12-6-2024

APLICACIÓN DEL BIG DATA A UNA EMPRESA DE ENTRENAMIENTO PERSONAL

FUNZIONA

ALEJANDRO FERNÁNDEZ SÁNCHEZ

CURSO 2023/ 2024

ÍNDICE

1. INTRODUCCIÓN

En el presente trabajo, analizamos las diferentes partes que componen la empresa, así como sus objetivos y motivaciones. Cómo hemos aplicado el big data a esta empresa, que resultados hemos obtenido y como interpretar dichos datos.

Presentación empresa

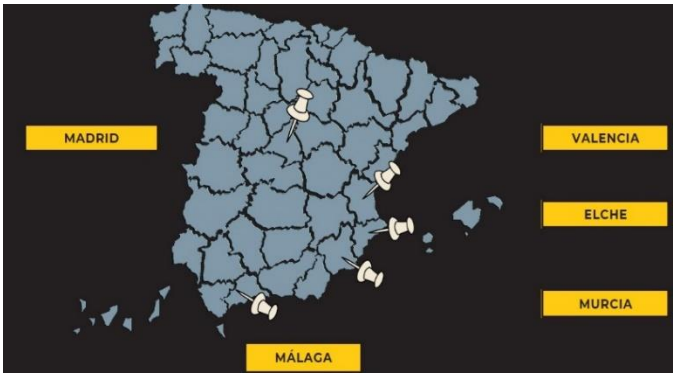
Funziona, es una empresa dedicada a la realización de entrenamientos personales dentro de gimnasios Basic Fit (estilo Low Cost).

Tiene como objetivo general mejorar calidad de vida de las personas a través de la actividad física, con una misión clara: hacer del entrenamiento personal una via para conseguir dicha calidad.

Como visión en el futuro, quieren llegar a ser la empresa de entrenamiento personal mas grande de España.

Actualmente cuenta con un total de 42 clubes repartidos por toda España entre Madrid, Valencia, Málaga, Alicante, Cartagena, Murcia y Elche. Sumando a todo esto, un número de 85 entrenadores, organizados entre Junior-Senior-Formación y Manager / Regionales.

El día a día, consiste en crear y fidelizar una cartera de clientes dentro de los gimnasios. Y a nivel económico se sostiene por las carteras de los entrenadores y teniendo como gasto principal el alquiler que abona a Basic Fit los propios RR.HH y la gestoría adecuada.



Objetivo del proyecto

El presente trabajo tiene la finalidad principal de conseguir información necesaria para el día a día de la empresa Funziona. Se busca:

- Mejorar la toma de decisiones.
- Conocer a nuestros entrenadores y clientes.
- Conseguir información relevante para el día a día de nuestros trabajadores.
- Encontrar nuevas fuentes de información / datos que nos permitan mejorar el desempeño y el servicio que prestamos.

Motivación

El día a día de los entrenadores, se divide entre acciones comerciales, entrenamiento y tareas administrativas. La recopilación de datos se hace a través de una aplicación.

No hay un feed-back de esos datos recopilados a los entrenadores. Lo que creemos que facilitaría la oferta y servicios de los entrenadores, de la búsqueda de contactos nuevos y de la capacidad de fidelización.

Cierto es, que hay una serie de datos que no se consiguen o queda a libertad de entrenador el hecho de etiquetarlos.

Todo este desempeño, lleva a una reflexión sobre la "calidad" de los datos que tenemos y sobre las acciones que deberíamos construir a partir de ellos. De ahí surge la idea de crear este trabajo, de cara a mejorar procesos y conocer mejor la empresa y sector en el que nos movemos.

Características de los datos.

Nos movemos a través de dos documentos csv, uno facilitado por la propia app que tiene la empresa y otro de elaboración propia a partir de datos encontrados en internet.

Uno de los documentos es: "Clubes.csv" (TABLA1), donde se recopilan datos directamente de los clubes y del entorno. Este documento es de creación propia, con datos que he ido recopilando de internet y otros datos facilitados por la empresa colaboradora.

Las columnas que tienen son:

Nombre club	Nombre del centro
CIUDAD	Donde está ubicado el club en España
BARRIO	Ubicación Dentro de la ciudad
DISTRITO	Ubicación Dentro del barrio
CP	El código postal del club
Nº SOCIOS	Socios del club
Nº CLIENTES	Clientes que tiene la empresa dentro del gimnasio
POBLACIÓN	Personas que viven en el barrio/distrito del club
MEDIA ALTAS	Cuantos clientes nuevos hace el club de media anual
% DE CLIENTES	La relación entre los clientes de la empresa y los clientes del club.
RENTA MEDIA PERSONA	Renta de alrededores del club.
Nº ENTRENADORES	Cantidad de entrenadores que hay en el club
DENSIDAD	Hab / km cuadrado del club.
FACTURACION	€ totales que factura el club en Mayo 2024.

TICKET MEDIO	La media de lo que pagan los clientes de la empresa en cada gimnasio
AÑO CLUB	En qué año cogió el club la empresa.
CANON CLUB	Cuanto paga la empresa por estar en el club.
CANON/ENTRENADOR	La relación del canon correspondiente a cada entrenador
HORAS ENTRENADORES MES	Cantidad de horas de entrenadores en el club
COSTE LABORAL/ HORA	Coste de entrenadores por hora.
COSTE CLUB	Suma de columnas anteriores.
PRUEBAS MES	Cantidad de pruebas realizadas al mes por entrenadores.
ALTAS	Media de altas de cada club.
LATITUD/LONGITUD	Ubicación del club.

TABLA 1 : " CLUBES.CSV "

El otro documento es "PRUEBAS.CSV" (TABLA 2) facilitado por el gestor de la empresa que recopila datos de los entrenadores , clientes, clubes y las sesiones.

ENTRENADOR	**Nombre de entrenador**
CLIENTE	Nombre del cliente.
TIPO_CLIENTE	Si es alta, potencial, o baja.
CP_CLUB	Código postal del club
SEXO	Hombre, Mujer o en blanco
FECHA DE NACIMIENTO	Edad del cliente.
CP	Código postal del cliente
CLUB	Club de pertenencia del cliente.
COSTE_SESION	Precio que paga el cliente por una clase

CANTIDAD_SESIONES_MES	Número de sesiones al mes que tiene el cliente.
DURACION_SESIONES	Tiempo que dura la sesión del cliente
COSTE_MENSUAL	Cuanto paga al mes
MEDIO_PAGO	Como paga el cliente.

TABLA2 : “PRUEBAS.CSV”

De este documento he sacado un Excel: “cambios definitivos.xls” , que será el que utilizaremos para introducirlo en Power BI.

2. METODOLOGÍA

Transformación y procesamiento de los datos

En este apartado, marcamos los pasos correspondientes que hemos dado de cara a preparar los datos para una visualización eficiente.

CLUBES.CSV

Empezamos con el documento “clubes.csv” , el cual, a rasgos generales es una recopilación de datos demográficos, económicos, y de resultados de los clubes que componen la empresa.

En una primera visualización del data set en con la función shape, encontramos que hay 54 filas y 27 columnas de las cuales, hay varias columnas con el total de sus valores “nulos” , y dos columnas “unnamed” sin dato ninguno, así que nuestro primer paso será borrar esas filas y líneas que no nos sirven, con la función “drop”.

```
[6] df.columns

Index(['Nombre club', 'CIUDAD', 'BARRIO', 'DISTRITO', 'CP', 'NºSOCIOS',
       'NºCLIENTES', 'POBLACION', 'MEDIA ALTAS', '% DE CLIENTES',
       'RENTA MEDIA PERSONA', 'Nºentrenadores', 'DENSIDAD ', 'Facturacion',
       'Ticket Medio', 'AÑO CLUB', 'CANON CLUB', 'Canon/entrenador',
       'Horas entrenadores MES', 'Coste laboral/hora', 'Coste club',
       'Pruebas mes', 'Altas', 'LATITUD', 'LONGITUD', 'Unnamed: 25',
       'Unnamed: 26'],
      dtype='object')

[7] #Borramos las columnas vacias y no necesarias.

    df=df.drop(columns=['Unnamed: 25',
           'Unnamed: 26', 'BARRIO', 'DISTRITO', 'CP','Coste club'])

[8] df.head()
```

	Nombre club	CIUDAD	NºSOCIOS	NºCLIENTES	POBLACION	MEDIA ALTAS	% DE CLIENTES	RENTA MEDIA PERSONA	Nºentrenadores	DENSIDAD	...	Ticket Medio	AÑO CLUB	CANON CLUB	Canon/entrenador	Horas entrenadores MES	Coste laboral/hora	Pruebas mes	Altas	LATITUD	LONGITUD
0	BFGM	ALCOBENDAS	1359,00	7,00	46789,00	99,00	0,52%	12443,00	1,00	2,656	...	304,29	2018,0	400,00	400,00	80,00	7,44	3,30	0,70	40,54819794	-3,639029331
1	BFSL	ALCOBENDAS	1757,00	8,00	46789,00	92,00	0,46%	10803,00	2,00	2,656	...	381,25	2024,0	900,00	450,00	140,00	7,44	9,20	1,30	40,53814071	-3,634884204
2	BFLI (ALC)	ALICANTE	2250,00	8,00	1467,00	249,00	0,36%	21009,00	2,00	1,742	...	210,78	2023,0	400,00	200,00	140,00	7,44	6,20	0,20	38,37534391	-0,509916235
3	BFPA (ALC)	ALICANTE	2028,00	19,00	8978,00	225,00	0,94%	14578,00	2,00	1,742	...	86,79	2023,0	400,00	200,00	140,00	7,44	4,70	0,80	38,34646876	-0,502560273
4	BFLF (CT)	CARTAGENA	1493,00	2,00	45048,00	133,00	0,13%	11660,00	1,00	391	...	105,00	2023,0	400,00	400,00	80,00	7,44	3,30	0,30	37,60883177	-0,983894889

5 rows × 27 columns

```
[9] df.tail()
```

	Nombre club	CIUDAD	NºSOCIOS	NºCLIENTES	POBLACION	MEDIA ALTAS	% DE CLIENTES	RENTA MEDIA PERSONA	Nºentrenadores	DENSIDAD	...	Ticket Medio	AÑO CLUB	CANON CLUB	Canon/entrenador	Horas entrenadores MES	Coste laboral/hora	Pruebas mes	Altas	LATITUD	LONGITUD
49	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
50	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
51	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
52	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
53	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 27 columns

```
[10] # Eliminamos filas que no sirven
     df=df.drop(df.index[40:])
```

A continuación, Cambiamos nombres de columnas a un modelo más accesible, quitando espacios para facilitar su comprensión y su uso en futuras visualizaciones. Para ello utilizamos la función “rename”

```
# CAMBIAR NOMBRE COLUMNAS
df.rename(columns={'MEDIA ALTAS': 'M/ALTAS',
                   '% DE CLIENTES': '%CLIENTES',
                   'RENTA MEDIA PERSONA': 'RM/PERSONA',
                   'Ticket Medio': 'Ticket',
                   'AÑO CLUB':'Fundacion',
                   'CANON CLUB':'CANON',
                   'Horas entrenadores MES':'Horas/entrenadores/MES',
                   'Coste laboral/hora':'Coste/laboral/hora',
                   'Pruebas mes':'Pruebas',
                   'Nombre club':'Club',
                   'NºSOCIOS':'Socios',
                   'NºCLIENTES':'Clientes',
                   'Nºentrenadores': 'Entrenadores',
                   'DENSIDAD ':'DENSIDAD'},
          inplace = True)
```

Se puede apreciar que, al usar “info” las columnas están en formato object y no float que es el que nos interesa:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40 entries, 0 to 39
Data columns (total 21 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Club                    40 non-null     object 
 1   CIUDAD                  40 non-null     object 
 2   Socios                  40 non-null     object 
 3   Clientes                40 non-null     object 
 4   POBLACION               40 non-null     object 
 5   M/ALTAS                 40 non-null     object 
 6   %CLIENTES               40 non-null     object 
 7   RM/PERSONA              40 non-null     object 
 8   Entrenadores            40 non-null     object 
 9   DENSIDAD                40 non-null     object 
 10  Facturacion             40 non-null     object 
 11  Ticket                  40 non-null     object 
 12  Fundacion               40 non-null     float64
 13  CANON                   40 non-null     object 
 14  Canon/entrenador        40 non-null     object 
 15  Horas/entrenadores/MES  40 non-null     object 
 16  Coste/laboral/hora      40 non-null     object 
 17  Pruebas                 40 non-null     object 
 18  Altas                   40 non-null     object 
 19  LATITUD                 40 non-null     object 
 20  LONGITUD                40 non-null     object 
dtypes: float64(1), object(20)
memory usage: 6.7+ KB
```

También podemos apreciar en los datos, que, al venir de Excel, tienen un formato de “,” como separador decimal y no dé “.” Así que aprovecho y creo una función para poder hacer esos cambios todos a la vez.

