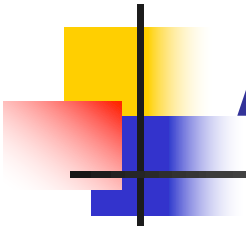
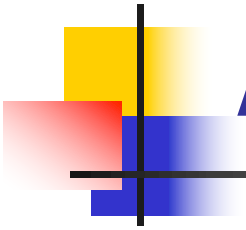
Aprendizaje de Máquina



Árboles de Decisión

- Aprendizaje supervisado
- Son algoritmos de clasificación y regresión
  - Para aproximar funciones objetivo con valores de salida continuos
  - Para aproximar funciones objetivo con valores de salida discretos

Árboles de Decisión

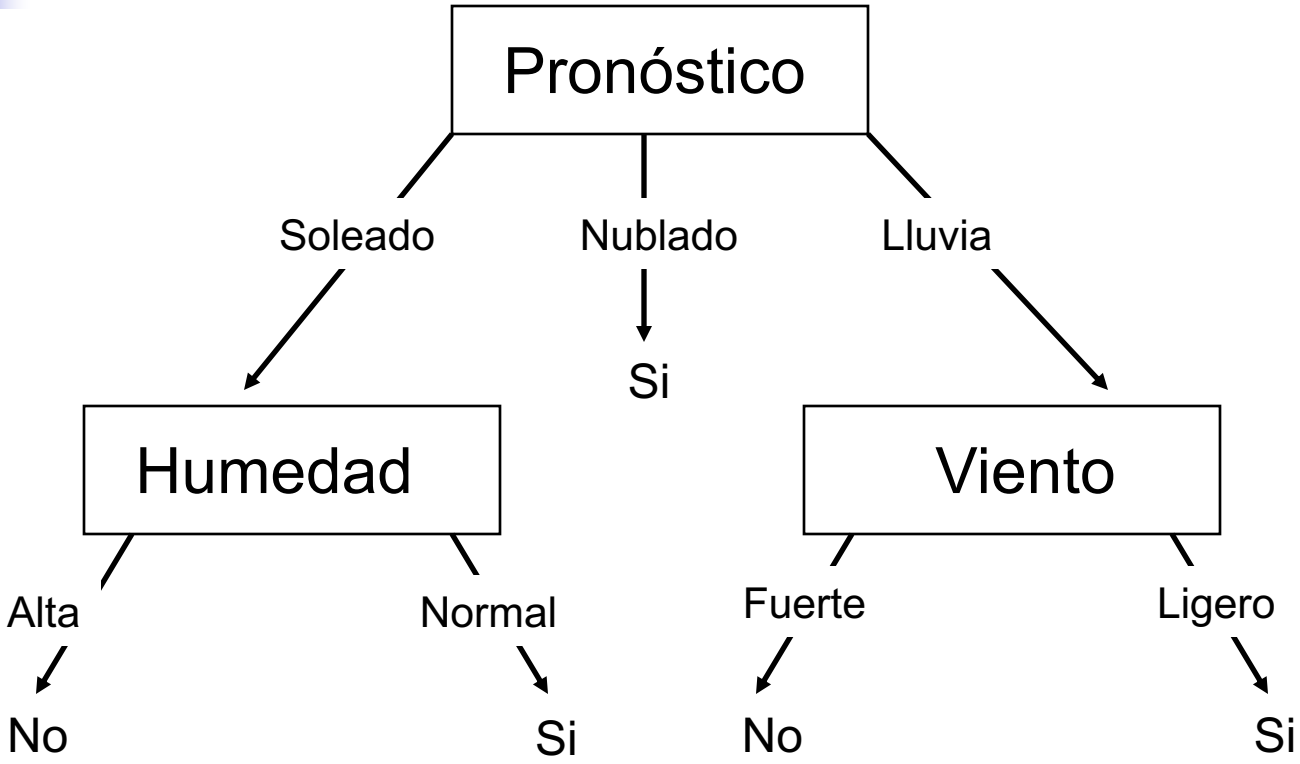


- Estos algoritmos proceden creando un árbol en donde cada nodo no terminal corresponde a una atributo (o rasgo) de las instancias del problema.
  - Los nodos terminales (las hojas) indican la clasificación de la instancia.
- En cada nodo se prueba el valor de un atributo y cada rama que desciende corresponde a un valor específico del atributo.
- Una instancia se clasifica moviéndose hacia abajo en el árbol (desde la raíz) por las ramas que corresponden a los valores de sus atributos

Árboles de Decisión

- Ejemplo (tomado de Quinlan 1986)
  - Las instancias tienen la siguiente forma:
    - <Pronostico,Temperatura, Humedad, Viento>
    - Los valores posibles son:
