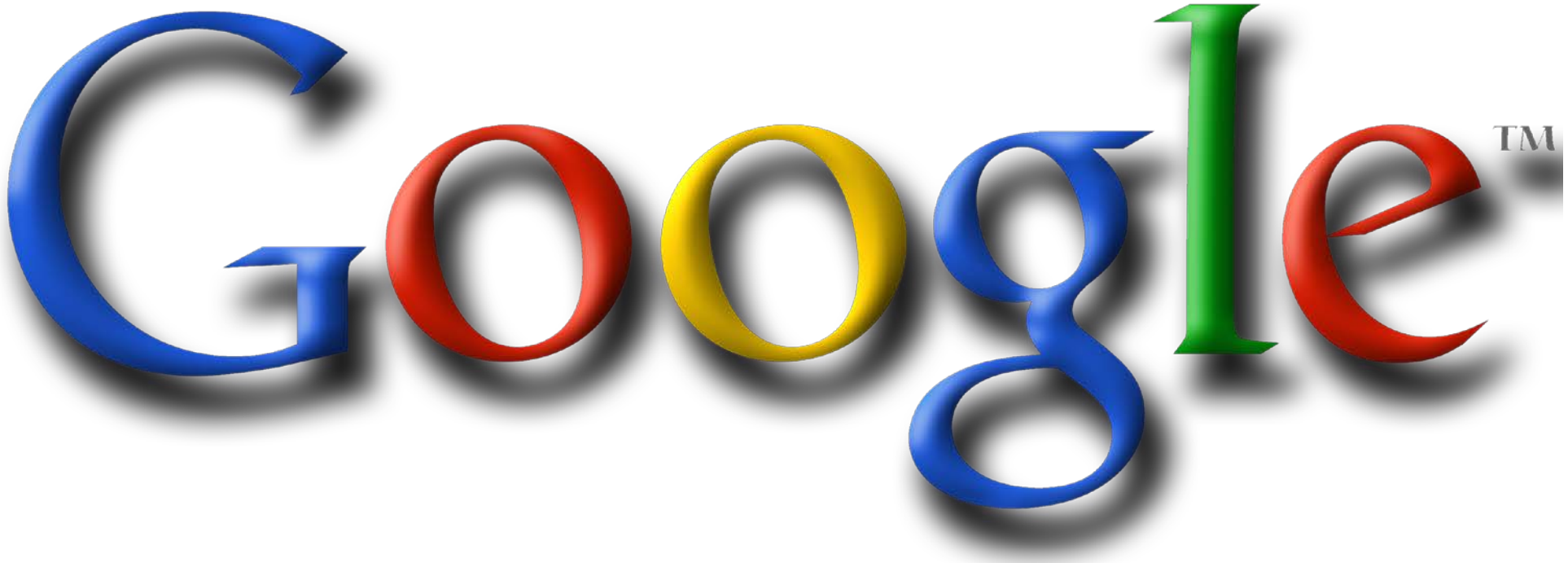
Numerical Linear Algebra – PageRank

CS370 Lecture 26 – March 15, 2017

Page Rank: The Not-So secret sauce behind Google.

