

Módulo 7

Aprendizaje de máquina no supervisado 2.2 Reglas de asociación, Algoritmo *Apriori*

Eduardo Espinosa Avila



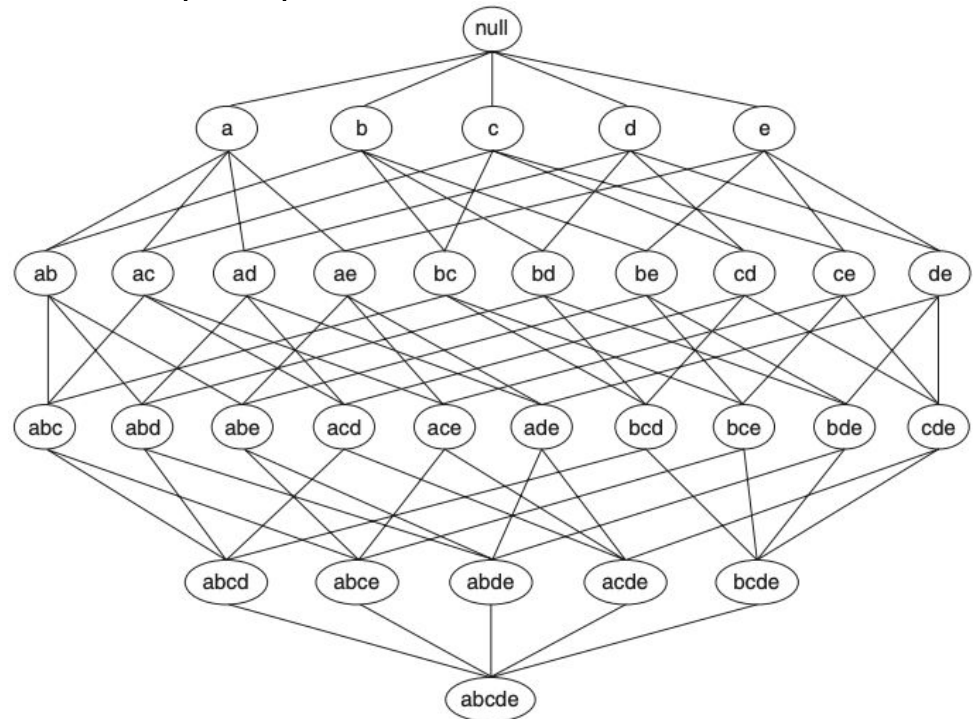
Reglas de asociación, algoritmo *Apriori*

Como mencionamos, la generación de reglas suele dividirse en:

- **Generación del *Frequent Itemset* (FIG)**
- **Generación de reglas**

Sin embargo, la FIG puede, en general, crecer de forma exponencial.

Se deben buscar alternativas eficientes.