```
18] # Selecciono las columnas a cambiar formato
    cols= [ 'Socios', 'Clientes', 'POBLACION', 'M/ALTAS','DENSIDAD',
           'RM/PERSONA', 'Entrenadores', 'Facturacion', 'Ticket',
           'Fundacion', 'CANON', 'Canon/entrenador', 'Horas/entrenadores/MES',
           'Coste/laboral/hora', 'Pruebas', 'Altas', 'LATITUD', 'LONGITUD']
```

```
19] def convert_to_float(value):
        try:
            # Reemplazar comas por puntos y convertir a float
            return float(value.replace(',', '.'))
        except AttributeError:
            return value  # Manejar casos donde el valor ya es de tipo float
```

```
    for col in cols:
        df[col] = df[col].apply(convert_to_float)
```

```
21] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40 entries, 0 to 39
Data columns (total 21 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Club                     40 non-null     object
 1   CIUDAD                   40 non-null     object
 2   Socios                   40 non-null     float64
 3   Clientes                 40 non-null     float64
 4   POBLACION                40 non-null     float64
 5   M/ALTAS                  40 non-null     float64
 6   %CLIENTES                40 non-null     object
 7   RM/PERSONA               40 non-null     float64
 8   Entrenadores             40 non-null     float64
 9   DENSIDAD                 40 non-null     float64
 10  Facturacion              40 non-null     float64
 11  Ticket                   40 non-null     float64
 12  Fundacion                40 non-null     float64
 13  CANON                    40 non-null     float64
 14  Canon/entrenador         40 non-null     float64
 15  Horas/entrenadores/MES   40 non-null     float64
 16  Coste/laboral/hora       40 non-null     float64
 17  Pruebas                  40 non-null     float64
 18  Altas                    40 non-null     float64
 19  LATITUD                  40 non-null     float64
 20  LONGITUD                 40 non-null     float64
dtypes: float64(18), object(3)
memory usage: 6.7+ KB
```

Vemos que faltaría una por convertir, y el “problema” es que tiene un símbolo de % en el dato lo cual no lo identifica como float. Hacemos otra función y así lo cambiamos y todas las columnas quedan en el tipo de que queremos.

PRUEBAS.CSV

En el segundo documento, el contenido es principalmente de los clientes que tiene la empresa y algunos datos de los mismos aportados por la empresa y su app.

En cuanto a la transformación de los mismos, el proceso es parecido al documento anterior. Empezamos viendo la composición de la data frame

```
[6] df.columns

Index(['ENTRENADOR', 'CLIENTE', 'TIPO_CLIENTE', 'CP CLUB', 'CLUB', 'SEXO',
       'FECHA_NACIM', 'CP', 'COSTE_SESION', 'CANTIDAD_SESIONES_MES',
       'DURACION_SESIONES', 'COSTE_MENSUAL', 'MEDIO_PAGO'],
      dtype='object')
```

```
[7] df.head()
```

	ENTRENADOR	CLIENTE	TIPO_CLIENTE	CP CLUB	CLUB	SEXO	FECHA_NACIM	CP	COSTE_SESION	CANTIDAD_SESIONES_MES	DURACION_SESIONES	COSTE_MENSUAL	MEDIO_PAGO
0	Camilo Restrepo	Alexandra Cruz	Entrenado	46120.0	BF Alboraya	F	04/04/1999	46131	24,95	8.0	45.0	199.600	Domiciliación
1	Camilo Restrepo	Ana Beatriz Saud	Potencial	46120.0	BF Alboraya	F	19/12/1986	46011	25	8.0	45.0	199.000	TPV Virtual
2	Camilo Restrepo	Antonio Plaza	Entrenado	46120.0	BF Alboraya	M	18/03/1994	46011	36	5.0	45.0	180.000	Domiciliación
3	Camilo Restrepo	Enrique Pavón Martí	Potencial	46120.0	BF Alboraya	M	15/04/1964	46011	35	4.0	45.0	140.000	Domiciliación
4	Camilo Restrepo	Giosevid Acosta	Potencial	46120.0	BF Alboraya	M	24/08/1986	NaN	NaN	0.0	NaN	NaN	Efectivo

Pasos siguientes: Generar código con df Ver gráficos recomendados

```
df.tail()
```

	ENTRENADOR	CLIENTE	TIPO_CLIENTE	CP CLUB	CLUB	SEXO	FECHA_NACIM	CP	COSTE_SESION	CANTIDAD_SESIONES_MES	DURACION_SESIONES	COSTE_MENSUAL	MEDIO_PAGO
5980	Pablo Veldhuizen	Rodolfo Rodríguez Jiménez	Entrenado	28033.0	BF Virgen del Carmen	NaN	NaN	NaN	NaN	0.0	NaN	NaN	Tarjeta
5981	Pablo Veldhuizen	Sergio Marrupe	Entrenado	28033.0	BF Virgen del Carmen	NaN	NaN	NaN	NaN	0.0	NaN	NaN	TPV Virtual
5982	Pablo Veldhuizen	Test - Agustin Fila	Entrenado	28033.0	BF Virgen del Carmen	M	NaN	NaN	NaN	NaN	NaN	NaN	Domiciliación
5983	Pablo Veldhuizen	Test - Sebas GymGy	Entrenado	28033.0	BF Virgen del Carmen	F	NaN	NaN	NaN	NaN	NaN	NaN	Domiciliación
5984	Pablo Veldhuizen	WeeLoon Tee	Entrenado	28033.0	BF Virgen del Carmen	NaN	NaN	NaN	NaN	0.0	NaN	NaN	TPV Virtual

```
[9] df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5985 entries, 0 to 5984
Data columns (total 13 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   ENTRENADOR            5985 non-null   object
 1   CLIENTE               5985 non-null   object
 2   TIPO_CLIENTE          5984 non-null   object
 3   CP CLUB               5983 non-null   float64
 4   CLUB                  5985 non-null   object
 5   SEXO                  5397 non-null   object
 6   FECHA_NACIM           3797 non-null   object
 7   CP                    4136 non-null   object
 8   COSTE_SESION          5822 non-null   object
 9   CANTIDAD_SESIONES_MES 5981 non-null   float64
 10  DURACION_SESIONES     5223 non-null   float64
 11  COSTE_MENSUAL         5830 non-null   object
 12  MEDIO_PAGO            5985 non-null   object
dtypes: float64(3), object(10)
memory usage: 608.0+ KB
```

```
[10] df.shape

(5985, 13)
```

En este documento, también tenemos el problema de la columna con datos tipo object y float, al igual que las “, “y los “. “ .Así que procedemos a usar la función del documento anterior para cambiarlo de nuevo.

Como queremos hacer una correlación entre las columnas tipo “float”, procedo a probar, pero al hacerlo nos aparecen algunos datos que no están en la forma correcta. Así que, elimino directamente esas filas para poder realizar la correlación fácilmente.

```
valor_a_eliminar = 'Alexandra Cruz'
```

```
df = df.loc[df['CLIENTE'] != valor_a_eliminar ]
print(df)
```

```
          ENTRENADOR                    CLIENTE TIPO_CLIENTE  CP_CLUB  \
1     Camilo Restrepo          Ana Beatriz Saud    Potencial  46120.0
2     Camilo Restrepo            Antonio Plaza    Entrenado  46120.0
3     Camilo Restrepo     Enrrique Pavon Marti    Potencial  46120.0
4     Camilo Restrepo         Clonesvid Acosta    Potencial  46120.0
5     Camilo Restrepo             Sandra Soto    Potencial  46120.0
...               ...                      ...          ...      ...
5980  Pablo Veldhuizen  Rodolfo Rodriguez Jimenez    Entrenado  28033.0
5981  Pablo Veldhuizen            Sergio Marrupe    Entrenado  28033.0
5982  Pablo Veldhuizen      Test. - Agustin Fila    Entrenado  28033.0
5983  Pablo Veldhuizen       Test. - Sahas GymGy    Entrenado  28033.0
5984  Pablo Veldhuizen             Weelcom Tee    Entrenado  28033.0

                    CLUB SEXO FECHA_NACIM     CP COSTE_SESION  \
1              BF Alboraya    F  19/12/1966  46011           25
2              BF Alboraya    M  16/01/1994  46011           30
3              BF Alboraya    M  15/04/1964  46011           35
4              BF Alboraya    M  24/08/1986    NaN          NaN
5              BF Alboraya    F  08/11/1979  46011           25
...                    ...  ...         ...    ...          ...
5980  BF Virgen del Carmen  NaN         NaN    NaN          NaN
5981  BF Virgen del Carmen  NaN         NaN    NaN          NaN
5982  BF Virgen del Carmen    M         NaN    NaN          NaN
5983  BF Virgen del Carmen    F         NaN    NaN          NaN
5984  BF Virgen del Carmen  NaN         NaN    NaN          NaN

      CANTIDAD_SESIONES_MES  DURACION_SESIONES  COSTE_MENSUAL       MEDIO_PAGO
1                       8.0               45.0       199.000      TPV Virtual
2                       5.0               45.0       180.000  Domiciliacion
3                       4.0               45.0       140.000  Domiciliacion
4                       0.0                NaN           NaN         Efectivo
5                       8.0               45.0       199.000  Domiciliacion
...                     ...                ...           ...              ...
5980                    0.0                NaN           NaN          Tarjeta
5981                    0.0                NaN           NaN      TPV Virtual
5982                    NaN                NaN           NaN  Domiciliacion
5983                    NaN                NaN           NaN  Domiciliacion
5984                    0.0                NaN           NaN      TPV Virtual

