Z-score, multi-labeled nodes

H Qin

10/29/2018

```
rm(list=ls())
debug = 1
library(igraph)
```

```
##
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

```
## The following object is masked from 'package:base':
##
##     union
```

```
pairs= read.csv("seats.csv", colClasses = c("character", "character"))
nodes = read.csv("students2.csv", colClasses = c("character", "character", "character"))
```

Order state,

```
#newpairs = data.frame( t(apply(pairs, 1, sort)) ); #oder id1 and id2
#cbind( newpairs, pairs)
#pairs = newpairs
names(pairs) = c("name1", "name2")
```

hobbies

```
pairs$hobbies1 = nodes$hobbies[match( pairs$name1, nodes$Names)]
pairs$hobbies2 = nodes$hobbies[match( pairs$name2, nodes$Names )]
```

function to find out all combinations of two vectors

```
allCombinationsOfTwoVectors = function (els1, els2 ) {
  tagbuffer = c();
  for (e1 in els1) {
    for (e2 in els2) {
       tmp = sort(c(e1, e2));
       current_tag = paste(tmp[1], tmp[2], sep="_")
       tagbuffer = c(tagbuffer, current_tag)
    }
  }
  return( tagbuffer)
```

```
}

allCombinationsOfTwoVectors( c("one", "two"), c("red", "blue", "orange"))
```

```
## [1] "one_red"    "blue_one"   "one_orange" "red_two"    "blue_two"
## [6] "orange_two"
```

```
pairsBuffer = data.frame(matrix(NA, nrow = 1, ncol=3))
names(pairsBuffer) = c("name1", "name2", "tag")
for ( i in 1:length(pairs[,1])){
  els1 = sort( unlist( strsplit(  pairs$hobbies1[i], split=",") ))
  els2 = sort( unlist( strsplit(  pairs$hobbies2[i], split=",") ))
  tagbuffer = allCombinationsOfTwoVectors ( els1, els2)  #all combinations

  # generate a dataframe buffer with ids
  currentBuffer = data.frame( cbind(rep(pairs$name1[i], length(tagbuffer)),
                        rep(pairs$name2[i], length(tagbuffer)),
                        tagbuffer                          ))
  names(currentBuffer) = c("name1", "name2", "tag")

  pairsBuffer = rbind( pairsBuffer, currentBuffer) #combine with dataframe buffer
}

F.obs = data.frame( table(pairsBuffer$tag))
names(F.obs) = c("tag", "freq")
F.obs
```

```
##            tag freq
## 1   bike_movie    2
## 2    bike_read    1
## 3     bike_run    3
## 4    game_read    2
## 5    game_walk    2
## 6  movie_movie    2
## 7   movie_read    1
## 8    movie_run    2
## 9    read_read    1
## 10    read_run    5
## 11   read_walk    1
## 12    run_walk    4
```

#load MS02 null networks

```
ms02files = list.files(path='MS02')
F.ms02 = data.frame(matrix(data=NA, nrow=1, ncol=3))
names(F.ms02) = c('tag', 'freq', 'file')

# file = "_ms02_seats.1.csv" #debug
for (file in ms02files ){
  if ( debug > 0 ) { print(file) }
  ms02_pairs= read.csv(paste("MS02/", file, sep=''), colClasses = c("character", "character"))
  ms02_pairs = ms02_pairs[,1:2]
  names( ms02_pairs) = c("name1", "name2")
  ms02_pairs$hobbies1 = nodes$hobbies[match( ms02_pairs$name1, nodes$Names)]
  ms02_pairs$hobbies2 = nodes$hobbies[match( ms02_pairs$name2, nodes$Names )]
```

```
  pairsBuffer = data.frame(matrix(NA, nrow = 1, ncol=3))
  names(pairsBuffer) = c("name1", "name2", "tag")
  for ( i in 1:length(ms02_pairs[,1])){
    els1 = sort( unlist( strsplit( ms02_pairs$hobbies1[i], split=",") ))
    els2 = sort( unlist( strsplit( ms02_pairs$hobbies2[i], split=",") ))
    tagbuffer = allCombinationsOfTwoVectors ( els1, els2)  #all combinations

