Das Pagerank-Verfahren

29. November 2010

Gegeben: Eine Sammlung von N Web-Seiten, die (teilweise) untereinander verlinkt sind.

Sei $L_{ij} = \begin{cases} 1 & \text{wenn Seite } i \text{ auf Seite } j \text{ verweist} \\ 0 & \text{sonst} \end{cases}$

Gegeben: Eine Sammlung von N Web-Seiten, die (teilweise) untereinander verlinkt sind.

Sei $L_{ij} = \begin{cases} 1 & \text{wenn Seite } i \text{ auf Seite } j \text{ verweist} \\ 0 & \text{sonst} \end{cases}$

Wir können die Sammlung als einen gerichteten Graphen $\mathcal{G}$ auffassen, dessen Knoten die N Seiten sind, und die als Links gerichtete Kanten darstellen.

(Im Jargon der Graphentheorie ist (L_{ij}) die *Adjazenzmatrix*.)

Gegeben: Eine Sammlung von N Web-Seiten, die (teilweise) untereinander verlinkt sind.

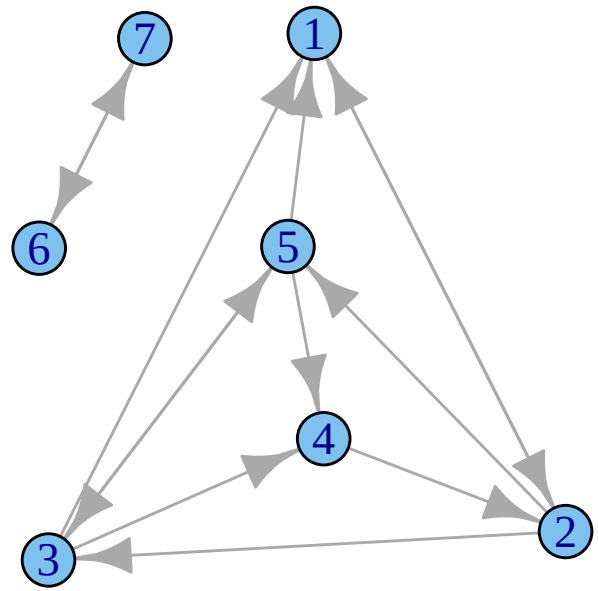
Sei $L_{ij} = \begin{cases} 1 & \text{wenn Seite } i \text{ auf Seite } j \text{ verweist} \\ 0 & \text{sonst} \end{cases}$

Wir können die Sammlung als einen gerichteten Graphen $\mathcal{G}$ auffassen, dessen Knoten die N Seiten sind, und die als Links gerichtete Kanten darstellen.

(Im Jargon der Graphentheorie ist (L_{ij}) die *Adjazenzmatrix*.)

Beispiel:

$$L = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Aufgabe: Die Web-Seiten nach „Wichtigkeit“ sortieren.

Idee: Eine Seite ist umso „wichtiger“, je mehr „wichtige“ Seiten auf sie verweisen.

Klingt erstmal recht selbstbezüglich?!

Aufgabe: Die Web-Seiten nach „Wichtigkeit“ sortieren.

Idee: Eine Seite ist umso „wichtiger“, je mehr „wichtige“ Seiten auf sie verweisen.

Klingt erstmal recht selbstbezüglich?!

Eine Art, „Wichtigkeit“ zu messen, liefert das Modell des *Zufalls-Surfers*.

Zufalls-Surfer

$\alpha \in (0, 1)$:

- Mit Wahrscheinlichkeit α: Folge einem rein zufällig ausgewählten Link auf der gerade betrachteten Seite (bzw. springe zu einer rein zufällig aus allen *N* Web-Seiten gewählten, wenn es auf der aktuellen Seite überhaupt keinen Link gibt).
- Mit Wahrscheinlichkeit $1 - \alpha$: Springe zu einer rein zufällig aus allen *N* gewählten Web-Seite. (Mögliche Interpretation: Dem Surfer wird beim Link-Verfolgen langweilig und er möchte etwas ganz anderes sehen.)