[5984 rows x 13 columns]
```

```
df = df[df['COSTE_MENSUAL'] != '2.317.100']
```

```
df = df[df['COSTE_MENSUAL'] != '1.170.000']
```

```
df = df[df['COSTE_MENSUAL'] != '1.138.000']
```

```
df['COSTE_SESION'] = df['COSTE_SESION'].astype(float)
df['COSTE_MENSUAL'] = df['COSTE_MENSUAL'].astype(float)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 5981 entries, 1 to 5984
Data columns (total 13 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   ENTRENADOR             5981 non-null   object
 1   CLIENTE                5981 non-null   object
 2   TIPO_CLIENTE           5980 non-null   object
 3   CP_CLUB                5899 non-null   float64
 4   CLUB                   5981 non-null   object
 5   SEXO                   5393 non-null   object
 6   FECHA_NACIM            3756 non-null   object
 7   CP                     4134 non-null   object
 8   COSTE_SESION           5818 non-null   float64
 9   CANTIDAD_SESIONES_MES  5977 non-null   float64
 10  DURACION_SESIONES      5319 non-null   float64
 11  COSTE_MENSUAL          5826 non-null   float64
 12  MEDIO_PAGO             5981 non-null   object
dtypes: float64(5), object(8)
memory usage: 654.3+ KB
```

Por último, creo una columna nueva con los nombres de los regionales para poder hacer evaluaciones por regiones en la empresa de cara a futuro. Pero esta información no la usaré para el dashboard de power bi.

3. ANÁLISIS DE DATOS

CLUBES

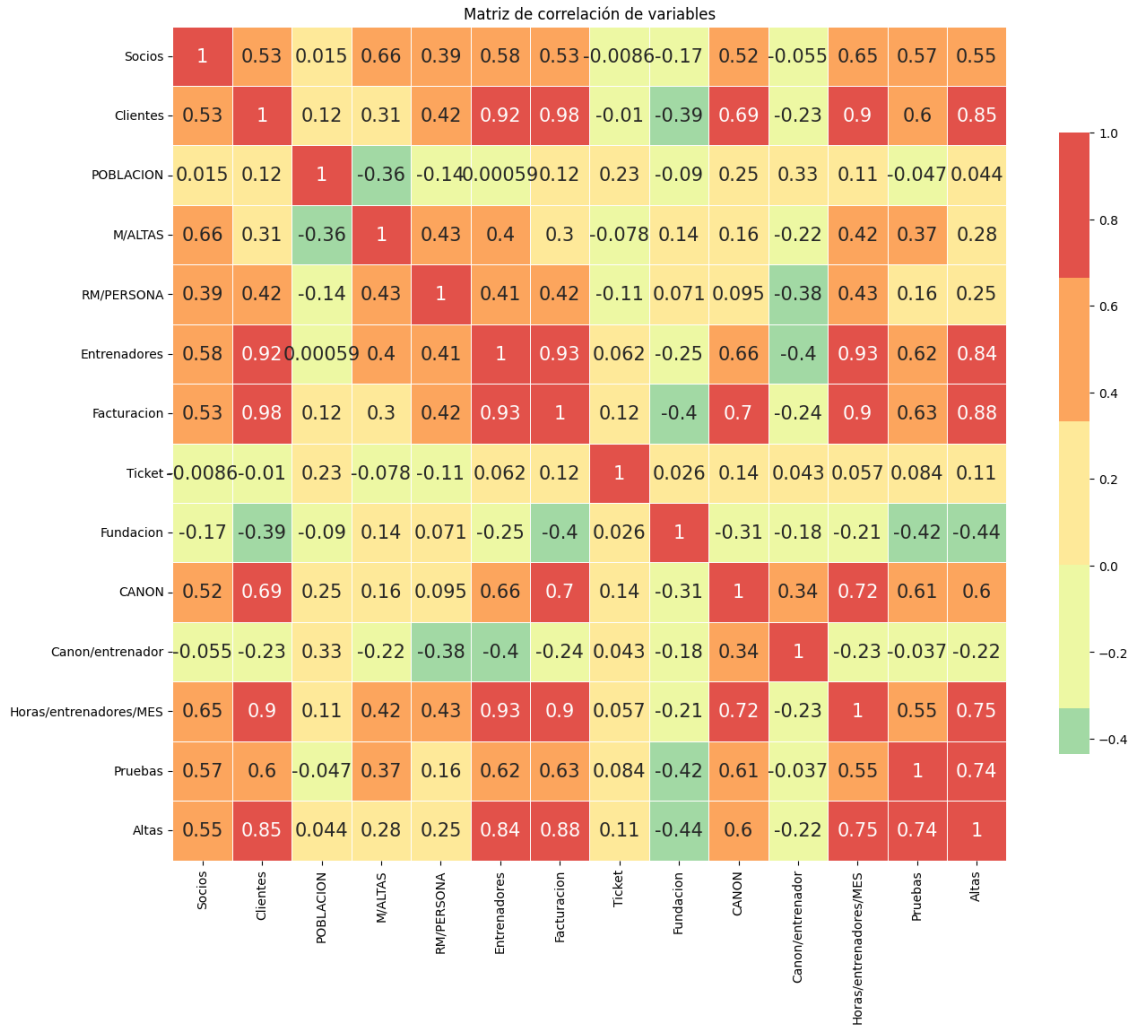
Aplicamos la función “describe” para ver algunos valores estadísticos:

```
df.describe()
```

	Socios	Clientes	POBLACION	M/ALTAS	%CLIENTES	RM/PERSONA	Entrenadores	DENSIDAD	Facturacion	Ticket	Fundacion	CANON	Canon/entrenador	Horas/entrenadores/MES	Coste/laboral/hora	Pruebas	Altas	LATITUD	LONGITUD
count	40.000000	40.000000	40.000000	40.000000	40.000000	40.000000	40.00000	40.000000	40.000000	40.00000	40.000000	40.000000	40.000000	40.000000	40.00	40.000000	40.000000	40.000000	40.000000
mean	2138.100000	20.175000	27826.150000	157.075000	0.987000	18636.181750	2.00000	8466.934750	3518.839000	175.05050	2022.050000	580.000000	338.535750	166.375000	7.44	5.777500	1.282500	39.274651	-2.006165
std	849.966077	20.825142	20771.489498	66.586487	1.208011	4178.162408	1.26085	12802.711776	3828.284227	62.67825	1.767043	294.566174	188.990985	192.317138	0.00	3.097185	1.382082	1.228831	1.669894
min	860.000000	1.000000	1467.000000	59.000000	0.050000	8940.560000	1.00000	1.397000	0.000000	0.00000	2018.000000	400.000000	171.430000	20.000000	7.44	1.500000	0.000000	38.680490	-4.460700
25%	1619.500000	7.000000	9383.000000	119.500000	0.412500	13086.750000	1.00000	5.772000	994.312500	143.80500	2021.750000	400.000000	200.000000	45.000000	7.44	3.800000	0.450000	38.330814	-3.873083
50%	1803.500000	15.000000	23888.500000	147.000000	0.740000	17165.805000	2.00000	5.772000	2783.350000	172.89000	2023.000000	400.000000	350.000000	80.000000	7.44	5.500000	1.000000	39.480720	-0.989653
75%	2494.750000	25.000000	45046.000000	194.500000	1.117500	20448.250000	2.00000	7210.000000	3763.687500	199.91750	2023.000000	800.000000	400.000000	160.000000	7.44	6.550000	1.350000	40.431275	-0.391633
max	5380.000000	105.000000	81880.000000	314.000000	7.440000	24440.980000	7.00000	42167.000000	19600.580000	381.25000	2024.000000	1200.000000	1200.000000	840.000000	7.44	15.500000	7.700000	40.548198	-0.266404

Este resumen estadístico ofrece una visión general de los datos, mostrando tanto la centralidad (medias y medianas) como la dispersión (desviación estándar y rangos) de las diferentes variables. La alta variabilidad en ciertas columnas, como "DENSIDAD" y "RM/PERSONA", sugiere que estas variables pueden tener una distribución de datos muy dispersa, mientras que otras como "Pruebas" y "Coste/laboral/hora" tienen una variabilidad nula.

A partir de ahí, creamos un nueva data frame, (df1), eliminando algunas columnas que no nos interesan. Y con ese data frame utilizamos la función. corr y hacemos una representación para ver las variables que más se relacionan:



Correlaciones Altas (cercanas a 1):

- **Clientes y Facturación (0.98)**: Existe una fuerte correlación positiva, lo que sugiere que a medida que el número de clientes aumenta, la facturación también lo hace.
- **Entrenadores y Horas/entrenadores/MES (0.93)**: Una fuerte correlación positiva indica que un mayor número de entrenadores está asociado con más horas trabajadas por entrenadores por mes.

- **Clientes y Altas (0.85)**: Existe una fuerte correlación positiva entre el número de clientes y las altas, sugiriendo que a medida que aumentan los clientes, también aumentan las altas.
- **Facturación y Altas (0.88)**: Similarmente, la facturación aumenta con las altas.

Correlaciones Negativas (cercanas a -1):

- **Fundación y Facturación (-0.24)**: Una correlación negativa débil, lo que sugiere que podría haber una ligera tendencia de que a medida que la fundación incrementa, la facturación podría disminuir.
- **CANON y Fundación (-0.4)**: Una correlación negativa moderada, indicando que podría haber una relación inversa entre el canon y la fundación.

Algunas correlaciones Intermedias:

- **Socios y Entrenadores (0.58):** Una correlación positiva moderada, lo que indica que más socios están asociados con más entrenadores.
- **RM/PERSONA y Facturación (0.42):** Una correlación positiva moderada, sugiriendo que una mayor relación entre RM por persona puede estar asociada con mayor facturación.

Y a modo de resumen, vemos que la variable (Fundación) es la que tiene una relación lineal negativa con el resto de las variables. Por lo que podemos pensar que el año que la empresa cogió el club, no es importante a la hora de conseguir más clientes.

Por otro lado, se puede apreciar que las variables “Clientes” y “Facturación”, son las que tiene en general correlaciones más fuertes.

Después de este tipo de información, probamos diferentes representaciones gráficas para que nos den información de cómo se estructuran los diferentes clubes en la empresa.

Hacemos un bucle para dividir los clubes en función de los entrenadores que tenemos por club.

