HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Para un clasificador, manejar atributos numéricos de flujo de datos es mucho más difícil que en un entorno sin flujo
    - En árboles de decisión, es necesario gestionar las estadísticas de los atributos numéricos para determinar los mejores puntos de división
  - En general, los métodos se basan en almacenar información del atributo para realizar aproximaciones
    - Precisión vs. almacenamiento y eficiencia

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Principales métodos en el escenario sin flujo de datos:
    - Igual anchura
    - Igual frecuencia
    - Método de Fayyad e Irani

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Principales métodos sin flujo de datos:
    - Igual anchura
      - El rango del atributo numérico se divide en una cantidad fija de intervalos o *bins* del mismo tamaño
      - Se necesitan los valores máximo y mínimo para calcular los valores superior e inferior de los *bins*
      - Es el método más sencillo, ya que no necesita ordenar los datos
      - Sin embargo, es vulnerable a la existencia de valores atípicos y a las distribuciones sesgadas

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Principales métodos sin flujo de datos:
    - Igual frecuencia
      - También utiliza un número fijo de intervalos, pero cada intervalo contiene el mismo número de elementos
        - Para n valores y k intervalos, cada intervalo contendrá n/k valores
      - Es adecuado para valores atípicos y distribuciones sesgadas
      - Requiere más tiempo de procesamiento
        - Necesita ordenar los valores

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Principales métodos sin flujo de datos:
    - Método de Fayyad e Irani:
      - También llamado MDLP
        - *Minimum Description Length Principle*
      - Se basa en calcular los mejores puntos de corte utilizando la ganancia de información
        - Como en los árboles de decisión
      - En primer lugar, ordena los datos
      - A continuación, cada punto entre pares de valores adyacentes se selecciona como candidato a división
        - Utilizando la ganancia de información, se selecciona el mejor punto de corte

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Principales métodos sin flujo de datos:
    - Método de Fayyad e Irani:
      - El procedimiento continúa recursivamente en cada una de las partes
        - Se necesita un criterio de parada para detener el proceso recursivo
        - Posibles criterios:
          - Los intervalos se vuelven puros
            - Con valores de una sola clase
          - El principio de longitud mínima de descripción estima que dividir el intervalo numérico no aportará más beneficios
            - *Minimum Description Length Principle*

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - En escenarios con flujo de datos:
    - VFML
    - *Exhaustive Binary Tree*
    - Resumen cuantílico de Greenwald-Khanna
    - Aproximación gaussiana

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - VFML
    - Incluido en el paquete VFML (*Very Fast Machine Learning*) para manejar atributos numéricos en VFDT y CVFDT
    - Los valores de los atributos numéricos se resumen mediante un conjunto de *bins* ordenados
    - El rango de valores cubierto por cada *bin* se fija en el momento de la creación y no cambia a medida que se ven más ejemplos
    - Un parámetro oculto sirve para limitar el número total de contenedores permitidos
      - En el paquete VFML, este parámetro toma el valor 1000

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - VFML
    - Inicialmente, por cada nuevo valor numérico único visto, se crea un nuevo *bin*
      - Con un contador
    - Una vez asignado el número fijo de *bins*, cada valor posterior del flujo actualiza el contador del bin más cercano
    - En esencia, se está creando un histograma compuesto por un máximo de 1000 *bins* (VFML)
      - Los límites de los *bins* están determinados por los primeros 1000 valores únicos vistos en el flujo
      - Después, los recuentos de los intervalos estáticos se actualizan de forma incremental

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - VFML
    - Este método puede plantear dos problemas:
      - Sensible al orden de los datos
        - Si los primeros 1.000 ejemplos vistos en un flujo están sesgados hacia un lado del rango total de valores, el resumen final no puede representar con precisión todo el rango de valores
    - Calcular el número óptimo de intervalos
      - Muy pocos → el resumen será pequeño pero impreciso
      - Demasiados → la precisión aumentará a costa del espacio

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - *Exhaustive Binary Tree*
    - Presentado en el sistema VFDTc
    - De cada dato no descarta más información que el orden de los valores
    - Funciona construyendo incrementalmente un árbol binario a medida que llegan los valores
      - El camino que sigue un valor en el árbol depende de si es menor, igual o mayor que el valor de un nodo concreto del árbol
      - Los valores se ordenan implícitamente a medida que se construye el árbol
    - El número de nodos del árbol en cada momento es el número de valores distintos observados

