Google PageRank

Suppose we have a "world wide web" consisting of 4 web sites.

\[
P = \begin{bmatrix}
0 & 0 & 1/3 & 0 \\
1/2 & 0 & 0 & 1/3 \\
1/2 & 0 & 0 & 0 \\
0 & 0 & 1/3 & 0
\end{bmatrix}
\]

# links out of:
- node 1: 2
- node 2: 0
- node 3: 3
- node 4: 1

Idea: Start at some node; do a random walk according to the matrix \( P \). If there is a limiting state, the probabilities of each node in that limit state can be used as a measure of importance of that node. That is:

- Set some \( x_0 \);
- Compute \( P^n x_0 \) for large \( n \);
- Is there a limit?
- Can the limit be found using the power method?

To ensure that this limit exists and it is fast to compute, we will need to modify the matrix \( P \).

(1) Let's replace \( P \) with a stochastic matrix — need to do something with column #2.
Replace \( P \) with
\[
S = \begin{bmatrix}
0 & \frac{1}{4} & \frac{1}{3} & 0 \\
\frac{1}{2} & \frac{1}{4} & \frac{1}{3} & 1 \\
\frac{1}{2} & \frac{1}{4} & 0 & 0 \\
0 & \frac{1}{4} & \frac{1}{3} & 0
\end{bmatrix}
\]

Using MATLAB:
(a) \( \lambda = 1 \) is the dominant eigenvalue
(b) \( \mathbf{v}_1 = [0.32, 0.81, 0.36, 0.32]^T \) is the corresponding eigenvector

So the ranks of each node are the entries of \( \mathbf{v}_1 \) resp.

Better yet, normalize \( \mathbf{v}_1 \) s.t. its entries add up to 1.

Another example:
\[
S = \begin{bmatrix}
0 & 0 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0
\end{bmatrix}
\]

\( |\lambda_1| = |\lambda_2| = |\lambda_3| = 1 \).

So, no dominant eigenvalue.
To resolve: use **DAMPING**:
- create a matrix \( Q \), the same size as \( S \) s.t. all entries of \( Q \) are identical and \( Q \) is a stochastic matrix
That is:

$$Q = \begin{bmatrix}
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
\frac{1}{3} & \frac{1}{3} & \frac{1}{3}
\end{bmatrix}$$

Therefore, \( \lambda = 1 \) is a dominant eigenvalue of \( G \); the corresponding eigenvector has positive entries. We use these entries as the ranks of each node.

3. Define a damping factor \( \alpha \) in \([0, 1]\).

4. Define the "Google" matrix

$$G = \alpha S + (1-\alpha) Q$$

5. Then repeat the method with \( G \) instead of \( S \).

Note:

1. \( G \) is a stochastic matrix

2. Each entry of \( G \) is positive. Provide \( \alpha \ll 1 \).