente de datos es el histórico de peleas de los luchadores el cual se encuentra de forma publica en la siguiente dirección:

Al hacer una exploración en los datos fuentes de históricos, no existe una forma directa de obtener la información, por lo cual se realizó una investigación sobre cómo obtener los datos.

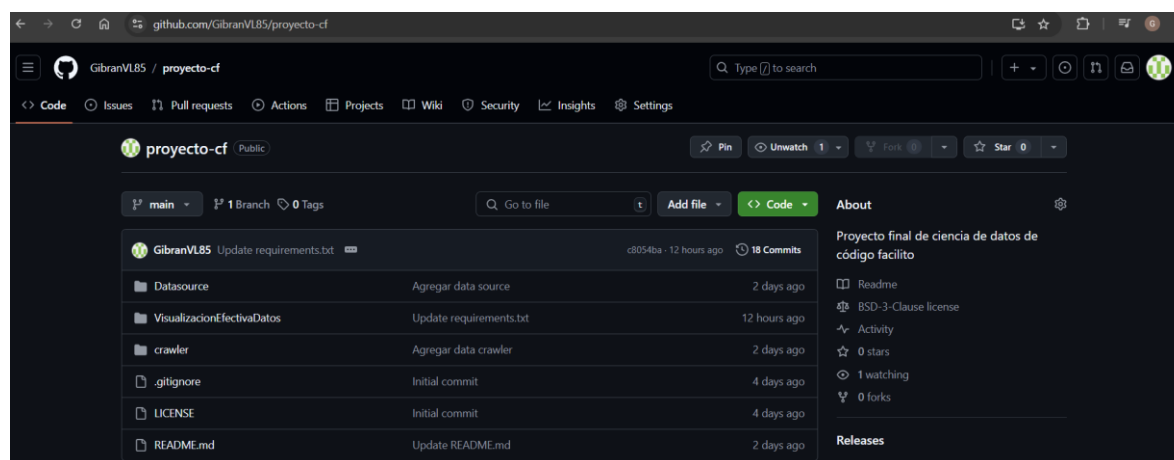
<http://www.ufcstats.com/>

El proyecto se integra en 3 partes:

- Visualización Efectiva del proyecto
- EDA
- Machine Learning

Cada sección cuenta con su propio archivo de Jupyter Notebooks, los cuales se pueden acceder a través del repositorio GIT del proyecto:

<https://github.com/GibranVL85/proyecto-cf>



Extracción de datos

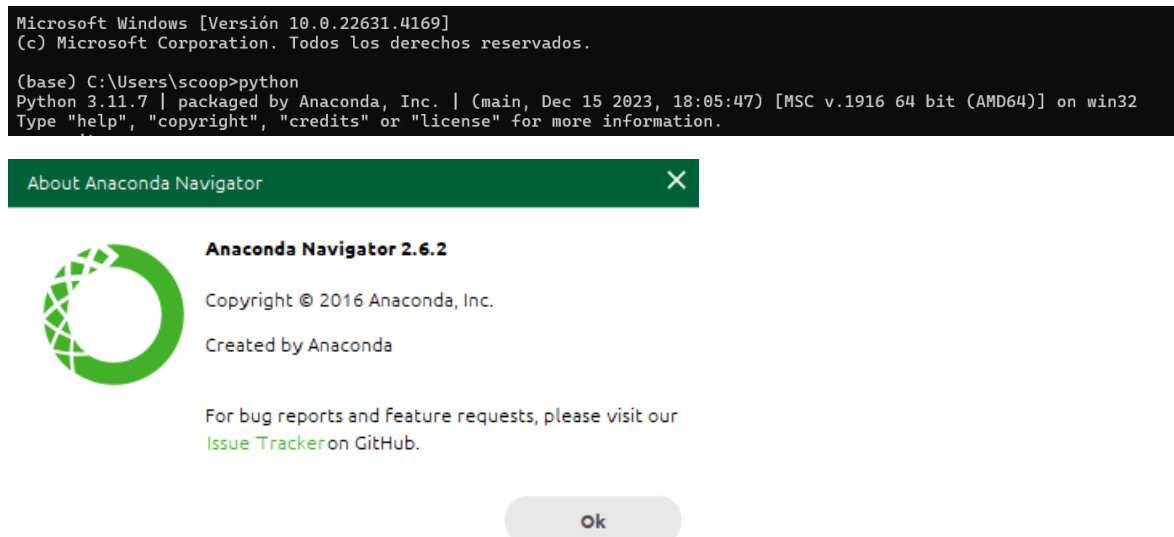
Tras una investigación en internet se encontró un scraper que utiliza la librería scrapy de Python.

<https://github.com/fanghuiz/ufc-stats-crawler>

La idea de utilizar el scraper es obtener la data que esta expuesta en el sitio web de la UFC y disponer de esta para realizar el análisis del proyecto.