    - Pronóstico $\in$ {Soleado, Nublado, Lluvia}
    - Temperatura $\in$ {Caliente, Templado, Frío}
    - Humedad $\in$ {Alta, Normal}
    - Viento $\in$ {Fuerte, Débil}
  - Queremos clasificar de acuerdo a esto si un día es apropiado para jugar tenis

Árboles de Decisión



- Por ejemplo la instancia
  - (Pronostico=Soleado,Temperatura=Caliente, Humedad=Normal, Viento=Fuerte) sería considerado como buena para jugar tenis

Árboles de Decisión

- Nota: Este árbol no usa todos los atributos. Es posible que un subconjunto de los atributos sea suficiente para la tarea en cuestión
- En general un árbol de decisión representa una disyunción de conjunciones. Por ejemplo, los días que son buenos para jugar se caracterizan así:

  ((Pronostico = Soleado) **and** (Humedad= Normal))

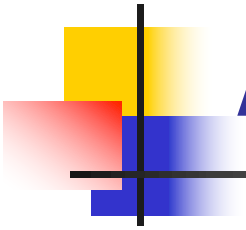
  or

  (Pronóstico= Nublado)

  or

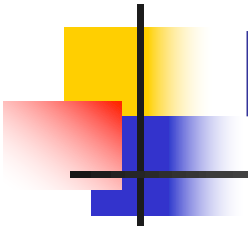
  ((Pronóstico=Lluvia) **and** (Viento=Débil))

Árboles de Decisión



- Ejemplos
  - Clasificación de enfermedades
  - Detección de fraudes
  - Detección de fallas

