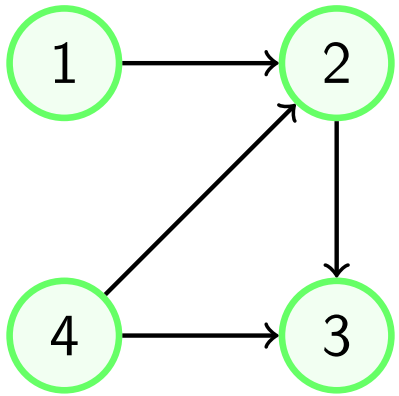
PageRank és kiszámolása

Gáspár Tamás

2019 május 30.

A PageRank definíciója

A PageRank egy Sergei Brin és Larry Page által alkotott módszer arra, hogy weboldalak egy halmazát a köztük lévű linkek alapján rangsoroljuk.
A weboldalak halmaza (röviden web) legyen egy irányított gráf, ahol az irányított élek fejezik ki az oldalak közti linkeket.



Egy 4 oldalból álló web

A PageRank definíciója

1. Definíció: Weboldal PageRankja. Legyen adott egy web, ahol V az oldalak halmaza. Legyen $v_i \in V$ oldal PageRankja $r(v_i)$, az oldalról kimenő linkek száma pedig $|v_i|$. Jelölje $B_i \subset V$ azon oldalak halmazát amelyen linkelnek v_i-re.
Ekkor v_i oldal PageRank-ja definíció szerint:

$$r(v_i) = \sum_{v_j \in B_i} \frac{r(v_j)}{|v_j|}.$$

Egy oldal fontossága azon múlik, hogy mennyi oldal linkel rá, és hogy ezek milyen fontosak.
Megjegyzés: Rekurzív definíció.

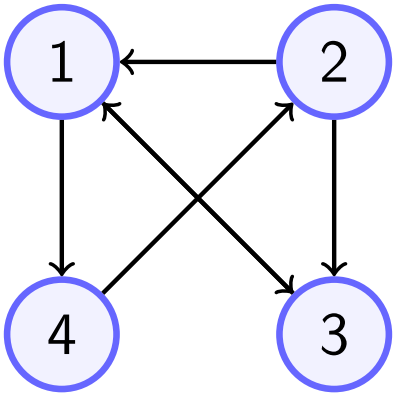
A linkmátrix

Az oldalak között kapcsolatot mutatja.

2. Definíció: Linkmátrix.

$$a_{i,j} = \begin{cases} \frac{1}{|v_i|}, & \text{ha } v_i \text{ linkel } v_j\text{-re,} \\ 0, & \text{egyébként.} \end{cases}$$

Egy példa: web és a hozzá tartozó linkmátrix.



$$\begin{pmatrix} 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Webhez rendelhető markov lánc

Megjegyzés: A linkmátrix sztochasztikus lesz (kivétel: ha van *"lógó oldal"*, de ez egyszerű helyettesítéssel megszüntethető).

Minden webhez rendelhetünk egy Markov-láncot, melynek állapotai az oldalak, átmenetvalószínűségei pedig a linkmátrix megfelelő elemei (ez lesz az átmenetmátrix).

A *"véletlen szörföző"* egyenletes eloszlás szerint halad az oldalon lévő linkeken.

Definíció és a linkmátrix kapcsolata, PageRank vektor

Kapcsolat van a linkmátrix és a definícióból kapott egyenletrendszer között. Legyen $\mathbf{x}$ az oldalak PageRank-jaiból álló sorvektor, ekkor:

$$\mathbf{x} = \mathbf{x}A,$$

ami átalakítva

$$\mathbf{x}^T = A^T\mathbf{x}^T.$$

A fenti egyenlet mutatja, hogy a $\mathbf{x}$ a webhez rendelt markov lánc invariáns eloszlása lesz, a lenti pedig azt, hogy az 1-hez tartozó sajátvektor is (az A^T mátrixban).

Megjegyzés: $\mathbf{x}$-et vegyük úgy, hogy a komponensek összege 1.

A hatványiteráció

Hogyan számoljuk ki **x**-et? A pontos eredmény nem kivitelezhető, túl sok oldal. A megoldás a hatványiteráció:

$$\mathbf{x}^{i+1} = A^T \mathbf{x}^i$$

Probléma: A hatványiteráció a domináns sajátértékhez tartozó sajátvektorhoz konvergál, ilyen nem biztos hogy van (például -1, vagy 1 többszörös multiplicitással). Különálló *"szubwebek"* esetén fordul elő.