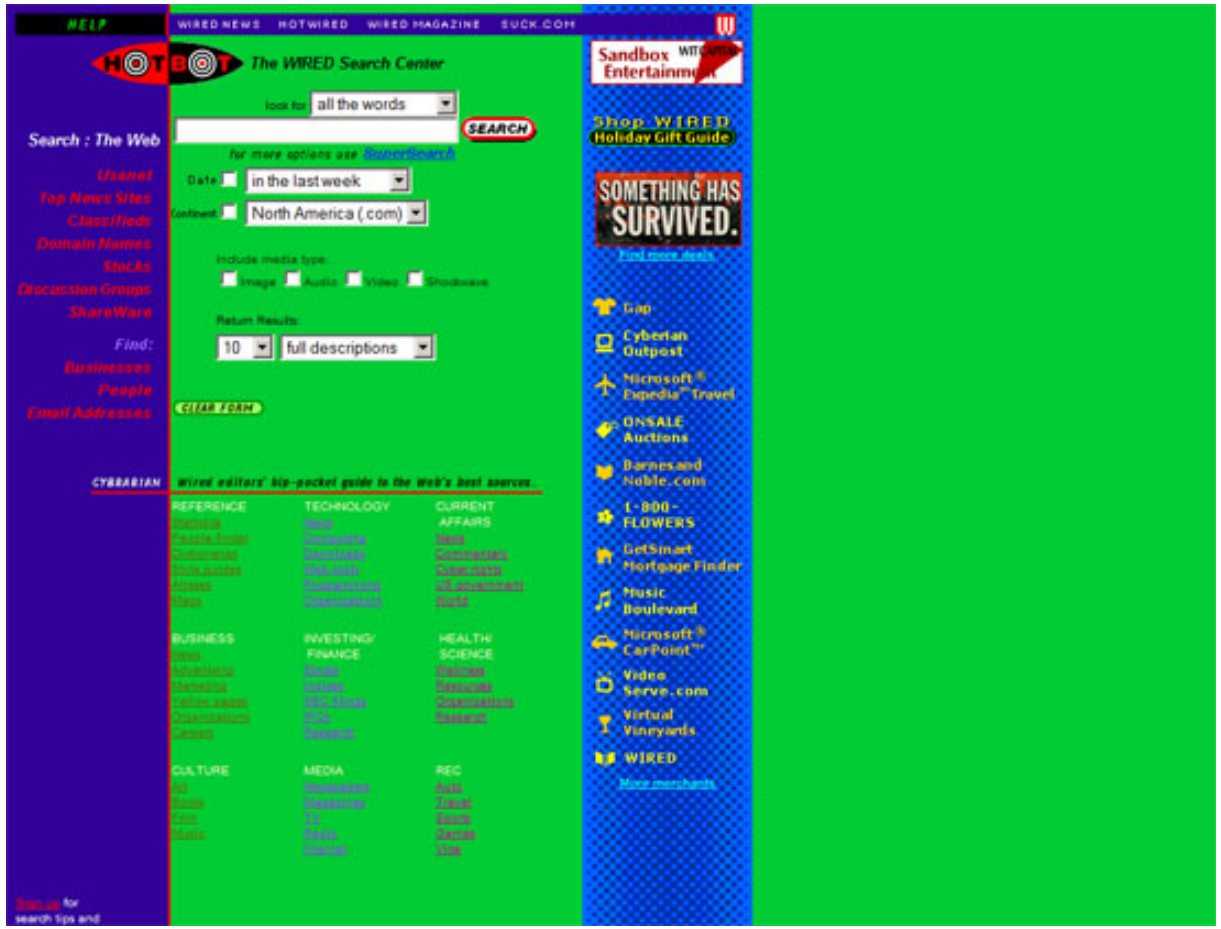


A Bit of History

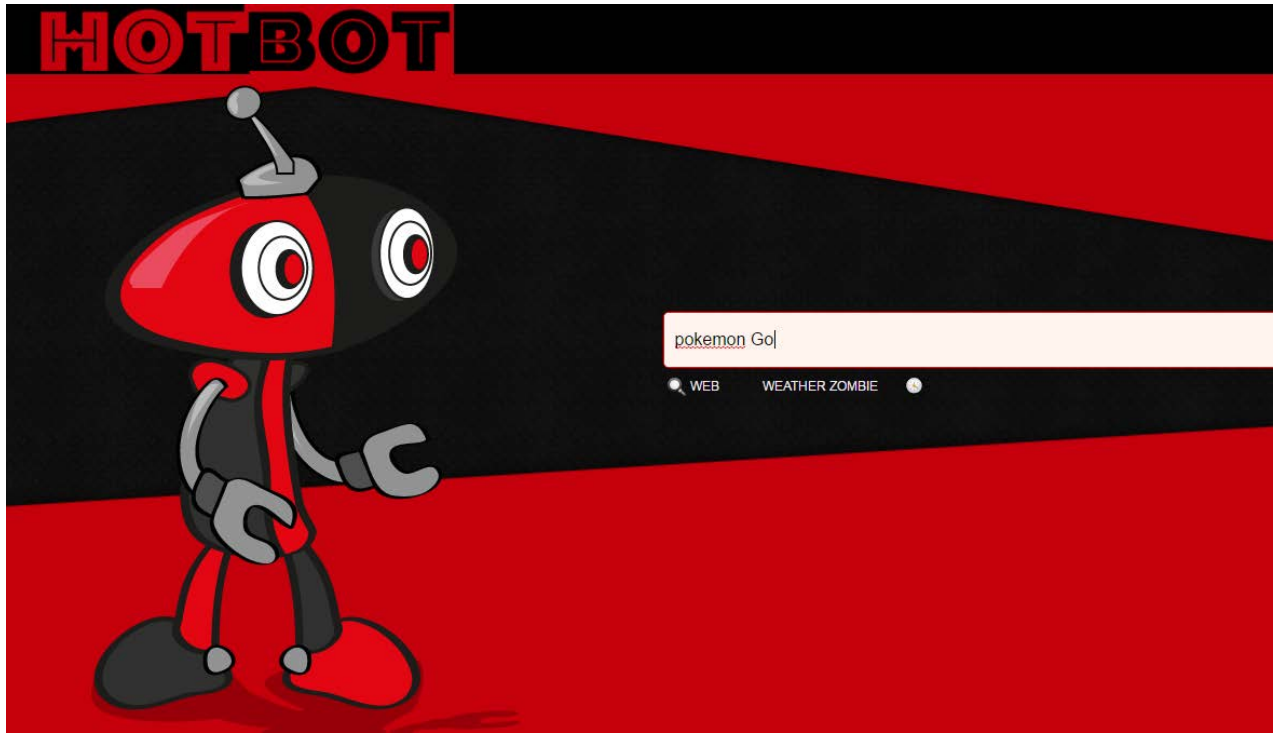
Back when I was a boy…

In the mid-90's, search engines were awful (not to mention hideous).

