

Perlombongan Data Graf

Asas teori graf

Definisi penting:

1. Graf : terdiri daripada 2 set (verteks[V] dan sisi[E])
2. Sisi : Pasangan tidak tertib bagi dua verteks (v_1, v_2)
3. Darjah (degree) : bilangan kali verteks berlaku sebagai titik akhir untuk sisi
4. Gelung (Loop) : Sisi yang menghubungkan verteks yang sama
5. Sisi berganda (Multiple edge) : Sisi yang menghubungkan dua verteks yang sama
6. Graf Mudah (Simple graph) : Graf yang tidak mempunyai gelung dan sisi berganda
7. Subgraf : Graf yang terdiri daripada sebahagian verteks dan sisi graf asal
8. Klik : Subgraf yang mana setiap verteks berhubung dengan setiap verteks yang lain
9. Laluan (Path) : Sisipan verteks yang berlainan
10. Graf Isomorfik : Graf yang boleh wujud dalam bentuk yang berbeza tetapi mempunyai bilangan verteks, sisi dan juga ciri ketersambungan sisi yang sama
11. Graf Automorfik : Graf yang mempunyai struktur yang sama, tetapi mempunyai tingkahlaku hubungan yang berbeza. Oleh itu, ianya bukanlah graf yang sama secara tepat.

Jenis jenis data graf

Graf terarah dan tak terarah

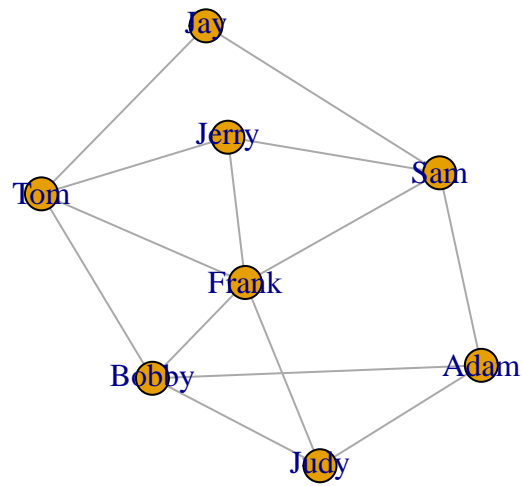
Graf tak terarah

input hubungan nod-nod

```
g <- graph_from_literal(1-2,1-3,1-7,3-4,2-3,2-4,3-5,4-5,
                        4-6,4-7,5-6,5-8,6-7,7-8)
```

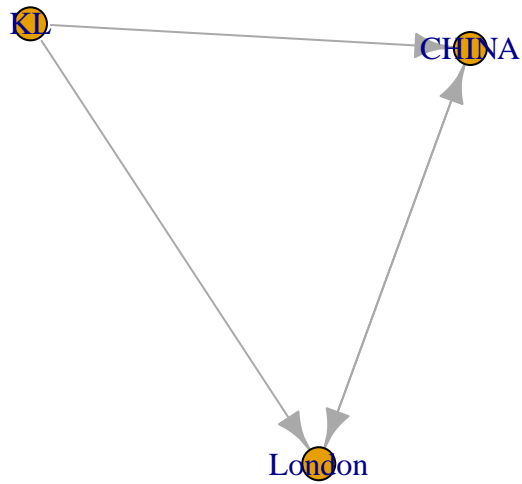
Labelkan nod-nod

```
V(g)$name <- c("Adam", "Judy", "Bobby", "Sam", "Frank",
               "Tom", "Jerry", "Jay")
set.seed(12)
plot(g)
```



Graf terarah

```
dg<- graph_from_literal(KL-->CHINA,KL-->London,CHINA-->London)
plot(dg)
```



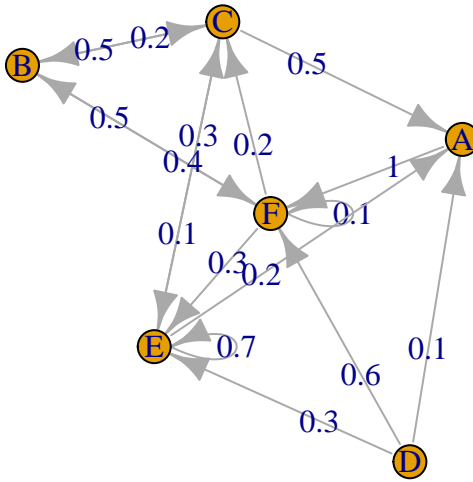
Graf berwajaran

```

m <- read.table(row.names=1, header=T, text =
               "  A  B  C  D  E  F
A  0  0  0  0  0  1
B  0  0  0.5  0  0  0.5
C  0.5  0.2  0  0  0.3  0
D  0.1  0  0  0  0.3  0.6
E  0.2  0  0.1  0  0.7  0
F  0  0.4  0.2  0  0.3  0.1")
m<- as.matrix(m)

ig <- graph_from_adjacency_matrix(m, weighted=T)
plot(ig, edge.label=E(ig)$weight)

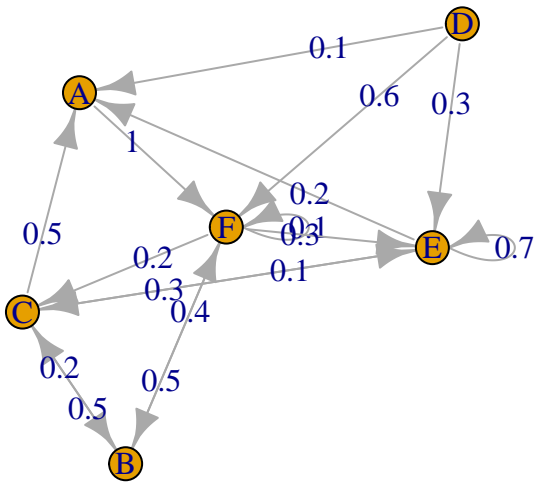
```