Árboles de Decisión Ejemplo

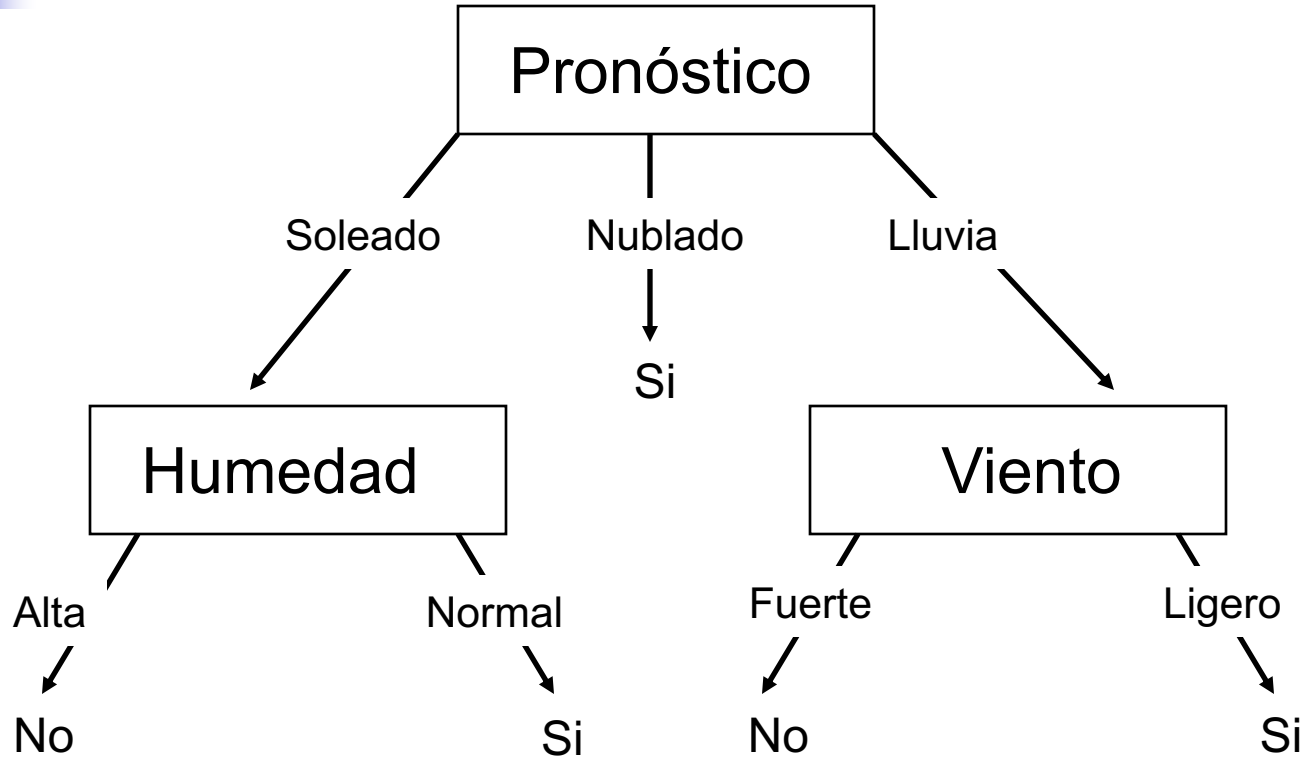


- Diseño
  - La función objetivo
    - Establece si un día es bueno o malo para jugar tenis dadas sus características
  - Una aproximación de la función objetivo
    - Estimar que tanta información provee cada atributo sobre la función objetivo
  - Método: Árbol de decisión ID3

Árboles de Decisión ID3

- Solamente utiliza atributos categóricos
- Este algoritmo comienza a construir el árbol desde la raíz
  - ¿Qué atributo debe estar en el nodo raíz?
- Cada atributo es evaluado estadísticamente para determinar que tan bien él solo clasifica los ejemplos de entrenamiento
- Una vez seleccionado éste, se crea un nodo sucesor para cada uno de los valores del atributo elegido
- Para elegir los nodos del siguiente nivel, se repite el proceso usando, para cada nodo, los ejemplos de entrenamiento correspondientes a esa rama del árbol (los ejemplos que por esa rama clasifica el nodo padre)
- Se repite el proceso hasta que todos los ejemplos estén clasificados

Árboles de Decisión
ID3



- Por ejemplo si se estima que Pronóstico es el atributo que por si solo clasifica mejor los ejemplos de entrenamiento, se pone en la raíz. Para cada nodo hijo se escoge un atributo de la misma forma, usando el subconjunto de los ejemplos de entrenamiento correspondientes

Árboles de Decisión Información

- ¿Cuál atributo es el mejor clasificador por si mismo?
- ¿Cuál atributo separa mejor los ejemplos de entrenamiento?
  - Nos referimos también a los ejemplos de entrenamiento como *M* o como instancias de entrenamiento
- La medida que se utiliza es la **ganancia de información**
- ¿Qué es la información?
  - La información que contiene un mensaje (evento, …) es la cantidad por la que éste reduce nuestra ignorancia
    - Mañana va a salir el sol
    - El número ganador de la lotería de la semana entrante es el 23 45 21 34 91

Árboles de Decisión Información

- La medida que buscamos debe dar valores altos a mensajes, eventos, etc., de cuyo contenido o resultado tenemos poca certeza y dar valor bajo a aquellos de los cuales sabemos mucho.
- Por ejemplo.
  - Supongamos una moneda balanceada
    - P(aguila)=0.5 y P(sol)= 0.5
  - Supongamos una moneda sesgada
    - P(aguila)=0.9 y P(sol)=0.1
- ¿Qué evento nos proporciona mas información? El resultado de tirar la moneda balanceada o el de la sesgada?
- ¿De qué evento tenemos más incertidumbre en promedio?