My favourite: HotBot



Actually still exists; (a bit) less ugly!



A Bit of History – Ranking Strategy

Websites were mostly ranked by counting search keywords appearing on each site, with some variations.

- Clearly easy to “cheat”.
- Often gave poor results.
- Alternative: Yahoo was a big hand-curated directory, essentially.

No consistently dominant search engine.

A Bit of History – The Dawn of Google

David Cheriton invested $200K in Google that year.

Around 1998, along came Stanford PhD students, Sergey Brin and Larry Page.

Rough idea: If many web pages link ***to*** your website, there must be some consensus that it is important.

An analogy…

Good indicator:

Everybody else tells you *Joe's Used Cars* is trustworthy.

Poor indicators:

- Joe himself constantly tells you he's honest.
- Joe publishes ads saying he's reliable.

Analogy: Paper Citations

Academic productivity is (sometimes, partially) measured similarly.

- I write a research paper, referencing/citing relevant earlier papers.
- If my paper later gets cited ***many times*** by other people's papers, this provides some indication that my paper was itself influential.

This provided some inspiration.

e.g. The original paper describing PageRank now has ~~7613~~ ~~9525~~ 10317 citations (Google Scholar).

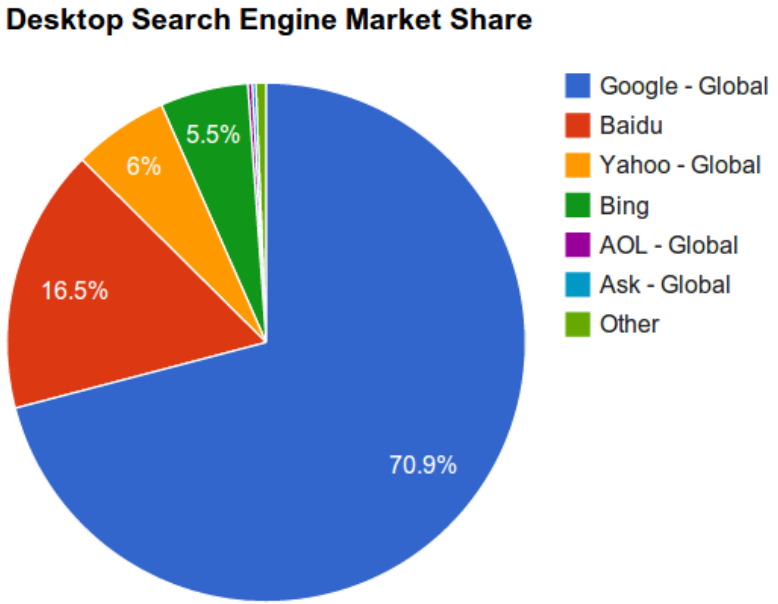
The **PageRank** citation ranking: Bringing order to the web.
L Page, S Brin, R Motwani, T Winograd - 1999 - ilpubs.stanford.edu
The importance of a Web page is an inherently subjective matter, which depends on the readers interests, knowledge and attitudes. But there is still much that can be said objectively about the relative importance of Web pages. This paper describes **PageRank**, a
Cited by 10318 Related articles All 19 versions Cite Save

Google Homepage in September 1998

Page and Brin launched Google to commercialize this PageRank idea.

Google rapidly took over the search engine market due to its superior results.

