Minería de datos sobre ”Video Game Sales with Ratings”

Nicolás Larrañaga Cifuentes[1] - Daniel Augusto Cáceres Salas[2]

***Abstract*— Basándonos en el dataset *Video Game Sales with Ratings* el cual ofrece información sobre las ventas de videojuegos en diferentes continentes junto a otra información relevante como lo es el rating, el desarrollador, el distribuidor, el genero, etc; se desea obtener información relevante sobre la correlación entre los meta-datos del juego y su éxito comercial en los continentes respectivos, con el objetivo de realizar procesos de clustering y de predicción mediante árboles de decisión.**

***Index Terms*— KDD, K-bins, Videojuegos, Ventas, K-means, K-modes, Apriori, FP-Growth, Decision Tree.**

I. Introducción

Actualmente la industria de los videojuegos se ha apropiado de un amplio porcentaje de del mercado mundial, únicamente en norteamérica el valor del mismo ronda los 18.4 mil millones de dólares. Enmarcado en el contexto de la fuerte competencia generada por las grandes casas diseñadoras de videojuegos, ha surgido una nueva tendencia la cual consiste en apuntar a grupos de nicho basados en la locación del público y el género del videojuego, de esta manera se busca crear una base de clientes frecuentes acostumbrados a adquirir los juegos de un género específico que sean producidos (o en algunos casos distribuidos) por alguna firma en específico.

Este set de datos posee la información de ventas, género, plataforma, calificación (crítica y usuarios) y rating ESRB de videojuegos de los últimos 30 años repartidos en un total de 16719 registros y 16 variables. Debido a la gran cantidad de videojuegos en la actualidad, estos datos se limitan a los juegos desarrollados por compañías reconocidas a nivel mundial, excluyendo los denominados videojuegos independientes o ”indie games” los cuales son creados por individuos o pequeños grupos, sin apoyo financiero de distribuidores.

La información mencionada es de acceso público y puede ser encontrada en el siguiente enlace `https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings/data`

La idea de este proyecto es crear un modelo predictivo que permita a las empresas desarrolladoras de videojuegos obtener un estimado de ventas de un producto nuevo en las diferentes regiones (Japón, Norteamérica, Europa) con base al género, rating y recepción crítica del mismo. Los conjuntos de datos (original y procesado) junto a los scripts utilizados para el tratamiento pueden ser encontrados en el siguiente repositorio de github `https://github.com/larranaga/UNAL-data-mining`.

[1] Estudiante de Ingeniería en Sistemas y Computación, Universidad Nacional De Colombia, Bogotá, Colombia `nlarranagac@unal.edu.co`

[2] Estudiante de Ingeniería en Sistemas y Computación, Universidad Nacional De Colombia, Bogotá, Colombia `daacaceressa@unal.edu.co`

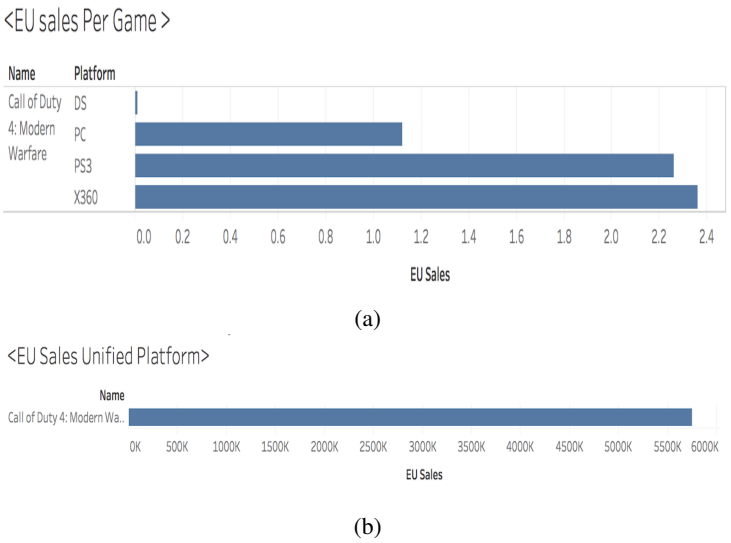


Fig. 1: (a) ventas distribuidas por plataformas (b) ventas de todas las plataformas unificadas

II. Descripción de las Variables

Las variables presentes en el set de datos original se encuentran en la tabla I

III. Preprocesamiento

Para el preprocesamiento de este set de datos realizamos los siguientes 8 pasos.