Zufallssurfer als Markovkette

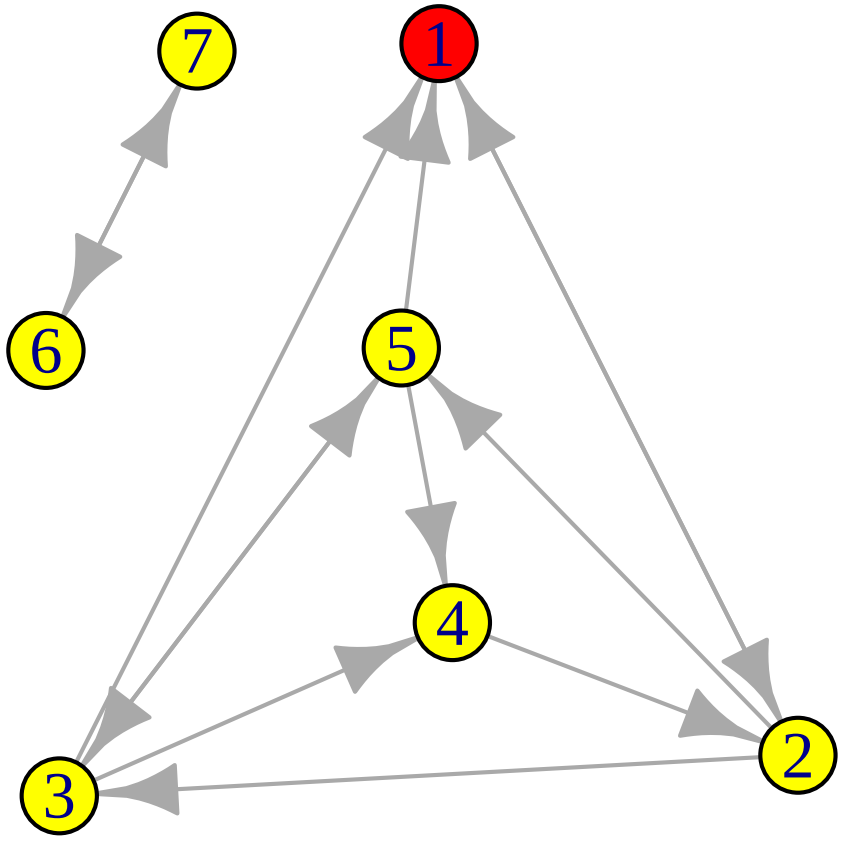
Übergangsmatrix:

$$A_{ij} = \alpha\Big(\frac{L_{ij}}{C_i}\mathbf{1}(C_i > 0) + \frac{1}{N}\mathbf{1}(C_i = 0)\Big) + (1-\alpha)\frac{1}{N}$$

wobei $C_i = \sum_{j=1}^{N} L_{ij}$ die Anzahl Links auf Seite i ist.
Beobachtung: (A_{ij}) ist irreduzibel und aperiodisch (dafür sorgt der Term $(1-\alpha)\frac{1}{N}$, denn damit sind alle Einträge strikt positiv).

$\alpha \in (0,1)$ ist ein „Tuning-Parameter“.

