

Absorbing Markov chains, spectral centrality and PageRank

MATH 2740 – Mathematics of Data Science – Unit U09

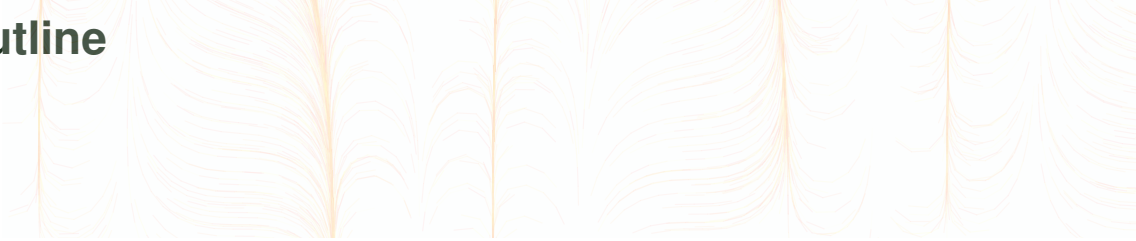
Julien Arino

`julien.arino@umanitoba.ca`

Department of Mathematics @ University of Manitoba

Fall 2026

Outline



From Unit 08 to Unit 09

In Unit 08, we studied irreducible and primitive non-negative matrices and saw that a regular Markov chain converges to a unique positive stationary distribution

$$\mathbf{w} = P\mathbf{w}$$

Two natural questions arise when a directed network is *not* regular:

1. What happens when a Markov chain has **absorbing states** (states that can be entered but never left), making P reducible?
2. How do we rank the importance of vertices in large directed networks (citations, social networks or the World Wide Web) using eigenvectors and linear systems?

Both questions are answered by the same two linear algebra tools:

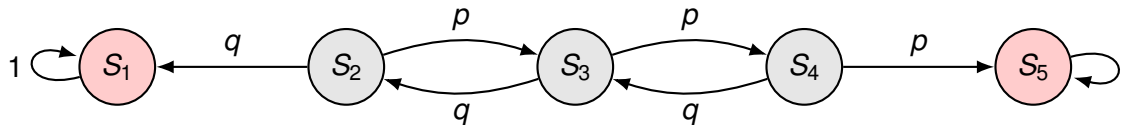
Perron-Frobenius theory and the **Neumann series** $(\mathbb{I} - M)^{-1} = \sum_{k=0}^{\infty} M^k$



Motivating example 1 – 1D random walk with absorbing walls

Consider a particle (or gambler's fortune) moving on the states $\{S_1, S_2, S_3, S_4, S_5\}$:

- ▶ From an interior state S_j ($j \in \{2, 3, 4\}$), the particle moves left to S_{j-1} with probability $q = 1 - p$ and right to S_{j+1} with probability $p \in (0, 1)$
- ▶ States S_1 (ruin) and S_5 (target jackpot) are **absorbing boundaries**: once reached, the process stays there forever



Motivating example 2 – Modifying the breeding experiment

We modify the sex-linked orange cat breeding setup from Unit 08 by repeatedly mating a female offspring with an **orange male** ($X^O Y$) at every generation

The state of the chain is the genotype of the female offspring:

- ▶ $S_1: X^O X^O$ (orange female)
- ▶ $S_2: X^o X^o$ (black female)
- ▶ $S_3: X^O X^o$ (tortoiseshell female)

Let us compute the transition probabilities $p_{ij} = \mathbb{P}(S_i(t+1) \mid S_j(t))$ when mating each female genotype with an $X^O Y$ male

Transitions for the modified breeding experiment

Current state $S_1 (X^O X^O)$

		X^O	Y
Mom	X^O	S_1	M
	X^O	S_1	M

100% $X^O X^O (S_1)$
 $\Rightarrow p_{11} = 1$

Current state $S_2 (X^O X^O)$

		X^O	Y
Mom	X^O	S_3	M
	X^O	S_3	M

100% $X^O X^O (S_3)$
 $\Rightarrow p_{32} = 1$

Current state $S_3 (X^O X^O)$

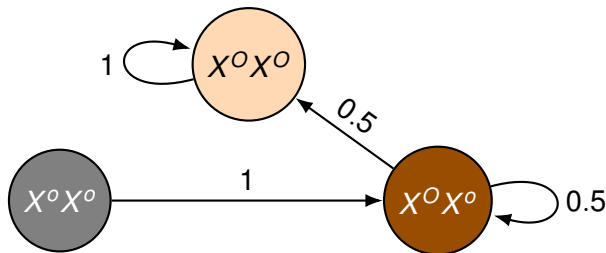
		X^O	Y
Mom	X^O	S_1	M
	X^O	S_3	M

50% S_1 , 50% S_3
 $\Rightarrow p_{13} = 1/2, p_{33} = 1/2$

Summary of the modified breeding chain

In our column-stochastic convention ($p_{ij} = \mathbb{P}(S_i(t+1) \mid S_j(t))$), the transition matrix is

$$P = \begin{pmatrix} 1 & 0 & 1/2 \\ 0 & 0 & 0 \\ 0 & 1 & 1/2 \end{pmatrix}$$



Absorbing states and absorbing Markov chains

Definition 136 (Absorbing state)

A state S_i in a Markov chain with column-stochastic transition matrix $P = [p_{ij}]$ is **absorbing** if once the chain enters S_i , it never leaves:

$$p_{ii} = 1 \quad \text{and} \quad p_{ji} = 0 \text{ for all } j \neq i$$

Equivalently, column i of P is the standard basis vector $\mathbf{e}_i$, and in $\mathcal{G}(P)$ vertex S_i has a self-loop of weight 1 and no outgoing arcs to any other vertex

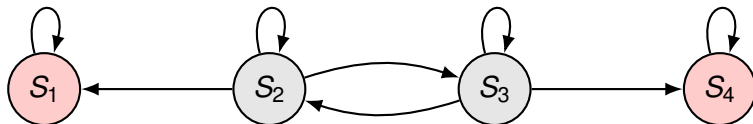
Definition 137 (Absorbing Markov chain)

A Markov chain is **absorbing** if:

1. It has at least one absorbing state
2. From every non-absorbing state, it is possible to reach at least one absorbing state in a finite number of steps (there is a directed walk in $\mathcal{G}(P)$ to an absorbing state)

Key questions about absorbing Markov chains

Suppose we have an absorbing chain such as a random walk with two absorbing boundaries S_1 and S_4 :



1. Does the process eventually reach an absorbing state with probability 1?
2. Starting in transient state S_j , what is the expected number of visits to transient state S_i ?
3. Starting in transient state S_j , what is the expected number of steps before absorption?
4. When there are multiple absorbing states, what is the probability of being absorbed in absorbing state S_i when starting in transient state S_j ?

Absorbing chains are reducible

Suppose a Markov chain on n states has $r \geq 1$ absorbing states and $m = n - r \geq 1$ transient states

- ▶ Because an absorbing state S_i has no outgoing arcs to any other vertex ($p_{ji} = 0$ for all $j \neq i$), there is no directed path from S_i to any transient state
- ▶ Therefore, the transition digraph $\mathcal{G}(P)$ is **not strongly connected**
- ▶ By the characterisation from Unit 08, the transition matrix P is **reducible**

Relabelling the states of the chain corresponds to applying a permutation matrix $\Pi \in \mathcal{M}_n$:

$$\tilde{P} = \Pi^T P \Pi$$

We always relabel the states so that the r **absorbing states come first** ($S_1, \dots, S_r$) and the $m = n - r$ **transient states come last** ($S_{r+1}, \dots, S_n$)

Example – Reordering the 5-state random walk into canonical form

In Motivating example 1 on $(S_1, S_2, S_3, S_4, S_5)$, states S_1 and S_5 are absorbing while S_2, S_3, S_4 are transient. In the original state ordering, the transition matrix is

$$P_{\text{orig}} = \begin{matrix} & \begin{matrix} S_1 & S_2 & S_3 & S_4 & S_5 \end{matrix} \\ \begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \end{matrix} & \begin{pmatrix} 1 & q & 0 & 0 & 0 \\ 0 & 0 & q & 0 & 0 \\ 0 & p & 0 & q & 0 \\ 0 & 0 & p & 0 & 0 \\ 0 & 0 & 0 & p & 1 \end{pmatrix} \end{matrix}$$

Reordering the states as $(S_1, S_5, S_2, S_3, S_4)$ via the permutation matrix $\Pi = [\mathbf{e}_1 \ \mathbf{e}_5 \ \mathbf{e}_2 \ \mathbf{e}_3 \ \mathbf{e}_4]$ gives the canonical block form $\tilde{P} = \Pi^T P_{\text{orig}} \Pi$:

$$\tilde{P} = \begin{matrix} & \begin{matrix} S_1 & S_5 \end{matrix} & \begin{matrix} S_2 & S_3 & S_4 \end{matrix} \\ \begin{matrix} S_1 \\ S_5 \\ S_2 \\ S_3 \\ S_4 \end{matrix} & \left(\begin{array}{cc|ccc} 1 & 0 & q & 0 & 0 \\ 0 & 1 & 0 & 0 & p \\ \hline 0 & 0 & 0 & q & 0 \\ 0 & 0 & p & 0 & q \\ 0 & 0 & 0 & p & 0 \end{array} \right) & = \begin{pmatrix} \mathbb{I}_2 & R \\ \mathbf{0}_{3,2} & Q \end{pmatrix} \end{matrix}$$