    # generate a dataframe buffer with ids
    currentBuffer = data.frame( cbind(rep(ms02_pairs$name1[i], length(tagbuffer)),
                        rep(ms02_pairs$name2[i], length(tagbuffer)),
                        tagbuffer                          ))
    names(currentBuffer) = c("name1", "name2", "tag")

    pairsBuffer = rbind( pairsBuffer, currentBuffer) #combine with dataframe buffer

  }#ms02_pair loop

  F.ms02current = data.frame( table(pairsBuffer$tag))
  F.ms02current$file = file
  names(F.ms02current) = c('tag', 'freq', 'file')
  F.ms02 =  data.frame( rbind(F.ms02, data.frame(F.ms02current)) )

}#file loop
```

```
## [1] "_ms02_seats.1.csv"
## [1] "_ms02_seats.10.csv"
## [1] "_ms02_seats.11.csv"
## [1] "_ms02_seats.12.csv"
## [1] "_ms02_seats.13.csv"
## [1] "_ms02_seats.14.csv"
## [1] "_ms02_seats.15.csv"
## [1] "_ms02_seats.16.csv"
## [1] "_ms02_seats.17.csv"
## [1] "_ms02_seats.18.csv"
## [1] "_ms02_seats.19.csv"
## [1] "_ms02_seats.2.csv"
## [1] "_ms02_seats.20.csv"
## [1] "_ms02_seats.21.csv"
## [1] "_ms02_seats.22.csv"
## [1] "_ms02_seats.23.csv"
## [1] "_ms02_seats.24.csv"
## [1] "_ms02_seats.25.csv"
## [1] "_ms02_seats.26.csv"
## [1] "_ms02_seats.27.csv"
## [1] "_ms02_seats.28.csv"
## [1] "_ms02_seats.29.csv"
## [1] "_ms02_seats.3.csv"
## [1] "_ms02_seats.30.csv"
## [1] "_ms02_seats.31.csv"
## [1] "_ms02_seats.32.csv"
## [1] "_ms02_seats.33.csv"
## [1] "_ms02_seats.34.csv"
## [1] "_ms02_seats.35.csv"
```

```
## [1] "_ms02_seats.36.csv"
## [1] "_ms02_seats.37.csv"
## [1] "_ms02_seats.38.csv"
## [1] "_ms02_seats.39.csv"
## [1] "_ms02_seats.4.csv"
## [1] "_ms02_seats.40.csv"
## [1] "_ms02_seats.41.csv"
## [1] "_ms02_seats.42.csv"
## [1] "_ms02_seats.43.csv"
## [1] "_ms02_seats.44.csv"
## [1] "_ms02_seats.45.csv"
## [1] "_ms02_seats.46.csv"
## [1] "_ms02_seats.47.csv"
## [1] "_ms02_seats.48.csv"
## [1] "_ms02_seats.49.csv"
## [1] "_ms02_seats.5.csv"
## [1] "_ms02_seats.50.csv"
## [1] "_ms02_seats.6.csv"
## [1] "_ms02_seats.7.csv"
## [1] "_ms02_seats.8.csv"
## [1] "_ms02_seats.9.csv"
```

```
F.ms02 = F.ms02[ !is.na(F.ms02$tag), ]
```

Initialize the Z-score matrix

```
unique_tags = unique( c(as.character(F.obs$tag), as.character(F.ms02$tag)))
Zs = data.frame(unique_tags)
names(Zs) = c('tag')
Zs$tag = as.character(Zs$tag)
Zs$freq = F.obs$freq[ match( Zs$tag , F.obs$tag) ]
Zs$freq[is.na(Zs$freq)] = 0;
Zs
```

