Четврти семинар **„Математика и примени“**, 13-14 декември 2019
Институт за математика, Природно-математички факултет,
Универзитет „Св. Кирил и Методиј“, Скопје

Алгоритмот Google PageRank

Адмир Хусеини,Ардиан Синани

План на презентацијата

Вовед

Како да препознаете кои страници се важни
Графови
Сопствени вектори

PageRank централност и метод на степенување

PageRank централност на темиња
PageRank како сопствен вектор
Алгоритмот за рангирање на интернет страници
Метод на степенување
Конвергенција на методот на степенување

Матрицата PageRank и случајни процеси

Пребарувањето како случаен процес
Веројатносна модификација на матрицата PageRank
Имплементацијата на Google

Анегдоти од докторските студии на Лери Пејџ

Кои страници се релевантни

Google тврди дека индексира 25 милијарди страници.

За повеќето пребарувања, има огромен број страници што ги содржат зборовите во фразата за пребарување.

Она што е потребно е сортирање со најважните страници на врвот на листата

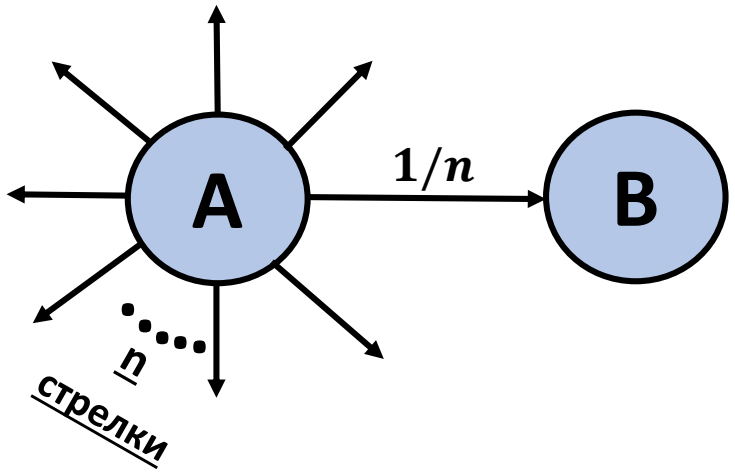
Централен дел на софтверот на Google

Алгоритмот **PageRank** на Google ја оценува **важноста** на интернет-страниците без човечка проценка на содржината.

Google тврди дека "**PageRank е центален дел на нашиот софтвер**."

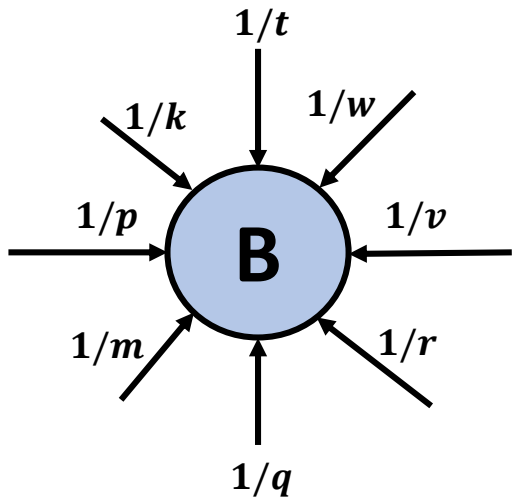
Основната идеја на PageRank е дека важноста на интернет страницата се оценува според бројот и важноста на страниците **што се поврзани** со неа.

Како да препознаете која страница е важна



Страницата A има n линкови.

Ако една од тие врски е кон страницата B, тогаш A ќе даде $1/n$ од нејзината важност на B



Важноста на B е збирот на сите придонеси направени од страниците кои се поврзуваат со него.