1) **Manejo de valores perdidos**
   El proceso inicia mediante la eliminación de los registros que tengan en la variable ‘Name’ un valor nulo; en la variable ‘User Score’ un valor tbd (to be determined); en las variables ‘Publisher’ y ‘Developer’ valores nulos. Ya que estos registros contienen demasiada información perdida como para ser utilizados.

   Una vez finalizada esta eliminación de registros se procede a insertar información en los registros cuyas variables ‘Rating’ , ‘Developer’ o ‘Publisher y ‘Year of release’ son nulas. Para ‘Developer’ y

TABLE I: Descripción de variables

Variable	Descripción	Tipo	Categoría
Name	Nombre del Juego	String	Nominal
Platform	Consola en la que se juega.	String	Nominal
Year of Release	Año de lanzamiento.	Numérico Discreto	Ordinal
Genre	Género (Acción, Deportes, RPG, Shooter, etc)	String	Nominal
Publisher	Firma encargada de distribuir el juego	String	Nominal
NA sales	Número de ventas en Norteamérica (millones de unidades)	Numérico Discreta	Radio
EU sales	Número de ventas en Europa (millones de unidades)	Numérico Discreta	Radio
Other Sales	Número de ventas en el resto del mundo (millones de unidades)	Numérico Discreta	Radio
Global Sales	Número de ventas a nivel mundial (millones de unidades)	Numérico Discreta	Radio
Critic Score	Promedio del puntaje dado por los críticos de la página Metacritic	Numérico Discreta	Radio
Critic Count	Cantidad de críticos que evaluaron y dieron un puntaje al videojuego	Numérico Discreta	Intervalo
User Score	Promedio del puntaje dado por los suscriptores de la página Metacritic	Numérico Discreta	Radio
User Count	Cantidad de usuarios que evaluaron y dieron un puntaje al videojuego	Numérico Discreta	Intervalo
Developer	Firma responsable de la creación del videojuego	String	Nominal
Rating	Clasificación dictada por el ESRB	String	Ordinal

'Publisher' se inserta el valor no nulo en la otra variable. Para 'Rating' se inserta el valor de la moda del developer en este atributo. Para 'Year of release' se inserta la media de este atributo, esta solución es factible debido a la naturaleza de los datos (los videojuegos llevan un tiempo relativamente corto en haber sido comercializados).

2) **Eliminación de ruido**
Debido al significado de las variables 'User Score' y 'Critic Score' se pueden encontrar registros sin significado, el valor dado por los usuarios debe ser entre 0 y 10, mientras que el valor dado por la crítica debe estar en 0 y 100. Por lo tanto cualquier registro que contenga un valor fuera de estos rangos en su respectiva variable debe ser eliminado.

3) **Normalización**
La variable 'Critic Score' es normalizada dividiendo entre 10 su valor para que se encuentre en el mismo rango que la variable 'User Score'. Esto se debe a que ambos atributos están muy relacionados por lo que es conveniente que se encuentren en el mismo rango.

4) **Muestreo**
El primer cambio sustancial en la información viene al comprimir la cantidad de registros mediante la dimensión 'Platform' al combinar los atributos de los registros del mismo juego en diferentes plataformas, es decir, diferentes registros de un juego se van a convertir en un único registro.

Como se puede evidenciar en la figura 1, los 4 registros de diferentes plataformas se convierten en uno solo. Las variables 'Name', 'Year of Release', 'Genre', 'Publisher', 'Developer' y 'Rating' son comunes en todos los registros del mismo juego por lo que en el registro unificado se mantienen iguales.

Las variables 'Sales', 'Critic Count' y 'User Count' se suman ya que este sería el valor total de las ventas por zona del juego.
Por último, las variables 'Critic Score' y 'User Score' se promedian con respecto a la cantidad de registros que tienen valores diferentes a 0.

5) **Reducción de dimensionalidad**
Al combinar los diferentes registros podemos eliminar el atributo 'Platform' sin ningún problema.