<https://scrapy.org/>

El ambiente se configuro en Windows10 con Anaconda2.6.2 y Python 3.11.7



Para implementar el proyecto se instalo en un ambiente de conda independiente:

```
(base) C:\Users\scoop>conda env list
# conda environments:
#
base                *  C:\Users\scoop\anaconda3
crawler              C:\Users\scoop\anaconda3\envs\crawler
estadistica          C:\Users\scoop\anaconda3\envs\estadistica
pandas-env           C:\Users\scoop\anaconda3\envs\pandas-env
pruebas_spyder       C:\Users\scoop\anaconda3\envs\pruebas_spyder
```

Se actualizo la versión de scrapy y date parser, ya que la versión actual tiene conflictos con el proyecto.

Versiones del proyecto

Versiones implementadas

```

≡ requirements.txt
1  dateparser==0.7.2
2  scrapy==1.7.4

```

Versiones actualizadas

task-core	2023.11.0	py311haa95532_0	
datashader	0.16.0	py311haa95532_0	
dateparser	1.2.0	pypi_0	pypi
dav1d	1.2.1	h2bbff1b_0	
debugpy	1.6.7	py311hd77b12b_0	
decorator	5.1.1	pyhd3eb1b0_0	
scipy	1.11.4	py311hc1ccb85_0	
scrapy	2.8.0	py311haa95532_0	
seaborn	0.12.2	py311haa95532_0	
semver	2.13.0	pyhd3eb1b0_0	

Tras actualizar las librerías del proyecto también fue necesario actualizar el proyecto en los siguientes archivos para evitar errores en la ejecución:

spider.py

```

6
7  def time_to_seconds(time_str):
8      """Convierte una cadena de tiempo 'm:s' a segundos."""
9      minutes, seconds = map(int, time_str.split(':'))
10     return minutes * 60 + seconds
11

```

```

158     n_rev = stats_total[9].css('p ::text').getall()
159     try:
160         n_rev = [time_to_seconds(i.strip()) for i in n_rev] # type: ignore
161     except ValueError:
162         # Si ocurre una excepción, probablemente el formato es incorrecto, así que lo dejamos como está
163         n_rev = [int(i.strip()) for i in n_rev if i.strip().isdigit()]
164
165

```

```

236 class FightersSpider(scrapy.Spider):
237     name = 'ufcFighters'
238     start_urls = ['http://ufcstats.com/statistics/fighters']
239
240     custom_settings = {
241         'format': 'csv',
242         'URI': 'data/fighter_stats/(time)s.csv'
243     }

```

```

326 class UpcomingFightsSpider(scrapy.Spider):
327     name = 'upcoming'
328     start_urls = ['http://ufcstats.com/statistics/events/completed']
329     time_created = print_time('now')
330
331     custom_settings = {
332         'format': 'csv',
333         'URI': f'data/upcoming/{time_created}.csv'
334     }

```

items.py

```

7
8 import scrapy
9 from scrapy.loader.processors import Identity, MapCompose, Join
10 from itemloaders.processors import TakeFirst, Compose
11
12
13
14
15
16 name = scrapy.Field(output_processor=TakeFirst())
17 height = scrapy.Field(output_processor=TakeFirst())
18 weight = scrapy.Field(output_processor=())
19 reach = scrapy.Field(output_processor=TakeFirst())
20 stance = scrapy.Field(output_processor=TakeFirst())
21 dob = scrapy.Field(output_processor=TakeFirst())

```

Tras las correcciones, se procedió a ejecutar el proyecto:

