Case study example

Import packages

```
library("NLP")
library("tm")
library("SnowballC")
library("RColorBrewer")
library("wordcloud")
library("igraph")
library("Matrix")
library("plyr")
library("rARPACK")
library("readr")
```

Import datasets

```
sp_matrix_read <- readMM("~/Desktop/citation/sparse_matrix.mtx")

all_paper <- read_csv("~/Desktop/citation/all_paper.csv")
```

Conductance computation algorithm

```
sweepcut_2 <- function(P_matrix, aPPR){
  n = length(aPPR)
  N = c(1:n)[order(-abs(aPPR))]
  c = 0
  S = NULL
  set_list = vector(mode = "list", length = n)
  vol = 0
  phi_set = rep(0,n)
  for(k in 1:n){
    ia = N[k]
    if (k == 1){
      c = c + sum(P_matrix[ia,])
    }else if(k == n){
      c = c - sum(P_matrix[S,ia])
    }else {
      c = c -  sum(P_matrix[S,ia]) + sum(P_matrix[ia,-S])
    }
    vol = vol + sum(P_matrix[ia,])
    S = c(S,ia)
    phi_set[k] = c/vol
    set_list[[k]] = S
  }
  return(list(phi_set = phi_set,set_list = set_list)  )
}
```

Construct the graph transition matrix

```
#construct undirected adjacency matrix for the citations between source papers
nr = dim(sp_matrix_read)[1]
stats_matrix = sp_matrix_read[1:nr,1:nr]
stats_matrix = stats_matrix + t(stats_matrix)
stats_matrix[which(stats_matrix > 1, arr.ind = TRUE)] = 1
diag(stats_matrix) = 0
# compute node degrees
d = rowSums(stats_matrix)
d[which(d == 0)] = 1
#the graph transition matrix
P_matrix = stats_matrix/d
```

Basic information of the whole ciation network

```
network <- graph_from_adjacency_matrix(stats_matrix  , mode='undirected', diag=F )


graph.density(network, loops=TRUE)  #density
```

```
## [1] 0.0009771169
```

```
transitivity(network,type = "average") #average clustering coefficient
```

```
## [1] 0.2520555
```

Topic paper searching and preference vector construction

```
# Search the keywords "flu" and "influenza" in papers' titles
case_index = which(grepl(" Flu ",all_paper$`Article Title`, ignore.case = TRUE)
| grepl("influenza",all_paper$`Article Title`, ignore.case = TRUE))
partial_matrix = sp_matrix_read[,case_index]

cite_num = rowSums(partial_matrix)
cite_num[which(cite_num < 5 )] = 0
pi_vector = cite_num/sum(cite_num) # the preference vector
```