6) **Manejo de valores perdidos**
Este paso es realizado una vez más ya que aún se encuentran muchos valores nulos en las dimensiones 'User Score' y 'Critic Score'. De los alrededor de 10.000 registros que se tienen hasta el momento, alrededor de la mitad poseen valores nulos en ambas variables. La solución es hallar la media de estos atributos según el developer y asignarla a estos registros. Esto se realiza bajo la suposición que un developer exitoso va a seguir produciendo juegos de la misma calidad, mientras que uno de bajo desempeño se mantendrá en esta categoría.

Aún después de esta gran suposición más de 1.000 datos siguen teniendo valores nulos en las variables 'User Score' y 'Critic Score', esto se debe a que muchos juegos fueron desarrollados por un developer que no posee más juegos registrados en este set de datos.

7) **Discretización**
Debido a la importancia de valores discretos para algunos algoritmos de asociación y clasificación se optó por realizar una discretización en las variables de tipo 'Sales' (NA, EU, JP y Other). Esta discretización fue realizada con la herramienta Tableau utilizando

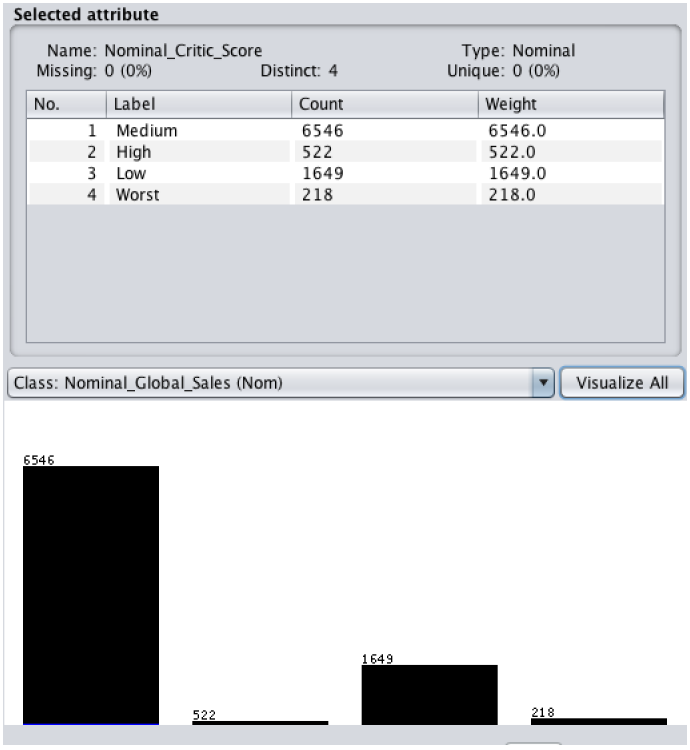
la técnica de K-BINs con una cantidad diferente de bins recomendada por Tableau para cada una de las variables.

Finalmente la variable 'Global Sales' fue re-calculada sumando los nuevos valores de las dimensiones de ventas.

IV. Asociación

Para la generación de las reglas de asociación se utilizó WEKA, para este proceso se realizaron algunos pasos de pre procesamiento adicionales a los datos, específicamente se categorizaron los datos numéricos a partir de una división en cuartiles; para ejemplificar la variable **critic-score** es de tipo numérico cuyo rango varia entre 0 y 10, sin embargo después del procesamiento esta variable se clasificó en una de 4 categorías como se puede ver en la figura 2. Una

Fig. 2: **critic-score** como valor nominal



vez creadas las variables nominales que representaran a las numéricas, se procedió a aplicar el algoritmo Apriori para generar las reglas de asociación. En un principio las reglas generadas no eran muy dicientes pues eran del estilo

$$\begin{aligned} &eu\_sales = high, \\ &jp\_sales = high, \\ &na\_sales = high \Rightarrow global\_sales = high \end{aligned} \tag{1}$$

Este tipo de implicaciones representaban en 80% de las reglas creadas, las cuales no aportaban ningún tipo de información nueva pues es evidente que si las ventas en todos los continentes son altas las ventas globales también lo serán; para solucionar esto se decidió omitir todas las dimensiones de ventas a excepción de la global, una vez hecho esto se obtuvieron las reglas que se ven en la tabla II, estas mismas reglas también fueron obtenidas por el algoritmo FPGrowth.