En la ruta del proyecto

```

(base) C:\Users\scoop\anaconda3\envs\crawler>dir
El volumen de la unidad C es OS
El número de serie del volumen es: F270-2D7C

Directorio de C:\Users\scoop\anaconda3\envs\crawler

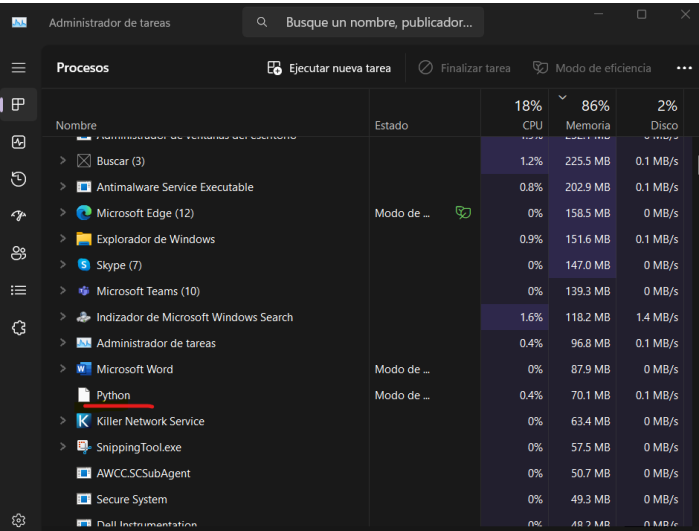
31/07/2024  11:59 a. m.      <DIR>          .
20/09/2024  06:37 p. m.      <DIR>          ..
20/10/2020  12:01 a. m.                1,872 .gitignore
23/07/2024  07:28 p. m.                0 .nonadmin
23/07/2024  07:28 p. m.      <DIR>          conda-meta
01/08/2024  07:30 p. m.      <DIR>          data
20/10/2020  12:01 a. m.                100 Dockerfile
23/07/2024  07:28 p. m.      <DIR>          etc
20/10/2020  12:01 a. m.            1,068 LICENSE
01/08/2024  07:30 p. m.      <DIR>          log
20/10/2020  12:01 a. m.                767 Makefile
31/07/2024  11:59 a. m.                140 packages_list.txt
20/10/2020  12:01 a. m.            3,092 README.md
20/10/2020  12:01 a. m.                31 requirements.txt
20/10/2020  12:01 a. m.                259 scrapy.cfg
31/07/2024  11:42 a. m.      <DIR>          ufcStats
          9 archivos              7,329 bytes
          7 dirs  235,113,820,160 bytes libres

```

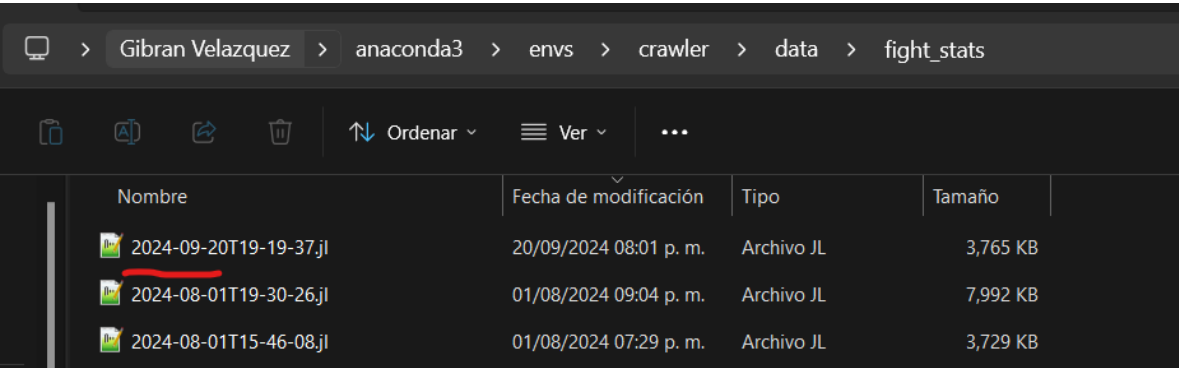
Ejecutar comando scrapy que manda llamar al crawler ufcFights

```
(base) C:\Users\scoop\anaconda3\envs\crawler>scrapy crawl ufcFights
```

Esto empezara a ejecutar un proceso Python



El cual se encarga de extraer la data de la web de UFCStats. Y generara un fichero CSV con la información de las peleas en el directorio con el nombre del crawler.



Visualización Efectiva

Para la visualización se consideraron los siguientes puntos:

- Claridad y precisión: Las graficas implementadas fueron seleccionadas con el objetivo de entenderse de manera fácil acorde al dataset analizado.
- Diversidad de visualizaciones: Se seleccionaron diferentes gráficas con diferentes sets para transmitir los hallazgos del dataset.
- Interpretación y presentación: Dentro del fichero de visualización efectiva

Para validar el desarrollo de la visualización efectiva es necesario revisar el archivo visualización-efectiva-datos.ipynb del repositorio de git:

<https://github.com/GibranVL85/proyecto-cf/tree/main/VisualizacionEfectivaDatos>

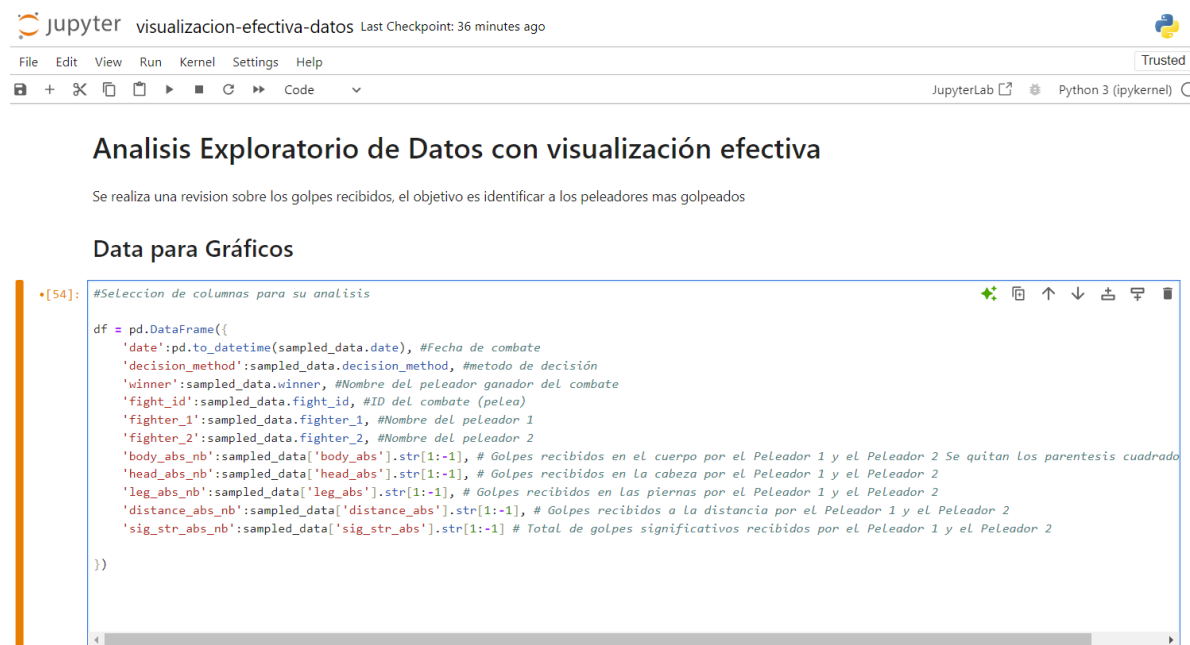


Análisis Exploratorio de Datos (EDA)

Para el EDA se realizaron 2 partes:

La primera parte del EDA se realizó como complemento a la visualización efectiva de datos, este complemento, en esta sección se realizó la preparación de datos, limpieza e identificación de insights. El acceso para esta sección es en el notebook de visualización efectiva:

<https://github.com/GibranVL85/proyecto-cf/tree/main/VisualizacionEfectivaDatos>



```
[54]: #Selección de columnas para su análisis

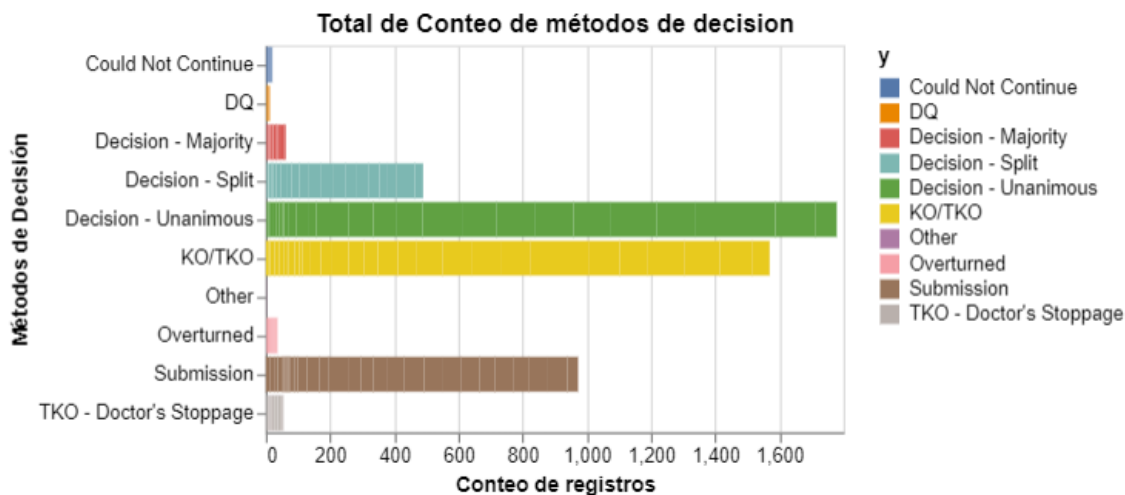
df = pd.DataFrame({
    'date':pd.to_datetime(sampled_data.date), #Fecha de combate
    'decision_method':sampled_data.decision_method, #metodo de decisión
    'winner':sampled_data.winner, #Nombre del peleador ganador del combate
    'fight_id':sampled_data.fight_id, #ID del combate (pelea)
    'fighter_1':sampled_data.fighter_1, #Nombre del peleador 1
    'fighter_2':sampled_data.fighter_2, #Nombre del peleador 2
    'body_abs_nb':sampled_data['body_abs'].str[1:-1], # Golpes recibidos en el cuerpo por el Peleador 1 y el Peleador 2 Se quitan los parentesis cuadrado
    'head_abs_nb':sampled_data['head_abs'].str[1:-1], # Golpes recibidos en la cabeza por el Peleador 1 y el Peleador 2
    'leg_abs_nb':sampled_data['leg_abs'].str[1:-1], # Golpes recibidos en las piernas por el Peleador 1 y el Peleador 2
    'distance_abs_nb':sampled_data['distance_abs'].str[1:-1], # Golpes recibidos a la distancia por el Peleador 1 y el Peleador 2
    'sig_str_abs_nb':sampled_data['sig_str_abs'].str[1:-1] # Total de golpes significativos recibidos por el Peleador 1 y el Peleador 2
})
```

