

# Clasificación usando Árboles y Reglas de Decisión

Jorge Gallego

Facultad de Economía, Universidad del Rosario

febrero 27 de 2017

# Introducción

- Supongamos que ustedes tienen que decidir qué carro comprar, entre varias opciones
- Deben intentar predecir su nivel de felicidad ante cada opción
- Una heurística típica en este tipo de decisiones es hacer un listado de ventajas y desventajas
- E ir descartando en función de ciertas reglas simples
- Por ejemplo, “si el carro es sedan, descartado”
- “Si vale más de 50 millones, descartado”

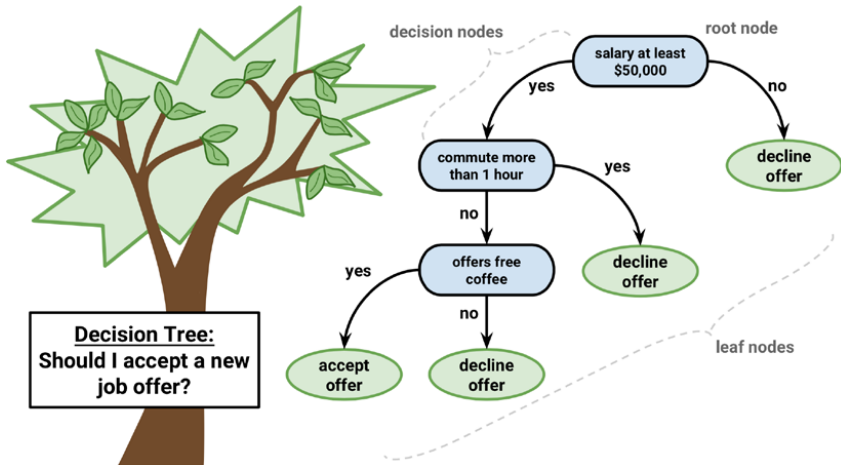
# Introducción

- Así, una decisión compleja se va simplificando
- Esta es la idea de los árboles de decisión y los *rule learners*
- Tomar decisiones complejas a partir de reglas simples
- Tras aplicar el método, los resultados se presentan como estructuras lógicas que no requieren formas estadísticas
- Esto hace que el algoritmo sea muy útil en diversos campos

# Árboles de Decisión

- Los árboles de decisión de aprendizaje sirven para hacer clasificaciones
- Utilizan una estructura de árbol para modelar la relación entre características y *outcomes* potenciales
- Arrancamos con un tronco que incluye todos los casos
- Y vamos creando ramificaciones en función de las características de interés
- Hasta que los objetos terminan clasificados en una clase pronosticada final

# Ejemplo Árbol de Decisión



# Árboles de Decisión

- En el ejemplo los objetos a clasificar son ofertas de trabajo
- La predicción es la clase a la que pertenece cada oferta
- Donde los niveles son si se acepta o no el trabajo
- La primera rama la define el salario. La segunda el tiempo de desplazamiento. La tercera si ofrece café
- El árbol tiene nodo raíz, nodos de decisión y nodos de hojas

# Árboles de Decisión

- Una ventaja de los árboles es que el formato no es exclusivo del método
- También se puede presentar de una forma amigable
- Lo cual es útil cuando se tiene que justificar una decisión
- Por ejemplo, cuando se le niega un crédito a alguien
- En términos legales o corporativos esto es muy útil

# Árboles de Decisión

De esta forma, algunas de las aplicaciones incluyen:

1. Modelos de *score crediticio*. Los criterios para rechazar a alguien deben ser documentados y libres de sesgo
2. Estudios de marketing de comportamiento del consumidor para medir satisfacción.
3. Diagnóstico de condiciones médicas basadas en tomas de laboratorio, síntomas o tasa de progresión de una enfermedad

Existen muchas otras aplicaciones, este es uno de los métodos más populares



# Árboles de Decisión

- Los árboles de decisión se basan en el principio de **partición**
- También se le llama *divide y triunfarás*
- Se dividen los datos en subconjuntos, y estos en subconjuntos más pequeños
- Así hasta que el algoritmo determina que los datos en cada subconjunto son suficientemente homogéneos

