ITESO, Universidad
Jesuita de Guadalajara

Discretización de variables

Dr. Gaddiel Desirena López

Primavera 2024

Contenido

Discretización de variables

Los recuentos brutos que abarcan varios órdenes de magnitud son problemáticos para muchos modelos.

- En un modelo lineal, el mismo coeficiente lineal tendría que funcionar para todos los valores posibles del recuento.
- Los recuentos grandes también podrían causar estragos en los métodos de aprendizaje no supervisados.

Una solución es agrupar los recuentos en contenedores y nos deshacemos de los valores de recuento reales.

El conjunto resultante es una **secuencia ordenada de contenedores** que representan una medida de intensidad en el conjunto real.

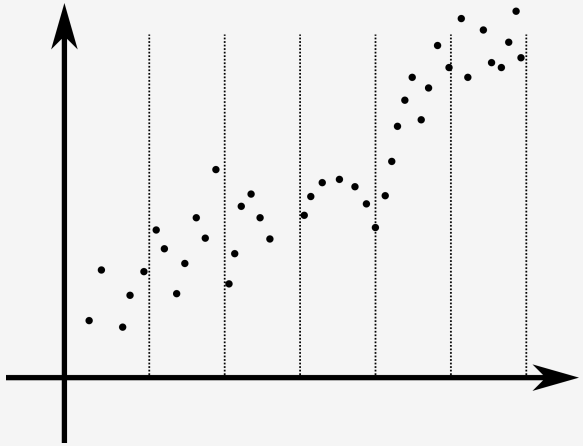
Discretización de variables

- Es el proceso de transformar variables continuas en variables discretas mediante la creación de un conjunto de intervalos contiguos (*bins*).
- La discretización se utiliza para cambiar la distribución de variables asimétricas y minimizar la influencia de valores atípicos y, por lo tanto, mejorar el rendimiento de algunos modelos de aprendizaje automático.

Intervalos de ancho constante

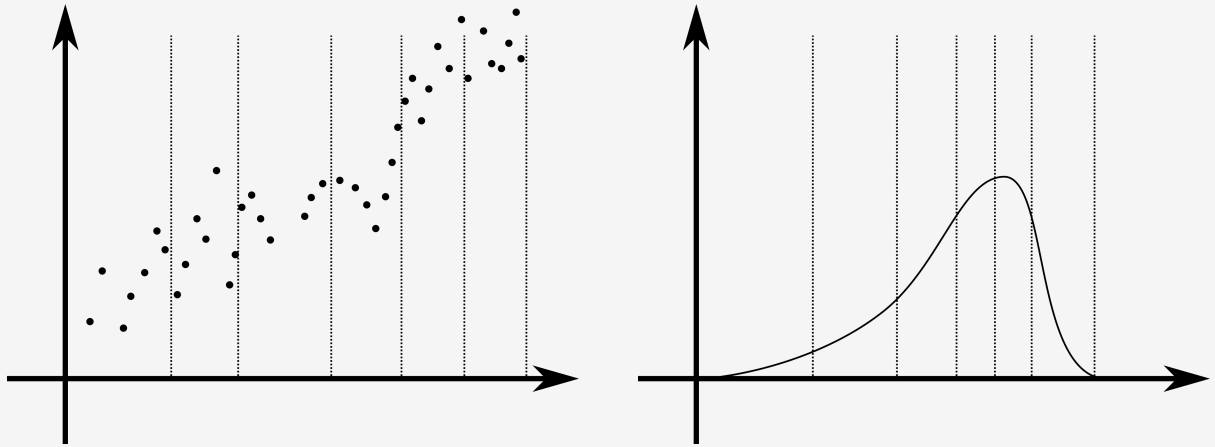
Con la agrupación de ancho fijo,

- cada bandeja (*bina*) contiene un rango numérico específico.
- Los rangos se pueden diseñar a medida o segmentar automáticamente, y se pueden escalar linealmente o exponencialmente.
- El agrupamiento de ancho fijo es fácil de calcular. Pero si hay grandes lagunas en los recuentos, habrá muchos contenedores vacíos sin datos.



Intervalos de frecuencia constante

- Si clasifica las observaciones en *bins* con la misma frecuencia, la discretización distribuye los valores de una variable sesgada de forma más homogénea en todo el rango de valores.
- Este problema de binas vacías se puede resolver colocando de forma adaptativa los contenedores en función de la distribución de los datos.
- Se usan los **cuantiles** de la distribución. Dividiendo los datos en partes iguales.



Discretización por K–Means

En la agrupación de k–means

- Se usa el algoritmo de K–means para clasificar.
- Las agrupaciones se definen por los elementos pertenecientes a cada cluster.
- El número de agrupaciones las define el usuario.

