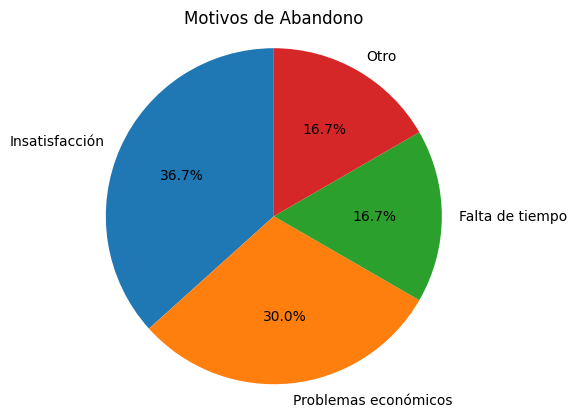
# 1.-Selección

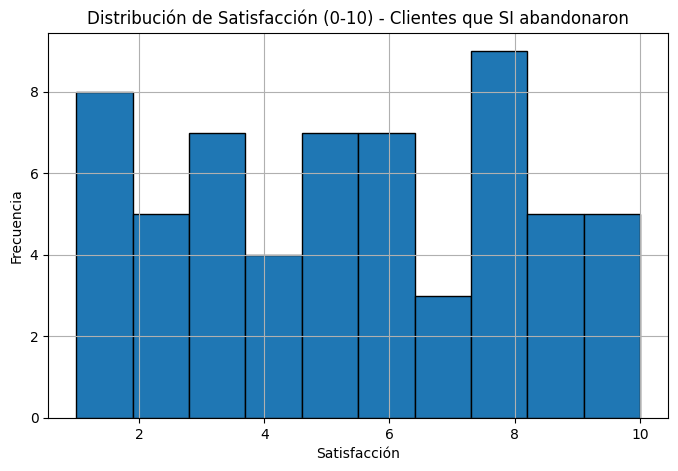
Pablo: Los datos los hemos obtenido realizando encuestas a los miembros del gimnasio, tanto para los que han abandonado como los que siguen en el gimnasio.

Jose Domingo: Dado que la matriz de correlación nos indica que no existen relaciones destacables entre ninguna columna, no podemos eliminar a ninguna basándonos en este criterio. Solo quitamos la columna ID debido a que no aporta nada al resto de las demás ni se correlaciona en nada con las mismas.

Guadalupe:

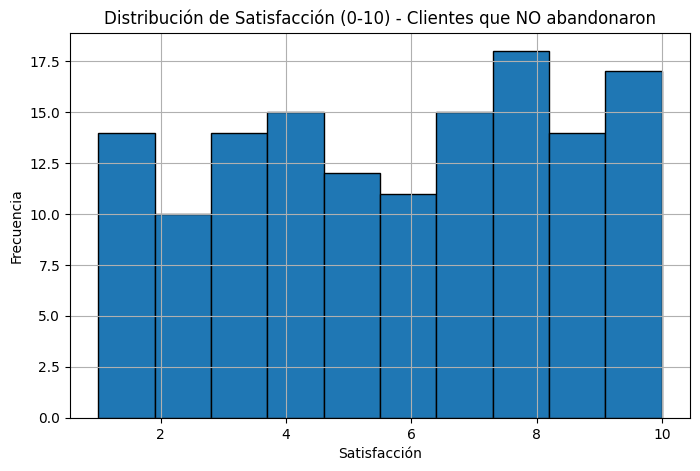
Daniel Marín: Vemos que los resultados de los valores que no son nulos reflejan los siguientes:

* El 36.7% abandonan por insatisfacción.
* El 30.0% abandonan por problemas económicos.
* El 16.7% abandonan por falta de tiempo.
* El 16.7% restante abandonan por otro motivo.



* La satisfacción se distribuye más hacia valores altos (8–10).
* Aunque hay gente con baja satisfacción que se quedó, la mayoría está en la mitad-alta del rango.

(Erich)



* Distribución más plana y dispersa.
* Muchos con niveles bajos de satisfacción (1–4), y menos con puntuaciones altas.
* No hay una concentración clara en valores altos.

(Erich)

|  | Erich | Daniel Marín |
| --- | --- | --- |
|  |  |  |

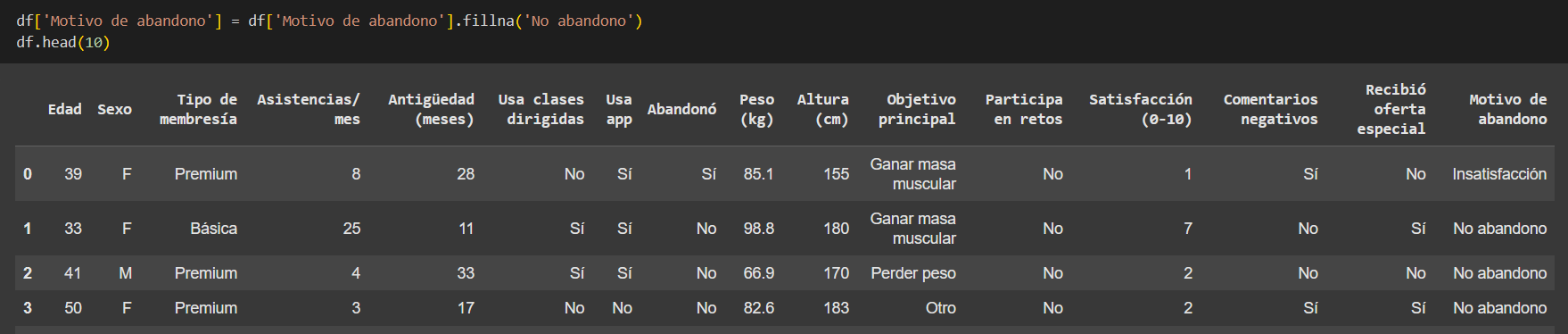
Erich: quitamos la columna de si recibió una oferta especial el cliente.

# 2.-Preprocesamiento

# 

Sergio

Fran Ruiz: Para el campo motivo de abandono se imputa el valor “no abandono” sobre los missing values de dicha columna, ya que representa a las personas que no han abandonado