Graf berkitar dan tak berkitar

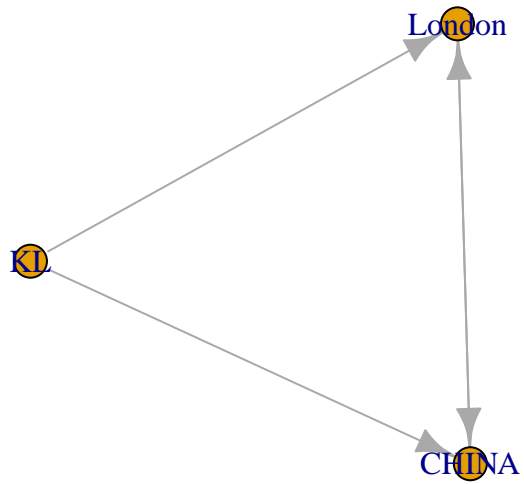
Grad berkitar

```
plot(ig, edge.label=E(ig)$weight)
```



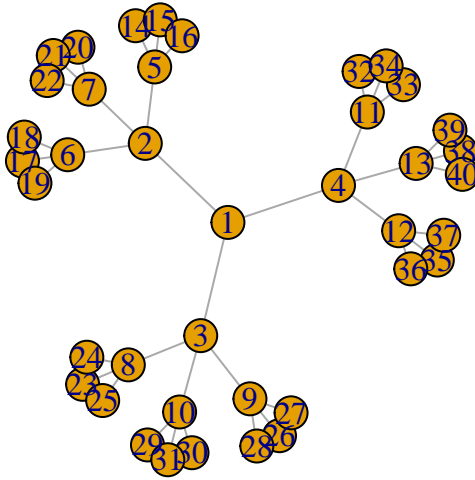
Graf tak berkitar

```
plot(dg)
```



Graf Pokok

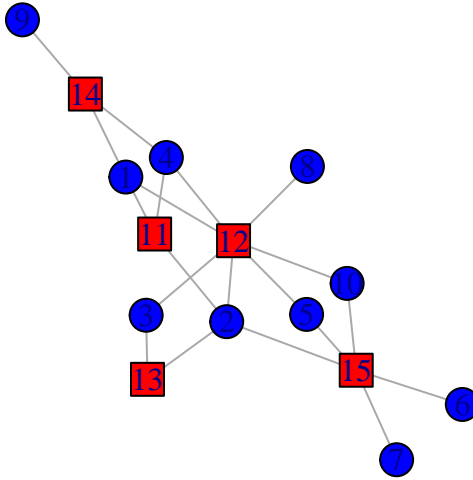
```
tr <- make_tree(40, children=3, mode='undirected')  
plot(tr)
```



Graf Bipartit

```
gb <- sample_bipartite(10,5,p=0.4)
colo<- c('blue','red')
shape <- c('circle','square')

plot(gb, vertex.color=colo[as.numeric(V(gb)$type)+1],
      vertex.shape=shape[as.numeric(V(gb)$type)+1])
```



Hiper Graf

```
library(HyperG)
```

```
## Loading required package: mclust
```

```
## Package 'mclust' version 6.1.1
```

```
## Type 'citation("mclust")' for citing this R package in publications.
```

```
##
```

```
## Attaching package: 'HyperG'
```

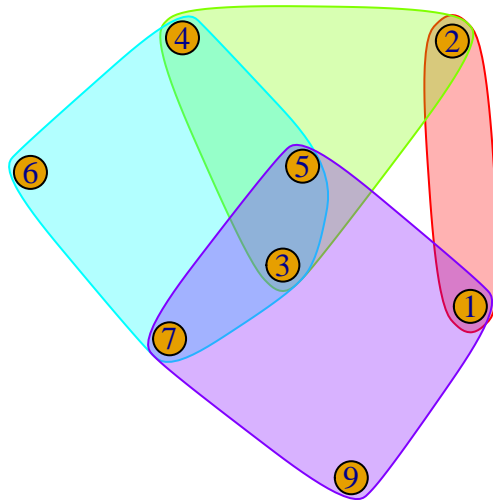
```
## The following objects are masked from 'package:igraph':
```

```
##
```

```
## is.simple, line.graph
```

```
h <- hypergraph_from_edgelist(list(1:2, 2:5, 3:7,  
                                  c(1,3,5,7,9)))
```