La segunda parte, busca realizar un análisis más exhaustivo de los datos implementando un dataframe más depurado.

<https://github.com/GibranVL85/proyecto-cf/tree/main/EDA>

Hallazgos con el EDA de la Visualización Efectiva de Datos

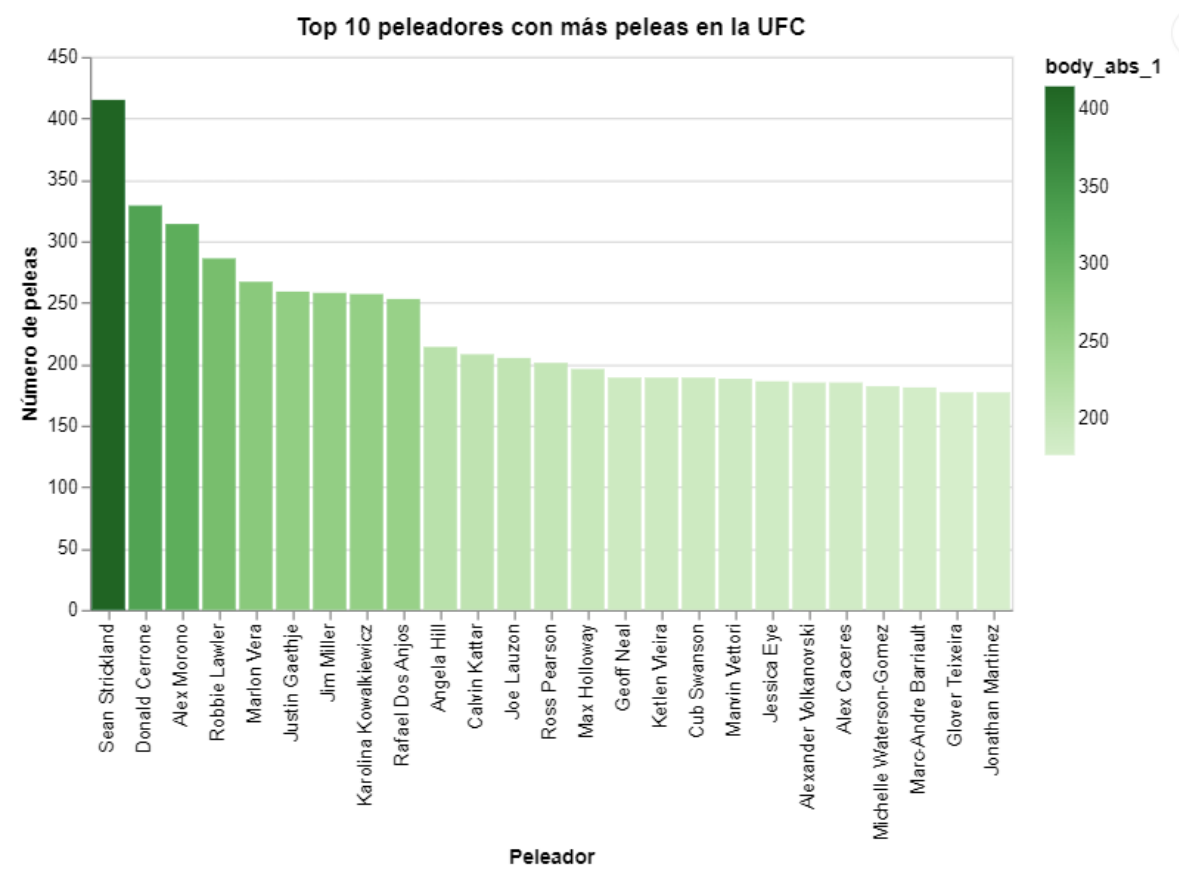
El método de decisión, es la forma en como termina una pelea, en la gráfica se puede identificar que la mayoría termina siendo una decisión unánime, seguida por KO/TKO y al final por submission.