Графови

Интернет граф

$G = (V, E)$,

$V = \{1, \ldots, n\}$, i индексот на дадена интернет страница

$E = \{(i, j) \mid i, j \in V, \text{постои хиперлинк од } i \text{ до } j\}$

Матрица на соседство

$$\boldsymbol{A_{ij}} = \begin{cases} \mathbf{1}, & \text{ако } (i,j) \in \boldsymbol{E} \\ \mathbf{0}, & \text{спротивно} \end{cases}$$

PageRank матрица

$$\boldsymbol{H_{ij}} = \begin{cases} \dfrac{\mathbf{1}}{\boldsymbol{deg}(\boldsymbol{v_i})}, & \text{ако } (i,j) \in \boldsymbol{E} \\ \mathbf{0}, & \text{спротивно} \end{cases}$$

Сопствени вектори

Сопствена вредност $\lambda \in \mathbb{R}$

За дадена матрица $A \in \mathbb{R}^{n \times n}$, сопствена вредност $\lambda \in \mathbb{R}$ е скалар, за кој постои вектор $v \in \mathbb{R}^n$ кој за ја испонува равенката

$$A \cdot v = \lambda \cdot v$$

Сопствен вектор $v \in \mathbb{R}^n$

Вектор v кој за дадена матрица $A \in \mathbb{R}^{n \times n}$ и сопствена вредност $\lambda \in \mathbb{R}$ ја исполнува равенката

$$A \cdot v = \lambda \cdot v$$

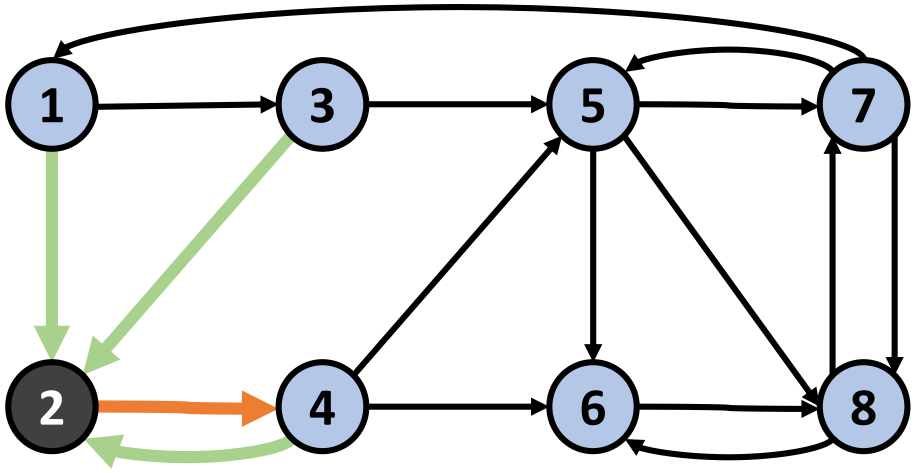
Доминантни сопствени вектори и вредности

Ако $\lambda_1, \lambda_2, \ldots, \lambda_n \in \mathbb{R}$ се сопствени вредности на матрицата $A \in \mathbb{R}^{n \times n}$, тогаш $\lambda_1 \in \mathbb{R}$ се вика **доминантна сопствена вредност** на A ако

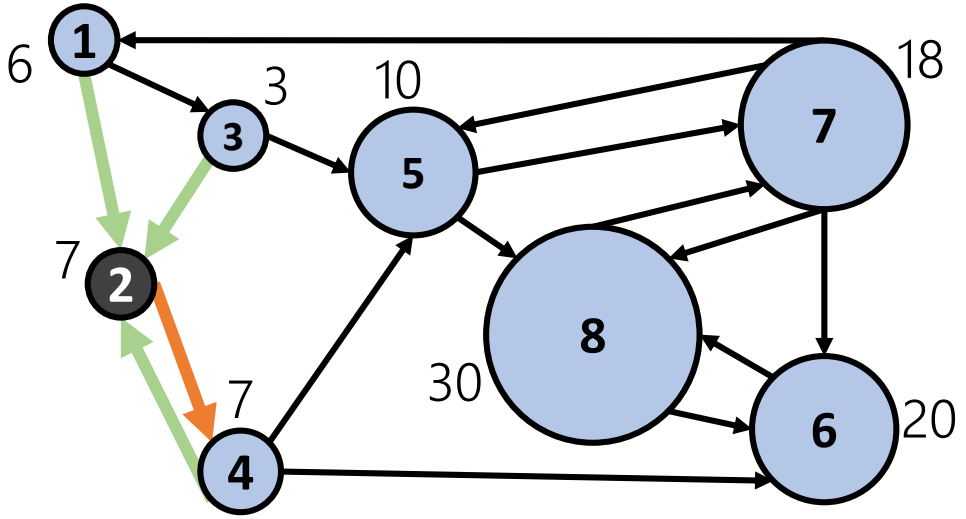
$$|\lambda_1| > |\lambda_i| \quad , i = 2, \ldots, n$$

Сопствениот вектор кој одговара на λ_1 се нарекува **доминантен сопствен вектор**.