Árboles de Decisión Información

- La medida que usaremos se llama Entropía y su fórmula para las monedas es:
  - $H(M)=-(P(\text{aguila})\log_2 P(\text{aguila}) + P(\text{sol})\log_2 P(\text{sol}))$
  - Si maximizamos esta fórmula tenemos que $P(\text{aguila})=P(\text{sol})=0.5$
- Esto significa que si tenemos un clasificador binario y de los 20 ejemplos de entrenamiento 10 pertenecen a la clase uno (con respecto al cierto atributo) y 10 a la clase dos, la entropía de los ejemplos es máxima

Árboles de Decisión
Información

- En general, la fórmula para *c* categorías es:
  - $H(M)=\Sigma_{i=1,\ c} -p_i \log_2 p_i$
  - La sumatoria desde i hasta c, donde pi es la proporción de *M* que pertenece a la clase i

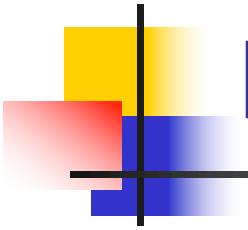
Árboles de Decisión ID3
Como escoger un nodo

- Regresando a cómo escoger un nodo para dividir *M*
- La ganancia de información es la reducción esperada en la entropía a causa de dividir *M* con respecto a un atributo *A* (¿cuánto se reduce nuestra incertidumbre?)
- Ganancia(*M*,*A*)=

  $$H(M)-\Sigma_{v\ \varepsilon\ Valores(A)}\ (|M_v|/|M|)*H(M_v)$$

  - Donde Valores(A) son los posibles valores del atributo A
  - El primer término es la entropía de *M* y el segundo es el valor promedio del entropía de la partición de M con respecto al atributo A
  - La Ganancia(*M*,*A*) es la reducción de la entropía dado A; la información que provee A acerca de la clasificación de los datos; en cuánto reduce A la incertidumbre

Árboles de Decisión ID3 Ejemplo



- Fijemos la atención sólo en dos atributos de nuestros ejemplos de entrenamiento *M*
  - Ejemplos: (Viento, Humedad,….)
    - Cada atributo tiene dos valores posibles: (Fuerte y Débil) y (Alto y Normal) respectivamente
- Supongamos que tenemos las siguientes 10 instancias

Árboles de Decisión Ejemplo

Viento	Humedad	Otros	Clase
Fuerte	Alta	...	no
Fuerte	Alta	...	no
Fuerte	Alta	...	no
Fuerte	Alta	...	no
Fuerte	Normal	...	si
Fuerte	Normal	...	si
Débil	Alta	...	no
Débil	Alta	...	no
Débil	Alta	...	si
Débil	Normal	...	si