```
# Distribución de clubes por niveles según entrenadores.
for i in range(len(df)):

  if df.Entrenadores.values[i] <= 2:
    df.loc[i,'Club_Entrenador'] = 'Pequeño'

  elif (df.Entrenadores.values[i]  >= 3) & (df.Entrenadores.values[i]  <= 4):
     df.loc[i,'Club_Entrenador'] = 'Mediano'

  else: df.loc[i,'Club_Entrenador'] = 'Grande'

df.Club_Entrenador.value_counts()
```

```
Club_Entrenador
Pequeño    34
Mediano     4
Grande      2
Name: count, dtype: int64
```

De esta manera también preparamos el bucle para cualquier otro tipo de variable que queramos categorizar. Por ejemplo, evaluar los clubes en función del número de clientes o del ticket medio que tienen.

También trasladamos varias funciones usadas en el Master, para poder graficar los datos y que nos den información. Los veremos en el siguiente punto.

CLIENTES

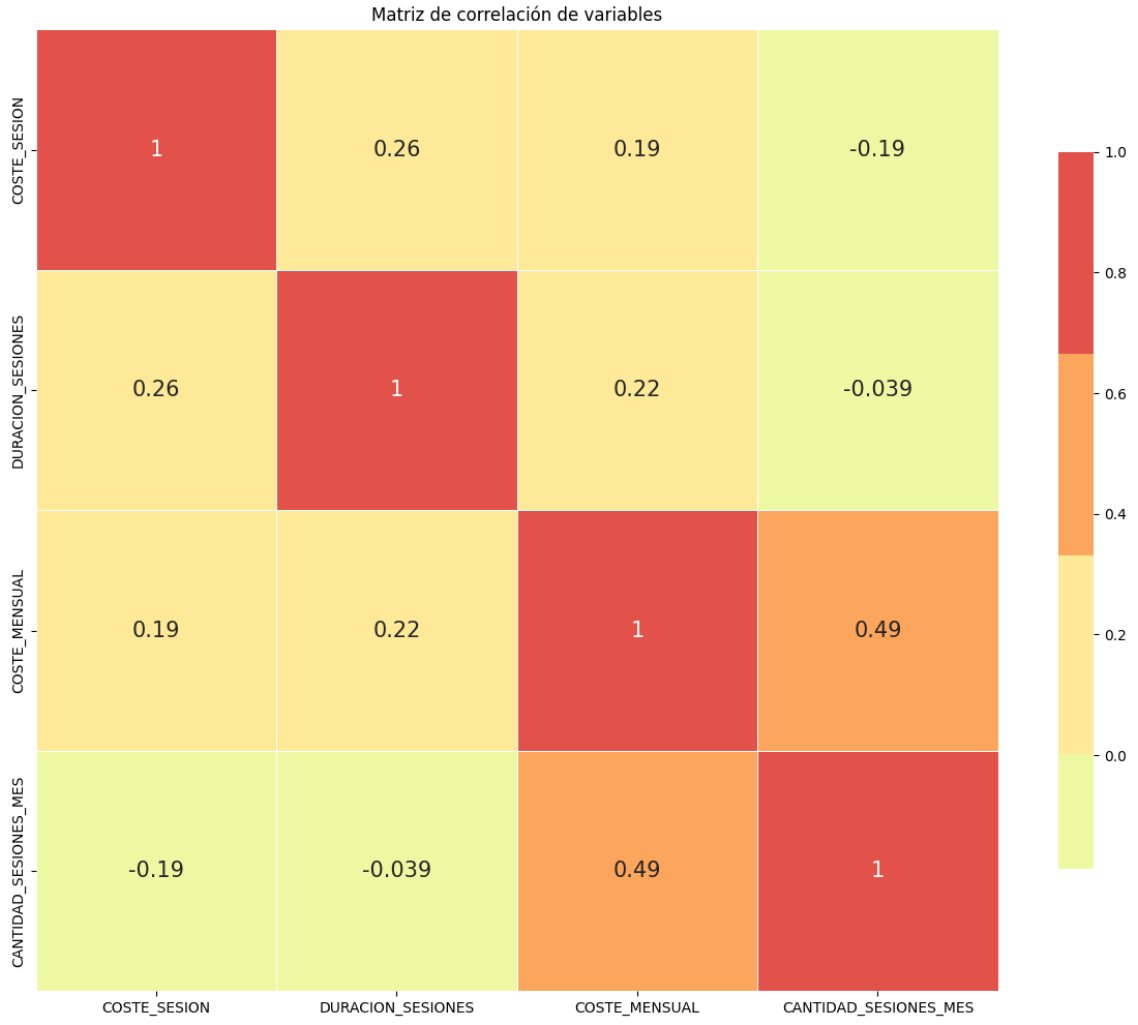
Aplicamos la función describe para conocer un poco nuestros datos numéricos:

```
df.describe()
```

	CP CLUB	COSTE_SESION	CANTIDAD_SESIONES_MES	DURACION_SESIONES	COSTE_MENSUAL
count	5899.000000	5981.000000	5981.000000	5981.000000	5981.000000
mean	30093.747076	30.558786	4.888148	33.626484	144.360491
std	10172.524789	20.410265	3.964689	16.537135	85.611141
min	3005.000000	0.000000	0.000000	0.000000	0.000000
25%	28002.000000	23.750000	4.000000	30.000000	91.950000
50%	28033.000000	28.450000	4.000000	30.000000	135.000000
75%	28823.000000	35.000000	8.000000	45.000000	190.000000
max	46920.000000	369.800000	140.000000	60.000000	994.000000

Hay variabilidad significativa en los códigos postales, los costos por sesión y los costos mensuales. La cantidad de sesiones por mes tiene una mediana de 4, lo que indica que muchas personas asisten a 4 sesiones mensuales. La duración de las sesiones es mayormente de 30 a 45 minutos.

Hacemos otra correlación de los valores numéricos para ver si hay alguna relación clara:



Se puede apreciar, que no hay una correlación muy fuerte entre ninguna variable, siendo la más potente la correlación entre las variables coste mensual y cantidad de sesiones mes. Si hay correlaciones lineales negativas.

Dividimos los data frames por el tipo de cliente, por si queremos hacer análisis directamente de un tipo de usuario.

```
[45] df2=df[df['TIPO_CLIENTE']=='Potencial']
```

```
[46] df2.head()
```

	ENTRENADOR	TIPO_CLIENTE	CP CLUB	CLUB	SEXO	FECHA_NACIM	CP	COSTE_SESION	CANTIDAD_SESIONES_MES	DURACION_SESIONES	COSTE_MENSUAL	MEDIO_PAGO	REGIONAL
1	Camilo Restrepo	Potencial	46120.0	BF Alboraya	F	19/12/1966	46011	25.0	8.0	45.0	199.0	TPV Virtual	PEDRO
3	Camilo Restrepo	Potencial	46120.0	BF Alboraya	M	15/04/1964	46011	35.0	4.0	45.0	140.0	Domicialización	PEDRO
4	Camilo Restrepo	Potencial	46120.0	BF Alboraya	M	24/08/1988	NaN	0.0	0.0	0.0	0.0	Efectivo	PEDRO
5	Camilo Restrepo	Potencial	46120.0	BF Alboraya	F	08/11/1979	46011	25.0	8.0	45.0	199.0	Domicialización	PEDRO
7	Emi Rossi	Potencial	46120.0	BF Alboraya	F	07/12/1985	46133	18.0	4.0	45.0	70.0	TPV Virtual	PEDRO