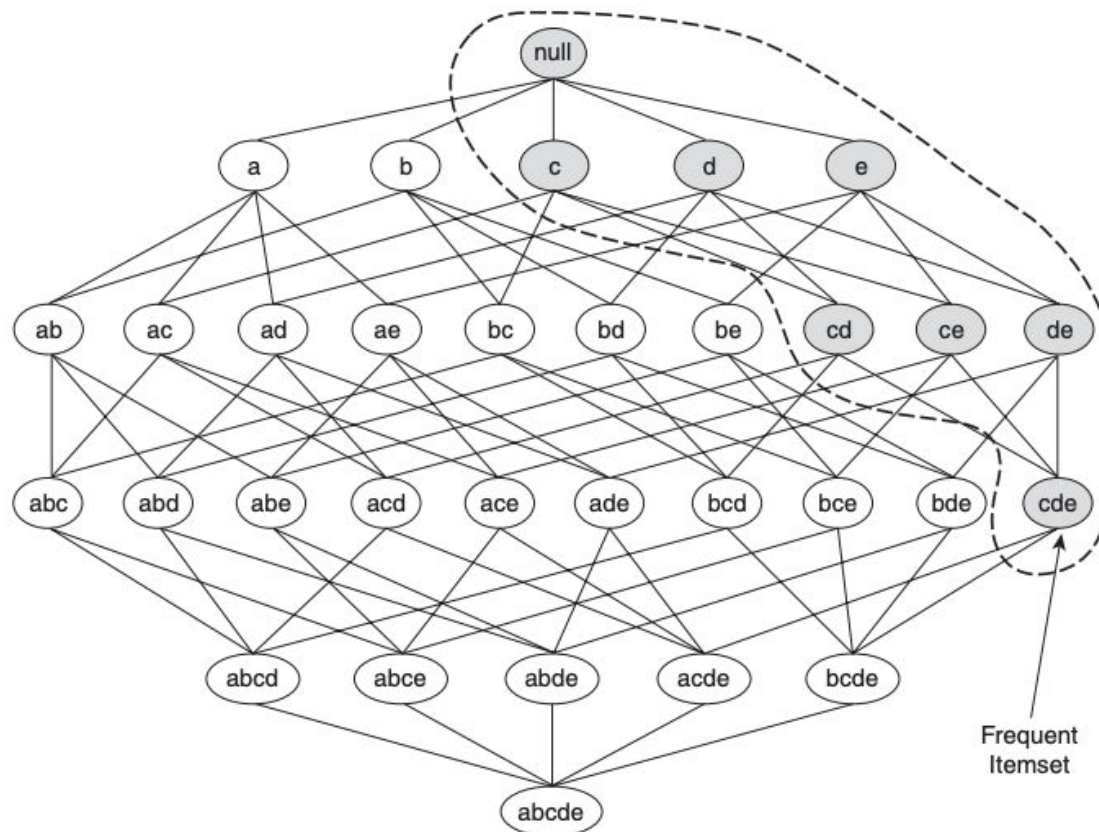
Algoritmo *Apriori*, el principio *Apriori*

Existen varias maneras de reducir la complejidad de la FIG. Las más usadas son:

- **Reducir el número de *itemsets* candidatos.** El principio *Apriori* es una forma efectiva de eliminar algunos conjuntos candidatos, sin la necesidad de contar sus elementos.
- **Reducir el número de comparaciones.** En lugar de comparar cada conjunto con cada transacción, es posible reducir su número, usando alguna estructura de datos avanzada (árboles).

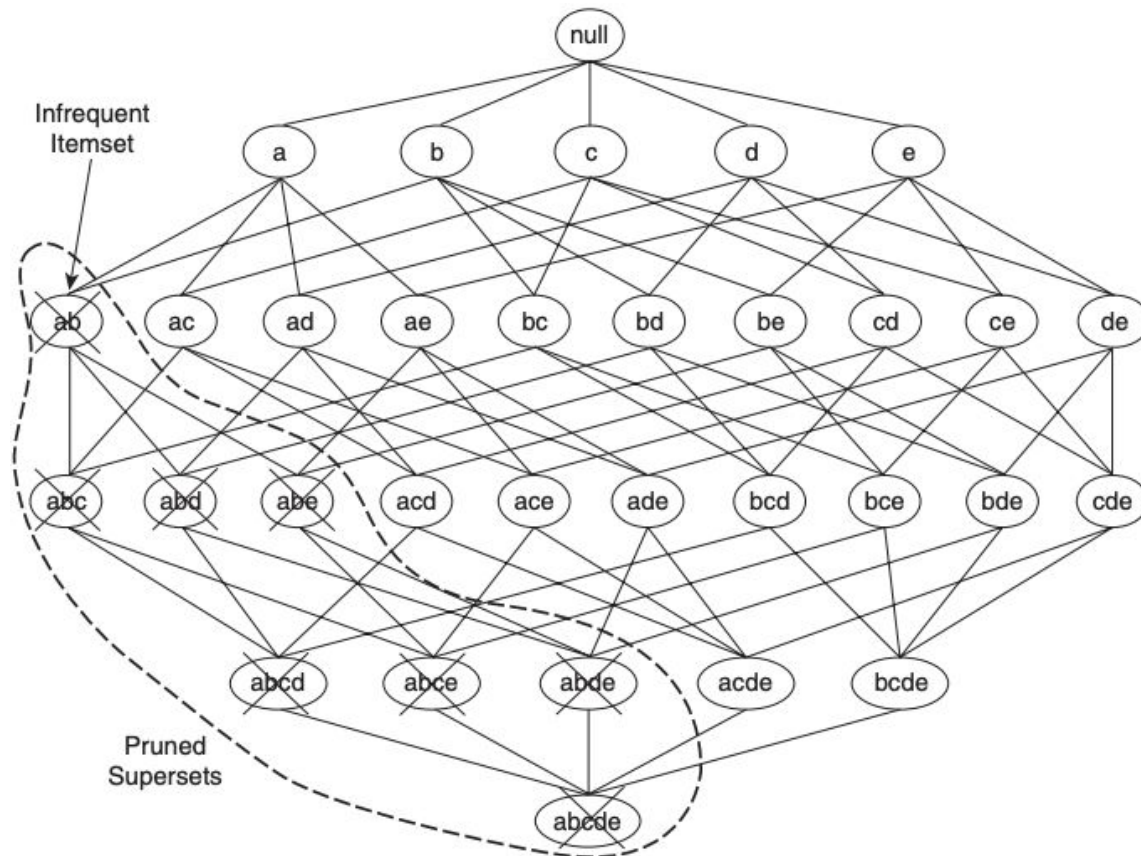
Algoritmo *Apriori*, el principio *Apriori*