Пример



Интернет граф G



Графот G нацртан според вредностите на централност

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

Матрицата на соседство

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

PageRank матрицата

$$I = \begin{bmatrix} 0.0600 \\ 0.0675 \\ 0.0300 \\ 0.0675 \\ 0.0975 \\ 0.2025 \\ 0.1800 \\ 0.2950 \end{bmatrix} \qquad 100I = \begin{bmatrix} 6 \\ 7 \\ 3 \\ 7 \\ 10 \\ 20 \\ 18 \\ 30 \end{bmatrix}$$

Сопствениот вектор – нормализиран и скалиран

PageRank централност на темиња

$$p(v_i) = \frac{1}{\lambda} \sum_{v_j \epsilon B_i} \frac{p(v_j)}{I_j}$$

$p(v_i)$ - Централноста на темето P_i

v_j - Бројот на соседи на темето P_j

B_i - Множеството на сите соседи на P_i

λ - Константна вредност

PageRank централноста $p(v_i)$ на темето v_i е еднакво на сумата на PageRank централностите $p(v_j)$ на сите негови соседи v_j кои покажуваат кон него поделено со бројот на нивните соседи I_j, коригиран со константен фактор λ.

PageRank како сопствен вектор

Да ги запишеме вредностите $\mathrm{p}(v_i)$ во вектор $\mathrm{p} = \left(\mathrm{p}(v_i), \mathrm{p}(v_2), \ldots, \mathrm{p}(v_n)\right)$

Тогаш од дефиницијата на PageRank за овој вектор следува

$$p_i = \frac{1}{\lambda} \sum_{P_j \epsilon B_i} \frac{p_j}{I_j} \Leftrightarrow \lambda \cdot p_i = \sum_{\mathrm{j}=1}^{n} H_{ij} p_j$$

$$Hp = \lambda \cdot p$$

Каде H е PageRank матрицата а p е векторот на PageRank централности

Алгоритмот за рангирање на интернет страници

Вредностите на централноста на дадено теме во интернет графот се еднакви на **компонентите на сопствен вектор** на PageRank матрицата.

Теоремата на Перон-Фробениус тврди дека ако дадена матрица има само позитивни вредности, тогаш таа има единствена доминантна вредност и нејзиниот доминантен вектор има само позитивни вредности.

Според тоа, проблемот на наоѓање на важноста на интернет страниците се редуцира на проблемот на наоѓање на доминантниот сопствен вектор на PageRank матрицата.

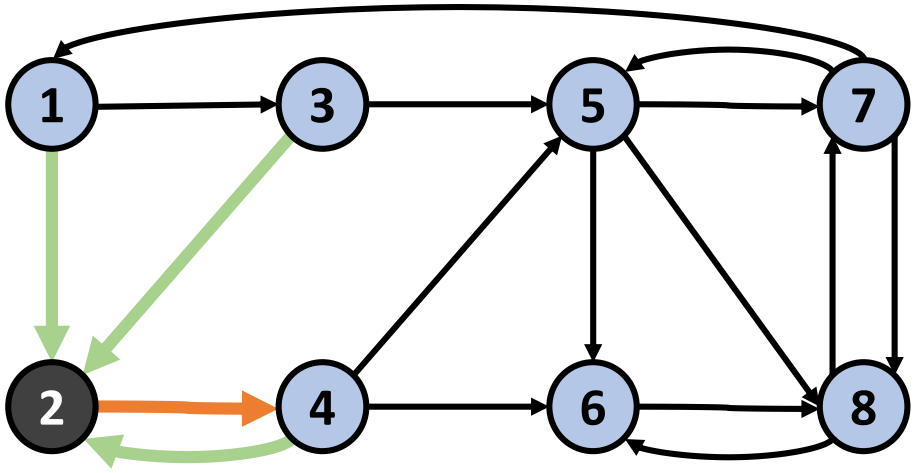
Алгоритмот за рангирање на интернет страници