That was the end of the story until 2009…

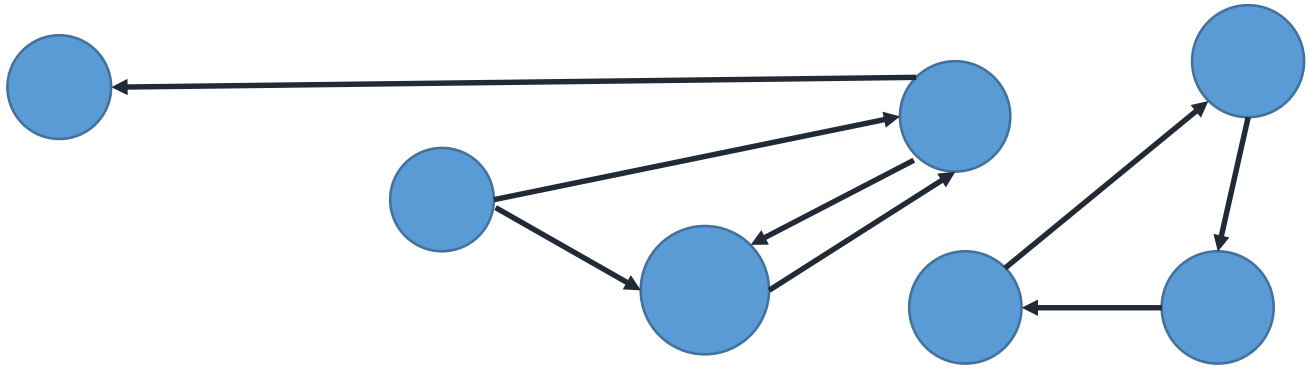


And then Bing came along.

And nothing much changed.

PageRank: From Links to Importance

Clearly the link structure between pages provides *some* useful indicator.



Page and Brin turned this vague notion into a concrete ***importance metric***, using tools from numerical linear algebra.

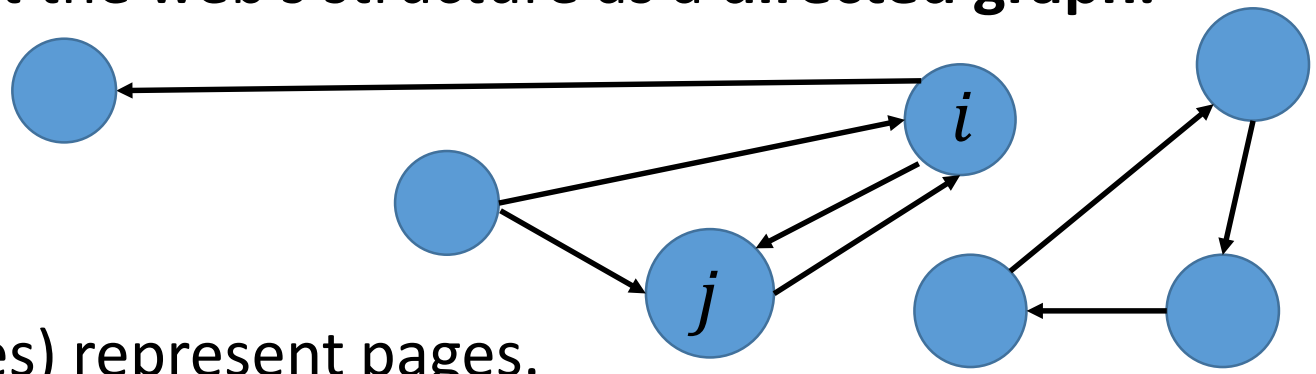
Key Takeaway

Numerical linear algebra could earn you $35 billion by age 43.



Web Links as a Graph

We represent the web's structure as a **directed graph.**



Nodes (circles) represent pages.

Arcs (arrows) represent links from one page to another.

We will use **degree** to refer to a node's *outdegree*, the number of arcs *leaving* that node.

e.g. $\deg(j) = 1, \deg(i) = 2.$

Adjacency Matrix

To store our directed graph, we might use some form of adjacency matrix, G.

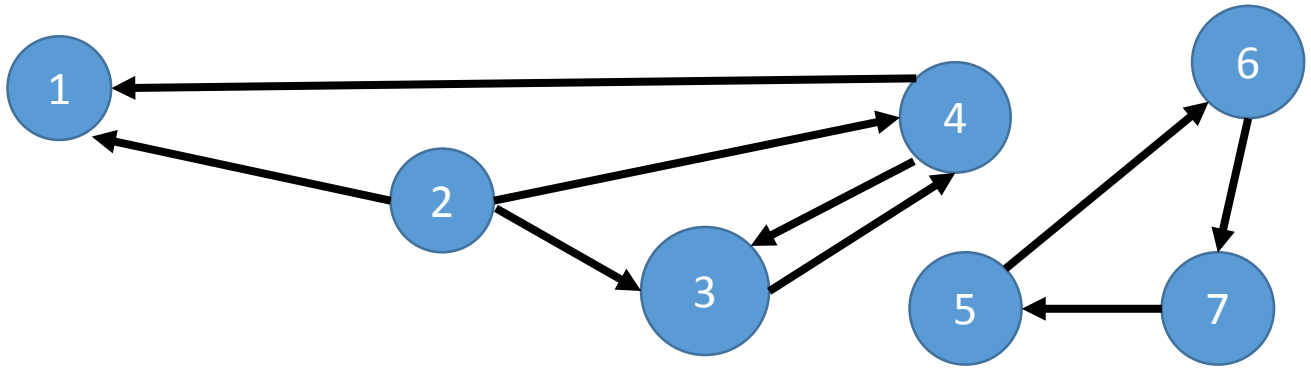
$$G_{ij} = \begin{cases} 1, & \text{if link } j \rightarrow i \text{ exists} \\ 0, & \text{otherwise} \end{cases}$$

Then the (out)degree for node q is the sum of entries in column q.