Canonical block form of an absorbing chain

Definition 138 (Canonical block form)

For an absorbing Markov chain with r absorbing states (listed first) and $m = n - r$ transient states (listed last), the column-stochastic transition matrix takes the **canonical block form** (or **standard form**)

$$P = \begin{pmatrix} \mathbb{I}_r & R \\ \mathbf{0}_{m,r} & Q \end{pmatrix}$$

where:

- ▶ $\mathbb{I}_r \in \mathcal{M}_r$ is the $r \times r$ identity matrix (absorbing states stay put)
- ▶ $\mathbf{0}_{m,r} \in \mathcal{M}_{mr}$ is the $m \times r$ zero matrix (absorbing states never move to transient states)
- ▶ $R \in \mathcal{M}_{rm}$ contains the one-step transition probabilities from transient states to absorbing states
- ▶ $Q \in \mathcal{M}_m$ contains the one-step transition probabilities among transient states

Visualising the canonical blocks

In our column-stochastic convention, columns correspond to the **from** state (at time t) and rows correspond to the **to** state (at time $t + 1$):

	From absorbing (r cols)	From transient (m cols)
To absorbing (r rows)	$\mathbb{I}_r \in \mathcal{M}_r$	$R \in \mathcal{M}_{rm}$
To transient (m rows)	$\mathbf{0}_{m,r} \in \mathcal{M}_{mr}$	$Q \in \mathcal{M}_m$

Notice that:

- ▶ Each column of $\begin{pmatrix} R \\ Q \end{pmatrix}$ sums to 1: $\mathbb{1}_r^T R + \mathbb{1}_m^T Q = \mathbb{1}_m^T$
- ▶ Q is non-negative and **column-substochastic**: $\mathbb{1}_m^T Q \leq \mathbb{1}_m^T$, with strict inequality in every column j that has a direct transition to an absorbing state ($\sum_{i=1}^r r_{ij} > 0$)

Powers of P in canonical form

To understand the long-term behaviour $\mathbf{p}(k) = P^k \mathbf{p}(0)$, we compute the k th power of P in block form

Proposition 139 (Powers of the canonical transition matrix)

Let $P = \begin{pmatrix} \mathbb{I}_r & R \\ \mathbf{0}_{m,r} & Q \end{pmatrix}$ be in canonical block form. For every integer $k \geq 1$,

$$P^k = \begin{pmatrix} \mathbb{I}_r & R(\mathbb{I}_m + Q + Q^2 + \cdots + Q^{k-1}) \\ \mathbf{0}_{m,r} & Q^k \end{pmatrix} = \begin{pmatrix} \mathbb{I}_r & R \sum_{j=0}^{k-1} Q^j \\ \mathbf{0}_{m,r} & Q^k \end{pmatrix}$$

The (i, j) entry of Q^k is the probability that the chain is in the i th transient state after k steps, given that it started in the j th transient state

Proof of Proposition ??

We proceed by mathematical induction on $k \geq 1$

Base case ($k = 1$): Since $Q^0 = \mathbb{I}_m$, the formula gives $\begin{pmatrix} \mathbb{I}_r & R\mathbb{I}_m \\ \mathbf{0}_{m,r} & Q^1 \end{pmatrix} = P$, which holds

Inductive step: Suppose the formula holds for $k \geq 1$. Using block matrix multiplication for $P^{k+1} = P^k P$:

$$\begin{aligned} P^{k+1} &= \begin{pmatrix} \mathbb{I}_r & R(\mathbb{I}_m + Q + \dots + Q^{k-1}) \\ \mathbf{0}_{m,r} & Q^k \end{pmatrix} \begin{pmatrix} \mathbb{I}_r & R \\ \mathbf{0}_{m,r} & Q \end{pmatrix} \\ &= \begin{pmatrix} \mathbb{I}_r \mathbb{I}_r + R(\dots) \mathbf{0}_{m,r} & \mathbb{I}_r R + R(\mathbb{I}_m + Q + \dots + Q^{k-1}) Q \\ \mathbf{0}_{m,r} \mathbb{I}_r + Q^k \mathbf{0}_{m,r} & \mathbf{0}_{m,r} R + Q^k Q \end{pmatrix} \\ &= \begin{pmatrix} \mathbb{I}_r & R(\mathbb{I}_m + Q + Q^2 + \dots + Q^k) \\ \mathbf{0}_{m,r} & Q^{k+1} \end{pmatrix} \end{aligned}$$

This completes the induction





Spectral radius of the transient block Q

Theorem 140 (Contraction of the transient block)

In any absorbing Markov chain with canonical transient block $Q \in \mathcal{M}_m$, the spectral radius satisfies

$$\rho(Q) < 1 \quad \text{and consequently} \quad \lim_{k \rightarrow \infty} Q^k = \mathbf{0}_m$$

Why does this hold?

- ▶ By definition of an absorbing chain, from every transient state $j \in \{1, \dots, m\}$ there is a directed walk to some absorbing state of length at most m
- ▶ Thus, starting from transient state j , the probability of being absorbed within m steps is strictly positive (> 0), so the probability of remaining in the transient states after m steps is strictly less than 1
- ▶ This means every column sum of Q^m is strictly less than 1:

$$\|Q^m\|_1 = \max_j \sum_{i=1}^m (Q^m)_{ij} < 1$$

- ▶ Therefore $\rho(Q)^m = \rho(Q^m) \leq \|Q^m\|_1 < 1$, which implies $\rho(Q) < 1$ and

Absorption occurs with probability 1

Because $(Q^k)_{ij}$ is the probability of being in transient state i at step k starting from transient state j , Theorem ?? immediately answers our first question

Theorem 141 (Eventual absorption)

In an absorbing Markov chain, the probability that the process eventually reaches an absorbing state is 1, regardless of the initial state

Indeed, starting from transient state j , the probability of still being in *some* transient state after k steps is the j th column sum $\sum_{i=1}^m (Q^k)_{ij} = (\mathbf{1}_m^T Q^k)_j \rightarrow 0$ as $k \rightarrow \infty$

The Neumann series for matrices

Recall the scalar geometric series: for $q \in \mathbb{R}$ with $|q| < 1$, $(1 - q)^{-1} = \sum_{k=0}^{\infty} q^k$.
The exact same identity holds for square matrices!

Theorem 142 (Neumann series)

Let $Q \in \mathcal{M}_m$ be a matrix with spectral radius $\rho(Q) < 1$ (equivalently, $\lim_{k \rightarrow \infty} Q^k = \mathbf{0}_m$). Then $\mathbb{I}_m - Q$ is invertible and its inverse is given by the convergent series

$$(\mathbb{I}_m - Q)^{-1} = \sum_{k=0}^{\infty} Q^k = \mathbb{I}_m + Q + Q^2 + Q^3 + \dots$$

In particular, since $Q \geq 0$ for an absorbing Markov chain, every term $Q^k \geq 0$ is non-negative, so $(\mathbb{I}_m - Q)^{-1} \geq \mathbb{I}_m \geq 0$ has strictly non-negative entries (and diagonal entries ≥ 1)

Proof of Theorem ?? (Neumann series)

Consider the partial sum $S_k = \mathbb{I}_m + Q + Q^2 + \dots + Q^{k-1}$ for $k \geq 1$. Multiplying by $\mathbb{I}_m - Q$ gives a telescoping sum:

$$\begin{aligned}(\mathbb{I}_m - Q)S_k &= (\mathbb{I}_m - Q)(\mathbb{I}_m + Q + Q^2 + \dots + Q^{k-1}) \\&= (\mathbb{I}_m + Q + Q^2 + \dots + Q^{k-1}) - (Q + Q^2 + \dots + Q^{k-1} + Q^k) \\&= \mathbb{I}_m - Q^k\end{aligned}$$

First, suppose $(\mathbb{I}_m - Q)\mathbf{x} = \mathbf{0}$. Then $\mathbf{x} = Q\mathbf{x} = Q^2\mathbf{x} = \dots = Q^k\mathbf{x} \rightarrow \mathbf{0}$ as $k \rightarrow \infty$ since $\rho(Q) < 1$. Hence $\text{null}(\mathbb{I}_m - Q) = \{\mathbf{0}\}$, so $\mathbb{I}_m - Q$ is invertible. Multiplying $(\mathbb{I}_m - Q)S_k = \mathbb{I}_m - Q^k$ on the left by $(\mathbb{I}_m - Q)^{-1}$ gives

$$S_k = (\mathbb{I}_m - Q)^{-1}(\mathbb{I}_m - Q^k) = (\mathbb{I}_m - Q)^{-1} - (\mathbb{I}_m - Q)^{-1}Q^k$$

Taking the limit as $k \rightarrow \infty$ and using $\lim_{k \rightarrow \infty} Q^k = \mathbf{0}_m$ yields

$$\sum_{j=0}^{\infty} Q^j = \lim_{k \rightarrow \infty} S_k = (\mathbb{I}_m - Q)^{-1} \quad \blacksquare$$

The fundamental matrix and absorption quantities

Definition 143 (Fundamental matrix)

For an absorbing Markov chain with canonical transient block $Q \in \mathcal{M}_m$, the matrix

$$N = (\mathbb{I}_m - Q)^{-1} = \mathbb{I}_m + Q + Q^2 + \cdots \in \mathcal{M}_m$$

is called the **fundamental matrix** of the absorbing Markov chain

Theorem 144 (Quantitative analysis of absorbing Markov chains)

Let $P = \begin{pmatrix} \mathbb{I}_r & R \\ \mathbf{0}_{m,r} & Q \end{pmatrix}$ be in canonical form with fundamental matrix

$N = [n_{ij}] = (\mathbb{I}_m - Q)^{-1}$. Then:

1. **Expected visits to transient states:** n_{ij} is the expected total number of times the chain is in the i th transient state, given that it starts in the j th transient state

2. **Expected time to absorption:** Let $\mathbf{t}^T = \mathbf{1}_m^T N = (t_1, \dots, t_m)$. Then

$t_j = \sum_{i=1}^m n_{ij}$ (the j th column sum of N) is the expected number of steps before

Proof of Theorem ??

