



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

# Introducción a la Minería de Datos

Verónica Guarín Escudero  
Escuela de Estadística

Correo: [jvguarine@unal.edu.co](mailto:jvguarine@unal.edu.co)

# Temas

- Introducción
- Estructura de datos FP-Tree
- Paso 1: Construcción del FP-Tree
- Paso 2: Generación de Itemsets Frecuentes
- Discusión



# Frequent Pattern Growth (FP-Growth) Algorithm

# Introducción

- **Apriori**: utiliza un enfoque de generar y probar: genera itemsets candidatos y probar si son frecuentes.

- La generación de itemsets candidatos es costosa (en espacio y tiempo)
- El conteo de soporte es costoso
  - Comprobación de subconjuntos (computacionalmente costosa)
  - Múltiples escaneos de bases de datos (E / S)

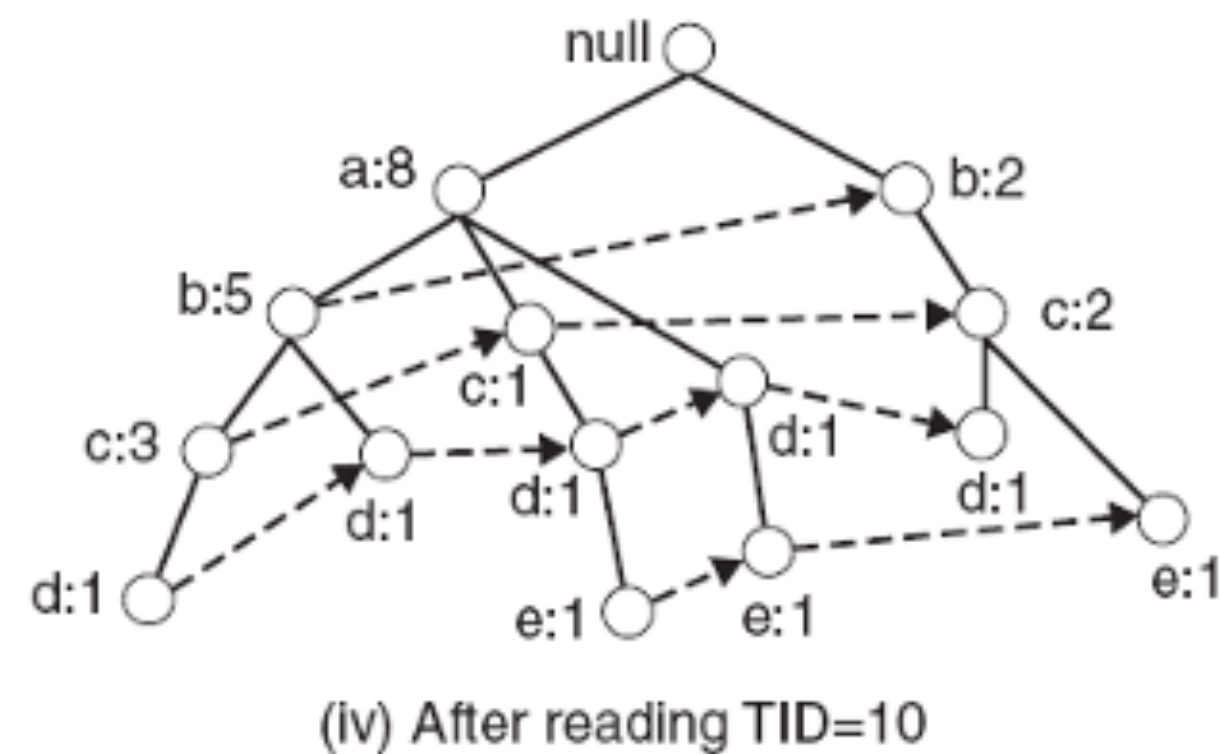
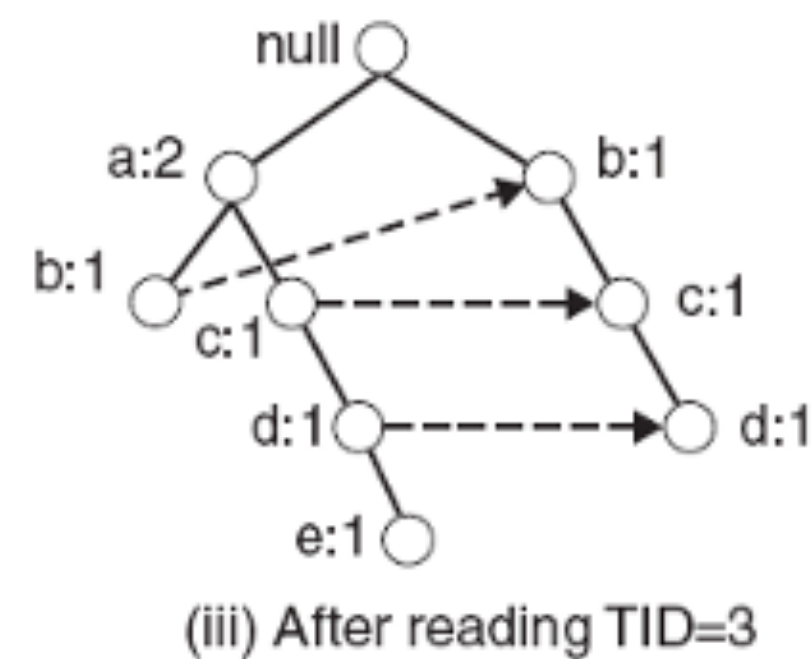
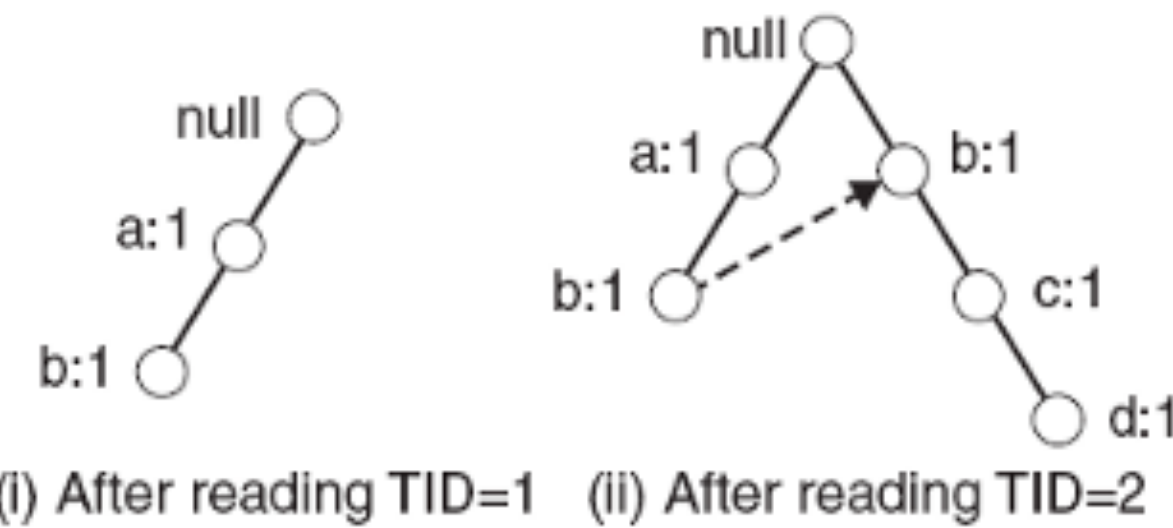
- **FP-Growth**: permite descubrir itemsets frecuentes **sin generación itemsets candidatos**.  
Enfoque de dos pasos:

- **Paso 1**: crea una estructura de datos compacta llamada: **FP-tree**
  - Construido haciendo **2 pasadas** sobre el conjunto de datos.
- **Paso 2**: extrae itemsets frecuentes directamente desde el **FP-tree**
  - A través del recorrido por el FP-Tree

# Core Data Structure: FP-Tree

Transaction Data Set

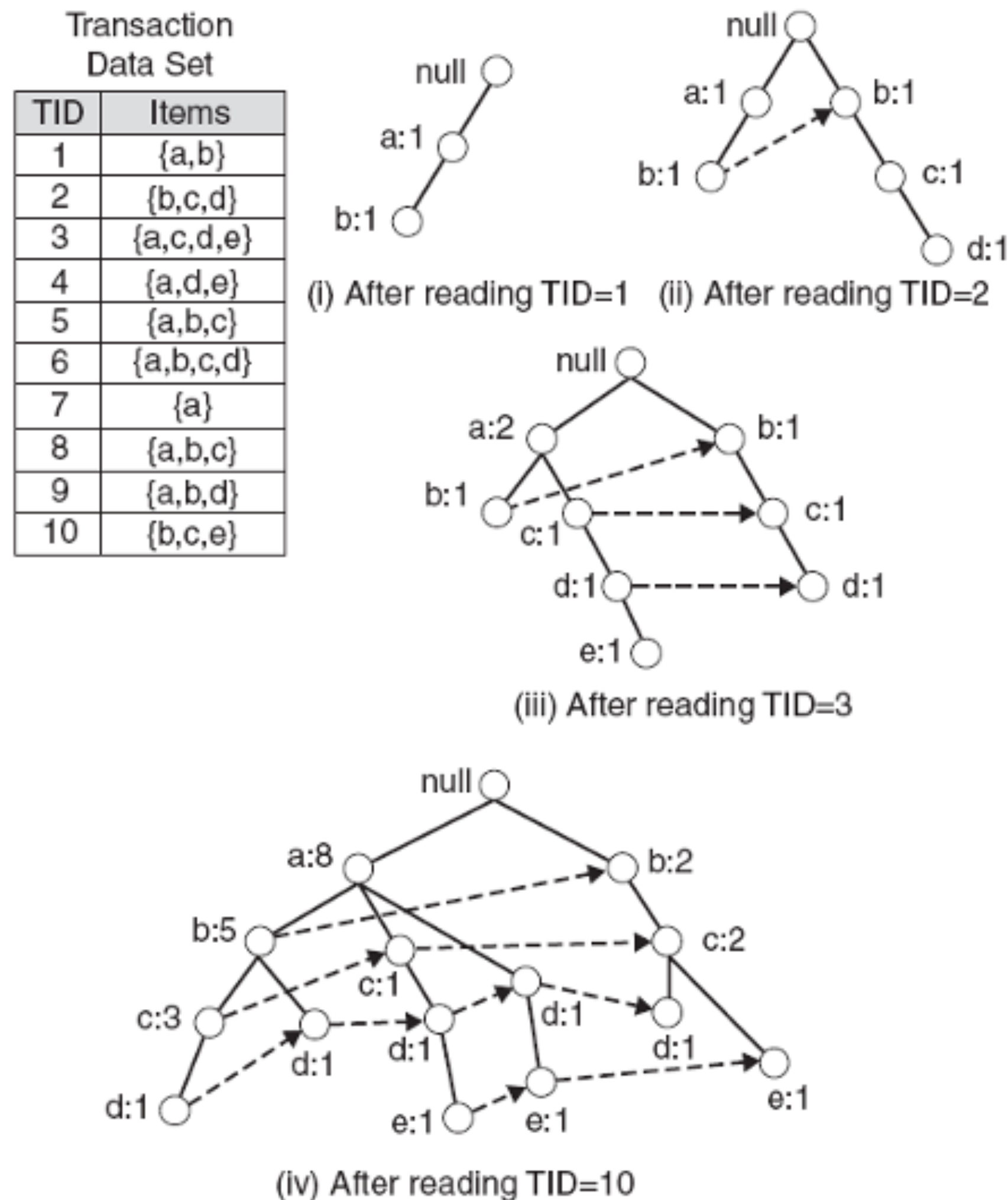
TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}