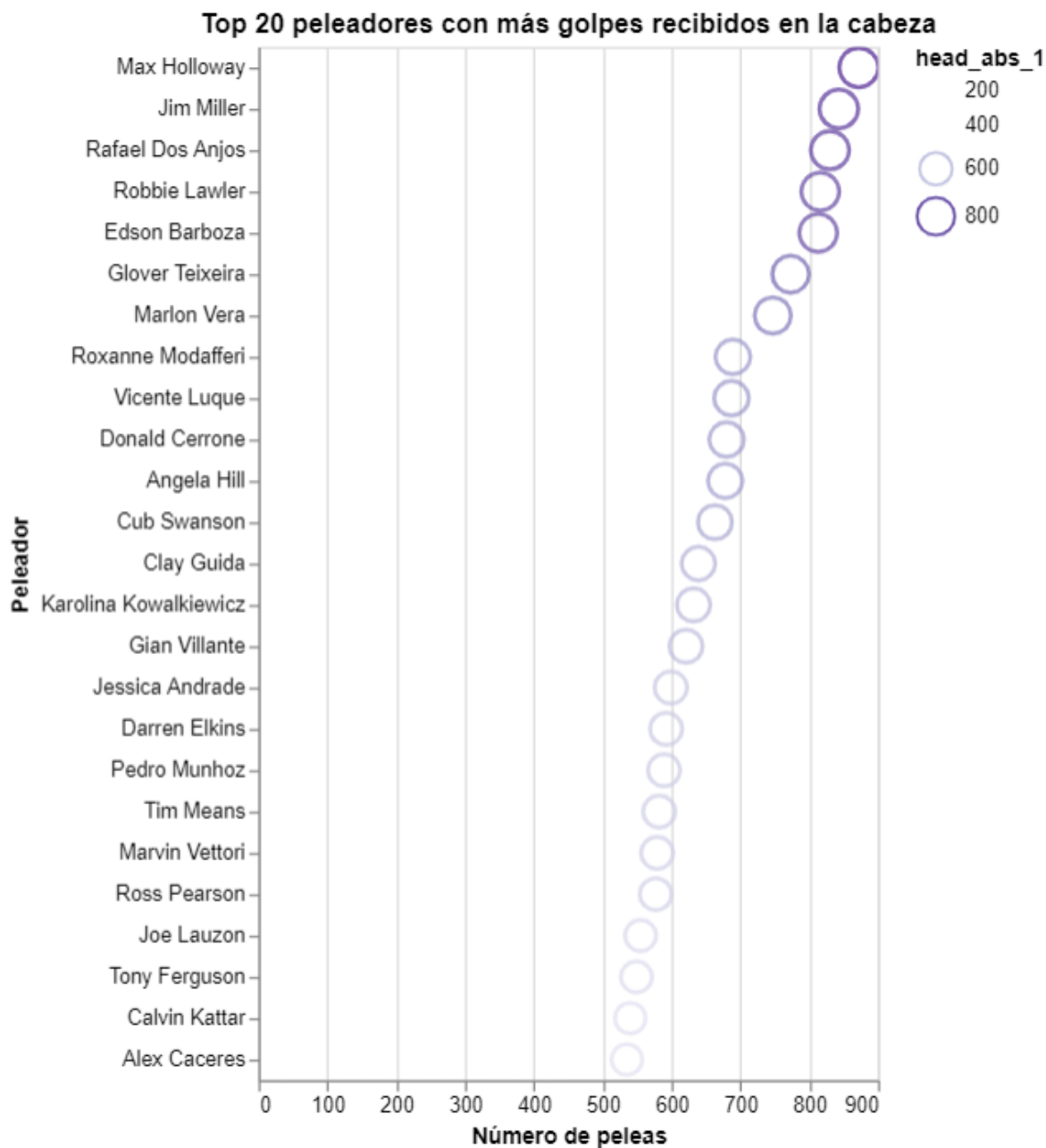
Se han identificado a los peleadores con mas peleas en cuyo caso sobresalen Jim Miller, Donald Cerrone y Rafael Dos Anjos.



Se identificaron a los peleadores con más golpes recibidos en el cuerpo identificando a Sean Strickland como la persona con mayor numero de golpes recibidos.

