

Visualization of Network and it's Node descriptive

Using the Teacher-Student network

```
library(igraph)
```

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

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

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

```
g4 <- graph( c('Susann', 'Jeanette', 'Harrison', 'Myron', 'Julie', 'Colten', 'Brandyn', 'Leland', 'Catherine',  
'Tracy', 'Bee', 'Maureen', 'Annika', 'Hans', 'Spenser', 'Angelo', 'Blanche', 'Stanford', 'Hope', 'Catherine',  
'Almira', 'Thomas', 'Henry', 'Ralph', 'Louvenia', 'German', 'Lura', 'Wilma', 'Lauretta', 'Maudie', 'Martha',  
'Henry', 'Loyd', 'Mayra', 'Noelle', 'Fabian', 'Ira', 'Zenobia', 'Noelle', 'Natalie', 'Brooke', 'German',  
'Lauretta', 'Zenobia', 'Hallie', 'Braxton', 'Lorraine', 'Alicia', 'Allie', 'Jaylon', 'Myrna', 'Darrel', 'Lura',  
'Hermon', 'Wyatt', 'Raoul', 'Althea', 'Laurel', 'Leola', 'Darrel', 'Giles', 'Merritt', 'Angelo', 'Kendrick',  
'Travis', 'Henry', 'Owen', 'Almira', 'Michelle', 'Lou', 'Stephan', 'Leola', 'Wyatt', 'Allie', 'Eli',  
'Brandyn', 'Mayra', 'Chin', 'Clarence', 'Marlee', 'Glenn', 'Reid', 'Glenn', 'Jerrold', 'Dale', 'Helena',  
'Alvie', 'Elden', 'Chaim', 'Merritt', 'Augustine', 'Curtiss', 'Alvie', 'Allie', 'Pollie', 'Catherine', 'Angelita',  
'Glenn', 'Allie', 'Braden', 'Manford', 'Raoul', 'Dwight', 'Pollie', 'Zona', 'Reggie', 'Dominic', 'Amy', 'Santos',  
'Faron', 'Leslie', 'Etha', 'Maureen', 'Juanita', 'Gust', 'Noelle', 'Darrel', 'Darin', 'Faron', 'Zona', 'Myron',  
'Hans', 'German', 'Elden', 'Mayra', 'Merritt', 'Myron', 'Gerald', 'Andrea', 'German', 'Angelita', 'Giles',  
'Reggie', 'Wyatt', 'Alvie', 'Darrel', 'Jarrett', 'Newell', 'Brandyn', 'Eloisa', 'Santos', 'Lou', 'Quintin',  
'Zechariah', 'Leola', 'Anthony', 'Eliga', 'Gust', 'Obie', 'Obie', 'Brandyn', 'Brooke', 'Dominic', 'Jacob',  
'Maureen', 'Hardie', 'Noelle', 'Braxton', 'Noelle', 'Leola', 'Marlee', 'Helena', 'Ira', 'Kermit', 'Wilda',  
'Eli', 'Chance', 'Hardie', 'Andrea', 'Johnnie', 'Lauretta', 'Tracy', 'Kathryn', 'Spenser', 'Lou', 'Dana',  
'Dominic', 'Rachelle', 'Glenn', 'Darrel', 'Deidra', 'Orpha', 'Hermon', 'Rosalind', 'Hans', 'Curtis',  
'Josue', 'Wyatt', 'Agatha', 'Wyatt', 'Chance', 'Glenn', 'Kendrick', 'Brooke', 'Hollie', 'Beverly', 'Katarina',  
'Josue', 'Leslie', 'Colten', 'Mayra', 'Darin', 'Hermon', 'Myron', 'Alicia', 'Leila', 'Kermit', 'Lamar', 'Dana',  
'Johnnie', 'Althea', 'Judson', 'Colten', 'Hope', 'Gust', 'Clarence', 'Leland', 'Matthew', 'Lorraine', 'Althea',  
'Colten', 'Jewel', 'Leila', 'Eli', 'Zenobia', 'German', 'Jaylon', 'Henry', 'Michelle', 'Johnnie', 'Judson',  
'Johnnie', 'Leslie', 'Wyatt', 'Lou', 'Maureen', 'Jacob', 'Noelle', 'Johnnie', 'Alvie', 'Leila', 'Myron',  
'Wilda', 'Brandyn', 'Kendrick', 'Gerald', 'Tyson', 'Jacob', 'Ari', 'Theo', 'Harrison', 'Carmen', 'Leonard',  
'Braden', 'Desi', 'Zona', 'Maureen', 'Ralph', 'Thomas', 'Myron', 'Austin', 'Leola', 'Loyd', 'Quintin', 'Lorraine',  
'Glenn', 'Merritt', 'Janet', 'Myron', 'Alicia', 'Maureen', 'Leslie', 'Carla', 'Vicky', 'Harley', 'Leslie',  
'Hannah', 'Gust', 'Andrea', 'Giles', 'Wilma', 'Henry', 'Hermon', 'Stefani', 'Henry', 'Harley', 'Noelle',  
'Kermit', 'Laurel', 'Hermon', 'Sebastian', 'Andrea', 'Eliga', 'Sigmund', 'Darrel', 'Noelle', 'Urban',  
'Almira', 'Noelle', 'Dwight', 'Beverly', 'Wyatt', 'Juanita', 'Reid', 'Kermit', 'Myrna', 'Pollie', 'Hope',  
'Leola', 'Leslie', 'Jacob', 'Pollie', 'Tiffany', 'Gust', 'Leland', 'Eliga', 'Lamar', 'Ira', 'Dale', 'Noelle',  
'Jarrett', 'Zenobia', 'Orpha', 'Glenn', 'Amy', 'Hermon', 'Myrna', 'Brandyn', 'Nell', 'Leslie', 'Edd',  
'Lura', 'Kendrick', 'Obie', 'Augustine', 'Zona', 'Glenn', 'Dominic', 'Jennifer', 'Rachelle', 'Faron',  
'Darrel', 'Pollie', 'Almira', 'Glenn', 'Wilma', 'Angus', 'Leila', 'Sherri'))
```

```
## Warning in matrix(edges, ncol = 2, byrow = TRUE): data length [349] is not a  
## sub-multiple or multiple of the number of rows [175]
```

```
V(g4)$name
```



```
edge_density(g4, loops=F)
```

```
## [1] 0.008266415
```

2. Finding the density of a graph- For directed graph

```
ecount(g4) / (vcount(g4) * (vcount(g4) - 1))
```

```
## [1] 0.008266415
```

Result: Density i.e. the proportion of present edges from all possible edges in the network was found.