Este principio establece que *si un itemset es frecuente, entonces todos sus subconjuntos también deben ser frecuentes*.



Algoritmo *Apriori*, el principio *Apriori*

A la inversa, si un itemset es infrecuente, entonces todos sus superconjuntos también deben ser infrecuentes.



Algoritmo *Apriori*, el principio *Apriori*

Con esto, el algoritmo *Apriori* es el siguiente:

F_k : *k-itemsets* frecuentes

L_k : *k-itemsets* candidatos

$k \leftarrow 1$

Generar $F_k = \{1\text{-itemsets frecuentes}\}$

Repetir hasta que F_k esté vacío

Generar candidatos: generar L_{k+1} a partir de F_k

Podar candidatos: podar los candidatos de L_{k+1} que contengan subconjuntos de longitud k que sean infrecuentes

Conteo de *support*: contar el número de ocurrencias de cada candidato, a partir de las filas del conjunto completo.

Eliminación de candidatos: eliminar los candidatos de L_{k+1} que sean infrecuentes, dejando solamente a los frecuentes: F_{k+1}

Algoritmo *Apriori*, aplicaciones

Una vez que se han extraído las reglas, se pueden utilizar para varias tareas.

- Recomendaciones de compras
 - Productos afines
- Colocación de productos
- Detección de fraudes
- Analizar datos de encuestas
 - Gracias a que, por su naturaleza, procesa datos categóricos



<https://twitter.com/lisachwinter/status/494063906625961984>

Referencias

- Tan, Pang-Ning, Steinbach, Michael, Karpatne, Anuj and Kumar, Vipin
Introduction to Data Mining (Second Edition)
/ Pang-Ning Tan, Michael Steinbach, Anuj Karpatne and Vipin Kumar --
Birmingham - Mumbai : Pearson, 2019
<https://www-users.cs.umn.edu/~kumar001/dmbook/index.php>
- Rakesh Agrawal, Tomasz Imieliński and Arun Swami
Mining association rules between sets of items in large databases
<https://dl.acm.org/doi/10.1145/170036.170072>
- Michael Hahsler, Bettina Grün, Kurt Hornik and Christian Buchta
Introduction to arules – A computational environment for mining association rules and frequent item sets
<https://mran.revolutionanalytics.com/web/packages/arules/vignettes/arules.pdf>

Contacto

Dr. Eduardo Espinosa Avila

laloea@fisica.unam.mx

Tels: 5556225000 ext. 5003

Redes sociales:

<https://twitter.com/laloea>

<https://www.linkedin.com/in/eduardo-espinosa-avila-84b95914a/>