K–Means

Consiste en

1. En el paso de inicialización, se eligen K observaciones al azar como los centroides de los K grupos.
2. Los datos restantes se asignan al grupo **más cercano** de cada centroide.
3. En el paso de iteración, los centroides se vuelven a calcular como los puntos promedio de todas las observaciones dentro del grupo.
4. Las observaciones se reasignan al grupo más cercano recién creado.

El paso de iteración continúa hasta que se encuentran los k centros óptimos.

Discretización por Arboles de decisión

Consiste en utilizar un árbol de decisión para identificar los bins óptimos en los que ordenar los valores de las variables.

- El árbol de decisión se construye utilizando la variable a discretizar y el objetivo.
- Cuando un árbol de decisión hace una predicción, asigna una observación a una de las N hojas finales, por lo tanto, cualquier árbol de decisión generará una salida discreta, cuyos valores son las predicciones en cada una de sus N hojas.
- La discretización con árboles de decisión crea una relación monótona entre los contenedores y el objetivo.

Arboles de decisión

El proceso se divide en dos etapas

- División sucesiva del espacio generando regiones disjuntas R_1, R_2, R_3, ... R_j.
- Predicción de la variable respuesta en cada región.

El objetivo es encontrar las J regiones (R_1, ... R_j) que minimicen la distancia entre la media de las respuestas de las j observaciones $\hat{y}_{R_j}$ y su objetivo y_i, esto es

$$\text{mín} \sum_{j=1}^{J} d(y_i, \hat{y}_{R_j})$$

Infortunadamente, no es posible considerar todas las posibles particiones del espacio, por esta razón, se recurre a lo que se conoce como *Partición binaria recursiva*.

Partición binaria recursiva

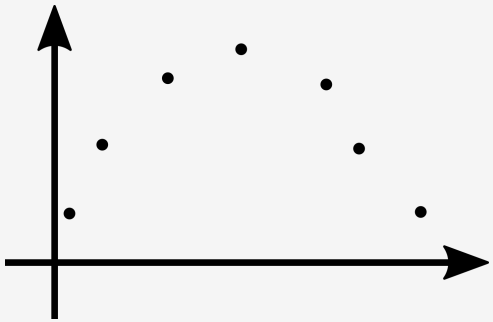
El objetivo es encontrar en cada iteración el predictor X_j y el umbral s tal que, si se distribuyen las observaciones en las regiones $\{X \mid X_j < s\}$ y $\{X \mid X_j \geq s\}$, se consigue la mayor reducción posible entre el objetivo y el promedio de la observación. El algoritmo es:

1. Se identifican todos los posibles puntos de corte s para cada uno de los predictores $(X_1, \ldots, X_p)$.
   - En el caso de predictores cualitativos, los posibles puntos de corte son cada uno de sus niveles.
   - Para predictores continuos, se ordenan de menor a mayor sus valores, el punto intermedio entre cada par de valores se emplea como punto de corte.
2. Se calcula la distancia total que se consigue con cada posible división.

$$d(y_i, \hat{y}_{R_1}) + d(y_i, \hat{y}_{R_2}), \qquad \forall\, i : x_i \in R_j, \;\; j = \{1, 2\}$$

3. Se selecciona el predictor X_j y el punto s que resulta en la menor distancia total.

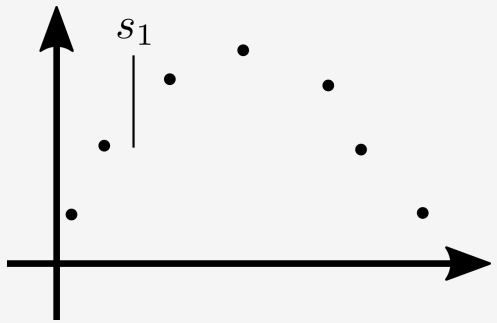
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

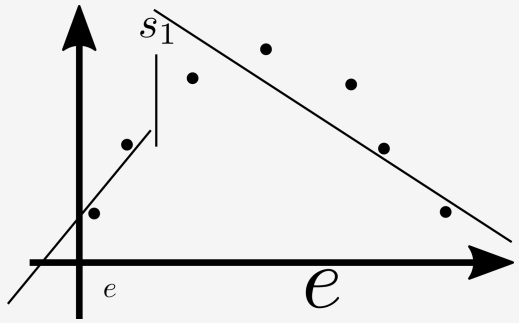
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

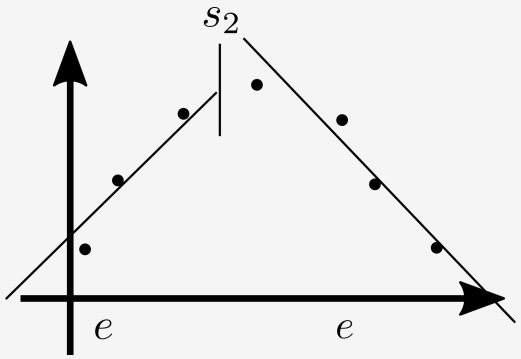
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