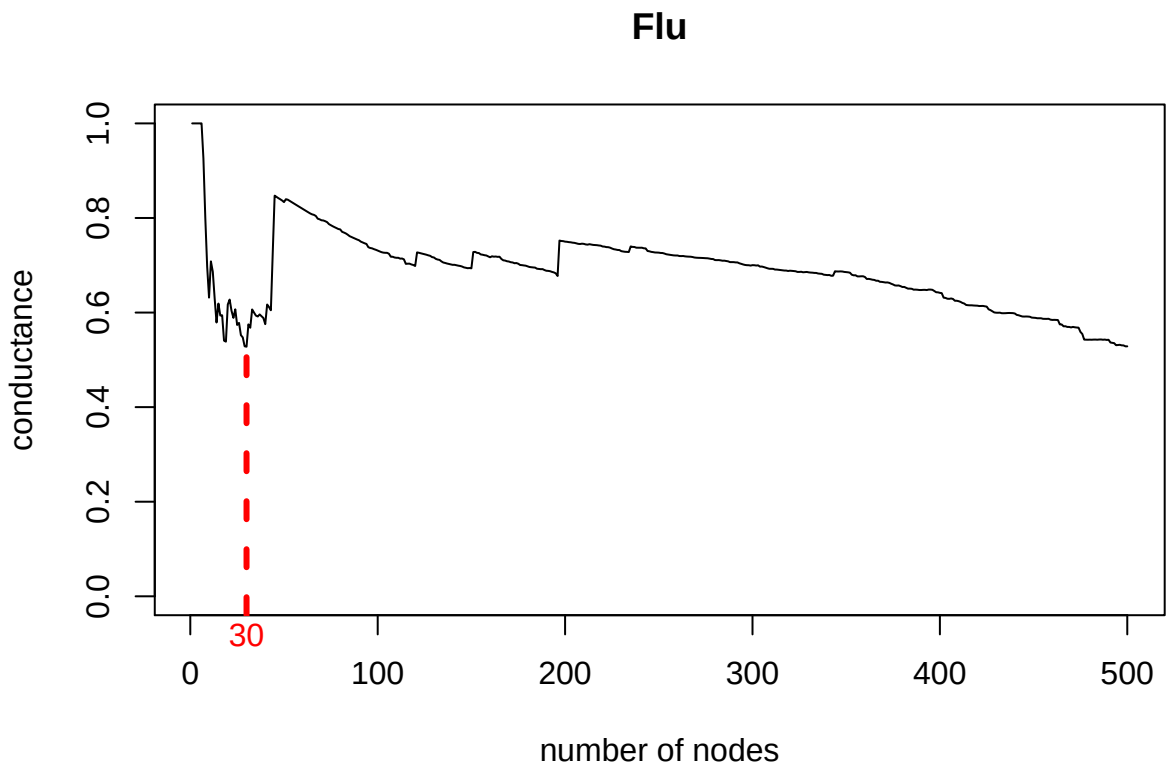
Compute the aPPR vector

```
alpha = 0.15 # teleportation constant
Pi = matrix(rep(pi_vector,nr), nrow  = nr, byrow = TRUE)
Q_matrix = alpha*Pi + (1 - alpha)*P_matrix
eig = eigs(t(Q_matrix), 1, which = "LM",tol = 1e-40)
eig_vector = abs(eig$vector[,1])
PPR = eig_vector
aPPR =  eig_vector/(d) #Adjusted Personalized PageRank
```

Decide the community size

```
#compute the conductance for each size
result = sweepcut_2(stats_matrix, aPPR)
#search for the local minimum on the conductance plot
select_reg = 100
x_min = which.min(result$phi_set[1:select_reg])
aPPR_high = sort(aPPR, decreasing =  TRUE)[x_min]
#the conductance plot
plot(c(1:500),result$phi_set[1:500], type = "l",xlab = "number of nodes"
     , ylab = "conductance", ylim = c(0,1), main = "Flu")
segments(x_min , -0.04, x_min  , result$phi_set[x_min], col= 'red',lwd = 3, lty = "dashed")
```

```
mtext(x_min, side = 1, at = x_min, col = "red")
```




```
#select the highly relevant papers
index_high = which(aPPR >= aPPR_high)
paper = all_paper[index_high,]
```

Proporties of subnetwork

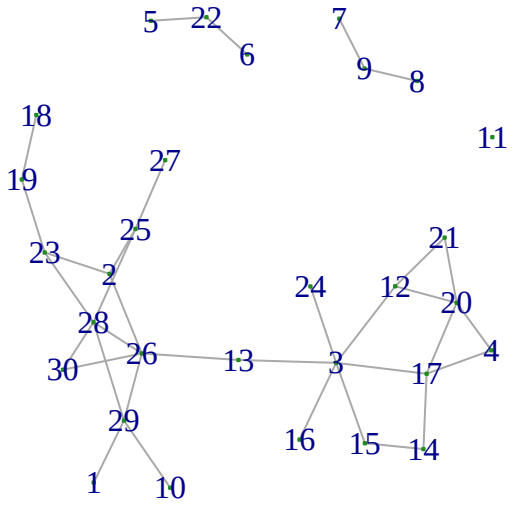
```
neighbor_matrix = stats_matrix[index_high,index_high]
network <- graph_from_adjacency_matrix(neighbor_matrix , mode='undirected', diag=F )

graph.density(network, loops=TRUE)  #density
```

```
## [1] 0.07311828
```

```
transitivity(network,type = "average") #average clustering coefficient
```

```
## [1] 0.2315789
```

```
#subnetwork appearance
plot(network,
     vertex.frame.color = "Forestgreen",
     vertex.shape=c("square"),
     vertex.size=1
)
```