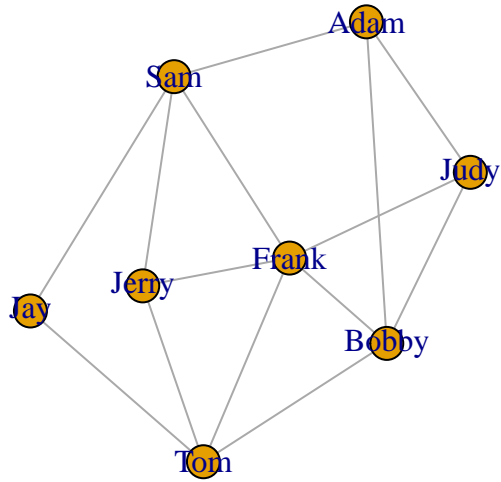
```
plot(h)
```

Perwakilan Data Graf

Senarai Bersebelahan

```
Adj.list1 <- as_adj_list(g)  
plot(g)
```



Adj.list1

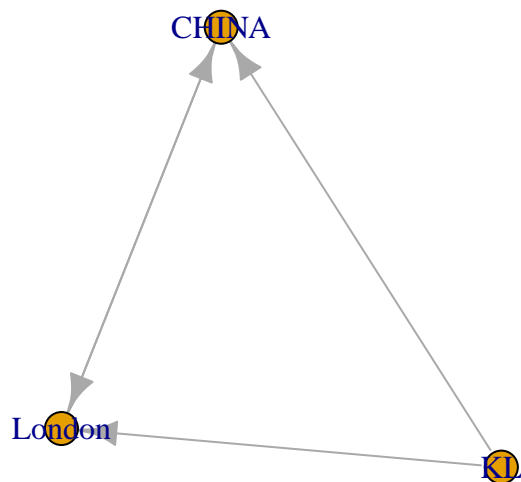
```

## $Adam
## + 3/8 vertices, named, from 35d901e:
## [1] Judy Bobby Sam
##
## $Judy
## + 3/8 vertices, named, from 35d901e:
## [1] Adam Bobby Frank
##
## $Bobby
## + 4/8 vertices, named, from 35d901e:
## [1] Adam Judy Frank Tom
##
## $Sam
## + 4/8 vertices, named, from 35d901e:
## [1] Adam Frank Jerry Jay
##
## $Frank
## + 5/8 vertices, named, from 35d901e:
## [1] Judy Bobby Sam Tom Jerry
##
## $Tom
## + 4/8 vertices, named, from 35d901e:
## [1] Bobby Frank Jerry Jay
##

```

```
## $Jerry
## + 3/8 vertices, named, from 35d901e:
## [1] Sam   Frank Tom
##
## $Jay
## + 2/8 vertices, named, from 35d901e:
## [1] Sam Tom
```

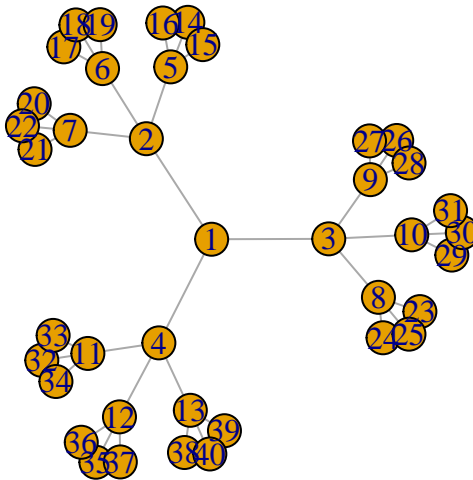
```
Adj.list2 <- as_adj_list(dg)
plot(dg)
```



```
Adj.list2
```

```
## $KL
## + 2/3 vertices, named, from 3758e89:
## [1] CHINA London
##
## $CHINA
## + 3/3 vertices, named, from 3758e89:
## [1] KL      London London
##
## $London
## + 3/3 vertices, named, from 3758e89:
## [1] KL      CHINA CHINA
```

```
Adj.list3 <- as_adj_list(tr)
plot(tr)
```



```
head(Adj.list3,5)
```

```
## [[1]]
## + 3/40 vertices, from 38ffc94:
## [1] 2 3 4
##
## [[2]]
## + 4/40 vertices, from 38ffc94:
## [1] 1 5 6 7
##
## [[3]]
## + 4/40 vertices, from 38ffc94:
## [1] 1 8 9 10
##
## [[4]]
## + 4/40 vertices, from 38ffc94:
## [1] 1 11 12 13
##
## [[5]]
## + 4/40 vertices, from 38ffc94:
## [1] 2 14 15 16
```

Seanari Sisi

```
Ed.list1<- as.data.frame(as_edgelist(g))
Ed.list2<- as.data.frame(as_edgelist(dg))
Ed.list3<- as.data.frame(as_edgelist(gb))
Ed.list1
```

```
##      V1    V2
## 1  Adam  Judy
## 2  Adam Bobby
## 3  Adam   Sam
## 4  Judy Bobby
## 5  Judy Frank
## 6 Bobby Frank
## 7 Bobby   Tom
## 8   Sam Frank
## 9   Sam Jerry
## 10  Sam   Jay
## 11 Frank   Tom
## 12 Frank Jerry
## 13   Tom Jerry
## 14   Tom   Jay
```

Matriks Bersebelahan

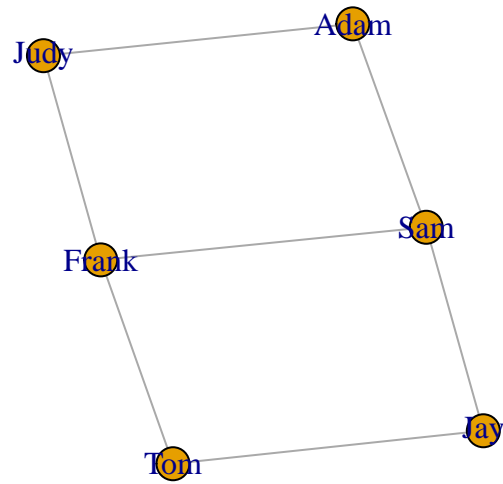
```
Adj.M1<- as_adjacency_matrix(g)
Adj.M2<- as_adjacency_matrix(dg)
Adj.M3<- as_adjacency_matrix(ig)
Adj.M1
```

```
## 8 x 8 sparse Matrix of class "dgCMatrix"
##      Adam Judy Bobby Sam Frank Tom Jerry Jay
## Adam      .    1    1    1      .    .    .    .
## Judy      1    .    1    .    1    .    .    .
## Bobby      1    1    .    .    1    1    .    .
## Sam        1    .    .    .    1    .    1    1
## Frank      .    1    1    1    .    1    1    .
## Tom        .    .    1    .    1    .    1    1
## Jerry      .    .    .    1    1    1    .    .
## Jay        .    .    .    1    .    1    .    .
```