3. Finding Reciprocity

```
reciprocity(g4)
```

```
## [1] 0
```

```
dyad_census(g4) # Mutual, asymmetric, and null node pairs
```

```
## $mut  
## [1] 0  
##  
## $asym  
## [1] 174  
##  
## $null  
## [1] 10411
```

```
2*dyad_census(g4)$mut/ecount(g4) # Calculating reciprocity
```

```
## [1] 0
```

Result: The proportion of reciprocated ties (for a directed network) was found

4. Finding Transitivity

```
transitivity(g4, type="global") # g4 is treated as an undirected network
```

```
## [1] 0.03267974
```

```
transitivity(as.undirected(g4, mode="collapse")) # same as above
```

```
## [1] 0.03267974
```

```
transitivity(g4, type="local")
```

```
## [1] 0.00000000      NaN 0.00000000 0.04761905      NaN 0.00000000
## [7] 0.06666667 0.00000000 0.33333333 0.00000000      NaN 0.00000000
## [13]      NaN 0.00000000 0.00000000 0.00000000      NaN      NaN
## [19] 0.33333333 0.00000000 0.00000000 0.00000000 0.00000000      NaN
## [25] 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000      NaN
## [31] 0.00000000 0.00000000 0.02222222      NaN 0.33333333 0.00000000
## [37]      NaN 0.00000000      NaN 0.00000000 0.00000000 0.33333333
## [43] 0.00000000 0.00000000 0.00000000 0.04761905 0.00000000 0.00000000
## [49] 0.00000000 0.00000000 0.00000000 0.06666667 0.00000000 0.00000000
## [55] 0.00000000      NaN      NaN 0.00000000 0.00000000      NaN
## [61] 0.33333333      NaN 0.00000000 0.00000000 0.00000000 0.00000000
## [67]      NaN 0.00000000 0.00000000 0.00000000 0.00000000      NaN
## [73] 0.00000000      NaN 0.10000000 0.00000000 0.00000000      NaN
## [79] 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
## [85] 0.00000000 0.00000000      NaN 0.00000000 0.00000000 0.00000000
## [91] 0.00000000 0.00000000 0.00000000      NaN      NaN 0.00000000
## [97]      NaN      NaN 0.00000000 0.00000000 0.00000000 0.00000000
## [103] 0.16666667 1.00000000 0.00000000 0.00000000      NaN 0.00000000
## [109] 0.00000000      NaN 0.00000000      NaN      NaN 0.00000000
## [115]      NaN      NaN 0.00000000      NaN 0.16666667 1.00000000
## [121] 0.00000000      NaN      NaN      NaN      NaN      NaN
## [127]      NaN      NaN      NaN      NaN      NaN      NaN
## [133]      NaN      NaN 0.00000000      NaN      NaN      NaN
## [139]      NaN      NaN      NaN      NaN      NaN      NaN
## [145]      NaN      NaN
```

```
triad_census(g4) # for directed networks
```

```
## [1] 483478 24153 0 95 109 240 0 0 2 3
## [11] 0 0 0 0 0 0
```

Result: Global transitivity i.e. the ratio of triangles (direction disregarded) to connected triples and local transitivity i.e. ratio of triangles to connected triples each vertex is part of was found.

5. Finding Diameter

```
diameter(g4, directed=F, weights=NA)
```

```
## [1] 12
```

```
diameter(g4, directed=F)
```

```
## [1] 7.066154
```

```
diam <- get_diameter(g4, directed=T)
diam
```

```
## + 16/146 vertices, named, from 45a4802:
## [1] Michelle Lou      Dana      Johnnie  Alvie    Darrel   Noelle   Kermit
## [9] Myrna      Brandyn  Leland   Eliga    Gust     Clarence Marlee  Helena
```

```
class(diam)
```

```
## [1] "igraph.vs"
```

```
as.vector(diam)
```

```
## [1] 58 59 108 106 70 46 33 103 45 7 8 99 89 63 64 69
```

```

vcol <- rep("gray40", vcount(g4))

vcol[diam] <- "gold"

ecol <- rep("gray80", ecount(g4))

ecol[E(g4, path=diam)] <- "orange"

E(g4, path=diam) #finds edges along a path, here 'diam'

```

```

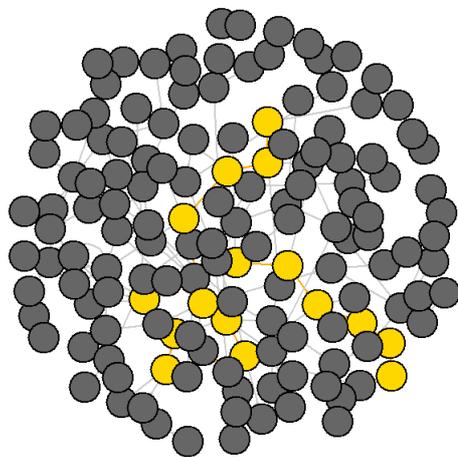
## + 15/175 edges from 45a4802 (vertex names):
## [1] Michelle->Lou      Lou      ->Dana   Dana    ->Johnnie Johnnie ->Alvie
## [5] Alvie   ->Darrel   Darrel  ->Noelle Noelle  ->Kermit  Kermit  ->Myrna
## [9] Myrna  ->Brandyn  Brandyn ->Leland Leland  ->Eliga   Eliga   ->Gust
## [13] Gust   ->Clarence Clarence->Marlee Marlee  ->Helena

```

```

plot(g4, vertex.color=vcol, edge.color=ecol, edge.arrow.size=.05, vertex.label=NA)

```



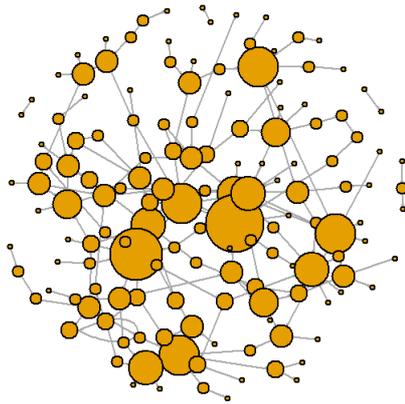
Result: A network diameter, the longest geodesic distance (length of the shortest path between two nodes) in the network was found. In igraph, diameter() returns the distance, while get_diameter() returns the nodes along the first found path of that distance. Edge weights are used by default, unless set to NA. A graph was also plotted with colored nodes along the diameter.