A linkmátrixon (és a Markov-láncon) módosítani kell úgy, hogy a domináns sajátérték garantált legyen.

A linkmátrix módosítása: Google mátrix

Megközelítés a Markov-láncok irányából: Ha a Markov-lánc irreducibilis és aperiodikus, akkor biztosan egyértelmű lesz az invariáns eloszlás.

3. Definíció: Google mátrix. Legyen $A \in \mathbb{R}^{n \times n}$ egy web linkmátrixa, S pedig egy egy olyan $n \times n$-es mátrix, melynek minden eleme $\frac{1}{n}$. Ekkor a Google mátrix definíció szerint

$$G := \alpha A + (1 - \alpha)S, \qquad \alpha \in [0, 1]$$

Példa: Egy linkmátrix és az $\alpha = 0,7$ paraméterrel kapott Google mátrix.

$$A = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & 0 & 1 & 0 \end{pmatrix}, G = \begin{pmatrix} 0{,}038 & 0{,}463 & 0{,}463 & 0{,}038 \\ 0{,}321 & 0{,}038 & 0{,}321 & 0{,}321 \\ 0{,}321 & 0{,}321 & 0{,}38 & 0{,}321 \\ 0{,}038 & 0{,}038 & 0{,}888 & 0{,}038 \end{pmatrix}$$

A Google mátrix

A Google mátrix szintén sztochasztikus, és a szükséges tulajdonságokat is teljesíti. A hatványiteráció (G transzponáltján végezve) tehát mindig PageRank vektorhoz fog konvergálni.

Kérdés: A konvergencia sebessége. A második legnagyobb sajátértéktől (λ_2) függ. Minél közelebb van ez 1-hez, annál lassabb lesz.

Megjegyzés: *(A helyettesítés mögötti heurisztika.)*
A véletlen szörföző most már nem csak a linkeken keresztül juthat el a következő oldalra, hanem $1 - \alpha$ valószínűséggel egy egyenletes eloszlás szerint választott véletlen oldalra ugrik.

Az α paraméter jelentősége

$$G := \alpha A + (1 - \alpha)S, \qquad \alpha \in [0, 1]$$

Minden Google mátrixra és α-ra: $\lambda_2 \leq \alpha$ (bizonyos feltételek* mellett $\lambda_2 = \alpha$).

Tétel

Ha a Google mátrix sajátértékei $1 > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$, *akkor* $\lambda_2 \leq \alpha$.

Tétel

Ha a Google mátrix sajátértékei $1 > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$ *és A-ban van legalább kettő irreducibilis zárt részhalmaz, akkor* $\lambda_2 = \alpha$.

Az α paraméter jelentősége

α hatása a konvergencia sebességére:

- Nagy α lelassítja konvergencia sebességet.
- Kis α elnyomja a linkek jelentőségét (a véletlen szörföző $1-\alpha$ valószínűséggel nem a linkeken halad tovább).

A pontos sebesség:

Tétel

Legyen $\boldsymbol{\pi}$ a PageRank vektor, $\mathbf{x}^i$ pedig a Google mátrixra alkalmazott hatványiteráció i. lépésének erdeménye. Ekkor $\exists C \in \mathbb{R}$:

$$\|\mathbf{x}^i - \boldsymbol{\pi}\| \leq C\alpha^i, \quad i = 1, 2, \ldots$$

Algoritmus a PageRank vektor meghatározására

1. Adjuk meg az (n oldalból álló) webhez tartozó linkmátrixot.
2. Végezzük el a helyettesítést a *"lógó oldalak"* kiküszöbölésére, azaz ha a linkmátrixban van csupa nulla sor, azt cseréljük le $1/n$ elemekből álló sorra.
3. Válasszunk egy $\alpha \in (0, 1)$ paramétert (kompromisszum). Konstruáljuk meg a Google mátrixot ezzel a paraméterrel:

$$G = \alpha A + (1 - \alpha)S,$$

4. Végezzük a hatványiterációt a G^T mátrixon, amíg a két iterációs lépés közötti különbség nem csökken valami kicsi ϵ érték alá, vagy adjunk meg konkrét iterációs lépésszámot.
5. Az iteráció eredménye a PageRank vektor közelítése, ahol az i. komponens az i. oldal fontosságát adja meg.