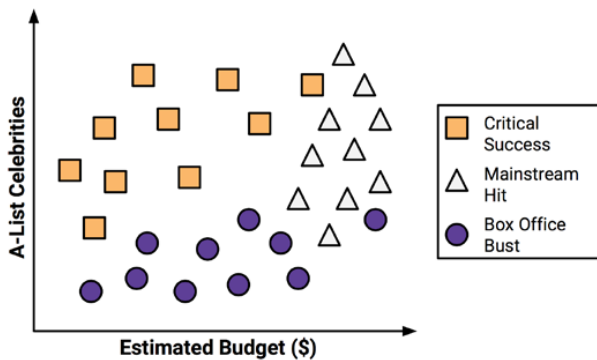
# Árboles de Decisión

- Veamos el proceso con un ejemplo: clasificación de películas
- Ud. trabaja para un estudio de Hollywood y debe decidir si el estudio debe producir un guión
- Tiene un montón de guiones sobre los cuales decidir pero no tiene tiempo para leerlos
- Decide desarrollar un árbol de decisión para predecir
- Debe predecir si cada guión cae en una de tres categorías: éxito para la crítica, éxito para el público o fracaso de taquilla

# Árboles de Decisión

- Para construir al árbol ud. examina los factores que llevaron al éxito o fracaso las últimas 30 películas
- Descubre la relación entre el presupuesto de la película, el número de actores estrella y el éxito
- De hecho ud. encuentra el siguiente *scatterplot* relacionando estas variables

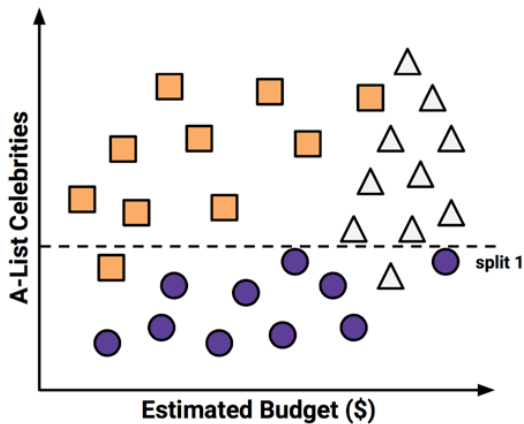
# Ejemplo Árbol de Decisión



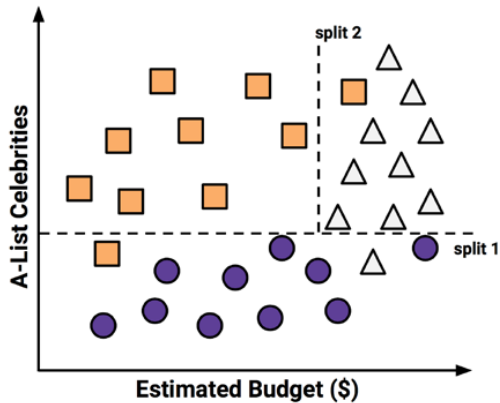
# Árboles de Decisión

- El patrón es claro: el número de actores estrella determina si la película será un fracaso de taquilla o no
- Después, dependiendo del presupuesto es un éxito para la crítica o un éxito de taquilla
- Para crear el nodo raíz, se dividen las películas entre las que tienen muchas o pocas celebridades
- Luego, entre las que tienen alto número de celebridades, dividimos entre las que gastan mucho y las que gastan poco