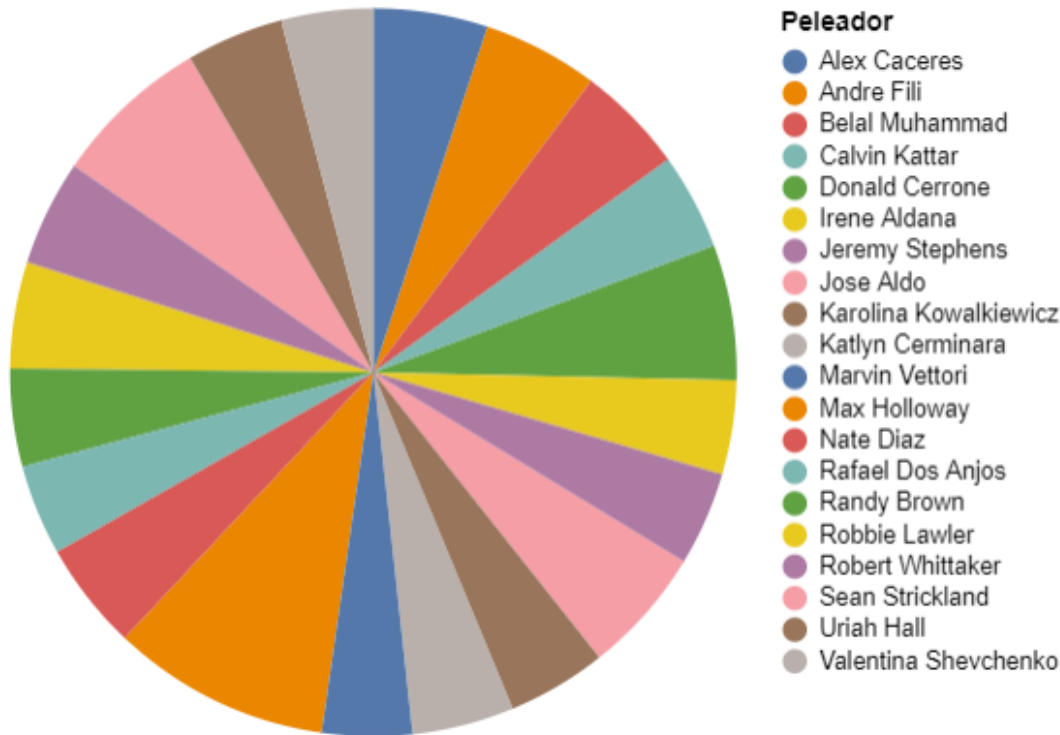


Se identificó al peleador Max Holloway como el peleador que mas golpes ha recibido en las peleas de la UFC.



Se identificaron a los peleadores con más golpes recibidos en las piernas entre los más golpeados están Andre Fili y Jose Aldo.

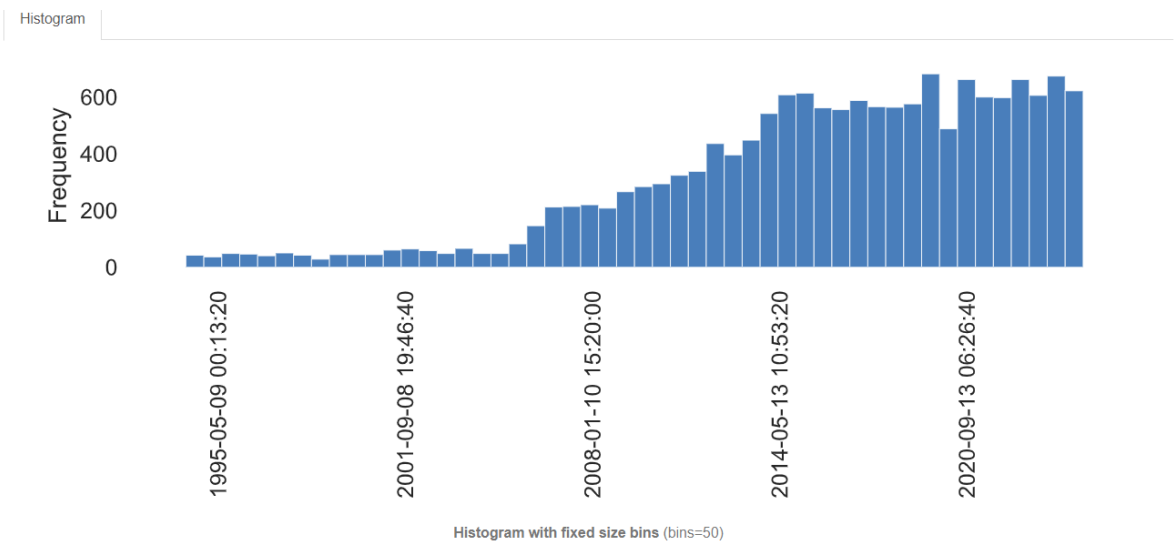
Top 20 peleadores con más golpes recibidos en las piernas



Hallazgos con el EDA generado con YDATA

De igual forma se detectaron los siguientes hallazgos:

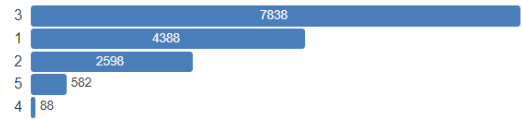
En la variable Date se detecto un incremento de eventos desde el origen de las peleas en 1995 hasta 2024.