```
##            tag freq
## 1   bike_movie    2
## 2    bike_read    1
## 3     bike_run    3
## 4    game_read    2
## 5    game_walk    2
## 6  movie_movie    2
## 7   movie_read    1
## 8    movie_run    2
## 9    read_read    1
## 10    read_run    5
## 11   read_walk    1
## 12    run_walk    4
## 13   bike_bike    0
## 14   bike_walk    0
## 15  movie_walk    0
## 16     run_run    0
## 17    game_run    0
## 18   walk_walk    0
```

```
## 19   bike_game    0
## 20  game_movie    0
```

calculate Z-score

```
for (i in 1 : length(Zs$tag)) {
#i = 2
  sub = F.ms02[ F.ms02$tag == Zs$tag[i], ]
  if(debug>0 ){
     print( paste( Zs$tag[i], "mean:", mean(sub$freq), "sd:", sd(sub$freq) ))
  }
  Zs$Z[i] = ( Zs$freq[i] - mean(sub$freq) ) / max(sd(sub$freq), 1E-2)
}
```

```
## [1] "bike_movie mean: 1.23076923076923 sd: 0.42966892442366"
## [1] "bike_read mean: 1.64583333333333 sd: 0.6010481388776"
## [1] "bike_run mean: 1.88888888888889 sd: 0.611340636419152"
## [1] "game_read mean: 1.36363636363636 sd: 0.48660709956248"
## [1] "game_walk mean: 1.12903225806452 sd: 0.340777100548239"
## [1] "movie_movie mean: 1.05555555555556 sd: 0.235702260395516"
## [1] "movie_read mean: 2.24 sd: 0.770899289327545"
## [1] "movie_run mean: 1.94 sd: 0.766918030369688"
## [1] "read_read mean: 1.4375 sd: 0.542109747858682"
## [1] "read_run mean: 3.94 sd: 0.766918030369688"
## [1] "read_walk mean: 1.64444444444444 sd: 0.679423403788949"
## [1] "run_walk mean: 2.51020408163265 sd: 0.915661167763305"
## [1] "bike_bike mean: 1 sd: 0"
## [1] "bike_walk mean: 1.24390243902439 sd: 0.488901207038705"
## [1] "movie_walk mean: 1.54347826086957 sd: 0.622058294486241"
## [1] "run_run mean: 1.47916666666667 sd: 0.618494511691824"
## [1] "game_run mean: 1.16129032258065 sd: 0.373878250552983"
## [1] "walk_walk mean: 1 sd: 0"
## [1] "bike_game mean: 1.07407407407407 sd: 0.266880256341812"
## [1] "game_movie mean: 1.10714285714286 sd: 0.314970394174356"
```

split the tags

```
tmp = as.vector(unlist(strsplit(as.character(Zs$tag), split="_")))
tmp2 = data.frame( matrix( tmp, nrow=10, ncol=2, byrow = T) )
names(tmp2) = c('c1', 'c2')
Zs = cbind( Zs, tmp2)
```

generate Z matrix

```
#unique categories
cats = sort( unique( c(as.character(Zs$c1), as.character(Zs$c2)) ))
Zmat = data.frame( matrix(NA, ncol=length(cats), nrow=length(cats)) )
names(Zmat) = cats;
```