TABLE II: Reglas obtenidas mediante Apriori y FPGrowth

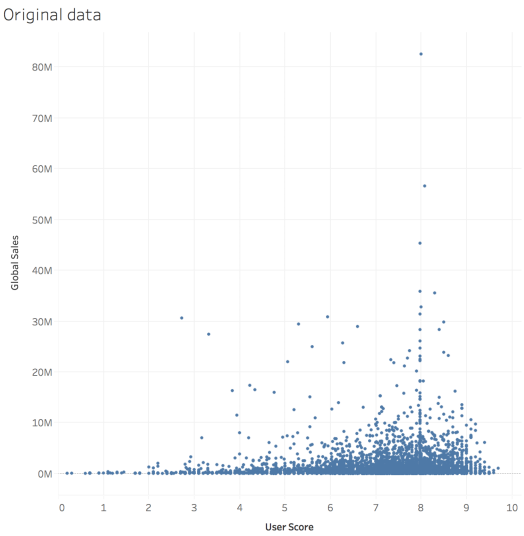
Antecedente	**Consecuente**	**Confianza**	**Soporte**
Genre = Role Playing	User_score = Medium	0.083	0.3
Rating = E	User_score = Medium	0.8	0.2
Rating = T	User_score = Medium	0.79	0.2
Genre = Action	User_score = Medium	0.79	0.1

De estas reglas obtenemos un primer acercamiento a la idea de que un juego que abarque un mayor rango de edades suele tener una mayor aceptación por el público, esto se denota mediante los ratings **E** y **T** que son respectivamente *Everyone* y *Teens*.

V. Clustering

Para el proceso de Clustering se utilizó el algoritmo de K-means sobre los datos originales (ver figura 3), se utilizó como variables iniciales las ventas globales , la calificación de usuario y de crítica, sin embargo esta aproximación generaba demasiado ruido entre los clusters como se puede ver en la figura 4.

Fig. 3: Datos originales



Tomando en cuenta lo anterior se decidió usar el **user score** para generar los clusters, se hicieron intentos usando un **k** variante entre 2 y 100, sin embargo cuando el número de clusters era muy bajo (véase figura 5) no se obtenía información alguna de los mismo, y adicionalmente los valores de cohesión y separación eran muy bajos como para ser considerados.