Teknik Manipulasi Graf

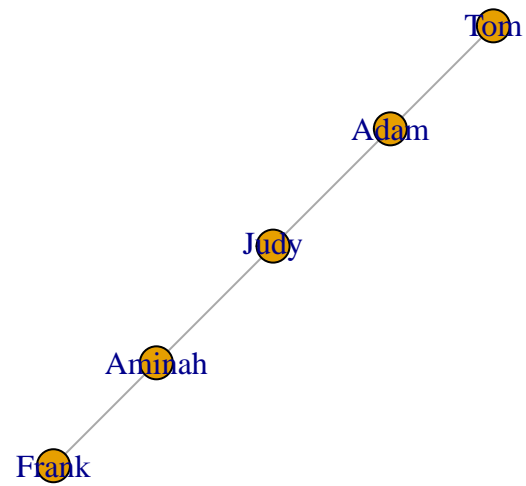
Keluarkan verteks tertentu

```
h <- g-vertices(c("Jerry","Bobby"))
plot(h)
```



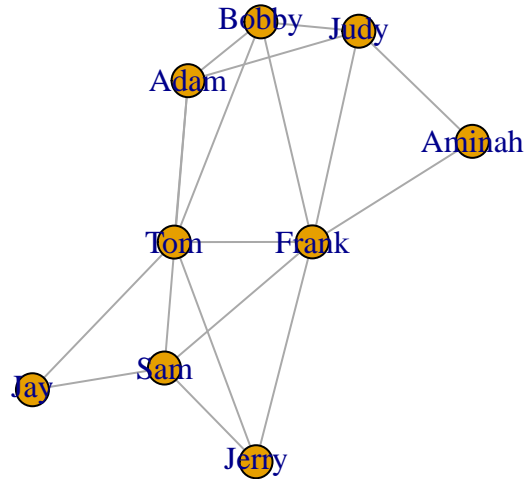
Menjana Subgraf

```
h2 <- graph_from_literal("Adam"- "Judy", "Adam"- "Tom",  
                        "Judy"- "Aminah", "Aminah"- "Frank")  
plot(h2)
```



Gabungkan graf

```
h3 <- union(h2,g)  
plot(h3)
```



Mengubah suai sisi data

E(h3)

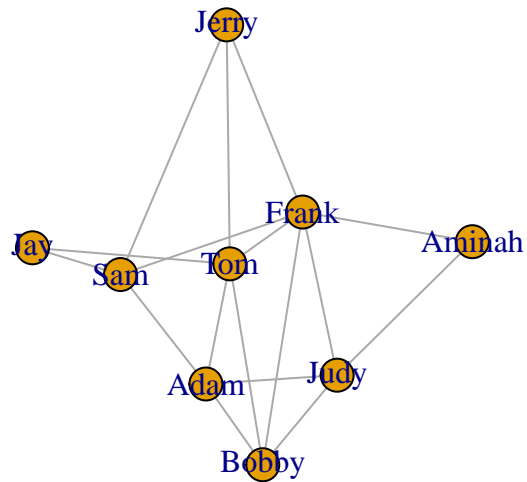
```
## + 17/17 edges from 3d60871 (vertex names):
## [1] Sam --Jay Sam --Jerry Frank --Jerry Frank --Sam Frank --Bobby
## [6] Aminah--Frank Tom --Jay Tom --Jerry Tom --Bobby Tom --Frank
## [11] Judy --Bobby Judy --Frank Judy --Aminah Adam --Sam Adam --Bobby
## [16] Adam --Tom Adam --Judy
```

```
E(h3)$type<- c("email", "phone", "FB", "email", "class",
               "Twitter", "neighbour", "phone", "FB", "email",
               "class", "neighbour", "phone", "email", "email",
               "FB", "neighbour")
E(h3)$weight<- c(10, 1, 3, 2, 2, 2, 1, 5, 9, 8, 1, 6, 2, 9,
                 3, 10, 7)
edge_attr(h3)
```

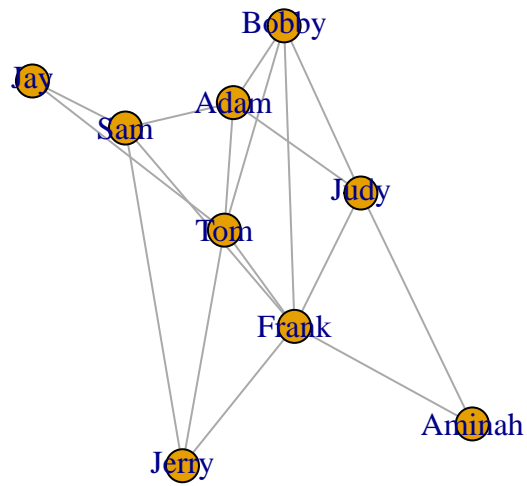
```
## $type
## [1] "email" "phone" "FB" "email" "class" "Twitter"
## [7] "neighbour" "phone" "FB" "email" "class" "neighbour"
## [13] "phone" "email" "email" "FB" "neighbour"
##
## $weight
## [1] 10 1 3 2 2 2 1 5 9 8 1 6 2 9 3 10 7
```