Notice: Matrix G is not necessarily symmetric about the diagonal!

What would this imply?

Example Adjacency Matrix



From

To	1	2	3	4	5	6	7
1	0	1	0	1	0	0	0
2	0	0	0	0	0	0	0
3	0	1	0	1	0	0	0
4	0	1	1	0	0	0	0
5	0	0	0	0	0	0	1
6	0	0	0	0	1	0	0
7	0	0	0	0	0	1	0

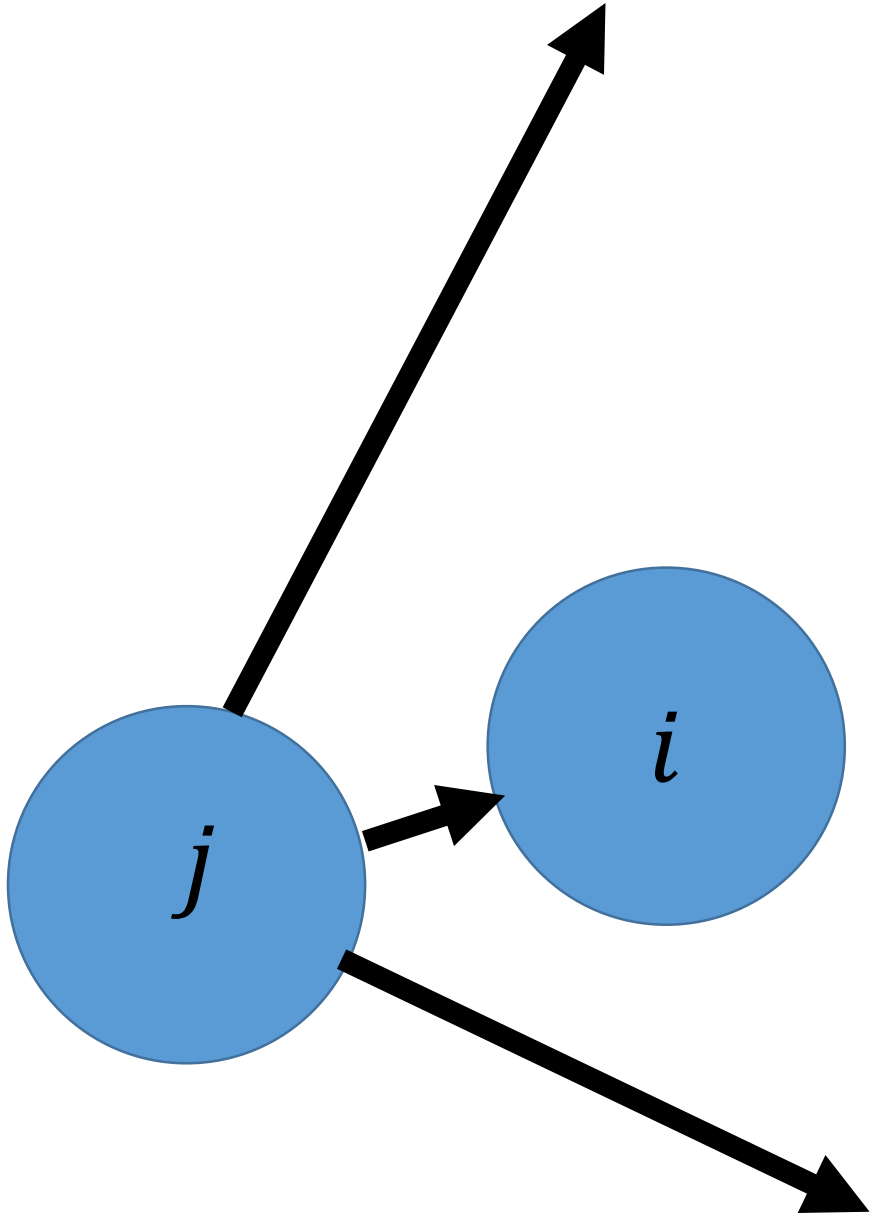
Interpreting Links as Votes

If page j links **to** page i, this is considered a "vote" by j that i is "important".

Outgoing links of a page j should have the same influence, so the importance that j "gives" to i is:

$$\frac{1}{\deg(j)}$$

In this case, j gives a 1/3 vote to i.