- Los nodos corresponden a un ítem y tienen un contador.
- **FP-Growth lee 1 transacción a la vez** y la asigna a una ruta.
- Se usa el orden fijo, por lo que las rutas se pueden superponer cuando las transacciones comparten elementos (cuando tienen el **mismo prefijo**).
- En este caso, los contadores se incrementan.
- Los punteros se mantienen entre los nodos que contienen el mismo elemento, creando listas unidas (líneas punteadas).
- Cuantas más rutas se solapan, mayor será la compresión. FP-tree puede caber en la memoria.
- Itemsets frecuentes son extraídos del FP-Tree.



# Paso 1: Construcción del FP-Tree (Ejemplo)



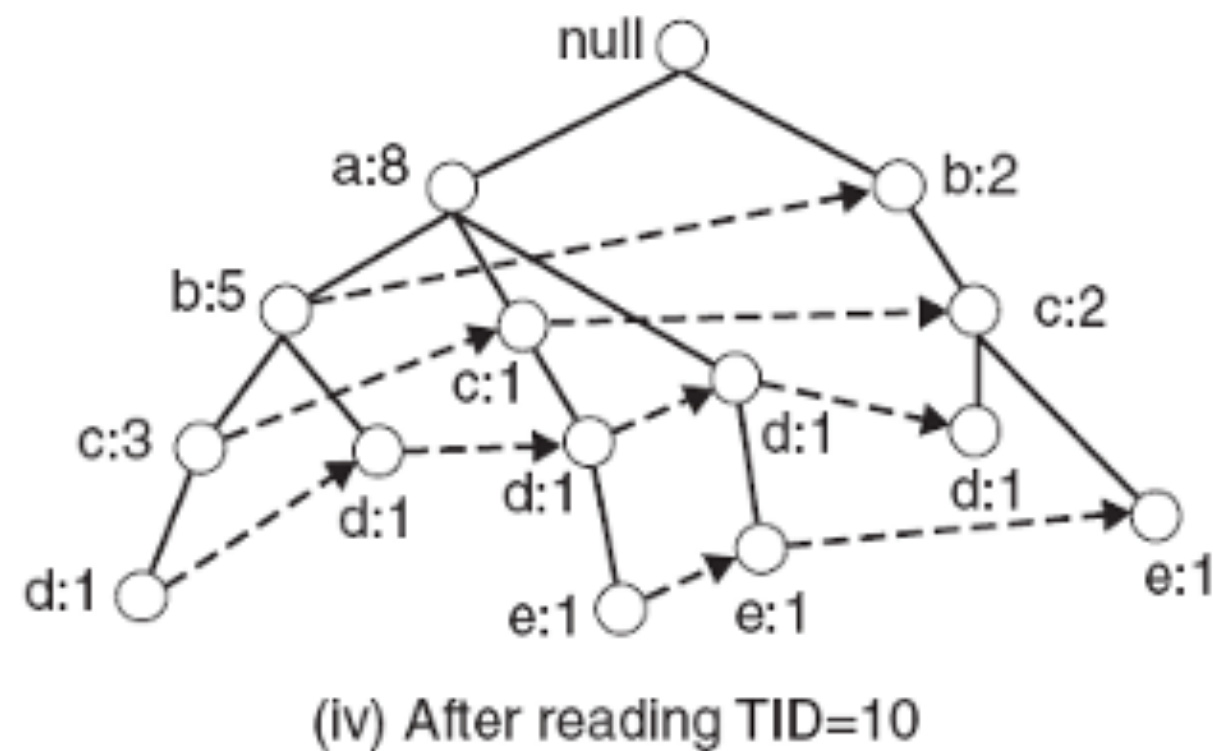
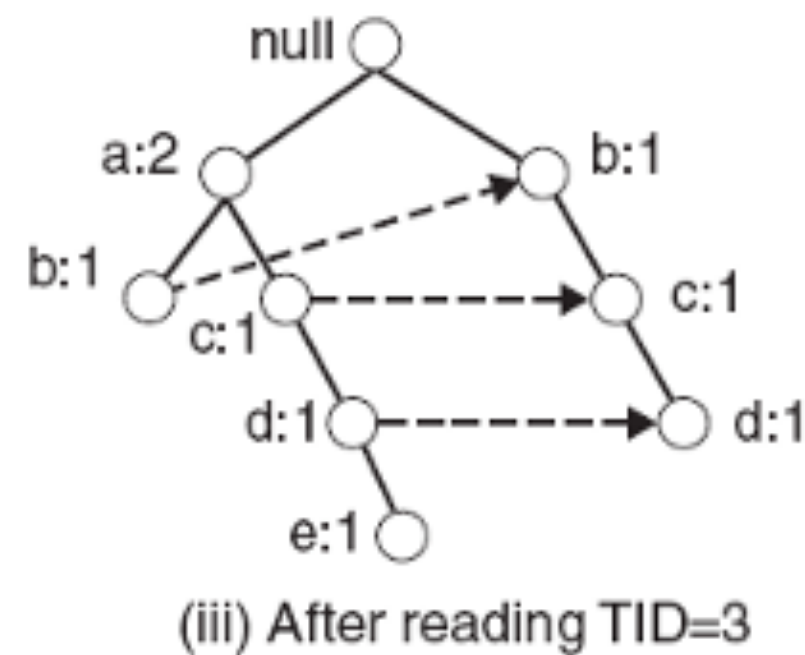
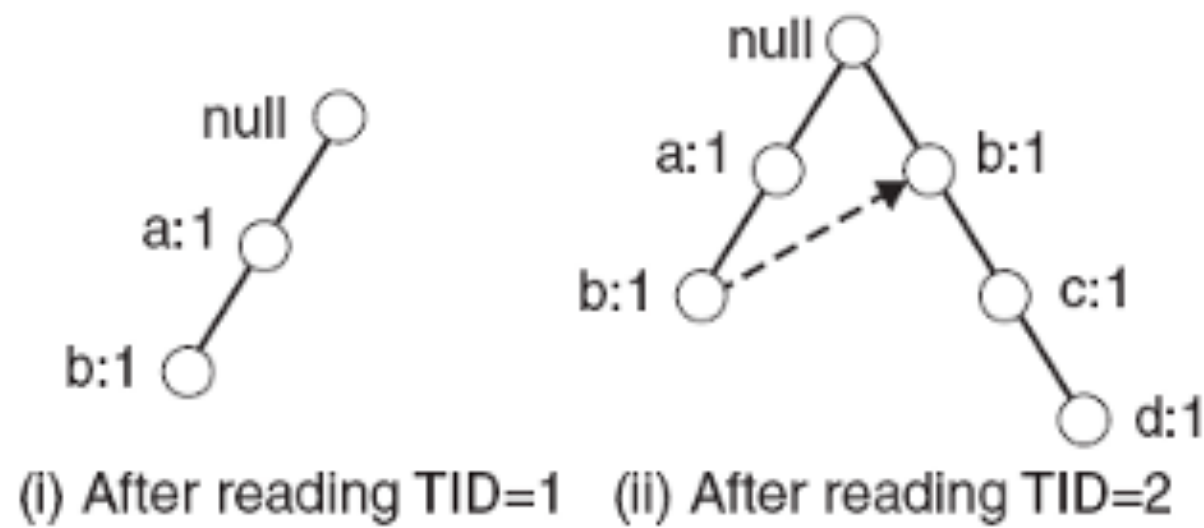
FP-Tree se construye usando 2 pasadas sobre el conjunto de datos:

