Árbol de expansión mínima (o máxima). Algoritmo de Prim

Asignatura: **Investigación Operativa II**, 2° Grado en Estadística

Autor: Miguel Rodríguez Rosa

Se dice que un grafo es un **árbol** si se puede encontrar un camino no dirigido (será entonces un **grafo no dirigido**) entre cualquier par de nodos y no contiene ciclos, es decir, entre cada par de nodos existe un único camino.

Dado un grafo no dirigido cualquiera, encontrar un árbol **de expansión** en él consiste en encontrar la mínima red de nodos y arcos que lo recubran, entendiendo mínima en el siguiente sentido:

- Dados dos nodos cualesquiera del grafo, en la red existe un único camino simple entre ellos (la red tiene que ser un árbol).
- Al eliminar un arco cualquiera de la red, deja de existir un camino entre al menos un par de nodos (la red dejaría de ser un árbol).
- Al añadir un arco cualquiera más a la red, se genera un ciclo redundante, es decir, entre al menos un par de nodos existen dos caminos distintos (la red dejaría de ser un árbol).
- Si el grafo tiene n nodos, el número de arcos de la red es $n - 1$.

El problema del árbol de expansión mínima (máxima) en **grafos ponderados** consiste en encontrar un árbol de expansión cuya longitud, o sea, la suma de los pesos de sus arcos, sea mínima (máxima).

Existen diversos algoritmos de búsqueda de árboles de expansión mínima (máxima). A continuación se presenta uno de los algoritmos más usuales para realizar esta búsqueda.

1. Algoritmo de Prim

Durante el proceso, los arcos que formen el árbol de expansión se marcan con etiquetas permanentes, se busca cuál de los arcos no marcados es la mejor opción para ampliar el árbol sin formar ciclos, y se repite hasta llegar a una iteración en la que todos los nodos ya estén conectados.

Se describe aquí el algoritmo de Prim para encontrar el árbol de expansión **mínima**, si se buscase el de expansión máxima solamente sería sustituir en el algoritmo todos los cálculos de mínimos por máximos.

- Se parte de la matriz de adyacencia del grafo, es decir, una matriz con tantas filas y columnas como nodos (n), en la que el elemento en la fila $i-$ésima y columna $j-$ésima sea el peso del arco que va del nodo i al nodo j, c_{ij} (los elementos c_{ii} en la diagonal se dejan vacíos).

- Iteración $k = 0$:
  - Se busca un par de nodos cuyo arco tenga el mínimo peso de toda la matriz de adyacencia. Sean i y j los nodos unidos por ese arco. Se marca ese arco con una etiqueta permanente, c^*_{ij}. Si dicho arco no es único, se elige uno cualquiera, existirán distintas soluciones al problema.
  - Se rodean las filas $i-$ y $j-$ésima de la matriz de adyacencia.
  - Se tachan las columnas $i-$ y $j-$ésima de la matriz de adyacencia.
- Iteración $k = 1, 2, \ldots$:
  - Se busca un par de nodos cuyo arco tenga el mínimo peso, no de toda la matriz de adyacencia, sino solamente de entre los elementos que estén rodeados y no tachados. Sean i y j los nodos unidos por ese arco. Se marca ese arco con una etiqueta permanente, c^*_{ij}. Si dicho arco no es único, se elige uno cualquiera, existirán distintas soluciones al problema.
  - Se rodea la fila $j-$ésima de la matriz de adyacencia (la fila $i-$ésima ya estará rodeada).
  - Se tacha la columna $j-$ésima de la matriz de adyacencia (la columna $i-$ésima ya estará tachada).
- Se repite el paso anterior hasta que todos los elementos de la matriz de adyacencia estén tachados, obteniendo así:
  - El árbol de expansión mínima, como el formado por todos los arcos con etiquetas permanentes.
  - La longitud del árbol de expansión mínima, como $\sum_{i-j} c^*_{ij}$, es decir, la suma de los pesos de todos los arcos con etiquetas permanentes.