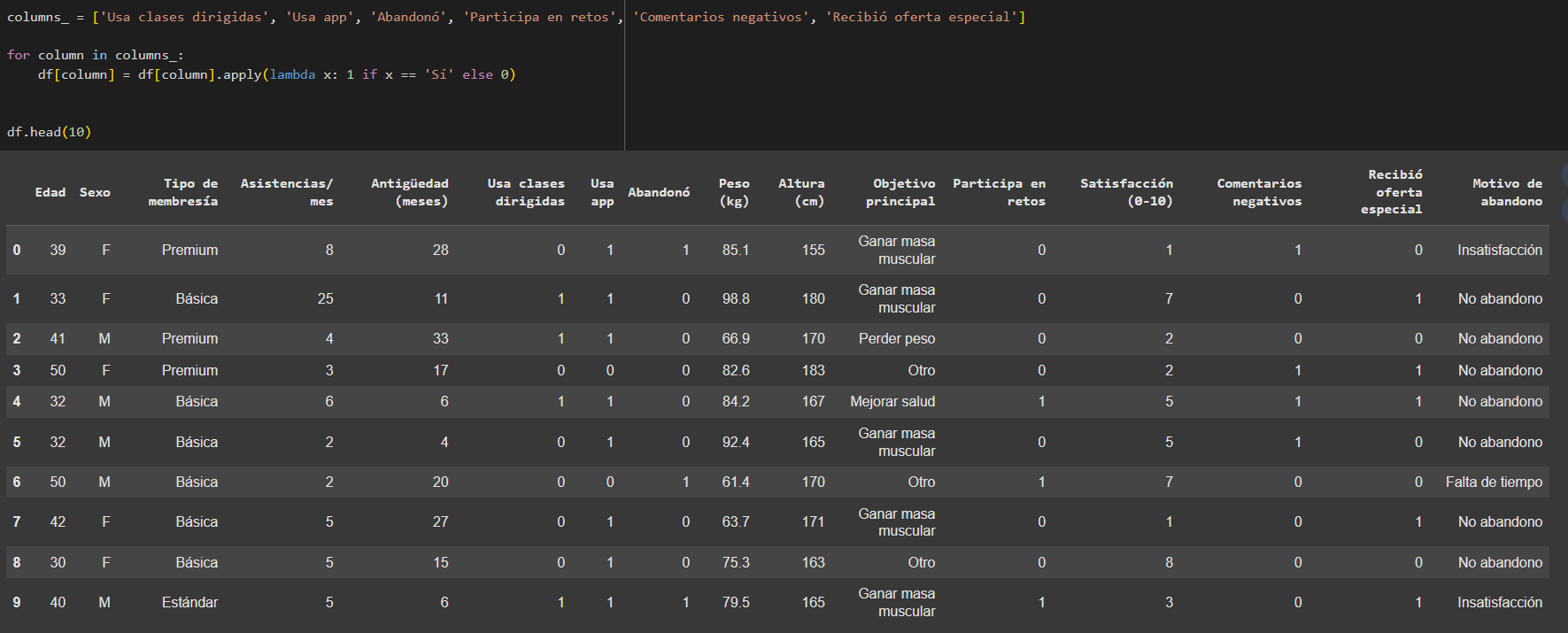


Víctor González: La media de edad de las personas que ha abandonado es de 35 años y medio

.

Víctor González: Hay gente que ha recibido oferta de precio especial, pero en motivo de abandono aparece “Problemas económicos”

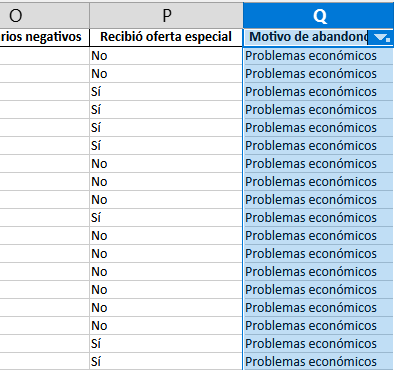
Marta: Podemos categorizar la variable numérica “Satisfacción” en tres niveles: Baja (0-3), Media (4-7), Alta (8-10)

Carlos Carrasquer: Una serie de variables contienen valores cómo “Sí”/”No”. Podemos cambiarlo a valores como 0 y 1 para ayudar al modelo:  
  




# 

Sergio: Tabla para ver insatisfacciones desde el 5 al 10 y se han marchado por insatisfacción.

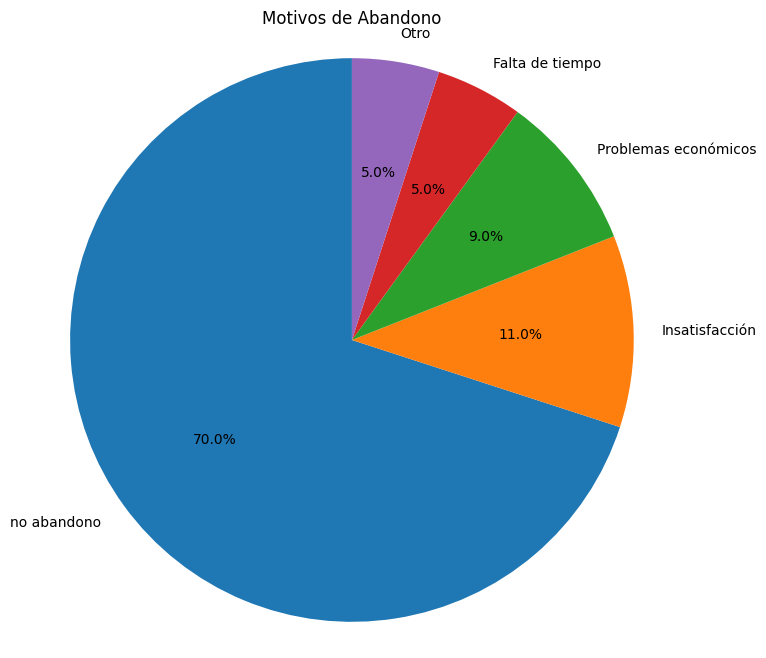


Daniel Marín: Aquí vemos aquellos clientes que aun teniendo la oferta especial han indicado que el motivo de abandono ha sido por problemas económicos.  
  
Carlos Carrasquer: Realizar rangos de edades en vez de usar los datos numéricos cómo tal.

### 

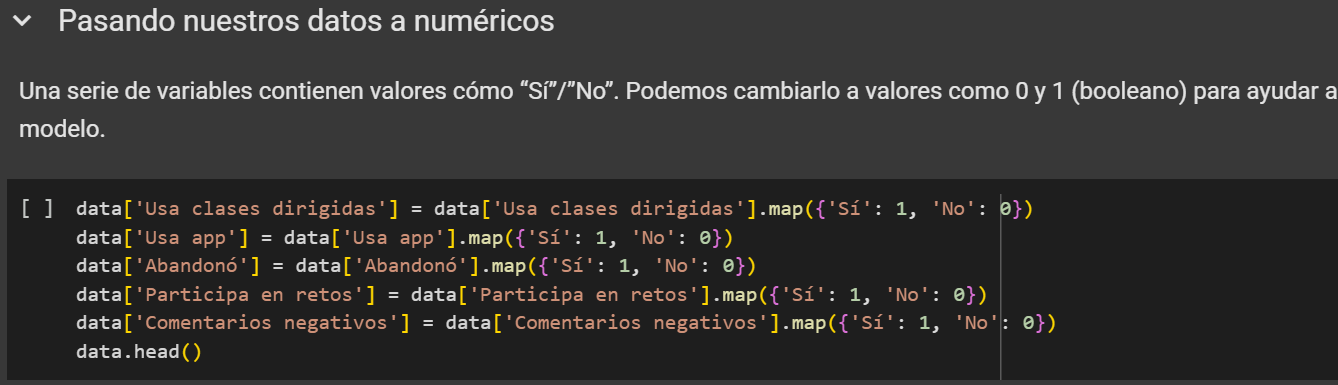
### Valores faltantes

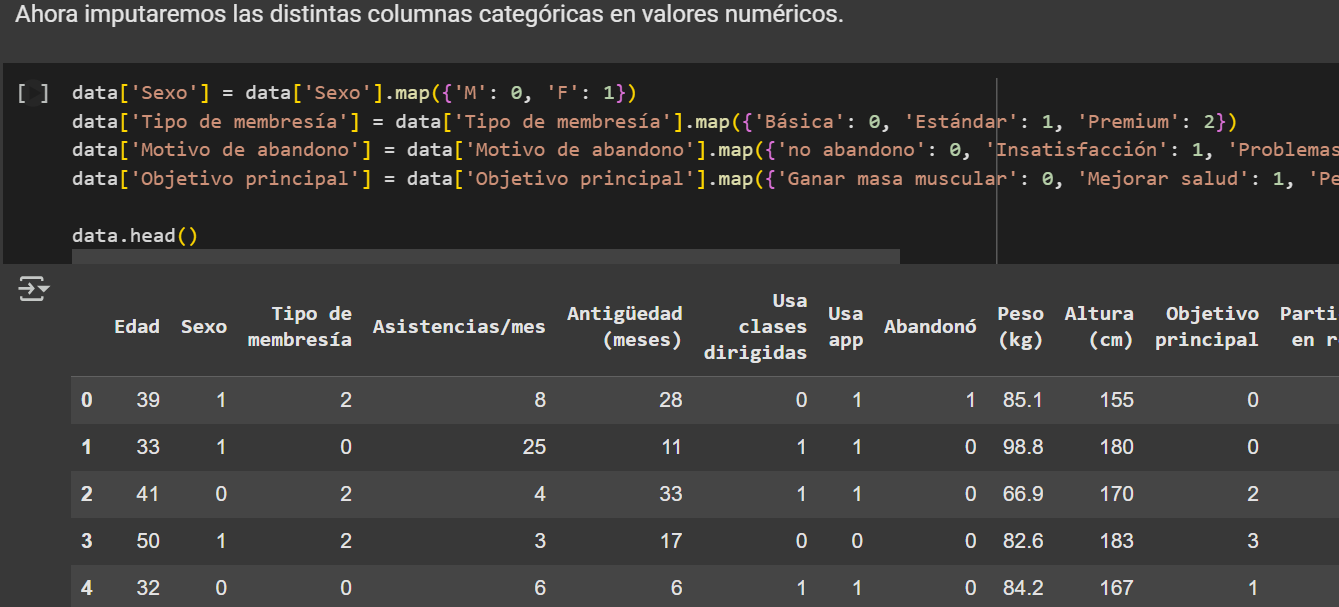
Daniel Marín: Para el campo motivo de abandono se imputa el valor “no abandono” sobre los missing values de dicha columna, ya que representa a las personas que no han abandonado.