## Pasada 1:

- Escanear los datos y calcular el soporte para cada ítem.
- Descartar ítems poco frecuentes.
- Clasificar los ítems frecuentes en orden decreciente en función del soporte.
  - Para el ejemplo: a, b, c, d, e
  - Usar este orden cuando construya el **FP-Tree**, para que los prefijos comunes puedan ser compartidos.

# Paso 1: Construcción del FP-Tree (Ejemplo)

Transaction Data Set	
TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}



**Pasada 2:** construir el FP-Tree.

- Leer transacción 1: {a, b}

- Crea 2 nodos a y b y la ruta **null** → a → b. Asignar los recuentos de a y b con 1.

- Lea la transacción 2: {b, c, d}

- Crea 3 nodos para b, c y d y la ruta **null** → b → c → d. Incrementar los nodos en 1.

- Tenga en cuenta que aunque las transacciones 1 y 2 comparten **b**, **las rutas son disjuntas** ya que no comparten un prefijo común. Agrega el enlace entre las b.

- Lea la transacción 3: {a, c, d, e}

- Comparte ítem de prefijo común **a** con la transacción 1, por lo que la ruta de las transacciones 1 y 3 se superpondrán y el recuento de frecuencias del nodo **a** se incrementará en 1.

- Agregue enlaces entre las **c** y las **d**.

- Continúe hasta que todas las transacciones estén asignadas a una ruta en el **FP-Tree**.

# Tamaño del FP-Tree

El FP-Tree generalmente tiene un tamaño más pequeño que los datos sin comprimir, por lo general, muchas transacciones comparten elementos (y, por lo tanto, prefijos).

- **El mejor de los casos:** todas las transacciones contienen el mismo conjunto de ítems.

- 1 ruta en el FP-Tree

- **El peor escenario posible:** cada transacción tiene un conjunto único de ítems (sin elementos en común)

- El tamaño del FP-Tree es al menos tan grande como los datos originales.

- Los requisitos de RAM para el FP-Tree son más altos. Necesita almacenar los punteros entre los nodos y los contadores.