6. Finding Node Degrees

```

deg <- degree(g4, mode="all")
plot(g4, vertex.size=deg*3, edge.arrow.size=.05, vertex.label=NA)

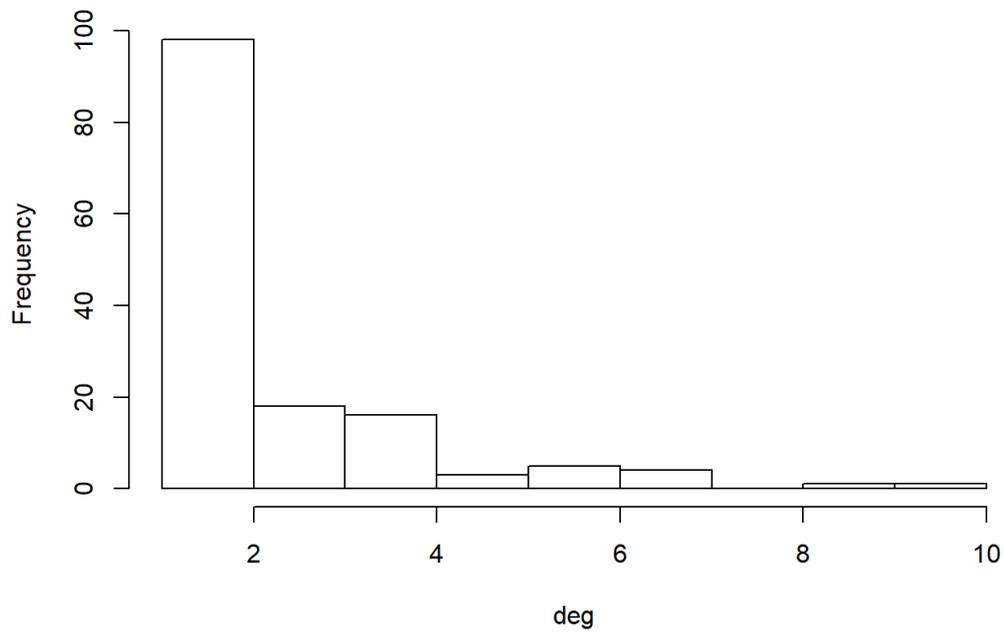
```



7. Visualizing the Degree histogram

```
hist(deg, main="Histogram of node degree")
```

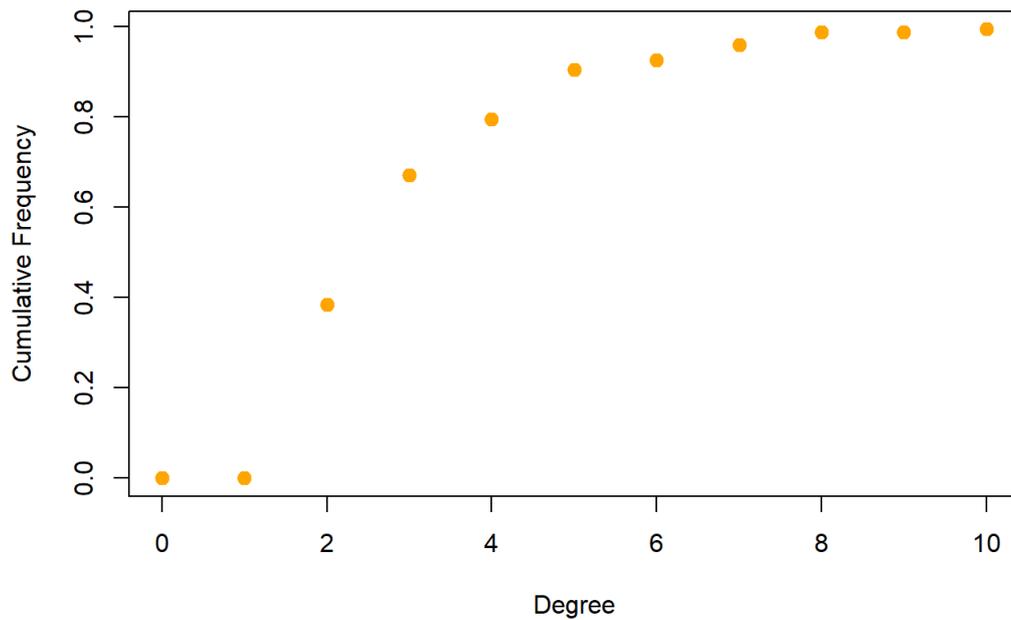
Histogram of node degree



Result: The node degrees were found and a histogram was plotted.

8. Finding degree distribution

```
deg.dist <- degree_distribution(g4, cumulative=T, mode="all")  
plot( x=0:max(deg), y=1-deg.dist, pch=19, cex=1.2, col="orange",  
      xlab="Degree", ylab="Cumulative Frequency")
```