Pengvisualan Graf

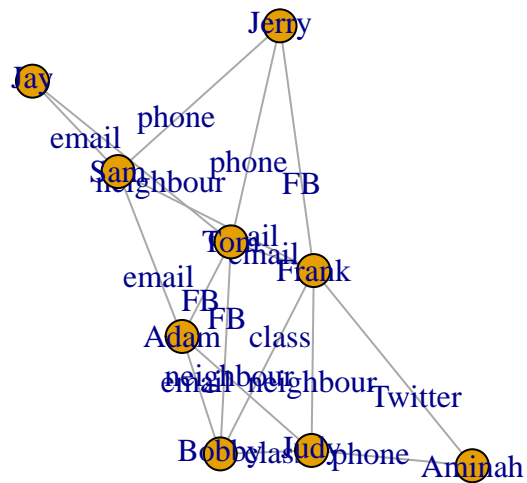
```
plot(h3)
```



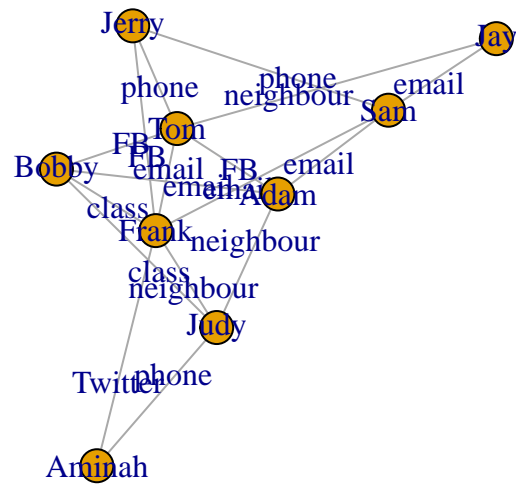
```
plot(h3, vertex.label=V(h3)$gender)
```



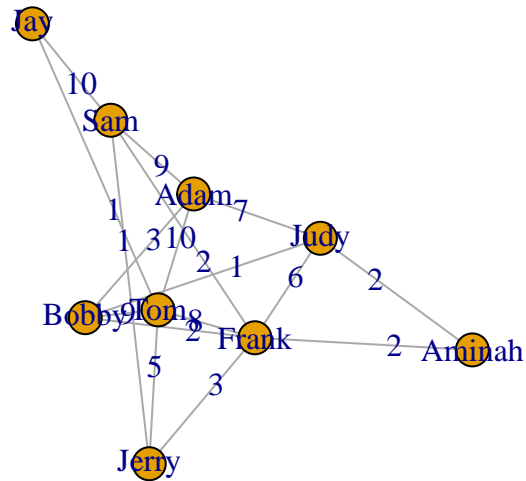
```
plot(h3, vertex.label=V(h3)$gender, edge.label=E(h3)$type)
```



```
plot(h3, vertex.label=V(h3)$name, edge.label=E(h3)$type)
```



```
plot(h3, vertex.label=V(h3)$name, edge.label=E(h3)$weight)
```



Analisis prominensi nod

```
library(statnet)
```

```
## Loading required package: tergm
```

```
## Loading required package: ergm
```

```
## Loading required package: network
```

```
##
```

```
## 'network' 1.19.0 (2024-12-08), part of the Statnet Project
```

```
## * 'news(package="network")' for changes since last version
```

```
## * 'citation("network")' for citation information
```

```
## * 'https://statnet.org' for help, support, and other information
```

```
##
```

```
## Attaching package: 'network'
```

```
## The following object is masked from 'package:HyperG':
```

```
##
```

```
## has.loops
```

```

## The following objects are masked from 'package:igraph':
##
##   %c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices,
##   get.edge.attribute, get.edges, get.vertex.attribute, is.bipartite,
##   is.directed, list.edge.attributes, list.vertex.attributes,
##   set.edge.attribute, set.vertex.attribute
##
## 'ergm' 4.8.1 (2025-01-20), part of the Statnet Project
## * 'news(package="ergm")' for changes since last version
## * 'citation("ergm")' for citation information
## * 'https://statnet.org' for help, support, and other information
##
## 'ergm' 4 is a major update that introduces some backwards-incompatible
## changes. Please type 'news(package="ergm")' for a list of major
## changes.
##
## Loading required package: networkDynamic
##
## 'networkDynamic' 0.11.5 (2024-11-21), part of the Statnet Project
## * 'news(package="networkDynamic")' for changes since last version
## * 'citation("networkDynamic")' for citation information
## * 'https://statnet.org' for help, support, and other information
##
## Registered S3 method overwritten by 'tergm':
##   method                from
##   simulate_formula.network ergm
##
## 'tergm' 4.2.1 (2024-10-08), part of the Statnet Project
## * 'news(package="tergm")' for changes since last version
## * 'citation("tergm")' for citation information
## * 'https://statnet.org' for help, support, and other information
##
## Attaching package: 'tergm'
##
## The following object is masked from 'package:ergm':
##
##   snctrl
##
## Loading required package: ergm.count
##
## 'ergm.count' 4.1.2 (2024-06-15), part of the Statnet Project
## * 'news(package="ergm.count")' for changes since last version
## * 'citation("ergm.count")' for citation information
## * 'https://statnet.org' for help, support, and other information
##
## Loading required package: sna