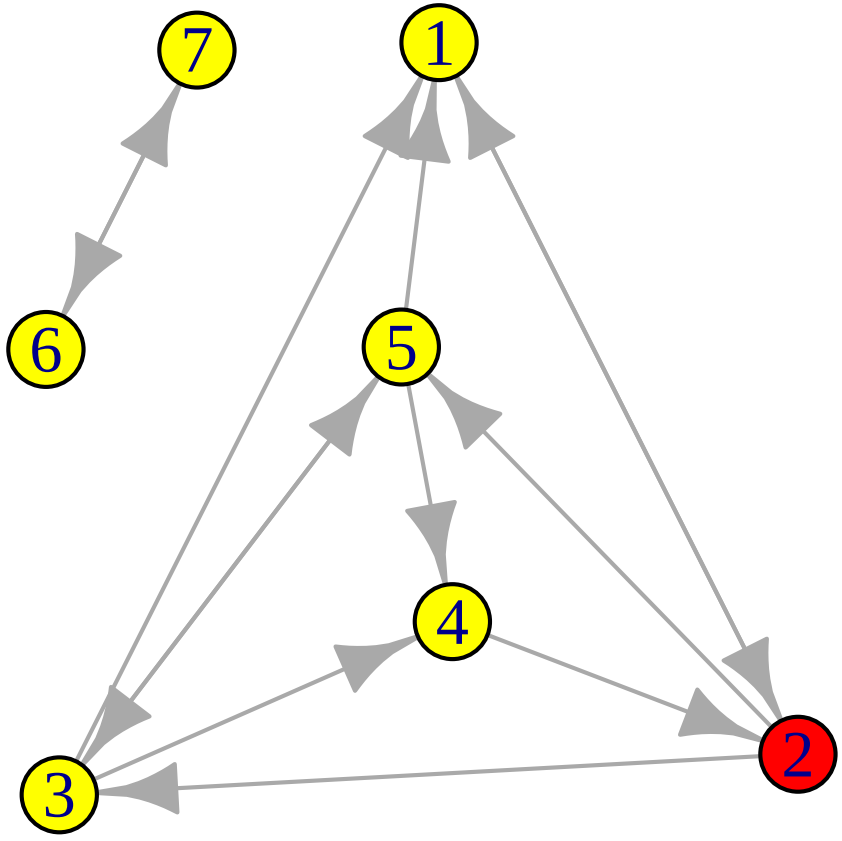
Beispiel: Ein Pfad durch die Web-Seiten



Starte auf Web-Seite 1

Zeit i	0	1	2	3	4	5
Ort X_i	1					
W'keit d. Überg.	–					

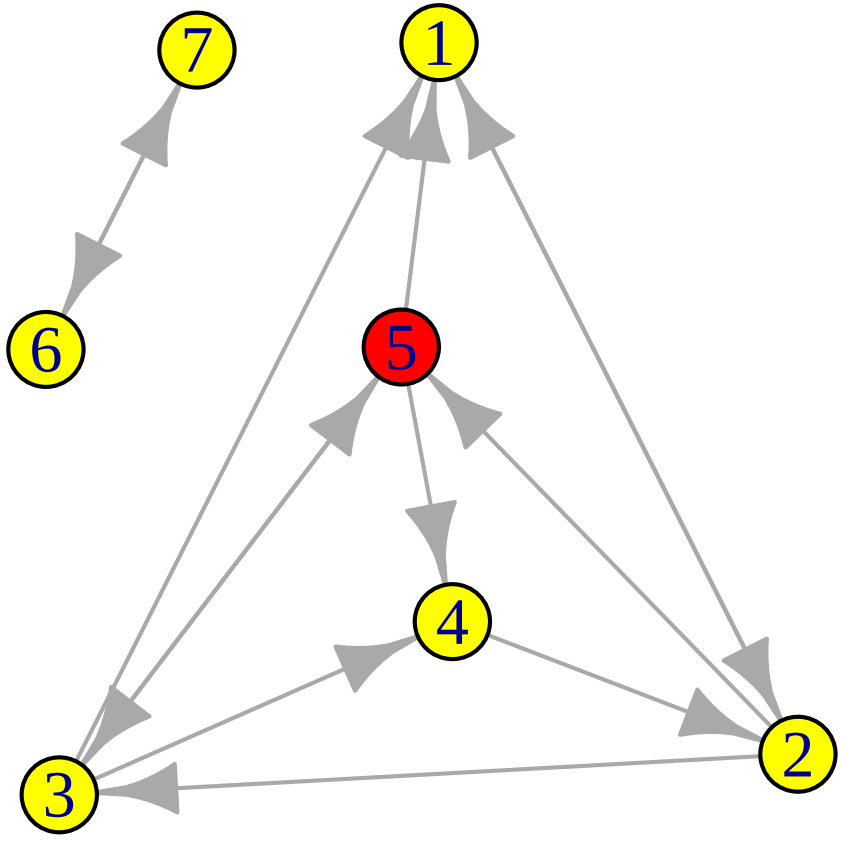
Beispiel: Ein Pfad durch die Web-Seiten



Starte auf Web-Seite 1

Zeit i	0	1	2	3	4	5
Ort X_i	1	2				
W'keit d. Überg.	–	$\frac{\alpha}{1} + \frac{1-\alpha}{7}$				