HOEFFDING TREES

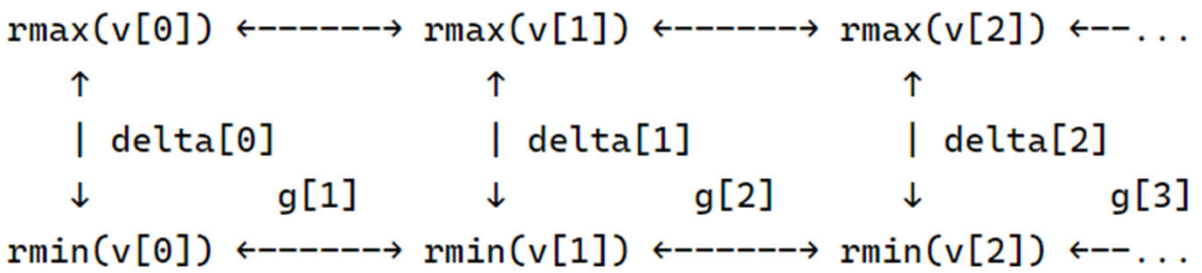
- Cómo trabajar con atributos numéricos
  - *Exhaustive Binary Tree*
    - Este método ahorra:
      - Espacio si el número de valores distintos es pequeño
      - Tiempo de búsqueda frente al almacenamiento de los valores en una matriz si que el árbol esté razonablemente equilibrado
        - En particular, si los valores llegan en orden, entonces el árbol degenera en una lista y no se ahorra tiempo de búsqueda
    - En general, este método consigue una precisión perfecta a expensas del espacio de almacenamiento

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Resumen cuantílico de Greenwald-Khanna
    - Se mantiene un subconjunto ordenado de observaciones: v_0 ß# v_1 ß# … ß v_{s-1}
      - Con cada nuevo valor se añade
    - Cada elemento v_i define un intervalo
      - Por tanto, para v_i es necesario almacenar los límites del intervalo $r_{min}(v_i)$ y $r_{max}(v_i)$
        - Mucha información que actualizar con nuevos valores
      - En lugar de ello, para cada elemento v_i se almacena una tupla $t_i = (v_i, g_i, \Delta_i)$, donde:
        - $g_i = r_{min}(v_i) - r_{min}(v_{i-1})$
        - $\Delta_i = r_{max}(v_i) - r_{min}(v_i)$
      - La tupla implícitamente almacena los límites del intervalo

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Resumen cuantílico de Greenwald-Khanna
    - Tupla $t_i = (v_i, g_i, \Delta_i)$:



    - Para mantener un número bajo de tuplas, periódicamente se fusionan tuplas contiguas
      - Si se decide fusionar t_i y t_{i+1}, se sustituyen por $t_* = (v_*, g_*, \Delta_*)$
        - $v_* = v_{i+1}$
        - $g_* = g_i + g_{i+1}$
        - $\Delta_* = \Delta_{i+1}$

HOEFFDING TREES

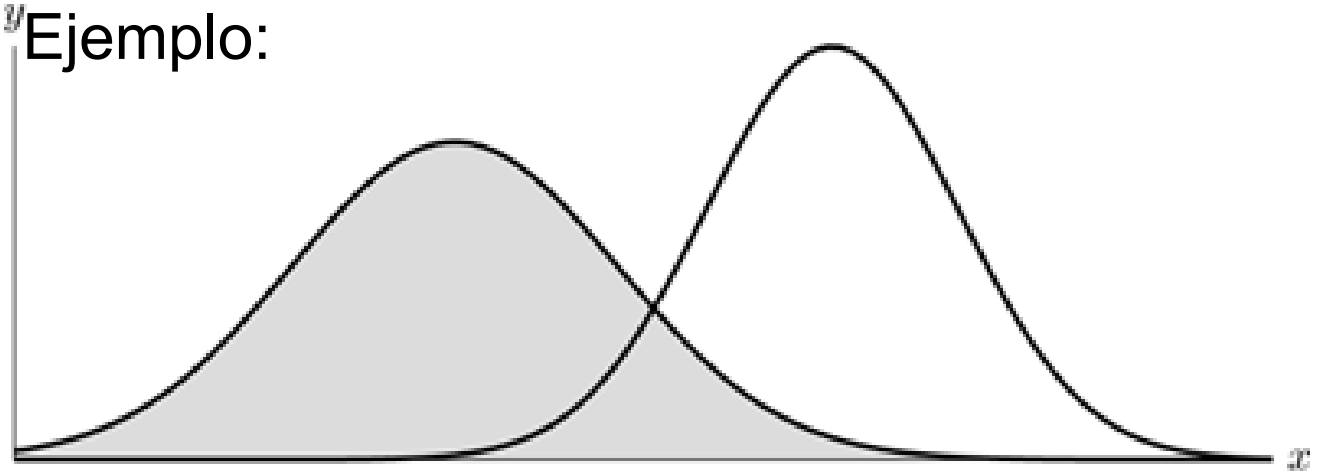
- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - Un método similar a este se utiliza en UFFT
    - Aproxima una distribución numérica utilizando una distribución gaussiana o normal
      - Esta distribución puede ser mantenida incrementalmente almacenando sólo tres números
      - De forma insensible al orden de los datos
    - Para cada atributo numérico, se mantiene una distribución gaussiana separada por etiqueta de clase