Ahora, con la nueva etiqueta de "no abandono" obtenemos la siguiente información:

* El 70% de los clientes no abandona el gimnasio.
* El 30% de los clientes abandona el gimnasio. De los cuales:
  + El 11% abandonan por insatisfacción.
  + El 9% abandonan por problemas económicos.
  + El 5% abandonan por falta de tiempo.
  + El 5% restante abandonan por otro motivo.



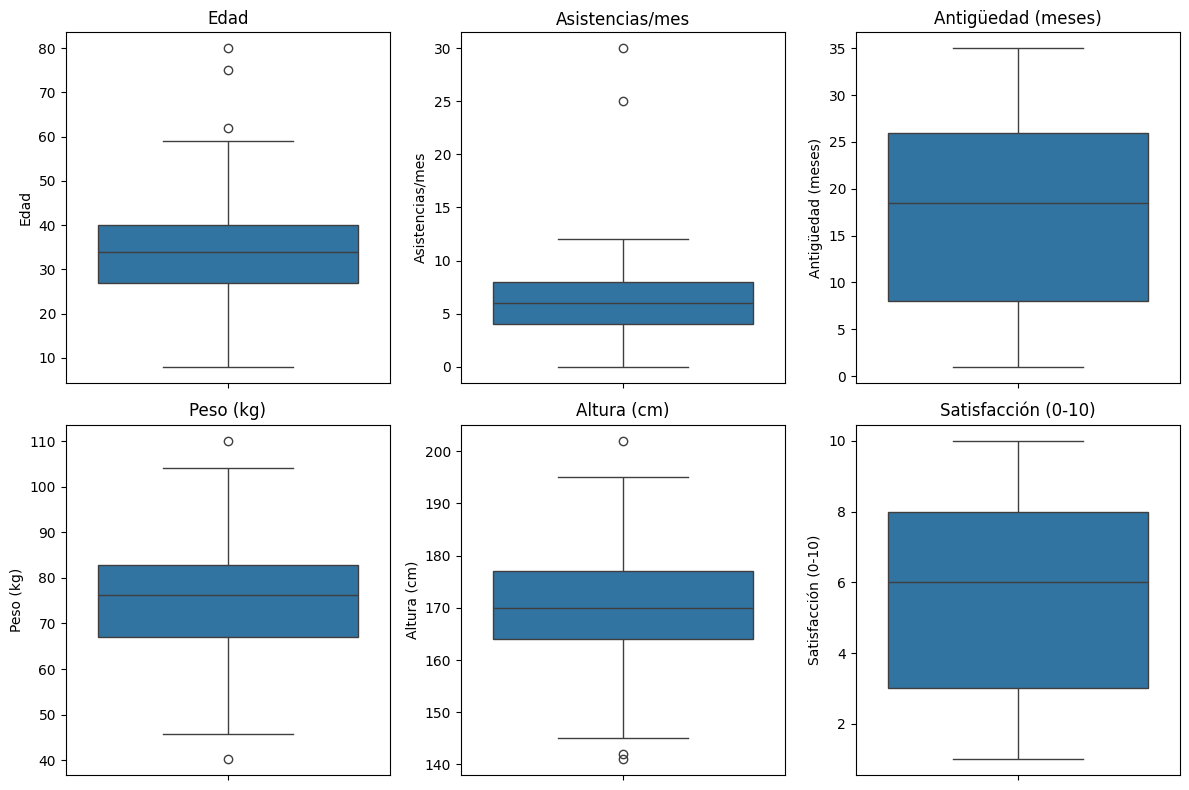


Víctor González: De las 60 personas que han abandonado el gimnasio, 40 no recibieron oferta especial (66,7%) y 20 (33,3%) sí recibieron.



### Outliers

Daniel Marín: Lo siguiente será ver si en nuestros datos tenemos outliers o inliers. En clase ya se encontraron algunos de ellos pero lo correcto sería buscarlos con un método, como Jackknife.



Parece que hemos conseguido identificar algunos outliers:

* Edad: 3 outliers
* Asistencias/mes: 2 outliers
* Peso: 2 outliers
* Altura: 3 outliers

Un total de 10 outliers detectados exitosamente. Tenemos que recordar que un outlier no es lo mismo que un valor influyente. Un valor influyente es aquel outlier que SÍ afecta al modelo de IA.

Tenemos diferentes formas de tratar los outliers:

* Borrarlos
* Imputarlos
* Usar otros métodos más sofisticados

IMPORTANTE: Debatir en clase si se puede escalar los datos, de ser así se puede usar StandardScaler que es resistente contra los outliers.

Si NO se usa escalado, se pueden intentar corregir ya que eliminar datos nos llevan a una pérdida de información.

### ¿Cómo abordar de manera inteligente los outliers?

Como hemos indicado anteriormente, podemos corregir los outliers de dos formas:

* Imputación con valores prototípicos.
* Realizar un análisis más sofisticado.

El primer método consiste en usar valores prototípicos como la media, mediana o moda. Y otra forma es hacer un análisis más profundo para encontrar patrones que nos ayuden a corregir estos outliers.

Por ejemplo: En los datos del Titanic se detectó que existe una relación entre la edad y la clase en la que viajaban los pasajeros de la siguiente forma:

* Edad > 30 ⟹ Primera clase
* 30 > Edad > 20 ⟹ Segunda clase
* Edad < 20 ⟹ Tercera clase

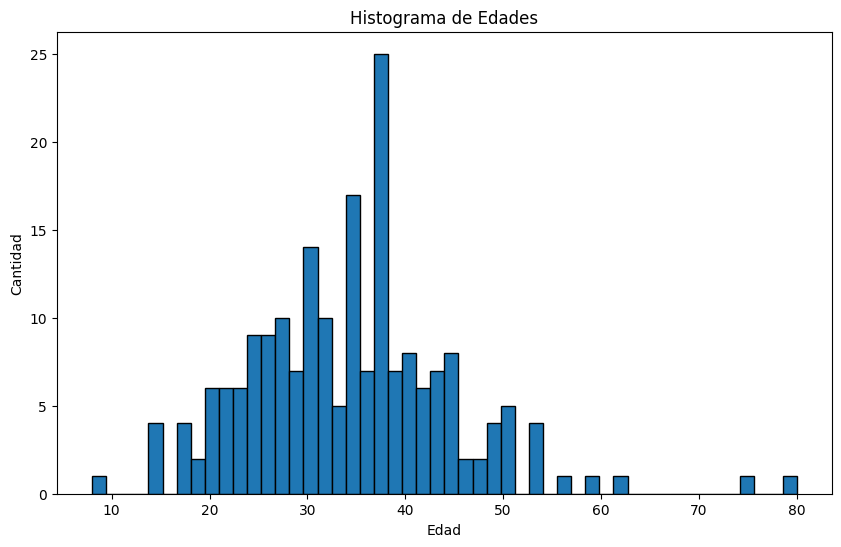
Lo mismo podemos intentar pensar en este caso, buscar qué características pueden estar relacionadas y en base a eso arreglar nuestros outliers.

Para ello podemos usar algunas propuestas de los compañeros.