Global Importance

So: If page i has many incoming links, it is probably important.

What if page i has just one incoming link, but the link is from page j and ***j* is known to be important**?

Then i is probably fairly important too!

There is a recursive, chicken-and-egg thing happening...

The Random Surfer Model

Imagine an internet user who starts at a page, and **follows links at random** from page to page for K steps.

They will “probably” end up on important pages more often!

Then, select a new start page, and follow K random links again. Repeat the process R times.

At the end, we estimate overall importance as:

$$Rank(page\ i) = (Visits\ to\ page\ i)/(Total\ visits\ to\ all\ pages)$$

Random Surfer Algorithm

$Rank(m) = 0\ ,\ m = 1, ..., R$

For $m = 1, ..., R$

    $j = m$

    For $k = 1, ..., K$

        $Rank(j) = Rank(j) + 1$

        Randomly select outlink l of page j

        $j = l$

    EndFor

EndFor

$Rank(m) = Rank(m)/(K * R)\ ,\ m = 1, ..., R$

Random Surfer Criticisms

Potential issues with this algorithm?

- The number of real web pages is monstrously huge : 1 billion-ish unique hostnames; **many** iterations (large K,R) needed.
- Number of steps taken per random surf sequence must be large, to get a representative sample.
- What about dead end links? (Stuck on **one** page!)
- What about cycles in the graph? (Stuck on a **closed subset** of pages!)

Clearly, better strategies are needed.

A *Markov Chain Matrix*

Let's think in terms of probabilities.

Let P be a (large!) matrix of probabilities, where P_{ij} is the probability of randomly transitioning from page j to page i.

$$P_{ij} = \begin{cases} \dfrac{1}{\deg(j)}, & \text{if link } j \rightarrow i \text{ exists} \\ 0, & \text{otherwise} \end{cases}$$

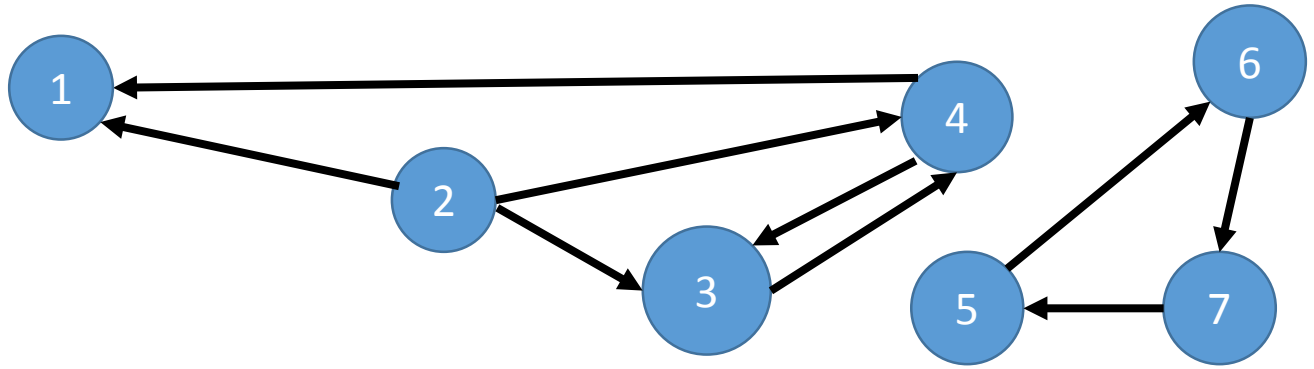
Markov Chain Matrix

We can build this matrix P from our adjacency matrix G.

Divide all entries of each column of G by the column sum (out-degree of the node).

From

To	1	2	3	4	5	6	7
1	0	1/3	0	1/2	0	0	0
2	0	0	0	0	0	0	0
3	0	1/3	0	1/2	0	0	0
4	0	1/3	1	0	0	0	0
5	0	0	0	0	0	0	1
6	0	0	0	0	1	0	0
7	0	0	0	0	0	1	0



Dead Ends

To deal with dead-end links, we will simply “teleport” to a new page at random!

Mathematically, we define a column vector d such that:

$$d_i = \begin{cases} 1, & \text{if } deg(i) = 0 \\ 0, & \text{otherwise} \end{cases}$$

and vector $e = [1,1,1 \dots 1,1]^T$ be a column vector of ones.

Then if R is the number of pages, we augment P to get P' defined by:

$$P' = P + \frac{1}{R} e d^T$$

Can you see why?