Result: The degree distribution vs cumulative frequency was plotted

Centrality & centralization 9. Finding Degree centrality

```
degree(g4, mode="in")
```

```
## Susann Jeanette Harrison Myron Julie Colten Brandyn Leland
## 1 1 0 4 0 2 3 1
## Catherine Tracy Bee Maureen Annika Hans Spenser Angelo
## 2 2 0 2 0 2 1 1
## Blanche Stanford Hope Almira Thomas Henry Ralph Louvenia
## 0 1 2 3 1 4 2 0
## German Lura Wilma Lauretta Maudie Martha Loyd Mayra
## 4 1 2 0 1 0 0 1
## Noelle Fabian Ira Zenobia Natalie Brooke Hallie Braxton
## 4 1 1 3 1 2 0 2
## Lorraine Alicia Allie Jaylon Myrna Darrel Hermon Wyatt
## 0 2 2 1 1 5 3 5
## Raoul Althea Laurel Leola Giles Merritt Kendrick Travis
## 2 1 1 3 2 3 1 0
## Owen Michelle Lou Stephan Eli Chin Clarence Marlee
## 0 0 2 0 2 1 1 1
## Glenn Reid Jerrold Dale Helena Alvie Elden Chaim
## 4 2 1 0 2 1 2 0
## Augustine Curtiss Pollie Angelita Braden Manford Dwight Zona
## 0 1 2 0 2 0 1 3
## Reggie Dominic Amy Santos Faron Leslie Etha Juanita
## 1 1 1 0 1 4 1 1
## Gust Darin Gerald Andrea Jarrett Newell Eloisa Quintin
## 3 2 2 1 0 1 1 1
## Zechariah Anthony Eliga Obie Jacoby Hardie Kermit Wilda
## 1 1 1 2 2 2 2 0
## Chance Johnnie Kathryn Dana Rachelle Deidra Orpha Rosalind
## 0 3 0 1 2 0 1 1
## Curtis Josue Agatha Hollie Beverly Katarina Leila Lamar
## 1 1 0 0 1 0 2 1
## Judson Matthew Jewel Johnnie Tyson Ari Theo Carmen
## 1 1 1 1 0 0 1 1
## Leonard Desi Austin Janet Carla Vicky Harley Hannah
## 0 0 0 1 0 1 1 0
## Stefani Sebastian Sigmund Urban Tiffany Nell Edd Jennifer
## 1 0 1 0 0 0 0 0
## Angus Sherri
## 0 0
```

```
centr_degree(g4, mode="in", normalized=T)
```

```
## $res
## [1] 1 1 0 4 0 2 3 1 2 2 0 2 0 2 1 1 0 1 2 3 1 4 2 0 4 1 2 0 1 0 0 1 4 1 1 3 1
## [38] 2 0 2 0 2 2 1 1 5 3 5 2 1 1 3 2 3 1 0 0 0 2 0 2 1 1 1 4 2 1 0 2 1 2 0 0 1
## [75] 2 0 2 0 1 3 1 1 1 0 1 4 1 1 3 2 2 1 0 1 1 1 1 1 1 2 2 2 2 0 0 3 0 1 2 0 1
## [112] 1 1 1 0 0 1 0 2 1 1 1 1 1 0 0 1 1 0 0 0 1 0 1 1 0 1 0 1 0 0 0 0 0 0
##
## $centralization
## [1] 0.02621634
##
## $theoretical_max
## [1] 21170
```

10. Finding Closeness centrality

```
closeness(g4, mode="all", weights=NA)
```

```
## Warning in closeness(g4, mode = "all", weights = NA): At centrality.c:
## 2784 :closeness centrality is not well-defined for disconnected graphs
```

```

##      Susann      Jeanette      Harrison      Myron      Julie      Colten
## 4.789272e-05 4.789043e-05 4.701457e-04 5.015045e-04 4.784689e-04 5.115090e-04
##      Brandyn      Leland      Catherine      Tracy      Bee      Maureen
## 5.246590e-04 4.970179e-04 4.982561e-04 4.805382e-04 4.821601e-04 5.157298e-04
##      Annika      Hans      Spencer      Angelo      Blanche      Stanford
## 4.424779e-04 4.705882e-04 4.364906e-04 4.633920e-04 4.756243e-05 4.756243e-05
##      Hope      Almira      Thomas      Henry      Ralph      Louvenia
## 5.053057e-04 4.987531e-04 4.901961e-04 5.133470e-04 4.980080e-04 4.816956e-04
##      German      Lura      Wilma      Laretta      Maudie      Martha
## 5.151984e-04 5.070994e-04 5.285412e-04 4.775549e-04 4.486317e-04 4.800768e-04
##      Loyd      Mayra      Noelle      Fabian      Ira      Zenobia
## 4.697041e-04 5.005005e-04 5.515720e-04 5.133470e-04 5.032713e-04 5.002501e-04
##      Natalie      Brooke      Hallie      Braxton      Lorraine      Alicia
## 5.133470e-04 5.117707e-04 4.805382e-04 5.138746e-04 5.260389e-04 5.091650e-04
##      Allie      Jaylon      Myrna      Darrel      Hermon      Wyatt
## 5.238345e-04 5.040323e-04 5.405405e-04 5.640158e-04 5.035247e-04 5.146680e-04
##      Raoul      Althea      Laurel      Leola      Giles      Merritt
## 4.816956e-04 5.047956e-04 4.885198e-04 5.485464e-04 5.099439e-04 4.889976e-04
##      Kendrick      Travis      Owen      Michelle      Lou      Stephan
## 4.933399e-04 4.800768e-04 4.672897e-04 4.703669e-04 5.017561e-04 5.107252e-04
##      Eli      Chin      Clarence      Marlee      Glenn      Reid
## 5.022602e-04 4.688233e-04 4.878049e-04 4.585053e-04 5.509642e-04 5.202914e-04
##      Jerrold      Dale      Helena      Alvie      Elden      Chaim
## 5.128205e-04 5.144033e-04 4.819277e-04 5.437738e-04 5.211047e-04 4.587156e-04
##      Augustine      Curtiss      Pollie      Angelita      Braden      Manford
## 4.739336e-04 4.454343e-04 5.184033e-04 5.232862e-04 4.897160e-04 4.522840e-04
##      Dwight      Zona      Reggie      Dominic      Amy      Santos
## 5.232862e-04 5.058169e-04 4.926108e-04 5.324814e-04 5.091650e-04 5.050505e-04
##      Faron      Leslie      Etha      Juanita      Gust      Darin
## 5.324814e-04 5.347594e-04 4.987531e-04 5.027652e-04 5.211047e-04 5.288207e-04
##      Gerald      Andrea      Jarrett      Newell      Eloisa      Quintin
## 4.854369e-04 5.144033e-04 4.690432e-04 4.411116e-04 4.899559e-04 4.420866e-04
##      Zechariah      Anthony      Eliga      Obie      Jacoby      Hardie
## 4.171882e-04 5.107252e-04 4.995005e-04 4.625347e-04 5.336179e-04 5.027652e-04
##      Kermit      Wilda      Chance      Johnnie      Kathryn      Dana
## 5.288207e-04 4.955401e-04 5.202914e-04 5.246590e-04 4.122012e-04 5.010020e-04
##      Rachelle      Deidra      Orpha      Rosalind      Curtis      Josue
## 4.972650e-04 4.800768e-04 5.133470e-04 4.714757e-04 4.424779e-04 4.816956e-04
##      Agatha      Hollie      Beverly      Katarina      Leila      Lamar
## 4.812320e-04 4.522840e-04 4.816956e-04 4.522840e-04 4.955401e-04 4.967710e-04
##      Judson      Matthew      Jewel      Johnnie      Tyson      Ari
## 5.030181e-04 4.657662e-04 4.784689e-04 4.422822e-04 4.977601e-04 4.756243e-05
##      Theo      Carmen      Leonard      Desi      Austin      Janet
## 4.756243e-05 4.420866e-04 4.593477e-04 4.734848e-04 5.107252e-04 4.587156e-04
##      Carla      Vicky      Harley      Hannah      Stefani      Sebastian
## 4.756243e-05 4.756243e-05 5.120328e-04 4.868549e-04 4.714757e-04 4.810005e-04
##      Sigmund      Urban      Tiffany      Nell      Edd      Jennifer
## 4.679457e-04 4.672897e-04 4.868549e-04 4.987531e-04 4.746084e-04 4.659832e-04
##      Angus      Sherri
## 4.644682e-04 4.789043e-05

