

Método da Potência

Gabriela Reis
Afonso Paiva Neto

30/05/2014

Seja $\mathbf{A} \in M(n, n)$ *diagonalizável* com

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|.$$

Objetivo

Encontrar λ_1 e seu autovetor associado \mathbf{v}_1 .

Algoritmo

Dado $\mathbf{q}^{(0)} \neq \bar{\mathbf{0}}$ com $\|\mathbf{q}^{(0)}\|_2 = 1$:

$$\mathbf{z}^{(k)} = \mathbf{A}\mathbf{q}^{(k-1)}$$

$$\mathbf{q}^{(k)} = \mathbf{z}^{(k)} / \|\mathbf{z}^{(k)}\|_2$$

$$\lambda^{(k)} = \mathbf{q}^{(k)} \cdot \mathbf{A}\mathbf{q}^{(k)}$$

Lembrando

$\lambda^{(k)} \approx \lambda_1$ e $\mathbf{q}^{(k)} \approx \mathbf{v}_1$.

Algoritmo

Dado $\mathbf{q}^{(0)} \neq \bar{\mathbf{0}}$ com $\|\mathbf{q}^{(0)}\|_2 = 1$:

$$\begin{aligned} \mathbf{z}^{(k)} &= \mathbf{A}\mathbf{q}^{(k-1)} \\ \mathbf{q}^{(k)} &= \mathbf{z}^{(k)} / \|\mathbf{z}^{(k)}\|_2 \\ \lambda^{(k)} &= \mathbf{q}^{(k)} \cdot \mathbf{A}\mathbf{q}^{(k)} \end{aligned}$$

Lembrando

$\lambda^{(k)} \approx \lambda_1$ e $\mathbf{q}^{(k)} \approx \mathbf{v}_1$.

```
function [lambda,q1,iter]=power_iter(A,q0,tol)
q0 = q0/norm(q0);
z0 = A*q0;
q1 = z0/norm(z0);
iter = 1;
while (abs(abs(q0'*q1) - 1) > tol)
    q0 = q1;
    z0 = A*q0;
    q1 = z0/norm(z0);
    iter = iter + 1;
end
lambda = q1'*A*q1;
```



pagerank algorithm



Pesquisar

Aproximadamente 1.760.000 resultados (0,16 segundos)

[PageRank - Wikipedia, the free encyclopedia](#)

[en.wikipedia.org/wiki/PageRank](#) - Traduzir esta página

Ir para [Simplified algorithm](#): **PageRank** is initialized to the same value for all pages. In the original form of **PageRank**, the sum of **PageRank** over all ...

↳ [Google Panda - Google Toolbar - Topic-Sensitive PageRank - TrustRank](#)

[Google PageRank - Algorithm](#)

[pr.efactory.de/e-pagerank-algorithm.shtml](#) - Traduzir esta página

The original **PageRank algorithm** was described by Lawrence Page and ... So, first of all, we see that PageRank does not rank web sites as a whole, but is ...

[Pagerank Explained, Google's PageRank and how to make the most...](#)

[www.webworkshop.net/pagerank.html](#) - Traduzir esta página

Pagerank explained, and what you can do with it. **PageRank** calculator.

[The PageRank Algorithm - Markhorrell.com](#)

[www.markhorrell.com/seo/pagerank.html](#) - Traduzir esta página

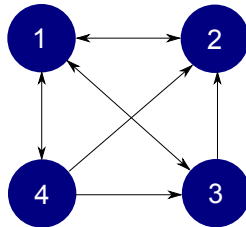
4 Dec 2002 – **PageRank** is the **algorithm** used by the Google search engine, originally formulated by Sergey Brin and Larry Page in their paper The Anatomy ...

[The Anatomy of a Search Engine](#)

[infolab.stanford.edu/~backrub/google.html](#) - Traduzir esta página

PageRank or PR(A) can be calculated using a simple iterative **algorithm**, and corresponds to the principal eigenvector of the normalized link matrix of the web.

Exemplo



- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

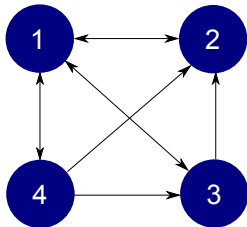
$$P_{ij} = \frac{A_{ij}}{c_j}, \quad \text{onde } c_j = \sum_{k=1}^n A_{kj}$$

- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

$$\mathbf{P}_{ij} = \frac{\mathbf{A}_{ij}}{c_j}, \quad \text{onde} \quad c_j = \sum_{k=1}^n \mathbf{A}_{kj}$$



$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{\mathbf{P}} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1/3 & 0 & 1 & 1 \\ 1/3 & 0 & 0 & 1 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1/2 & 1/3 \\ 1/3 & 0 & 1/2 & 1/3 \\ 1/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{\mathbf{P}} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1/3 & 0 & 1 & 1 \\ 1/3 & 0 & 0 & 1 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1/2 & 1/3 \\ 1/3 & 0 & 1/2 & 1/3 \\ 1/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{\mathbf{P}} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1/3 & 0 & 1 & 1 \\ 1/3 & 0 & 0 & 1 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1/2 & 1/3 \\ 1/3 & 0 & 1/2 & 1/3 \\ 1/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

$$\mathbf{P}_{ij} = \frac{\mathbf{A}_{ij}}{c_j}, \quad \text{onde} \quad c_j = \sum_{k=1}^n \mathbf{A}_{kj}$$

- Em seguida, iremos usar o método das potências em \mathbf{P} ;
 - Por quê? Porque esse autovetor dominante nos dará o PageRank!

- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

$$\mathbf{P}_{ij} = \frac{\mathbf{A}_{ij}}{c_j}, \quad \text{onde} \quad c_j = \sum_{k=1}^n \mathbf{A}_{kj}$$

- Em seguida, iremos usar o método das potências em \mathbf{P} ;
 - Por quê? Porque esse autovetor dominante nos dará o PageRank!

- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

$$\mathbf{P}_{ij} = \frac{\mathbf{A}_{ij}}{c_j}, \quad \text{onde} \quad c_j = \sum_{k=1}^n \mathbf{A}_{kj}$$

- Em seguida, iremos usar o método das potências em \mathbf{P} ;
 - Por quê? Porque esse autovetor dominante nos dará o PageRank!

- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

$$\mathbf{P}_{ij} = \frac{\mathbf{A}_{ij}}{c_j}, \quad \text{onde} \quad c_j = \sum_{k=1}^n \mathbf{A}_{kj}$$

- Em seguida, iremos usar o método das potências em \mathbf{P} ;
- De posse do autovetor dominante $\mathbf{q}^{(k)}$, podemos concluir:
 - Seja $q_i^{(k)} = \max\{q_j^{(k)} \mid j = 1, \dots, n\}$, a página i é a mais relevante.

- É definida uma matriz de conectividade \mathbf{A} :

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{se a página } j \text{ tem link para página } i \\ 0, & \text{caso contrário} \end{cases}$$

- Da matriz \mathbf{A} é criada uma matriz de probabilidade \mathbf{P} ;

$$\mathbf{P}_{ij} = \frac{\mathbf{A}_{ij}}{c_j}, \quad \text{onde} \quad c_j = \sum_{k=1}^n \mathbf{A}_{kj}$$

- Em seguida, iremos usar o método das potências em \mathbf{P} ;
- De posse do autovetor dominante $\mathbf{q}^{(k)}$, podemos concluir:
 - Seja $q_i^{(k)} = \max\{q_j^{(k)} \mid j = 1, \dots, n\}$, a página i é a mais relevante.

Calcule o PageRank usando:

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1/2 & 1/3 \\ 1/3 & 0 & 1/2 & 1/3 \\ 1/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

com $\mathbf{q}^{(0)} = (1, 0, 0, 0)^T$ e $tol = 10^{-6}$.

Solução

Temos que $\lambda_1 = 1.0000$ e seu autovetor associado $\mathbf{v}_1 \approx (0.755, 0.503, 0.3356, 0.2518)^T$. Portanto, o site 1 é o mais relevante enquanto que o site 4 é o menos relevante.

Calcule o PageRank usando:

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1/2 & 1/3 \\ 1/3 & 0 & 1/2 & 1/3 \\ 1/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

com $\mathbf{q}^{(0)} = (1, 0, 0, 0)^T$ e $tol = 10^{-6}$.

Solução

Temos que $\lambda_1 = 1.0000$ e seu autovetor associado $\mathbf{v}_1 \approx (0.755, 0.503, 0.3356, 0.2518)^T$. Portanto, o site 1 é o mais relevante enquanto que o site 4 é o menos relevante.

Teorema de Perron-Frobenius

Se a matriz $\mathbf{A} \in M(n, n)$ tem apenas entradas não negativas e o somatório de cada uma de suas colunas é 1, então:

- 1 Um dos seus autovalores é positivo e dominante;
- 2 O seu maior autovalor é 1;
- 3 Existe um único autovetor positivo correspondente ao autovalor dominante.

Método da Potência Inversa

Seja $\mathbf{A} \in M(n, n)$ *invertível*, então encontrar o menor autovalor em módulo e o seu respectivo autovetor. Lembrando que

$$|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots > |\lambda_n|.$$

Assim,

$$\mathbf{A}\mathbf{v}_n = \lambda_n\mathbf{v}_n \Rightarrow \mathbf{v}_n = \lambda_n\mathbf{A}^{-1}\mathbf{v}_n \Rightarrow \mathbf{A}^{-1}\mathbf{v}_n = \frac{1}{\lambda_n}\mathbf{v}_n$$

Se λ_n é o menor autovalor em módulo de $\mathbf{A} \Rightarrow \frac{1}{\lambda_n}$ é o maior autovalor em módulo de \mathbf{A}^{-1} .