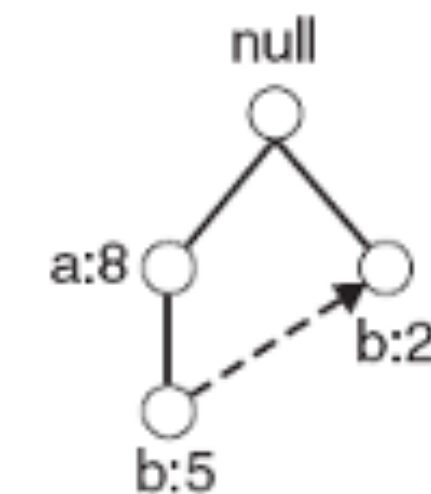
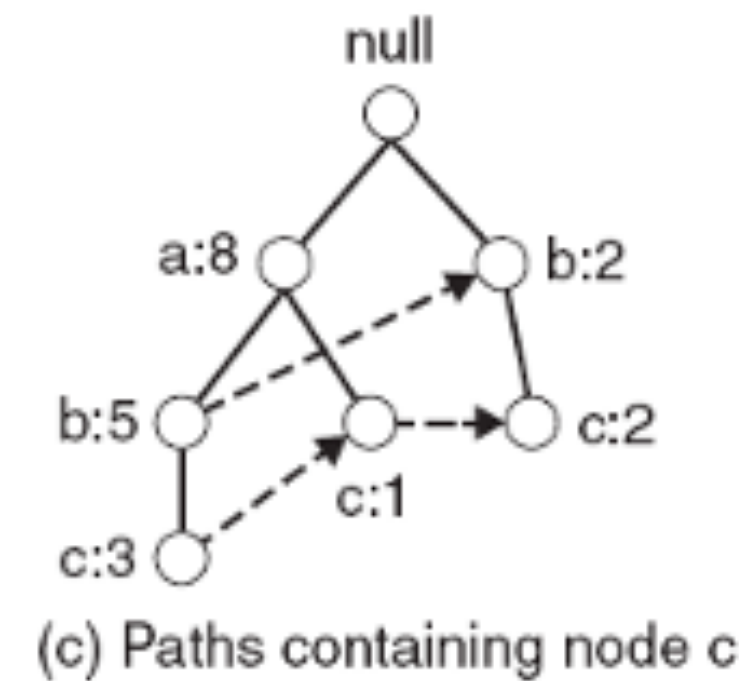
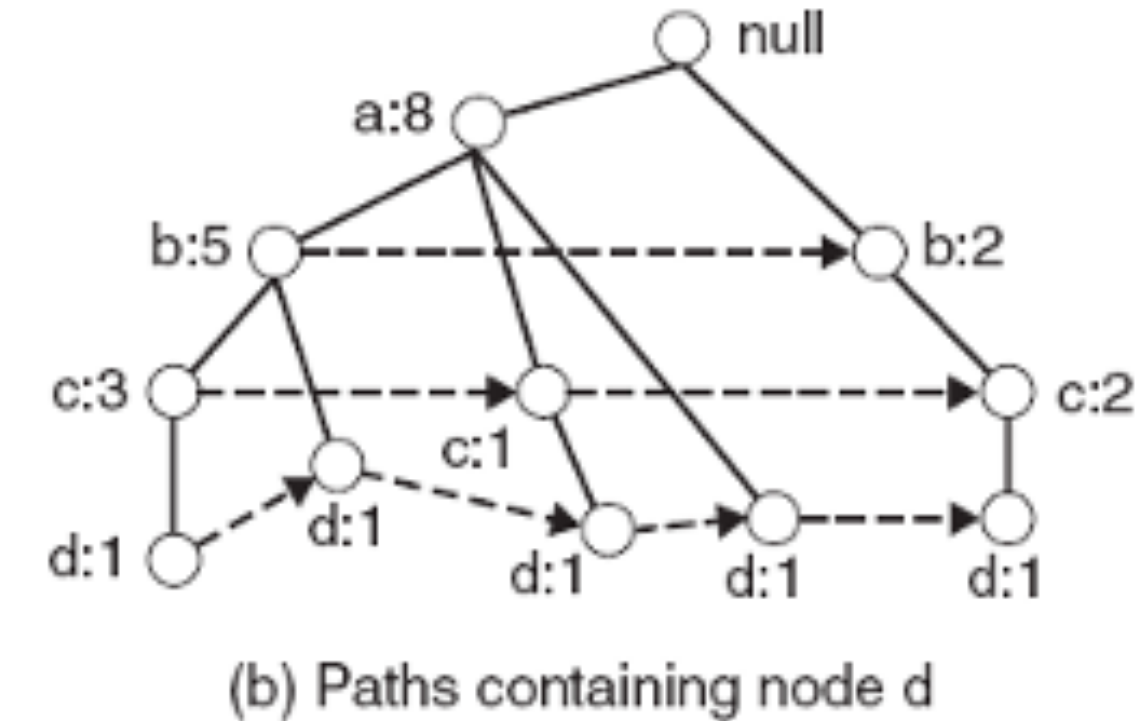