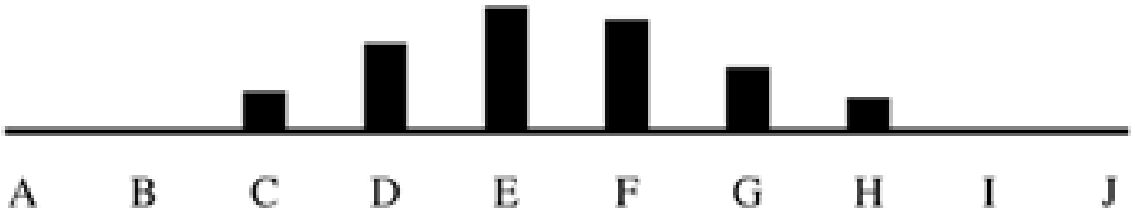
HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - Los valores posibles para partir se reducen a un conjunto de puntos repartidos equitativamente en el rango, entre los valores mínimo y máximo observados
      - El número de puntos de evaluación se determina mediante un parámetro
    - Para cada punto candidato, el peso de los valores a cada lado de la división puede aproximarse para cada clase
      - Utilizando sus respectivas curvas de Gauss
      - La ganancia de información se calcula a partir de estos pesos

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - Ejemplo:





Information gain

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - Ejemplo:
      - 2 curvas gaussianas, para un atributo numérico, 2 clases
      - Cada curva puede describirse utilizando tres valores: la media, la varianza y el peso total de los ejemplos
        - Ejemplo: La clase izq. tiene menor media, mayor varianza y un mayor peso de ejemplos que la otra clase
      - Debajo de las curvas, el rango de valores se ha dividido en diez puntos de división, etiquetados de la A a la J
      - Barra vertical de la parte inferior:
        - Cantidad relativa de ganancia de información calculada para cada punto
          - Punto de división que se elige: E (mayor ganancia info)

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - Mejora:
      - Almacenar los valores mínimo y máximo de cada clase
      - Esto permite:
        - Determinar cuándo los valores de las clases se sitúan completamente a un lado de una división
          - Elimina la incertidumbre presente en las colas de las curvas de Gauss
        - Establecer el mínimo y el máximo de toda la gama de valores
          - Esto ayuda a determinar la posición de los puntos de división que hay que evaluar

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - En general, esta aproximación no captará todos los detalles de una distribución numérica compleja
      - Se perderá precisión
      - Sin embargo, es un método eficiente tanto en cálculo como en memoria
    - En el otro extremo, *Exhaustive Binary Trees* tiene gran coste de memoria
      - Pero resulta muy preciso

HOEFFDING TREES

- Cómo trabajar con atributos numéricos
  - Aproximación gaussiana
    - A pesar de que se pierde precisión, esto no resulta muy perjudicial
      - Habrá más oportunidades de refinar las decisiones de división en un atributo concreto volviendo a dividir más abajo en el árbol
    - Además, la aproximación mediante unos pocos parámetros simples puede ser más robusta y resistente al ruido y a los valores atípicos
      - Los métodos más complicados se concentran en detalles más “finos”
        - Pueden sobreajustarse a los datos