Método da Potência Inversa

Seja $\mathbf{A} \in M(n, n)$ *invertível*, então encontrar o menor autovalor em módulo e o seu respectivo autovetor. Lembrando que

$$|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots > |\lambda_n|.$$

Assim,

$$\mathbf{A}\mathbf{v}_n = \lambda_n\mathbf{v}_n \Rightarrow \mathbf{v}_n = \lambda_n\mathbf{A}^{-1}\mathbf{v}_n \Rightarrow \mathbf{A}^{-1}\mathbf{v}_n = \frac{1}{\lambda_n}\mathbf{v}_n$$

Se λ_n é o menor autovalor em módulo de $\mathbf{A} \Rightarrow \frac{1}{\lambda_n}$ é o maior autovalor em módulo de \mathbf{A}^{-1} .

Seja $\mathbf{A} \in M(n, n)$ *invertível*, então encontrar o menor autovalor em módulo e o seu respectivo autovetor. Lembrando que

$$|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots > |\lambda_n|.$$

Assim,

$$\mathbf{A}\mathbf{v}_n = \lambda_n\mathbf{v}_n \Rightarrow \mathbf{v}_n = \lambda_n\mathbf{A}^{-1}\mathbf{v}_n \Rightarrow \mathbf{A}^{-1}\mathbf{v}_n = \frac{1}{\lambda_n}\mathbf{v}_n$$

Se λ_n é o menor autovalor em módulo de $\mathbf{A} \Rightarrow \frac{1}{\lambda_n}$ é o maior autovalor em módulo de \mathbf{A}^{-1} .

Algoritmo

Dado $\mathbf{q}^{(0)} \neq \bar{\mathbf{0}}$ com $\|\mathbf{q}^{(0)}\|_2 = 1$

$$\mathbf{z}^{(k)} = \mathbf{A}^{-1}\mathbf{q}^{(k-1)} \Rightarrow \text{resolva o sist. linear } \mathbf{A}\mathbf{z}^{(k)} = \mathbf{q}^{(k-1)}$$

$$\mathbf{q}^{(k)} = \mathbf{z}^{(k)} / \|\mathbf{z}^{(k)}\|_2$$

$$\lambda^{(k)} = \mathbf{q}^{(k)} \cdot \mathbf{A}\mathbf{q}^{(k)}$$

Lembrando

$$\lambda^{(k)} \approx \lambda_n \text{ e } \mathbf{q}^{(k)} \approx \mathbf{v}_n.$$

Algoritmo

Dado $\mathbf{q}^{(0)} \neq \bar{\mathbf{0}}$ com $\|\mathbf{q}^{(0)}\|_2 = 1$

$$\mathbf{z}^{(k)} = \mathbf{A}^{-1}\mathbf{q}^{(k-1)} \Rightarrow \text{resolva o sist. linear } \mathbf{Az}^{(k)} = \mathbf{q}^{(k-1)}$$

$$\mathbf{q}^{(k)} = \mathbf{z}^{(k)} / \|\mathbf{z}^{(k)}\|_2$$

$$\lambda^{(k)} = \mathbf{q}^{(k)} \cdot \mathbf{Aq}^{(k)}$$

Lembrando

$$\lambda^{(k)} \approx \lambda_n \text{ e } \mathbf{q}^{(k)} \approx \mathbf{v}_n.$$

Algoritmo

Dado $\mathbf{q}^{(0)} \neq \bar{\mathbf{0}}$ com $\|\mathbf{q}^{(0)}\|_2 = 1$

$$\mathbf{z}^{(k)} = \mathbf{A}^{-1}\mathbf{q}^{(k-1)} \Rightarrow \text{resolva o sist. linear } \mathbf{Az}^{(k)} = \mathbf{q}^{(k-1)}$$

$$\mathbf{q}^{(k)} = \mathbf{z}^{(k)} / \|\mathbf{z}^{(k)}\|_2$$

$$\lambda^{(k)} = \mathbf{q}^{(k)} \cdot \mathbf{Aq}^{(k)}$$

Lembrando

$\lambda^{(k)} \approx \lambda_n$ e $\mathbf{q}^{(k)} \approx \mathbf{v}_n$.

```
function [lambda,q1,iter]=powerinv_iter(A,q0,tol)
q0 = q0/norm(q0);
z0 = A\q0;
q1 = z0/norm(z0);
iter = 1;
while(abs(abs(q0'*q1) - 1) > tol)
    q0 = q1;
    z0 = A\q0;
    q1 = z0/norm(z0);
    iter = iter + 1;
end
lambda = q1'*A*q1
```

Encontrar o menor autovalor em módulo e seu respectivo autovetor, sendo:

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 1/2 & 1/3 \\ 1/3 & 0 & 1/2 & 1/3 \\ 1/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 0 \end{bmatrix}$$

com $\mathbf{q}^{(0)} = (1, 0, 0, 0)^T$ e $tol = 10^{-6}$.