```
df3=df[df['TIPO_CLIENTE']=='Entrenado']
df3.head()
```

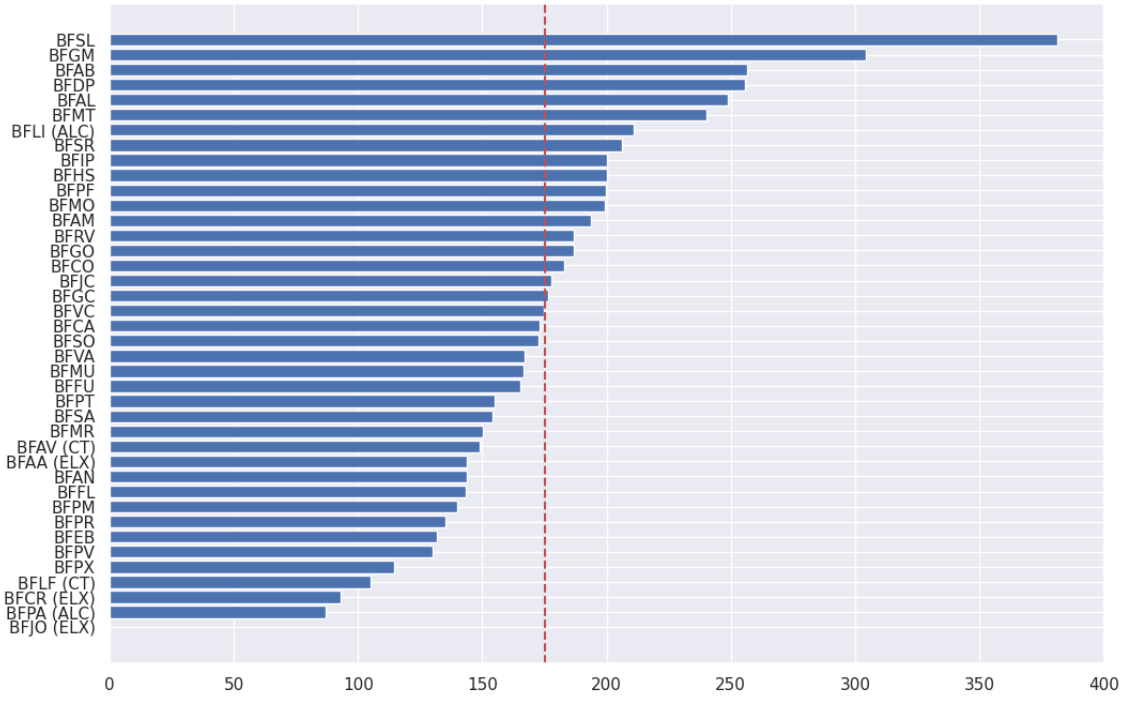
	ENTRENADOR	TIPO_CLIENTE	CP CLUB	CLUB	SEXO	FECHA_NACIM	CP	COSTE_SESION	CANTIDAD_SESIONES_MES	DURACION_SESIONES	COSTE_MENSUAL	MEDIO_PAGO	REGIONAL
2	Camilo Restrepo	Entrenado	46120.0	BF Alboraya	M	18/03/1994	46011	36.0	5.0	45.0	180.0	Domicialización	PEDRO
6	Camilo Restrepo	Entrenado	46120.0	BF Alboraya	F	30/05/1978	46011	35.0	2.0	45.0	70.0	TPV Virtual	PEDRO
14	Entrenador Bajas Entrenadores	Entrenado	46120.0	BF Alboraya	F	04/08/1999	46020	35.0	2.0	45.0	70.0	TPV Virtual	PEDRO
32	Victoria Abbondanza	Entrenado	46120.0	BF Alboraya	NaN	23/01/1977	46011	35.0	4.0	0.0	140.0	Domicialización	PEDRO
36	Victoria Abbondanza	Entrenado	46120.0	BF Alboraya	NaN	25/02/1979	NaN	40.0	5.0	0.0	200.0	Domicialización	PEDRO

4. VISUALIZACIONES

A continuación, pasamos a mostrar y analizar algunos de los gráficos realizados en nuestros análisis.

Aprovechamos las gráficas usadas en el master, para hacer varios “bar plot” , observando así cuales son los clubes “líderes” en ciertos aspectos:

Listado clubes según el ticket medio que tienen por cliente.

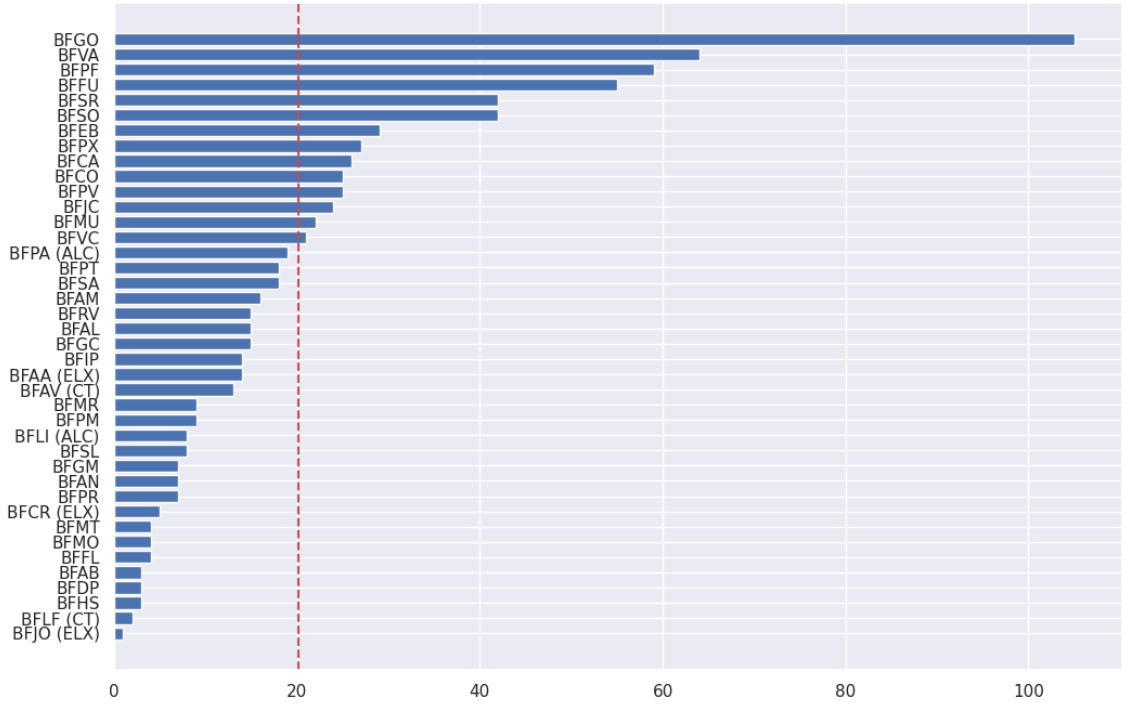


Podemos destacar dos clubes como **BFSL** y **BFGM** que tienen un ticket medio notablemente alto, lo que podría indicar un mayor valor por transacción en estas categorías. Encontraríamos áreas de mejora en:las categorías con tickets medios bajos, como **BFJO (ELX)** y **BFPA (ALC)**, podrían necesitar estrategias para aumentar el valor medio de sus transacciones.

Observamos que el punto de referencia (La línea roja) de la media global ayuda a identificar rápidamente cuáles categorías están por encima o por debajo del promedio, facilitando la identificación de outliers y tendencias.

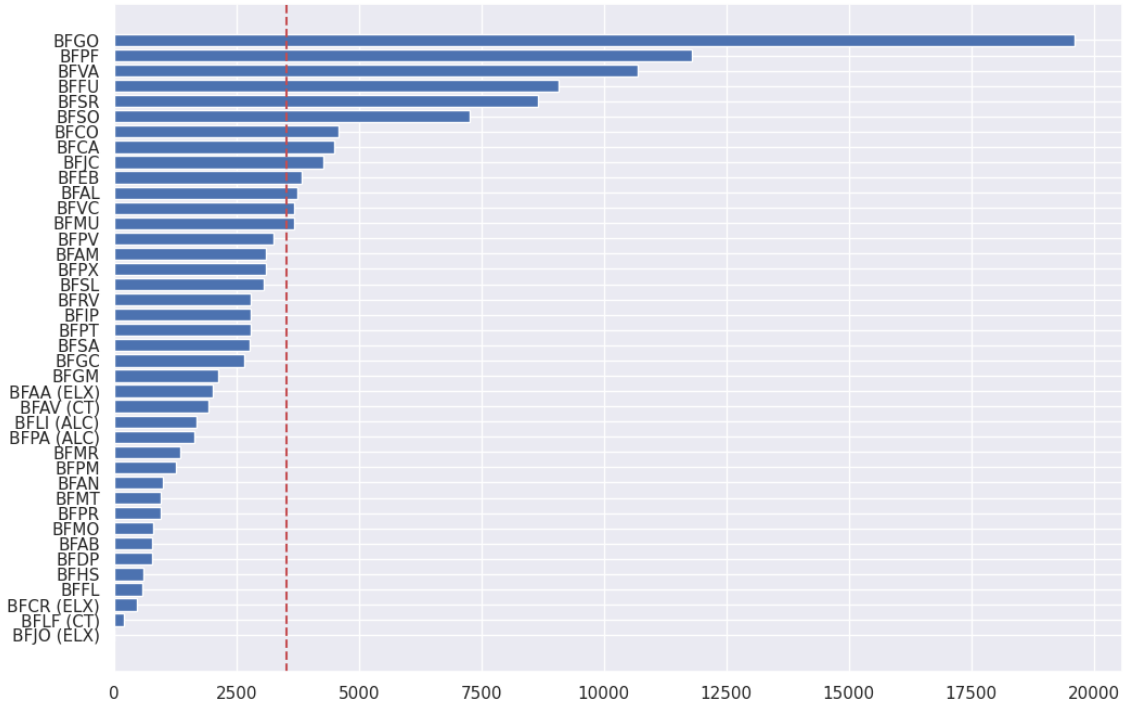
Este análisis puede ser útil para tomar decisiones estratégicas, como enfocar esfuerzos de marketing en categorías con tickets medios bajos o investigar por qué ciertas categorías superan el promedio global.

Listado clubes según los clientes que tienen



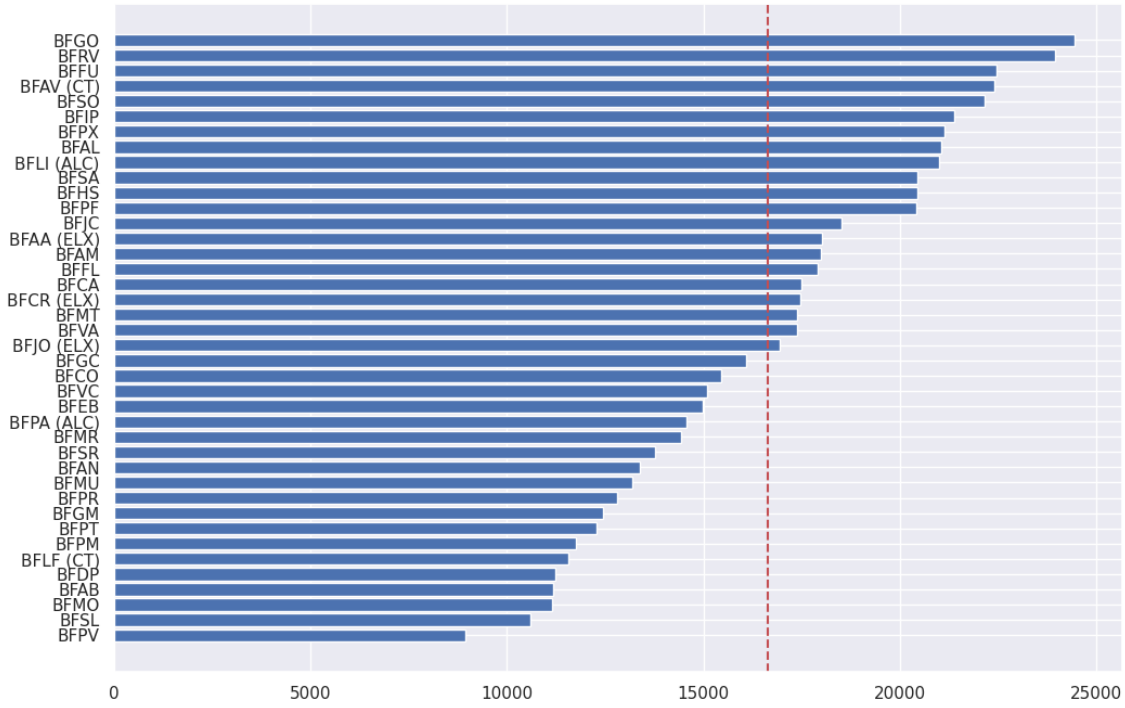
Podemos observar que en función de los clientes, hay un club que destaca por encima del resto, y al igual que en el gráfico anterior, la mayoría de los clubes están por debajo de la media. Este análisis nos puede ayudar a encontrar que entidades necesitan más apoyo y cuales necesitan una mayor asignación de recursos para mejorar.

Clasificación clubes según facturación.



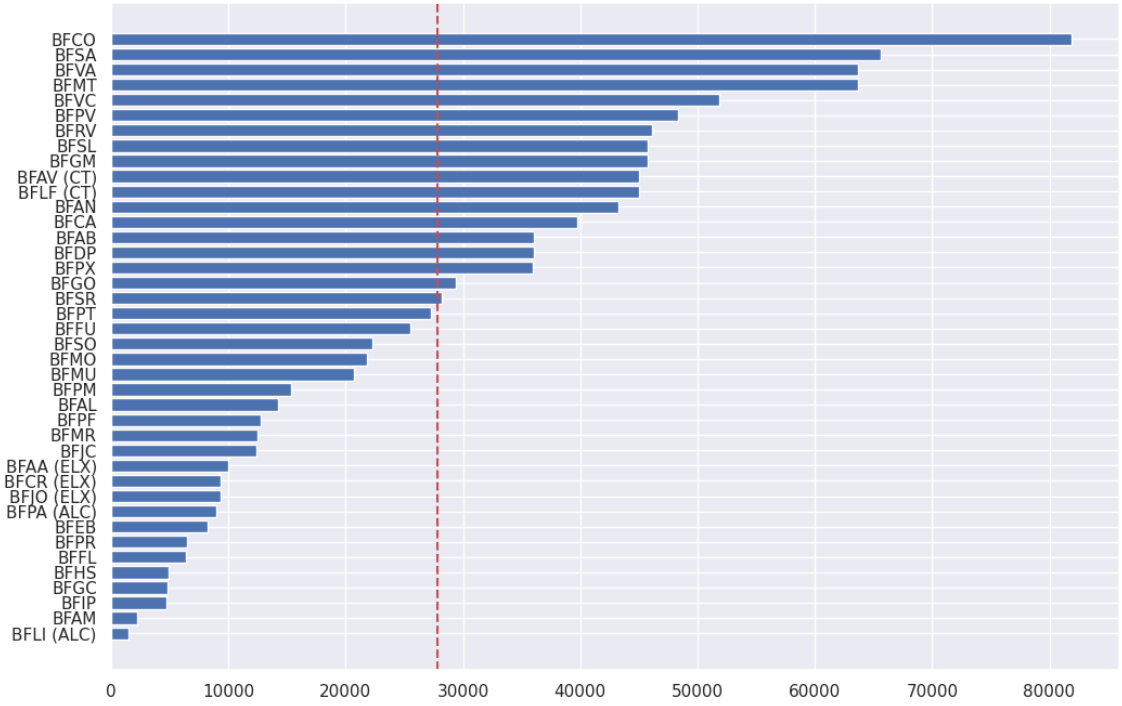
Vemos como en este gráfico hay menos clubes destacados, o por encima de la media, Proporcionando una visión clara del rendimiento financiero de los diferentes clubes, pudiendo ser de utilidad para toma de decisiones estratégicas.

Clasificación clubes según la renta neta media por persona



Vemos que, en esta gráfica, los clubes están más repartidos. Viendo algunos clubes que en otras gráficas anteriores están en posiciones inferiores y en este "avanzan" en la clasificación.

Clasificación clubes según población.



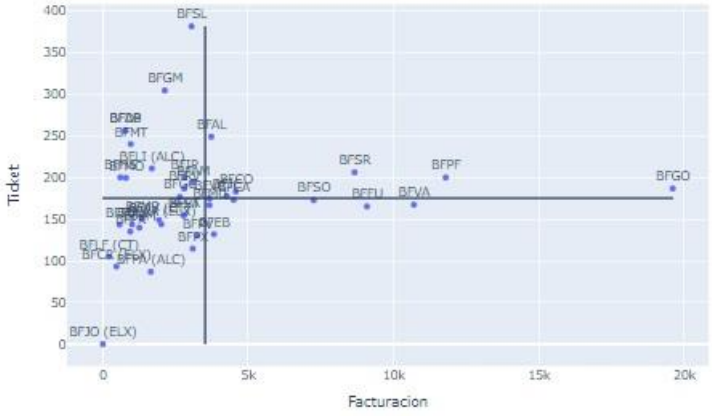
En cuanto a la población, de alrededor del club, vemos que también está bastante repartido en los clubes, cambiando el ranking, no estando directamente relacionados con clubes de mayor o menor facturación como en gráficos anteriores.

También, a través de un scatter plot, visualizamos dos variables a la vez:

Visualización de número de clientes en relación al número de socios:

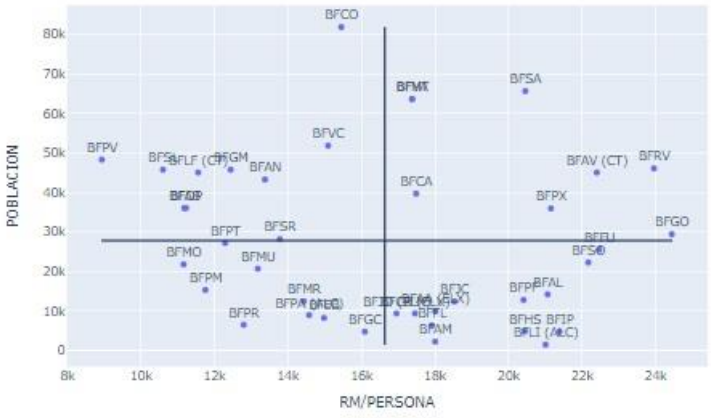
En el anterior gráfico, analizamos dos variables, una “interna” de la empresa y otra del club con el cual tenemos contrato, sobre todo para ver la relación entre el número de clientes que tiene la empresa y los socios del gimnasio. Podemos apreciar que, al igual que en otros gráficos, la mayoría de los clubes están por debajo de la media en ambas categorías, siendo, el cuadrante inferior derecho y el cuadrante superior izquierdo donde se concentran menor número de clubes.

