L27. The PageRank Computation

Google PageRank

Background

Index all the pages on the Web from 1 to N. (N is around ten billion.)

The PageRank algorithm orders these pages from "most important" to "least important".

It does this by analyzing links, not content.

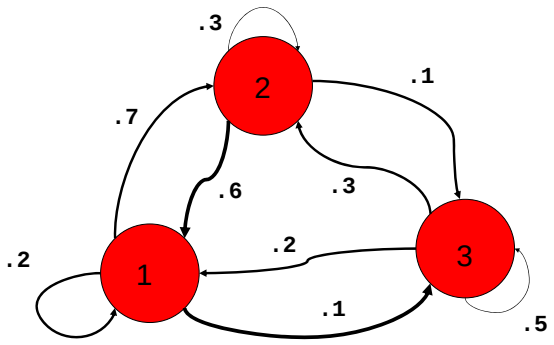
Key Ideas

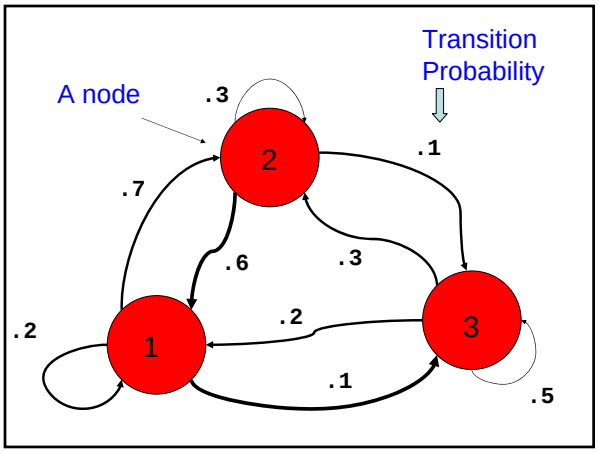
The Transition Probability Array

A Very Special Random Walk

The Connectivity Array

A Network



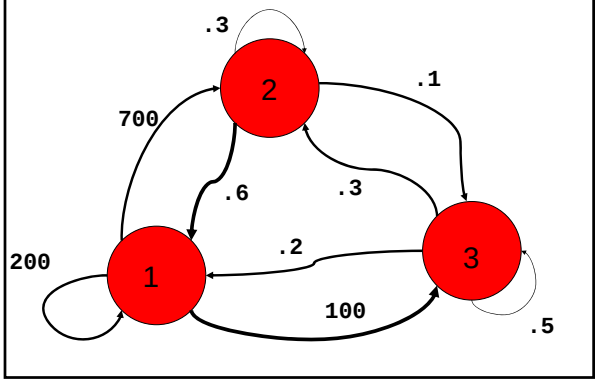


A Random Process

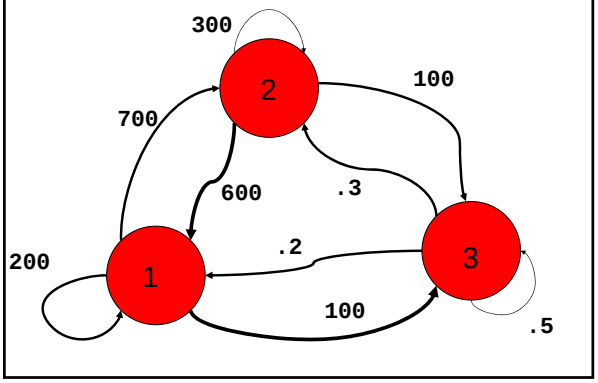
Suppose there are a 1000 people on each node.

At the sound of a whistle they hop to another node in accordance with the "outbound" probabilities.