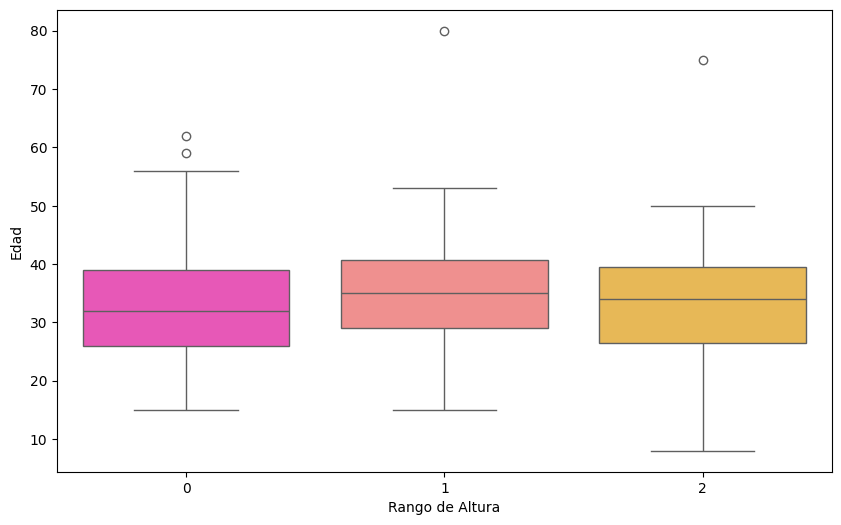
Carlos Carrasquer: Realizar rangos de edades en vez de usar los datos numéricos cómo tal.

Exacto, podemos crear rangos para poder saber donde es probable que X dato tenga este valor u otro en base al rango que pertenece. En este caso podemos hacer un rango de las alturas y representar cajas para ver donde se posicionan las edades. Para mantener que nuestro dataset siga siendo numérico podemos hacer lo siguiente:

* Baja altura ⟹ 0
* Media altura ⟹ 1
* Alta altura ⟹ 2



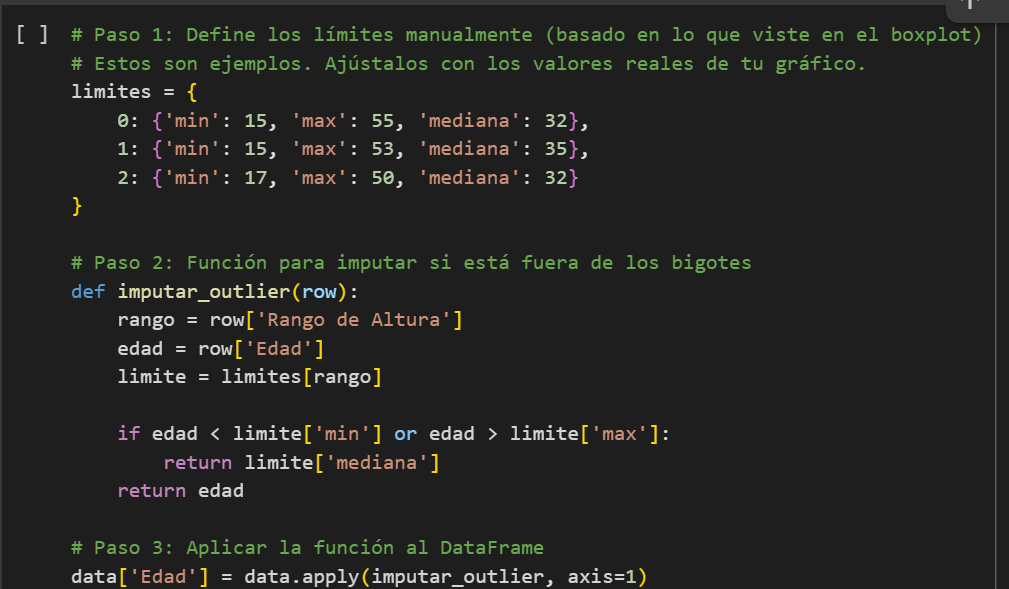
Ahora haremos nuestro gráfico de cajas en base a la Edad y la Altura.

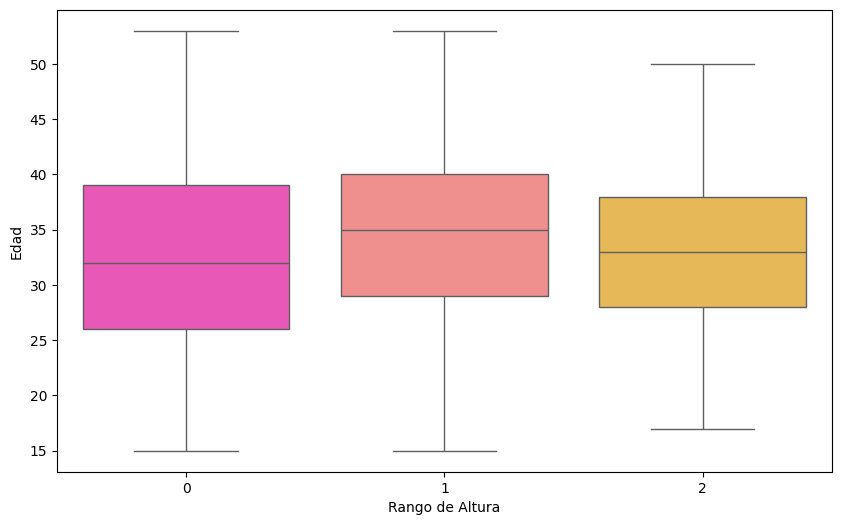


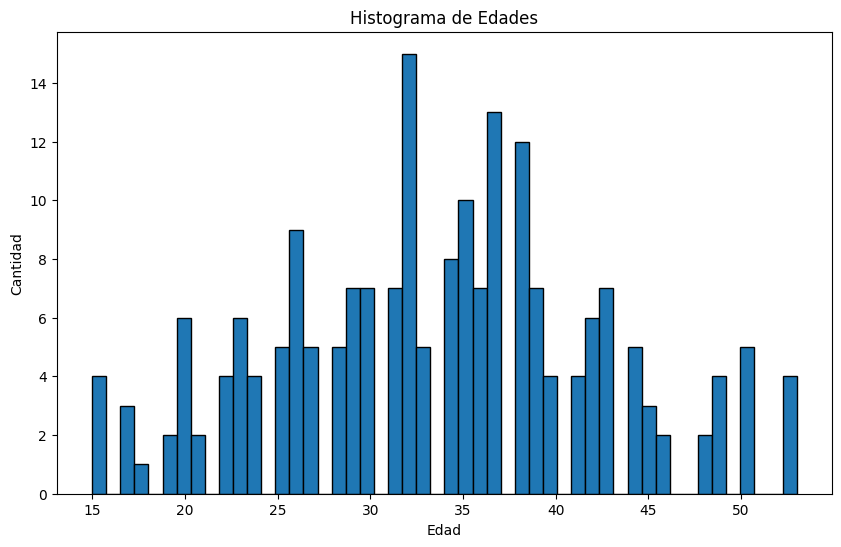
Vemos que con esta gráfica se han detectado nuevos outliers que no aparecieron antes. Además que obtenemos la siguiente información:

* La altura baja ⟹ Tiene una mediana de 32 años.
* La altura media ⟹ Tiene una mediana de 35 años.
* La altura alta ⟹ Tiene una mediana de 32 años.

Con esto, podemos corregir la edad de la siguiente forma:







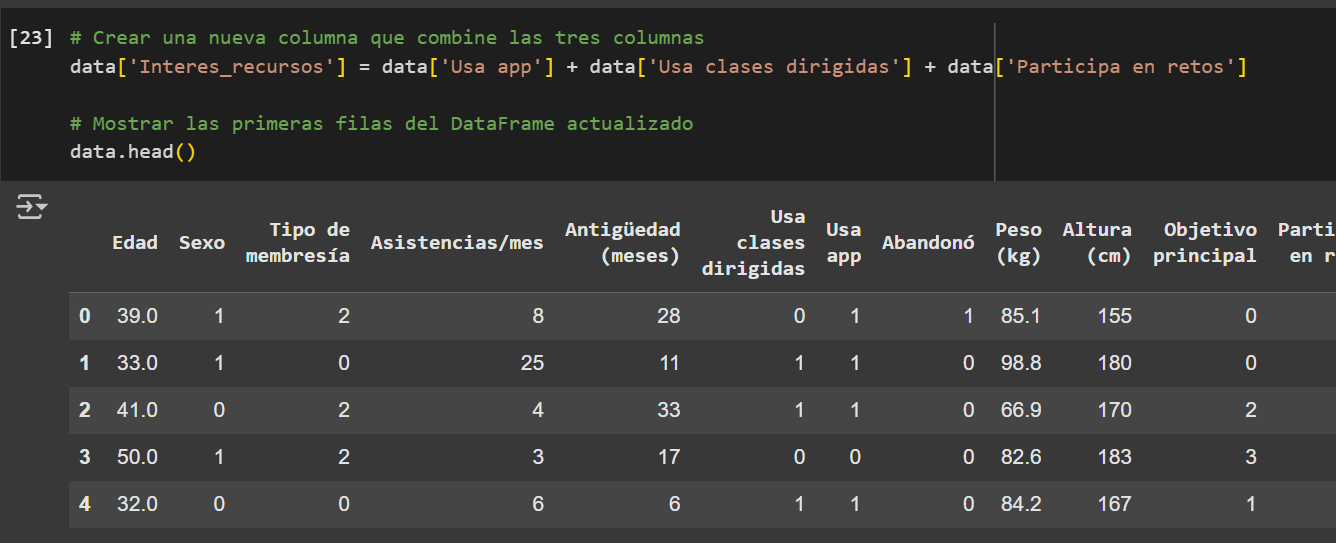
De esta forma podemos solucionar manualmente los outliers presentes. Habría que buscar otras relaciones que nos pudieran dar más outliers. Todo esto se puede comprobar en el colab en los enlaces de interés.

# 3.-Transformación

### Combinar columnas

Combinar las columnas uso de la app, uso de clases dirigidas y participación en retos para comprobar si el usuario tiene interés en la utilización de recursos del gimnasio es más o menos probable que abandone.

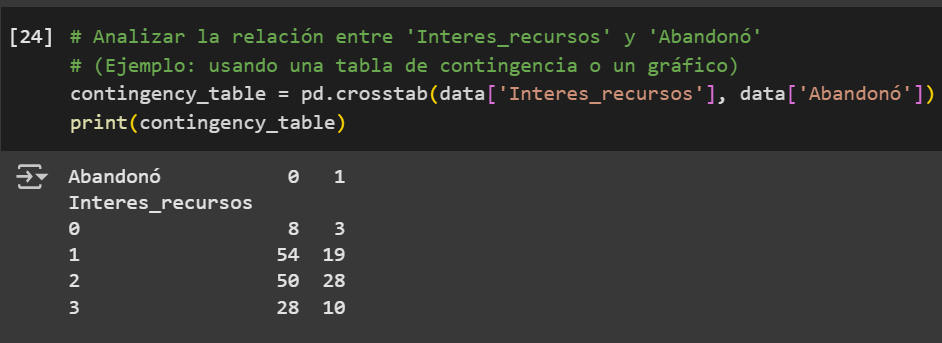
Daniel Marín: Realizamos la combinación en código

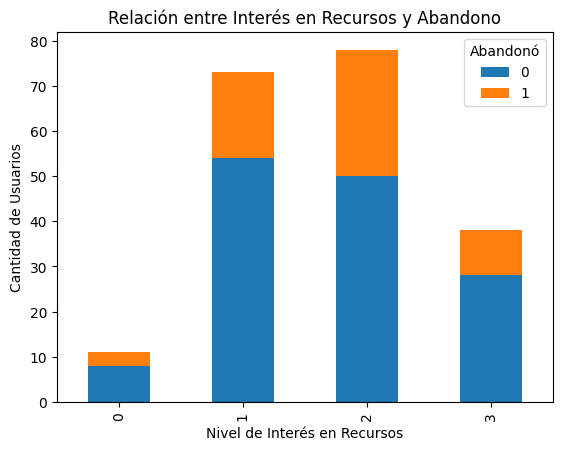


Como las columnas son númericas, hay que indicar el contexto de lo que se está mostrando. La columna Interes\_recursos muestra lo siguiente:

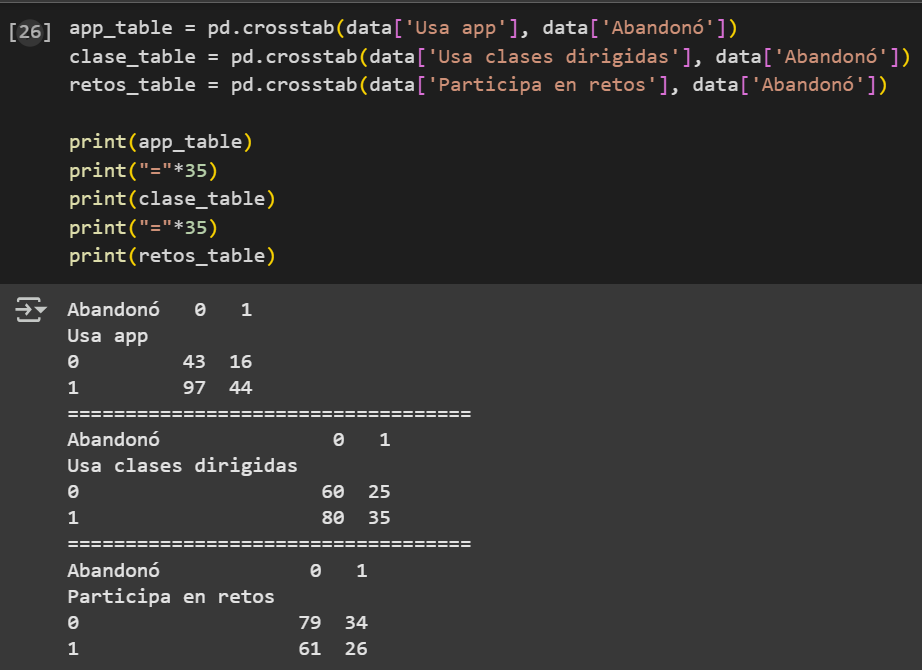
* 0 ⟹ No se usaron recursos
* 1 ⟹ Se usó 1 recurso
* 2 ⟹ Se usaron 2 recursos
* 3 ⟹ Se usaron todos los recursos

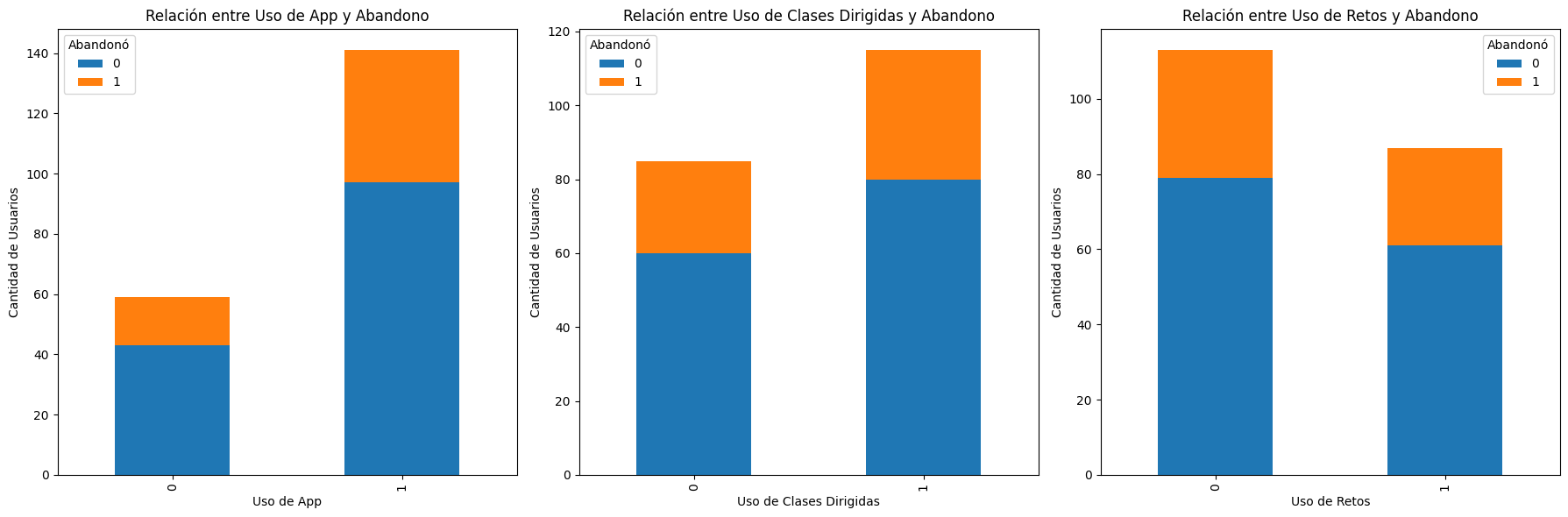
Hacemos una tabla de contigencia para analizar la relación entre el uso de recursos y el abandono.





