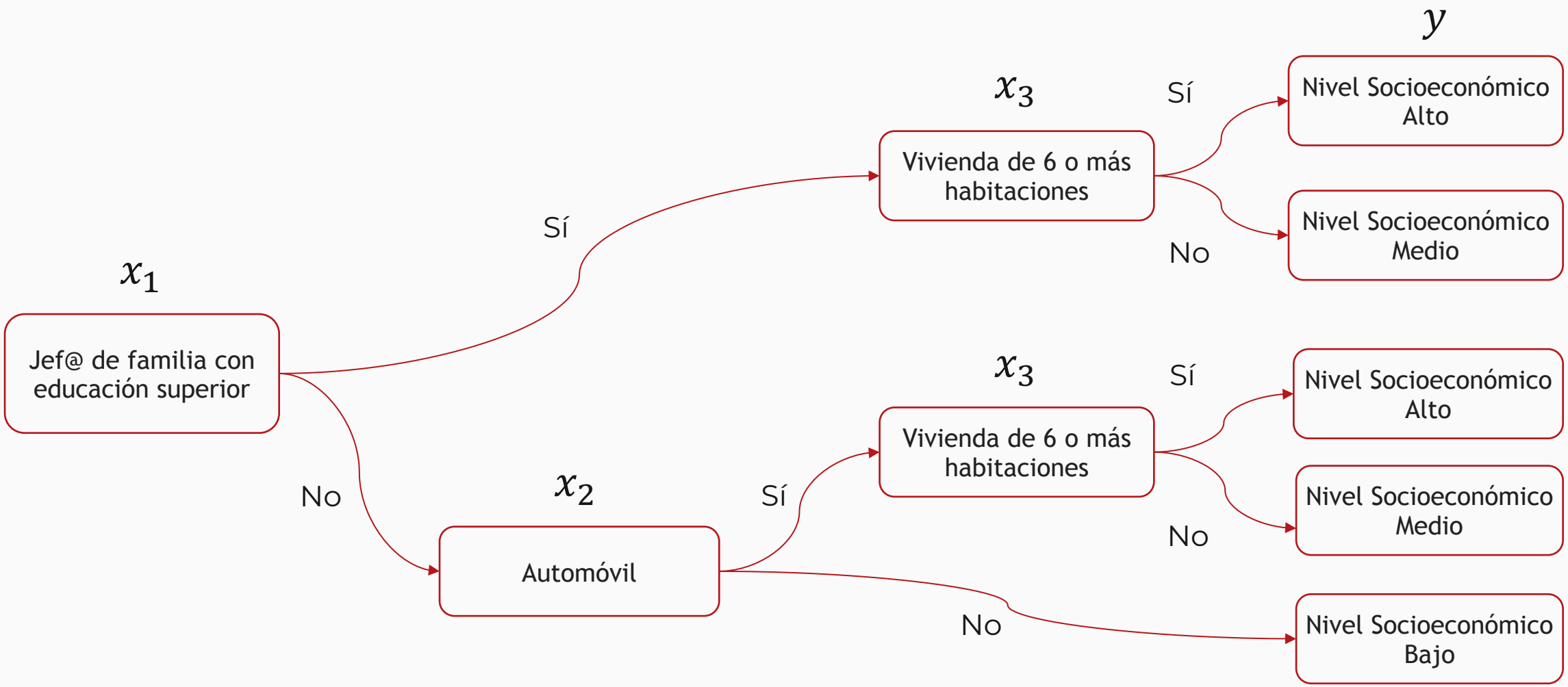
Tecnológico de Monterrey
Escuela de Ciencias Sociales
y Gobierno

Métodos Basados en Árboles

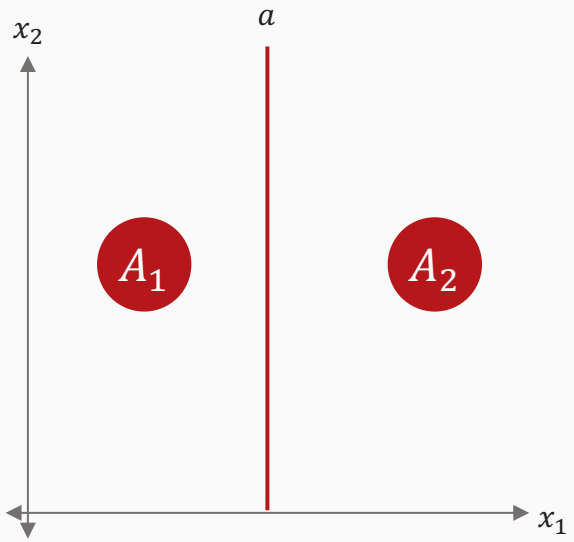
Mtro. René Rosado González
Director de Programa LTP