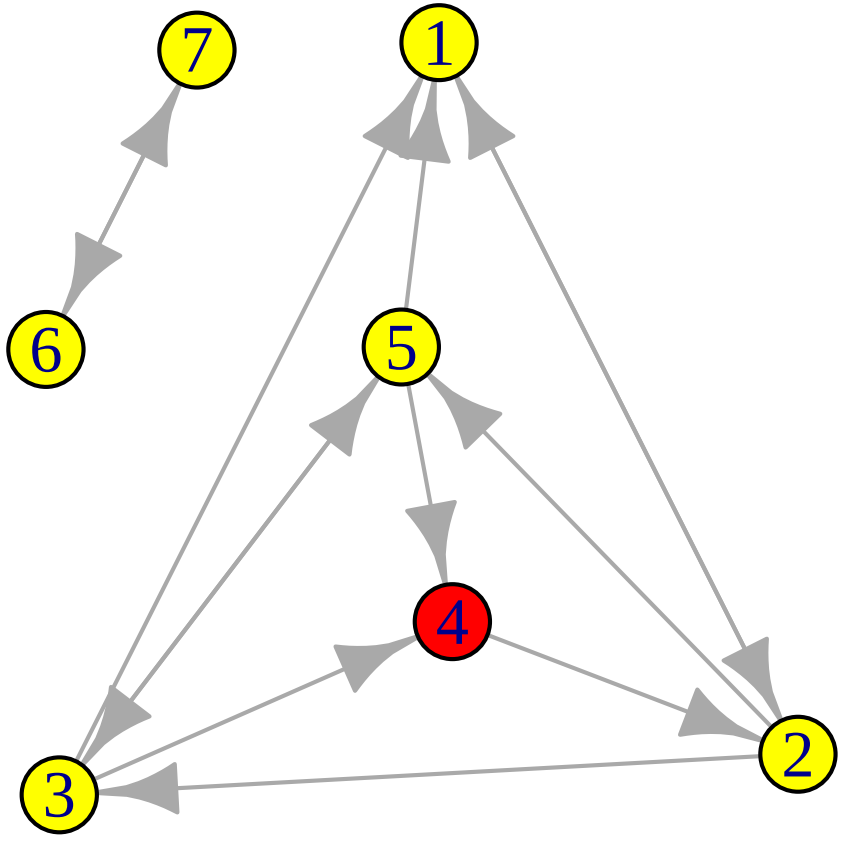
Beispiel: Ein Pfad durch die Web-Seiten



Starte auf Web-Seite 1

Zeit i	0	1	2	3	4	5
Ort X_i	1	2	5			
W'keit d. Überg.	–	$\frac{\alpha}{1} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$			

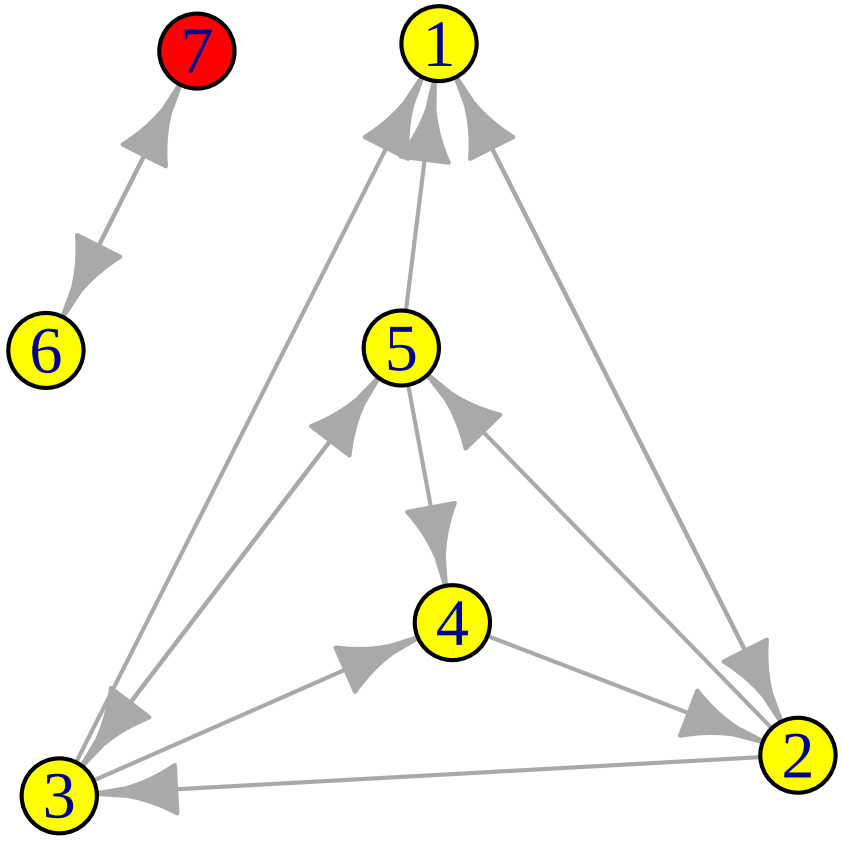
Beispiel: Ein Pfad durch die Web-Seiten



Starte auf Web-Seite 1

Zeit i	0	1	2	3	4	5
Ort X_i	1	2	5	4		
W'keit d. Überg.	–	$\frac{\alpha}{1} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$		