```

```
centr_clo(g4, mode="all", normalized=T)
```

```
## Warning in centr_clo(g4, mode = "all", normalized = T): At centrality.c:
## 2784 :closeness centrality is not well-defined for disconnected graphs
```

```

## $res
## [1] 0.006944444 0.006944112 0.068171133 0.072718154 0.069377990 0.074168798
## [7] 0.076075551 0.072067594 0.072247135 0.069678039 0.069913211 0.074780815
## [13] 0.064159292 0.068235294 0.063291139 0.067191844 0.006896552 0.006896552
## [19] 0.073269328 0.072319202 0.071078431 0.074435318 0.072211155 0.069845857
## [25] 0.074703761 0.073529412 0.076638478 0.069245463 0.065051593 0.069611138
## [31] 0.068107093 0.072572573 0.079977937 0.074435318 0.072974333 0.072536268
## [37] 0.074435318 0.074206755 0.069678039 0.074511819 0.076275644 0.073828921
## [43] 0.075955998 0.073084677 0.078378378 0.081782290 0.073011078 0.074626866
## [49] 0.069845857 0.073195356 0.070835369 0.079539221 0.073941866 0.070904645
## [55] 0.071534287 0.069611138 0.067757009 0.068203198 0.072754641 0.074055158
## [61] 0.072827725 0.067979372 0.070731707 0.066483265 0.079889807 0.075442248
## [67] 0.074358974 0.074588477 0.069879518 0.078847200 0.075560188 0.066513761
## [73] 0.068720379 0.064587973 0.075168481 0.075876504 0.071008815 0.065581185
## [79] 0.075876504 0.073343450 0.071428571 0.077209798 0.073828921 0.073232323
## [85] 0.077209798 0.077540107 0.072319202 0.072900955 0.075560188 0.076679006
## [91] 0.070388350 0.074588477 0.068011257 0.063961182 0.071043606 0.064102564
## [97] 0.060492282 0.074055158 0.072427572 0.067067530 0.077374600 0.072900955
## [103] 0.076679006 0.071853320 0.075442248 0.076075551 0.059769167 0.072645291
## [109] 0.072103431 0.069611138 0.074435318 0.068363979 0.064159292 0.069845857
## [115] 0.069778633 0.065581185 0.069845857 0.065581185 0.071853320 0.072031793
## [121] 0.072937626 0.067536097 0.069377990 0.064130916 0.072175212 0.006896552
## [127] 0.006896552 0.064102564 0.066605420 0.068655303 0.074055158 0.066513761
## [133] 0.006896552 0.006896552 0.074244752 0.070593963 0.068363979 0.069745070
## [139] 0.067852129 0.067757009 0.070593963 0.072319202 0.068818225 0.067567568
## [145] 0.067347887 0.006944112
##
## $centralization
## [1] 0.02888468
##
## $theoretical_max
## [1] 72.24913

```

11. Finding Eigenvector centrality

```
eigen_centrality(g4, directed=T, weights=NA)
```

```

## $vector
##      Susann      Jeanette      Harrison      Myron      Julie      Colten
## 0.000000e+00 0.000000e+00 2.403304e-17 2.718716e-01 2.403304e-17 3.787944e-01
##      Brandyn      Leland      Catherine      Tracy      Bee      Maureen
## 5.953273e-01 4.298795e-01 7.573553e-01 5.468782e-01 2.403304e-17 0.000000e+00
##      Annika      Hans      Spenser      Angelo      Blanche      Stanford
## 2.403304e-17 1.963156e-01 0.000000e+00 0.000000e+00 2.403304e-17 0.000000e+00
##      Hope      Almira      Thomas      Henry      Ralph      Louvenia
## 5.986209e-01 3.250975e-01 2.347494e-01 1.426187e-01 1.029835e-01 2.403304e-17
##      German      Lura      Wilma      Laretta      Maudie      Martha
## 5.822831e-01 0.000000e+00 0.000000e+00 2.403304e-17 0.000000e+00 2.403304e-17
##      Loyd      Mayra      Noelle      Fabian      Ira      Zenobia
## 2.403304e-17 0.000000e+00 8.634570e-01 6.234931e-01 5.214131e-01 3.765069e-01
##      Natalie      Brooke      Hallie      Braxton      Lorraine      Alicia
## 6.234931e-01 4.298795e-01 2.403304e-17 6.234931e-01 2.403304e-17 1.963156e-01
##      Allie      Jaylon      Myrna      Darrel      Hermon      Wyatt
## 2.735234e-01 1.975084e-01 7.220894e-01 9.716309e-01 0.000000e+00 3.787944e-01
##      Raoul      Althea      Laurel      Leola      Giles      Merritt
## 2.735234e-01 0.000000e+00 0.000000e+00 6.234931e-01 0.000000e+00 0.000000e+00
##      Kendrick      Travis      Owen      Michelle      Lou      Stephan
## 0.000000e+00 2.403304e-17 2.403304e-17 2.403304e-17 0.000000e+00 2.403304e-17
##      Eli      Chin      Clarence      Marlee      Glenn      Reid
## 1.023615e-01 0.000000e+00 1.618526e-01 1.168720e-01 0.000000e+00 0.000000e+00
##      Jerrold      Dale      Helena      Alvie      Elden      Chaim
## 0.000000e+00 2.403304e-17 8.439207e-02 0.000000e+00 4.204604e-01 2.403304e-17
##      Augustine      Curtiss      Pollie      Angelita      Braden      Manford
## 2.403304e-17 0.000000e+00 4.502178e-01 2.403304e-17 1.975084e-01 2.403304e-17
##      Dwight      Zona      Reggie      Dominic      Amy      Santos
## 6.234931e-01 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 2.403304e-17
##      Faron      Leslie      Etha      Juanita      Gust      Darin
## 0.000000e+00 5.245810e-01 3.787944e-01 0.000000e+00 2.241448e-01 7.016044e-01
##      Gerald      Andrea      Jarrett      Newell      Eloisa      Quintin
## 1.963156e-01 0.000000e+00 2.403304e-17 0.000000e+00 4.298795e-01 0.000000e+00
##      Zechariah      Anthony      Eliga      Obie      Jacoby      Hardie