Podemos observar que la cantidad de clientes que abandonan el gimnasio es más probable que hayan usado al menos un recurso del gimnasio, el problema es que no se sabe que recurso han usado. Podemos hacer un estudio por cada recurso para ver el impacto de cada recurso.

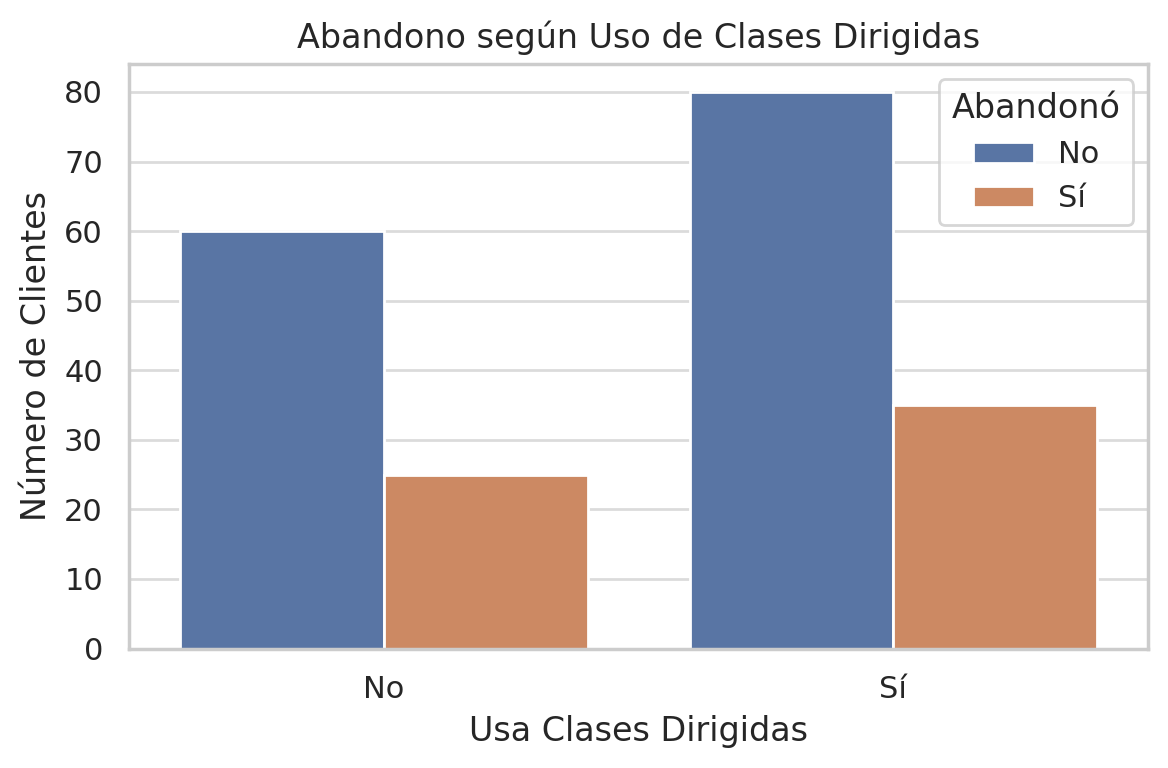
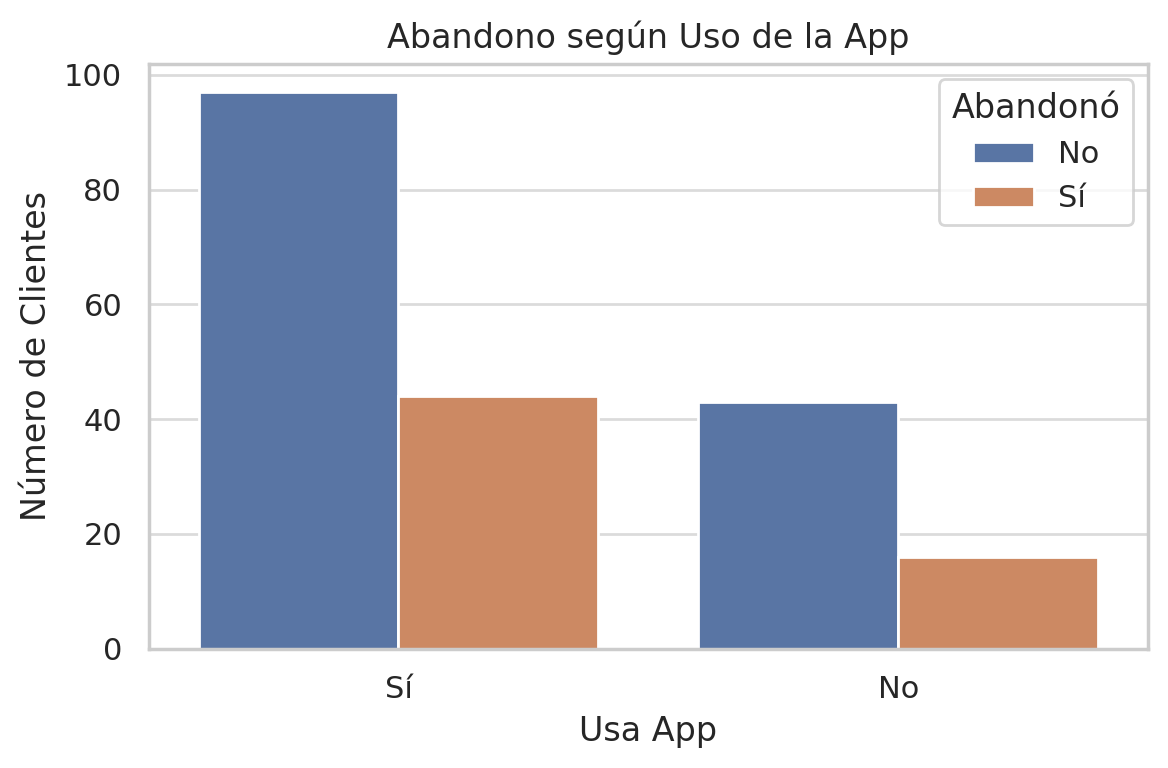




Vemos que de los tres recursos, los que tienen un mayor número de gente que ha abandonado son la App y las clases dirigidas. Aquí hay algunas hipótesis que nos pueden hacer pensar el por qué ocurre esto:

* La relación del uso de la App puede ser por la oferta especial.
* La relación de las clases dirigidas puede ser por los comentarios negativos.

Todo esto se puede comprobar en el colab en los enlaces de interés.



# 4.-Minería de datos

Como modelo de Machine Learning para sacar el porqué la gente abandona el gimnasio se va a optar por usar un árbol de decisión, ya que es un modelo de caja blanca de fácil interpretabilidad basado en reglas que nos ayuda a ver claramente el objetivo que buscamos con los datos recogidos.

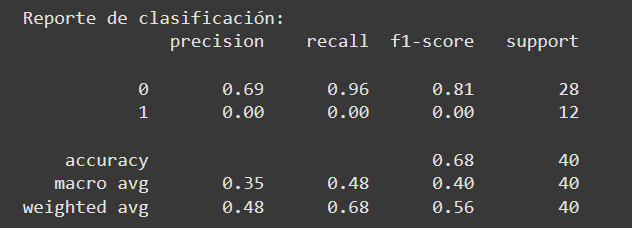
## Creación del árbol:

Se usará un clasificador como árbol de decisión dónde se evalúa el abandono o no abandono de cada usuario del gimnasio, al tener pocos datos es el método más eficiente para obtener el conocimiento y respuestas que estamos usando.