1. Expected visits $N = (\mathbb{I}_m - Q)^{-1}$: Starting in transient state j , let $X_{ij}^{(k)} \in \{0, 1\}$ be the indicator random variable that equals 1 if the chain is in transient state i at step $k \geq 0$, and 0 otherwise. Then

$\mathbb{E}[X_{ij}^{(k)}] = \mathbb{P}(\text{in transient state } i \text{ at step } k \mid \text{start in } j) = (Q^k)_{ij}$. By linearity of expectation, the expected total number of visits to transient state i is

$$\mathbb{E} \left[\sum_{k=0}^{\infty} X_{ij}^{(k)} \right] = \sum_{k=0}^{\infty} (Q^k)_{ij} = \left(\sum_{k=0}^{\infty} Q^k \right)_{ij} = n_{ij}$$

2. Time to absorption $t^T = \mathbb{1}_m^T N$: The total number of steps spent in *all* transient states before absorption starting from j is the sum over all $i = 1, \dots, m$ of visits to transient state i :

$$t_j = \sum_{i=1}^m n_{ij} = (\mathbb{1}_m^T N)_j$$

3. Absorption probabilities $B = RN$: Taking $k \rightarrow \infty$ in Proposition ?? gives

$$\lim_{k \rightarrow \infty} P^k = \begin{pmatrix} \mathbb{I}_r & R \sum_{j=0}^{\infty} Q^j \\ \mathbf{0}_{m-r} & \lim_{k \rightarrow \infty} Q^k \end{pmatrix} = \begin{pmatrix} \mathbb{I}_r & RN \\ \mathbf{0}_{m-r} & \mathbf{0}_m \end{pmatrix} = \begin{pmatrix} \mathbb{I}_r & B \\ \mathbf{0}_{m-r} & \mathbf{0}_m \end{pmatrix} \quad \blacksquare$$



Example 1 – Back to the genetic example

In the modified orange cat breeding chain ($S_1 = X^O X^O$, $S_2 = X^o X^o$, $S_3 = X^O X^o$), state S_1 is absorbing ($r = 1$) and S_2, S_3 are transient ($m = 2$). The matrix is already in canonical form:

$$P = \begin{pmatrix} 1 & 0 & 1/2 \\ 0 & 0 & 0 \\ 0 & 1 & 1/2 \end{pmatrix} = \begin{pmatrix} \mathbb{I}_1 & R \\ \mathbf{0}_{2,1} & Q \end{pmatrix} \quad \text{with} \quad R = \begin{pmatrix} 0 & 1/2 \end{pmatrix}, \quad Q = \begin{pmatrix} 0 & 0 \\ 1 & 1/2 \end{pmatrix}$$

We compute $\mathbb{I}_2 - Q$ and invert it using $\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$:

$$\mathbb{I}_2 - Q = \begin{pmatrix} 1 & 0 \\ -1 & 1/2 \end{pmatrix} \implies N = (\mathbb{I}_2 - Q)^{-1} = \frac{1}{1/2} \begin{pmatrix} 1/2 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 2 & 2 \end{pmatrix}$$

Interpreting Example 1

The fundamental matrix for transient states $(S_2, S_3) = (X^o X^o, X^o X^o)$ is

$$N = \begin{pmatrix} 1 & 0 \\ 2 & 2 \end{pmatrix}$$

- ▶ Starting with a black female (S_2 , column 1 of N): expected visits are $n_{11} = 1$ generation in S_2 (at $t = 0$) and $n_{21} = 2$ generations in S_3 (tortoiseshell)
- ▶ Starting with a tortoiseshell female (S_3 , column 2 of N): $n_{12} = 0$ generations in S_2 and $n_{22} = 2$ generations in S_3

Expected number of generations before absorption into S_1 ($X^o X^o$):

$$\mathbf{t}^T = \mathbf{1}_2^T N = \begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 3 & 2 \end{pmatrix}$$

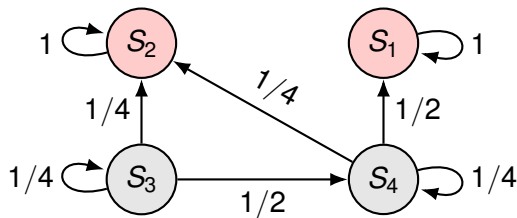
Absorption probability matrix (into the sole absorbing state S_1):

$$B = RN = \begin{pmatrix} 0 & 1/2 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \end{pmatrix}$$

Example 2 – 4-state chain with 2 absorbing states

Consider a 4-state absorbing chain where S_1 (pass/graduated) and S_2 (withdraw) are absorbing ($r = 2$), and S_3 (Year 1) and S_4 (Year 2) are transient ($m = 2$):

$$P = \begin{array}{c|cccc} & S_1 & S_2 & S_3 & S_4 \\ \hline S_1 & 1 & 0 & 0 & 1/2 \\ S_2 & 0 & 1 & 1/4 & 1/4 \\ \hline S_3 & 0 & 0 & 1/4 & 0 \\ S_4 & 0 & 0 & 1/2 & 1/4 \end{array}$$



Here the canonical blocks are

$$R = \begin{pmatrix} 0 & 1/2 \\ 1/4 & 1/4 \end{pmatrix} \in \mathcal{M}_2, \quad Q = \begin{pmatrix} 1/4 & 0 \\ 1/2 & 1/4 \end{pmatrix} \in \mathcal{M}_2$$

Example 2 – Hand calculation of N , t^T and B

Step 1 (Fundamental matrix N): Compute $\mathbb{I}_2 - Q$ and its determinant:

$$\mathbb{I}_2 - Q = \begin{pmatrix} 3/4 & 0 \\ -1/2 & 3/4 \end{pmatrix}, \quad \det(\mathbb{I}_2 - Q) = \frac{3}{4} \cdot \frac{3}{4} - 0 = \frac{9}{16}$$

Inverting via the 2×2 inverse formula gives

$$N = (\mathbb{I}_2 - Q)^{-1} = \frac{16}{9} \begin{pmatrix} 3/4 & 0 \\ 1/2 & 3/4 \end{pmatrix} = \begin{pmatrix} 4/3 & 0 \\ 8/9 & 4/3 \end{pmatrix}$$

Step 2 (Expected steps to absorption $t^T = \mathbb{1}_2^T N$):

$$t^T = (1 \quad 1) \begin{pmatrix} 4/3 & 0 \\ 8/9 & 4/3 \end{pmatrix} = (20/9 \quad 4/3) \quad (\text{starting from } S_3 \text{ and } S_4)$$

Step 3 (Absorption probabilities $B = RN$):

$$B = RN = \begin{pmatrix} 0 & 1/2 \\ 1/4 & 1/4 \end{pmatrix} \begin{pmatrix} 4/3 & 0 \\ 8/9 & 4/3 \end{pmatrix} = \begin{pmatrix} 4/9 & 2/3 \\ 5/9 & 1/3 \end{pmatrix}$$

Notice each column of B sums to 1: starting in S_3 , a student graduates (S_1) with probability $4/9$ and withdraws (S_2) with probability $5/9$

Example 3 – Symmetric 5-state random walk

Return to Motivating example 1 on $\{S_1, S_2, S_3, S_4, S_5\}$ with $p = q = 1/2$ and absorbing walls S_1, S_5 . Reordering states as $(S_1, S_5, S_2, S_3, S_4)$ puts P in canonical form with $r = 2$ and $m = 3$:

$$R = \begin{array}{c} S_1 \\ S_5 \end{array} \begin{array}{ccc} S_2 & S_3 & S_4 \\ \left(\begin{array}{ccc} 1/2 & 0 & 0 \\ 0 & 0 & 1/2 \end{array} \right) \end{array}, \quad Q = \begin{array}{c} S_2 \\ S_3 \\ S_4 \end{array} \begin{array}{ccc} S_2 & S_3 & S_4 \\ \left(\begin{array}{ccc} 0 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 \end{array} \right) \end{array}$$