```

```

## 0.000000e+00 4.502178e-01 3.104114e-01 0.000000e+00 0.000000e+00 0.000000e+00
## Kermit Wilda Chance Johnnie Kathryn Dana
## 1.000000e+00 2.403304e-17 2.403304e-17 0.000000e+00 2.403304e-17 0.000000e+00
## Rachelle Deidra Orpha Rosalind Curtis Josue
## 0.000000e+00 2.403304e-17 0.000000e+00 0.000000e+00 1.417574e-01 0.000000e+00
## Agatha Hollie Beverly Katarina Leila Lamar
## 2.403304e-17 2.403304e-17 0.000000e+00 2.403304e-17 1.417574e-01 7.220894e-01
## Judson Matthew Jewel Johnnie Tyson Ari
## 0.000000e+00 3.104114e-01 2.735234e-01 0.000000e+00 2.403304e-17 2.403304e-17
## Theo Carmen Leonard Desi Austin Janet
## 0.000000e+00 0.000000e+00 2.403304e-17 2.403304e-17 2.403304e-17 0.000000e+00
## Carla Vicky Harley Hannah Stefani Sebastian
## 2.403304e-17 0.000000e+00 1.029835e-01 2.403304e-17 0.000000e+00 2.403304e-17
## Sigmund Urban Tiffany Nell Edd Jennifer
## 2.241448e-01 2.403304e-17 2.403304e-17 2.403304e-17 2.403304e-17 2.403304e-17
## Angus Sherri
## 2.403304e-17 2.403304e-17
##
## $value
## [1] 1.38487
##
## $options
## $options$bm
## [1] "I"
##
## $options$n
## [1] 146
##
## $options$which
## [1] "LR"
##
## $options$nev
## [1] 1
##
## $options$tol
## [1] 0
##
## $options$ncv
## [1] 0
##
## $options$ldv
## [1] 0
##
## $options$ishift
## [1] 1
##
## $options$maxiter
## [1] 1000
##
## $options$nb
## [1] 1
##
## $options$mode
## [1] 1
##
## $options$start
## [1] 1
##
## $options$sigma
## [1] 0
##
## $options$sigmai
## [1] 0
##
## $options$info
## [1] 0
##
## $options$iter
## [1] 5
##
## $options$nconv
## [1] 1
##

```

```
## $options$numop
## [1] 58
##
## $options$numopb
## [1] 0
##
## $options$numreo
## [1] 24
```

```
centr_eigen(g4, directed=T, normalized=T)
```

```
## $vector
## [1] 0.000000e+00 0.000000e+00 0.000000e+00 2.718716e-01 0.000000e+00
## [6] 3.787944e-01 5.953273e-01 4.298795e-01 7.573553e-01 5.468782e-01
## [11] 0.000000e+00 1.290774e-16 0.000000e+00 1.963156e-01 0.000000e+00
## [16] 1.644882e-17 0.000000e+00 0.000000e+00 5.986209e-01 3.250975e-01
## [21] 2.347494e-01 1.426187e-01 1.029835e-01 0.000000e+00 5.822831e-01
## [26] 1.396862e-17 3.572454e-16 0.000000e+00 0.000000e+00 0.000000e+00
## [31] 0.000000e+00 0.000000e+00 8.634570e-01 6.234931e-01 5.214131e-01
## [36] 3.765069e-01 6.234931e-01 4.298795e-01 0.000000e+00 6.234931e-01
## [41] 0.000000e+00 1.963156e-01 2.735234e-01 1.975084e-01 7.220894e-01
## [46] 9.716309e-01 4.044745e-16 3.787944e-01 2.735234e-01 5.105029e-20
## [51] 0.000000e+00 6.234931e-01 8.370350e-17 1.285601e-16 4.319572e-17
## [56] 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
## [61] 1.023615e-01 5.317325e-18 1.618526e-01 1.168720e-01 6.217226e-17
## [66] 5.259037e-16 3.664746e-16 0.000000e+00 8.439207e-02 3.792494e-16
## [71] 4.204604e-01 0.000000e+00 0.000000e+00 4.183173e-18 4.502178e-01
## [76] 0.000000e+00 1.975084e-01 0.000000e+00 6.234931e-01 1.671994e-17
## [81] 1.662641e-16 2.110827e-16 2.744507e-16 0.000000e+00 0.000000e+00
## [86] 5.245810e-01 3.787944e-01 1.494382e-16 2.241448e-01 7.016044e-01
## [91] 1.963156e-01 1.585067e-17 0.000000e+00 0.000000e+00 4.298795e-01
## [96] 1.255000e-17 4.660628e-17 4.502178e-01 3.104114e-01 1.299551e-15
## [101] 3.032980e-16 2.108926e-16 1.000000e+00 0.000000e+00 0.000000e+00
## [106] 1.797581e-16 0.000000e+00 5.963123e-17 3.154508e-16 0.000000e+00
## [111] 6.864657e-19 6.020268e-16 1.417574e-01 0.000000e+00 0.000000e+00
## [116] 0.000000e+00 0.000000e+00 0.000000e+00 1.417574e-01 7.220894e-01
## [121] 1.789918e-17 3.104114e-01 2.735234e-01 1.710860e-17 0.000000e+00
## [126] 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
## [131] 0.000000e+00 3.373790e-16 0.000000e+00 8.887438e-18 1.029835e-01
## [136] 0.000000e+00 6.302328e-16 0.000000e+00 2.241448e-01 0.000000e+00
## [141] 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
## [146] 0.000000e+00
##
## $value
## [1] 1.38487
##
## $options
## $options$bmat
## [1] "I"
##
## $options$n
## [1] 146
##
## $options$which
## [1] "LR"
##
## $options$nev
## [1] 1
##
## $options$tol
## [1] 0
##
## $options$ncv
## [1] 0
##
## $options$ldv
## [1] 0
##
## $options$ishift
## [1] 1
##
```

```
## $options$maxiter
## [1] 1000
##
## $options$nb
## [1] 1
##
## $options$mode
## [1] 1
##
## $options$start
## [1] 1
##
## $options$sigma
## [1] 0
##
## $options$sigma_i
## [1] 0
##
## $options$info
## [1] 0
##
## $options$iter
## [1] 5
##
## $options$nconv
## [1] 1
##
## $options$numop
## [1] 58
##
## $options$numopb
## [1] 0
##
## $options$numreo
## [1] 24
##
##
## $centralization
## [1] 0.8610265
##
## $theoretical_max
## [1] 145
```