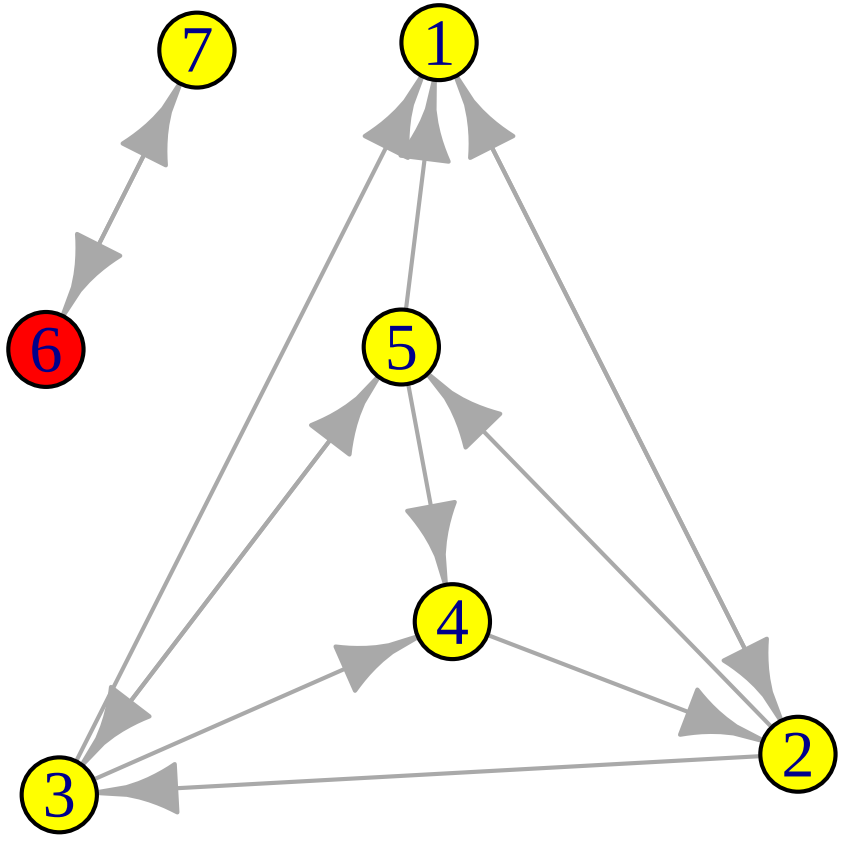
Beispiel: Ein Pfad durch die Web-Seiten



Starte auf Web-Seite 1

Zeit i	0	1	2	3	4	5
Ort X_i	1	2	5	4	7	
W'keit d. Überg.	–	$\frac{\alpha}{1} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$	$0 + \frac{1-\alpha}{7}$	

Beispiel: Ein Pfad durch die Web-Seiten



Starte auf Web-Seite 1

Zeit i	0	1	2	3	4	5
Ort X_i	1	2	5	4	7	6
W'keit d. Überg.	–	$\frac{\alpha}{1} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$	$\frac{\alpha}{3} + \frac{1-\alpha}{7}$	$0 + \frac{1-\alpha}{7}$	$\frac{\alpha}{1} + \frac{1-\alpha}{7}$

Vorschlag: Wir messen für jede Seite *i* den relativen Anteil π_i der Zeit, den der Zufalls-Surfer auf Seite *i* verbringt

und postulieren:

Wichtigkeit ist proportional zu diesem Anteil.