```

```

## Loading required package: statnet.common

##
## Attaching package: 'statnet.common'

## The following object is masked from 'package:ergm':
##
##      snctrl

## The following objects are masked from 'package:base':
##
##      attr, order

## sna: Tools for Social Network Analysis
## Version 2.8 created on 2024-09-07.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
## For citation information, type citation("sna").
## Type help(package="sna") to get started.

##
## Attaching package: 'sna'

## The following objects are masked from 'package:igraph':
##
##      betweenness, bonpow, closeness, components, degree, dyad.census,
##      evcent, hierarchy, is.connected, neighborhood, triad.census

## Loading required package: tsna

##
## 'statnet' 2019.6 (2019-06-13), part of the Statnet Project
## * 'news(package="statnet")' for changes since last version
## * 'citation("statnet")' for citation information
## * 'https://statnet.org' for help, support, and other information

## unable to reach CRAN

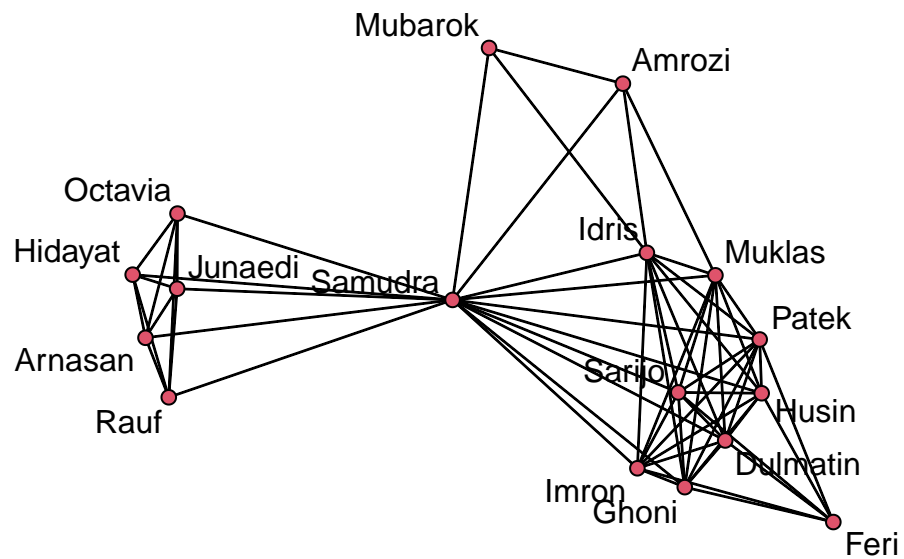
#import pakej daripada platfom github
library(devtools)

## Loading required package: usethis

#install_github("DougLuke/UserNetR")
library(UserNetR)

data(Bali)
par(mar=c(1,1,1,1))
plot(Bali, displaylabels=T)

```



Nama nod

```
Name<- Bali%v%"vertex.names"
```

Role nod

```
Role<- Bali%v%"role"
```

Atribut Sisi

```
Attr<- Bali%e%"IC"
```

Ukuran Nod Prominen

Kepusatan Darjah


```
deg<- degree(Bali)
```

Kepusatan Kedekatan

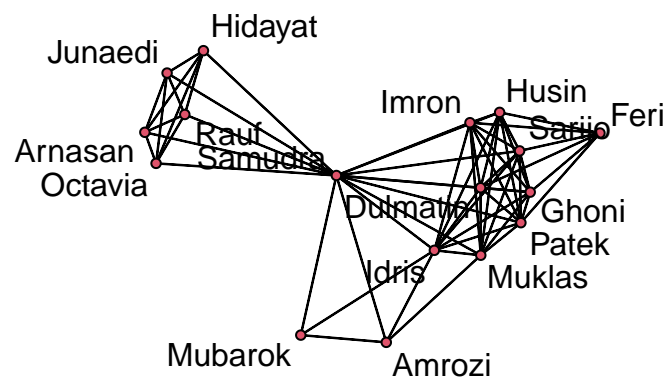
```
cls<- closeness(Bali)
```

Kepusatan Antara

```
btw<- betweenness(Bali)
```

Pemusatan

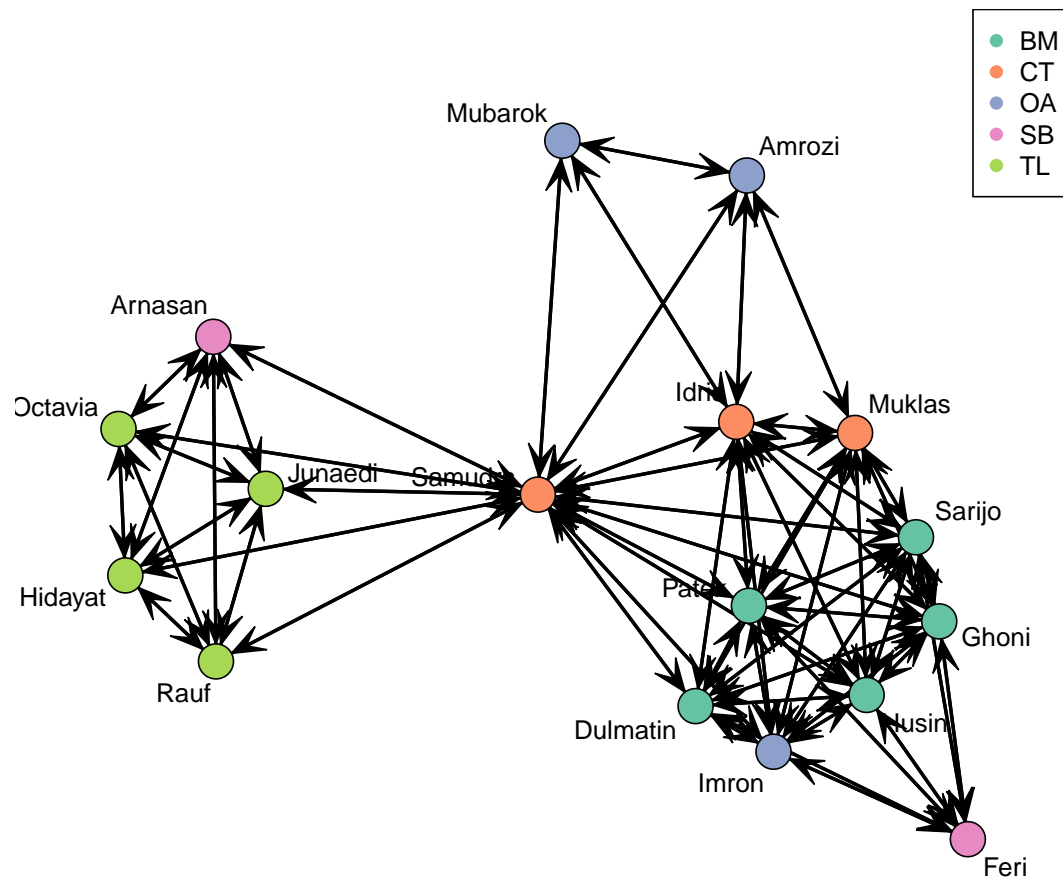
```
library(statnet)  
library(UserNetR)  
data(Bali)  
plot(Bali, displaylabels=T)
```



```

rolecat <- Bali%%v%"role"
name <- Bali%%v%"vertex.names"
library(RColorBrewer)
library(ggplot2)
my_pal <- brewer.pal(5, "Set2")
gplot(Bali, displaylabels=T,
      vertex.col=my_pal[as.factor(rolecat)])
legend("topright", legend=c("BM", "CT", "OA", "SB", "TL"),
      col=my_pal, pch=19, cex=1)