Inverting $\mathbb{I}_3 - Q = \begin{pmatrix} 1 & -1/2 & 0 \\ -1/2 & 1 & -1/2 \\ 0 & -1/2 & 1 \end{pmatrix}$ gives

$$N = (\mathbb{I}_3 - Q)^{-1} = \begin{pmatrix} 3/2 & 1 & 1/2 \\ 1 & 2 & 1 \\ 1/2 & 1 & 3/2 \end{pmatrix}, \quad \mathbf{t}^T = \mathbf{1}_3^T N = (3 \quad 4 \quad 3), \quad B = RN = \begin{pmatrix} 3/4 & 1/2 \\ 1/4 & 1/2 \end{pmatrix}$$

R implementation – Direct matrix calculation

We can verify Example 3 directly in base R using `solve`:

```
R_mat <- matrix(c(0.5, 0, 0,
                  0, 0, 0.5), nrow = 2, byrow = TRUE)
Q_mat <- matrix(c(0, 0.5, 0,
                  0.5, 0, 0.5,
                  0, 0.5, 0), nrow = 3, byrow = TRUE)
colnames(R_mat) <- colnames(Q_mat) <- rownames(Q_mat) <- c("S2", "S3", "S4")
rownames(R_mat) <- c("S1", "S5")

N_mat <- solve(diag(3) - Q_mat)
t_vec <- colSums(N_mat)
B_mat <- R_mat %*% N_mat
N_mat

##      S2 S3 S4
## S2 1.5  1 0.5
## S3 1.0  2 1.0
```

Using the markovchain package

Note that markovchain uses row-stochastic matrices, so we pass `t(P_abs)`

```
library(markovchain)
P_abs <- matrix(c(1, 0, 0.5,
                  0, 0, 0,
                  0, 1, 0.5), nrow = 3, byrow = TRUE)
mc_abs <- new("markovchain",
              states = sprintf("S_%d", 1:3),
              transitionMatrix = t(P_abs),
              name = "orange_abs")
summary(mc_abs)

## orange_abs Markov chain that is composed by:
## Closed classes:
## S_1
## Recurrent classes:
## {S_1}
## Transient classes:
```

Canonical form, absorption times and probabilities in R

```
canonicForm(mc_abs)

## orange_abs
## A 3 - dimensional discrete Markov Chain defined by the following states
## S_1, S_2, S_3
## The transition matrix (by rows) is defined as follows:
##      S_1 S_2 S_3
## S_1 1.0  0 0.0
## S_2 0.0  0 1.0
## S_3 0.5  0 0.5

meanAbsorptionTime(mc_abs)

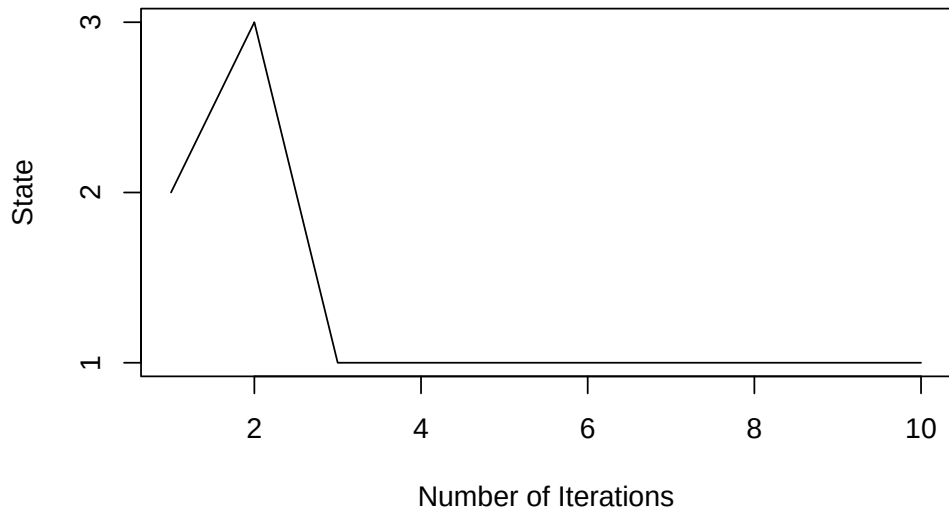
## S_2 S_3
## 3 2

absorptionProbabilities(mc_abs)
```

Simulating a realisation with DTMCPack

```
library(DTMCPack)
set.seed(2740)
IC <- c(0, 1, 0) # Start in S_2 (black female)
sol <- DTMC(t(P_abs), IC, 10, trace = TRUE)

## NULL
```





Why spectral centrality?

Combinatorial centrality measures on graphs are often based on counting shortest paths (such as closeness and betweenness centrality)

Computational bottleneck on massive networks:

- ▶ All-pairs shortest-path algorithms require $O(nm)$ or $O(n^3)$ operations
- ▶ For a social network or web graph with $n \sim 10^8$ vertices, $n^2 \sim 10^{16}$ is completely intractable

The spectral alternative:

- ▶ Express vertex importance via matrix-vector products $A\mathbf{x}$, eigenvectors $A\mathbf{x} = \lambda\mathbf{x}$ or linear systems $(\mathbb{I} - \alpha A)\mathbf{x} = \mathbf{b}$
- ▶ When the adjacency matrix A is **sparse** ($m = |E| \ll n^2$ non-zero entries), one sparse matrix-vector product $A\mathbf{x}$ takes only $O(m)$ time!

Convention for directed adjacency matrices

To remain consistent with our column-stochastic Markov chain convention $\mathbf{p}(t+1) = \mathbf{P}\mathbf{p}(t)$ where column j records transitions *from* j *to* i , we use the following convention for a directed graph $\mathcal{G} = (V, E)$ on n vertices:

$$a_{ij} = \begin{cases} 1 & \text{if there is a directed arc } j \rightarrow i \text{ (from } j \text{ to } i) \\ 0 & \text{otherwise} \end{cases}$$

With this convention:

- ▶ **Row** i of A records arcs *entering* vertex i , so the **in-degree** of vertex i is $d_i^{\text{in}} = \sum_{j=1}^n a_{ij} = (A\mathbf{1}_n)_i$
- ▶ **Column** j of A records arcs *leaving* vertex j , so the **out-degree** of vertex j is $d_j^{\text{out}} = \sum_{i=1}^n a_{ij} = (\mathbf{1}_n^T A)_j$
- ▶ For an undirected graph, $A = A^T$ is symmetric and $d_i^{\text{in}} = d_i^{\text{out}} = d_i$

First-order measure – In-degree centrality

The simplest measure of prestige in a directed network is **in-degree centrality**:

$$\mathbf{d}^{\text{in}} = A\mathbf{1}_n, \quad d_i^{\text{in}} = \sum_{j=1}^n a_{ij}$$

Each arc $j \rightarrow i$ is treated as one “vote” for the importance of vertex i

Fundamental limitation:

- ▶ In-degree treats all incoming links as equal
- ▶ Receiving a link from a major hub or authoritative page counts the exact same (+1) as receiving a link from an isolated spam page
- ▶ We want a **recursive** notion of prestige: *a vertex is important if it is pointed to by other important vertices*

Eigenvector centrality

Instead of summing 1 for each incoming neighbour $j \rightarrow i$, we sum the *importance scores* x_j of those neighbours and require x_i to be proportional to that sum

Definition 145 (Eigenvector centrality)

Let $A = [a_{ij}] \in \mathcal{M}_n$ be the adjacency matrix of a graph or digraph (with $a_{ij} = 1$ if $j \rightarrow i$). The **eigenvector centrality** (or **prestige score**) of vertex i is the i th entry of the non-negative eigenvector $\mathbf{x} \geq \mathbf{0}$ satisfying

$$\lambda x_i = \sum_{j=1}^n a_{ij} x_j, \quad i = 1, \dots, n \iff A\mathbf{x} = \lambda\mathbf{x}$$

associated with the dominant eigenvalue $\lambda = \rho(A)$

Notice that $x_i = \frac{1}{\lambda} \sum_{j \in \mathcal{N}^{\text{in}}(i)} x_j$ depends on both the **quantity** of in-neighbours $|\mathcal{N}^{\text{in}}(i)|$ and the **quality** x_j of each in-neighbour

Why Perron-Frobenius theory is essential here

For $\mathbf{x}$ to be a valid centrality ranking, its entries must be **non-negative** ($x_i \geq 0$) and uniquely determined (up to normalisation $\mathbb{1}_n^T \mathbf{x} = 1$)

Theorem 146 (Existence and uniqueness of eigenvector centrality)

Let $\mathcal{G} = (V, E)$ be a strongly connected digraph (or connected undirected graph) with adjacency matrix $A \in \mathcal{M}_n$. Then A is irreducible and non-negative, so by the Perron-Frobenius theorem (Unit 08):

- 1. The spectral radius $\lambda_1 = \rho(A) > 0$ is a simple eigenvalue of A*
- 2. There exists a unique eigenvector $\mathbf{x} > \mathbf{0}$ (the **Perron eigenvector**) satisfying*

$$A\mathbf{x} = \rho(A)\mathbf{x} \quad \text{and} \quad \mathbb{1}_n^T \mathbf{x} = \sum_{i=1}^n x_i = 1$$

- 3. No other eigenvalue has a non-negative eigenvector*

Constructing a stochastic matrix from an adjacency matrix

Let $A = [a_{ij}] \in \mathcal{M}_n$ be the adjacency matrix of a simple undirected graph $\mathcal{G} = (V, E)$ with no isolated vertices, and let $D = \text{diag}(d_{11}, \dots, d_{nn})$ be its degree matrix, with diagonal entries

$$d_{jj} = \sum_{i=1}^n a_{ji} = \sum_{i=1}^n a_{ij} > 0$$

Proposition 147 (Random walk transition matrix on an undirected graph)

*The matrix $P = AD^{-1}$ is **column-stochastic**, and defines the transition matrix of a random walk on $\mathcal{G}$ where a walker at vertex j moves to a uniformly chosen neighbour $i \in \mathcal{N}(j)$ with probability $1/d_{jj}$*

Proof of Proposition ??