Visualización de facturación en relación con el ticket y a la facturación.



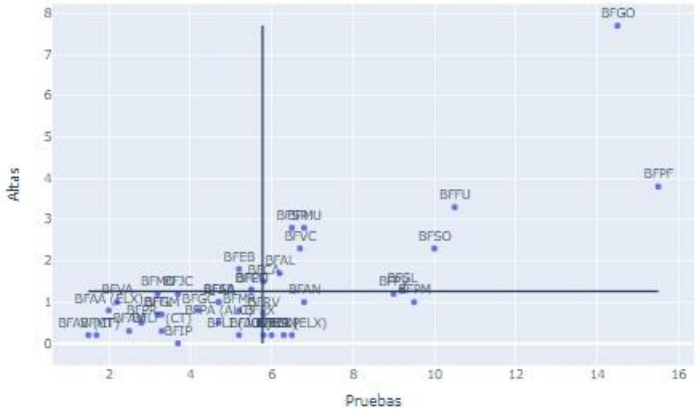
Cotejando estas dos variables, vemos que vuelve a haber un club destacado del resto, sobre todo en cuanto a facturación, aunque estén ligeramente por encima del ticket medio. Viendo que los clubes con un ticket muy alto , no necesariamente tiene más facturación. Siendo BFSR, el club que más facturación tiene con el ticket más alto.

Visualización de la población y la renta media por persona:



Este es el gráfico con más dispersión de todos, donde vemos la gran variedad que hay dentro de la empresa. En un primer vistazo, vemos clubes que en esta faceta no están “destacados,” pero en anteriores gráficos sí.

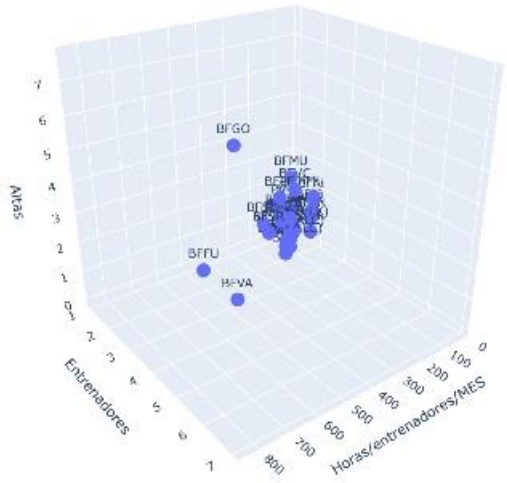
Visualización del número de pruebas y el número de altas:



Esta gráfica, es la que más depende de los “entrenadores”, es la menos dispersa de todas las que hemos visto, donde la tendencia principal es a agruparse en el “centro”. Estando el mayor grupo en el cuadrante inferior izquierdo.. ¿Puede explicar esto un comportamiento general?

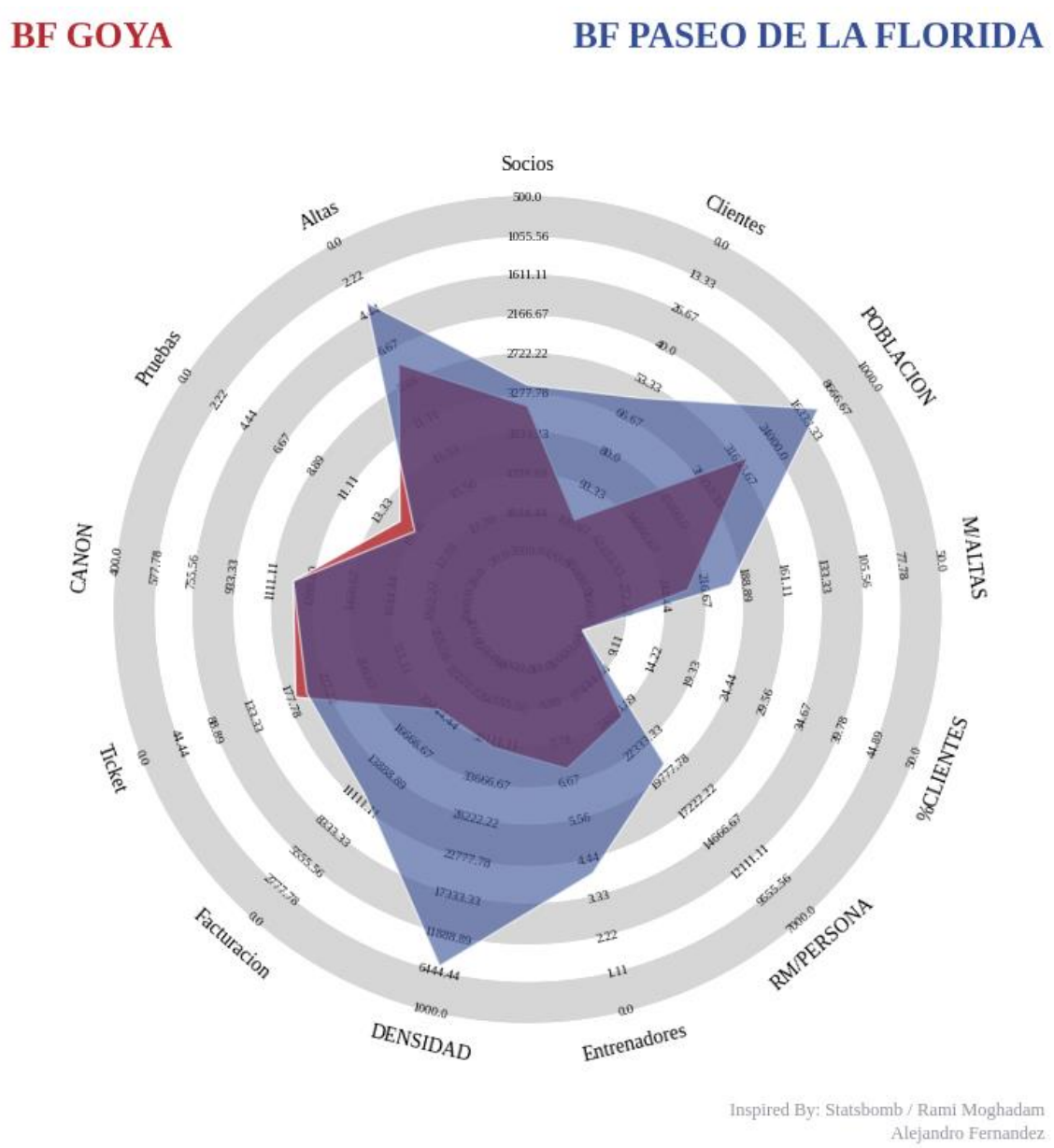
También usamos el scatter plot 3D para visualizar tres variables al mismo tiempo:

Visualización Scatter Plot 3D de Altas, Entrenadores y las horas de entrenadores/mes



En este gráfico 3D, al comparar tres variables, vemos que clubes están “destacados” en una, dos o hasta 3 variables. Es interesante para cotejar varias variables a la vez, y así poder evaluar el desempeño de cada club individualmente.

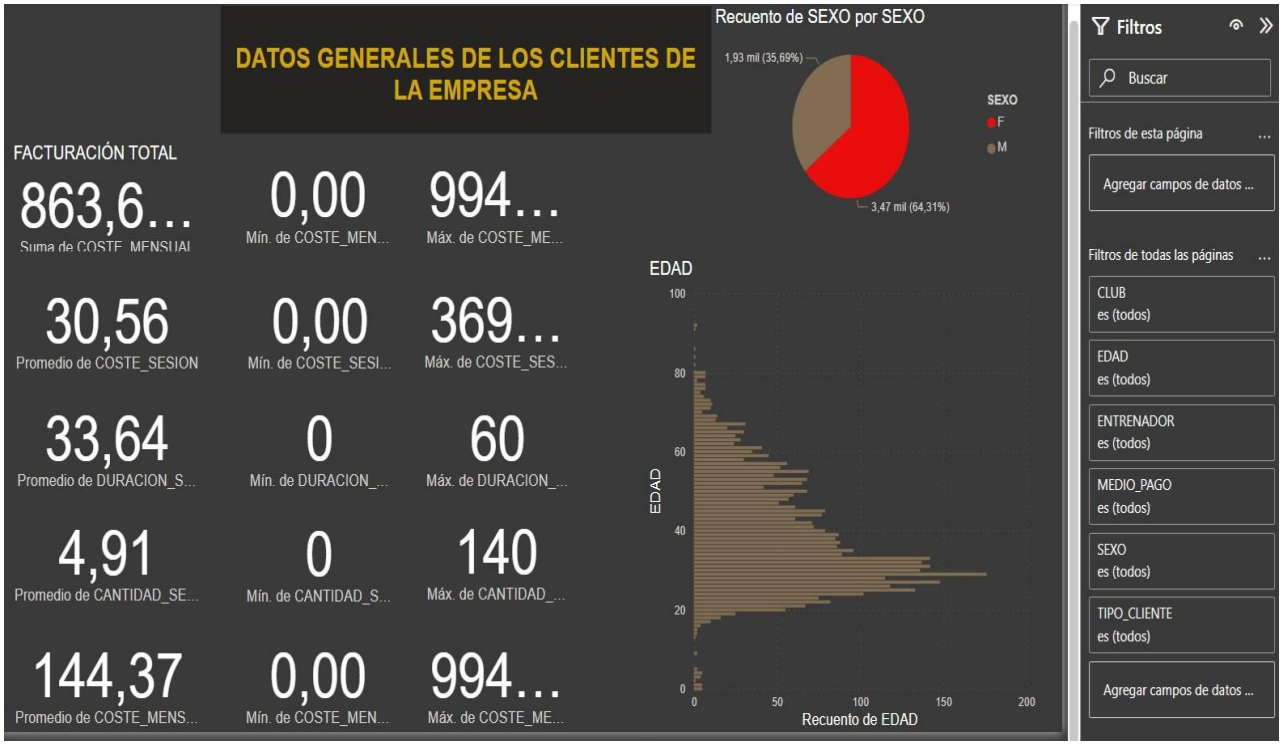
Y, por último, usamos los gráficos de radar de stats bomb para aplicar los datos de cada club y poder comparar club con club.



Eligiendo dos clubes con rendimiento/facturación por encima de la media, vemos que hay diferencias notables entre uno y otro. El análisis comparativo entre clubes puede ser una buena manera de encontrar patrones o diferencias notables entre clubes.

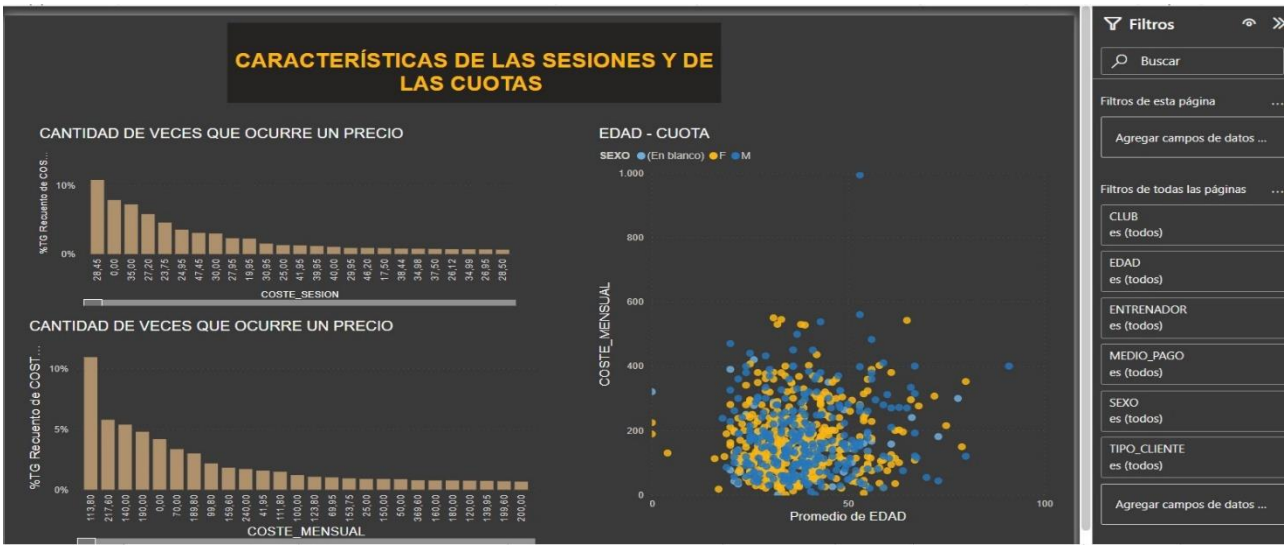
En cuanto al documento de clientes, lo traspaso a power bi, donde hacemos un dashboard para ver los datos más interesantes de los mismos.

Dashboard inicial, resumen, de la hoja “clientes”.



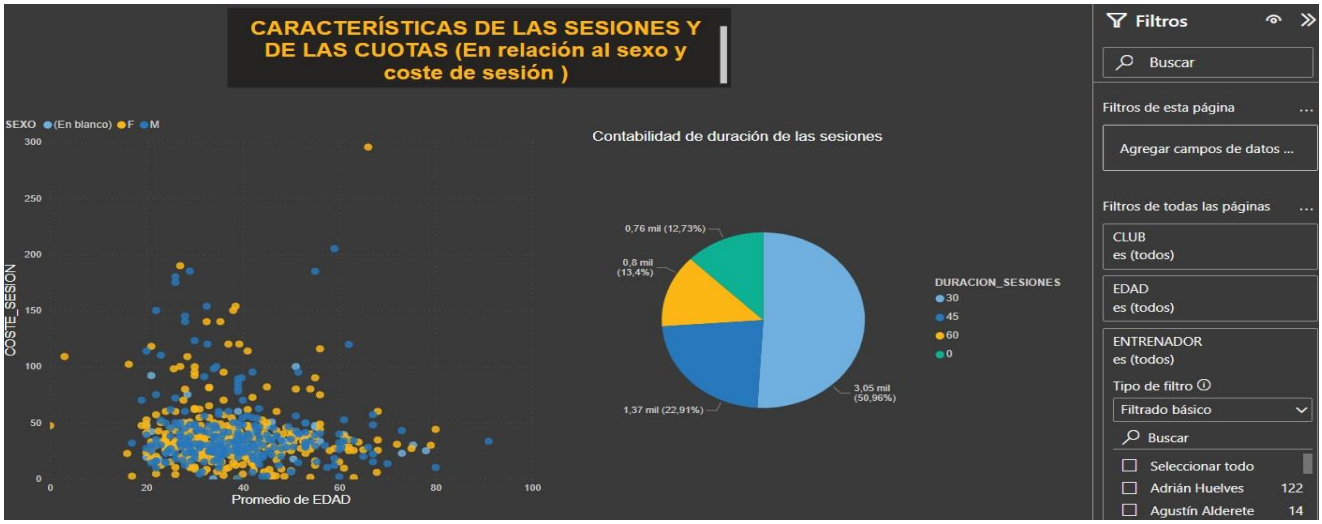
Esta foto, representa, los porcentajes de clientes Masculinos o Femeninos que tiene la empresa, viendo que el femenino es el porcentaje mayor, en torno al 65% siendo el resto clientes masculinos, también se aprecia un rango de edades entre los 30 y los 40 años. Además, incluimos algunos KPI, de suma de facturación, coste de sesión, duración de sesión, cantidad de sesiones a la semana, y el ticket medio.

Recuento de la cantidad de veces que ocurre un precio de una clase o de una cuota mensual. Además de un gráfico de la relación entre edad-sexo-cuota mensual.



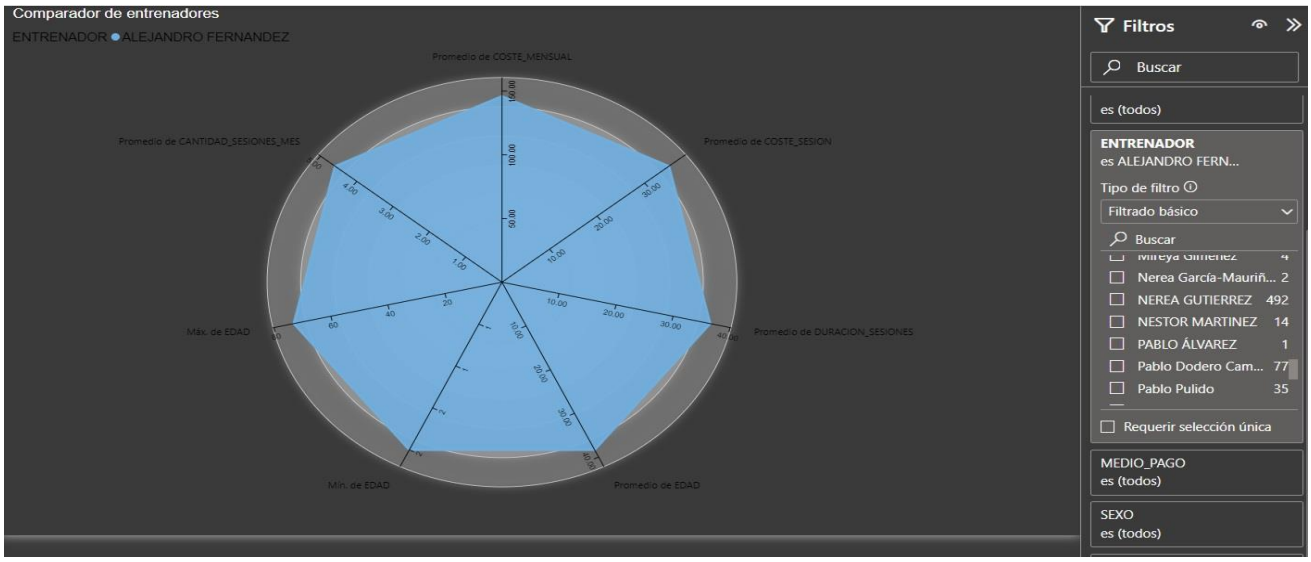
En la foto anterior, vemos que el precio que más se repite es la cuota más pequeña de las que posee la empresa, tanto en coste mensual como en coste sesión. En el gráfico de dispersión, apreciamos un mayor cúmulo en la misma franja de edades independientemente del sexo o la edad

Relación edad-sexo- coste de sesión:



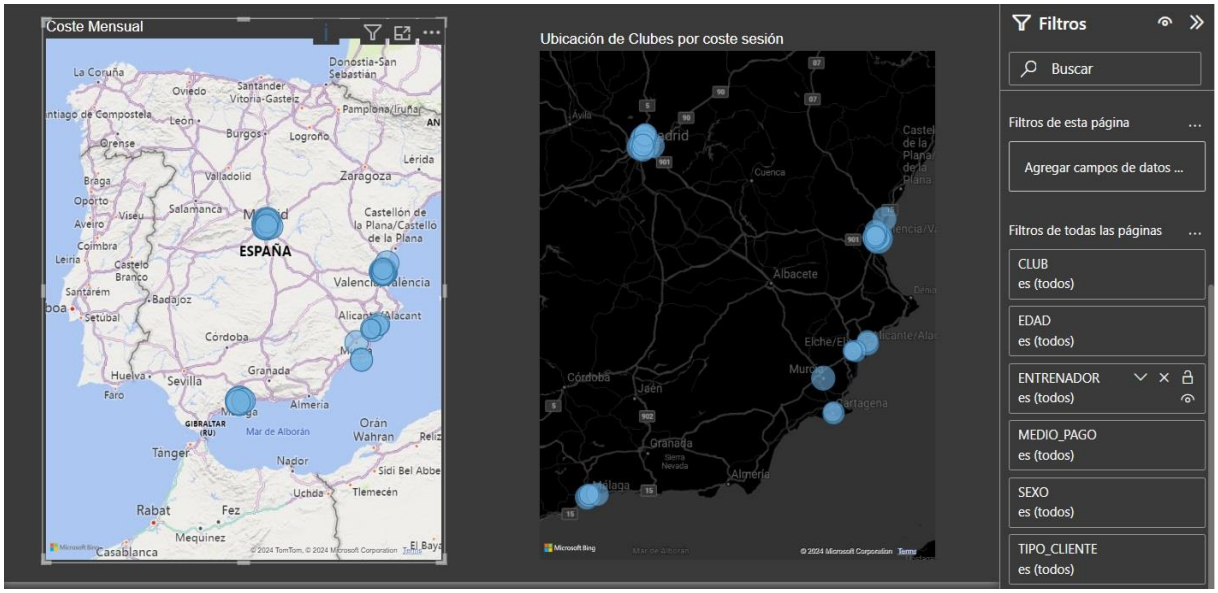
Apreciamos un gráfico de dispersión que en una franja “baja” de los precios, al igual que en el gráfico circular, apreciamos una que la mayoría de las sesiones son de 30´.

Radar chart para comparar entrenadores de manera rápida:



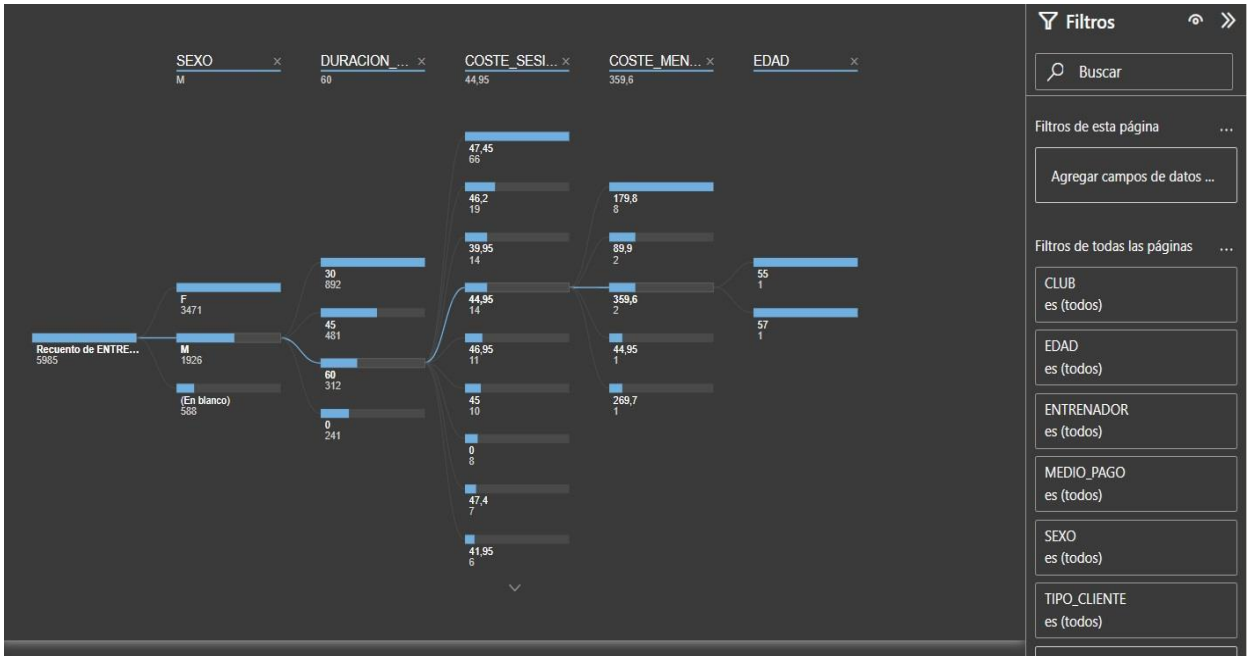
Este gráfico de RADAR, lo utilizaremos para comparar entrenadores, están incluidos las variables mencionadas en fotos anteriores para la evaluación de los mismos.