```
rownames(Zmat) = cats;
for (i in 1:length(cats)){#row
  for (  j in 1:length(cats)) { #column
    tmp = sort(c(cats[i], cats[j]))
    mytag = paste(tmp[1], tmp[2], sep="_")
    if ( mytag %in% Zs$tag ) {
      Zmat[i,j] = Zs$Z[ Zs$tag == mytag ]
    } else { #tag was never found? !
      Zmat[i,j] = NA
    }
    if (debug ) {
      print (paste(mytag, Zmat[i,j] ) )
    }
  }
}
```

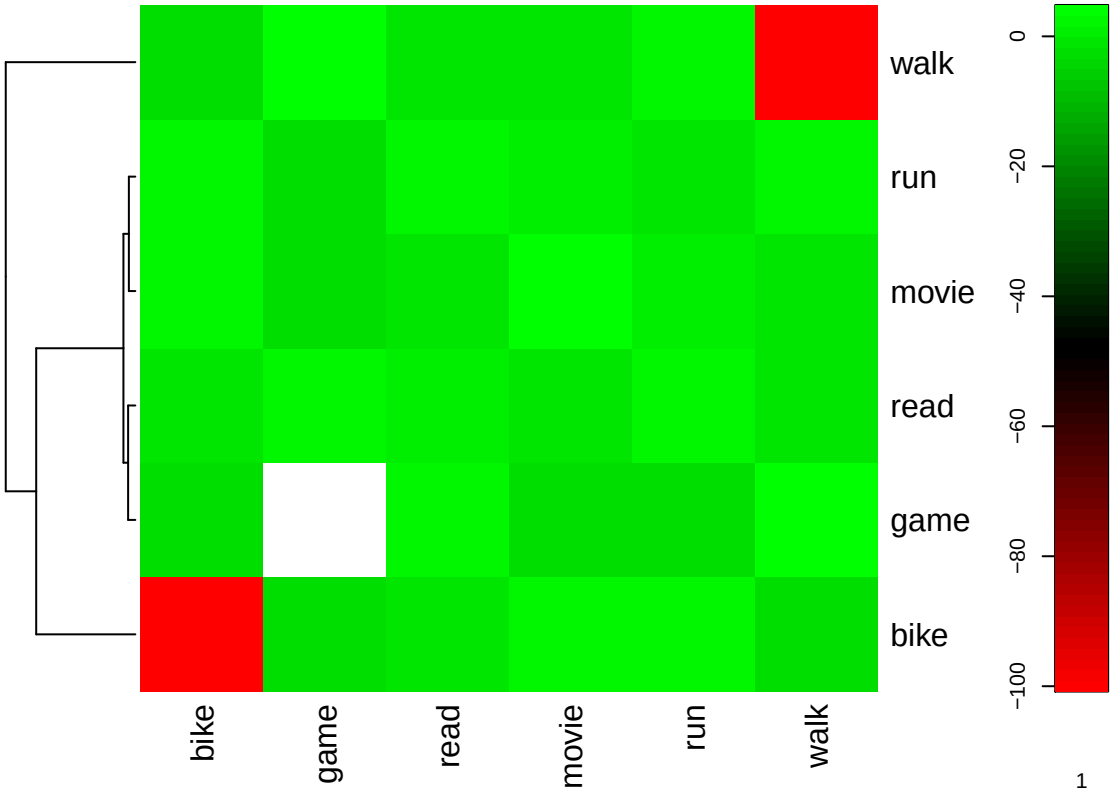
```
## [1] "bike_bike -100"
## [1] "bike_game -4.02455426563452"
## [1] "bike_movie 1.79028718509858"
## [1] "bike_read -1.07451182618977"
## [1] "bike_run 1.81749918935424"
## [1] "bike_walk -2.54428179173203"
## [1] "bike_game -4.02455426563452"
## [1] "game_game NA"
## [1] "game_movie -3.51506959898581"
## [1] "game_read 1.30775658007416"
## [1] "game_run -3.10606546613247"
## [1] "game_walk 2.55582825411179"
## [1] "bike_movie 1.79028718509858"
## [1] "game_movie -3.51506959898581"
## [1] "movie_movie 4.00693842672377"
## [1] "movie_read -1.60851101715459"
## [1] "movie_run 0.0782352189204333"
## [1] "movie_walk -2.48124375890579"
## [1] "bike_read -1.07451182618977"
## [1] "game_read 1.30775658007416"
## [1] "movie_read -1.60851101715459"
## [1] "read_read -0.807032158576953"
## [1] "read_run 1.38215553426099"
## [1] "read_walk -0.948516699381509"
## [1] "bike_run 1.81749918935424"
## [1] "game_run -3.10606546613247"
## [1] "movie_run 0.0782352189204333"
## [1] "read_run 1.38215553426099"
## [1] "run_run -2.39155989051636"
## [1] "run_walk 1.62701659829748"
## [1] "bike_walk -2.54428179173203"
## [1] "game_walk 2.55582825411179"
## [1] "movie_walk -2.48124375890579"
## [1] "read_walk -0.948516699381509"
## [1] "run_walk 1.62701659829748"
## [1] "walk_walk -100"
```