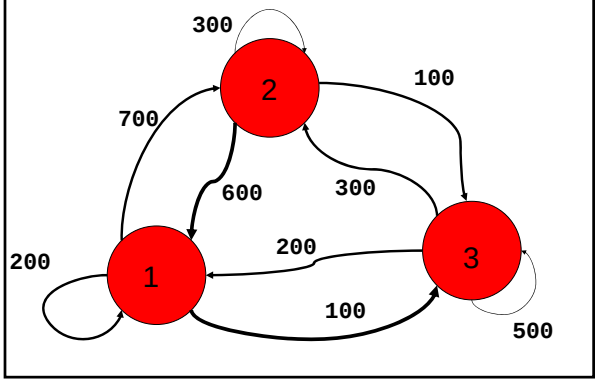
At Node 1



At Node 2



At Node 3



New Distribution

	Before	After
Node 1	1000	1000
Node 2	1000	1300
Node 3	1000	700

Repeat

	Before	After
Node 1	1000	1120
Node 2	1300	1300
Node 3	700	580

State Vectors

[1000 1000 1000]

→ [1000 1300 700]

→ [1120 1300 580]

After 100 Iterations

	Before	After
Node 1	1142.85	1142.85
Node 2	1357.14	1357.14
Node 3	500.00	500.00

Appears to reach a Steady State

The Stationary Vector

[1142.85 1357.14 500]

Transition Probability Array

P:

.2	.6	.2
.7	.3	.3
.1	.1	.5

P(i,j) is the probability of hopping form node j to node i

Formula for the New State Vector

.2	.6	.2
.7	.3	.3
.1	.1	.5

```
W(1) = .2*v(1) + .6*v(2) + .2*v(3)
W(2) = .7*v(1) + .3*v(2) + .3*v(3)
W(3) = .1*v(1) + .1*v(2) + .5*v(3)
```

Formula for the New State Vector

.2	.6	.2
.7	.3	.3
.1	.1	.5

```
W(1) = P(1,1)*v(1) + P(1,2)*v(2) + P(1.3)*v(3)
W(2) = P(2,1)*v(1) + P(2,2)*v(2) + P(2,3)*v(3)
W(3) = P(3,1)*v(1) + P(3,2)*v(2) + P(3,3)*v(3)
```

The General Case

```
function w = Update(P,v)
n = length(v);
w = zeros(n,1);
for i=1:n
   for j=1:n
      w(i) = w(i) + P(i,j)*v(j);
   end
end
```

The Stationary Vector

```
function [w,err,its] = Stat(P,v,tol,kMax)
k = 0;
while k==0 ||(k<kMax && err>tol)
      w = v;
      v = Update(P,w);
      k = k+1;
      err = max(abs(w-v));
end
w = v;
its = k;
```

A Random Walk on the Web

Repeat:

- You are on a webpage.
- There are m outlinks.
- Choose one at random.
- Click on the link.

What if no outlinks? Dead End

A Connectivity Array

G(i,j) is 1 if there is a link on page j to page i

G:

```
0 1 0 0 1 0 1 0
1 0 0 0 0 0 1 1
0 1 0 0 1 0 0 0
1 0 1 1 0 1 0 0
0 0 0 1 0 0 1 0
0 1 1 0 0 1 0 0
1 0 0 0 0 0 1 0
0 0 1 0 0 1 0 0
```

The Transition Array

a = 1/3

b = 1/2

c = 1/4

G:

```
0 a 0 0 b 0 c 0
a 0 0 0 0 0 c 1
0 a 0 0 b 0 0 0
a 0 a b 0 a 0 0
0 0 0 b 0 0 c 0
0 a a 0 0 a 0 0
a 0 0 0 0 0 c 0
0 0 a 0 0 a 0 0
```

Connectivity→ Transition

```
[n,n] = size(G);
P = zeros(n,n);
for j=1:n
   P(:,j) = G(:,j)/sum(G(:,j));
end
```

Connectivity

```
0   0   0   0   0   0   1   1
1   0   0   1   0   0   0   0
1   0   1   0   0   1   0   1
0   0   0   0   1   0   0   0
1   0   1   0   0   0   0   1
0   0   1   0   0   0   0   1
0   0   1   0   0   0   0   0
0   1   0   1   0   0   0   0
```