## Ejemplo Árbol de Decisión



# Ejemplo Árbol de Decisión

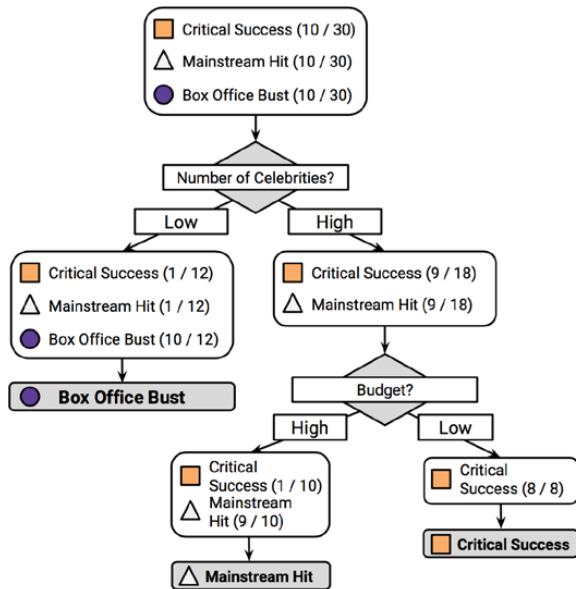


# Árboles de Decisión

- Quedamos con tres grupos
- La esquina superior izquierda solo con éxitos para la crítica
- La esquina superior derecha mayoritariamente con éxitos de taquilla
- La sección inferior con fracasos de taquilla
- Podemos predecir: un candidato nuevo, en función del número de celebridades y el presupuesto, cae en uno de los grupos
- Este modelo de decisión lo podemos representar como un árbol



# Ejemplo Árbol de Decisión



# Algoritmo C5.0

- Quedamos con tres grupos
- Hay diversas formas de implementar los árboles de decisión
- Uno de los más usados es el algoritmo C5.0. Es el estándar en la industria
- Es la evolución del algoritmo C4.5, que a su vez es la evolución de ID3, y así sucesivamente
- Es popular porque generalmente tiene buen desempeño, es fácil de entender y de implementar

# Algoritmo C5.0

Las principales fortalezas del C5.0 son:

1. Clasificador para todos los propósitos de buen desempeño en casi todo problema
2. Proceso de aprendizaje altamente automático, capaz de manejar variables numéricas o categóricas o valores ausentes
3. Excluye características irrelevantes
4. Produce un modelo que puede interpretarse sin contexto matemático
5. Más eficiente que otros modelos más complejos

# Algoritmo C5.0

Las principales debilidades del C5.0 son:

1. Los árboles suelen tener un sesgo hacia divisiones en características con muchos niveles
2. Es fácil sobreestimar o subestimar el modelo
3. Puede ser difícil modelar algunas relaciones porque se basa en divisiones paralelas a los ejes
4. Pequeños cambios en los datos de entrenamiento pueden llevar a grandes cambios en la lógica de decisión
5. Árboles muy grandes pueden ser difíciles de interpretar y las decisiones pueden ser contraintuitivas

# ¿Cómo Dividir de la Mejor Forma?

- El primer reto es decidir sobre qué característica empezar a dividir
- En el caso de las películas, dividimos para lograr que cada subconjunto tuviera solo una clase de objeto
- **Pureza:** nivel al cual un subconjunto de ejemplos contiene una única clase
- **Subconjunto Puro:** Aquel que contiene una única clase

# Índice de Entropía

- Existen varias formas de medir pureza
- La más utilizada es el índice de entropía:

$$Entropy(S) = \sum_{i=1}^c -p_i \log_2(p_i)$$

- Donde  $S$  es un segmento de datos,  $c$  es el número de niveles de clase
- $p_i$  es la proporción de clases en  $S$  que caen en el nivel  $c$
- Cuánto mayor sea el índice, más heterogeneidad. Buscamos minimizar la entropía