12. Finding Betweenness centrality

```
betweenness(g4, directed=T, weights=NA)
```

```

##      Susann  Jeanette  Harrison  Myron  Julie  Colten  Brandyn
##      1.00000  0.00000  0.00000  808.00000  0.00000  165.00000  1105.50000
##      Leland  Catherine  Tracy  Bee  Maureen  Annika  Hans
##      929.00000  78.00000  0.00000  0.00000  61.66667  0.00000  55.00000
##      Spenser  Angelo  Blanche  Stanford  Hope  Almira  Thomas
##      7.00000  12.00000  0.00000  0.00000  61.00000  451.00000  449.00000
##      Henry  Ralph  Louvenia  German  Lura  Wilma  Lauretta
##      317.50000  0.00000  0.00000  54.00000  56.00000  73.50000  0.00000
##      Maudie  Martha  Loyd  Mayra  Noelle  Fabian  Ira
##      0.00000  0.00000  0.00000  4.00000  1794.50000  0.00000  126.50000
##      Zenobia  Natalie  Brooke  Hallie  Braxton  Lorraine  Alicia
##      32.50000  0.00000  74.50000  0.00000  0.00000  0.00000  663.00000
##      Allie  Jaylon  Myrna  Darrel  Hermon  Wyatt  Raoul
##      394.00000  259.00000  451.50000  871.50000  263.00000  477.33333  0.00000
##      Althea  Laurel  Leola  Giles  Merritt  Kendrick  Travis
##      11.00000  25.00000  666.50000  6.00000  7.00000  15.00000  0.00000
##      Owen  Michelle  Lou  Stephan  Eli  Chin  Clarence
##      0.00000  0.00000  78.33333  0.00000  694.00000  0.00000  107.00000
##      Marlee  Glenn  Reid  Jerrold  Dale  Helena  Alvie
##      54.00000  284.00000  0.00000  0.00000  0.00000  0.00000  374.66667
##      Elden  Chaim  Augustine  Curtiss  Pollie  Angelita  Braden
##      0.00000  0.00000  0.00000  0.00000  488.00000  0.00000  0.00000
##      Manford  Dwight  Zona  Reggie  Dominic  Amy  Santos
##      0.00000  422.00000  44.66667  55.66667  104.00000  60.00000  0.00000
##      Faron  Leslie  Etha  Juanita  Gust  Darin  Gerald
##      41.66667  702.66667  0.00000  5.00000  921.00000  0.00000  0.00000
##      Andrea  Jarrett  Newell  Eloisa  Quintin  Zechariah  Anthony
##      57.00000  0.00000  0.00000  0.00000  1.00000  0.00000  0.00000
##      Eliga  Obie  Jacoby  Hardie  Kermit  Wilda  Chance
##      873.00000  0.00000  133.00000  0.00000  635.00000  0.00000  0.00000
##      Johnnie  Kathryn  Dana  Rachelle  Deidra  Orpha  Rosalind
##      331.66667  0.00000  87.66667  0.00000  0.00000  63.00000  0.00000
##      Curtis  Josue  Agatha  Hollie  Beverly  Katarina  Leila
##      0.00000  15.00000  0.00000  0.00000  15.00000  0.00000  713.00000
##      Lamar  Judson  Matthew  Jewel  Johnie  Tyson  Ari
##      128.50000  41.00000  0.00000  0.00000  0.00000  0.00000  0.00000
##      Theo  Carmen  Leonard  Desi  Austin  Janet  Carla
##      0.00000  0.00000  0.00000  0.00000  0.00000  0.00000  0.00000
##      Vicky  Harley  Hannah  Stefani  Sebastian  Sigmund  Urban
##      0.00000  187.50000  0.00000  0.00000  0.00000  0.00000  0.00000
##      Tiffany  Nell  Edd  Jennifer  Angus  Sherri
##      0.00000  0.00000  0.00000  0.00000  0.00000  0.00000