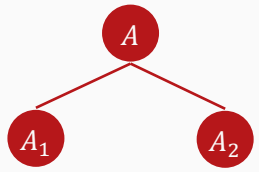
Árboles Aleatorios



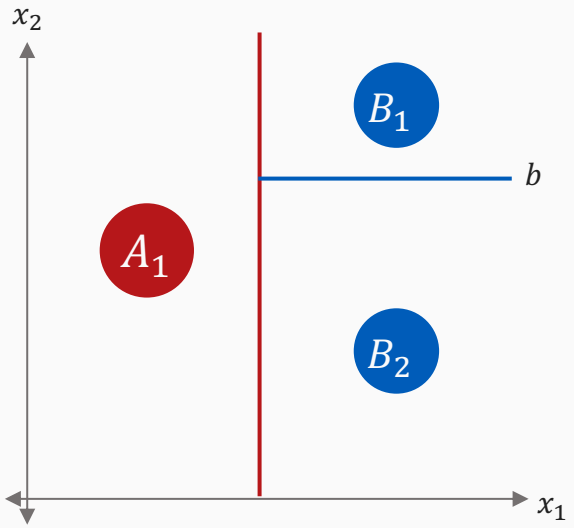
Árboles Aleatorios

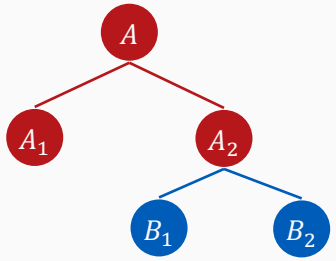
Primera Partición



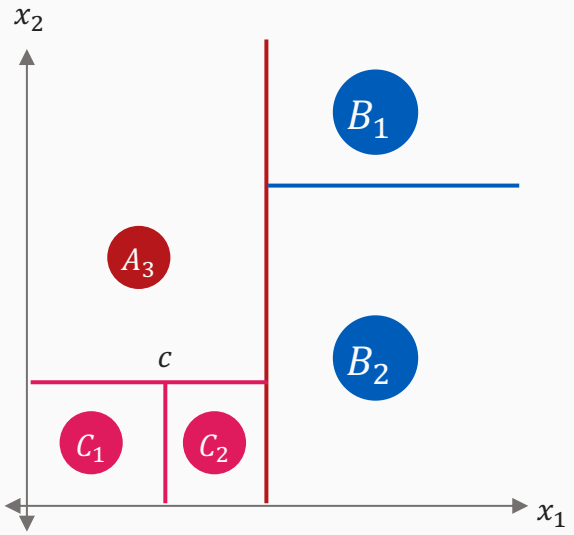


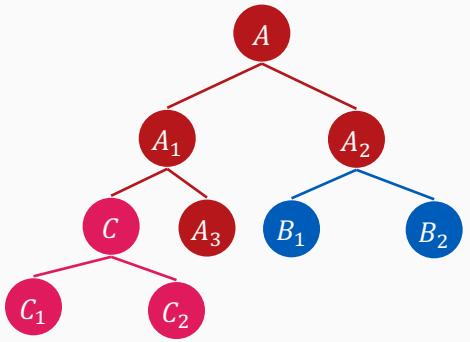