H(M)=-(6/10)*$\log_2$(6/10) - (4/10)*$\log_2$(4/10)=0.97

Con respecto al Viento

H(M_F) =-(4/6)*$\log_2$(4/6) -(2/6)*$\log_2$(2/6)=0.92

H(M_D)= -(2/4)*$\log_2$(2/4) -(2/4)*$\log_2$(2/4)=1

Ganancia(M,Viento)= 0.97-6/10*0.92-4/10*1=0.018

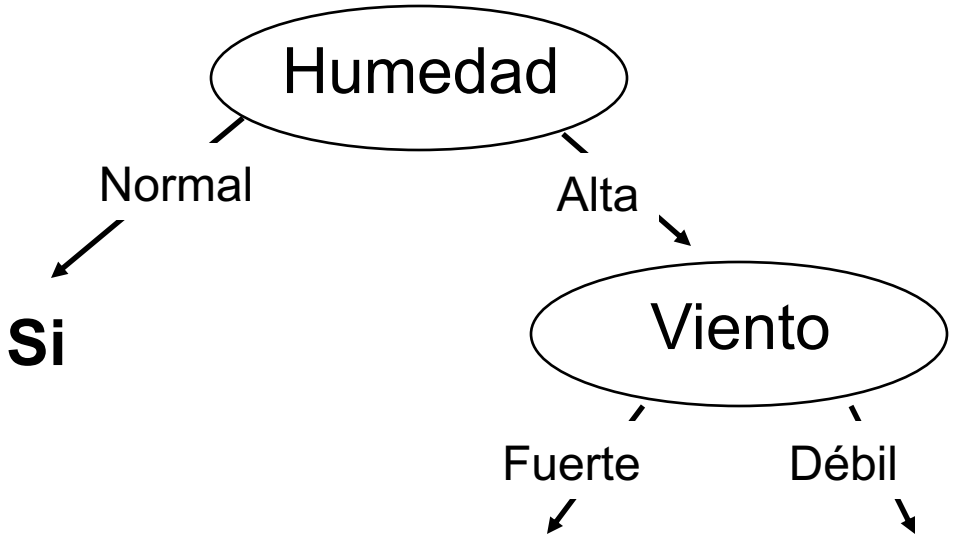
Con respecto a Humedad

H(M_A)= -(6/7)*$\log_2$(6/7) -(1/7)*$\log_2$(1/7)=0.59

H(M_N)= -(3/3)*$\log_2$(3/3) -(0/3)*$\log_2$(0/3)=0

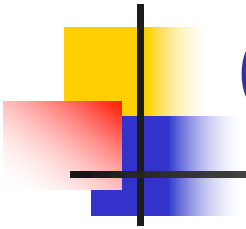
Ganancia(M,Hume)= 0.97 -7/10*0.59=0.55

Árboles de Decisión
Ejemplo



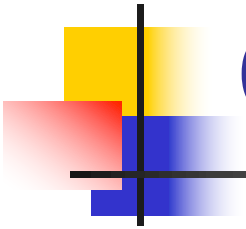
Si con 1/3
No con 2/3
Clasificamos
Como **No**

Generalización a Atributos Continuos



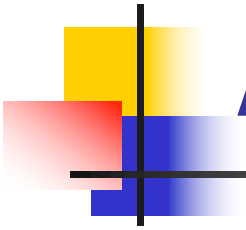
- El árbol que vimos hasta ahora sirve sólo para atributos categóricos
- Es posible generalizar esto para usar atributos continuos
- Una estrategia es simplemente discretizar los las atributos continuos

Generalización a Atributos Continuos



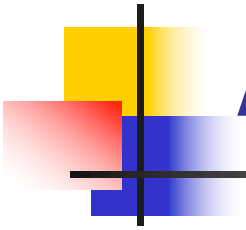
- Otra estrategia similar consiste en encontrar estos rangos de manera automática
  - Por lo general únicamente se divide en dos rangos (por nivel). La razón es que encontrar más rangos hace que el algoritmo sea más lento y no suele aportar mucho valor
  - Encontrar por atributo continuo el punto de corte que minimiza el error

Árboles para Regresión



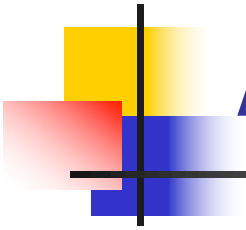
- Los árboles vistos hasta ahora sirven sólo para problemas de clasificación
- Es posible extender el método para valores continuos de la función objetivo, i.e., para problemas de regresión
- Los árboles mas comunes para esto se llaman CART (classification and regression trees)

Árboles para Regresión



- El algoritmo para la creación de árboles que discutimos requiere que el valor de la función objetivo sea discreto
- El algoritmo utiliza esta propiedad para calcular la ganancia de información de cada atributo con respecto a los ejemplos de entrenamiento
- ¿Qué hacemos si el valor de f.o. es continuo?