```



Ukuran Pemusatan Graf

```
centralization(Bali, degree)
```

```
## [1] 0.5375
```

```
centralization(Bali, closeness)
```

```
## [1] 0.3343513
```

Analisis titik potong

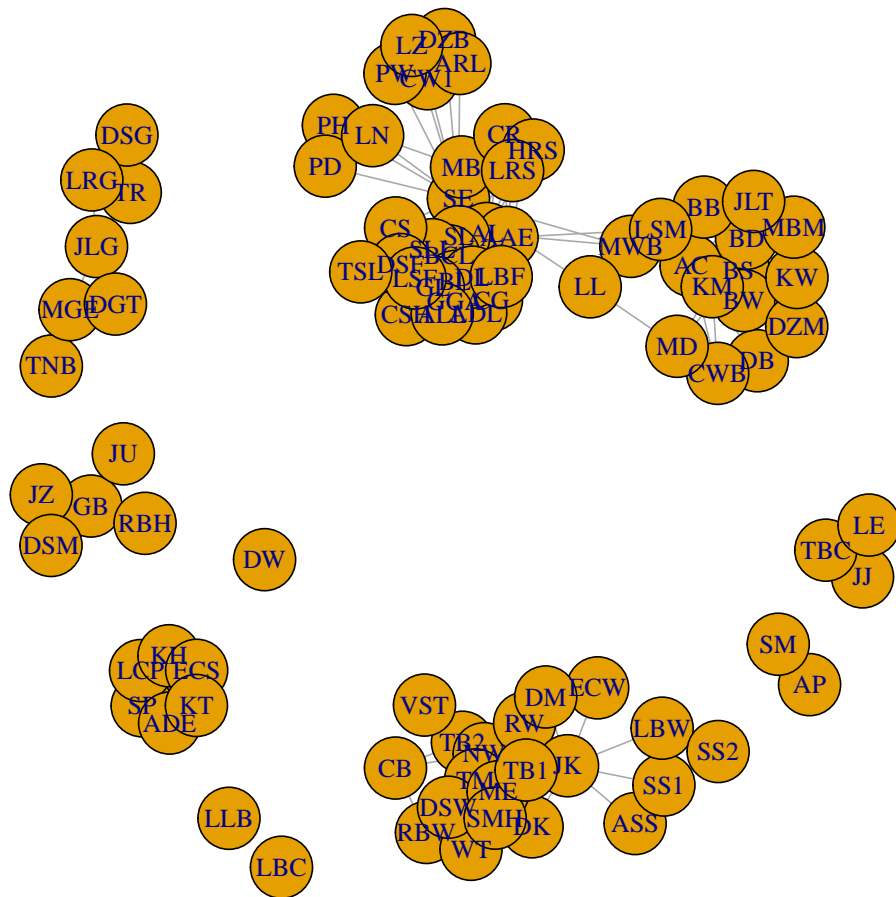
```
net <- Bali  
cpnet <- cutpoints(net, return.indicator = T)  
# cpnet = 4 "TRUE"  
name[4]
```

```
## [1] "Samudra"
```

Analisis Sub Kumpulan

```
data(Facebook)  
plot(Facebook)
```

```
## This graph was created by an old(er) igraph version.  
## i Call 'igraph::upgrade_graph()' on it to use with the current igraph version.  
## For now we convert it on the fly...
```