Segunda Parición





Tercera Partición

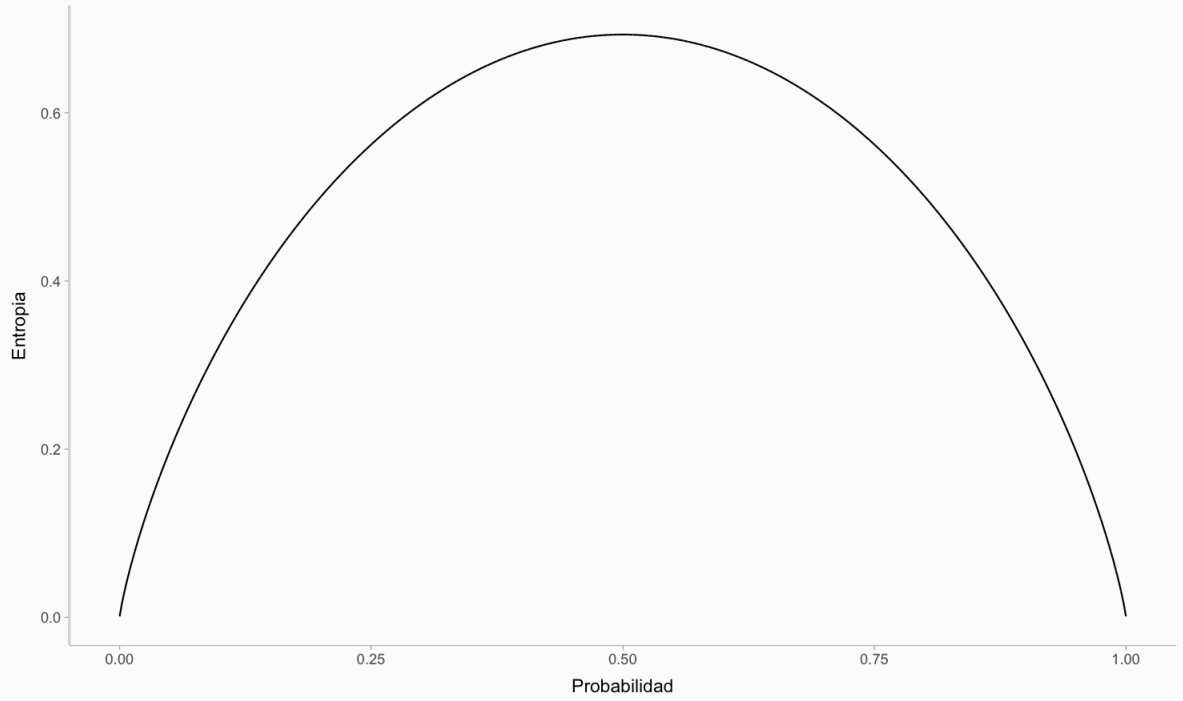




Árboles Aleatorios

Medidas de Impureza

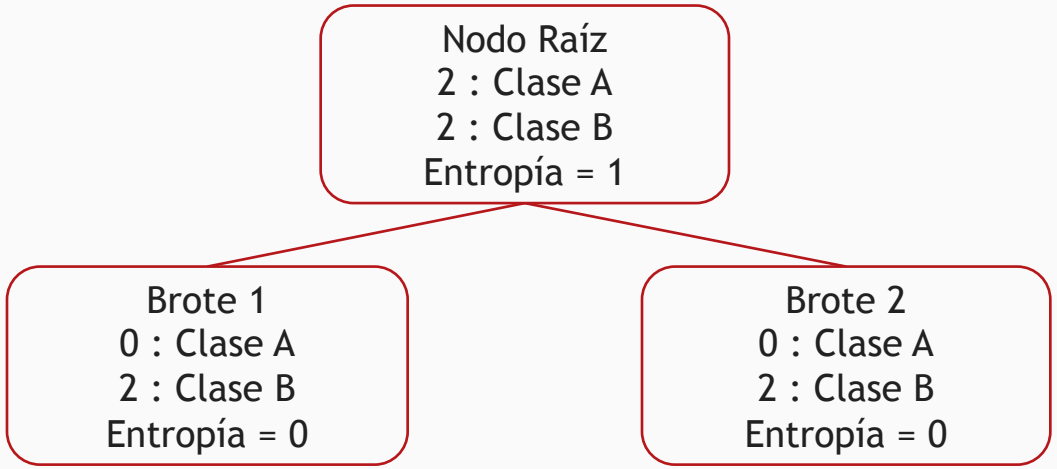
$$entropía(x) = \sum_{j=1}^{J} -\big(p(x)\log_2\big(p(x)\big) + \big(1-p(x)\big)\log_2\big(1-p(x)\big)\big)$$