## Ejemplo: Índice de Entropía

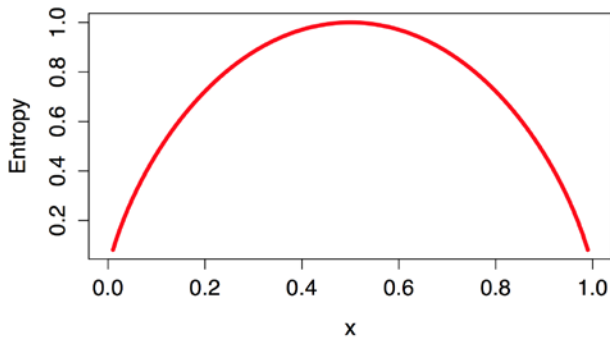
- Supongamos que tenemos una partición de los datos  $S$  con dos clases ( $c = 2$ ):
- Rojo:  $p_1 = 60\%$ ; Blanco:  $p_2 = 40\%$
- El índice de entropía es

$$Entropia(S) = -0.6 * \log_2(0.6) - 0.4 \log_2(0.4) = 0.97$$

- Hay poca homogeneidad. En cambio, si tuviéramos  $p_1 = .95$  y  $p_2 = .5$ , la entropía sería 0.28.
- Mucha más homogeneidad en este caso

# Índice de Entropía

Si  $x$  es la proporción en el grupo 1, tenemos:





# ¿Cómo Dividir de la Mejor Forma?

- Pero la pregunta es cuál es la característica óptima para dividir
- Para ello el algoritmo calcula el cambio en la homogeneidad que resultaría de dividir cada posible categoría
- Esta medida se llama *ganancia de información*
- Para una característica  $F$  la ganancia de información es

$$InfoGain(F) = Entropy(S_1) - Entropy(S_2)$$

- $Entropy(S_1)$  es la entropía del segmento antes de la partición y  $Entropy(S_2)$  en las particiones después de la partición

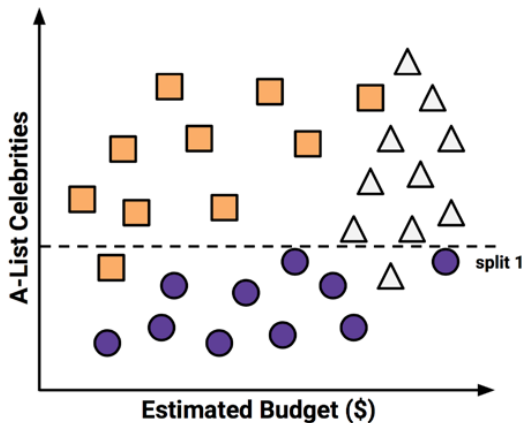
# ¿Cómo Dividir de la Mejor Forma?

- ¿Cómo se calcula la entropía de las particiones que genera una división?

$$Entropy(S) = \sum_{i=1}^n w_i Entropy(p_i)$$

- donde  $w_i$  es la proporción de ejemplos que caen en la partición  $i$ , para las  $n$  particiones generadas
- Es decir, se hace una suma ponderada de las entropías de cada grupo

## ¿Cómo Dividir de la Mejor Forma?



# ¿Cómo Dividir de la Mejor Forma?

- En el ejemplo la ganancia de información es mayor cuando se divide primero en número de celebridades
- Al hacerlo así nos queda una partición donde prácticamente todos son de la misma clase
- Dividiendo primero por presupuesto quedaríamos con dos grupos bastante heterogéneos
- Si la ganancia de información es cero, la división no sirve de nada pues se tiene la misma entropía
- Cuando la ganancia es igual a la entropía original, es porque la entropía final es 0, y los grupos resultantes son homogéneos

# ¿Cómo Dividir de la Mejor Forma?

- En el método anterior se asumen características nominales
- ¿Y si son numéricas? Se hace algo parecido
- Se prueban diferentes umbrales de la característica numérica
- Y se escoge la que genere mayor ganancia de información

# ¿Cuándo Parar?

- Es riesgoso seguir dividiendo indiscriminadamente el árbol
- Esto llevaría a un *overfitting* del modelo. Lo cual hace que las reglas sean inútiles para predecir
- Hay reglas para saber cuándo parar: pueden ser pre-poda o post-poda
- Si ex-ante se establece un número máximo de decisiones o de ejemplos en los nodos de decisión se está ante pre-poda
- Si se deja crecer el árbol para luego cortar lo que no es necesario, se está ante post-poda