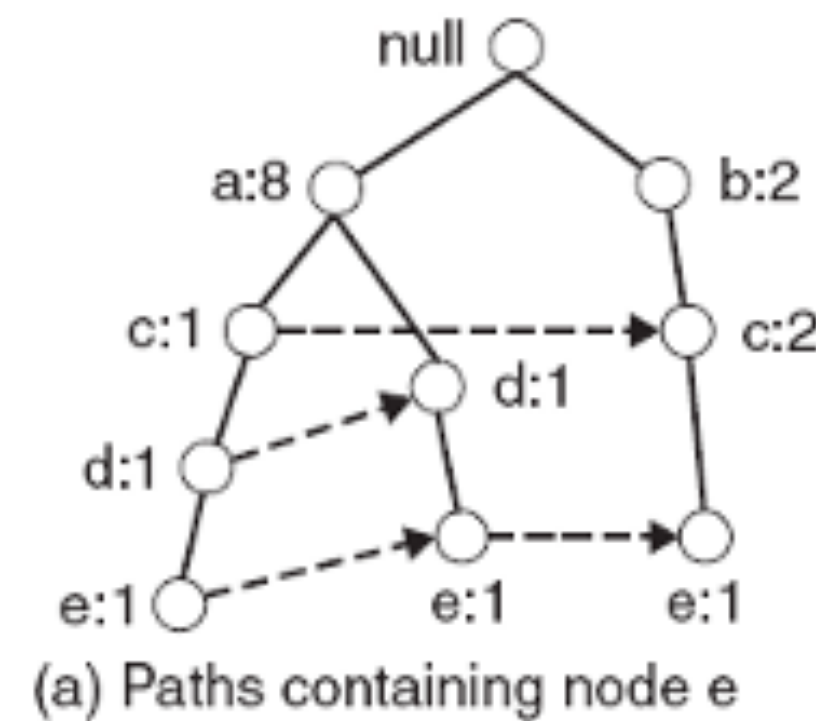
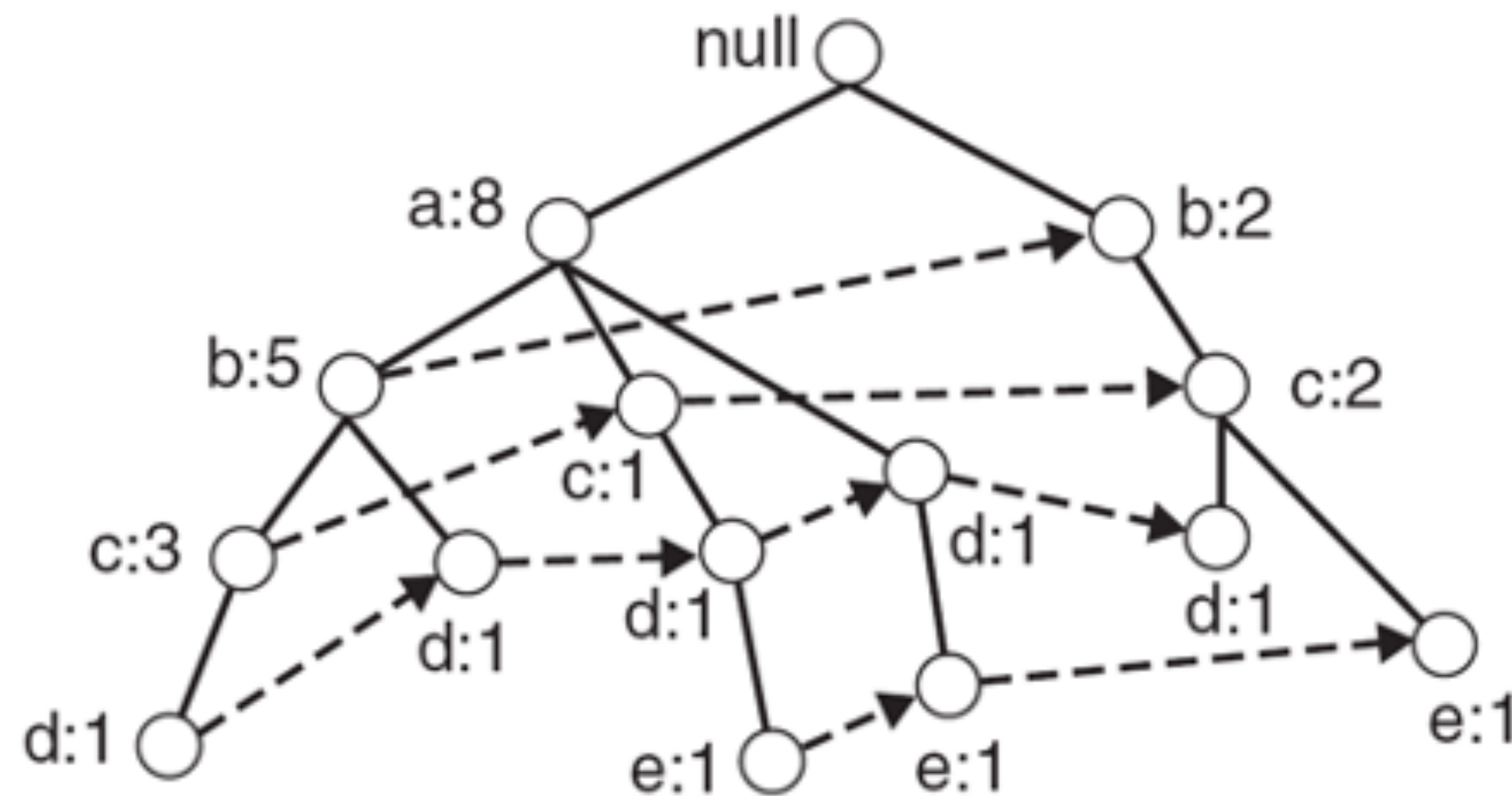
- El tamaño del FP-Tree depende de cómo se ordenan los ítems

