

El algoritmo de **PageRank** en GraphX es una implementación del famoso algoritmo utilizado originalmente por Google para **medir la importancia o autoridad de cada vértice (nodo)** dentro de un grafo. Su funcionamiento se basa en la idea de que los nodos importantes son aquellos que reciben muchos enlaces desde otros nodos importantes.

Aquí te explico los aspectos clave de cómo funciona:

1. Idea Fundamental:

- Un nodo es importante si muchos nodos enlazan hacia él.
- La importancia de un nodo que enlaza a otro se distribuye entre todos sus enlaces salientes. Es decir, un enlace desde un nodo muy importante tiene más peso que un enlace desde un nodo poco importante.

2. Proceso Iterativo:

El algoritmo de PageRank en GraphX opera de forma iterativa, actualizando la puntuación de cada nodo en cada iteración hasta que las puntuaciones convergen a un valor estable o se alcanza un número máximo de iteraciones. Cada iteración consta básicamente de dos fases:

- **Envío de Contribuciones:** Cada nodo envía una fracción de su puntuación actual a sus nodos vecinos a través de sus aristas salientes. La fracción enviada es su puntuación dividida por el número de sus aristas salientes.
- **Actualización de Puntuaciones:** Cada nodo actualiza su puntuación basándose en las puntuaciones que recibe de sus vecinos entrantes. La nueva puntuación de un nodo se calcula sumando las contribuciones recibidas de sus vecinos y aplicando un **factor de amortiguamiento (damping factor)**.

3. Factor de Amortiguamiento (Damping Factor):

- El factor de amortiguamiento (normalmente denotado como α o d , y con un valor típico de 0.85) representa la probabilidad de que un "navegador aleatorio" continúe siguiendo enlaces en lugar de saltar a una página aleatoria.
- Esto ayuda a evitar problemas como los "sink nodes" (nodos sin enlaces salientes) que podrían acumular toda la importancia.
- La fórmula básica para actualizar el PageRank de un nodo u en la iteración $k+1$ es:

$$PR(u)(k+1) = (1-d) + d \sum_{v \in B(u)} \frac{1}{L(v)} PR(v)(k)$$

Donde:

- $PR(u)(k+1)$ es el PageRank del nodo u en la iteración $k+1$.
- d es el factor de amortiguamiento.
- $B(u)$ es el conjunto de nodos que enlazan a u .
- $PR(v)(k)$ es el PageRank del nodo v en la iteración k .
- $L(v)$ es el número de enlaces salientes del nodo v .

4. Inicialización:

- Al comienzo del algoritmo, a cada nodo se le asigna una puntuación inicial, que generalmente es la misma para todos los nodos (por ejemplo, $1/N$, donde N es el número total de nodos).

5. Convergencia:

- El algoritmo continúa iterando hasta que el cambio en las puntuaciones de PageRank entre iteraciones consecutivas es menor que un umbral predefinido (tolerancia) o se alcanza un número máximo de iteraciones. Cuando el algoritmo converge, las puntuaciones de PageRank reflejan la importancia relativa de cada nodo en el grafo.

En resumen, el algoritmo de PageRank en GraphX:

1. **Inicializa** una puntuación para cada nodo.
2. **Iterativamente**, cada nodo distribuye su puntuación entre sus vecinos y actualiza su propia puntuación basándose en las puntuaciones recibidas, aplicando un factor de amortiguamiento.
3. Este proceso se **repite** hasta que las puntuaciones se estabilizan.
4. El resultado es una **puntuación de PageRank para cada nodo**, que indica su importancia dentro de la estructura del grafo.

En GraphX, la implementación del algoritmo de PageRank se encuentra dentro de las librerías de algoritmos de grafos y se puede aplicar a objetos `Graph` para calcular la importancia de los vértices.