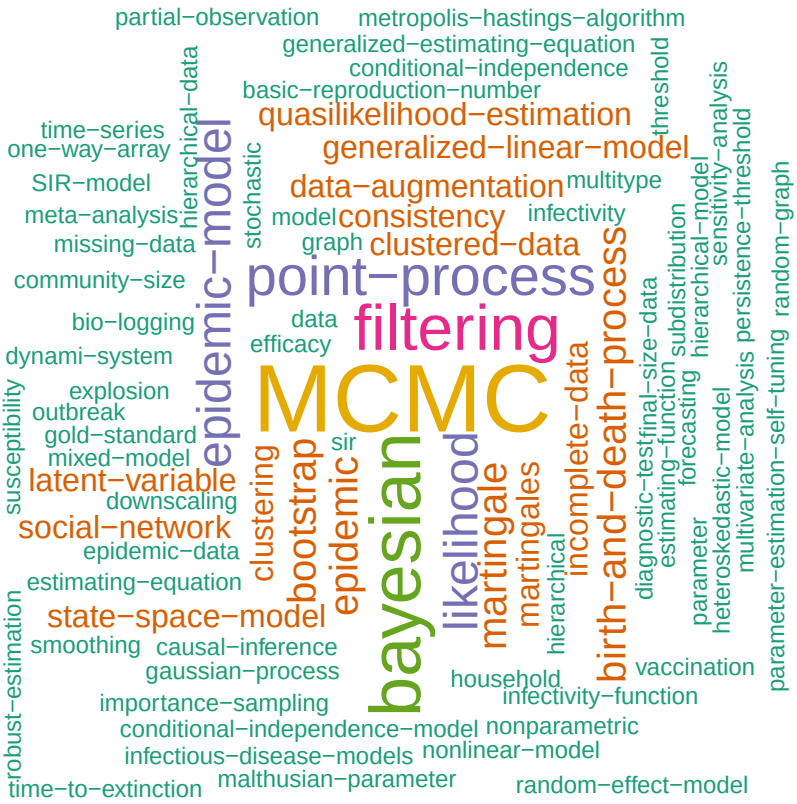
Wordcloud

```
#Selected papers' keywords
head(paper$`Author Keywords`)
```

```
## [1] "Birth and death process; Explosion; Malthusian parameter; Markov process; Marked point process;
## [2] "estimation; household epidemic; multitype epidemic; outbreak data; stochastic epidemic; thresho
## [3] "Flu; Google correlate; Google insights; Google searches; Google trends; H1N1; Infectious Diseas
## [4] "Iterated filtering; Partially observed Markov process; Plasmodium falciparum; Sequential Monte
## [5] "Bootstrap; Clustered data; Quasi-likelihood estimation; Robust estimation; Unbalanced data; Var
## [6] "Clustered data; Mixed model; Multivariate analysis; Nonparametric method"
```

```
# Cleaning keywords by hand
text = "Birth-and-death-process Explosion Malthusian-parameter MCMC
point-process Martingale Partial-observation Quasilikelihood-estimation
household epidemic multitype epidemic outbreak data stochastic epidemic
threshold parameter vaccination
filtering
filtering MCMC MCMC
Bootstrap Clustered-data Quasilikelihood-estimation Robust-estimation
Unbalanced-data Variance-component
Clustered-data Mixed-model Multivariate-analysis Nonparametric
Bayesian Diagnostic-test generalized-linear-model Gold-standard Meta-analysis
Missing-data hierarchical model
causal-inference likelihood principal-stratification sensitivity-analysis
 bootstrap clustering efficacy generalized-estimating-equation
 hierarchical-model random-effect-model secondary-attack-rate;
Bayesian conditional-independence-model data-augmentation MCMC
subdistribution martingale likelihood transformation-model
Bayesian filtering likelihood nonlinear-model
parameter-estimation-self-tuning smoothing
Epidemic-model Birth-and-death-process  transition-probability;
 Gaussian-process heteroskedastic-model latent-variable generalized-linear-model
Bayesian state-space-modeling SIR-model forecasting time-series
Bayesian downscaling bio-logging conditional-independence
MCMC filtering MCMC Bayesian SIR
epidemic-model simulation-model social-network
Graph social-network  latent-variable epidemic-model
State-space-model filtering MCMC likelihood
```

```
Dynami-system MCMC filtering importance-sampling state-space-model MCMC
 sums-of-squares bootstrap clustering hierarchical-data one-way-array
Bayesian MCMC Metropolis-Hastings-algorithm random-graph
stochastic-epidemic-models