```
Zmat
```

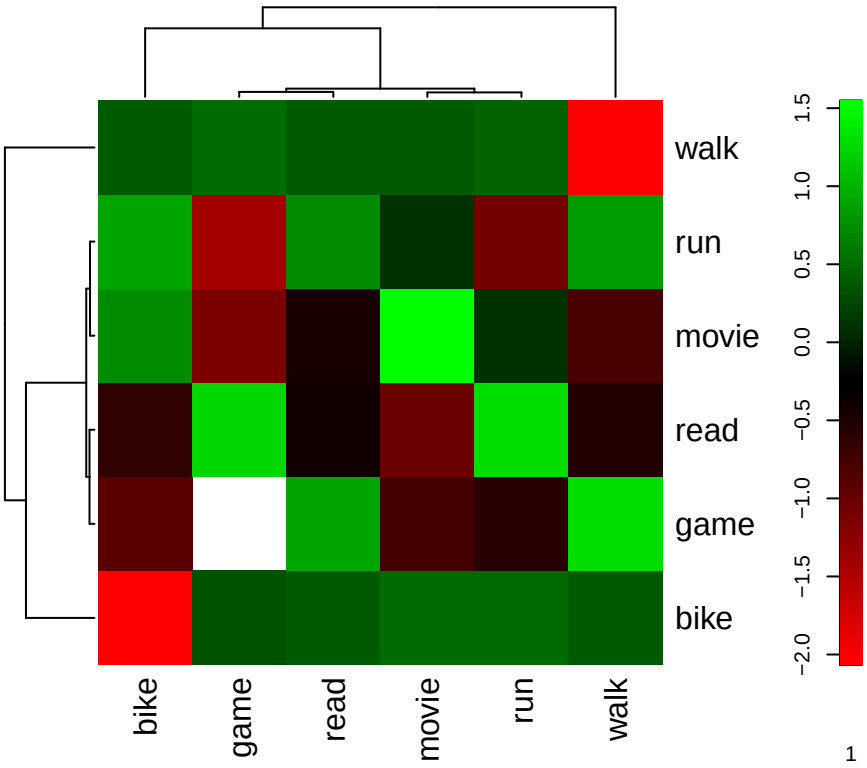
```
##              bike      game       movie       read         run
## bike  -100.000000 -4.024554  1.79028719 -1.0745118  1.81749919
## game    -4.024554        NA -3.51506960  1.3077566 -3.10606547
## movie    1.790287 -3.515070  4.00693843 -1.6085110  0.07823522
## read    -1.074512  1.307757 -1.60851102 -0.8070322  1.38215553
## run      1.817499 -3.106065  0.07823522  1.3821555 -2.39155989
## walk    -2.544282  2.555828 -2.48124376 -0.9485167  1.62701660
##               walk
## bike    -2.5442818
## game     2.5558283
## movie   -2.4812438
## read    -0.9485167
## run      1.6270166
## walk  -100.0000000
```

#heatmap

```
library(Heatplus)
heatmap_2(as.matrix(Zmat),scale="none", legend=4, do.dendro=c(T,F), col=RGBColVec(64));
```




```
heatmap_2(as.matrix(Zmat), legend=4, col=RGBColVec(64))
```


walk
run
movie
read
game
bike
bike
game
read
movie
run
walk
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
1