

BHao_Assign10

Playing with PageRank

```
# form A and B
A = matrix(c( 0, 1/2, 1/2, 0, 0, 0,
              0, 0, 0, 0, 0, 0,
              1/3, 1/3, 0, 0, 1/3, 0,
              0, 0, 0, 0, 1/2, 1/2,
              0, 0, 0, 1/2, 0, 1/2,
              0, 0, 0, 1, 0, 0),
           6, 6, byrow = TRUE)

# start with uniform rank and perform 1000 iterations
r = rep(1/6, 6)

# replace disconnected node 2 with uniform ranking
A[2,] = r

# create B matrix after transposing A
B = 0.85 * t(A) + 0.15 / nrow(A)

r = B %>% 50 %>% r
r

##           [,1]
## [1,] 0.05170475
## [2,] 0.07367926
## [3,] 0.05741241
## [4,] 0.34870369
## [5,] 0.19990381
## [6,] 0.26859608

# compute eigen-decomposition of B
eig = eigen(B)
eig$values

## [1] 1.00000000+0i 0.57619235+0i -0.42500000+0i -0.42500000-0i
## [5] -0.34991524+0i -0.08461044+0i

eig$vectors[,1]

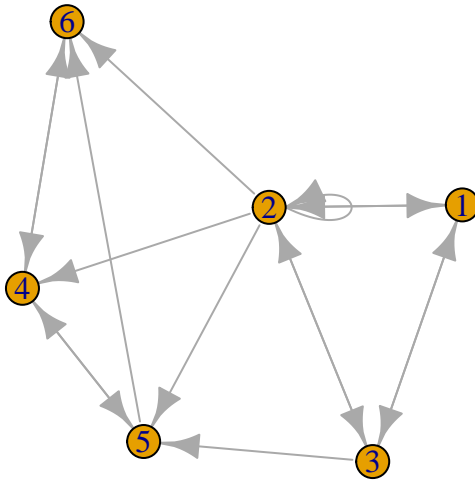
## [1] 0.1044385+0i 0.1488249+0i 0.1159674+0i 0.7043472+0i 0.4037861+0i
## [6] 0.5425377+0i

eig$vectors[,1] / sum(eig$vectors[,1]) # normalize

## [1] 0.05170475+0i 0.07367926+0i 0.05741241+0i 0.34870369+0i 0.19990381+0i
## [6] 0.26859608+0i

# use page.rank to compute page rank of A
# g2 = graph.formula(1 -> 2, 1 + 3,
#                    3 -> 2, 3 -> 5,
#                    4 + 5, 4 + 6,
#                    5 -> 6)
```

```
g = graph_from_adjacency_matrix(A, weighted = T, mode = 'directed')
plot(g)
```



```
page.rank(g, damping = 0.85)
```

```
## $vector
## [1] 0.05170475 0.07367926 0.05741241 0.34870369 0.19990381 0.26859608
##
## $value
## [1] 1
##
## $options
## NULL
```