Анализа на хиперлинковите

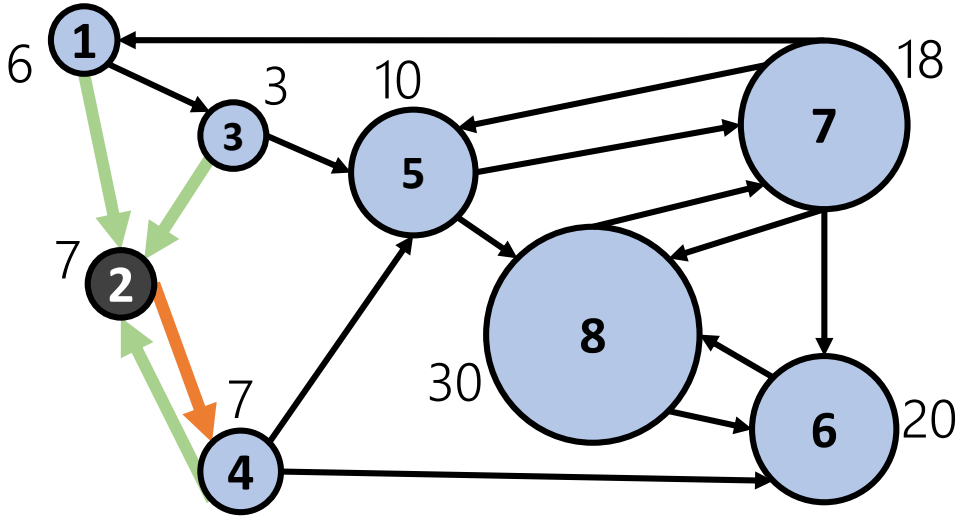
Градење на интернет и PageRank матриците

Наоѓање на доминантниот сопствен вектор на PageRank матрицата

Пример



Интернет граф G



Графот G нацртан според вредностите на централност

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

Матрицата на соседство

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

PageRank матрицата

$$I = \begin{bmatrix} 0.0600 \\ 0.0675 \\ 0.0300 \\ 0.0675 \\ 0.0975 \\ 0.2025 \\ 0.1800 \\ 0.2950 \end{bmatrix} \qquad 100I = \begin{bmatrix} 6 \\ 7 \\ 3 \\ 7 \\ 10 \\ 20 \\ 18 \\ 30 \end{bmatrix}$$

Сопствениот вектор – нормализиран и скалиран

Метод на степенување

Како да се пресмета доминантниот сопствениот вектор за дадена матрица A?

Метод на степенување

Ако матрицата A има доминантна сопствена вредност тогаш итерациите

$$x_0 \in \mathbb{R}^n$$
$$x_1 = \mathrm{A} \cdot x_0$$
$$x_2 = \mathrm{A} \cdot x_1 = \mathrm{A}^2 \cdot x_0$$
$$\dots$$
$$x_n = \mathrm{A} \cdot x_{n-1} = \mathrm{A}^n \cdot x_0$$

конвергираат кон доминантната сопствена вредност на A.

Метод на степенување

- **Идеја на доказот**
  - На место да дадеме екзактен доказ, ќе ја дадеме идејата на методот.
  - За матрица со n различни сопствени вектори $x_1, x_2, \dots, x_n \in \mathbb{R}^n$ за n различни сопствени вредности $\lambda_1, \lambda_2, \dots, \lambda_n \in \mathbb{R}$ подредени според големина, каде $|\lambda_{i+1}| > |\lambda_i|$
  - Бидејќи сопствените вектори $x_1, x_2, \dots, x_n$ се линеарно независни, тие формираат база на $\mathbb{R}^n$, затоа било кој вектор $x_0 \in \mathbb{R}^n$ може да се запише како

$$x_0 = c_1 \cdot x_1 + c_2 \cdot x_2 + \cdots + c_n \cdot x_n$$

$$A \cdot x_0 = c_1 \cdot (A \cdot x_1) + c_2 \cdot (A \cdot x_2) + \cdots + c_n \cdot (A \cdot x_n)$$

$$A \cdot x_0 = c_1 \cdot (\lambda_1 \cdot x_1) + c_2 \cdot (\lambda_2 \cdot x_2) + \cdots + c_n \cdot (\lambda_n \cdot x_n)$$

$$A^k \cdot x_0 = c_1 \cdot \left(\lambda_1^k \cdot x_1\right) + c_2 \cdot \left(\lambda_2^k \cdot x_2\right) + \cdots + c_n \cdot \left(\lambda_n^k \cdot x_n\right)$$