Árboles Aleatorios

Ganancias de Información

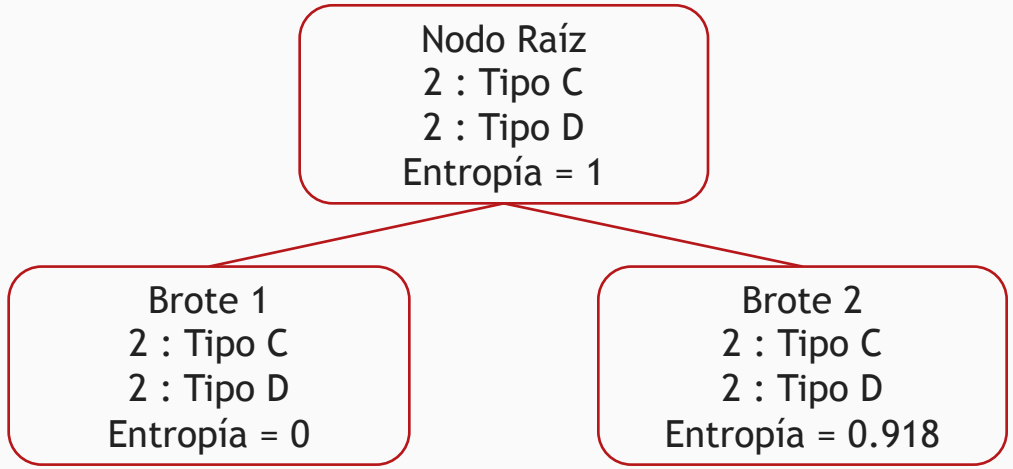
Partición por Clases



$$GI = 1 - \left(\frac{2}{4} * 0\right) + \left(\frac{2}{4} * 0\right) = 1$$

Entropía del nodo raíz — Entropía del brote 1 — Entropía del brote 2

Partición por Tipos

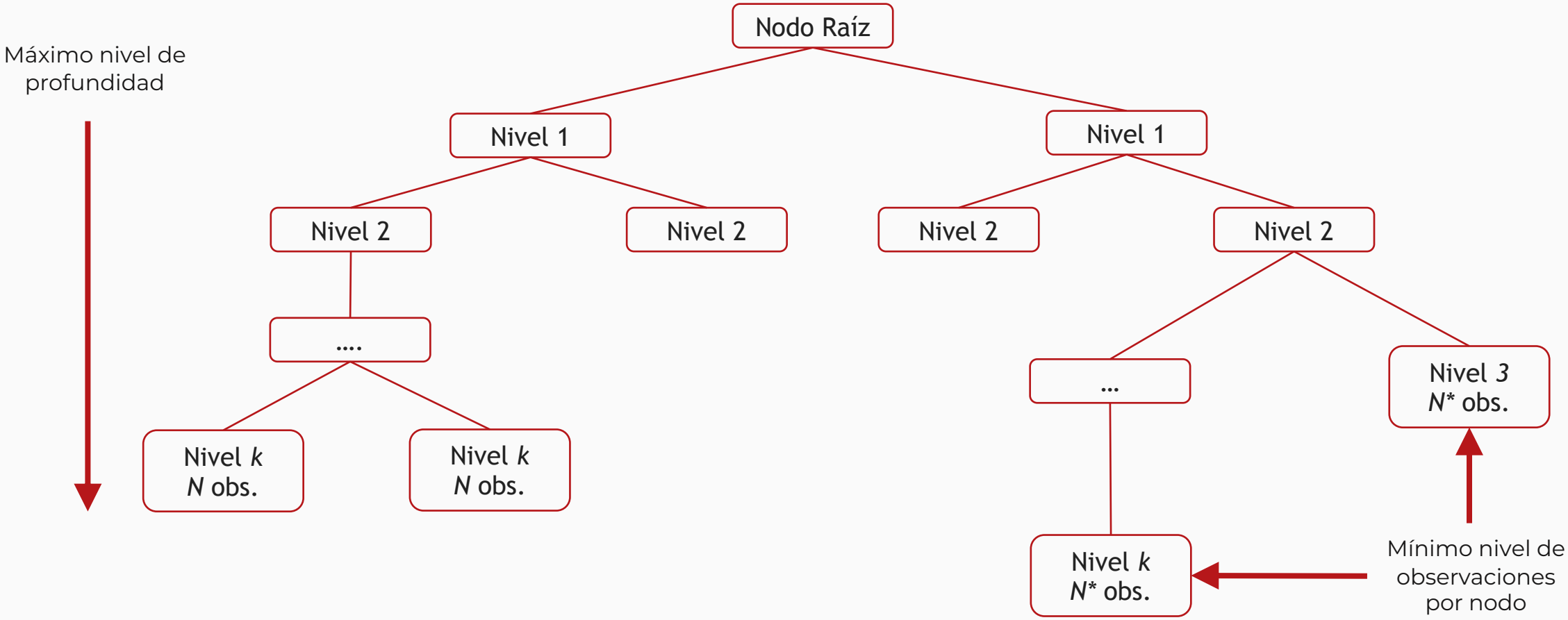


$$GI = 1 - \left(\frac{1}{4} * 0\right) + \left(\frac{3}{4} * 0.918\right) = 0.312$$