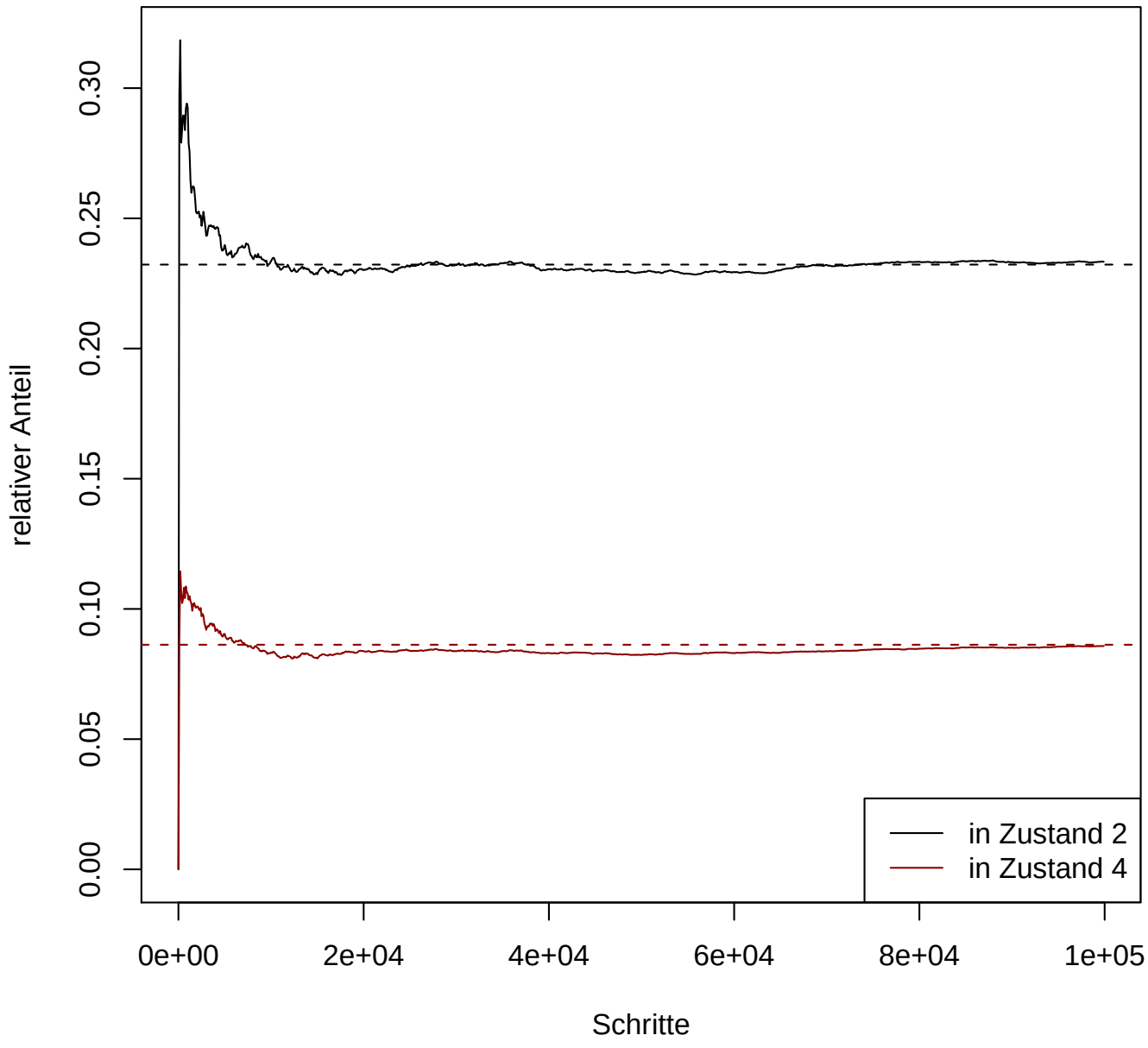
Die Übergangsmatrix des Beispiels in R:

```
# Adjazenzmatrix
L <- matrix(c(0,1,1,0,1,0,0,
              1,0,0,1,0,0,0,
              0,1,0,0,1,0,0,
              0,0,1,0,1,0,0,
              0,1,1,0,0,0,0,
              0,0,0,0,0,0,1,
              0,0,0,0,0,1,0), ncol=7)
N <- 7
alpha <- 0.9

# Bau der Uebergangsmatrix "von Hand":
A <- matrix(0, nrow=N, ncol=N)
for (i in 1:N) {
  c <- sum(L[i,])
  for (j in 1:N) {
    A[i,j] <- alpha*ifelse(c>0, L[i,j]/c, 1/N)+(1-alpha)/N
  }
}

# (Alternativ als Zweizeiler:)
C <- matrix(rep(rowSums(L),each=N), nrow=N, byrow=TRUE)
Aalt <- alpha*ifelse(C>0, L/C, 1/N)+(1-alpha)*matrix(1/N,nrow=N,ncol=N)
```

Simulation der Kette in R:

```
# Simulieren eines Uebergangsschritts gemaess Uebergangsmatrix A von Startpunkt x aus:
sim.MKschritt <- function(A, x)
  sample(1:ncol(A), 1, prob=A[x,])

# Einen Pfad der Markovkette mit Uebergangsmatrix A fuer eine gewisse Anzahl Schrite sim
schritte <- 100000

pfad <- integer(schritte)

# Startpunkt
pfad[1] <- sample(1:N)

for (i in 2:schritte)
  pfad[i] <- sim.MKschritt(A, pfad[i-1])

# relative Zeitanteile in verschiedenen Zustaenden verfolgen:
sapply(1:N, function(i) sum(pfad==i))/schritte
anteil2 <- cumsum(pfad==2)/(1:schritte)
anteil4 <- cumsum(pfad==4)/(1:schritte)

# und zeichnen:
zeitgitter <- seq(from=1, to=schritte, by=100)

plot(zeitgitter, anteil2[zeitgitter], type="l", xlab="Schritte", col="black",
     ylab="relativer Anteil")
points(zeitgitter, anteil4[zeitgitter], type="l", col="darkred")
legend("bottomright",legend=c("in Zustand 2", "in Zustand 4"), lty=1,
                        col=c("black","darkred"))
```