Write the entries of the diagonal matrix D^{-1} as d_{ij}^{-1} , where $d_{ii}^{-1} = 1/d_{ii}$ for $1 \leq i \leq n$ and $d_{ij}^{-1} = 0$ for $i \neq j$

By matrix multiplication, the (i, j) entry of AD^{-1} is

$$(AD^{-1})_{ij} = \sum_{k=1}^n a_{ik} d_{kj}^{-1} = a_{ij} d_{jj}^{-1} = \frac{a_{ij}}{d_{jj}} \geq 0, \quad i, j = 1, \dots, n$$

Therefore, the sum of column j of AD^{-1} is

$$\begin{aligned} \sum_{i=1}^n a_{ij} d_{jj}^{-1} &= d_{jj}^{-1} \sum_{i=1}^n a_{ij} \\ &= d_{jj}^{-1} d_{jj} = 1 \end{aligned}$$

Hence every column sums to 1 and AD^{-1} is column-stochastic



Computing eigenvector centrality – The power method

For large networks, computing the full characteristic polynomial is impossible.

Instead, we compute the dominant eigenpair $(\lambda_1, \mathbf{x})$ using the **power method**:

Start with an initial positive vector $\mathbf{x}^{(0)} > \mathbf{0}$ (such as $\mathbf{x}^{(0)} = \frac{1}{n}\mathbf{1}_n$). For $k = 1, 2, \dots$:

1. Multiply by A : $\mathbf{y}^{(k)} = A\mathbf{x}^{(k-1)}$
2. Normalise (using the ℓ_1 norm $\|\mathbf{y}^{(k)}\|_1 = \mathbf{1}_n^T \mathbf{y}^{(k)}$ or the maximum-magnitude entry $m(\mathbf{y}^{(k)})$):

$$\mathbf{x}^{(k)} = \frac{\mathbf{y}^{(k)}}{\|\mathbf{y}^{(k)}\|_1}$$

until $\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|_1 < \varepsilon$

Theorem 148 (Convergence of the power method)

If A is primitive (so $\lambda_1 = \rho(A) > |\lambda_2| \geq \dots \geq |\lambda_n|$), then $\mathbf{x}^{(k)} \rightarrow \mathbf{x}$ as $k \rightarrow \infty$, and the error decays at the geometric rate

$$\|\mathbf{x}^{(k)} - \mathbf{x}\| = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right)$$



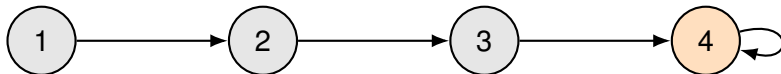
What goes wrong on general directed graphs?

Real-world directed networks (citation graphs, web graphs, DAGs) are rarely strongly connected! When $\mathcal{G}(A)$ is **reducible**, eigenvector centrality suffers from a severe pathology:

- ▶ Suppose vertex i is a **source node** (in-degree $d_i^{\text{in}} = 0$, so nobody points to i)
- ▶ Then row i of A is all zeros, so the eigenvector equation $A\mathbf{x} = \lambda\mathbf{x}$ with $\lambda > 0$ gives

$$\lambda x_i = \sum_{j=1}^n a_{ij} x_j = 0 \implies x_i = 0$$

- ▶ **Domino effect:** If vertex k only receives links from source nodes with $x_i = 0$, then $\lambda x_k = \sum_{j \rightarrow k} x_j = 0 \implies x_k = 0$ as well!



Here $x_1 = 0 \implies x_2 = 0 \implies x_3 = 0$, and only vertex 4 gets non-zero eigenvector centrality ($x_4 = 1$)!

Katz centrality (1953) – Baseline “free” importance

Leo Katz (1953) resolved the zero-propagation problem on directed graphs by giving every vertex a small baseline “free” importance $\beta > 0$, plus damped importance $\alpha > 0$ from its in-neighbours

Definition 149 (Katz centrality)

Let $A \in \mathcal{M}_n$ be the adjacency matrix of a directed graph, $\alpha > 0$ be an **attenuation factor** (or damping parameter) and $\beta > 0$ be a baseline constant. The **Katz centrality** vector $\mathbf{x} \in \mathbb{R}^n$ is the solution to

$$x_i = \alpha \sum_{j=1}^n a_{ij} x_j + \beta, \quad i = 1, \dots, n \iff \mathbf{x} = \alpha \mathbf{A} \mathbf{x} + \beta \mathbf{1}_n$$

Rearranging terms gives the linear system

$$(\mathbb{I}_n - \alpha \mathbf{A}) \mathbf{x} = \beta \mathbf{1}_n$$

Neumann series and walk-counting interpretation of Katz centrality

When is $\mathbb{I}_n - \alpha A$ guaranteed to be invertible with a strictly positive solution $\mathbf{x} > \mathbf{0}$?

Theorem 150 (Existence and walk-counting expansion of Katz centrality)

Let $A \in \mathcal{M}_n$ be a non-negative adjacency matrix and suppose $0 < \alpha < \frac{1}{\rho(A)}$ and $\beta > 0$. Then $\rho(\alpha A) = \alpha \rho(A) < 1$, so by Theorem ?? (the Neumann series), $\mathbb{I}_n - \alpha A$ is invertible and

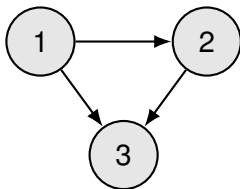
$$\mathbf{x} = \beta(\mathbb{I}_n - \alpha A)^{-1} \mathbf{1}_n = \beta \left(\sum_{k=0}^{\infty} \alpha^k A^k \right) \mathbf{1}_n = \beta \left(\mathbf{1}_n + \alpha A \mathbf{1}_n + \alpha^2 A^2 \mathbf{1}_n + \cdots \right) > \mathbf{0}$$

Walk-counting interpretation:

- ▶ Recall from Unit 07 that $(A^k)_{ij}$ is the number of directed walks of length k from j to i
- ▶ Therefore $(A^k \mathbf{1}_n)_i = \sum_{j=1}^n (A^k)_{ij}$ is the total number of directed walks of length k from *any* vertex j terminating at vertex i
- ▶ Katz centrality sums walks of **all lengths** $k \geq 0$ ending at i , discounted by α^k :

Example – Katz centrality on a 3-vertex digraph

Consider the 3-vertex directed graph with arcs $1 \rightarrow 2$, $1 \rightarrow 3$ and $2 \rightarrow 3$:



With $a_{ij} = 1$ if $j \rightarrow i$, the adjacency matrix is strictly lower triangular:

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

Here $\rho(A) = 0$, so any $\alpha > 0$ satisfies $\rho(\alpha A) = 0 < 1$

Let us compute Katz centrality with $\alpha = 1/2$ and $\beta = 1$ by solving $(\mathbb{I}_3 - \frac{1}{2}A)\mathbf{x} = \mathbb{1}_3$:

$$\mathbb{I}_3 - \frac{1}{2}A = \begin{pmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ -1/2 & -1/2 & 1 \end{pmatrix} \Rightarrow \begin{cases} x_1 = 1 \\ x_2 = \frac{1}{2}x_1 + 1 = \frac{3}{2} \\ x_3 = \frac{1}{2}x_1 + \frac{1}{2}x_2 + 1 = \frac{1}{2} + \frac{3}{4} + 1 = \frac{9}{4} \end{cases}$$

Normalising $\mathbf{x} = (1, 3/2, 9/4)^T = \frac{1}{4}(4, 6, 9)^T$ by $\mathbb{1}_3^T \mathbf{x} = 19/4$ gives

$\tilde{\mathbf{x}}_{\text{Spec}} = (\frac{4}{19}, \frac{6}{19}, \frac{9}{19})^T$, ranking $3 \succ 2 \succ 1$



What makes an important webpage?

In the early days of the World Wide Web (mid-1990s), search engines ranked pages primarily by keyword frequency on the page itself, which was easily manipulated

In 1998, Sergey Brin, Larry Page, Rajeev Motwani and Terry Winograd introduced **PageRank**, the algorithm at the heart of Google

Core idea: Use the directed hyperlink structure of the Web to measure authority

- ▶ Webpages are vertices $V = \{1, \dots, n\}$; a hyperlink from page j to page i is a directed arc $j \rightarrow i$
- ▶ A link $j \rightarrow i$ is a recommendation or “vote” from page j for the importance of page i
- ▶ **Random surfer model:** Imagine a web surfer who clicks links at random from page to page. The PageRank of page i is the long-run proportion of time the random surfer spends on page i

From Katz centrality to vote splitting

Why not simply use Katz or eigenvector centrality on the Web adjacency matrix A ?