de elementos en noro raíz — # de elementos en broes — Ganancias de Información

Árboles Aleatorios

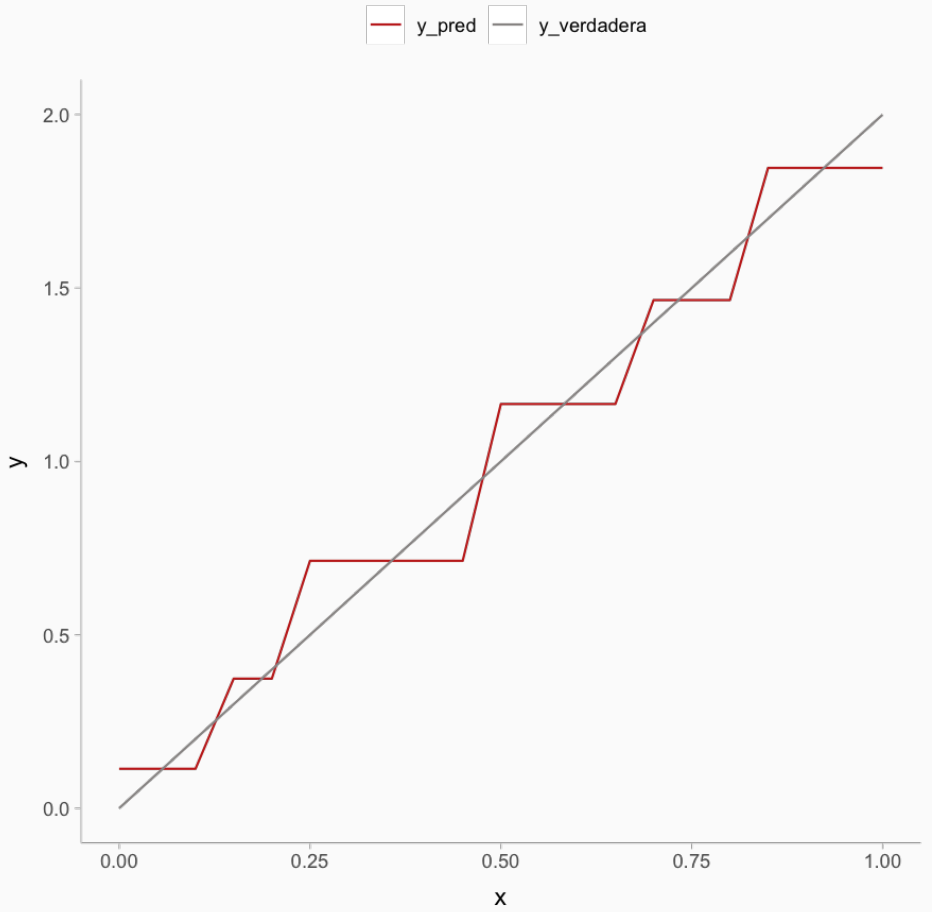
Reglas de Paro



Un ejemplo

Árboles Aleatorios

para problemas lineales



Árboles Aleatorios

Ventajas:

- Árboles chicos son relativamente fáciles de explicar
- Capturan interacciones entre las variables de entrada
- Son robustos en el sentido de que
  - valores numéricos atípicos no hacen fallar al método
  - no es necesario transformar (monótonamente) variables de entrada
  - hay formas fáciles de lidiar con datos faltantes (cortes sucedáneos)
- Se ajustan rápidamente y son relativamente fáciles de interpretar
- Árboles grandes generalmente no sufren de sesgo.

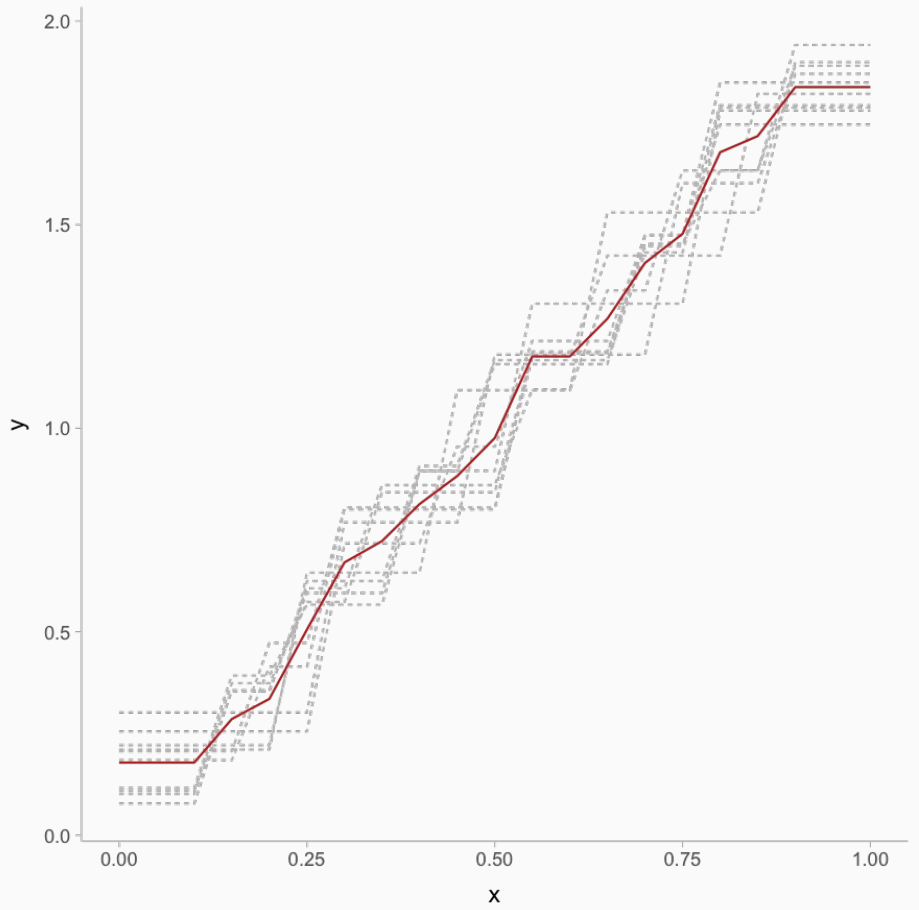
Desventajas:

- Tienen dificultades en capturar estructuras lineales.
- Muchas veces algunas variables de entrada “enmascaran” a otras.
- Son inestables (varianza alta) por construcción.
- Esto produce desempeño predictivo relativamente malo.
- No son apropiados cuando hay variables categóricas con muchas niveles: en estos casos, el árbol sobreajusta desde los primeros cortes, y las predicciones son malas.

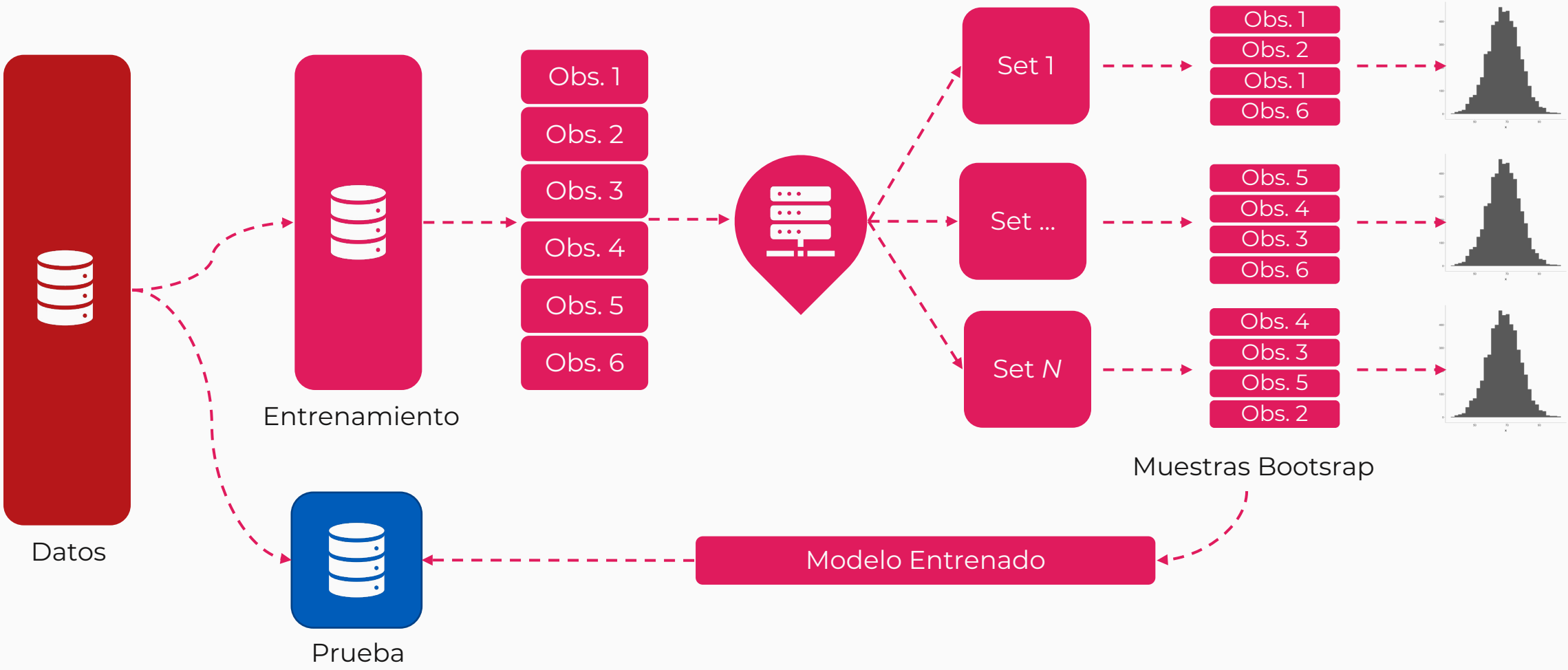
Bagging en Árboles Aleatorios

Bootsrap Aggregation

- Los árboles grandes tienen la ventaja de tener sesgo bajo, pero sufren de varianza alta. Podemos explotar el sesgo bajo si logramos controlar la varianza.
- Una alternativa es perturbar la muestra de entrenamiento de distintas maneras y producir árboles distintos.
- La perturbación más usada es tomar muestras *bootstrap* de los datos y ajustar un árbol a cada muestra *bootstrap*.
- Promediar el resultado de todos estos árboles para hacer predicciones.
- El proceso de promediar reduce la varianza, sin tener pérdidas en sesgo.