- El orden de disminución de soporte se usa normalmente, pero no siempre lleva al árbol más pequeño (es una heurística).



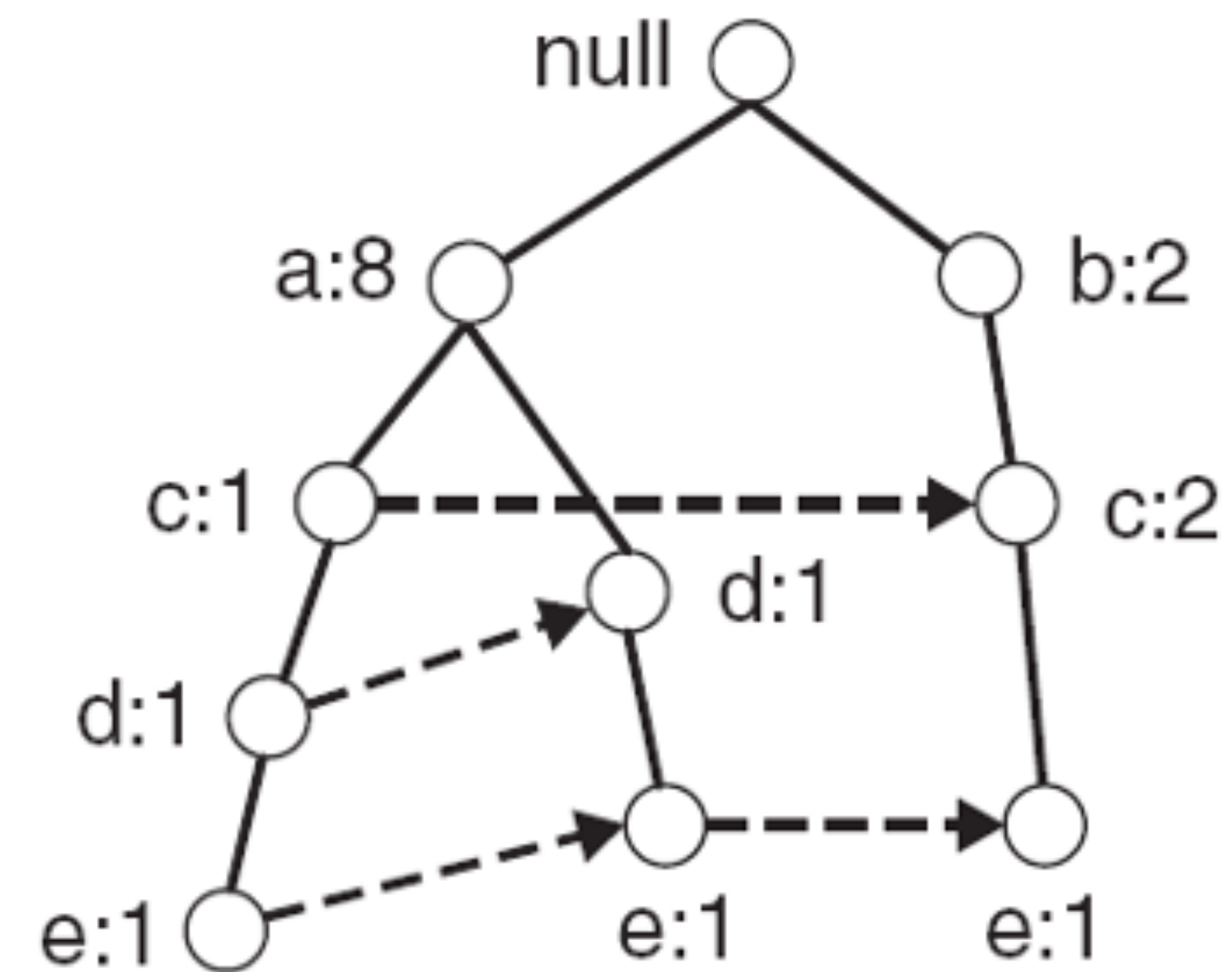
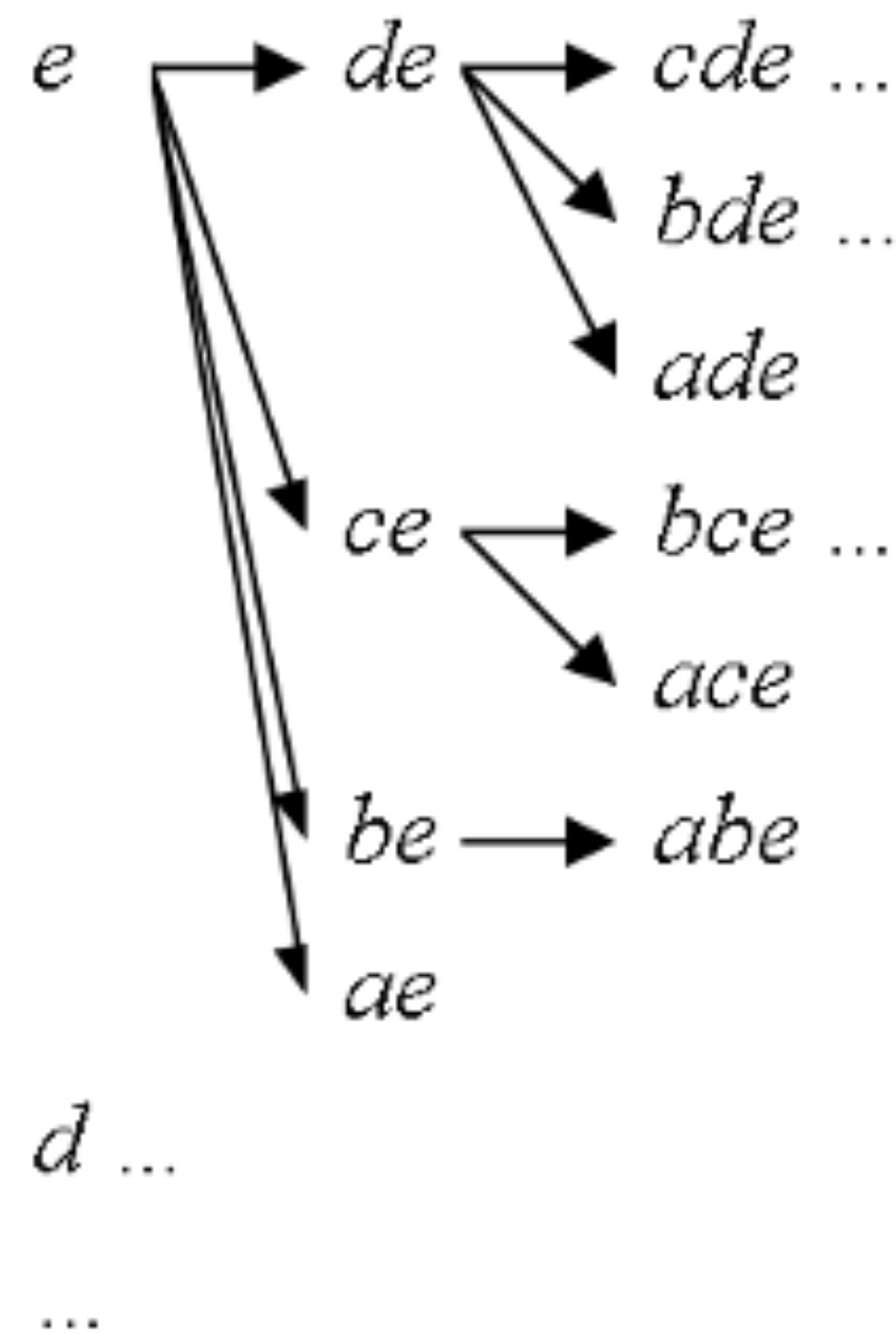
# Paso 2: Generación Itemsets Frecuentes

- **FP-Growth** extrae conjuntos de elementos frecuentes del FP-tree.
- Algoritmo Bottom-Up desde las hojas hacia la raíz.
  - **Divide and Conquer**: primero busca conjuntos de ítems frecuentes que terminen en **e**, luego **de**, etc. . . entonces **d**, luego **cd**, etc. .
  - Primero, extrae los subárboles de un prefix path que terminan en un ítem o ítemset.



