TP2

Basic PageRank

Functions

These are the functions we created to compute the PageRank index.

Sum powers of a matrix

Computes the sum of n first powers of matrix m.

n.d is a damping factor applied to the powers of m.

Example : *sum.powers.matrix(m, 3, 2)* returns: **m + (1 / (2^2)) * m^2 + (1 / (3^2)) * m^3**

```
sum.powers.matrix <- function(m, n, n.d) {
    powers <- c(1:n)
    res <- Reduce('+', lapply(powers, function(x) (1 / (x ^ n.d)) * (m %^% x)))

    return(res)
}
```

Remove auto-references

Removes all the auto-references in matrix m (ie puts the diagonal to 0).

```
remove.autoreferences <- function(m) {
    res <- m
    diag(res) <- 0

    return(res)
}
```

Computes an iteration of the PageRank algorithm.

Parameters:

- refs: the references matrix

- n: the number of powers of refs to consider (ie the depth of references)

- d: the PageRank damping factor

- n.d: the damping factor for powers of refs (see function sum.powers.matrix)

- pr: the current PageRank values

pagerank.iteration <- function(refs, n, d, n.d, pr) { # Number of articles n.articles <- dim(refs)[1]

```
# n-level references
m <- sum.powers.matrix(refs, n, n.d)

# remove auto-references
m <- remove.autoreferences(m)

# Compute PageRank
pr.res <- (1-d)/n.articles + (d * (m %*% (pr/colSums(m))))

return(pr.res)

}
```

Read source file

data <- read.table("citeseer.rtable")

Cast data to matrix

references <- as.matrix(data)

m <- matrix(c(0,1,1,0,0,1,1,0,0),3) pr <- rep(1,3)

pr <- pagerank.iteration(m, 2, 0.85, 2, pr)

pr <- pagerank.iteration(m, 2, 0.85, 2, pr)

pr <- pagerank.iteration(m, 2, 0.85, 2, pr) print(pr)