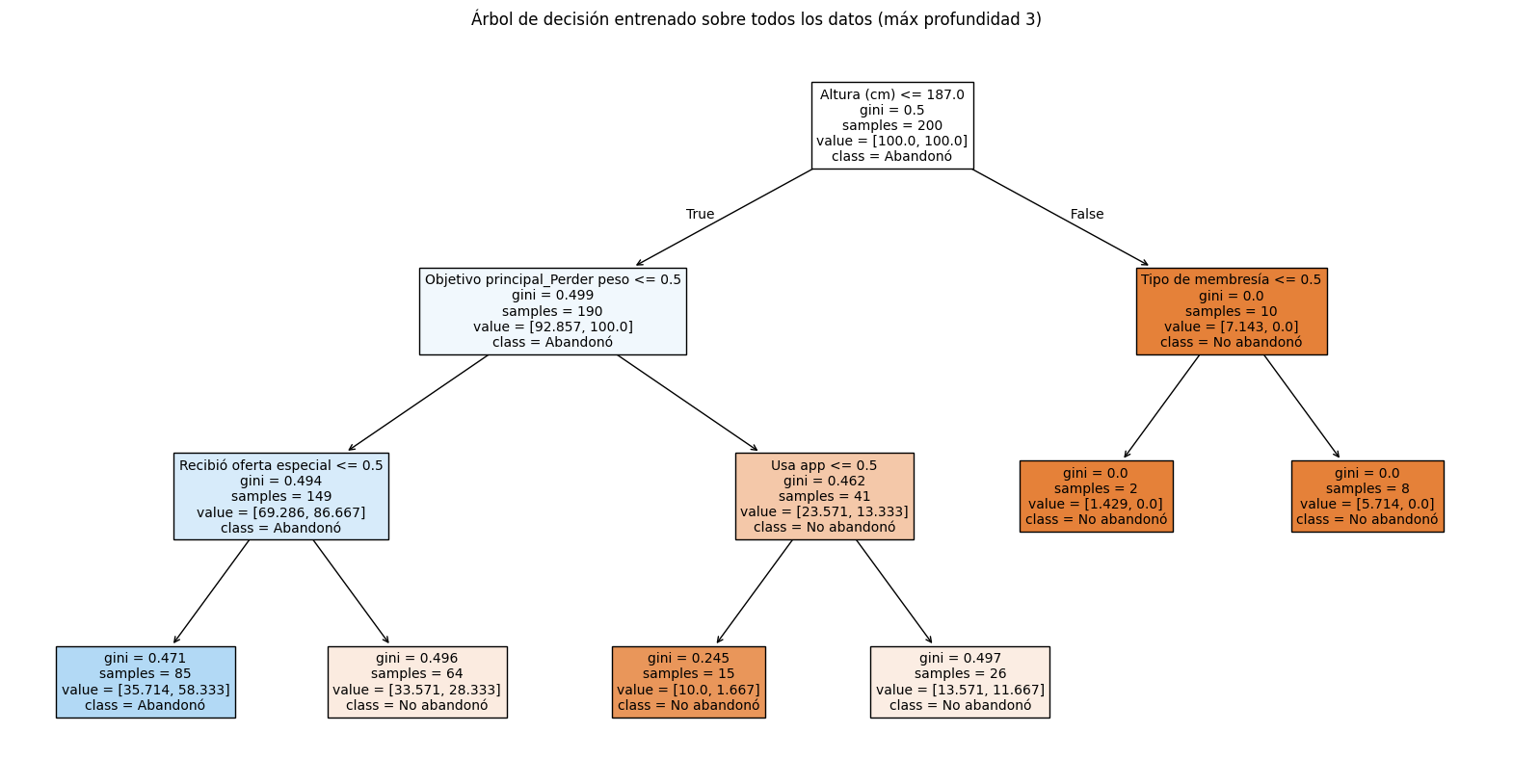
Pablo

Erich ⬇️

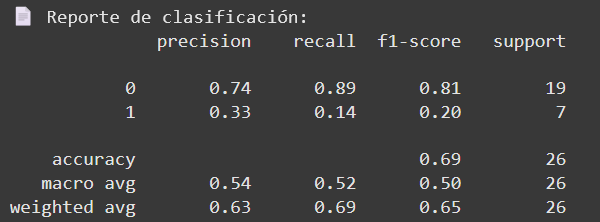
Primero hemos hecho un random forest sin aplicar un SMOTE y se obtuvo el reporte de clasificación:



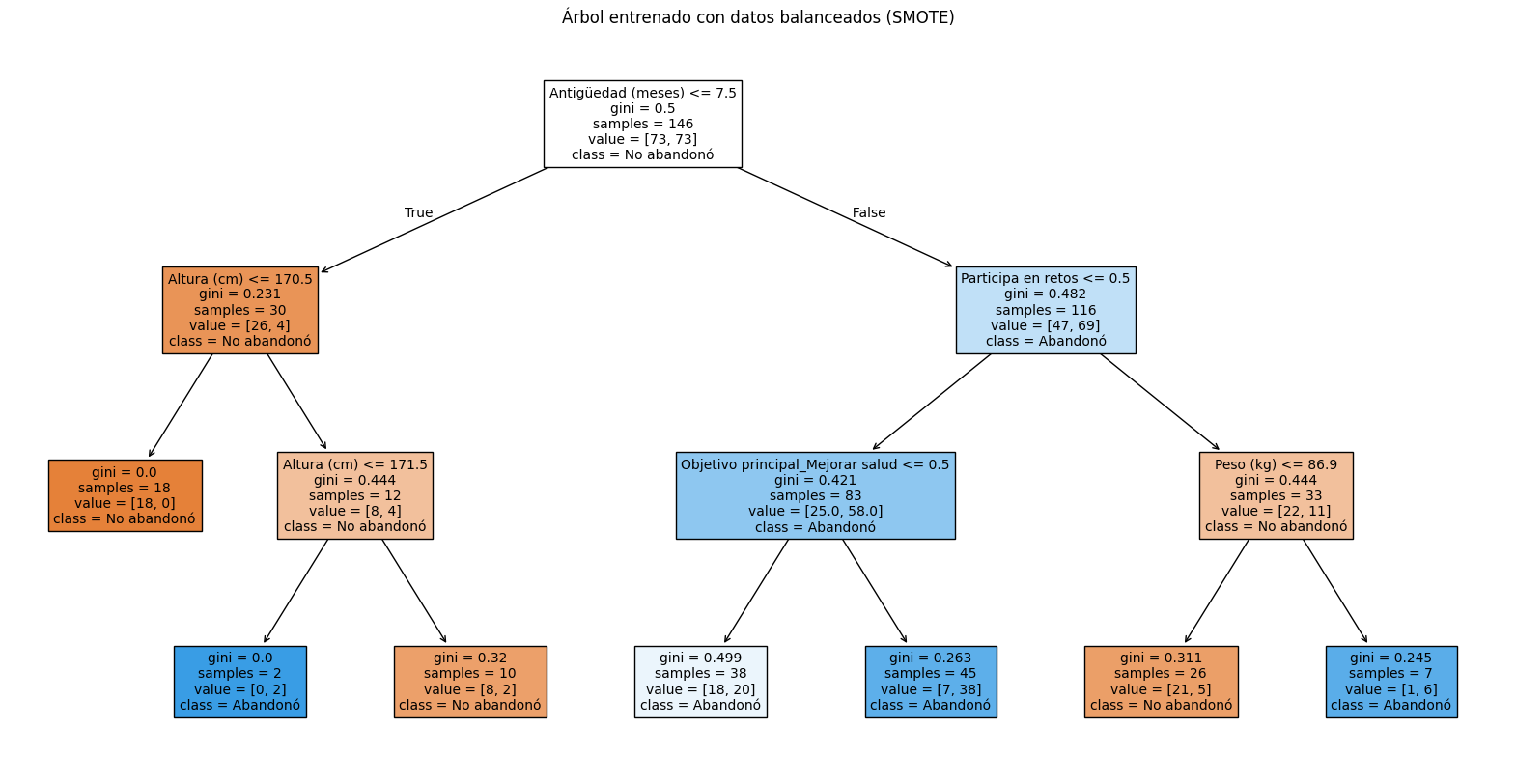
Con el siguiente árbol:



Luego, dado que acierta un cero por ciento de los que abandonan. Se aplicó a los datos de entrenamiento un SMOTE para balancear los datos. Se obtuvo el siguiente reporte de clasificación:



Y el árbol:



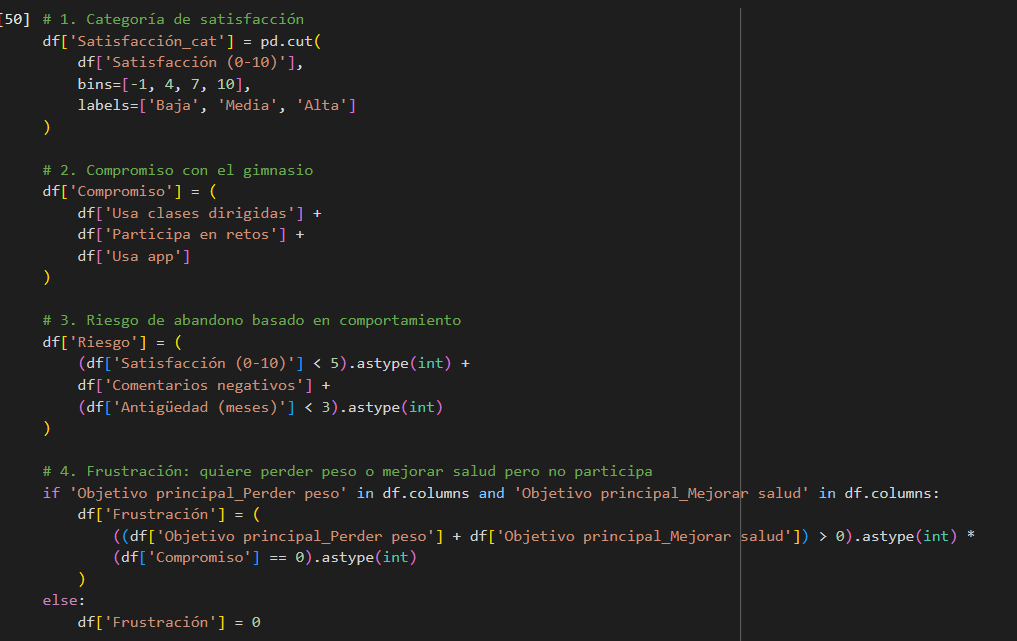
Con estos dos árboles se podía interpretar lo siguiente:

El **árbol con SMOTE es mucho más útil** para interpretación y toma de decisiones.

Las variables que mejor explican abandono en tu modelo actual:

* Antigüedad (meses)
* Participa en retos
* Objetivo principal\_Mejorar salud
* Peso (kg) (como indicador indirecto de frustración o falta de resultados)

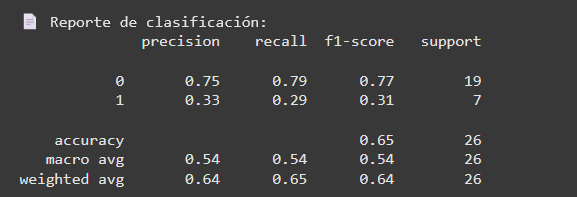
Entonces la siguiente estrategía es crear nuevas variables. Por ejemplo:



Con esto pretendemos mejorar el recall.

Se utilizó SMOTE y XGBoost en este caso, en vez de Random Forest.

Y se obtiene:

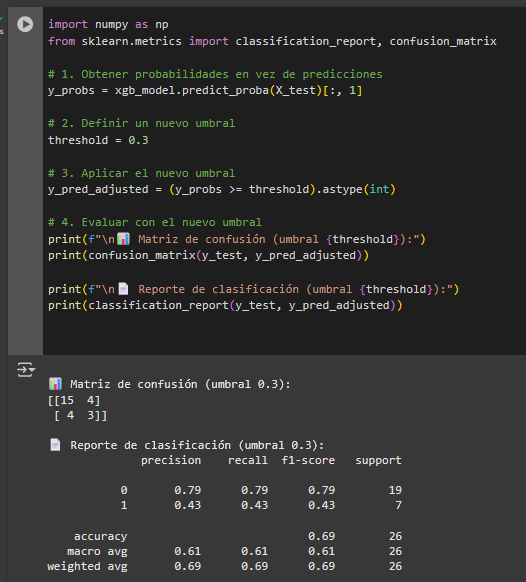


Que indica que mejoramos un poco más. Pero seguimos con el problema de desbalance.

El siguiente paso para intentar mejorar aún más el modelo dentro de todas las dificultades es:

* Ajustar el **threshold (umbral de probabilidad)** con el que se decide si un cliente abandonará o no, para mejorar el rendimiento del modelo en la clase minoritaria ("Abandonó").

El código:



Tras aplicar técnicas de ingeniería de variables, balanceo con SMOTE y ajuste del umbral de decisión, logramos una mejora significativa en la capacidad del modelo para identificar casos de abandono. Sin embargo, los resultados siguen siendo limitados, especialmente en términos de recall y precisión sobre la clase minoritaria.

Esto indica que, a pesar de los esfuerzos de modelado, el conjunto de datos actual **no** contiene suficiente información predictiva o ejemplos representativos de abandono.

Conclusión: La solución más viable en este punto es **ampliar la base** de datos, especialmente aumentando los casos de abandono. Sin una representación adecuada de la clase positiva, cualquier modelo seguirá enfrentando dificultades para generalizar correctamente.

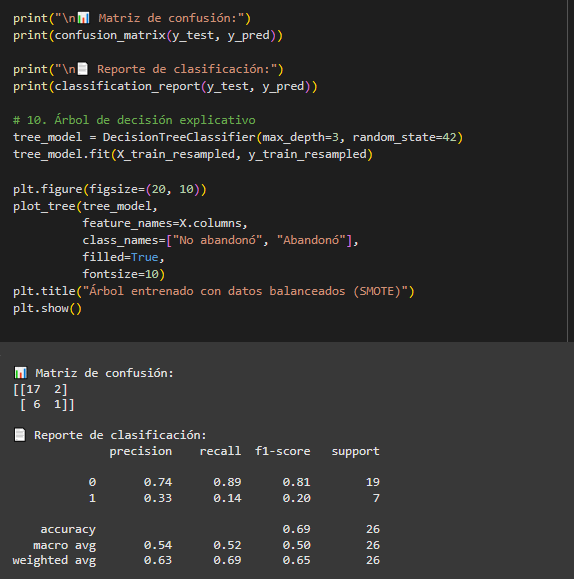
Colab para comprobar en enlaces de interés.

# 5.-Interpretación/Evaluación

Fran R:

Para realizar una interpretación y evaluación de los datos existen diversas métricas, como por ejemplo la matriz de confusión donde separamos entre verdaderos positivos, falsos positivos, falsos negativos y verdaderos negativos, donde, en la captura adjunta, podemos observar que ha clasificado correctamente a 17 personas que no abandonaron y 1 persona que abandonó, mientras que se clasificaron a 2 personas como desertoras que no abandonaron realmente y a 6 personas como no desertoras que realmente abandonaron

Adicionalmente podemos realizar un accuracy score para conocer el porcentaje de acierto de nuestro modelo basándonos en diferentes métricas



Fran S:

Además de la matriz de confusión, el reporte de clasificación muestra métricas clave como precisión, recall y f1-score. El modelo identifica mejor a quienes no abandonan (recall de 0.89) que a quienes sí lo hacen (recall de 0.60). Esto indica un posible sesgo hacia la clase mayoritaria. También se presentan promedios macro y ponderado que ofrecen una visión global del rendimiento. Estas métricas ayudan a evaluar si el modelo es equilibrado y efectivo.

6.-Enlaces de Interés

Daniel Marín: [**Enlace al cuaderno**](https://colab.research.google.com/drive/1aNw8FMpEwCznjulIuKUc8XJ-1bf7bNqe#scrollTo=gd7AMR2Rxta2)

Erich: [**Enlace al cuaderno**](https://colab.research.google.com/drive/1MY1tUYsHQSACzrQQGGBXH3nRxXsjghDw?usp=sharing)