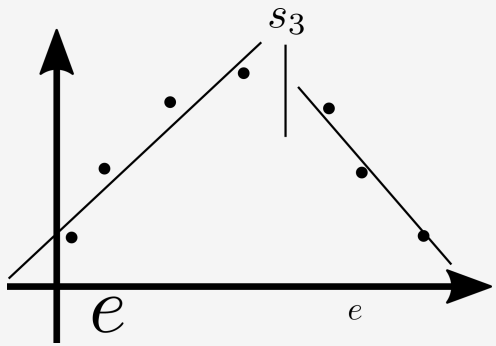
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

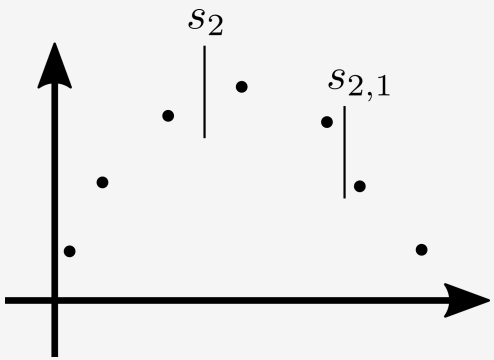
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

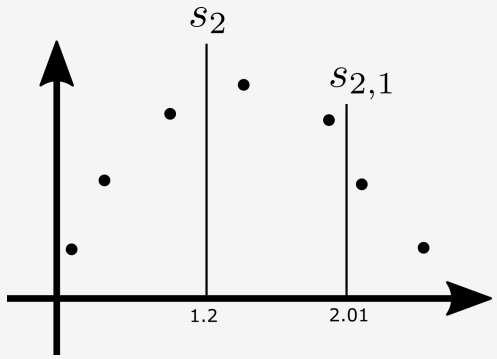
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

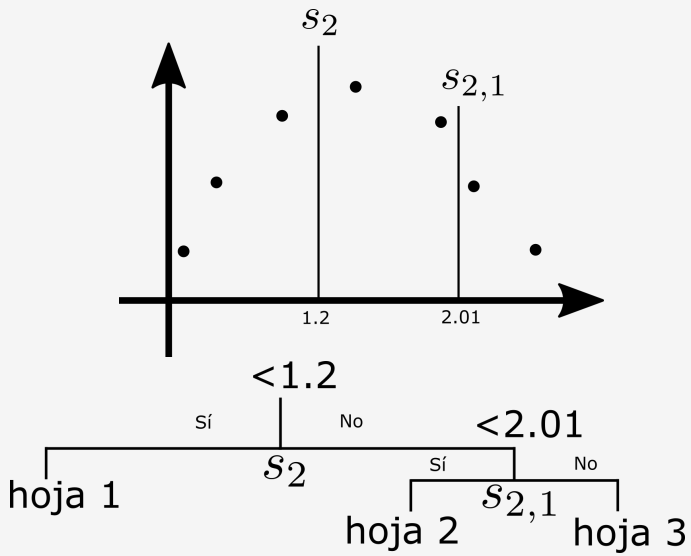
Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

Partición binaria recursiva



Criterios de paro

- Que ninguna región contenga un mínimo de n observaciones.
- Que el árbol tenga un máximo de nodos terminales.
- Que la incorporación del nodo reduzca el error en al menos un porcentaje mínimo.

Métricas

- **Residual Sum of Squares (RSS):**

$$RSS = \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2,$$

- **Distancia de Minkowski:**

$$d(y, \hat{y}_{R_j}) = \left(\sum_{i=1}^{J} |y_i - \hat{y}_{R_j}|^p \right)^{1/p},$$

  - cuando $p = 2$, se tiene la distancia **Euclideana**.
  - cuando $p = 1$, se tiene la distancia **Manhattan**.
  - cuando $p = \infty$, se tiene la **máxima distancia**. En este caso, la distancia corresponde a la componente $|y_i - \hat{y}_{R_j}|$ con el valor mas alto.

Métricas

Para construir un árbol de clasificación, como la variable respuesta en cualitativa, no es posible emplear un criterio de selección continuo. Para ello, existen varias alternativas:

- **Gini index:** En el conjunto de las K clases del nodo m, se tiene

$$G_m = \sum_{k=1}^{K} \hat{p}_{m,k}(1 - \hat{p}_{m,k}),$$

  donde $\hat{p}_{m,k}$ es la proporción de observaciones del nodo m que pertenecen a la clase k.
- **Information gain: Cross entropy:**

$$D = -\sum_{k=1}^{K} \hat{p}_{m,k} \log(\hat{p}_{m,k}).$$

- **Chi–cuadrada χ^2:**

$$\chi^2 = \sum_{k} \frac{(y_k - \hat{y}_k)^2}{\hat{y}_k}$$