# Paso 2: Generación Itemsets Frecuentes

- Cada subárbol de ruta de prefijo se procesa recursivamente para extraer los conjuntos de elementos frecuentes. Las soluciones se fusionan.
  - Por ejemplo: El subárbol con la ruta del prefijo para **e** se usará para extraer conjuntos de elementos frecuentes que terminan en **e**, luego en **de**, **ce**, **be** y **ae**, luego en **cde**, **bde**, **cde**, etc.



Prefix path sub-tree ending in **e**

# Condicional FP-Tree

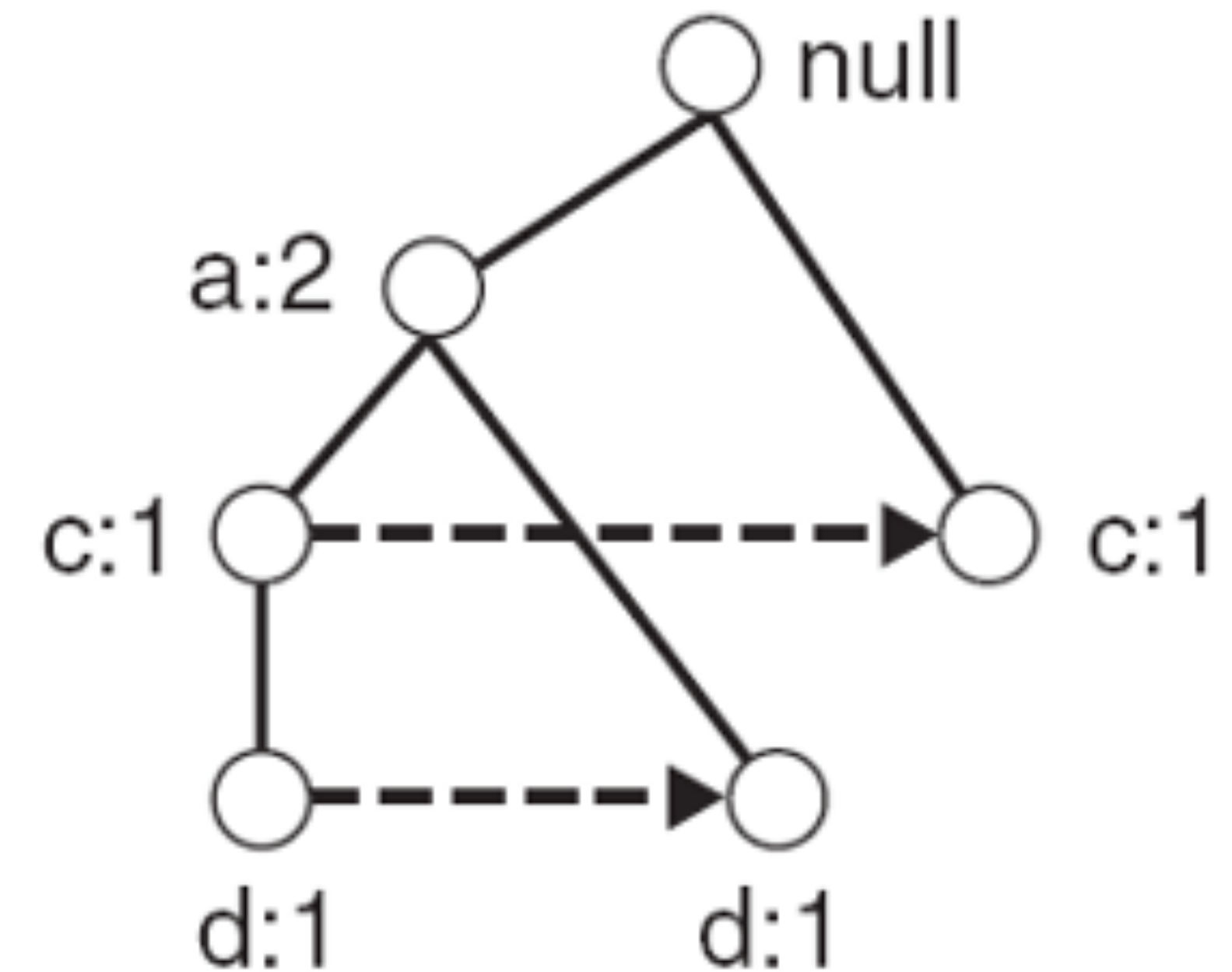
Como  $\{e\}$  es frecuente, el algoritmo debe resolver los subproblemas de encontrar conjuntos de elementos frecuentes que terminan en de, ce, be y ae.

- Es el FP-Tree que se construiría si solo consideramos las transacciones que contienen un itemset particular.

- Todos los demás son desestimados.

- FP-Tree condicional de **e**

TID	Items
<del>1</del>	<del>{a,b}</del>
<del>2</del>	<del>{b,c,d}</del>
3	{a,c,d, <del>e</del> }
4	{a,d, <del>e</del> }
<del>5</del>	<del>{a,b,e}</del>
<del>6</del>	<del>{a,b,c,d}</del>
<del>7</del>	<del>{a}</del>
<del>8</del>	<del>{a,b,c}</del>
<del>9</del>	<del>{a,b,d}</del>
10	{b,c, <del>e</del> }





# Bibliografía

- ▣ Han, J., Pei, J., & Yin, Y. (2000, May). Mining frequent patterns without candidate generation. In ACM sigmod record (Vol. 29, No. 2, pp. 1-12). ACM. [[pdf](#)]
- ▣ Tan, P. N. (2006). Introduction to data mining. Pearson Education India. [[pdf](#)]







UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

# Gracias