



# EVITAR EL ACOSO CALLEJERO CON EL USO DE LA TECNOLOGIA

# Presentación del equipo



**Aiverson  
Castaño**  
Investigación  
de algoritmos



**Juan Pablo  
Forero**  
Desarrollo del  
código



**Andrea Serna**  
Revisión de  
la literatura

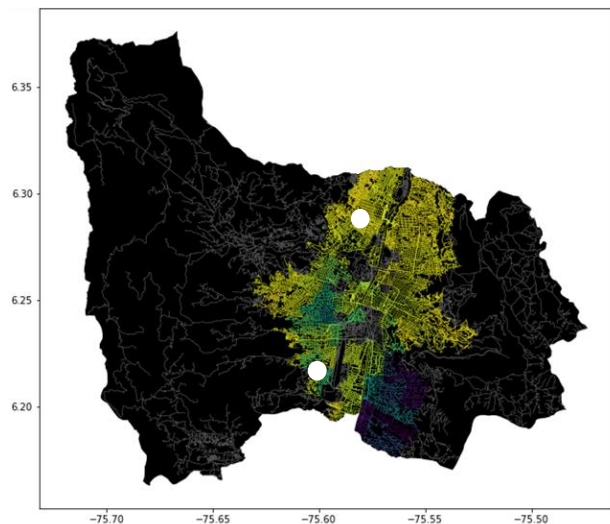


**Mauricio Toro**  
Preparación  
de los datos

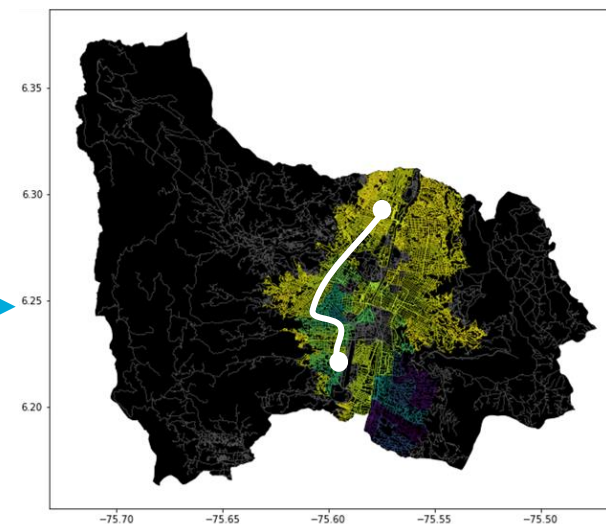
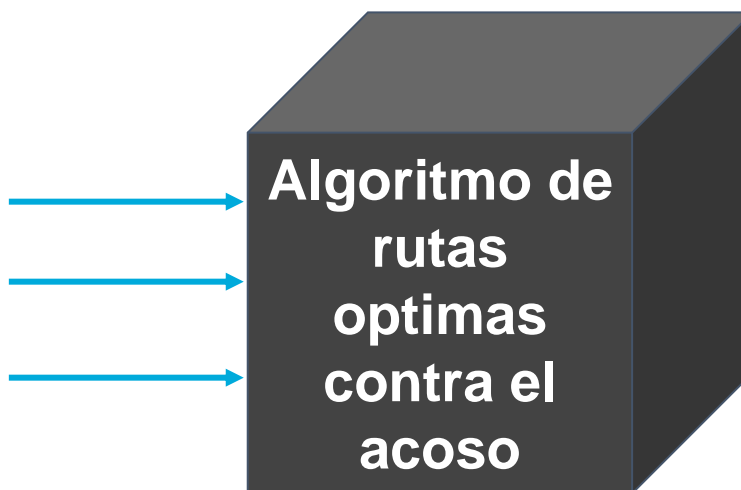