The “generous hub” flaw of Katz/eigenvector centrality:

- ▶ In Katz centrality ($x_i = \alpha \sum_{j \rightarrow i} x_j + \beta$), if an important directory page j links to 10,000 pages, *every single one* of those 10,000 pages receives αx_j importance!
- ▶ A page with more outgoing links exerts more total influence on the network

Brin and Page’s vote-splitting principle:

- ▶ Each page j has a single unit of voting power equal to its own importance x_j
- ▶ If page j has $d_j^{\text{out}} > 0$ outgoing links, it splits its importance **equally** among them, passing x_j/d_j^{out} along each outgoing link:

$$x_i = \sum_{j \rightarrow i} \frac{x_j}{d_j^{\text{out}}}$$

The hyperlink transition matrix

Let $A = [a_{ij}] \in \mathcal{M}_n$ be the directed adjacency matrix ($a_{ij} = 1$ if page j links to page i , with no self-loops). Suppose first that every page has at least one outgoing link ($d_j^{\text{out}} = \sum_{i=1}^n a_{ij} \geq 1$)

Define the diagonal out-degree matrix $D^{\text{out}} = \text{diag}(d_1^{\text{out}}, \dots, d_n^{\text{out}})$ and the **hyperlink matrix** $P = [p_{ij}] \in \mathcal{M}_n$:

$$p_{ij} = \begin{cases} \frac{1}{d_j^{\text{out}}} & \text{if page } j \text{ links to page } i \\ 0 & \text{otherwise} \end{cases} \quad \Longleftrightarrow \quad P = A(D^{\text{out}})^{-1}$$

Proposition 151

If $d_j^{\text{out}} \geq 1$ for all $j = 1, \dots, n$, then $P = A(D^{\text{out}})^{-1}$ is **column-stochastic**, and the vote-splitting equations $x_i = \sum_{j \rightarrow i} x_j / d_j^{\text{out}}$ take the fixed-point form

$$P\mathbf{x} = \mathbf{x}$$

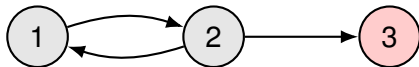
Two linear algebra obstacles on real web graphs

If we try to solve $P\mathbf{x} = \mathbf{x}$ directly on the real Web, two fatal problems occur:

Obstacle 1: Dangling nodes

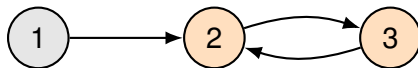
($d_j^{\text{out}} = 0$)

- ▶ Many pages (PDF files, images, dead-end pages) have **no outgoing links**: $d_j^{\text{out}} = 0$
- ▶ Column j of the raw hyperlink matrix is **0**, so probability mass “leaks” out of the network and P is not stochastic!



Obstacle 2: Reducibility and spider traps

- ▶ The Web is not strongly connected ($\mathcal{G}(P)$ is reducible) and contains **spider traps** (closed strongly connected components) and periodic cycles
- ▶ $P\mathbf{x} = \mathbf{x}$ may have multiple linearly independent solutions, assign $x_i = 0$ outside traps, or cause power iteration to oscillate!



Fixing Obstacle 1 – Handling dangling nodes

Before forming the Google matrix, we must ensure that P is genuinely column-stochastic ($\mathbb{1}_n^T P = \mathbb{1}_n^T$)

Whenever a vertex j is a **dangling node** ($d_j^{\text{out}} = 0$, giving a zero column in the raw hyperlink matrix), there are two standard conventions:

1. **Uniform teleportation from dead ends:** Replace the zero column j with the uniform probability vector $\frac{1}{n}\mathbb{1}_n = (1/n, \dots, 1/n)^T$ (a random surfer reaching a dead-end page picks any page on the Web uniformly at random)
2. **Self-loop at dead ends:** Add a self-loop $j \rightarrow j$ so that $p_{jj} = 1$ (column j becomes $\mathbf{e}_j$)

In both conventions, every column of the resulting matrix P is a valid probability vector ($P \geq 0$ and $\mathbb{1}_n^T P = \mathbb{1}_n^T$). From now on, $P \in \mathcal{M}_n$ denotes this column-stochastic web transition matrix

Fixing Obstacle 2 – Teleportation and the Google matrix

To overcome reducibility, spider traps and periodicity, Brin and Page introduced a **damping factor** (or teleportation parameter) $\alpha \in (0, 1)$, typically chosen as $\alpha = 0.85$:

- ▶ With probability α , the random surfer follows an outgoing link on the current page (according to P)
- ▶ With probability $1 - \alpha$, the surfer gets bored and jumps (teleports) to a page chosen uniformly at random from all n pages (according to $\frac{1}{n}\mathbb{1}_n$)

Definition 152 (Google matrix)

Let $P \in \mathcal{M}_n$ be a column-stochastic transition matrix and let $\alpha \in (0, 1)$ be a damping factor. The **Google matrix** is the $n \times n$ matrix

$$G = \alpha P + (1 - \alpha) \frac{1}{n} \mathbb{1}_n \mathbb{1}_n^T = \alpha P + \frac{1 - \alpha}{n} \mathbb{J}_n$$

where $\mathbb{J}_n = \mathbb{1}_n \mathbb{1}_n^T \in \mathcal{M}_n$ is the $n \times n$ matrix of all ones

Existence, uniqueness and convergence of PageRank

Theorem 153 (PageRank theorem)

Let $P \in \mathcal{M}_n$ be a column-stochastic matrix and let $\alpha \in (0, 1)$. Then the Google matrix

$$G = \alpha P + (1 - \alpha) \frac{1}{n} \mathbb{1}_n \mathbb{1}_n^T$$

is **column-stochastic** and **strictly positive** ($G > 0$), hence **primitive**. By Perron's theorem (Unit 08):

1. The spectral radius $\lambda_1 = \rho(G) = 1$ is a simple, strictly dominant eigenvalue of G (in fact, $|\lambda| \leq \alpha < 1$ for every other eigenvalue of G)
2. There exists a unique stationary probability vector $\mathbf{x} > \mathbf{0}$ (the **PageRank vector**) satisfying

$$G\mathbf{x} = \mathbf{x} \quad \text{and} \quad \mathbb{1}_n^T \mathbf{x} = 1$$

3. For any initial probability distribution $\mathbf{x}^{(0)}$, the power iteration $\mathbf{x}^{(k)} = G^k \mathbf{x}^{(0)}$ converges to $\mathbf{x}$ as $k \rightarrow \infty$ at the geometric rate $O(\alpha^k)$

Proof of Theorem ??

1. Strict positivity ($G > 0$): Since $P \geq 0$ and $\alpha \in (0, 1)$, we have $\alpha P \geq 0$. Every entry of $\frac{1-\alpha}{n} \mathbb{1}_n$ equals $\frac{1-\alpha}{n} > 0$. Therefore, every entry of G satisfies

$$g_{ij} = \alpha p_{ij} + \frac{1-\alpha}{n} \geq \frac{1-\alpha}{n} > 0$$

so $G > 0$ is strictly positive (and therefore irreducible and primitive, as $G^1 > 0$)

2. Column-stochasticity ($\mathbb{1}_n^T G = \mathbb{1}_n^T$): Since $\mathbb{1}_n^T P = \mathbb{1}_n^T$ and $\mathbb{1}_n^T \mathbb{1}_n = n$, we compute

$$\mathbb{1}_n^T G = \alpha(\mathbb{1}_n^T P) + \frac{1-\alpha}{n}(\mathbb{1}_n^T \mathbb{1}_n) \mathbb{1}_n^T = \alpha \mathbb{1}_n^T + \frac{1-\alpha}{n} n \mathbb{1}_n^T = (\alpha + 1 - \alpha) \mathbb{1}_n^T = \mathbb{1}_n^T$$

Thus G is column-stochastic, which implies $\rho(G) = 1$

3. Application of Perron's theorem: Since $G > 0$ is a positive column-stochastic matrix, Perron's theorem guarantees that $\lambda_1 = \rho(G) = 1$ is simple and strictly dominant, with a unique positive eigenvector $\mathbf{x} > \mathbf{0}$ normalised by $\mathbb{1}_n^T \mathbf{x} = 1$, and $\lim_{k \rightarrow \infty} G^k \mathbf{x}^{(0)} = \mathbf{x}$ 



Reformulating PageRank as a linear system

Although $G\mathbf{x} = \mathbf{x}$ is an eigenvalue problem for the eigenvalue $\lambda = 1$, we can use the normalisation condition $\mathbb{1}_n^T \mathbf{x} = 1$ to turn it into a **non-homogeneous linear system** $(\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n}\mathbb{1}_n$!