Метод на степенување

- **Идеја на доказот**
  - Последната равенка може да се запише како

$$A^k \cdot x_0 = \lambda_1^k \cdot \left[c_1 \cdot x_1 + c_2 \cdot \left(\frac{\lambda_2}{\lambda_1}\right)^k \cdot x_2 + \cdots + c_n \cdot \left(\frac{\lambda_n}{\lambda_1}\right)^k \cdot x_n \right]$$

  - Бидејќи $\frac{\lambda_2}{\lambda_1}, \frac{\lambda_3}{\lambda_1}, \ldots, \frac{\lambda_n}{\lambda_1}$ се помали од 1 по апсолутна вредност

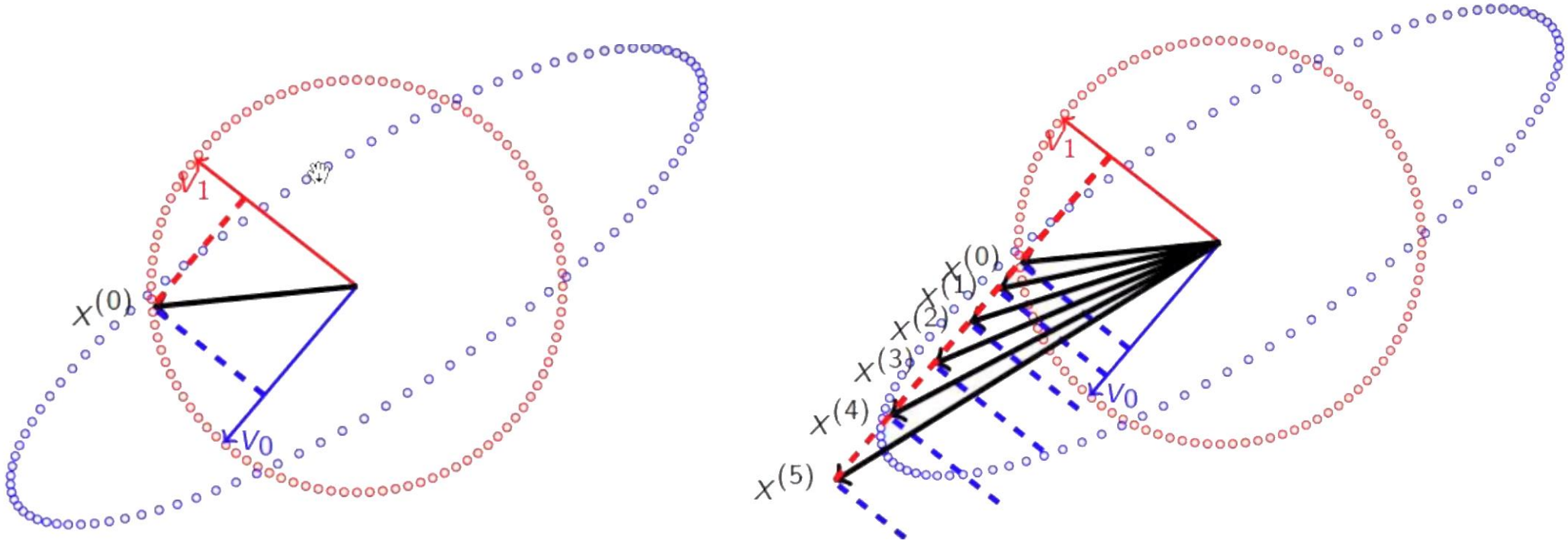
$$\left(\frac{\lambda_2}{\lambda_1}\right)^k, \left(\frac{\lambda_3}{\lambda_1}\right)^k, \ldots, \left(\frac{\lambda_n}{\lambda_1}\right)^k$$

    тие конвергираат кон 0 кога $k \to \infty$.
  - Според тоа

$$A^k \cdot x_0 \approx \lambda_1^k \cdot c_1 \cdot x_1, \qquad c \neq 0$$

    што значи дека $A^k \cdot x_0$ конвергира кон доминантниот сопствен вектор.

Метод на степенување



Слика 1. Визуализација на методот на степенување.
Со секоја итерација векторот $x^{(0)}$ го апроксимира доминантниот сопствен вектор v_0.