Relativer Anteil der Zeit in den Zuständen 2 und 4 für einen simulierten Pfad der Markovkette



Tatsächlich gilt (ganz allgemein für irreduzible und aperiodische Markovketten auf endlichen Mengen):
Sei X_n = Ort der Kette im n-ten Schritt. Für jedes i gilt

$$\frac{1}{T}\sum_{n=1}^{T}\mathbf{1}(X_n = i) \to \pi_i$$

(mit Wahrscheinlichkeit 1), wo $\pi = (\pi_i)_{i=1,\ldots,N}$ das (eindeutig bestimmte) Gleichgewicht zu A ist, d.h.

$$\pi = \pi A \tag{1}$$

(so normiert, dass $\sum_{i=1}^{N} \pi_i = 1$).

(1) ist ein lineares Gleichungssystem für $(\pi_1, \ldots, \pi_N)$ (das unterbestimmt ist, Lösung eindeutig unter Gesamtsummenbedingung).

Gesucht: Gleichgewicht, $\pi A = \pi$

Iterative (approximative) Lösung:

Sei μ irgendeine Verteilung auf $\{1, \ldots, N\}$ (die uniforme Vert. auf den N Web-Seiten, $\mu = (\frac{1}{N}, \ldots \frac{1}{N})$), ist vielleicht die naheliegendste Wahl), so gilt

$$\pi = \lim_{n \to \infty} \mu A^n \tag{2}$$

Gesucht: Gleichgewicht, $\pi A = \pi$

Iterative (approximative) Lösung:

Sei μ irgendeine Verteilung auf $\{1, \ldots, N\}$ (die uniforme Vert. auf den N Web-Seiten, $\mu = (\frac{1}{N}, \ldots \frac{1}{N})$), ist vielleicht die naheliegendste Wahl), so gilt

$$\pi = \lim_{n \to \infty} \mu A^n \tag{2}$$

Wahrscheinlichkeitsinterpretation: Die Markovkette (X_n) erfüllt für jedes i

$$\lim_{n \to \infty} \mathbb{P}_\mu(X_n = i) = \pi_i.$$

Gesucht: Gleichgewicht, $\pi A = \pi$

Iterative (approximative) Lösung:

Sei μ irgendeine Verteilung auf $\{1, \ldots, N\}$ (die uniforme Vert. auf den N Web-Seiten, $\mu = (\frac{1}{N}, \ldots \frac{1}{N})$), ist vielleicht die naheliegendste Wahl), so gilt

$$\pi = \lim_{n \to \infty} \mu A^n \tag{2}$$

Wahrscheinlichkeitsinterpretation: Die Markovkette (X_n) erfüllt für jedes i

$$\lim_{n \to \infty} \mathbb{P}_\mu(X_n = i) = \pi_i.$$

Die Konvergenz in (2) ist sehr schnell: Untersuchen wir den ℓ^1-Abstands zum Gleichgewicht, $\sum_{i=1}^{N} \left|(\mu A^n)_i - \pi_i\right|$ mit R