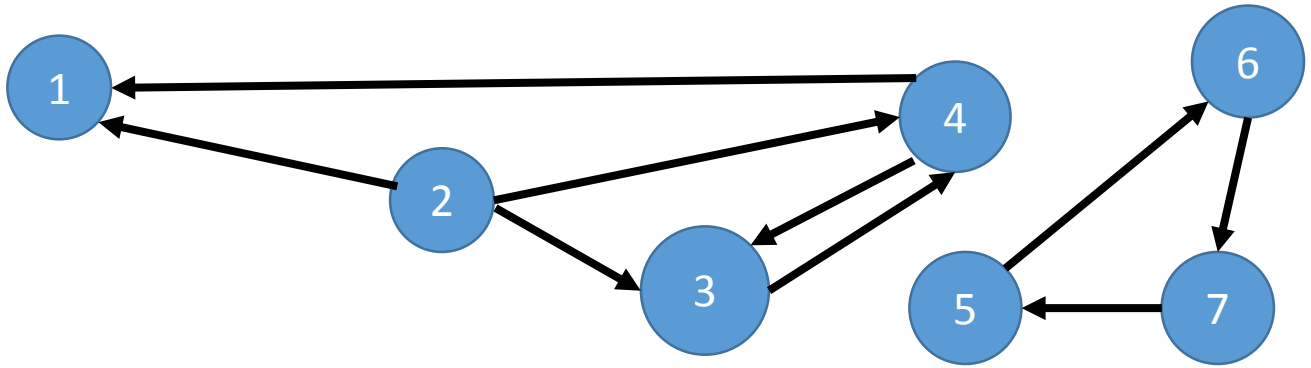
Dead Ends

The **matrix** $\frac{1}{R}ed^T$ is a matrix of probabilities such that **from** any dead end page ($d_i = 1$), we transition **to** every other page with equal probability.

$$d = [1,0,0,0,0,0,0]^T$$
$$e = [1,1,1,1,1,1,1]^T$$
$$R = 7$$

From

To	1	2	3	4	5	6	7
1	1/7	0	0	0	0	0	0
2	1/7	0	0	0	0	0	0
3	1/7	0	0	0	0	0	0
4	1/7	0	0	0	0	0	0
5	1/7	0	0	0	0	0	0
6	1/7	0	0	0	0	0	0
7	1/7	0	0	0	0	0	0



Escaping Cycles

How can we apply a similar trick to escape closed cycles of pages?

Most of the time (a fraction α), follow links randomly, via P'.

Occasionally, with some (usually small) probability, $(1-\alpha)$, teleport from **any** page to **any** other page.

$$M = \underbrace{\alpha P'}_{\text{Surf Randomly (only teleporting out of dead ends)}} + \underbrace{(1-\alpha)\frac{1}{R}ee^T}_{\text{Teleport Randomly (from any page).}}$$

Escaping Cycles

The $\frac{1}{R}ee^T$ matrix looks like:

$$\begin{bmatrix} \frac{1}{R} & \frac{1}{R} & \cdots & \cdots & \frac{1}{R} \\ \frac{1}{R} & \frac{1}{R} & \cdots & \cdots & \frac{1}{R} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{1}{R} & \frac{1}{R} & \cdots & \cdots & \frac{1}{R} \end{bmatrix}$$

Teleport randomly from one page to another with equal probability, regardless of links.

Google Matrix

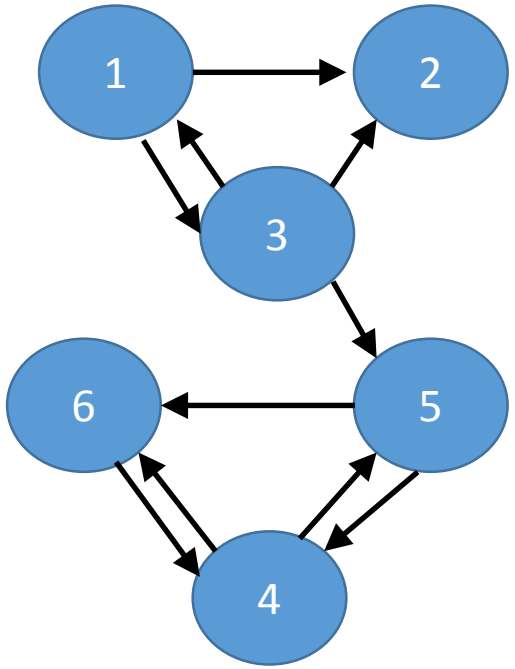
We will call the combined matrix $M = \alpha P' + (1 - \alpha)\frac{1}{R}ee^T$ our "google matrix".

Most of the time this just follows links (and *always* teleports out of dead ends), but also *occasionally* teleports randomly to escape cycles.

Google purportedly uses $\alpha \approx 0.85$.

M expresses one step of random surfing. Consider how we might **use** this to estimate page ranks…

Page Rank example (Notes Ex. 7.3)



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

$$d = [0, 1, 0, 0, 0, 0]^T.$$

(Page 2 is a dead end!)