Кога работите ќе тргнат наопаку

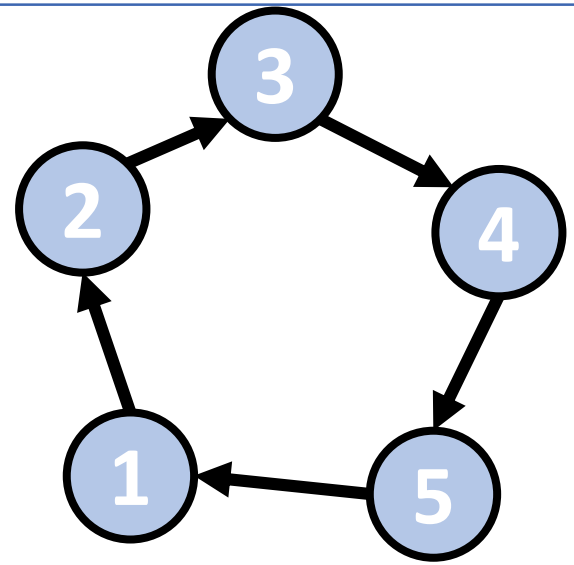
Да претпоставиме дека нашиот интернет граф изгледа како во сликата. Во овој случај, матрицата S е дадена подолу.

Графот на сликата претставува „граф прстен“

Матрицата S има 5 сопствени вектори чии сопствени вредности имаат иста апсолутна вредност еднаква на 1.

Методот на степени нема да конвергира.

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$



Граф прстен и неговата матрица S .

Веројатносна интерпретација на H

Замислете дека ние сурфаме на интернет по случаен избор. Кога се наоѓаме на дадена интернет-страница, ние по случаен избор следиме една од нејзините врски до друга страница.

На пример, ако сме на страницата P_j со l_j линкови, од кои една од нас нè води кон страницата P_i, веројатноста дека следниот пат завршуваме на страницата P_i е

$$T_i = \sum_{P_j \epsilon B_i} T_j / I_j$$

Ова ни овозможува да ја интерпретираме PageRank централноста на интернет страната како дел од времето што случајниот сурфер го поминува на таа интернет-страница.

Конечна модификација

За да ја направиме нашата модификација, ние прво ќе избереме параметар α помеѓу 0 и 1. Сега претпоставиме дека нашиот случаен сурфер се движи на малку поинаков начин.

Со веројатност α тој е воден од S. Со веројатност $1-\alpha$, тој ја избира следната страница по случаен избор.

Ако ние ја означиме со 1 матрицата $n\ x\ n$ чии записи се сите единици, ја добиваме матрицата на Google:

$$G = \alpha S + (1-\alpha)\frac{1}{n}1$$

Имплементацијата на Google

Google графот се состои од милиони редици и колони со редок број на единици.
Не постои компјутер на светот, кој има доволна меморија за го изврши множењето на оваа матрица со вектор.

Како да се имплементира ефикасно множење за оваа огромна матрица?

Подели и освои(divide and conquer) со блок матрици

$$A \cdot X = \begin{pmatrix} E & F \\ G & H \end{pmatrix} \cdot \begin{pmatrix} S \\ T \end{pmatrix} = \begin{pmatrix} E \cdot S + F \cdot T \\ G \cdot S + H \cdot T \end{pmatrix}$$

Ова разложување може да се итерира, такашто на пример матриците E, F, G, H се поделуваат на уште помали блок матрици.
Секое множење се прави паралелно во различен компјутер и резултатите се агрегираат.
Google инвестира огромни суми во ефикасен хардвер за изведување на овие множења.

Анекдоти за Лери Пејџ (1973 -)

Родителите на Лери биле “Computer geek”-ови. Од нив потекнува неговата инспирација за технологија.

Во основно училиште бил единствениот ученик што ги запишувал задачите на Word.

Пејџ имал 23 години кога го започнал проектот за Google, како дел од неговите докторски студии. Идеата за пребарувачката машина, буквално му дошла во сон.

Проектот бил наречен “BackRub” поради особината да ги следи линковите насочени кон дадена страница.

Името Google се појавило кога еден помошник погрешно го напишал зборот “Googol” (=10 на степен 100)