Transition

0	0	0	0	0	0	1	.25
.33	0	0	.50	0	0	0	0
.33	0	.25	0	0	1	0	.25
0	0	0	0	1	0	0	0
.33	0	.25	0	0	0	0	.25
0	0	.25	0	0	0	0	.25
0	0	.25	0	0	0	0	0
0	1	0	.50	0	0	0	0

Stationary Vector→ PageRank

statVec	sorted	idx	pR
0.5723	0.8911	6	4
0.8206	0.8206	2	2
0.7876	0.7876	3	3
0.2609	0.5723	1	6
0.2064	0.4100	8	8
0.8911	0.2609	4	1
0.2429	0.2429	7	7
0.4100	0.2064	5	5

```
for k=1:8
   j = idx(k) % index of kth largest
   pR(j) = k
end
```

statVec	sorted	idx	pR
0.5723	0.8911	6	4
0.8206	0.8206	2	2
0.7876	0.7876	3	3
0.2609	0.5723	1	6
0.2064	0.4100	8	8
0.8911	0.2609	4	1
0.2429	0.2429	7	7
0.4100	0.2064	5	5

PageRank vs InLinkRank

Page	inLinks	outLinks	PageRank	InLinkRank
1	2	4	8	5
2	2	1	2	6
3	2	1	1	7
4	5	4	3	1
5	3	3	5	2
6	2	2	6	8
7	3	4	7	3
8	3	3	4	4

A Random Walk on the Web

Repeat:

You are on a webpage.
There are m outlinks.
Choose one at random.
Click on the link.

What if no outlinks?

A New Random Walk on the Web

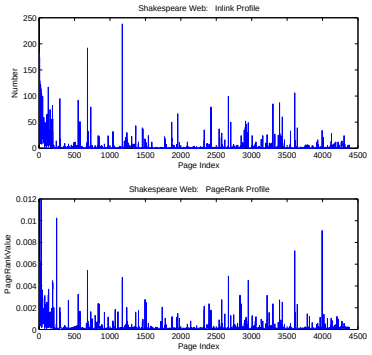
Repeat:
You are on a webpage.
If there are no outlinks
Pick a random page and go there
else
Flip an unfair coin
if heads
Click on a random outlink and go there
else
Pick a random page and go there
end
end

The Unfair Coin

It comes up heads with probability p = .85.

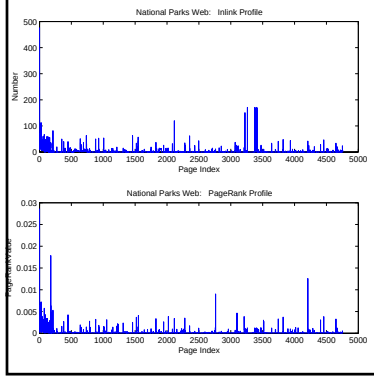
This value "works best."

Shakespeare SubWeb (n=4383)



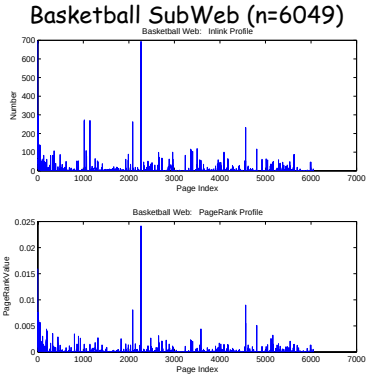
PRank	InRank
1	24
2	417
3	110
4	14
5	68
6	8
7	37
8	54
9	2
10	261
11	1
12	67
13	118
14	50
15	3

Nat'l Parks SubWeb (n=4757)



PRank	InRank
1	1
2	100
3	77
4	386
5	62
6	110
7	37
8	109
9	127
10	32
11	28
12	830
13	169
14	168
15	64

Basketball SubWeb (n=6049)



PRank	InRank
1	2
2	1
3	20
4	19
5	3
6	61
7	23
8	43
9	91
10	28
11	85
12	358
13	313
14	71
15	68