Remuestreo Bootstrap



Bagging en Árboles Aleatorios

Bootsrap Aggregation

Consideremos una muestra L con la que ajustamos un árbol T_L

$$L \rightarrow T_L$$

La idea del *Bagging* es construir B muestras con la que podremos ajustar B árboles.

$$T_1, T_2, \dots, T_B$$

Y construir un árbol promedio

$$T(x) = \frac{1}{B}\sum_{i=b}^{B} T_b(x)$$

Bagging en Árboles Aleatorios

Bootsrap Aggregation

El sesgo del árbol promedio es

$$E[T(x)] = \frac{1}{B}\sum_{i=b}^{B} E[T_b(x)]$$

Dado que todos lo árboles se construyen de la misma manera a partir de una muestra L_b extraida de forma independiente, el sesgo de cada árbol T_b es igual al del árbol promedio:

$$E[T(x)] = E[T_b(x)]$$

La varianza del árbol promedio se construye con las varianzas de las muestras L_b :

$$Var[T(x)] = Var\left(\frac{1}{B}\sum_{i=b}^{B} T_b(x)\right) = \frac{1}{B^2}\sum_{i=b}^{B} Var[T_b(x)] = \frac{1}{B}Var[T_b(x)]$$

Dato que las muestras se contruyen de forma independiente la varianza del árbol promedio es menor que la de cualquier otro árbol.

Bagging en Árboles Aleatorios

Bootsrap Aggregation

Sea $L = \{x_i, y_i\}_{\mathrm{i}}^{\mathrm{n}}$ nuestra muestra de entrenamiento de la cual obtuvimos L_B muestras bootsrap

- Para cada muestra L_b contruimos un árbol T_b
- (Regresión) Promediamos los árboles para reducir la varianza

$$\frac{1}{B}\sum_{i=b}^{B} T_b^*(x)$$

- (Clasificación) Tomamos votos sobre todos los árboles

$$T^*(x) = argmax_g\{i|T_b^*(x) = g\}$$

O calculamos los promedios de probabilidades.

Bagging en Árboles Aleatorios

Corolario

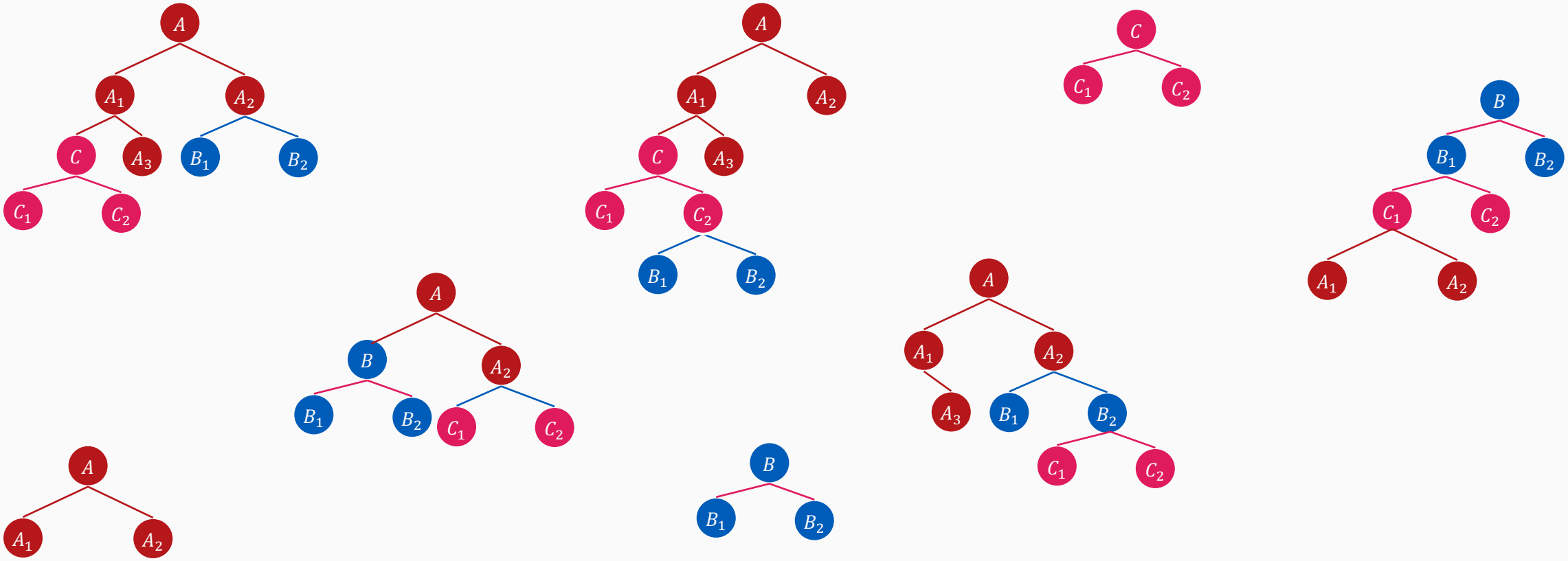
- Nuestro modelo de mejora cuando promediamos muchos árboles.
- La mejora está dada por la correlación entre ellos: cuanto más grande es la correlación, menor beneficio en reducción de varianza obtenemos.
- Podemos alterar el proceso para producir árboles menos correlacionados.
- Sin embargo, estas alteraciones generalmente están acompañadas de incrementos en la varianza.

