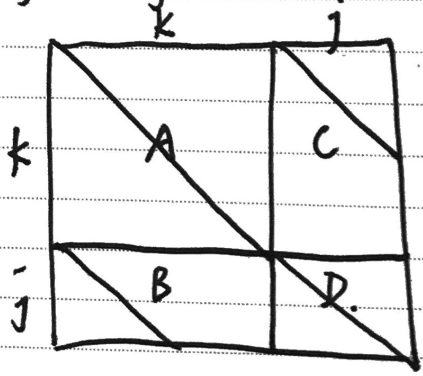
P7.1.8.

Compute the eigenvalues of M, where A, B, C, D ~~is~~ are diagonal.

$$M = \begin{array}{cc} \begin{bmatrix} A & C \\ B & D \end{bmatrix} & \begin{array}{c} k \\ j \end{array} \\ \begin{array}{cc} \;\; k & \;\; j \end{array} & \end{array}$$

If $k > j$, We have



We need to move the diagonal entries of B and C to the off-diag entries in D, using permutation matrices on both sides of M, changing the rows and cols at the same time.

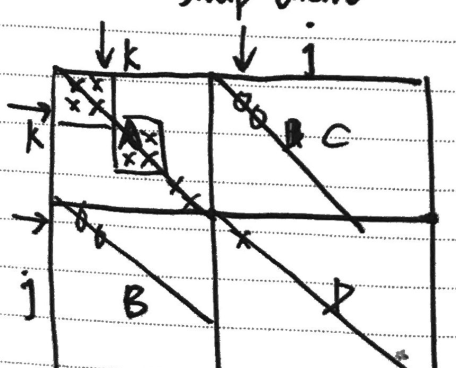
$N = P^{-1}MP$ and M are similar, $\lambda(M) = \lambda(N)$.

In this way, we can zero out B, C and fill those entries in A, D.

And in A, D, the diagonal will be either 1×1 or 2×2, we can compute them easily.

If $k < j$, We have

swap them



The same method works for $k < j$.

After we do the swaps, we can get a nice almost diagonal matrix and compute the 2×2 block ones and get the 1×1 block ones as eigenvalues.

P3.

H is the sparse connectivity matrix of websites. $(n\times n)$

A is the transition posibility matrix. $(n\times n)$

$\alpha = 0.85$.

Let c_j be the column sums of H.

$$c_j = \sum_i H_{ij}.$$

So if $c_j \neq 0$, the probability of row i page is chosen is

$$\alpha \cdot \frac{H(i,j)}{c_j} + (1-\alpha)\cdot \frac{1}{n}$$

And if $c_j = 0$, the probability becomes $\frac{1}{n}$.

So A matrix can be formed as

$$A(i,j) = \begin{cases} \frac{\alpha H(i,j)}{c_j} + (1-\alpha)\frac{1}{n} & , c_j \neq 0 \\ \frac{1}{n} & , c_j = 0 \end{cases}.$$

which can be also written as

$$A = \alpha \cdot H \cdot C + \text{ones}(n,1)\cdot u^T$$

where C is a sparse matrix with the reciprocals of column sums in its diagonal. u is $n\times 1$ vector with $(1-\alpha)\frac{1}{n}$ and $\frac{1}{n}$ in its entries.

$$u_i = \begin{cases} (1-\alpha)\frac{1}{n} & , c_j \neq 0 \\ \frac{1}{n} & , c_j = 0 \end{cases}$$

And because $x = Ax = \alpha \cdot \underbrace{H \cdot C}_{\text{sparse}} \cdot x + \underset{n\times 1}{\text{ones}(n,1)} \cdot \underbrace{\left(\underset{1\times n}{u^T} \cdot \underset{n\times 1}{x}\right)}_{\text{scalar}}$

We can use power method and avoid using huge dense matrix.