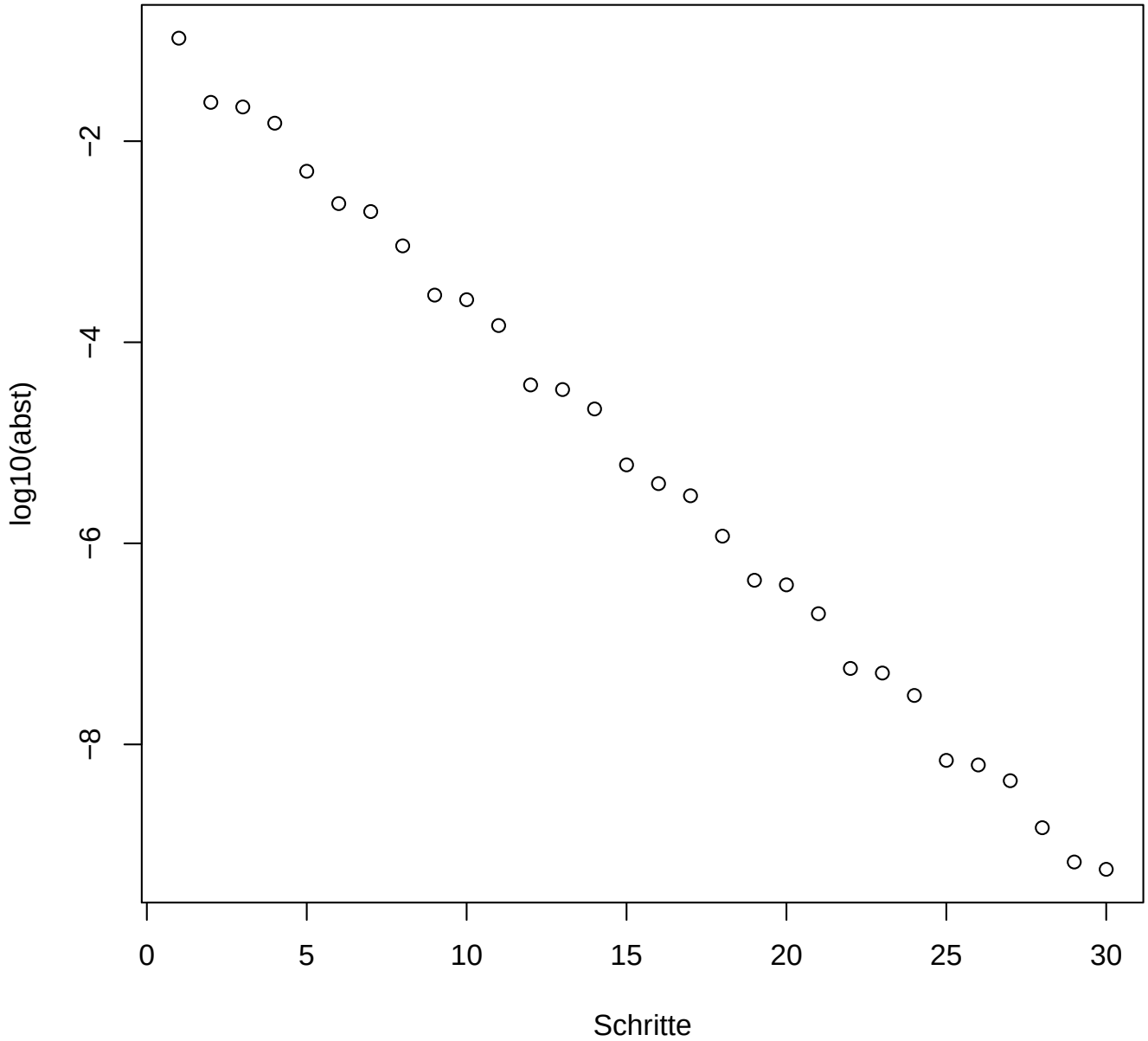
π via iteration mit R bestimmen:

```
# 2^n-te Potenz einer Matrix (schnell) berechnen:
matrix2erpotenz <- function(M, n) {
  B <- M
  for (i in 1:n)
    B <- B %*% B
  B
}

# Verteilung nach 2^15 Schritten ist numerisch vom
# Gleichgewicht ununterscheidbar:
pi <- rep(1/N, times=N) %*% matrix2erpotenz(A,15)
pi
pi %*% A
pi - pi %*% A
```

Abstand vom Gleichgewicht mit R bestimmen:

```
schritte <- 30
stvert <- rep(1/N, times=N)
abst <- numeric(schritte)
B <- A
for (i in 1:schritte) {
  abst[i] <- sum(abs(stvert %*% B - pi))
  B <- B %*% A
}
plot(1:schritte, log10(abst), xlab="Schritte")
```

(Zehner-)Logarithmus des ℓ^1-Abstands zum Gleichgewicht, $\sum_{i=1}^{N} \left|(\mu A^n)_i - \pi_i\right|$, als Funktion der Schrittanzahl n



Bemerkungen

- Die Firma Google benutzt diese Methode (unter anderen), um Suchergebnisse nach „Relevanz“ zu sortieren (das sog. PageRank-Verfahren, es ist patentiert und der Name als Markenzeichen geschützt).
- Diese Tatsache liegt auch den sog. Google-Bomben zugrunde, bei denen Webautoren absichtlich viele Links auf eine gewisse Seite setzen (zusammen mit einem geeigneten “anchor text”), um deren Rang bei Google-Suche zu erhöhen. (Im Jahr 2007 war es beispielsweise Gegnern von George W. Bush gelungen, seine offizielle Seite als Treffer Nummer 1 für den Suchbegriff “miserable failure” bei Google zu etablieren.)
- Google (und andere Internet-Suchmaschinen) löst ein nicht-triviales Pronlem der numerischen linearen Algebra: $N > 10^9$