Random Surfing
(but gets stuck in dead ends)

Page Rank example

$$P + \frac{1}{6}ed^T = \begin{bmatrix} 0 & \frac{1}{6} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{6} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{6} & 0 & 0 & \frac{1}{2} & 1 \\ 0 & \frac{1}{6} & \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{6} & 0 & \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix}$$

Add Teleportation
out of Dead Ends
(fills in empty cols)

$$M = \begin{bmatrix} \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} & \frac{1}{6}\alpha + \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha \\ \frac{1}{3}\alpha + \frac{1}{6} & \frac{1}{6} & \frac{1}{6}\alpha + \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha \\ \frac{1}{3}\alpha + \frac{1}{6} & \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha \\ \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{3}\alpha + \frac{1}{6} & \frac{5}{6}\alpha + \frac{1}{6} \\ \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} & \frac{1}{6}\alpha + \frac{1}{6} & \frac{1}{3}\alpha + \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} - \frac{1}{6}\alpha \\ \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha & \frac{1}{3}\alpha + \frac{1}{6} & \frac{1}{3}\alpha + \frac{1}{6} & \frac{1}{6} - \frac{1}{6}\alpha \end{bmatrix}$$

Add Occasional Random
Teleportation to Also Escape Cycles

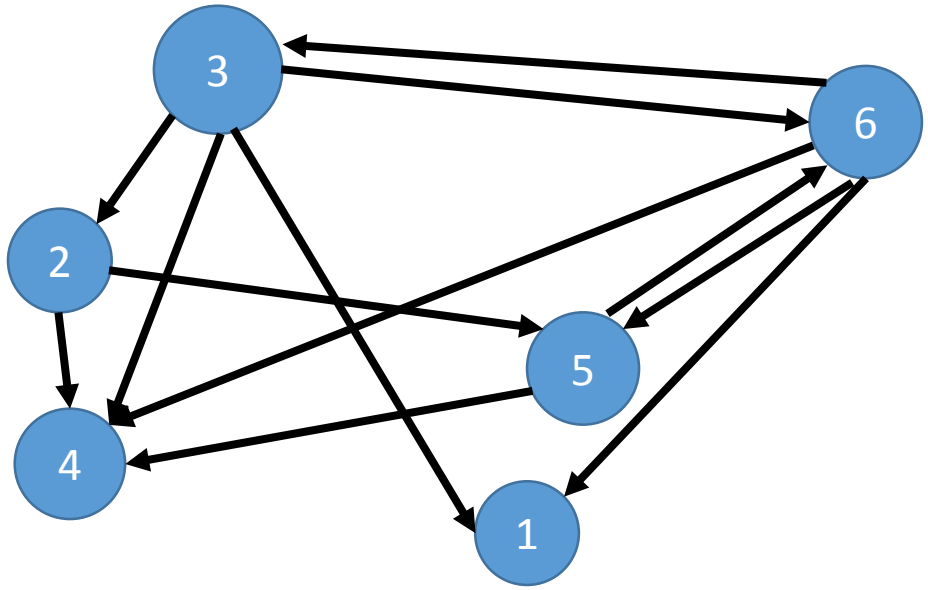
Page Rank example – Final Google matrix

For $\alpha = 0.85$, we have:

$$M = \begin{bmatrix} \frac{1}{40} & \frac{1}{6} & \frac{37}{120} & \frac{1}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{9}{20} & \frac{1}{6} & \frac{37}{120} & \frac{1}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{9}{20} & \frac{1}{6} & \frac{1}{40} & \frac{1}{40} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{1}{6} & \frac{1}{40} & \frac{1}{40} & \frac{9}{20} & \frac{7}{8} \\ \frac{1}{40} & \frac{1}{6} & \frac{37}{120} & \frac{9}{20} & \frac{1}{40} & \frac{1}{40} \\ \frac{1}{40} & \frac{1}{6} & \frac{1}{40} & \frac{9}{20} & \frac{9}{20} & \frac{1}{40} \end{bmatrix}.$$

Example Problem:

Construct the google matrix $M = \alpha\left(P + \frac{1}{R}ed^T\right) + (1-\alpha)\frac{1}{R}ee^T$ for the small web shown here, using $\alpha = 1/2$, and $R = 6$ pages.



Recall:

$$P_{ij} = \begin{cases} \frac{1}{\deg(j)}, & \text{if link } j \rightarrow i \text{ exists} \\ 0, & \text{otherwise} \end{cases}$$

$$d_i = \begin{cases} 1, & \text{if } deg(i) = 0 \\ 0, & \text{otherwise} \end{cases}$$

$$e = [1,1,1,\ldots,1]^T$$