Theorem 154 (Linear system formulation of PageRank)

Let $P \in \mathcal{M}_n$ be column-stochastic and $\alpha \in (0, 1)$. A vector $\mathbf{x} \in \mathbb{R}^n$ with $\mathbb{1}_n^T \mathbf{x} = 1$ satisfies $G\mathbf{x} = \mathbf{x}$ if and only if it satisfies the linear system

$$(\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n}\mathbb{1}_n$$

Moreover, $\mathbb{I}_n - \alpha P$ is invertible and its inverse is given by the Neumann series $(\mathbb{I}_n - \alpha P)^{-1} = \sum_{k=0}^{\infty} \alpha^k P^k$, so the unique PageRank vector is

$$\mathbf{x} = \frac{1-\alpha}{n}(\mathbb{I}_n - \alpha P)^{-1}\mathbb{1}_n$$

Proof of Theorem ??

1. Equivalence with $G\mathbf{x} = \mathbf{x}$: For any vector $\mathbf{x}$ satisfying $\mathbb{1}_n^T \mathbf{x} = 1$, associativity of matrix multiplication gives

$$\mathbb{J}_n \mathbf{x} = (\mathbb{1}_n \mathbb{1}_n^T) \mathbf{x} = \mathbb{1}_n (\mathbb{1}_n^T \mathbf{x}) = \mathbb{1}_n (1) = \mathbb{1}_n$$

Substituting this into $G\mathbf{x} = \mathbf{x}$ gives

$$G\mathbf{x} = \mathbf{x} \iff \alpha P\mathbf{x} + \frac{1-\alpha}{n} \mathbb{1}_n \mathbb{1}_n^T \mathbf{x} = \mathbf{x} \iff \alpha P\mathbf{x} + \frac{1-\alpha}{n} \mathbb{1}_n = \mathbf{x} \iff (\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n} \mathbb{1}_n$$

2. Invertibility of $\mathbb{I}_n - \alpha P$: Since P is column-stochastic, $\rho(P) = 1$. Because $\alpha \in (0, 1)$, we have $\rho(\alpha P) = \alpha \rho(P) = \alpha < 1$. By Theorem ?? (the Neumann series), $\mathbb{I}_n - \alpha P$ is invertible with $(\mathbb{I}_n - \alpha P)^{-1} = \sum_{k=0}^{\infty} \alpha^k P^k \geq \mathbf{0}$

3. Automatic normalisation ($\mathbb{1}_n^T \mathbf{x} = 1$): Left-multiplying $(\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n} \mathbb{1}_n$ by $\mathbb{1}_n^T$ and using $\mathbb{1}_n^T P = \mathbb{1}_n^T$ yields

$$\mathbb{1}_n^T (\mathbb{I}_n - \alpha P)\mathbf{x} = (1 - \alpha) \mathbb{1}_n^T \mathbf{x} = \frac{1-\alpha}{n} \mathbb{1}_n^T \mathbb{1}_n = 1 - \alpha \implies \mathbb{1}_n^T \mathbf{x} = 1 \quad \blacksquare$$

Unifying thread of Unit 09 – The resolvent $(\mathbb{I} - M)^{-1}$

Notice how all three topics in this unit are governed by the exact same matrix inverse (the **Neumann series**):

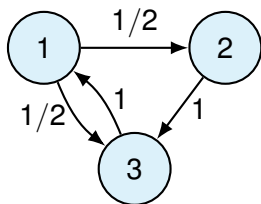
Problem	Matrix M	Condition $\rho(M) < 1$	Key quantity
Absorbing Markov chains	Transient block Q	Absorption reachable	$N = (\mathbb{I}_m - Q)^{-1}$
Katz centrality	αA	$0 < \alpha < 1/\rho(A)$	$\mathbf{x} = \beta(\mathbb{I}_n - \alpha A)^{-1} \mathbf{1}_n$
PageRank	αP	$0 < \alpha < 1$ ($\rho(P) = 1$)	$\mathbf{x} = \frac{1-\alpha}{n}(\mathbb{I}_n - \alpha P)^{-1} \mathbf{1}_n$

For hand calculations on small graphs ($n = 3$ or 4), solving the linear system $(\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n}\mathbf{1}_n$ is much faster than computing the dense matrix G and finding its nullspace!

Worked exam-style example 1 – 3-page web digraph

Consider a 3-page miniature web where:

- ▶ Page 1 links to Page 2 and Page 3 ($d_1^{\text{out}} = 2$)
- ▶ Page 2 links to Page 3 ($d_2^{\text{out}} = 1$)
- ▶ Page 3 links to Page 1 ($d_3^{\text{out}} = 1$)



The column-stochastic transition matrix $P = [p_{ij}]$ (where $p_{ij} = 1/d_j^{\text{out}}$ if $j \rightarrow i$) is

$$P = \begin{pmatrix} 0 & 0 & 1 \\ 1/2 & 0 & 0 \\ 1/2 & 1 & 0 \end{pmatrix}$$

With damping factor $\alpha = 1/2$, the Google matrix is

$$G = \frac{1}{2}P + \frac{1}{6}\mathbb{J}_3 = \begin{pmatrix} 1/6 & 1/6 & 2/3 \\ 5/12 & 1/6 & 1/6 \\ 5/12 & 2/3 & 1/6 \end{pmatrix}$$

Worked exam-style example 1 – Solving $(\mathbb{I}_3 - \alpha P)\mathbf{x} = \frac{1-\alpha}{3}\mathbf{1}_3$

Let us compute the PageRank vector $\mathbf{x}$ for $\alpha = 1/2$ by solving $(\mathbb{I}_3 - \frac{1}{2}P)\mathbf{x} = \frac{1}{6}\mathbf{1}_3$:

$$\mathbb{I}_3 - \frac{1}{2}P = \begin{pmatrix} 1 & 0 & -1/2 \\ -1/4 & 1 & 0 \\ -1/4 & -1/2 & 1 \end{pmatrix} \Rightarrow \begin{cases} x_1 - \frac{1}{2}x_3 = \frac{1}{6} & \text{(Row 1)} \\ -\frac{1}{4}x_1 + x_2 = \frac{1}{6} & \text{(Row 2)} \\ -\frac{1}{4}x_1 - \frac{1}{2}x_2 + x_3 = \frac{1}{6} & \text{(Row 3)} \end{cases}$$

Express x_2 and x_3 in terms of x_1 using Rows 1 and 2:

$$x_2 = \frac{1}{4}x_1 + \frac{1}{6}, \quad x_3 = 2x_1 - \frac{1}{3}$$

Substitute into the normalisation condition $x_1 + x_2 + x_3 = 1$:

$$x_1 + \left(\frac{1}{4}x_1 + \frac{1}{6}\right) + \left(2x_1 - \frac{1}{3}\right) = 1 \Rightarrow \frac{13}{4}x_1 = \frac{7}{6} \Rightarrow x_1 = \frac{14}{39}$$

Then $x_2 = \frac{1}{4}\left(\frac{14}{39}\right) + \frac{1}{6} = \frac{10}{39}$ and $x_3 = 2\left(\frac{14}{39}\right) - \frac{1}{3} = \frac{15}{39}$, so

$$\mathbf{x} = \begin{pmatrix} 14/39 \\ 10/39 \\ 15/39 \end{pmatrix} \approx \begin{pmatrix} 0.359 \\ 0.256 \\ 0.385 \end{pmatrix} \Rightarrow \text{Ranking: Page 3} \succ \text{Page 1} \succ \text{Page 2}$$

Why Page 1 ranks almost as high as Page 3

Look closely at the ranking in our 3-page example:

- ▶ **In-degrees:** Page 3 has in-degree 2 (from 1 and 2), while Page 1 and Page 2 both have in-degree 1
- ▶ By raw in-degree, Page 1 and Page 2 would be tied!
- ▶ However, in PageRank, Page 1 receives the **undivided vote** of the most important page (Page 3, with $d_3^{\text{out}} = 1$), whereas Page 2 only receives **half a vote** from Page 1 ($d_1^{\text{out}} = 2$)
- ▶ Consequently, Page 1 ($x_1 = 14/39$) ranks far above Page 2 ($x_2 = 10/39$) and nearly matches Page 3 ($x_3 = 15/39$)

What if we use the standard Google damping factor $\alpha = 0.85$ on the same 3-page network?

$$(\mathbb{I}_3 - 0.85P)\mathbf{x} = \frac{0.15}{3}\mathbb{1}_3 \implies \mathbf{x} \approx \begin{pmatrix} 0.388 \\ 0.215 \\ 0.397 \end{pmatrix}$$

Worked exam-style example 2 – 4-page web with a dangling node

Suppose we add a dangling Page 4 ($d_4^{\text{out}} = 0$) to a 4-page web where $1 \rightarrow 2$, $1 \rightarrow 3$, $2 \rightarrow 3$, $3 \rightarrow 1$ and $2 \rightarrow 4$ (so $d_1^{\text{out}} = 2$, $d_2^{\text{out}} = 2$, $d_3^{\text{out}} = 1$ and $d_4^{\text{out}} = 0$):

$$P_{\text{raw}} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1/2 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \end{pmatrix} \xrightarrow{\text{replace zero col 4 by } \frac{1}{4} \mathbb{1}_4} P = \begin{pmatrix} 0 & 0 & 1 & 1/4 \\ 1/2 & 0 & 0 & 1/4 \\ 1/2 & 1/2 & 0 & 1/4 \\ 0 & 1/2 & 0 & 1/4 \end{pmatrix}$$

Notice why we can simplify $(\mathbb{I}_4 - \alpha P)\mathbf{x} = \frac{1-\alpha}{4}\mathbb{1}_4$ even faster when column 4 is $\frac{1}{4}\mathbb{1}_4$:

- ▶ Write $P = P_{\text{raw}} + \frac{1}{4}\mathbb{1}_4\mathbf{e}_4^T$, so $P\mathbf{x} = P_{\text{raw}}\mathbf{x} + \frac{x_4}{4}\mathbb{1}_4$
- ▶ Then $G\mathbf{x} = \mathbf{x} \iff (\mathbb{I}_4 - \alpha P_{\text{raw}})\mathbf{x} = \left(\frac{1-\alpha}{4} + \frac{\alpha x_4}{4}\right)\mathbb{1}_4 = c\mathbb{1}_4$, where P_{raw} is sparse!
- ▶ Thus we can solve $(\mathbb{I}_4 - \alpha P_{\text{raw}})\mathbf{y} = \mathbb{1}_4$ by sparse back-substitution and normalise $\mathbf{x} = \mathbf{y}/(\mathbb{1}_4^T \mathbf{y})$!