Acorde a la variable figth_duration_lastrnd, se detecta que la mayoría de las peleas duran 3 rounds, seguido de las peleas que duran 1 round esto debido a la submission y/o finalización por KO/TKO.

figth_duration_lastrnd
Categorical

Distinct	5
Distinct (%)	< 0.1%
Missing	0
Missing (%)	0.0%
Memory size	121.2 KiB



Overview Words Characters

Overview Words Characters

A word cloud of boxing terms. The word 'bout' is the largest and most central. Other prominent words include 'heavyweight', 'lightweight', 'welterweight', 'flyweight', 'bantamweight', 'middleweight', 'featherweight', 'strawweight', 'light', 'weight', 'catch', 'title', 'interim', 'women's', 'ufc', 'tournament', 'fighter', 'ultimate', and 'open'.

Variables con alta correlación

Acorde al análisis se detectaron las variables con alta correlación

Alerts

body_abs_1	is highly overall correlated with	body_att_1	and 8 other fields
body_att_1	is highly overall correlated with	body_abs_1	and 11 other fields
body_def_1	is highly overall correlated with	body_abs_1	and 7 other fields
body_land_1	is highly overall correlated with	body_abs_1	and 10 other fields
clinch_abs_1	is highly overall correlated with	body_abs_1	and 4 other fields
clinch_att_1	is highly overall correlated with	body_att_1	and 6 other fields
clinch_def_1	is highly overall correlated with	clinch_abs_1	and 2 other fields
clinch_land_1	is highly overall correlated with	body_att_1	and 5 other fields
ground_abs_1	is highly overall correlated with	ground_def_1	
ground_att_1	is highly overall correlated with	ground_land_1	
ground_def_1	is highly overall correlated with	ground_abs_1	
ground_land_1	is highly overall correlated with	ground_att_1	
head_abs_1	is highly overall correlated with	body_abs_1	and 7 other fields
head_att_1	is highly overall correlated with	body_abs_1	and 10 other fields
head_def_1	is highly overall correlated with	body_abs_1	and 9 other fields
head_land_1	is highly overall correlated with	body_abs_1	and 8 other fields
leg_abs_1	is highly overall correlated with	body_abs_1	and 5 other fields
leg_att_1	is highly overall correlated with	body_att_1	and 4 other fields
leg_def_1	is highly overall correlated with	leg_abs_1	
leg_land_1	is highly overall correlated with	body_att_1	and 4 other fields
td_att_1	is highly overall correlated with	td_land_1	
td_land_1	is highly overall correlated with	td_att_1	

Modelado de Datos

Por cuestiones de tiempo no fue posible completar el modelado.

Hallazgos y conclusiones

Tras realizar la revisión de los datos se puede concluir que un peleador profesional de la UFC recibe muchos golpes a lo largo de su trayectoria como peleador lo cual indica que son sujetos susceptibles de sufrir cualquier tipo de contusiones, daños y/o traumatismos físicos.

Derivado del análisis se puede recomendar a los peleadores considerar análisis periódicos para detectar alguna anomalía física y evitar secuelas que a lo largo del tiempo puedan complicarse.

De igual forma se presentan los insights identificados:

- La mayoría de las peleas terminan por método de decisión unánime, lo cual indica que se realiza una pelea completa. Esto implica mayor desgaste físico en los peleadores. Se recomienda a los peleadores considerar campamentos donde se trabaje el cardio, esto con la finalidad de cubrir el tiempo completo de la pelea sin denotar mayor cansancio.
- El segundo método de decisión es el KO/TKO lo cual señala que la empresa debe estar preparada y tener un equipo médico adecuado para este tipo de finalizaciones, por el lado de los peleadores se les recomienda preparación física para evitar este tipo de finalizaciones.
- La tercera forma de finalización de una pelea es la de submission, esto apunta a recomendar a los peleadores a implementar en su campamento, preparación en técnicas de lucha, jiu-jitsu y zambo para evitar sumisiones.
- LA parte más golpeada en el cuerpo de un peleador es la cabeza, esto indica que los peleadores deben realizar revisiones médicas periódicas o al terminar su pelea, al recibir un mayor daño en la cabeza el peleador se convierte en propenso a sufrir alguna contusión o daño cerebral tras recibir muchos golpes a lo largo de su carrera como peleador.
- Se identificó al peleador Max Holloway como el peleador que más golpes ha recibido en las peleas de la UFC, se recomienda al peleador, una revisión general médica para detectar posibles daños colaterales.
- Se identificaron a los peleadores con más golpes recibidos en las piernas, por lo que se les recomienda realizar análisis médicos para detectar posibles fisuras, hematomas o posibles daños causados por la cantidad de golpes recibidos.
- Se identificaron a los peleadores con más golpes recibidos en el cuerpo a los cuales se les recomienda realizar análisis médicos enfocados en las zonas del hígado, riñones y las costillas siendo estas las zonas más comúnmente golpeadas.