<http://github.com/jpforero/ST0245-002>

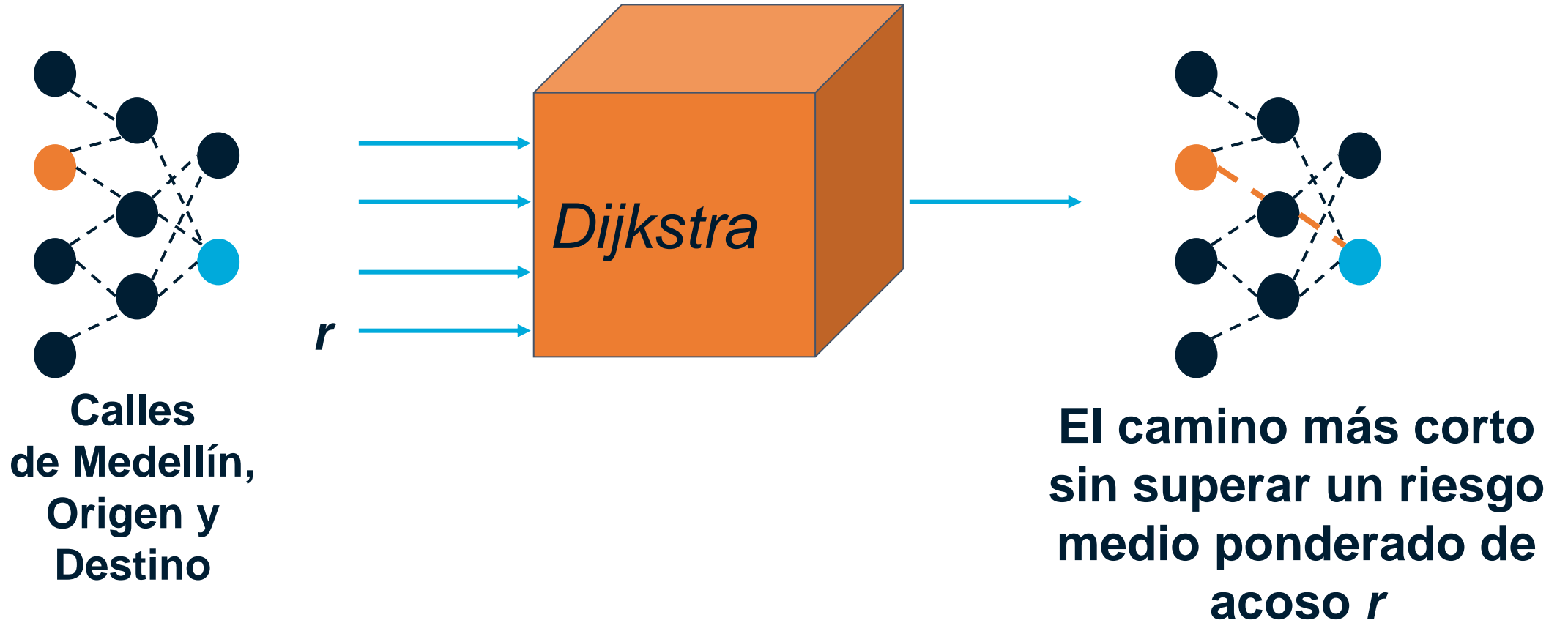
# Planteamiento del problema



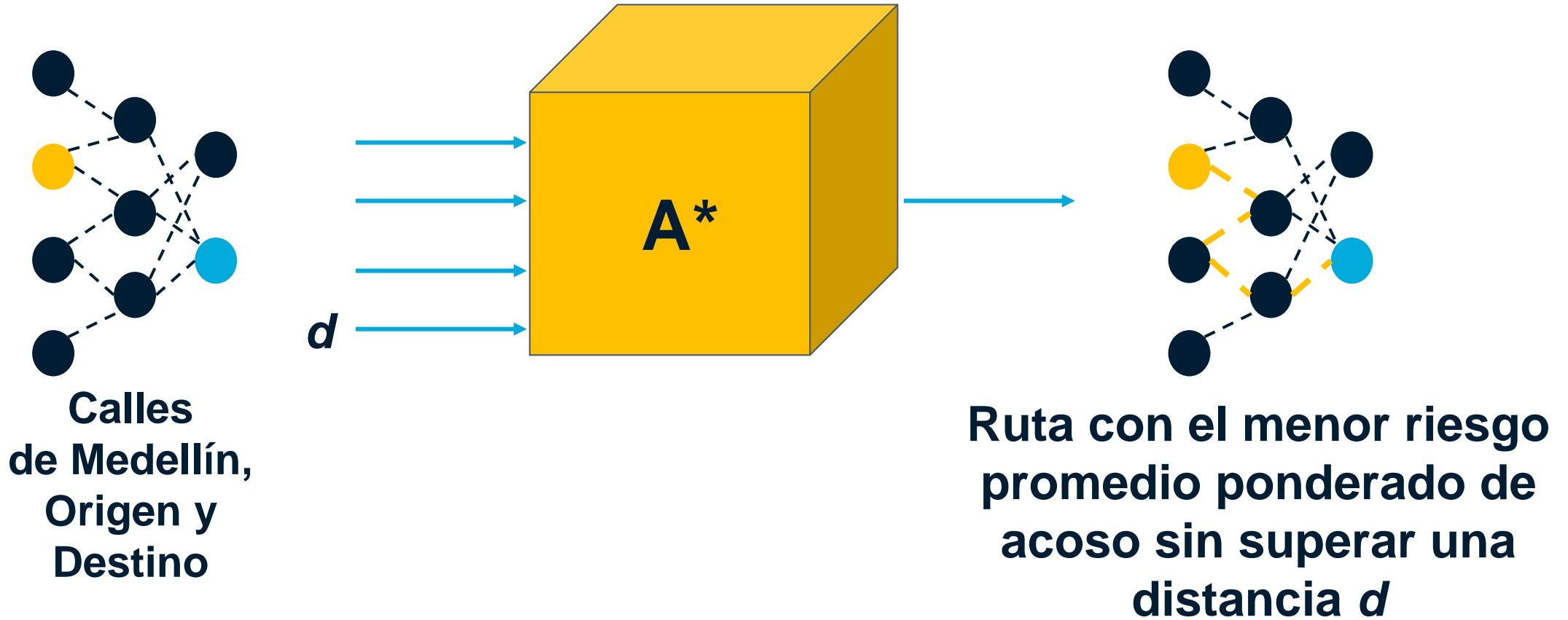
**Calles  
de Medellín,  
Origen y  
Destino**