Computing PageRank at web scale – Sparse power iteration

For the real Web ($n \sim 10^{10}$ pages), how is PageRank computed in practice?

- ▶ **Direct Gaussian elimination** on $(\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n}\mathbb{1}_n$ costs $O(n^3)$ time and $O(n^2)$ memory due to fill-in, which is completely impossible for $n = 10^{10}$
- ▶ **Forming the dense Google matrix** G explicitly also requires n^2 entries because $\mathbb{J}_n$ is completely dense!
- ▶ **Matrix-free power iteration:** Starting from $\mathbf{x}^{(0)} = \frac{1}{n}\mathbb{1}_n$, for any probability vector $\mathbf{x}^{(k)}$ ($\mathbb{1}_n^T \mathbf{x}^{(k)} = 1$) we compute

$$\mathbf{x}^{(k+1)} = G\mathbf{x}^{(k)} = \alpha P\mathbf{x}^{(k)} + \frac{1-\alpha}{n}\mathbb{1}_n$$

Why is this fast?

- ▶ We **never** form G : each step only requires one sparse matrix-vector multiplication $P\mathbf{x}^{(k)}$ ($O(m)$ operations where $m \approx 10n$ is the number of hyperlinks) plus adding the scalar $\frac{1-\alpha}{n}$ to each entry!
- ▶ Since $|\lambda_2(G)| \leq \alpha = 0.85$, after $k = 50$ iterations the error is bounded by $0.85^{50} \approx 3 \times 10^{-4}$

R implementation – Checking our 3-page example

We verify our hand calculation for $\alpha = 0.5$ and $\alpha = 0.85$ in R:

```
P3 <- matrix(c(0, 0, 1,
               0.5, 0, 0,
               0.5, 1, 0), nrow = 3, byrow = TRUE)

# Solve (I - alpha * P) x = ((1 - alpha) / n) * 1 for alpha = 0.5
alpha <- 0.5
x_half <- solve(diag(3) - alpha * P3, rep((1 - alpha) / 3, 3))
MASS::fractions(x_half)

## [1] 14/39 10/39 5/13

# Compare with igraph::page_rank at alpha = 0.85
library(igraph)
g3 <- graph_from_edgelist(matrix(c(1,2, 1,3, 2,3, 3,1), ncol = 2, byrow = TRUE))
page_rank(g3, damping = 0.85)$vector
```

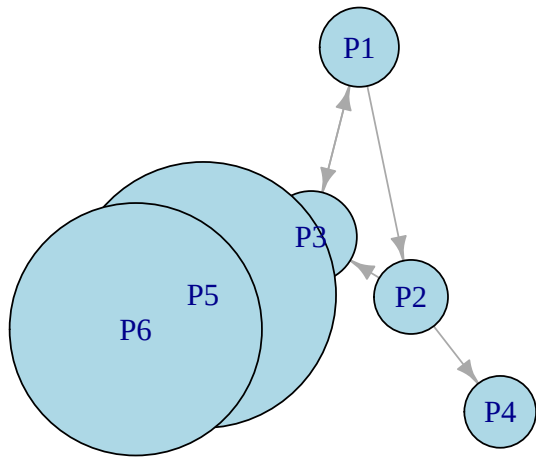
R example – 6-page web with a dangling node and spider trap

Consider a 6-page web graph where Page 4 is a dangling node ($d_4^{\text{out}} = 0$) and Pages {5, 6} form a spider trap:

```
edges <- matrix(c(1,2, 1,3, 2,3, 3,1, 2,4, 3,5, 5,6, 6,5),
               ncol = 2, byrow = TRUE)
g6 <- graph_from_edgelist(edges, directed = TRUE)
V(g6)$name <- paste0("P", 1:6)
pr6 <- page_rank(g6, damping = 0.85)$vector
round(pr6, 4)

##      P1      P2      P3      P4      P5      P6
## 0.0728 0.0646 0.0920 0.0611 0.3653 0.3442

par(mar = c(0, 0, 0, 0))
plot(g6, vertex.size = 350 * pr6 + 18, vertex.color = "lightblue",
     vertex.label.cex = 1.1, edge.arrow.size = 0.7)
```



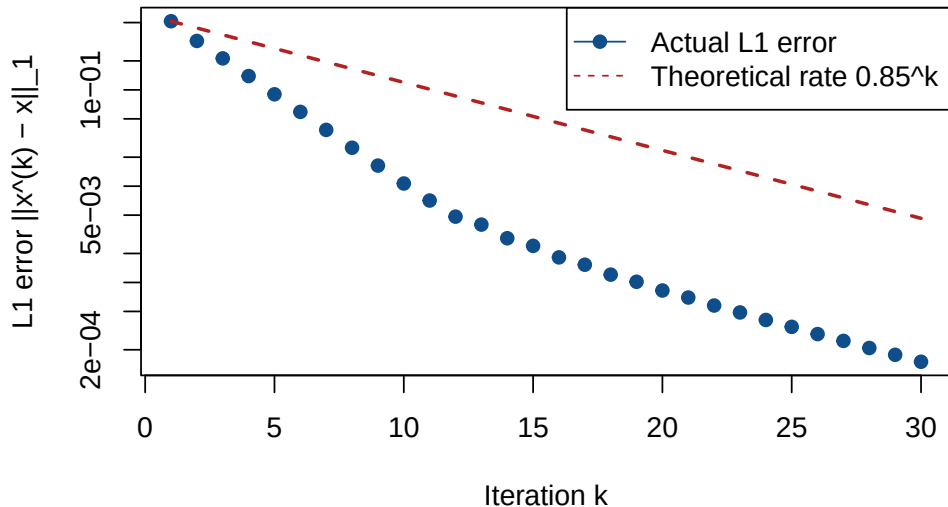
Power iteration vs linear system in R

We implement the dangling-node fix and sparse power iteration from scratch and plot convergence:

```
A6 <- as.matrix(as_adjacency_matrix(g6)) # row i -> col j in igraph, so tran
A6 <- t(A6)                               # now a_{ij} = 1 if j -> i
n <- nrow(A6)
out_deg <- colSums(A6)
P6 <- matrix(0, n, n)
for (j in 1:n) {
  if (out_deg[j] > 0) P6[, j] <- A6[, j] / out_deg[j] else P6[, j] <- rep(1
}
alpha <- 0.85
x_exact <- solve(diag(n) - alpha * P6, rep((1 - alpha) / n, n))

x_k <- rep(1 / n, n)
err <- numeric(30)
for (k in 1:30) {
```

Geometric convergence of PageRank power iteration



Summary of Unit 09

- ▶ **Absorbing Markov chains:** Reordering states so the r absorbing states precede the m transient states gives the canonical column-stochastic block form

$$P = \begin{pmatrix} \mathbb{I}_r & R \\ \mathbf{0}_{m,r} & Q \end{pmatrix}, \quad \rho(Q) < 1$$

- ▶ **Fundamental matrix:** $N = (\mathbb{I}_m - Q)^{-1} = \sum_{k=0}^{\infty} Q^k$ counts expected visits to transient states, $\mathbf{t}^T = \mathbb{1}_m^T N$ gives expected absorption times and $B = RN$ gives absorption probabilities
- ▶ **Spectral centrality:** Eigenvector centrality ($A\mathbf{x} = \lambda\mathbf{x}$) requires strong connectivity; Katz centrality ($(\mathbb{I}_n - \alpha A)\mathbf{x} = \beta\mathbb{1}_n$ for $\alpha < 1/\rho(A)$) fixes directed acyclic and source nodes by adding baseline importance
- ▶ **PageRank:** Normalising votes by out-degree and adding teleportation yields the primitive column-stochastic Google matrix $G = \alpha P + \frac{1-\alpha}{n} \mathbb{1}_n \mathbb{1}_n^T$, whose unique stationary vector solves $(\mathbb{I}_n - \alpha P)\mathbf{x} = \frac{1-\alpha}{n} \mathbb{1}_n$