Mapas de la distribución de los clubes en función de su coste mensual, y del precio de las sesiones:



Vemos en los mapas, la distribución de los clubes en función del ticket medio o del precio de la clase, para encontrar más fácilmente como están catalogados los clubes.

Esquema jerárquico para explicar los clientes de cada entrenador:



Con este esquema jerárquico, lo que queremos ver principalmente es como se comporta un cliente de cada entrenador en función de diferentes características.

5. CONCLUSIONES

RESUMEN

En el apartado de estudio de los clubes, podemos ver que hay grandes diferencias entre clubes, sobre todo en lo relativo a datos de "facturación" o número de clientes, estando la mayoría de los clubes por debajo de esa media.

Y, cuando aplicamos. describe al documento, vemos que hay desviaciones muy grandes entre los diferentes clubes. ¿Qué hace diferente unos de otros?

Por los datos reflejados parece que el dato más clave en cuanto a rendimiento, se hace importante en cuanto el número de entrenadores sea mayor. A mayor número de entrenadores mayor facturación.

Se podría pensar que el entorno influye mucho en la facturación, refiriéndome principalmente a la densidad/población, o renta media por persona, pero no están directamente relacionado con el rendimiento de un club. O por lo menos, no es lo que reflejan nuestros datos. Al igual que estos datos tampoco influyen en el ticket medio que tiene el club en cuestión.

Analizando otros aspectos, como las altas o las pruebas, efectivamente vemos que, a más pruebas de servicio mayor número de altas. Explicando esto la base del negocio que es hacer pruebas. En este aspecto tendríamos nos faltarían una serie de datos que propondré en el apartado siguiente, de cara a medir la fidelización de los clientes o el tiempo que duran con nosotros. Y analizarlo por club y entrenador.

Actualmente la política de la empresa se hace a través de la contratación de entrenadores de manera moderada. Y habría que plantear fases comerciales. Donde se dedique mucho tiempo a la generación de nuevos leads y de pruebas.

En cuanto al análisis de nuestros clientes, podemos apreciar que en un 65% el público que atendemos es femenino siendo la edad más repetida los 30 años, viendo como la "pirámide" es más ancha en la franja entre los 30-40 años. Y estrechándose a medida que bajas o subes de esas edades. ¿Puede ser por motivo de compra?, ¿por poder adquisitivo?, ¿por percepción de valor? Ese tipo de preguntas nos las debemos de plantear como empresa, sobre todo para facilitar el proceso de vida de ese cliente, desde que es un lead hasta que se convierte en cliente.

A esto, podríamos sumarle el aspecto de fidelización del cliente.

Siguiendo nuestro análisis, podemos apreciar que las mayores cuotas están muy repartidas entre diferentes grupos de edad, no habiendo una diferencia clara entre las cuotas entre diferentes edades o sexos.

Podemos ver que la cuota más repetida es 113,80 que corresponde a la de 1 día a la semana de 30´. Con una tarifa de 27€ la sesión, que a su vez también es el precio sesión más repetido.

Con respecto a la duración de las sesiones, lo que más se repite, son las sesiones de 30´.

En lo relativo a los mapas, vemos que el ticket medio mayor se concentra en Madrid, aunque no muy lejano en el resto de ciudades, siendo la región más "baja" en cuanto a ticket: Elche/Alicante.

Avanzando en el análisis pasamos a la franja de los entrenadores, y vemos que los entrenadores con más "recorrido" o más número de valores asociados, son los que tienen todas las medias más elevadas, demostrando que "cuanto más alto el