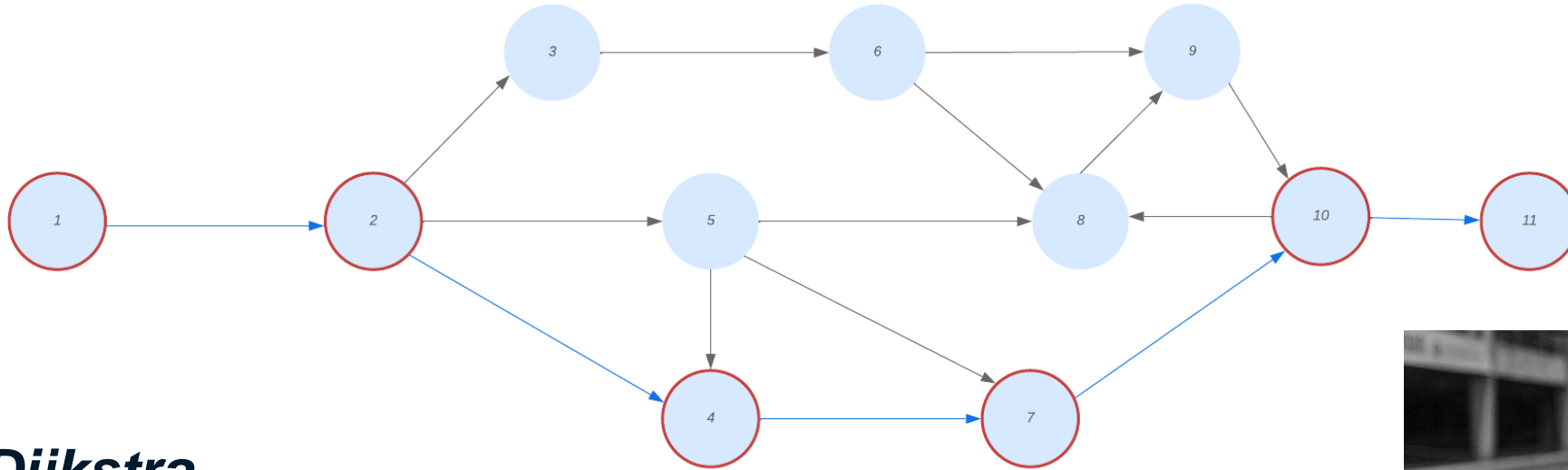
**El más camino más corto  
restringido**



## Segundo algoritmo



# Explicación del algoritmo



## *Dijkstra*

A partir de las listas de adyacencia y la representación de estas en grafos, se puede hacer una idea de como están distribuidas las calles.

Esto facilita el entender como el algoritmo calcula la ruta mas corta restringida.





	Complejidad temporal	Complejidad de la memoria
Dijkstra	$O(n^2)$	$O(E + V \log V)$

**Complejidad en tiempo:** Es el tiempo que puede tardar el algoritmo en dar un resultado.

**Complejidad en memoria:** Es la cantidad de espacio en memoria que utiliza el Algoritmo al ejecutarse.

V representa los vértices y E representa las aristas