Un ejemplo

Bosques Aleatorios

Random Forest

Son un conjunto de árboles bagging descorrelacionados



Bosques Aleatorios

Random Forest

Sea m fija en una muestra de variables en L_B muestras bootstrap con reemplazo

Para cada muestra bootstrap L_B construimos un árbol T_B de la siguiente forma:

1. En cada nodo candidato a particionar, escogemos al azar m variables de las disponibles
2. Buscamos la mejor variable y punto de corte *pero solo entre las variables que seleccionamos al azar*.
3. *Seguimos hasta construir un árbol grande*.
4. (Regresión) Promediamos los árboles para reducir la varianza
5. (Clasificación) Tomamos votos sobre todos los árboles

Bosques aleatorios muchas veces reduce el error de predicción gracias a una reducción a veces considerable de varianza.

Bosques Aleatorios

Observaciones

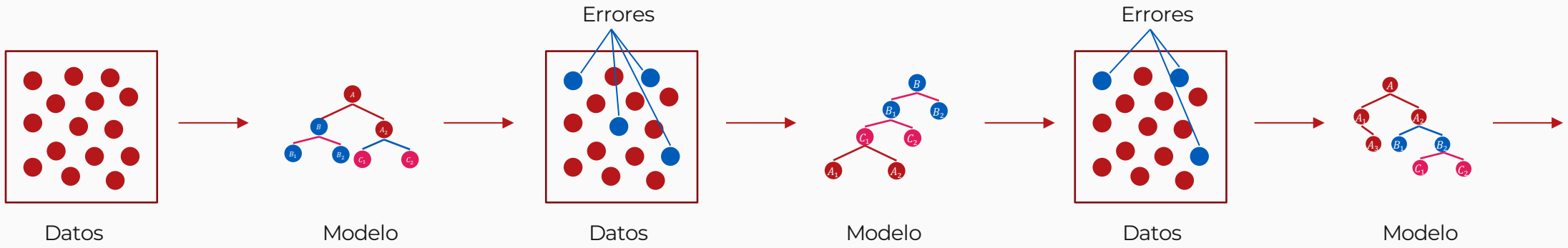
- El número de variables que se seleccionan en cada nodo es un parámetro que hay que escoger.
- Como inducimos aleatoriedad en la construcción de árboles, este proceso reduce la correlación aunque también incrementa su varianza.
- Los bosques aleatorios funcionan bien cuando la mejora en correlación es más grande que la pérdida en varianza.
- Reducir el número de variables :
  - Aumenta el sesgo del bosque (pues es más restringido el proceso de construcción)
  - Disminuye la correlación entre árboles y aumenta la varianza de cada árbol

- Incrementar el número de variables:
  - Disminuye el sesgo del bosque (menos restricción)
  - Aumenta la correlacción entre árobles y disminuye la varianza de cada árbol
  - Cuando usamos bosques aleatorios para estimar probabilidades de clase, como siempre, es necesario checar la calibración de esas probabilidades

Un ejemplo

Boosting

- Se aplica de manera más efectiva a modelos con alto sesgo y baja varianza.
- Agrega nuevos modelos al conjunto de forma secuenciada.
- Resuelve la compensación de sesgo-varianza al comenzar con un modelo débil y secuencialmente aumenta su rendimiento construyendo nuevos modelos.
- Cada nuevo modelo en la secuencia intenta arreglar dónde el anterior cometió los mayores errores



Boosting