Fig. 4: Clustering utilizando user score y critic score

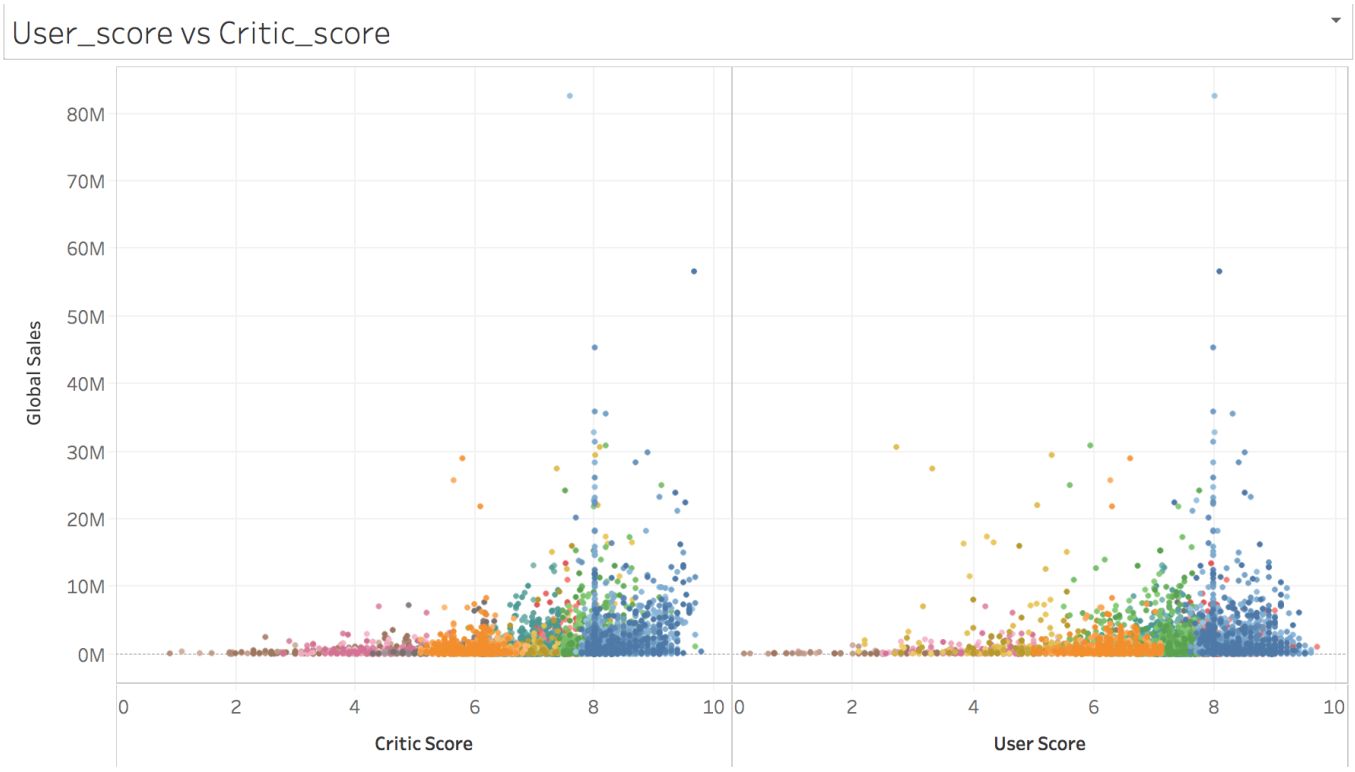


Fig. 5: Clustering usando user score (2 clusters)

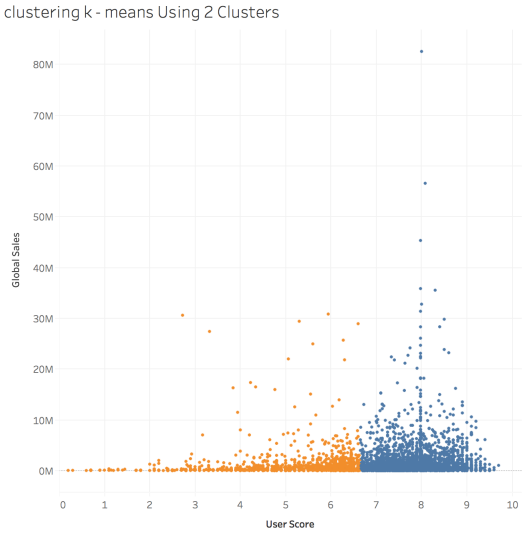
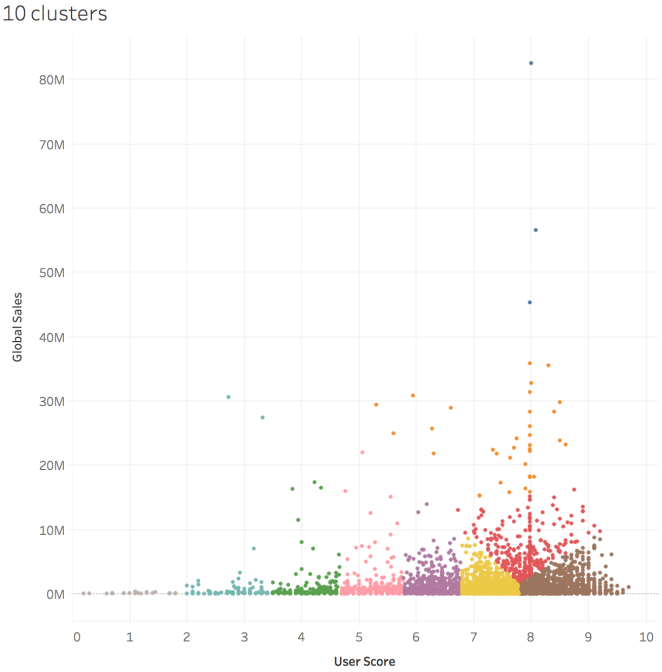


Fig. 6: Clustering usando user score (10 clusters)



Finalmente al considerar 10 clusters se obtuvo un coeficiente de cohesión de **81.026** y de separación de **63.099**, estos coeficientes se validaron mediante el criterio Calinski - Harabasz, obteniendo la figura 6

De este agrupamiento pudimos obtener información relevante. Para empezar los juegos que registran el mayor número de ventas (mostrados en rojo en la gráfica 6) tienen en común que están bajo el Rating **E**, lo cual reafirma las observaciones hechas en la etapa de asociación. Adicionalmente se registra un pico de user score para el cluster café, el

cual tiene como característica común el Rating **T**, aportando evidencia a nuestra hipótesis expuesta.