precio” mejor sale el entrenador valorado. Comparados con otros entrenadores con menor recorrido se ve la diferencia clara en cuanto a los diferentes ítems. Aunque aquí faltaría, evaluar otra serie de aspectos que estarían en propuestas.

Con el esquema jerárquico, evaluamos cada entrenador particularmente para ver cuál es el perfil que más se repite, y podremos buscar si hay algún patrón general, o los entrenadores tienen comportamientos diferentes, aunque en un primer vistazo, normalmente los entrenadores cumplen los rasgos de la empresa, solo se ven diferencias en los tickets. Esto, nos da información de cómo se comporta nuestro servicio en el lugar que estamos, pudiendo facilitar la formación en ciertos aspectos.

RECOMENDACIONES / PROPUESTAS

Paso a redactar algunas propuestas de mejora de cara a poder mejorar el servicio y crear un buen sistema de obtención de datos, para utilizar los mismos a nuestro favor:

1. Cuantificación de la fecha de alta de los entrenadores, para ver el desempeño de entrenadores más veteranos vs los más jóvenes.
2. Cuantificación de la fecha de baja de los entrenadores y ver su desempeño en varias áreas.
3. Cuantificación de la fecha de alta y baja de los clientes, que pueda aparecer en el Excel que se ha enviado. Para poder evaluar la fidelización del cliente
4. Ajustar el servicio en base a características de clientes, para ello sería necesario poner como obligatorio en la recopilación de datos por qué se apunta al cliente.
5. Facilidad en la toma de datos, hoja de PRIMER CONTACTO y luego EVALUACIÓN INICIAL. Diferenciar bien una de otra, para que así podamos cuantificar bien la pirámide desde que conseguimos el Lead, hasta que hagamos la prueba y luego el resultado de esa misma prueba.

6. Facilitar la toma de datos poniendo la obligatoriedad de datos de ciertos aspectos para que no se pierda información.
7. Ajustar el servicio en base a las características del cliente: protocolización de calentamientos, progresiones etc., dejando a criterio del entrenador el paso a diferentes estadios y trato con cliente.
8. Poder apuntar las clases realizadas en la aplicación, para poder evaluar patrones de cara a predecir bajas o anticipar subidas de cuotas.

TRABAJO A FUTURO:

Conseguir diferentes datos que nos sean útiles para poder hacer algoritmos de IA, como clustering, regresiones, redes neuronales.

Tener claro cuál es el comportamiento de los clientes o potenciales usuarios, para poder establecer cuál es el cliente objetivo de cada entrenador.

Evaluar a cada entrenadore con más datos, para poder predecir comportamientos y poder cuantificar y marcar objetivos y KPI's más claros y cuantificables. De cara a conseguir mejorar el rendimiento de los entrenadores en sus primeros meses de entrada en la empresa, y poder así, mejorar los procesos formativos.

6. REFERENCIAS.

- APLICACIÓN GYM – GY de la empresa FUNZIONA.
- Material educativo del campus virtual de Sport Data CAMPUS.
- https://valenciaplaza.com/jorge-juan-rascanya-calles-valencia-mayor-menor-renta-espana
- https://www.epdata.es/datos/renta-municipios-datos-estadisticas-agencia-tributaria/201/sagunto/6234
- https://www.informacion.es/economia/2023/10/28/son-barrios-ricos-pobres-alicante-93232683.html
- https://econet.carm.es/inicio/-/crem/sicrem/PU_CartagenaCifrasNEW/P8017/sec2.html
- https://datosabiertos.malaga.eu/
- https://www.diariosur.es/malaga-capital/consulta-barrios-rico-pobre-malaga-20210429203237-nt.html?ref=https%3A%2F%2Fwww.diariosur.es%2Fmalaga-capital%2Fconsulta-barrios-rico-pobre-malaga-20210429203237-nt.html
- https://www.murcia.es/documents/11263/2168395/Poblacion-de-Derecho-Habitantes-Barrios-Pedanias.pdf
- https://gestrisam.malaga.eu/export/sites/gestrisam/.galleries/Poblacion-2022/6_Poblacion_-por_Distritos_Municipales_Barrio_y_Sexo.pdf
- https://es.wikipedia.org/wiki/Demograf%C3%ADa_de_Valencia
- https://citypopulation.de/