basic-reproduction-number consistency final-size-data
 data-augmentation epidemic-data structured-community
 incomplete-data infectivity-function martingale n
community-size persistence-threshold quasi-stationary-distribution
stochastic-fade-out time-to-extinction
consistency point-process estimating-equation martingales
susceptibility  infectivity
Birth-and-death-process epidemic-model estimating-function
incomplete-data martingales transmission-parameter
point-process infectious-disease-models point-process point-process"


text = tolower(text)
text = Corpus(VectorSource(text))
df = TermDocumentMatrix(text)
m <- as.matrix(df)
v <- sort(rowSums(m),decreasing=TRUE)
dd <- data.frame(word = names(v),freq=v)
dd$word = as.character(dd$word)
dd$word[which(dd$word =="mcmc")] = "MCMC"
dd$word[which(dd$word =="sir-model")] = "SIR-model"
dd$word[which(dd$word =="em-algorithm")] = "EM-algorithm"
dd$word[which(dd$word =="pca")] = "PCA"
dd$word[which(dd$word =="etas")] = "ETAS"

#plot wordcloud
set.seed(1234)
wordcloud(words = dd$word , freq = dd$freq, min.freq = 1,
          max.words=100, random.order=FALSE, rot.per=0.35,  scale=c(3,.5),
          colors=brewer.pal(6, "Dark2"))
```


partial−observation
metropolis−hastings−algorithm
generalized−estimating−equation
conditional−independence
basic−reproduction−number
quasilikelihood−estimation
generalized−linear−model
data−augmentation
multitype
consistency
infectivity
clustered−data
point−process
filtering
MCMC
bayesian
likelihood
martingale
martingales
hierarchical
incomplete−data
birth−and−death−process
bootstrap
epidemic
clustering
sir
epidemic−model
hierarchical−data
stochastic
model
graph
data
efficacy
time−series
one−way−array
SIR−model
meta−analysis
missing−data
community−size
bio−logging
dynami−system
explosion
outbreak
gold−standard
mixed−model
latent−variable
downscaling
social−network
epidemic−data
estimating−equation
state−space−model
smoothing
causal−inference
gaussian−process
importance−sampling
conditional−independence−model
infectious−disease−models
malthusian−parameter
time−to−extinction
robust−estimation
susceptibility
household
infectivity−function
vaccination
nonparametric
nonlinear−model
random−effect−model
diagnostic−testfinal−size−data
estimating−function
forecasting
subdistribution
hierarchical−model
parameter
heteroskedastic−model
multivariate−analysis
persistence−threshold
sensitivity−analysis
parameter−estimation−self−tuning
random−graph
threshold