VI. Predicción

El aprendizaje basado en árboles de decisión es un método comúnmente utilizado en la minería de datos, este tiene como objetivo crear un modelo que predice el valor de una variable de destino en función de diversas variables de entrada. Cada hoja representa un valor de la variable de destino dados los valores de las variables de entrada representados por el camino desde la raíz a la hoja. Se supone que todas las variables tienen dominios discretos finitos, y existe una sola característica de destino llamada la clasificación.

Entre las ventajas de los arboles de decisión se encuentra que es capaz de manejar tanto datos numéricos como categorizados, además, utiliza un modelo de caja blanca donde se puede extraer el conocimiento al ver el árbol.

Para el proceso de predicción se realizó la construcción de un árbol de decisión utilizando las categorías nominales de Critic_Score, User_Score, Rating, Genre, Developer, Publisher y Year_of_Release.

A. Construcción

Se utilizó una medida de impureza de Gini la cual utiliza la siguiente formula:

$$I = \sum_{k=1}^{K} p_k(1 - p_k)$$

Donde p_k indica la probabilidad de la clase k en el nodo que esta siendo calculada la impureza. Esta medida se calcula en cada división para cada categoría y se escoge la división que tenga menor impureza. Esto resulta en una altura máxima del árbol igual a la cantidad de categorías en el dataset, sin embargo, esto es propenso a llevar al overfitting.

B. Poda

Para evitar el overfitting se realizó una poda una vez se tenía una impureza de 0 en todos las hojas del árbol. La poda consiste en utilizar otra medida de impureza llamada Misclassification para medir el nivel impureza que se obtiene al remover todas las hojas de un nodo, si este valor junto con unas variables parametrizadas es menor o igual al nivel de impureza junto con las mismas variables parametrizadas en el árbol actual se realiza la poda sobre todas las hojas. La formula que se utiliza es la siguiente:

$$C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m + \alpha\,|T|$$

Donde T es el set de hojas en el árbol, N_m es la cantidad de hijos del nodo m, Q_m es el valor de impureza del nodo bajo la formula de Misclassification y por último α es un valor parametrizado. La idea de la poda es balancear entre la cantidad de nodos y el nivel de impureza en las hojas utilizando α.

C. Validación

Otra técnica para evitar el overfitting es realizar cross validation con el data set, utilizando una parte de los datos como conjunto de validación y el resto de los datos para crear el árbol de decisión. En este caso se escogió el árbol que obtuvo menor valor de éxito sobre su respectivo conjunto de validación, esto para evitar al máximo el overfitting. El árbol escogido como solución tiene un porcentaje de éxito sobre su conjunto de validación del 80%.

La figura 7 es una parte del árbol de decisión ya que el resultado final tiene alrededor de 125 hojas, sin embargo, con este fragmento del árbol es posible realizar algunas predicciones. Para un árbol de decisión mas completo mirar el anexo 1 de este documento. Utilizando el repositorio que se encuentra en Github es posible realizar predicciones sin ningún tipo de restricción.

VII. Conclusiones

El conjunto de datos original presentaba problemas con el número de datos faltantes, sin embargo estos valores pudieron ser calculados a costa de perder 6000 datos de 16000. Adicionalmente el análisis de reducción de dimensionalidad permitió ver que las variables tienen una colinealidad alta entre sí, permitiendo la reducción de 16 originales a 4, lo cual da una ventaja para la elaboración de un modelo predictivo.

A partir del uso de lo métodos de asociación y agrupación, parece existir evidencia de que abarcar un rango de edades más amplio (mediante la clasificación ESRB) da más campo a que el juego tenga un mayor número de ventas.

La gran ventaja de los arboles de decisión es que permiten extraer el conocimiento y no son simples cajas negras. Esto permite llegar a conclusiones como que el User_Score y el Critic_Score son los factores mas decisivos a la hora de predecir las ventas en un juego. Tambien se puede concluir que los desarrolladores mas conocidos son los únicos que poseen una gran cantidad de venta de unidades, esto puede deberse a la cantidad de marketing que se invierte en los productos.

Fig. 7: Fragmento del árbol de decisión