Clique

```
clique <- cliques(Facebook)
```

Clique Terbesar

```
max_clique <- max_cliques(Facebook)
```

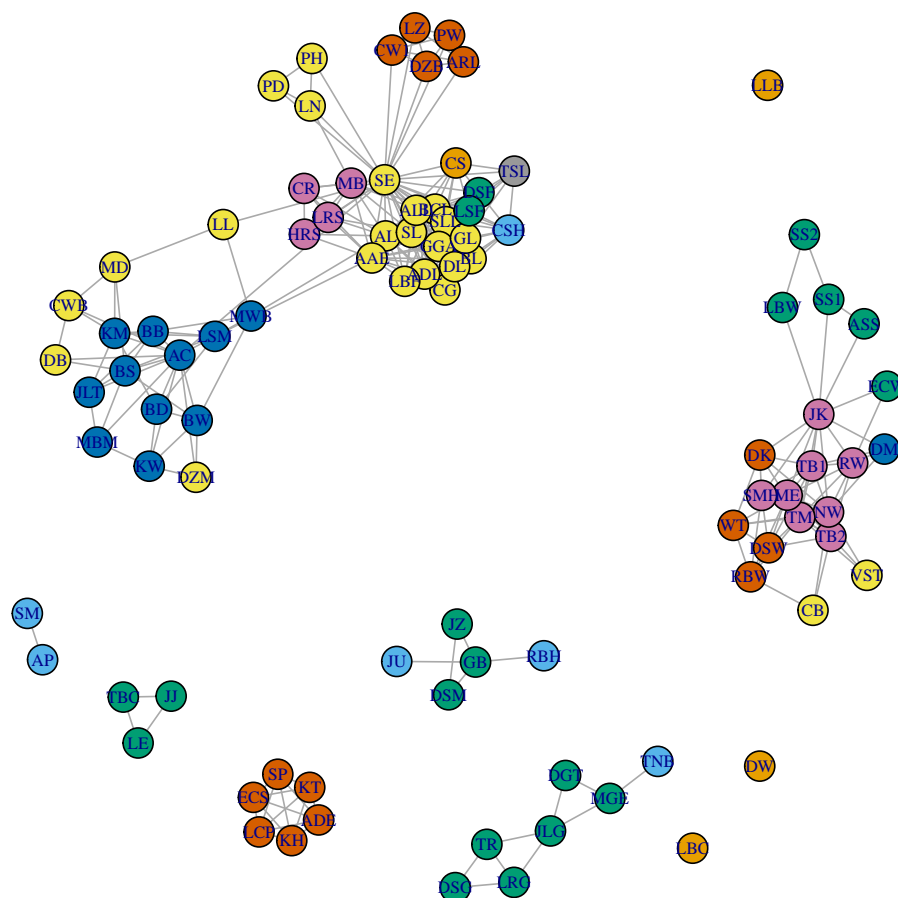
K-teras

```
k_core <- coreness(Facebook)
```

plot teras yang sama

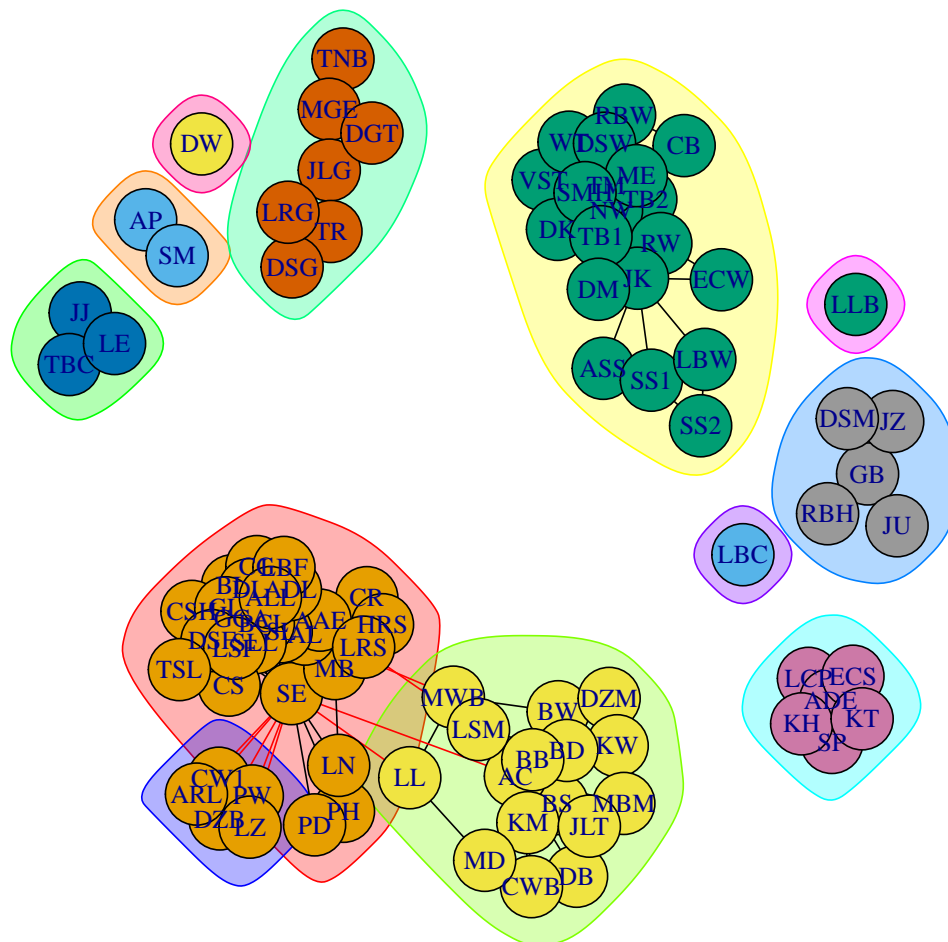
```
V(Facebook)$color<- as.factor(k_core)
plot(Facebook, main='sub=kumpulan mengikuti k-teras',
      vertex.size=7, vertex.label.cex=0.7)
```

sub=kumpulan mengikuti k-teras



Pengesanan komuniti & ukuran modulariti

```
Komuniti_D <- cluster_louvain(Facebook)
plot(Komuniti_D, Facebook)
```



```
skor_modulariti <- modularity(Komuniti_D)
skor_modulariti
```

```
## [1] 0.6321301
```