Árboles para Regresión



- Ya no podemos usar las mismas medidas para partir
- Estrategia: Hojas
  - El valor estimado del modelo en una hoja se define como la media de la f.o. de los ejemplos de entrenamiento que le corresponden
  - Definimos el error de la hoja como el error cuadrático medio de dichos ejemplos

Árboles para Regresión

- Estrategia: Nodos Intermedios
  - Determinar los grupos en los que se divide un atributo
    - Si es categórico: las categorías
    - Si es continuo: los rangos
  - Para cada grupo calcular el error cuadrático medio de los ejemplos que le corresponden (como la diferencia de cada valor de entrenamiento con la media de los valores de entrenamiento para ese grupo (varianza))
  - Calcular el error del atributo como el promedio de los errores del grupo
  - Escoger el atributo con el mínimo error

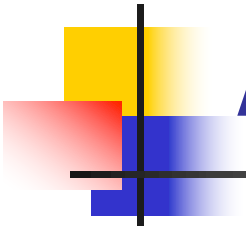
$$\arg\min_{A}\left(\sum_{V\in Valores(A)}\frac{|M_V|}{|M|}Error(M_V)\right)$$

Otras Generalizaciones y Variaciones

- Estos árboles toman de una en una variable. Existen algunos que se fijan en conjuntos más grandes de variables a la vez. Los resultados de éstos árboles suelen ser comparables a los "univariados" pero a un mayor costo. Por esto no se utilizan tanto.
- Otras medidas de ganancia para árboles clasificadores
  - Gini impurity: es una medida de qué tantas clases están representadas en un nodo

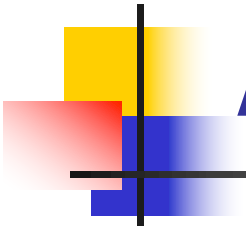
$$I_G = \sum_{i \in Categorias} f_i(1 - f_i) = 1 - \sum_{i \in Categorias} f_i^2$$

  - Donde f_i es el proporción de datos con categoría i representados en el nodo



Algunos puntos

- El algoritmo que vimos termina hasta que clasifica todos los ejemplos o cuando ya no haya más atributos que considerar
  - Esto tiene como posible consecuencia el sobre entrenamiento, un modelo demasiado complejo
- Para mitigar el problema del sobre entrenamiento se utilizan técnicas para podar los árboles (remover hojas) o controlar la profundidad máxima
  - Una manera de podar un árbol es mediante el uso de conjunto de datos de validación.

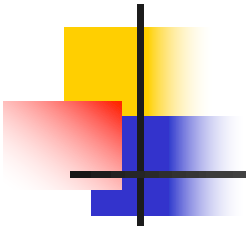


Algunos puntos

- Generalizaciones de Id3
  - Maneja atributos continuos
  - Utilización de otras técnicas para la selección de atributos
  - Manejo de datos con atributos faltantes
  - Técnicas para evitar el sobre-entrenamiento
- Esto resulta en un sistema de nombre C4.5 (diseñado por Quinlan). Ahora hay una nueva versión C5 (see5)

Árboles de Decisión
Cuando Usarlos

- Cuando las instancias son parejas de atributos con su valor
- Cuando los datos de entrenamiento tienen algunos errores
  - Son robustos
- Cuando algunas instancias tienen atributos faltantes
- Cuando se necesita una explicación de la clasificación
  - Cada ruta del árbol se puede convertir en una regla inteligible



Ejercicio

- Obtenga un archivo con datos de internet
- Elabore un modelo con un árbol de decisión en Python
- Obtenga las medidas de desempeño
- Grafique el árbol