```

```
edge_betweenness(g4, directed=T, weights=NA)
```

```

##      [1]  2.00000  52.00000  5.00000  982.00000  79.00000  19.00000  2.00000
##      [8] 14.00000  1.00000  63.00000  502.00000  70.00000  2.00000  32.00000
##     [15]  1.00000  15.00000  5.00000  54.00000  82.50000  54.00000  76.50000
##     [22]  3.00000  1.00000  19.00000  333.00000  140.00000  80.00000  75.00000
##     [29] 27.00000 163.00000  8.00000  18.00000  15.00000  52.00000  60.00000
##     [36] 52.00000 416.33333 745.00000  2.00000 109.00000  6.00000  6.00000
##     [43]  1.00000  9.00000  2.00000  1.00000  51.66667  93.00000  63.00000
##     [50] 75.00000  1.00000 473.00000  60.66667  67.00000  42.66667  75.00000
##     [57] 10.00000 811.00000  54.00000  14.66667 108.00000  55.00000  4.00000
##     [64] 54.00000  4.00000  3.00000  70.66667 366.00000  1.00000  54.00000
##     [71] 20.33333  2.00000  54.00000  870.00000  0.00000 120.50000  88.00000
##     [78]  5.00000  54.00000 615.50000  55.00000  95.00000  1.00000  1.00000
##     [85]104.00000  1.00000  8.00000  90.66667  7.00000 174.50000  64.00000
##     [92] 13.00000  56.00000  30.00000  15.00000  63.00000  12.00000  16.00000
##     [99] 16.00000 236.00000  2.00000 291.00000 714.00000 181.50000 141.66667
##    [106] 43.00000  92.00000 161.00000  54.00000  12.00000  77.00000  746.00000
##    [113] 34.50000 273.00000  1.00000  95.00000 405.66667  46.66667 117.00000
##    [120] 384.66667 18.00000  51.00000  4.00000  53.00000  1.00000  1.00000
##    [127]  1.00000 17.00000  5.00000 500.00000  52.00000  2.00000  37.00000
##    [134]  8.00000 697.00000  59.66667  1.00000 201.50000  52.00000  6.00000
##    [141] 88.50000 13.00000 261.50000  593.00000  80.00000  58.00000  54.00000
##    [148] 868.50000  52.00000 475.00000  30.00000  6.00000 504.50000  46.00000
##    [155] 500.50000  68.00000  52.00000  926.00000 179.50000  51.00000  3.00000
##    [162] 126.00000 115.00000 362.50000  15.00000  57.00000  4.00000  17.00000
##    [169] 110.00000  1.00000  81.00000  400.00000  49.50000  52.00000  2.00000

```

```
centr_betw(g4, directed=T, normalized=T)
```

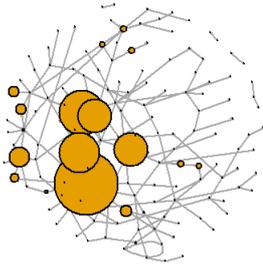
```
## $res
## [1] 1.00000 0.00000 0.00000 808.00000 0.00000 165.00000
## [7] 1105.50000 929.00000 78.00000 0.00000 0.00000 61.66667
## [13] 0.00000 55.00000 7.00000 12.00000 0.00000 0.00000
## [19] 61.00000 451.00000 449.00000 317.50000 0.00000 0.00000
## [25] 54.00000 56.00000 73.50000 0.00000 0.00000 0.00000
## [31] 0.00000 4.00000 1794.50000 0.00000 126.50000 32.50000
## [37] 0.00000 74.50000 0.00000 0.00000 0.00000 663.00000
## [43] 394.00000 259.00000 451.50000 871.50000 263.00000 477.33333
## [49] 0.00000 11.00000 25.00000 666.50000 6.00000 7.00000
## [55] 15.00000 0.00000 0.00000 0.00000 78.33333 0.00000
## [61] 694.00000 0.00000 107.00000 54.00000 284.00000 0.00000
## [67] 0.00000 0.00000 0.00000 374.66667 0.00000 0.00000
## [73] 0.00000 0.00000 488.00000 0.00000 0.00000 0.00000
## [79] 422.00000 44.66667 55.66667 104.00000 60.00000 0.00000
## [85] 41.66667 702.66667 0.00000 5.00000 921.00000 0.00000
## [91] 0.00000 57.00000 0.00000 0.00000 0.00000 1.00000
## [97] 0.00000 0.00000 873.00000 0.00000 133.00000 0.00000
## [103] 635.00000 0.00000 0.00000 331.66667 0.00000 87.66667
## [109] 0.00000 0.00000 63.00000 0.00000 0.00000 15.00000
## [115] 0.00000 0.00000 15.00000 0.00000 713.00000 128.50000
## [121] 41.00000 0.00000 0.00000 0.00000 0.00000 0.00000
## [127] 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
## [133] 0.00000 0.00000 187.50000 0.00000 0.00000 0.00000
## [139] 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
## [145] 0.00000 0.00000
##
## $centralization
## [1] 0.08041122
##
## $theoretical_max
## [1] 3027600
```

Result: 1. Degree i.e. number of ties was found 2. Closeness i.e. centrality based on distance to others in the graph or Inverse of the node's average geodesic distance to others in the network was found 3. Eigenvector i.e. centrality proportional to the sum of connection centralities) or Values of the first eigenvector of the graph matrix was found 4. Betweenness i.e. centrality based on a broker position connecting others or Number of geodesics that pass through the node or the edge was found

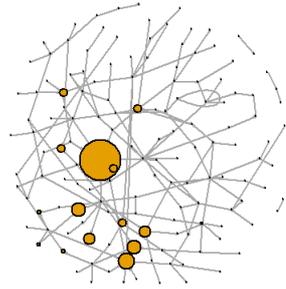
13.Finding Hubs and authorities

```
hs <- hub_score(g4, weights=NA)$vector
as <- authority_score(g4, weights=NA)$vector
par(mfrow=c(1,2))
plot(g4, vertex.label=NA, edge.arrow.size=.05, vertex.size=hs*50, main="Hubs")
plot(g4, vertex.label=NA, edge.arrow.size=.05, vertex.size=as*30, main="Authorities")
```

Hubs



Authorities



Result: Hubs and authorities were found with hubs contain catalogs with a large number of outgoing links; while authorities get many incoming links from hubs, presumably because of their high-quality relevant information.

Conclusion: Visualization of Network and it's node descriptive using Density, Reciprocity, Transitivity, Diameter, Node degrees, Degree distribution, Centrality & centralization, Hubs and authorities were applied